

DAYANANDA SAGAR UNIVERSITY



SCHOOL OF ENGINEERING

(A State Private University under the Karnataka Act No. 20 of 2013)

Approved by UGC & AICTE, New Delhi, Accredited by NAAC with A+ grade

SCHEME FOR BACHELOR OF TECHNOLOGY (B. Tech)

COMPUTER SCIENCE & ENGINEERING (DATA SCIENCE)

Batch 2026-2030



Dayananda Sagar University

School of Engineering

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South Dist. – 562 112

Vision and Mission of DSU

VISION

To be a centre of excellence in education, research & training, innovation & entrepreneurship and to produce citizens with exceptional leadership qualities to serve national and global needs.

MISSION

To achieve our objectives in an environment that enhances creativity, innovation and scholarly pursuits while adhering to our vision.

Vision and Mission of School of Engineering

VISION

Transform lives through excellence in engineering education, research and innovation with an emphasis on sustainability, inclusive technologies and global needs.

MISSION

- Design and deliver contemporary engineering curricula to address regional and global needs while emphasizing ethics, values, integrity and regional relevance.
- Carryout high impact academic research, industry projects and innovation activities with active student engagement to advance science and engineering knowledge and state-of-the-art industry practices.
- Develop regional and national leaders to advance the society and economy.

Vision and Mission of Dept. of CSE (Data Science)

VISION

To produce Engineers for Industry and Society in the field of Computer Science and Engineering (Data Science) by providing excellence in Education, Research and Entrepreneurship with a focus on sustainable solutions to fulfill global needs.



Dayananda Sagar University

School of Engineering

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South Dist. – 562 112

MISSION

Impart quality technical education, critical thinking and sustainable learning practices in the domain of Computer Science and Engineering (Data Science) with ethical values and leadership qualities. Inculcate Interdisciplinary Research and Innovation by establishing Industry-Academia collaboration to solve critical problems.

Prepare graduates to become Ethical Data Science practitioners to contribute in data driven global society.

Definitions / Descriptions

Definition of Credit	
1 Hour Lecture (L) Per Week	01 Credit
1 Hour Tutorial (T) Per Week	01 Credit
1 Hour Practical (P) Per Week	0.5 Credit
1 Hour Project (J) Per Week	0.5 Credit

Course code and Definition	
BSC	Basic Science Courses
ESC	Engineering Science Courses
HSMC	Humanities and Social Sciences including Management Courses
IPCC	Integrated Professional Core Course
PCC	Professional Core Courses
PEC	Professional Elective Courses
OEC	Open Elective Courses
SEC	Skill Enhancement Courses
UHV	Universal Human Value Course
PROJ	Project Work
INT	Internship



Dayananda Sagar University

School of Engineering

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South Dist. – 562 112

Implementation of National Education Policy (NEP) 2020 for the B.Tech students of Batch 2026-2030

The implementation of Curriculum follows NEP 2020 and addresses the following features and categories of courses:

1. Student Centric flexible curriculum.
2. Interdisciplinary Courses,
3. Multi-disciplinary Courses,
4. Ability Enhancement Courses,
5. Skill Enhancement Courses,
6. Value Added Courses,
7. Product Design and Development,
8. Internship (Rural Internship, Industry Internship, Research/Development Internship), and
9. Multiple Exit and Multiple Entry
 - Certificate in Engineering after completion of first year.
 - Diploma in Engineering after completion of second year.
 - Advanced Diploma in Engineering after completion of third year.
 - Degree in Engineering after completion of fourth year
 - Degree in Engineering after completion of fourth year



Dayananda Sagar University

School of Engineering

Devarakagalahalli, Harohalli, Kanakapura Road, Bengaluru South Dist. – 562 112

I Semester(Physics Cycle)											
S.No	Course Code	Course Title	Teaching Hours / Week				Examination				Credits
			Lecture	Tutorial	Practical	Project	Contact Hours	CIE Marks	SEE Marks	Total Marks	
			L	T	P	J					
1.		Linear Algebra and Data Foundations	3	0	0	0	3	60	40	100	3
2.		Engineering Physics (Computer Science Cluster)	3	0	2	0	5	60	40	100	4
3.		Computational Thinking & Programming Using Python	2	0	2	0	4	60	40	100	3
4.		Foundations of AI & Authentic Intelligence	3	0	0	0	3	60	40	100	3
5.		Engineering Science Course - I (Computer Science Cluster) Introduction to Electrical & Electronics Engineering	3	0	0	0	3	60	40	100	3
6.		Engineering Communication & Technical Writing	2	0	0	0	2	100	-	100	2
7.		Biology for Engineers	2	0	0	0	2	100	-	100	2
8.		Kannada Kali / Manasu	1	0	0	0	1	50	-	50	1
9.		Cognitive and Technical Skills (CTS)-I	-	-	-	-	-	-	-	-	P/F
TOTAL			19	0	4	0	23	550	200	750	21



Dayananda Sagar University

School of Engineering

Devarakagalahalli, Harohalli, Kanakapura Road, Bengaluru South Dist. – 562 112

I Semester (Chemistry Cycle)											
S.No	Course Code	Course Title	Teaching Hours / Week				Examination				Credits
			Lecture	Tutorial	Practical	Project	Contact Hours	CIE Marks	SEE Marks	Total Marks	
			L	T	P	J					
1.		Linear Algebra and Data Foundations	3	0	0	0	3	60	40	100	3
2.		Engineering Chemistry (Computer Science Cluster)	2	0	2	0	4	60	40	100	3
3.		Engineering Graphics & Generative Design	2	0	2	0	4	60	40	100	3
4.		Human-Centered Design, Ethics & Values	2	0	0	0	2	60	40	100	2
5.		Engineering Science Course I (Computer Science Cluster) - Introduction to C Programming	2	0	2	0	4	60	40	100	3
6.		Design Studio I: Engineering Problem Solving	0	1	2	0	3	100	-	100	2
7.		Environmental Science	2	0	0	0	2	100	-	100	2
8.		Indian Constitution & Law	1	0	0	0	1	50	-	50	1
9.		Cognitive and Technical Skills (CTS)-I	-	-	-	-	-	-	-	-	P/F
TOTAL			14	1	8	0	23	550	200	750	19



Dayananda Sagar University

School of Engineering

Devarakagalahalli, Harohalli, Kanakapura Road, Bengaluru South Dist. – 562 112

II Semester(Physics Cycle)											
S.No	Course Code	Course Title	Teaching Hours / Week				Examination				Credits
			Lecture	Tutorial	Practical	Project	Contact Hours	CIE Marks	SEE Marks	Total Marks	
			L	T	P	J					
1.		Calculus & Mathematical Modelling	3	0	0	0	3	60	40	100	3
2.		Engineering Physics (Computer Science Cluster)	3	0	2	0	5	60	40	100	4
3.		Computational Thinking & Programming Using Python	2	0	2	0	4	60	40	100	3
4.		Foundations of AI & Authentic Intelligence	3	0	0	0	3	60	40	100	3
5.		Engineering Science Course - I (Computer Science Cluster) Introduction to Electrical & Electronics Engineering	3	0	0	0	3	60	40	100	3
6.		Engineering Communication & Technical Writing	2	0	0	0	2	100	-	100	2
7.		Biology for Engineers	2	0	0	0	2	100	-	100	2
8.		Kannada Kali / Manasu	1	0	0	0	1	50	-	50	1
9.		Cognitive and Technical Skills (CTS)-II	-	-	-	-	-	-	-	-	P/F
TOTAL			19	0	4	0	23	550	200	750	21



Dayananda Sagar University

School of Engineering

Devarakagalahalli, Harohalli, Kanakapura Road, Bengaluru South Dist. – 562 112

II Semester(Chemistry Cycle)											
S.No	Course Code	Course Title	Teaching Hours / Week				Examination				Credits
			Lecture	Tutorial	Practical	Project	Contact Hours	CIE Marks	SEE Marks	Total Marks	
			L	T	P	J					
1.		Calculus & Mathematical Modelling	3	0	0	0	3	60	40	100	3
2.		Engineering Chemistry (Computer Science Cluster)	2	0	2	0	4	60	40	100	3
3.		Engineering Graphics & Generative Design	2	0	2	0	4	60	40	100	3
4.		Human-Centered Design, Ethics & Values	1	1	0	0	2	60	40	100	2
5.		Engineering Science Course II (Computer Science Cluster) - Introduction to C Programming	2	0	2	0	4	60	40	100	3
6.		Design Studio I: Engineering Problem Solving	1	0	0	2	3	100	-	100	2
7.		Environmental Science	2	0	0	0	2	100	-	100	2
8.		Indian Constitution & Law	1	0	0	0	1	50	-	50	1
9.		Cognitive and Technical Skills (CTS)-II	-	-	-	-	-	-	-	-	P/F
TOTAL			14	1	6	2	23	550	200	750	19



Dayananda Sagar University

School of Engineering

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South Dist. – 562 112

SCHEME - B. TECH – 2026-2030 ONWARDS III SEM

SL. No	COURSE CODE	COURSE TITLE	SCHEME OF TEACHING					Duration in Hours	Examination (Maximum Marks)		
			L	T	P	J	C		CIE	SEE	TM
1	26DS2301	Applied Statistics & Stochastic Processes	3	0	0	0	3	3	60	40	100
2	26DS2302	Data Structures	3	0	2	0	4	5	60	40	100
3	26DS2303	Computer Organization and Architecture	3	0	0	0	3	3	60	40	100
4	26DS2304	IoT and Sensor Systems	2	0	2	0	3	3	60	40	100
5	26DS2305	Database Management System	3	0	2	0	4	5	60	40	100
6	26DS2306	Sustainability Engineering & Environmental Social and Governance	2	0	0	0	2	2	60	40	100
7	26DS23XX	Skill Enhancement Course I (Choice of Courses)	1	0	2	0	2	3	100	-	100
8	26DS2307	Cognitive and Technical Skills (CTS)-III	-	-	-	-	-	-	-	-	P/F
Total			18	0	6	0	21	24	460	240	700

L – Lecture, T – Tutorial, P – Practical, J – Project, C – No. of Credits, CIE – Continuous Internal Evaluation, SEE- Semester End Examinations, TM – Total Marks.

SKILL ENHANCEMENT COURSE I		
SL. NO	COURSE CODE	COURSE NAME
1	26DS2308	JAVA Programming
2	26DS2309	BI Tools
3	26DS2310	No Coding/Low Coding Development



Dayananda Sagar University

School of Engineering

Devarakagalahalli, Harohalli, Kanakapura Road, Bengaluru South Dist. – 562 112

SCHEME - B. TECH – 2026-2030 ONWARDS IV SEM

SL. No	COURSE CODE	COURSE TITLE	SCHEME OF TEACHING					Duration in Hours	Examination (Maximum Marks)		
			L	T	P	J	C		CIE	SEE	TM
1	26DS2401	Differential Equations & Optimization	3	0	0	0	3	3	60	40	100
2	26DS2402	Introduction to Data Science	2	0	0	0	2	2	60	40	100
3	26DS2403	Design and Analysis of Algorithms	3	0	2	0	4	5	60	40	100
4	26DS2404	Machine Learning	3	0	2	0	4	5	60	40	100
5	26DS2405	Operating systems	3	0	0	0	3	3	60	40	100
6	26DS2406	Design Studio II-(AI Enabling System)	1	0	0	2	2	2	100	-	100
7	26DS24XX	Skill Enhancement Course II	1	0	2	0	2	3	100	-	100
8	26DS2407	Liberal Studies	2	0	0	0	2	2	60	40	100
Total			18	0	6	2	22	25	560	240	800

L – Lecture, T – Tutorial, P – Practical, J – Project, C – No. of Credits, CIE – Continuous Internal Evaluation, SEE- Semester End Examinations, TM – Total Marks.

SKILL ENHANCEMENT COURSE II		
SL. NO	COURSE CODE	COURSE NAME
1	26DS2408	R Programming
2	26DS2409	MATLAB for Data science
3	26DS2410	Research ethics and Technical writing
4	26DS2411	Full Stack Development



Dayananda Sagar University

School of Engineering

Devarakagalahalli, Harohalli, Kanakapura Road, Bengaluru South Dist. – 562 112

LIBERAL STUDIES		
Sl. No.	Course Code	Course Title
1	26LS0001	Introduction to Japanese Language
2	26LS0002	Introduction to German Language
3	26LS0003	Law for Engineers
4	26LS0004	Introduction to Painting
5	26LS0005	Vedic Mathematics
6	26LS0006	Fundamentals of Critical Thinking
7	26LS0007	Cyber Crimes, Policies & Laws
8	26LS0008	Introduction to Film Studies
9	26LS0009	Practicing Yoga & Meditation
10	26LS0010	Universal Human Values
11	26LS0011	Data Governance

MOOC		
Sl. No.	Course Code	Course Title
1	*CID	Soft Skill Development (NPTEL)
2	*CID	Enhancing Soft Skills and Personality



Dayananda Sagar University

School of Engineering

Devarakagalahalli, Harohalli, Kanakapura Road, Bengaluru South Dist. – 562 112

SCHEME - B. TECH – 2026-2030 ONWARDS V SEM

SL. No	COURSE CODE	COURSE TITLE	SCHEME OF TEACHING					Duration in Hours	Examination (Maximum Marks)		
			L	T	P	J	C		CIE	SEE	TM
1	26DS3501	Responsible AI, Ethics & Policy	2	0	0	0	2	2	60	40	100
2	26DS3502	Multimodal AI for Data Science	3	0	0	0	3	3	60	40	100
3	26DS3503	Big Data Analytics	3	0	2	0	4	5	60	40	100
4	26DS3504	Deep Learning Essentials	3	0	2	0	4	5	60	40	100
5	26DS3505	Theory of Computation with Compiler Design	3	0	0	0	3	3	60	40	100
6	26DS3506	Design Studio III: Innovation, Design Thinking & IP	1	0	0	2	2	2	100	-	100
7	26DS35XX	Professional Elective I / MOOC	3	0	0	0	3	3	60	40	100
8	26DS3507	Cognitive and Technical Skills (CTS)-IV	-	-	-	-	-	-	-	-	P/F
Total			18	0	4	2	21	23	460	240	700

L – Lecture, T – Tutorial, P – Practical, J – Project, C – No. of Credits, CIE – Continuous Internal Evaluation, SEE- Semester End Examinations, TM – Total Marks.



Dayananda Sagar University

School of Engineering

Devarakagalahalli, Harohalli, Kanakapura Road, Bengaluru South Dist. – 562 112

SCHEME - B. TECH – 2026-2030 ONWARDS VI SEM

SL. No	COURSE CODE	COURSE TITLE	SCHEME OF TEACHING					Duration in Hours	Examination (Maximum Marks)		
			L	T	P	J	C		CIE	SEE	TM
1	26DS3601	Sustainable Software Engineering	2	0	0	0	2	2	60	40	100
2	26DS3602	CUDA for High Performance Computing	3	0	2	0	4	5	60	40	100
3	26DS3603	Cloud Data Engineering	3	0	0	0	3	3	60	40	100
4	26DS3604	Computer Networks	3	0	0	0	3	3	60	40	100
5	26DS36XX	Professional Elective II / MOOC	3	0	0	0	3	3	60	40	100
6	26DS3605	Minor Project (CoE-based / Start-up / Research Track)	0	0	0	4	2	-	100	-	100
7	26OEXXXX	Open Elective I	3	0	0	0	3	3	60	40	100
8	26DS3606	Cognitive and Technical Skills (CTS)-V	-	-	-	-	-	-	-	-	P/F
Total			17	0	2	4	20	19	460	240	700

L – Lecture, T – Tutorial, P – Practical, J – Project, C – No. of Credits, CIE – Continuous Internal Evaluation, SEE- Semester End Examinations, TM – Total Marks

OPEN ELECTIVE I		
Sl. No.	COURSE CODE	COURSE NAME
1	23OE0014	Statistical Computing for Data Science
2		Data Analytics for Decision Making
3		Smart Cities & Urban Data Analytics
4		R Programming for Data Analytics



Dayananda Sagar University

School of Engineering

Devarakagalahalli, Harohalli, Kanakapura Road, Bengaluru South Dist. – 562 112

SCHEME - B. TECH – 2026-2030 ONWARDS VII SEM

SL. No	COURSE CODE	COURSE TITLE	SCHEME OF TEACHING					Duration in Hours	Examination (Maximum Marks)		
			L	T	P	J	C		CIE	SEE	TM
1	26DS4701	Entrepreneurial Finance & Deep Tech Start-ups	2	0	0	0	2	2	100	-	100
2	26DS4702	Principles of DEVOPS & MLOPS	3	0	2	0	4	5	60	40	100
3	26DS47XX	Professional Elective III	3	0	0	0	3	3	60	40	100
4	26DS47XX	Professional Elective IV	3	0	0	0	3	3	60	40	100
5	26OEXXXX	Open Elective II	3	0	0	0	3	3	60	40	100
6	26DS4703	Capstone Project I	0	0	0	8	4	-	60	40	100
Total			14	0	2	8	19	16	400	200	600

L – Lecture, T – Tutorial, P – Practical, J – Project, C – No. of Credits, CIE – Continuous Internal Evaluation, SEE- Semester End Examinations, TM – Total Marks

Open Elective Course- II		
Sl. No	COURSE CODE	COURSE NAME
1.	23OE0015	Data Exploration Using R
2.		MATLAB for Data Science
3.		Social Media Analytics
4.		Industrial Data Analytics



Dayananda Sagar University

School of Engineering

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South Dist. – 562 112

SCHEME - B. TECH – 2026-2030 ONWARDS VIII SEM

SL. No	COURSE CODE	COURSE TITLE	SCHEME OF TEACHING					Duration in Hours	Examination (Maximum Marks)		
			L	T	P	J	C		CIE	SEE	TM
1	26DS48XX	Professional Elective V / MOOC	3	0	0	0	3	3	60	40	100
2	26DS4801	Leadership, Policy & Authentic Intelligence	2	0	0	0	2	2	-	-	100
3	26DS4802	Internship (Industry / Research / CoE based)	0	0	0	4	2	-	100	-	100
4	26DS4803	Capstone Project Phase II	0	0	0	20	10	-	60	40	100
Total			5	0	0	24	17	05	220	80	400

L – Lecture, T – Tutorial, P – Practical, J – Project, C – No. of Credits, CIE – Continuous Internal Evaluation, SEE- Semester End Examinations, TM – Total Marks

I -21 II -19 III -21 IV – 22 V -21 VI -20 VII – 19 VIII – 17

NOTE: Total Credits (I-Sem to VIII Sem) = 160credits.



Dayananda Sagar University

School of Engineering

Devarakagalahalli, Harohalli, Kanakapura Road, Bengaluru South Dist. – 562 112

NOTE: Professional elective courses Domain-wise

Sl.No	CSE (Data Science) Professional Elective Courses										
	Domain		PEC-I		PEC-II		PEC-III		PEC-IV		PEC-V
			5 th Semester		6 th Semester				7 th Semester		8 th Semester
1	GEN AI (DLI Mapped)	26DS3512	NLP	26DS3607	Generative AI and LLM	26DS4704	Agentic AI	26DS4709	Multimodal RAG Systems		26DS4804 Explainable AI
2	Data Science Applications (COE Mapped)	26DS3513	Optimization Techniques for Data Science	26DS3608	Data-Driven Recommendation Systems	26DS4705	Immersive Data Science	26DS4710	AI for Cybersecurity Analytics		26DS4805 Autonomous System
3	AI+X	26DS3514	AI for Healthcare	26DS3609	AI for Agriculture	26DS4706	AI for FinTech & Risk Analytics	26DS4711	Geo-Spatial AI		26DS4806 Supply Chain AI
4	Industrial AI (DLI Mapped)	26DS3515	AI for Industry 5.0	26DS3610	Industrial Digital Twin Technology	26DS4707	Embedded AI System & Edge Deployment	26DS4712	Scalable Enterprise AI		26DS4807 Enterprise Deployment
5	Data Security	26DS3516	Cryptography and Network Security	26DS3611	Cloud Data Security	26DS4708	Information security	26DS4713	Blockchain Technology		26DS4808 Adversarial Machine learning



Dayananda Sagar University

School of Engineering

Devarakagalahalli, Harohalli, Kanakapura Road, Bengaluru South Dist. – 562 112

Sl.No	MINORS (18 CREDITS)			
	Semester	Course Title	DLI Mapping	Credits
1	4th	AI-Driven Data Engineering	3	3
2	5th	GPU Accelerated Data Mining	3	3
3	6th	High-Performance Big Data Analytics	3	3
4	7th	Deep Learning for AI	3	3
5	8th	Data Science Project DGX Cluster	6	6
Total			18	18

Sl.No	HONORS (18 CREDITS)			
	Semester	Course Title	DLI Mapping	Credits
1	5th	Applied Deep Learning for Computer Vision Systems	3	3
2	6th	Accelerated Data Processing & Engineering	3	3
3	6th	Deep Learning for Autonomous & Video Analytics Systems	3	3
4	7th	Scalable AI Model Training & Optimization	3	3
5	7th	Conversational AI Systems & Intelligent Assistants	3	3
6	8th	Cybersecurity Analytics using AI & Deep Learning	3	3
Total			18	18

Dayananda Sagar University

School of Engineering

Devarakagalahalli, Harohalli, Kanakapura Road, Bengaluru South Dist. – 562 112

LINEAR ALGEBRA & DATA FOUNDATIONS

[As per choice based Credit system (CBCS) Scheme]

Course Code: 26XXXXXXX	Semester: I
Programs Offered:	B Tech
Prerequisites: Basic Maths	
Course Type: Basic Sciences	Hours/Week: 03
Total Hours: 30+15	Credits: 03
L – T – P – J: 3-0-0-0	

Course Vision

To develop the ability to analyze engineering problems using linear algebra, probability, and data analytics concepts, and to apply these mathematical foundations in building intelligent systems and AI-driven solutions responsibly

Course Objectives

This course will enable students to:

1. **Understand** the fundamental principles of linear algebra and vector spaces used in computational systems.
2. **Apply** matrix operations and linear transformations for modelling engineering and AI problems.
3. **Analyze** uncertainty and randomness using probability theory and statistical methods.
4. **Apply** data preprocessing and regression techniques for predictive modelling.
5. **Use** mathematical tools for data-driven decision making in intelligent engineering systems.



Course Outcomes (CO)

CO	Outcome	Bloom's Taxonomy Level
CO1	Apply matrix operations and solve systems of linear equations to perform computations used in AI models such as neural networks.	L3
CO2	Analyze vector spaces, basis, and linear transformations to understand feature representation and dimensionality reduction techniques in AI.	L4
CO3	Compute eigenvalues and eigenvectors and apply them in AI techniques like Principal Component Analysis (PCA), image compression, and pattern recognition.	L3
CO4	Apply probability laws and random variable concepts to develop probabilistic AI models such as Naïve Bayes classifiers and predictive systems.	L3
CO5	Perform data preprocessing and implement simple regression models to build foundational supervised learning models for AI-based decision-making.	L5

Teaching-Learning Process (General Instructions)

Following are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes:

1. Lecture-based conceptual teaching with illustrative examples.
2. Problem-solving sessions after each topic.
3. Use of MATLAB/Python for computational verification.
4. Group discussions and peer learning.
5. Activity-based and application-oriented learning.
6. Continuous internal assessment through quizzes and assignments.
7. Real-world case studies in probability and regression.



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakagalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Course Modules

Module 1: Systems of Linear Equations [09 Hours]

Topics: System of Linear equations, Row reduction and echelon form, Rank of a matrix by row echelon form, Gauss elimination method, Inverse of a matrix using Gauss-Jordan method, LU decomposition.

Applications: Engineering system modeling, Circuit analysis, Computer graphics transformations, and Robotics control systems.

Module 2: Vector Spaces [07 Hours]

Topics: Vector spaces, Subspaces, Linear independence, Span, Bases and dimensions, Finite dimensional vector spaces.

Applications: Signal processing, Computer graphics, Feature representation in data science, Image compression techniques.

Module 3: Linear Transformations and Eigen Value [10 Hours]

Topics: Linear transformations, Matrices of linear transformations, Vector space of linear transformations, Inner product spaces, Orthogonal vectors, Projections, Gram-Schmidt orthogonalization process, Eigenvalues and eigenvectors and interpretation, Diagonalization.

Applications: Image processing, Computer vision systems, Signal filtering, Structural engineering analysis.

Module 4: Probability and Random Variables [09 Hours]

Topics: Definition of random variable, Random experiments, Algebra of events, Definition of probability (Classical, Empirical, Axiomatic), Laws of probability, Conditional probability, Baye's theorem, Independence of events, Random variables (discrete and continuous), Expectation (Mean), Variance, Standard deviation, Sum of independent random variables, Applications of probability to business and economics.

Applications: Risk analysis, Financial forecasting, Decision making under uncertainty, Reliability engineering.

Module 5: Data Analytics and Regression [09 Hours]

Topics: : Introduction to data, Data types and collection, Data preprocessing, Features of vectors, Multidimensional data representation, Dataset dimensionality, Simple regression models with one independent variable, Assumptions of regression models, Estimation of parameters, Standard error of estimator.



Applications Business analytics, Demand forecasting, Market trend analysis, Quality control systems.

LAB EXERCISES

S.No	Exercise Title	Aim / Objective	Tools	Hrs	CO
MODULE I: Exercises 1–3: Systems of Linear Equations (03 Hours)					
1	Basic of Linear Algebra and Solving Methods	Solve systems of linear equations using Gauss Elimination.	MATLAB/ PYTHON	1 h	C01
2	Matrix Inverse & LU Decomposition	Compute matrix inverse and perform LU decomposition	MATLAB/ PYTHON	1 h	C01
3	Vectorized Matrix Computation	Implement matrix multiplication to simulate a single-layer neural network forward pass. Perform batch matrix operations to understand vectorized AI computation.	MATLAB/ PYTHON	1 h	C01
MODULE II: Exercises 4–6: Vector Spaces (03 Hours)					
4	Linear Independence & Basis	Verify linear independence of feature vectors and Compute basis and dimension of datasets	MATLAB/ PYTHON	1 h	C02
5	Vector Space Operations	Implement vector space operations (span, subspace verification).	MATLAB/ PYTHON	1 h	C02
6	Feature Space Visualization	Represent datasets as vectors in multidimensional space and Visualize 2D and 3D feature spaces.	MATLAB/ PYTHON	1 h	C02



**MODULE III: Exercises 7–9: Linear Transformations and Eigen Value
(06 Hours)**

7	Eigen Computation	Compute eigenvalues and eigenvectors.	MATLAB/ PYTHON	1 h	C03
8	PCA Implementation	Implement Principal Component Analysis (PCA) from scratch and Apply PCA on real dataset.	MATLAB/ PYTHON	1 h	C03
9	Orthogonalization & Projection	Implement Gram-Schmidt Orthogonalization. Perform vector projection and interpret results.	MATLAB/ PYTHON	1 h	C03

**MODULE IV: Exercises 10–12: Probability and Random Variables
(06 Hours)**

10	Statistical Measures	Simulate random experiments. Compute mean, variance, and standard deviation of datasets.	MATLAB/ PYTHON	1 h	C04
11	Naïve Bayes Classifier	Implement Naïve Bayes classifier.	MATLAB/ PYTHON	1 h	C04
12	Random Experiments	Generate discrete and continuous random variables.	MATLAB/ PYTHON	1 h	C04

**MODULE V: Exercises 13–15: Transaction Management
(06 Hours)**

13	Data Preprocessing	Load and preprocess dataset. Handle missing values and normalize data.	MATLAB/ PYTHON	1 h	C05
14	Linear Regression	Implement Simple Linear Regression from scratch.	MATLAB/ PYTHON	1 h	C05
15	Error Analysis & Visualization	Compute standard error and residual analysis. Visualize regression results.	MATLAB/ PYTHON	1 h	C05



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakagalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

- **AI Integration**
 - Matrix computations used in neural networks
 - Linear systems used in machine learning optimization
 - Feature vector representation in machine learning
 - Data compression and noise reduction
 - Data preprocessing for machine learning and AI models
- **Authentic Intelligence Reflection:**
 - 1. Importance of verifying computational outputs in AI models
 - 2. Responsible interpretation of data representations
 - 3. Interpreting AI dimensionality reduction results responsibly
 - 4. Understanding uncertainty in AI predictions
 - 5. Importance of transparent data processing and model interpretation

RECOMMENDED TOOLS

1. Python
2. MATLAB

SUGGESTED READING

TEXT BOOKS:

1. Anton, Howard, and Chris Rorres. Elementary linear algebra: applications version. John Wiley & Sons, 2013.
2. Lay, David C. Linear algebra and its applications. Pearson Education India, 2003.
3. Erwin Kreyszig, Advanced Engineering Mathematics, 10th Edition, Wiley India, 2015.
4. Strang, Gilbert. Introduction to linear algebra. Wellesley-Cambridge Press, 2022.
5. Ross, Sheldon M. A first course in probability. Harlow, UK: Pearson, 2020.
6. Montgomery, Douglas C., and George C. Runger. Applied statistics and probability for engineers. John Wiley & Sons, 2010.



REFERENCE BOOKS:

1. Bruce, Peter, Andrew Bruce, and Peter Gedeck. Practical statistics for data scientists: 50+ essential concepts using R and Python. O'Reilly Media, 2020.
2. Feller, William. An introduction to probability theory and its applications, Volume 2. Vol. 2. John Wiley & Sons, 1991.
3. James, Gareth, et al. An introduction to statistical learning: with applications in R. Vol. 103. New York: Springer, 2013.
4. Klein, Philip N. "Coding the matrix: linear algebra through applications to computer science." Newtonian Press (2013)..

Mapping Levels of COs to POs / PSOs

Table: Mapping Levels of COs to POs / PSOs													
COs	Program Outcomes (POs)											PSOs	
	1	2	3	4	5	6	7	8	9	10	11	1	2
CO-1	3	3	2	2	1	1			1	2		3	2
CO-2	3	3	2	2	1				1	2		3	2
CO-3	3	3	3	2	2	1			1	2		3	3
CO-4	3	2	2	3	3	1			1	2	1	2	3
CO-5	3	3	3	3	3	2	1		2	2	1	3	3

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Engineering Physics for Computer Science Cluster

Course Code: 26ENXXXX	Semester: I/II
Programs Offered:	B.Tech. Computer Science Cluster
Prerequisites: Basic knowledge of classical mechanics, wave theory, and limitations of classical physics to understand the need for quantum mechanics. Introductory understanding of modern physics concepts to support topics like semiconductors, lasers, devices, and Nano science.	
Course Type: Basic Science	Hours/Week: 05 (3 T+2 P)
Total Hours: 45+30	Credits: 04
L – T – P – J: 3 – 0 – 2 – 0	

Course Vision

To develop a strong foundation in engineering physics concepts while integrating AI and computational tools for modern problem-solving. The course aims to bridge fundamental physics with emerging technologies such as quantum computing, semiconductor devices, and data-driven modelling. It prepares students to apply physics principles in designing intelligent systems and next-generation engineering solutions.

Course Objectives

Students will:

1. Understand fundamental principles of quantum mechanics and apply them to engineering problems.
2. Explore quantum information and quantum computing concepts using quantum simulation tools.
3. Learn principles of quantum optics, lasers, and optical fibre systems and their role in data transmission.
4. Comprehend band structure of solids and analyse semiconductor device applications foundational to AI chip design. Understand thin-film fabrication and nanomaterial synthesis with applications in engineering, medicine, and AI sensors.



Use AI/ML tools (Python, MATLAB, PhET) to simulate physical systems and interpret experimental data. Understand how physics based LLM work.

Course Outcomes (CO)

CO	Outcome Description	Bloom's Taxonomy Level
CO1	Explain fundamental postulates of quantum mechanics and apply them to simple physical systems relevant to engineering.	L3
CO2	Analyze basic concepts of quantum information, qubits, and evaluate quantum processing and computational advantages.	L3
CO3	Illustrate working principles of lasers and optical fibres, and apply them to modern communication and engineering systems.	L3
CO4	Interpret band theory of solids and characterise electrical behavior of semiconductors and their device applications. Compare thin-film fabrication techniques, explain nanoscale synthesis, and evaluate applications in engineering and medicine.	L3
CO5	Apply fundamental physics-based concepts such as probability distributions, energy landscapes, and scaling laws to understand and analyse machine learning models and AI systems	L4

Broad Course Description

The course provides a comprehensive introduction to fundamental principles of engineering physics, including quantum mechanics, quantum information, optics, semiconductor physics, and nanotechnology. It emphasizes the application of these concepts to modern engineering systems, particularly in computing and electronics. With an integrated approach, the course incorporates AI and computational tools such as Python, simulation platforms, and machine learning techniques to model, analyse, and interpret physical phenomena. This enables students to connect theoretical physics with real-world applications in areas like quantum computing, AI hardware, optical communication, and advanced materials. Overall, the course is designed to develop analytical thinking, problem-solving skills, and interdisciplinary understanding, preparing students to contribute to emerging technologies at the intersection of physics and artificial intelligence.



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakagalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Course Modules

Module 1 — Quantum Mechanics (09 Hours)

Failure of Classical Mechanics, Planck's hypothesis, Wave-Particle Duality, Uncertainty Principle, Wave function, Eigen functions and Eigen values, Schrodinger Time-Independent Wave Equation in 1D (Derivation), Particle in a Potential Well (Derivation), Numerical.

Module 2 — Quantum Information & Computing (09 Hours)

Introduction to Quantum Information, Qubit, Bloch Sphere, Superposition, Entanglement, Quantum Gates, Quantum Measurement, Quantum Parallelism, Quantum Supremacy. Engineering Applications: Quantum Key Distribution (QKD) and Quantum Sensors.

Module 3 — Quantum Optics & Fibre Communication (09 Hours)

Quantum Optics: Absorption, spontaneous and stimulated emission, Einstein coefficients, pumping, population inversion, resonators, laser characteristics, Nd:YAG and semiconductor lasers

Engineering applications: Material Processing & Manufacturing, Medical Engineering and Defense & Aerospace.

Fibre Communication: Optical fibre basics, numerical aperture, acceptance angle, step-index and graded-index fibres, fibre losses, applications.

Module 4 — Semiconductor Physics, Devices & Nano science (10 Hours)

Semiconductor and Device Physics:

Classification of solids, band structure, Fermi level (intrinsic and extrinsic), Fermi-Dirac distribution, electron concentration in conduction band (Derivation), intrinsic carrier concentration, conductivity, Numerical. Devices: Zener diode, BJT and LED.

Nanoscience & Technology:

Nanomaterials, classification, size-dependent properties, quantum confinement, top-down and bottom-up approaches (ball milling, photolithography),

Engineering applications: Nano-engineered transistors, quantum computing components, and carbon nanotubes

Module 5 — Physics for AI & ML (08 Hours)

Statistical Mechanics and Distribution Functions: Microscopic and Macroscopic Descriptions, Microstates and Macrostates, Probability and entropy, Equal A Priori Probability Postulate, Boltzmann Entropy Relation, Probability Distribution, Boltzmann Distribution, Partition Function, Ensembles, Average Values, Maxwell-Boltzmann distribution (Softmax function) and Bose-Einstein distribution (Occupancy constraints, Resource allocation models, Quantum-inspired algorithms).



AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

- **Module 1:** Students utilise Python and PhET simulations to solve the 1D time-independent Schrodinger equation numerically, visualise probability densities for a particle in a box, and verify the Heisenberg Uncertainty Principle computationally using Gaussian wave packets.
- **Module 2:** Students use IBM Qiskit to implement and simulate basic quantum gates (Hadamard, CNOT, Pauli-X), visualise qubit states on the Bloch sphere, and model simple quantum circuits. AI tools (Claude AI, Google Colab) are used to explore QKD protocols and analyse quantum parallelism.
- **Module 3:** Python-based simulation platforms (PhET, Colab) are used to model population inversion, simulate laser rate equations, and compare Nd:YAG and semiconductor laser characteristics. Students use AI tools to simulate step-index vs. graded-index optical fibre light propagation and analyse fibre attenuation parameters.
- **Module 4:** AI-assisted tools (Python, Colab) are used for semiconductor parameter optimisation, simulation of band structure and Fermi level variation with doping, and device characteristics. AI models are employed to predict size-dependent properties of nanomaterials, optimize fabrication parameters, and analyse quantum confinement effects.
- **Module 5:** Students use Python/Colab to implement Maxwell-Boltzmann, Bose-Einstein, and Fermi-Dirac distribution functions. The softmax function is implemented and compared with the Maxwell-Boltzmann distribution. AI tools are used to model energy landscapes, partition functions, and explore quantum-inspired resource allocation algorithms.

Engineering Physics Laboratory

List of Experiments

No.	Experiment	Module Link
1	I-V Characteristics of Zener Diode	Module IV
2	Four Probe Technique — Resistivity	Module IV
3	Dielectric Constant Determination	Module IV
4	Fibre Optic System — NA, Acceptance Angle & Losses	Module III
5	Band Gap Energy of Intrinsic Semiconductor	Module IV
6	Diffraction Grating — Wavelength of Laser Light	Module III
7	Planck's Constant using LED	Module IV
8	LCR Series & Parallel Resonance	Module IV
9	Transistor Characteristics (NPN-CE)	Module IV
10	Hall Effect — Concentration, Mobility & Conductivity	Module IV
11	Realization of the hadamard gate and its geometric representation on the Bloch sphere via quantum-circuit computing platform	Module II
12	Realization of the c-not gate and its bloch sphere visualization using	Module II



Assessment Scheme

Professional Core Courses (Theory)

Criteria	Weightage of Marks
Lab Reports (experiments)	15
Class Tests / Quizzes	10
Mid-Semester Examination	30
AI Mini-Project (Group)	5
End-Semester Examination	40
Total	100 marks

Primary Textbooks

1. **Shatendra Sharma & Jyotsna Sharma** — Engineering Physics (2019), Pearson, Noida, India.
2. **Nielsen, M.A. & Chuang, I.L.** — Quantum Computation and Quantum Information, Cambridge University Press, 10th Anniversary Ed.
3. **Kasap, S.O.** — Principles of Electronic Materials and Devices, 4th Ed., McGraw-Hill.

Additional / Reference Resources

Supplementary Textbooks:

- Shankar, R. — Principles of Quantum Mechanics, 2nd Ed., Springer.
- Thyagarajan, K. & Ghatak, A.K. — Lasers: Fundamentals and Applications, 2nd Ed., Springer India.
- Keiser, G. — Optical Fiber Communications, 5th Ed., McGraw-Hill.
- Beiser, A. — Concepts of Modern Physics, 6th Ed., McGraw-Hill.
- Engineering Physics Laboratory Manual, Dept. of Physics, DSU — Internal Publication, 2025.

Online & Simulation Resources:

- IBM Qiskit Textbook — Free at qiskit.org/learn
- NPTEL Lectures: Quantum Computing, Semiconductor Physics & Devices — nptel.ac.in
- Google Colab Python Notebooks — colab.research.google.com
- PhET Simulations — phet.colorado.edu
- Materials Project Database — materialsproject.org

- Goodfellow et al. — Deep Learning — Free at deeplearningbook.org

NVIDIA Deep Learning Institute (DLI) Recommended Paths:

- Fundamentals of Deep Learning — [learn.nvidia.com](https://www.learn.nvidia.com) (Course: DLI+C-FX-01+V3)
- Physics-Informed Machine Learning — [learn.nvidia.com](https://www.learn.nvidia.com) (Course: DLI+S-OV-12+V1)
- Simulation and Modelling with NVIDIA Modulus (PhysicsNeMo) — [learn.nvidia.com](https://www.learn.nvidia.com) (Course: DLI+S-OV-06+V1)

Note: DLI content is recommended for self-paced supplementary study. Instructors are encouraged to map each DLI module to the corresponding lecture topic before student enrolment to ensure curriculum alignment.

Mapping Levels of COs to POs / PSOs													
CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
C01	3	2	1	2	1							3	1
C02	3	3	2	2	2	1			1			2	3
C03	3	2	2	1	2	2	1					3	2
C04	3	3	2	3	2	2	2					3	2
C05	3	3	3	2	3	2	1		2	1		2	3
3: Substantial (High) 2: Moderate (Medium) 1: Low													



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Engineering Physics for Electronics Cluster

Course Code: 26ENXXXX	Semester: I/II
Programs Offered:	B.Tech. Electronics Cluster
Prerequisites: A solid foundation in physics and mathematics, along with basic computer proficiency, is expected. Familiarity with programming and data analysis tools is beneficial, but not essential.	
Course Type: Basic Science	Hours/Week: 05 (3 T+ 2 P)
Total Hours: 45+30	Credits: 04
L – T – P – S: 3 – 0 – 2 – 0	

Course Vision

The course aims to build a strong foundation in engineering physics integrated with Artificial Intelligence, enabling students to analyze, model, and solve real-world problems in Electronics and Communication Engineering using fundamental principles of Physics, computational tools, and data-driven techniques. It prepares learners for emerging technologies such as Quantum technology and computing, Novel and smart materials, semiconductor devices, nanotechnology, and intelligent physical systems, while fostering interdisciplinary thinking, innovation, and effective problem-solving skills.

Course Objectives

Students will:

1. Understand fundamental principles of quantum mechanics and apply them to engineering problems.
2. Introduce and Explore quantum information concepts and their applications in quantum computing and AI hardware.
3. Learn principles involving working of optical fibres, optical sources and detectors, and optical fibre systems and their role in AI data transmission.
4. Comprehend band structure of solids and analyse semiconductor device applications foundational to AI chip design. Understand thin-film fabrication and nanomaterial synthesis with applications in engineering, medicine, and AI sensors.



5. Learn various types of novel materials and its discovery at accelerated rate using AI simulation tools for the implementation of next-generation electronics, quantum computing, and neuromorphic hardware. Use AI/ML tools (Python, Qiskit, MATLAB, PhET) to simulate physical systems and interpret experimental data.

Course Outcomes (CO)

CO	Outcome Description	Bloom's Taxonomy Level
CO1	Explain fundamental postulates of quantum mechanics and apply them to simple physical systems relevant to engineering. Analyze basic concepts of quantum information, qubits.	L3
CO2	Interpret band theory of solids, Concept of Fermi energy level, density of states, carrier concentration and Hall effect and derivation of Hall Coefficient. Understanding various Semiconductor electronic devices and characterise electrical behaviour of semiconductors and their device applications.	L3
CO3	Explore various Novel materials and their key properties for the implementation of next-generation novel electronics/optoelectronic and spintronic devices. Compare thin-film fabrication techniques, explain Nano scale synthesis, and evaluate applications in engineering and medicine.	L3
CO4	Illustrate working principles of optical fibres, optical sources and detectors and apply them to modern communication and engineering systems.	L2
CO5	Ability to design and apply AI-enabled physical systems by integrating simulation, modelling, and real-time deployment tools to address real-world engineering challenges.	L4

Broad Course Description

Engineering Physics – Electronics and Communication Engineering Cluster is a foundational Basic Science course offered in the first year of the B. Tech programme, designed to build a strong scientific base for engineering students. This course provides a comprehensive foundation in Quantum Mechanics, semiconductor physics, and advanced electronic materials, integrating classical concepts with emerging technologies. It begins with quantum principles such as wave functions, the Heisenberg Uncertainty Principle, and the Schrodinger Equation,



alongside fundamentals of Quantum Computing. The course explores semiconductor behaviour, band theory, and carrier concentration and the basic semiconductor devices construction and working followed by novel engineering materials. It introduces novel materials such as graphene, perovskites, and topological insulators, highlighting their role in modern devices. Learners study optoelectronic components, optical fiber communication, thin-film technologies, and nanotechnology. The course also emphasizes AI integration with computational methods to enhance learning regarding semiconductor device modelling, process optimization, and lithography pattern analysis. Overall, it equips students with interdisciplinary knowledge for next-generation electronics, quantum technologies, and intelligent semiconductor system design, strongly assists the students in developing analytical thinking, solving ability and computational skills needed for innovation and research.

Course Modules

Module 1 — Quantum Mechanics and Introduction of Quantum Computing (07 Hours)

Foundations of quantum theory, Wave function and its properties, de-Broglie hypothesis, Heisenberg Uncertainty principle. One-dimensional time independent Schrodinger wave equation, Eigen values and Eigen functions. Applications: one-dimensional motion of an electron in a potential well. Basics of Quantum computing - Concepts of Superposition, Entanglement, Interference and Qubit.

Module 2 — Semiconductor Physics and Devices (11 Hours)

Semiconductor Physics: Classification of Solids, Band structure, Fermi level in intrinsic and extrinsic semiconductors, Density of energy states in conduction and valence bands of a semiconductor (Mention the expression), Expression for concentration of electrons in conduction band (Derivation), Intrinsic carrier concentration, Conductivity of semiconductors, Hall effect, Derivation of Hall Coefficient, Numerical.

Semiconducting devices for optoelectronic applications: Principle and working of Zener diode, Light Emitting Diode, Photodiode, Solar cell and BJT characteristics.

Module 3 — Novel Electronic Materials, Devices and Fabrication Techniques (10 Hours)

Novel materials: Two-Dimensional (2D) Materials-Graphene and Transition Metal Dichalcogenides (TMDs – e.g., MoS₂, WS₂, Hexagonal Boron Nitride (h-BN); Growth of 2D materials using Chemical Vapor deposition, Topological Insulators (Materials: Bi₂Se₃, Bi₂Te₃) Perovskite Materials, Engineering Applications: Perovskite Solar Cells (PSCs) & Photodetectors.

Thin films technology: Introduction to thin films-Advantages of thin films over bulk materials, Thin film deposition processes - Physical vapor deposition (Thermal evaporation technique and sputtering technique) process,

Engineering Applications: Anti reflection coating and gas sensors.



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Nano Science & technology: Introduction to Nano materials, Size dependent properties of materials, Top-down and Bottom-up approach- Ball-milling and Chemical synthesis, Photolithography, Process, Engineering applications: Nano-engineered transistors and quantum computing components.

Module 4 — Optical Fibers and Optical Communication Systems (8 Hours)

Introduction to Optical Fibers: Optical Fiber and its components, Principle of Operation- Total Internal Reflection (TIR). Optical Parameters-Critical Angle: Acceptance Angle (θ_a), Numerical Aperture (NA), Types of Optical Fibers based on Modes and Based on Refractive Index Profile, Losses in Optical Fibers.

Fiber Optic Communication System: Transmitter: Optical Sources- Laser Diodes, Receiver- Photodetectors (PIN Photodiodes) Fiber Optic Communication system (Basic elements, construction and working) Applications of Optical Fibers in Telecommunications, Medical Imaging, Industrial Automation, and Défense, Numerical Problems

Module 5 — AI for Semiconductor Technology (09 Hours)

Semiconductor device process flow and fabrication, TCAD (Technology Computer-Aided Design) for designing, modelling, and optimizing semiconductor devices, Dataset preparation for lithography, Image-to-image translation in semiconductor patterns

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

- **Module 1:** Students utilize Python, PhET, Qiskit to predict the probability density, visualization of particles in a box, implementing basic quantum gates.
- **Module 2:** AI assisted tools (Claude AI, Python, Colab) are used for Semiconductor parameter optimization. Python-based computational tools are used for the simulation of the characteristics of various semiconductor devices.
- **Module 3:** Python-based computational tools are used for investigating quantum-confinement energy vs. particle size, AI-driven property prediction of 2D, novel materials via Materials Project API. Utilizing simulation and data-driven methods to investigate thin film deposition processes and the behavior of nanomaterials. AI models are employed to predict size-dependent properties, optimize fabrication parameters, and analyze quantum confinement effects.



- **Module 4:** Students utilize simulation platforms such as PhET, Python, and Colab to model step-by step simulation of population inversion, Gaussian beam propagation visualisation, light bending simulation, compare fibre types for AI data-centre links
- **Module 5:** Advanced AI platforms— employed for the prediction of the electrical characteristics of semiconductor devices based on the process flow and parameters and estimation of the device performance, Preparation of dataset for lithography for improved semiconductor mask layout pattern

Engineering Physics Laboratory

List of Experiments

No.	Experiment	Module Link
1	I-V Characteristics of Zener Diode	Module II
2	Four Probe Technique -Resistivity	Module II
3	Laser Wavelength determination using Diffraction grating	Module IV
4	Band Gap Energy of Intrinsic Semiconductor	Module II
5	Planck's Constant using LED	Module II
6	Numerical and Attenuation of Optical Fiber	Module IV
7	Transistor Characteristics (NPN-CE)	Module II
8	Hall Effect - Concentration, Mobility & Conductivity	Module II

Assessment Scheme

Professional Core Courses (Theory)

Criteria	Weightage of Marks
Lab Reports (experiments)	15
Class Tests / Quizzes	10
Mid-Semester Examination	30
AI Mini-Project (Group)	5
End-Semester Examination	40
Total	100 marks



Primary Textbooks

1. **Shankar, R.** — Principles of Quantum Mechanics, 2nd Ed., Springer
2. **Shatendra Sharma & Jyotsna Sharma** — Engineering Physics (2019), Pearson, Noida, India.
3. **Keiser, G.** — Optical Fiber Communications, 5th Ed., McGraw-Hill
4. **Yue Tang and Hua Xu,** — Two-Dimensional Materials for Nonlinear Optics: Fundamentals, Preparation Methods, and Applications, 1st edition, WILEY-VCHGmbH.

Additional / Reference Resources

Supplementary Textbooks:

- Nielsen, M.A. & Chuang, I.L. — Quantum Computation and Quantum Information, Cambridge Univ. Press, 10th Anniversary Ed.
- Thyagarajan, K. & Ghatak, A.K. — Lasers: Fundamentals and Applications, 2nd Ed., Springer India.
- Kasap, S.O. — Principles of Electronic Materials and Devices, 4th Ed., McGraw-Hill
- Beiser, A. — Concepts of Modern Physics, 6th Ed., McGraw-Hill.
- Engineering Physics Laboratory Manual, Dept. of Physics, DSU — Internal Publication, 2025.

Online & Simulation Resources:

1. IBM Qiskit Textbook — Free at qiskit.org/learn
2. NPTEL Lectures: Quantum Computing, Semiconductor Physics & Devices — nptel.ac.in
3. Google Colab Python notebooks — colab.research.google.com
4. PhET Simulations — phet.colorado.edu
5. IBM Qiskit Textbook — Free at qiskit.org/learn
6. NPTEL Lectures: Quantum Computing, Semiconductor Physics & Devices — nptel.ac.in
7. Google Collab Python notebooks — colab.research.google.com
8. PhET Simulations — phet.colorado.edu
9. Materials Project database — materialsproject.org
10. Goodfellow et al. — Deep Learning — Free at deeplearningbook.org

NVIDIA Deep Learning Institute (DLI) Recommended Paths:

- Fundamentals of Deep Learning — learn.nvidia.com (Course: DLI+C-FX-01+V3)
- Physics-Informed Machine Learning — learn.nvidia.com (Course: DLI+S-OV-12+V1)
- Simulation and Modelling with NVIDIA Modulus (PhysicsNeMo) — learn.nvidia.com (Course: DLI+S-OV-06+V1)

Note: DLI content is recommended for self-paced supplementary study. Instructors are encouraged to map each DLI module to the corresponding lecture topic before student enrolment to ensure curriculum alignment.



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakagalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Engineering Physics for Mechanical Cluster

Course Code: 26ENXXXX	Semester: I/II
Programs Offered:	B. Tech (Mechanical and Aerospace Cluster)
Prerequisites: This course requires basic knowledge of Physics and Mathematics, including mechanics, waves, heat, electricity, and modern physics. Familiarity with algebra, trigonometry, and elementary calculus is essential for understanding crystallography, thermal properties of solids, oscillations and waves, semiconductor physics, and Nano science. Basic exposure to computational tools and AI-assisted learning methods will be beneficial.	
Course Type: Basic Science	Hours/Week: 05 (03 T + 2 P)
Total Hours: 45 + 30	Credits: 04
L – T – P – J: 3 – 0 – 2 – 0	

Course Vision

To build a strong foundation in Engineering Physics by integrating fundamental scientific principles with modern mechanical, aerospace, and intelligent engineering technologies, enabling students to understand advanced materials, wave phenomena, semiconductor systems, nanotechnology, aerospace applications, and emerging AI-driven engineering systems.

Course Objectives

Student will:

1. Understand the fundamentals of crystallography, crystal structures, X-ray diffraction techniques, and defects in solids.
2. Explain lattice vibrations, phonons, thermal properties, and heat transport mechanisms in crystalline materials.
3. Apply the principles of oscillations, wave motion, ultrasonic waves, and shock waves in engineering systems.
4. Understand semiconductor physics, semiconductor devices, and nanomaterials for modern engineering applications.
5. Understand the physical principles underlying robotics and Physical AI systems, including rigid body motion, rotational dynamics, kinematics, friction, and collisions.



Course Outcomes (CO)

CO	Outcome Description	Bloom's Taxonomy Level
CO1	Students will be able to explain crystal structures, lattice systems, crystal defects, and analyze materials using X-ray diffraction techniques.	L3
CO2	Students will be able to understand lattice vibrations, phonons, and the thermal properties of solids using classical and quantum models.	L2
CO3	Students will be able to apply the principles of oscillations, waves, ultrasonic waves, and shock waves to mechanical and aerospace engineering applications.	L3
CO4	Students will be able to analyze semiconductor materials, semiconductor devices, and nanomaterial's for modern engineering and technological applications.	L4
CO5	Students will analyze the fundamentals of rigid-body dynamics, rotational motion, coordinate transformations, kinematics, friction, and collision principles relevant to robotics and Physical AI systems.	L4

Broad Course Description

This course provides a comprehensive introduction to Engineering Physics for Mechanical and Aerospace Engineering students by integrating fundamental physical principles with modern engineering applications. The course covers crystallography, crystal structures, X-ray diffraction techniques, lattice vibrations, thermal properties of solids, oscillations, wave phenomena, shock waves, semiconductor physics, semiconductor devices, nanomaterials, and engineering applications of advanced materials. Emphasis is placed on the application of physics concepts to thermal management, vibrations, sensing technologies, aerospace systems, and modern engineering materials. The course also introduces the fundamentals of rigid body dynamics, kinematics, rotational motion, and intelligent engineering systems relevant to robotics and Physical AI. AI/ML-assisted platforms and simulation tools are incorporated through demonstrations and case studies to familiarize students with emerging technologies in mechanical, aerospace, and autonomous engineering systems. The course aims to develop analytical thinking, problem-solving abilities, and an understanding of the role of physics in modern engineering and technological advancements.



Course Modules

Module 1 - Crystallography (8 hours)

Crystallography: Lattice, Unit cell, Lattice parameters, Crystal systems, Bravais lattices, Packing fraction for SC, BCC and FCC crystal systems. Introduction to Miller Indices, Determination of crystal structures using Miller Indices. Expression for Inter-planar distance (Qualitative), X-ray diffraction, Bragg's law (derivation) and Determination of Crystal structure by Powder method, Applications of XRD in science and technology, Defects in crystals, Numerical.

Module 2 - Thermal Properties of Solids (8 Hours)

Monoatomic and diatomic lattices, Lattice vibrations and normal modes, Acoustic and optical phonons, Phonons and phonon momentum, Density of states and dispersion relations, Specific heat of solids – Dulong–Petit law, Classical, Einstein and Debye models (Qualitative), Debye temperature, Thermal conductivity in solids, Lattice thermal resistance, Normal and Umklapp processes, Numerical.

Engineering Application: Bimetallic Strips as switch for thermostat and Thermal insulation

Module 3 - Oscillations And Waves (10 Hours)

Simple harmonic motion (SHM), differential equation for SHM, Spring-mass systems, Free and damped oscillations, Quality factor, Forced oscillations (Qualitative) and resonance, Wave motion, Longitudinal and transverse waves, Standing waves, Ultrasonic waves and applications, Shock waves, Mach number, Properties of shock waves, Numerical.

Engineering Application: Generating thrust in rockets and jet engines

Module - 4: Modern Physics (13 Hours)

Semiconductor Physics: Classification of Solids, Band structure, Fermi level in intrinsic and extrinsic semiconductors, Fermi-Dirac distribution function, Intrinsic carrier concentration (Qualitative), Conductivity of semiconductors, Hall effect, Derivation of Hall Coefficient, Numerical.

Semiconductor Devices: Zener diode: breakdown mechanisms and applications, Light Emitting Diode (LED): working principle and characteristics.

Nano Science: Nanomaterials and their classification, size-dependent properties and quantum confinement, nanocomposites and materials selection criteria, top-down and bottom-up synthesis approaches,

Engineering Application: Advanced Wear Resistance and Lightweight Composites.

Module - 5: Physics for Robotics & Physical AI (6 Hours)

Rigid body dynamics: Foundations (rigid body definition, translational motion, rotational motion, degrees of freedom, torque, angular acceleration, angular momentum, rotational equilibrium), Coordinate Frames and Transformation, Rotational dynamics and inertia, Kinematics, Contacts, Friction and collision.



AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

- **Module-1:** AI-assisted visualization and simulation tools will be used to understand crystal structures, Bravais lattices, Miller indices, and X-ray diffraction patterns. Computational approaches will help students analyze crystal planes and defects in materials. Students will critically interpret AI-generated structural models and diffraction data using fundamental crystallography concepts.
- **Module-2:** Interactive simulations and AI-supported computational tools will be used to study lattice vibrations, phonons, thermal conductivity, and specific heat models. Visualization of dispersion relations and thermal transport phenomena will enhance conceptual understanding. Students will evaluate the physical significance and limitations of theoretical and AI-based thermal models.
- **Module-3:** AI-enabled simulations will support the study of oscillatory motion, resonance, wave propagation, ultrasonic waves, and shock waves. Computational visualization tools will help students understand supersonic and hypersonic motion relevant to aerospace systems. Students will compare simulation results with physical interpretations and engineering applications.
- **Module-4:** AI-assisted tools and simulations will be used to visualize semiconductor band structures, carrier transport, Hall effect, semiconductor devices, and nanoscale phenomena. Computational methods will support the study of quantum confinement and size-dependent properties of nanomaterials. Students will critically analyze AI-generated outputs and relate them to modern engineering technologies.
- **Module-5:** AI/ML-assisted platforms and simulation tools will be introduced to study rigid body dynamics, rotational motion, kinematics, friction, collisions, and robotics-based engineering systems. Demonstrations and case studies related to Physical AI, autonomous systems, and intelligent robotics will help students connect fundamental physics concepts with modern mechanical and aerospace engineering applications while encouraging critical evaluation of AI-assisted engineering solutions.

Engineering Physics Laboratory

List of Experiments

No.	Experiment	Module Link
1	I-V Characteristics of Zener Diode	Module IV
2	Four Probe Technique -Resistivity measurement	Module IV
3	Dielectric Constant Determination	Module IV
4	Determination of the moment of inertia of a circular disc using a torsional pendulum	Module III



5	Band Gap Energy of an Intrinsic Semiconductor	Module IV
6	Planck's Constant using LED	Module IV
7	LCR Series and Parallel Resonance	Module IV
8	Transistor Characteristics (NPN-CE)	Module IV
9	Hall Effect - Concentration, Mobility and Conductivity	Module IV

Assessment Scheme

Professional Core Courses (Theory)

Criteria	Weightage of Marks
Lab Reports (experiments)	15
Class Tests / Quizzes	10
Mid-Semester Examination	30
AI Mini-Project (Group)	5
End-Semester Examination	40
Total	100 marks

Primary Textbooks

1. **S. O. Pillai** - Solid State Physics (2018), New Age International Publishers, New Delhi, Revised Edition.
2. **N. K. Bajaj** - Waves and Oscillations (1998), Tata McGraw-Hill Education, 1st Edition,.
3. **Sergey M. Frolov, Anatoly A. Borisov** - Shock Waves and Explosions (2012), Springer, 1st Edition.
4. **S. O. Kasap** - Principles of Electronic Materials and Devices (2017), 4th Edition, McGraw-Hill Education

Additional / Reference Resources

Supplementary Textbooks:

- **William D. Callister Jr., David G. Rethwisch** - Materials Science and Engineering (2019), Global Edition, 10th Edition
- **Charles Kittel** - Introduction to Solid State Physics (2013), Wiley India Pvt. Ltd., 8th Edition.
- **A. Beiser** - Concepts of Modern Physics (2003), 6th Edition, McGraw-Hill Education.



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Computational Thinking & Programming using Python

[As per Choice Based Credit System (CBCS) scheme]

Course Code: 26XXXXXX	Semester: I/II
Programs Offered:	All programs of B.Tech
Prerequisites: Fundamental computer knowledge	
Course Type: Core	Hours/Week: 04 Hrs (02 + 02)
Total Hours: 30 Hrs (L) + 30 Hrs (P)	Credits: 03
L – T – P – J: 2 – 0 – 2 – 0 (2 Credits Theory + 1 Credit Lab) - Integrated	

Course Vision

This course aims to develop computational thinking and problem-solving skills using Python. It enables students to design and implement programs for real-world applications with a focus on data handling and practical problem solving. The course introduces Python libraries for data analysis and visualization and provides foundational exposure to artificial intelligence concepts such as data preprocessing, feature extraction, pattern recognition, and basic predictive analysis using structured datasets.

Course Objectives

Students will:

1. Gain knowledge of fundamental programming concepts such as variables, expressions and basic constructs in Python.
2. Make use of functions, modules, and strings to develop Python programs for data processing.
3. Gain skills to develop programs using core data structures, file handling techniques, and standard Python libraries.
4. Understand object-oriented programming concepts including classes, objects, inheritance and polymorphism.
5. Make use of operator overloading and exception handling techniques to develop robust Python applications.



Course Outcomes (CO)

CO	Outcome	Bloom's Taxonomy Level
CO1	Apply Python fundamentals including variables, data types, conditionals and loops to develop simple programs.	L3
CO2	Apply functions, modules and string operations to develop modular Python programs.	L3
CO3	Analyze and manipulate data using Python data structures (lists, tuples, sets, dictionaries), file handling techniques and standard Python libraries: <i>math, random, datetime, NumPy, Pandas, and Matplotlib</i> .	L4
CO4	Implement programs using object-oriented principles, including classes, objects, inheritance, and polymorphism.	L3
CO5	Implement operator overloading and exception handling techniques to develop robust Python applications.	L3

Broad Course Description

This course introduces computational thinking and problem-solving using Python programming. It covers fundamental concepts such as control structures, functions, strings, data structures, and object-oriented programming with an emphasis on modular and efficient coding. The course incorporates the use of standard Python libraries for data handling and processing, along with file handling and exception handling to develop reliable programs. Students gain practical experience through laboratory exercises focused on real-world problem solving. The course also introduces foundational artificial intelligence concepts such as data preprocessing and basic pattern identification using Python libraries, along with awareness of modern computing paradigms such as accelerated computing for handling large-scale data applications. The course prepares students to design and implement structured Python-based solutions.

Detailed Course Syllabus

The course consists of five modules. Each module integrates:

- Core topics
- Engineering applications



Course Modules

Module 1: Python Fundamentals and Control Structures [6 Hours]

Topics: Features of Python, variables, data types, operators, input/output handling, indentation and comments, type conversions, conditional statements, looping constructs, nested loops, break, continue, pass, use of else with loops.

Engineering Applications: Basic engineering calculator, Temperature alert system and Traffic signal control system using conditional statements and loops

Module 2: Functions, Modules and Strings [6 Hours]

Topics: Need for functions, function definition and function call, scope and lifetime of variables, return statement, lambda functions, recursive functions, modules and their usage, string operations, immutability, string formatting, built-in string methods, string slicing, membership operators, comparison of strings, and iteration over strings, introduction to regular expressions (re module) for basic pattern matching and text validation.

Engineering Applications: Log file analyzer for system monitoring and error detection, Text data validation and cleaning for healthcare or retail data

Module 3: Data Structures in Python, File Handling and Libraries [7 Hours]

Topics: Lists, tuples, sets, and dictionaries, operations on data structures, file handling techniques including file paths, file types, file operations (read, write, append), file positions, and file management (rename and delete), use of standard Python libraries: *math*, *random*, *datetime*, *NumPy*, *Pandas*, and *Matplotlib* for data processing and visualization, introductory exposure to scikit-learn for basic data modeling workflows.

Engineering Applications: Student record management system for academic data storage and retrieval, CSV-based sales data analysis and visualization in retail systems, Patient dataset processing and pattern identification in healthcare systems

Module 4: Object-Oriented Programming [5 Hours]

Topics: Classes and objects, class methods and use of *self*, constructors and destructors, class variables, public and private data members, private methods, calling methods, static methods, inheritance, types of inheritance, method overriding, and polymorphism, use of dataclasses for simple and structured data representation

Engineering Applications: Banking system for account management and transactions, Inventory management system for retail operations, Smart city resource management system for monitoring utilities

Module 5: Operator Overloading and Exception Handling [6 Hours]

Topics: Operator overloading, implementation of operator overloading, reverse addition, method overriding, errors and exception handling, handling multiple exceptions, *else* and *finally* clauses, raising exceptions, built-in and user-defined exceptions, clean-up actions, use of



logging module to record program execution events and error messages (INFO, WARNING, ERROR levels); introduction to warnings module for handling runtime warnings

Engineering Applications: File-based transaction system with exception handling for invalid operations, Matrix operations using operator overloading, User input validation for order processing systems with exception handling for incorrect entries.

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

- **AI Integration:**

- **Module 1:** Implement a simple rule-based temperature alert system
 - a. Take temperature input
 - b. Use conditional statements to compare with threshold
 - c. Use loops to continuously monitor values
 - d. Display alert messages based on conditions
- **Module 2:** Implement a text validation system:
 - a. Accept input data (e.g., email or ID)
 - b. Use string operations or **re module** for pattern matching
 - c. Validate format and clean the input data
 - d. Display valid/invalid results
- **Module 3:** Implement a simple data analysis system:
 - a. Load dataset using **Pandas**
 - b. Perform preprocessing (filtering, handling missing values)
 - c. Compute basic statistics
 - d. Visualize results using Matplotlib
 - e. Identify simple patterns or trends
 - f. Relate the data processing workflow to parallel execution and accelerated computing concepts used in GPU-based systems (conceptual exposure only)
- **Module 4:** Implement a class-based system:
 - a. Define a class (e.g., BankAccount or Student)
 - b. Implement attributes and methods
 - c. Use inheritance for extending functionality
 - d. Simulate real-world operations
- **Module 5:** Develop a fault-tolerant input system:
 - a. Accept user input
 - b. Validate using conditions
 - c. Handle errors using exception handling
 - d. Use logging to record errors
 - e. Ensure reliable program execution without failures



Authentic Intelligence Reflection:

- **Module 1:** Analysis of how conditional statements and loops are used to implement rule-based decision systems, evaluation of program logic correctness, and reflection on how simple automated decisions are applied in real-world applications.
- **Module 2:** Evaluation of how functions and modular programming improve code reusability and efficiency, and analysis of how string processing and pattern validation support handling of real-world textual data.
- **Module 3:** Analysis of how data structures, file handling, and Python libraries are used to process and manage structured datasets, evaluation of data analysis results, and reflection on how patterns are identified from real-world data.
- **Module 4:** Analysis of how object-oriented programming models real-world systems, evaluation of class design for modularity and maintainability, and reflection on how structured design supports scalable software development.
- **Module 5:** Evaluation of the effectiveness of exception handling and operator overloading in building robust programs, analysis of failure scenarios and error conditions, and reflection on the importance of reliability and fault tolerance in real-world systems.

Laboratory Component

The laboratory emphasizes hands-on implementation of Python programs for problem solving. Students will use IDLE, VS Code, Jupyter Notebook or Google Colab and standard libraries (such as *math*, *random*, *datetime*, *re*, *json*, *logging*, *NumPy*, *Pandas*, and *Matplotlib*) to develop, analyze, and validate solutions for data processing and computational tasks.

Lab Experiments

1. Develop a program for a temperature alert system that triggers warnings when temperature exceeds a threshold using the random module for simulated input.
2. Develop a traffic signal control system using conditional statements and loops.
3. Create a log file analyzer to detect errors and generate reports using re for pattern matching.
4. Build a text validation system for healthcare or retail data inputs using re for format validation.
5. Develop a program to perform file operations for storing and retrieving records using json for structured data storage.
6. Develop a shopping cart application using lists and dictionaries to add, remove, update items, and calculate total bill with discounts.
7. Develop a student record management system using lists and dictionaries with file handling to add, update, delete, and retrieve student records, and store data persistently.



8. Develop a program to analyze sales data using Pandas, compute total sales, identify top-selling products, and visualize trends using Matplotlib.
9. Perform CSV data analysis and visualization using Pandas and Matplotlib libraries to load data, clean and process records, compute summary statistics, and generate visual representations such as bar charts or line graphs.
10. Develop an object-oriented banking system to create and manage accounts, perform operations such as deposit, withdrawal, and balance inquiry, and implement transaction handling using classes and methods.
11. Implement matrix operations using operator overloading with optional use of NumPy for comparison.
12. Develop a robust input validation system with exception handling and use of logging for error tracking.
13. Develop an ATM system using functions and file handling to perform operations such as balance inquiry, deposit, and withdrawal with proper validation and exception handling.

AI-Integrated Lab Exercises (Indicative)

1. Develop a Python program to load a dataset (CSV file) using Pandas, perform preprocessing (handling missing values, sorting, filtering), and compute basic statistics (mean, max, min). Visualize the results using Matplotlib.
2. Develop a Python program using math and random libraries to: Generate random numbers, Perform mathematical operations (square root, logarithm, trigonometric functions), Simulate a simple scenario (e.g., random score generation and analysis)
3. Develop a Python program to load a dataset (e.g., student marks) using Pandas and classify records into categories such as *pass/fail* or *high/medium/low* based on predefined conditions. Visualize the classification results using Matplotlib.
4. Develop a Python program to use the datetime library to process timestamps and analyze time-based data (e.g., login records or transactions). Store and process the data using Pandas, and generate summary insights.

Assessment Scheme

Definitions

1. **“Internal Assessment”** refers to the Continuous Internal Assessment (CIA). The weightage of this component is **60%**. It includes:
 - a. One **Mid Semester Examination (MSE)** (Descriptive-based),
 - b. Six **Continuous Assessment Tool (CAT)**: It may include: Surprise test (Computer-based), Quiz, Project based learning, Reflection Note on Guest Lecture, Industrial visit, E-course certification, Building models, Group discussion, Case study, Seminar, or Open Book Assignment
 - c. One **AI-powered Assignment (APA)**: Use of AI tools for problem-solving



while simultaneously developing Authentic Intelligence through critical evaluation, ethical reasoning, and responsible decision-making.

2. **“External Assessment”** refers to the Semester End Examinations (SEE). The weightage of this component is **40%**. At the end of semester, the final examination is conducted to evaluate a student's performance.

Integrated Professional Core Courses (Theory + Practical)

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT) every 2 weeks (Minimum 05 assessments)	10
Mid Semester Examination (MSE)	20
Certification	10
Lab Record	10
Lab Examination	10
Semester end exam	40
Total	100 marks

Recommended Tools

1. **Programming:** Python (IDLE / VS Code / Jupyter Notebook / Google Colab)
2. **Data Handling & Visualization:** NumPy, pandas, matplotlib
3. **AI-Assisted Development Tools:** GitHub Copilot, ChatGPT, code analysis and debugging tools

Suggested Reading

Primary Textbooks

1. Allen B. Downey, *“Think Python: How to Think Like a Computer Scientist”*, Green Tea Press, 2nd Edition.
2. Fabio Nelli, *“Python Data Analytics”*, Apress, 2018.

Reference Books

1. John V. Guttag, *“Introduction to Computation and Programming Using Python”*, MIT Press, 3rd Edition, 2021.
2. Tony Gaddis, *“Starting Out with Python”*, Pearson, 4th Edition, 2019.
3. Al Sweigart, *“Automate the Boring Stuff with Python”*, No Starch Press.



4. Luciano Ramalho, “Fluent Python”, O’Reilly Media.
5. Reema Thareja, “Python Programming: Using Problem Solving Approach”, 2nd Edition, Oxford University Press, 2019.

Additional Resources

1. [Infosys Springboard](#)
2. [Python Crash Course \(Coursera\)](#)
3. [Python Specialization \(Coursera\)](#)
4. [Python for Applied Data Science & AI \(Coursera\)](#)
5. [Python Official Documentation](#)

DLI Course Mapped

Fundamentals of Accelerated Computing with CUDA Python
(https://learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+C-AC-02+V1)

DLI Learning Path

Accelerated Computing

Strength of CO Mapping to PO/PSOs:

Table: Mapping Levels of Cos to Pos / PSOs													
COs	Program Outcomes (Pos)											PSOs	
	1	2	3	4	5	6	7	8	9	10	11	1	2
C01	2	2	3		2		1				2		1
C02	2	2	3		2		1				2		1
C03	3	3	3	2	3		1				2	2	1
C04	2	3	3		2		1				2	2	1
C05	2	3	3	2	2		1				2	2	1
Avg	11/15	13/15	15/15	4/15	11/15		5/15				10/15	6/15	5/15

3: Substantial (High)

2: Moderate (Medium)

1: Poor (Low)



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

FOUNDATIONS OF AI & AUTHENTIC INTELLIGENCE

Course Code: 26XXXXXX	Semester: I/II
Programs Offered:	All programs of B.Tech
Prerequisites: Fundamental computer knowledge	
Course Type: Core	Hours/Week: 03Hrs
Total Hours: 45 Hrs (L)	Credits: 03
L – T – P – J: 3 – 0 – 0– 0 (<i>3 Credits Theory</i>)	

Course Learning Objectives:

This course will enable students to:

1. Understand and develop a strong foundation in artificial intelligence, including its history, core principles, types (narrow vs. general AI), and real-world applications.
2. Explore how intelligent agents operate, including search algorithms, state-space representation, and decision-making strategies.
3. Learn to represent knowledge using logical and probabilistic models and apply reasoning techniques to solve complex problems.
4. Understand introductory machine learning concepts and evaluate the importance of fairness, transparency, accountability, and authenticity in AI systems.
5. Examine the distinction between artificial and authentic intelligence, focusing on human cognition, trust in AI, and responsible deployment in society.

Course Description:

This course introduces first-year engineering students to the fundamentals of Artificial Intelligence along with the concept of Authentic Intelligence, emphasizing human-centered and ethical decision-making. It covers core AI topics such as intelligent agents, search strategies, knowledge representation, and problem-solving techniques. The course integrates real-world applications including fintech systems, block chain analytics, and AI-driven decision-making tools. Students are exposed to AI-assisted learning using tools like Python, Excel, and visualization platforms. A unique AI² pedagogy is adopted, where students not only use AI tools but also critically evaluate their outputs. The course highlights biases, risks, and limitations of AI systems in practical scenarios. It promotes problem-first learning by connecting concepts with real-world challenges. Research-based learning is incorporated through case studies and student presentations. The course also encourages ethical reasoning, sustainability thinking, and financial inclusion awareness. Overall, it prepares students for responsible participation in AI-driven ecosystems.



Course Outcome:

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course, the student will be able to:		
1	Evaluate the characteristics and advantages of different intelligent agents	L2
2	Identify the different search strategies in AI	L3
3	Apply to solve AI problems	L3
4	Creating software agents in different ways of designing software agents	L3
5	Develop the applications of AI & solve the problems	L3

Course Modules :

UNIT – I : Introduction and Searching decisions [9 Hours]

Introduction, Definition, Future of Artificial Intelligence, Characteristics, Typical Intelligent agents, Problem solving approach, Problem Solving methods, Search strategies, Uniformed and informed, Heuristics, Local search, Algorithm and optimization problems, Searching with partial observations Constraint satisfactory problems ,Constraint propagation ,Back tracking search ,Game playing Optimal decision.

Applications: Heuristics are intelligent estimation techniques that improve search efficiency by prioritizing promising solution paths, Responsible deployment of AI chatbots, systematic techniques.

UNIT – II : AI Reasoning [9 Hours]

Alpha beta pruning, Stochastic games, First order predicate logic, Prolog programming, Unification, Forward Chaining, Backward chaining, Resolution, Knowledge Representation, Ontological Engineering, Categories, Objects, Events, Mental Events, Mental Objects, Reasoning Systems, Reasoning with default information, Typical AI Problems.

Applications: Heuristics are intelligent estimation techniques that improve search efficiency by prioritizing promising solution paths, Responsible deployment of AI chatbots, systematic techniques.

UNIT – III : Intelligent Agents [9 Hours]

Architecture for intelligent agents, Agent communication, Negotiation, Bargaining, Argumentation, Agents, Trust, Reputation, Multi agent systems, AI applications, Language Models, Information Retrieval, Information extraction, Natural language processing, Machine translation, Speech recognition, Robot Hardware, Perception.



Applications: Heuristics are intelligent estimation techniques that improve search efficiency by prioritizing promising solution paths, Responsible deployment of AI chatbots, systematic techniques.

UNIT – IV : Foundations of Authentic Intelligence [9 hours]

Concept and principles of Authentic Intelligence, Augmented Intelligence vs Artificial Intelligence, Human-centered AI design principles, Value alignment between AI systems and human values, Human cognition and intelligence augmentation, Human-AI collaboration models.

Applications: Ethical design of intelligent assistants, Responsible deployment of AI chatbots

UNIT – V : Authentic Intelligence in Emerging Intelligent Systems [9 Hours]

AI for human creativity and innovation, AI-driven knowledge discovery and learning systems, Intelligent systems for adaptive learning and personalization, Generative AI systems and creative intelligence, Authentic Intelligence and human-AI partnership.

Applications: Generative AI for design, art, and content creation, Intelligent tutoring systems and personalized education

AI INTEGRATION & AUTHENTIC INTELLIGENCE REFLECTION:

- **AI Integration:**

1. Apply AI-based search algorithms and heuristic techniques to solve structured and unstructured problem spaces efficiently.
2. Use AI techniques such as logic programming, inference mechanisms, and knowledge representation to model intelligent reasoning systems.
3. Design and implement intelligent agents and AI applications including NLP, information retrieval, and multi-agent systems.
4. Incorporate human-centered AI design principles to develop systems aligned with human values and cognitive processes.
5. Apply generative AI and intelligent systems to enhance creativity, learning, and adaptive decision-making.

- **Authentic Intelligence Reflection:**

6. Critically evaluate algorithmic decisions and ensure appropriate human judgment in selecting optimal problem-solving strategies.
7. Assess the correctness, limitations, and interpretability of AI-driven reasoning in real-world decision-making contexts.
8. Evaluate trust, reliability, and ethical implications of AI agents interacting in dynamic and real-world environments.
9. Reflect on ethical considerations, value alignment, and responsible AI usage in human-AI collaboration.



10. Analyze the balance between human creativity and AI augmentation while ensuring ethical and responsible innovation.

RESEARCH INTEGRATION TEACHING

1. Paper Review: Students study recent AI research on agents, search, and reasoning.
2. Case Studies: Analyze real-world AI systems to link theory with practice.
3. Mini Presentation: Present short reviews on emerging AI topics.
4. AI-Assisted Research: Use AI tools for summarization and comparison of studies.
5. Critical Reflection: Identify research gaps, limitations, and ethical issues.

ASSESSMENT SCHEME :

1. **“Internal Assessment (IA)”** refers to the Continuous Internal Assessment (CIA) with a weightage of 60%. It includes:
 - a. **Mid Semester Examination (MSE):** One descriptive examination to evaluate conceptual understanding of AI fundamentals and problem-solving approaches.
 - b. **Continuous Assessment Tools (CAT – Six Components):** Includes activities such as quizzes, surprise tests, case studies, group discussions, project-based learning, seminars, industrial visits, reflection notes, model building, e-course certifications, or open book assignments.
 - c. **AI-Powered Assignment (APA):** Students utilize AI tools for solving problems while demonstrating Authentic Intelligence through critical evaluation, validation of AI outputs, ethical reasoning, and responsible decision-making.
2. **“External Assessment (EA)”** refers to the Semester End Examination (SEE) with a weightage of 40%. This examination evaluates students’ overall understanding of AI concepts, application skills, and their ability to analyze and solve problems effectively.

A. Professional Core Courses (Theory)

Criteria	Weightage of Marks
Continuous Internal Assessment every 2 weeks (Minimum 06 assessments)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment / Mini Project*	10



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakagalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Semester End Exam	40
Total	100 marks

AI-POWERED ASSIGNMENTS:

Assignment 1: AI-Based Problem Solving using Search Algorithms

Students prompt an AI tool: "Solve a path finding problem using A or heuristic search*"

Students must:

1. Analyze how the algorithm arrives at the solution
2. Evaluate efficiency and optimality
3. Suggest improvements or alternative approaches

Assignment 2: AI-Based Logical Reasoning Analysis

Students prompt an AI tool: "Solve a problem using propositional or predicate logic."

Students must:

1. Verify correctness of logical reasoning
2. Identify missing assumptions or errors
3. Evaluate interpretability of the solution

Assignment 3: AI-Based Intelligent Agent Design

Students prompt an AI tool: "Design an intelligent agent for a real-world scenario (e.g., chatbot, traffic system)."

Students must:

1. Analyze agent structure and environment
2. Evaluate decision-making capability
3. Identify ethical and real-world limitations

Assignment 4: AI-Based NLP Application Evaluation

Students prompt an AI tool: "Summarize or translate a given text using NLP techniques."

Students must:

1. Evaluate accuracy and contextual understanding
2. Identify biases or incorrect interpretations
3. Suggest improvements for better performance

Assignment 5: AI-Based Ethical Decision-Making Case Study

Students prompt an AI tool: "Provide a solution for an ethical dilemma involving AI (e.g., autonomous systems)."



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakagalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Students must:

1. Analyze ethical reasoning provided by AI
2. Identify potential risks or biases
3. Propose a human-centered and responsible solution

Text Books:

1. C o-Intelligence: Living and Working with AI, E. Mollick, 2024, Penguin Publishing Group.
2. Augmented intelligence: the business power of human-machine collaboration, J. Hurwitz, H. Morris, C. Sidner, and D. Kirsch, 2019, Auerbach Publications.
3. The Cambridge handbook of responsible artificial intelligence: interdisciplinary perspectives, S. Voeneky, P. Kellmeyer, O. Mueller, and W. Burgard, eds., 2022, Cambridge University Press.
4. S. Russell and P. Norvig, "Artificial Intelligence: A Modern Approach", Prentice Hall, Third Edition, 2009.

Reference Books:

1. Bratko, —Prolog: Programming for Artificial Intelligence, Fourth edition, Addison-Wesley Educational Publishers Inc., 2011.
2. M. Tim Jones, —Artificial Intelligence: A Systems Approach (Computer Science) , Jones and Bartlett Publishers, Inc.; First Edition, 2008.
3. William F. Clocksin and Christopher S. Mellish, Programming in Prolog: Using the ISO Standard, Fifth Edition, Springer, 2003.

E-Resources : (Recommended Tools)

1. ChatGPT / Gemini for ideation
2. Jupyter Notebook / Google Colab / AnacondaGoogle Vertex AI
3. NetworkX or draw.io
4. AutoML platforms
5. OpenAI Gym
6. UCI Machine Learning Repository
7. https://learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+S-RX-02+V2

Activity Based Learning (Suggested Activities in Class)

1. Group discussion on different search strategies to identify the nearest Patterns.
2. Collaborative Activity is minor project development with a team of 4 students.



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Mapping Levels of COs to POs / PSOs													
COs				Program Outcomes (POs)								PSOs	
	1	2	3	4	5	6	7	8	9	10	11	1	2
CO1	3	2	1	-	2	-	-	2	1	-	2	1	1
CO2	3	3	1	1	2	-	-	-	-	-	-	2	2
CO3	3	3	-	1	2	-	-	-	-	-	2	1	2
CO4	3	2	3	1	3	-	-	1	2	1	-	2	2
CO5	3	3	3	2	3	2	1	2	2	2	3	2	2
3: Substantial (High)				2: Moderate (Medium)						1: Poor (Low)			



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

INTRODUCTION TO ELECTRICAL AND ELECTRONICS ENGINEERING

[As per Choice Based Credit System (CBCS) scheme]

Course Code: 26XXXXXX	Semester: I/II
Programs Offered:	B.Tech (CSE and ME cluster)
Prerequisites: Fundamental physics	
Course Type: Core	Hours/Week: 03Hrs
Total Hours: 45Hrs (L)	Credits: 03
L – T – P – J: 3 – 0 – 0 – 0	

Course Learning Objectives:

This course enables students to

1. To provide a comprehensive understanding of the fundamentals of power electronic converters and devices.
2. To understand electric vehicle architecture and propulsion systems.
3. To analyze the interaction between batteries, power electronics, and electric drives using modeling and simulation by adopting AI tools.
4. To study charging techniques, battery management systems, and energy recovery mechanisms by adopting AI tools for modeling and simulation.
5. To familiarize with current trends and challenges in electric vehicle development.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods where teachers can use to accelerate the Attainment of the various course outcomes.

1. **The lecture method** means it includes not only the traditional lecture method, but also different types of teaching methods may be adopted to develop the course outcomes.
2. **Interactive Teaching:** Adopt the **Active learning** that includes brainstorming, discussing, group work, focused listening, formulating questions, note taking, annotating, and roleplaying.
3. Show **Video/animation** films to explain functioning of various concepts.
4. Encourage **Collaborative** (Group Learning) Learning in the class.
5. To make **Critical thinking**, ask at least three higher order Thinking questions in the



class.

6. Adopt **Problem Based Learning**, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyze information utilizing AI tools.
7. Show the **different ways to solve** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every **concept can be applied to the real world** - and when that's possible, it helps improve the students' understanding.

Course Modules:

Module I: Electrical Circuit Fundamentals [9 Hours]

Basic electrical quantities such as charge, voltage, current, resistance, power, and energy; Ohm's Law; Kirchhoff's Voltage and Current Laws (KVL and KCL); series and parallel resistor networks; voltage and current division rules; introduction to AC and DC waveforms including RMS, peak, and average values.

Applications: Battery sizing for portable electronics, LED current limiting circuits, household wiring load estimation

MODULE II: Electrical Machines and Power Systems [9 hours]

Magnetic circuits including flux, flux density, magnetomotive force (MMF), and reluctance; working principle, types, and applications of transformers; working principle of DC generators and motors; basics of electrical power generation, transmission, and distribution; introduction to renewable energy sources.

Applications: Transformer design for mobile phone chargers and SMPS power supplies, DC motor control in robotics and electric vehicles.

MODULE III: Semiconductor Devices and its application [9 Hours]

Overview of diode construction; diode under no-bias, forward bias, and reverse bias conditions; half-wave and full-bridge rectifiers; construction and operation of npn and pnp BJTs; input and output characteristics of the common-base configuration; regions of transistor operation active, cut-off, and saturation; transistor as a switch.

Applications: AC to DC power conversion in mobile chargers and SMPS, signal demodulation in AM radio receivers, transistor switching in microcontroller output drivers.

MODULE IV: Operational amplifiers [9 Hours]

Op-amp symbols and terminals; modes of operation including single-ended, differential, and common mode; properties of ideal and practical op-amps; applications such as inverting amplifier, non-inverting amplifier, summing amplifier, differential amplifier, differentiator, and integrator.



Applications: Sensor signal amplification in biomedical instrumentation, audio mixing consoles, PID controller implementation in embedded systems.

MODULE V: Digital Electronics and Embedded applications [9 hours]

Binary number system; logic gates and their truth tables—AND, OR, NOT, XOR, NAND, NOR; combination of logic gates and basic numerical operations; sequential logic circuits including JK and D Flip-Flops; application of Flip-Flops in 4-bit binary counters; overview of microcontrollers at the block diagram level.

Applications: Digital arithmetic units in microprocessors, binary counter design for frequency dividers and timers, flip-flop based registers in CPU design.

NVIDIA DLI Courses (Relevant to this Module): Getting Started With AI on NVIDIA Jetson Nano. https://learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+S-RX-02+V2

AI Integration:

1. LTSpice simulation of DC resistive networks; automated nodal equation solving using NumPy; data logging and V-I characteristic plotting using Matplotlib and Pandas.
2. Flux-AI for circuit design.

Authentic Intelligence:

1. Risks of over-relying on simulation without hardware validation; importance of measurement uncertainty and tolerance; ethical responsibility in accurate reporting of experimental data.

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
CO1	Demonstrate a clear understanding of the basic principles of electrical and electronic components , including resistors, diodes, transistors, operational amplifiers, logic gates, transformers and microcontrollers.	L2
CO2	Apply basic circuit laws and principles to analyze electrical circuits , including resistive networks, AC/DC waveforms, and basic transformer and DC machines	L3



CO3	Apply knowledge of semiconductor devices , such as diodes and transistors, to explain rectification and switching behavior in electronic circuits.	L3
CO4	Apply the principles of operational amplifiers to configure basic circuits such as inverting, non-inverting, and summing amplifiers.	L3
CO5	Demonstrate conceptual understanding of digital logic and microcontroller systems , including logic gate combinations, flip-flops, counters, and block diagram-level microcontroller architecture.	L2

TEXT BOOKS:

1. R. L. Boylestad and L. Nashelsky, “Electronic Devices and Circuit Theory”, 11th Ed, Pearson Education, 2013.
2. M. Moris. Mano and Michael D. Ciletti, “Digital Electronics”, 4th Ed, Pearson Education, 2006.
3. D.P.Kothari and I.J. Nagrath, “Basic Electrical Engineering”, 4th Edition, Tata McGraw Hill, 2019.
4. B.L. Theraja and A.K. Therja, “A textbook of electrical technology, Vol. I (Basic electrical Engineering)”, S. Chand Publishing, 23rd Rev Ed, 2006.

REFERENCE BOOKS:

1. David A Bell, “Electronic Devices and Circuits”, 5th Ed, Oxford university press, 2008.
2. Millman & Halkias, “Electronics Devices and Circuits”, 2nd Ed, McGraw Hill, 2010.
3. William H Hayt and Jack E Kimberly and Steven M Durbin, “Engineering Circuit Analysis” 8th Edition McGraw-Hill, 2013.

Alternative NPTEL/SWAYAM Course:

1. NPTEL- <https://nptel.ac.in/courses/122/106/122106025>
2. NPTEL- <https://nptel.ac.in/courses/108/108/108108076>
3. Virtual Labs- <http://vlabs.iitkgp.ac.in/be/>

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)	PSOs
-----	------------------------	------



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka - 562 112

	1	2	3	4	5	6	7	8	9	10	11	1	2
C01	3	2	-	-	-	-	-	-	-	-	2	1	1
C02	3	3	-	-	-	-	-	-	-	-	2	1	2
C03	3	3	-	-	-	-	-	-	-	-	2	1	1
C04	3	3	-	-	-	-	-	-	-	-	2	1	1
C05	3	2	-	-	-	-	-	-	-	-	2	1	1
3: Substantial (High)				2: Moderate (Medium)						1: Poor (Low)			



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakagalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

INTRODUCTION TO ELECTRICAL ENGINEERING

Course Code: 26XXXXXX	Semester: I/II
Programs Offered:	B.Tech (EC cluster)
Prerequisites: Fundamental physics	
Course Type: Core	Hours/Week: 03Hrs
Total Hours: 45Hrs (L)	Credits: 03
L – T – P – J: 3 – 0 – 0 – 0 (3 Credits Theory)	

Course Objectives

This course enables students to

1. To provide a comprehensive understanding of the fundamentals of power electronic converters and devices.
2. To understand electric vehicle architecture and propulsion systems.
3. To analyze the interaction between batteries, power electronics, and electric drives using modeling and simulation by adopting AI tools.
4. To study charging techniques, battery management systems, and energy recovery mechanisms by adopting AI tools for modeling and simulation

Course Outcomes (CO)

CO	Outcome	Bloom's Taxonomy level
CO1	Explain fundamental electrical concepts — voltage, current, resistance, power, energy, and electromagnetic principles underlying DC circuits, AC circuits, and magnetic circuits.	L2
CO2	Apply Kirchhoff's laws, network theorems, and phasor analysis techniques to solve DC and AC circuit problems analytically and through LTSpice simulation.	L3
CO3	Apply the construction, EMF equation, losses, and efficiency of single-phase transformers through theoretical derivation and laboratory	L3



	measurement.	
CO4	Examine the operating principles, drive electronics, and selection criteria of BLDC and stepper motors for embedded and robotic system design.	L3
CO5	Describe power system structure, protection devices, earthing practices, and electricity tariff calculations with reference to IS:732-2019 and IEC:60446 standards.	L2

Course Modules

Module 1: DC Circuit Analysis

Topics: Electrical charge, potential difference, current, power and energy; active and passive elements; independent and dependent sources. Ohm's Law; series and parallel resistance; voltage divider and current divider rules. Kirchhoff's Current Law (KCL) and Kirchhoff's Voltage Law (KVL); nodal and mesh analysis; Thevenin's theorem, Norton's theorem, Superposition theorem; maximum power transfer. Lab: Component familiarization and multimeter usage; Ohm's Law verification — V-I plotting; KVL and KCL verification in series and parallel networks; LTSpice simulation of DC circuits and comparison with hardware results.

Engineering Applications: Sensor biasing, battery equivalent modelling, load analysis in embedded circuits.

MODULE 2: AC Circuits and Phasors

Topics: Sinusoidal waveforms — amplitude, frequency, phase, RMS and average values; phasor representation; V-I relationships for R, L, and C; impedance and admittance of RC, RL, and RLC circuits; active power (P), reactive power (Q), apparent power (S), power factor; three-phase systems — star and delta connections, line and phase quantities. Lab: RC and RL circuits under AC excitation — oscilloscope observation, phase shift, power factor measurement; LTSpice AC and transient analysis.

Engineering Applications: RC/RL filters, impedance matching, power factor correction in motor drives, three-phase power measurement.

MODULE 3: Magnetic Circuits and Transformers

Topics: Magnetic flux, MMF, field intensity, flux density, permeability, reluctance; B-H curve; Faraday's law, Lenz's law, Lorentz force; self and mutual inductance, coefficient of coupling. Single-phase transformer — construction, working principle, EMF equation, turns ratio, step-up and step-down; copper and core losses (introductory treatment).



Engineering Applications: Inductor design, wireless power transfer coils, isolation transformers, instrument transformers (CT and PT), gate driver transformers.

MODULE 4: Motors for Modern Applications

Principle of electromechanical energy conversion; torque, back EMF, losses and efficiency. BLDC motor — construction (stator, rotor, Hall sensors), electronic commutation, trapezoidal back EMF, speed-torque characteristics, PWM speed control, three-phase inverter bridge. Stepper motor — types (variable reluctance, permanent magnet, hybrid), step angle, full step, half step and microstepping, driver ICs (A4988, DRV8825), open-loop position control. Motor selection criteria for embedded and robotic systems.

Engineering Applications: Drone propulsion, robotic joint actuators, 3D printer axes, CNC positioning, gimbal systems, medical syringe pumps.

MODULE 5: Power Systems and Electrical Safety

Power system structure — generation, transmission and distribution; conventional and renewable energy sources (solar PV, wind, biomass); smart meters; electricity tariff types — flat rate, block rate, time-of-day; domestic and industrial tariff calculation. IS:732-2019 and IEC:60446 — conductor colour codes for single-phase and three-phase wiring; fuse, MCB, construction, operating principle and comparison; earthing — pipe and plate earthing, equipment and system earthing; electric shock hazards and safe isolation procedures.

Engineering Applications: Rooftop solar sizing, energy audit, PCB layout grounding, embedded system enclosure wiring, lab and domestic installation safety.

AI Integration:

- LTSpice simulation of DC resistive networks; automated nodal equation solving using NumPy; data logging and V-I characteristic plotting using Matplotlib and Pandas. Flux-AI for circuit design.

Authentic Intelligence:

- Risks of over-relying on simulation without hardware validation; importance of measurement uncertainty and tolerance; ethical responsibility in accurate reporting of experimental data.
- Risks of open-loop motor control in precision applications; ethical considerations in autonomous robotic systems; responsible deployment of motor-driven systems in safety-critical environments.



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakagalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Suggested Reading

Textbooks

1. D.P.Kothari and I.J. Nagrath, "Basic Electrical Engineering", 4th Edition, Tata McGraw Hill, 2019.
2. B.L. Theraja and A.K. Therja, "A textbook of electrical technology, Vol. I (Basic electrical Engineering)", S. Chand Publishing, 23rd Rev Ed, 2006.

Reference book

1. Clayton Paul, Syed A Nasar and Louis Unnewehr, "Introduction to Electrical Engineering", 2nd Edition, McGraw-Hill, 1992.
2. William H Hayt and Jack E Kimberly and Steven M Durbin, "Engineering Circuit Analysis" 8th Edition, McGraw-Hill, 2013

Additional Sources

NPTEL Course Name: Introduction to Electrical Engineering (Link: <https://nptel.ac.in/courses/108108076>)

Table: Mapping Levels of COs to POs / PSOs

CO \ PO/PSO	P01	P02	P03	P04	P05	P06	P07	P08	P09	P010	P011		PS01	PS02
C01	3	2	-	-	-	1	-	-	-	-	2		2	1
C02	3	3	2	-	3	1	-	1	-	-	2		2	2
C03	3	3	2	2	2	1	-	2	-	-	2		2	1
C04	2	2	3	3	3	1	-	1	1	-	2		3	3
C05	2	1	1	-	1	3	2	2	1	1	2		1	1
Avg	13/1 5	11/1 5	8/15	5/15	9/15	7/15	2/15	6/15	2/15	1/15	10/1 5		10/1 5	8/15



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakagalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Engineering Mechanics

Course Code: 26XXXXX	Semester: I/II
Programs Offered: B-Tech -Mechanical Engineering and Aerospace Engineering.	
Prerequisites: Engineering Physics, Basic Mathematics	
Course Type: Foundation Engineering for first year students	
L – T – P – J: 3 – 0 – 0 – 0 (3 Credits Theory)	Credits: 3

Course Vision

This course helps to develop a strong foundation in the fundamental principles of Engineering mechanics for analyzing forces and motion. and it helps students to apply mechanical concepts to solve real-world engineering problems effectively. And also To enhance visualization and analytical skills through the integration of modern computational and AI-based tools and prepare students for advanced engineering subjects and promote problem-solving, critical thinking, and innovation.

Course Objectives

Students will:

1. Illustrate Couples and equivalent force couple system
2. Understand the principles of resolution and composition of forces
3. Calculate moment of coplanar concurrent and coplanar non-concurrent forces
4. Draw free body diagrams of objects subjected to coplanar concurrent and non-concurrent force systems
5. Calculate center of gravity/centroid for various planar figures
6. Determine area moment of inertia for various planar geometrical objects and standard symmetrical sections
7. Explain Limiting friction and Laws of Friction
8. Solve numerical on wedge friction, ladder friction
9. Explain assumptions made in analysis of Trusses
10. Determine axial forces in members of Planar determinate Truss



Course Outcomes (CO)

CO	Outcome	Bloom's Taxonomy Level
C01	Compute Resultant and reactions by principles and resolution of forces in a plane.	L3
C02	Analyze the objects under the action of applied and frictional forces in a plane by equations of equilibrium.	L3
C03	Determine the Moment of Inertia of composite geometrical sections in a plane	L3
C04	Analyze determines two-dimensional truss by the method of joints and method of section.	L3
C05	Analyze the motion of objects by equations of motion, equations of equilibrium, and Newton's laws of motion and calculate quantities in projectile motion by equations of motion.	L3

Broad Course Description

The Engineering Mechanics course provides a foundational understanding of the physical principles that govern the behavior of solid bodies under the action of forces. The course covers essential topics such as force systems, equilibrium, the moment of forces, friction, and the analysis of simple structures. Students will learn to apply these concepts in both two-dimensional and three-dimensional contexts, developing skills in problem-solving, free-body diagram construction, and the calculation of resultant forces and moments. The course emphasizes the importance of mechanical equilibrium and introduces key concepts like centroids, center of gravity, and moment of inertia, which are critical for analyzing static systems in engineering.

Incorporating modern AI-based tools and computational software, this course enhances students' understanding of complex mechanical concepts through interactive simulations and step-by-step problem-solving. Students will also be exposed to real-world applications, such as analyzing trusses and understanding dynamics in motion, preparing them for more advanced engineering subjects. The integration of these technologies helps students visualize abstract concepts and strengthens their ability to model and solve engineering problems efficiently, fostering a deeper understanding of both theory and application.



Detailed Course Syllabus

The course consists of five modules. Each module integrates:

- Core topics
- Engineering applications
- AI integration
- Authentic Intelligence reflection

Course Modules

Module 1: Introduction to Engineering Mechanics

Topics: Introduction to Engineering Mechanics, Force Systems Basic concepts, Particle Equilibrium in 2-D; System of Forces, Co-planar Concurrent Forces, Resultant- Moment of Forces and its Application; Couples and Resultant of force System, Equilibrium of System of Forces.

Engineering Applications: Analysis of different types of forces and to determine its net resultant effect

Module 2: Centroid, Centre and gravity and Moment of inertia

Topics: Introduction, Centroid of simple figures from first principle, centroid of standard sections Centre of Gravity and its implications; Area moment of inertia Definition, Moment of inertia of plane sections from first principles, Theorems of moment of inertia, Moment of inertia of standard sections.

Engineering Applications: Used in the design and analysis of machine components, beams, bridges, and structures to determine stability, load distribution, and resistance to bending and rotation.

Module 3: Friction

Topics: Introduction, Free body diagrams, Equations of Equilibrium. Types of friction, Limiting friction, Cone of Friction, Laws of Friction, Static and Dynamic Friction; Motion of Bodies, related problems.

Engineering Applications: Application of equilibrium and friction principles in the design, analysis, and safe operation of mechanical systems such as brakes, clutches, lifting devices, conveyors, and structural components.

Module 4: Dynamics

Topics: Introduction, Rectilinear motion; Plane curvilinear motion (rectangular path, and polar coordinates); Projectile motion, Basic terms, general principles in dynamics; Types of motion, motion and simple problems, Kinetics- Newton's laws of motion and related problems.

Engineering Applications: Used in the analysis and design of machines, vehicles, robotic systems, projectiles, and automated industrial mechanisms involving motion and force interactions.



Module 5: Truss

Topics: Introduction, Classification of trusses, Equilibrium in two and three dimension; Method of Joints; To determine if a member is in tension or compression; Simple Trusses; Zero force members.

Engineering Applications: Truss analysis is widely used in the design and evaluation of engineering structures such as bridges, roofs, and transmission towers to ensure strength, stability, and efficient load distribution.

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

○ AI Integration:

□ Module 1: Fluid Properties and Statics

- Use AI tools to visualize different types of force systems.
- AI-based simulations for determining resultant and reactions.

□ Module 5: Truss

- AI-based pattern recognition can automatically detect zero-force members from diagrams. Simulation AI can optimize truss geometry for minimum material usage

□ Authentic Intelligence Reflection:

- **Module 1:** Students critically evaluate AI-generated outputs for resultant and reactions and identify possible accurate solution to achieve static equilibrium.
- **Module 5:** Students can develop better conceptual clarity through visualization and can experience faster problem solving and better visualization through AI driven solution.

Assessment Scheme

Definitions

1. **“Internal Assessment”** refers to the Continuous Internal Assessment (CIA). The weightage of this component is **60%**. It includes:
 - a. One **Mid Semester Examination (MSE)** (Descriptive-based),
 - b. Six **Continuous Assessment Tool (CAT)**: It may include: Surprise test (Computer-based), Quiz, Project based learning, Reflection Note on Guest Lecture, Industrial visit, E-course certification, Building models, Group discussion, Case study, Seminar, or Open Book Assignment



- c. One **AI-powered Assignment (APA)**: Use of AI tools for problem-solving while simultaneously developing Authentic Intelligence through critical evaluation, ethical reasoning, and responsible decision-making.
2. **“External Assessment”** refers to the Semester End Examinations (SEE). The weightage of this component is **40%**. At the end of the semester, the final examination is conducted to evaluate student’s performance.

A. Professional Core Courses (Theory)

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT) every 3 weeks (Minimum 06 assessments)	20
Mid Semester Examination (MSE)	30
AI-powered Assignment “Or” Mini Project*	10
Semester End Exam	40
Total	100 marks

*NOTE:

- If any theory course ***consists of a Mini project component***, then the total marks of **10 (Ten marks)** shall be evaluated for that mini project and hence there is no APA for such courses.
- If any theory course ***does not consist of mini project***, then APA is mandatory awarded for **10 marks**.

Recommended Tools

- Simulation: ANSYS, MATLAB
- Visualization: Geogebra
- AI Tools: ChatGPT, Wolfram Alpha

Suggested Reading

Textbooks

1. Irving H. Shames, Engineering Mechanics, 8th Edition, Prentice Hall publications.
2. A Nelson , Engineering Mechanics: Statics and dynamics, Tata McGraw Hill publications.
3. Sawant and Nitsure, Elements of Civil Engineering, Laxmi publications.



Reference book

1. R.C. Hibbler, Engineering Mechanics: Principles of Statics and Dynamics, Pearson Press.
2. H.J. Sawant, S.P Nitsure, Elements of Civil Engineering and Engineering Mechanics, Technical Publications.

Additional Sources

NPTEL Online Courses – Engineering Mechanics (IIT lectures)

Table: Mapping Levels of COs to POs / PSOs													
CO \ PO/PS O	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	1	1	1	0	0	0	0	0	1	0	3
CO2	3	3	1	2	1	1	0	0	0	0	1	0	3
CO3	3	2	1	2	1	0	0	0	0	0	1	1	3
CO4	3	3	2	2	1	1	0	1	0	0	1	1	3
CO5	3	3	1	2	1	0	0	0	0	0	1	0	3
Avg	15/15	13/15	6/15	9/15	5/15	2/15	0/15	1/15	0/15	0/15	5/15	2/15	15/15



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakagalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka - 562 112

Engineering Communication & Technical Writing

Course Code: 26XXXXXX	Semester: I/II
Programs Offered:	All Engineering programs
Prerequisites: Basic English language proficiency and fundamental digital literacy	
Course Type: HSS	L - T - P - J: 2 - 0 - 0 - 0(02 Credits)

Course Vision

To develop communicative competence of engineering students within an AI-enhanced professional environment. To equip students with soft skills and employability competencies for effective technical discourse. To integrate AI tools to optimize professional communication, research writing, and technical documentation.

Course Objectives

Students will:

1. Develop lexical and grammatical competence through AI-assisted practice.
2. Enhance listening, speaking, reading, and writing skills specifically for professional contexts.
3. Optimize oral and written communication utilizing AI feedback loops.
4. Familiarize themselves with employability skills and modern job search techniques.
5. Inculcate critical thinking and research skills, with a focus on AI-supported research writing and referencing.



Course Outcomes (CO)

CO	Outcome	Bloom's Taxonomy Level
C01	Apply appropriate grammar and vocabulary in technical and professional communication; develop competence for presentations and group discussions.	L3
C02	Demonstrate advanced professional discourse skills, including critical thinking and presentation design.	L3
C03	Produce academic, research, and technical writing (emails, reports, papers) using AI tools for clarity and proper referencing.	L3
C04	Use AI tools responsibly to enhance communication, research, and writing tasks.	L3
C05	Write professional documents like resumes and cover letters while applying soft skills such as emotional intelligence and teamwork.	L3

Broad Course Description

The course provides a comprehensive introduction to professional communication in the age of Artificial Intelligence. It covers fundamental language competence, advanced frameworks for professional discourse, and technical writing conventions. Students will learn to leverage AI tools for grammar enhancement, presentation design, and research documentation while maintaining human-centric soft skills. Emphasis is placed on employability through resume drafting, interview techniques, and collaborative project execution.

Communication and Technical Writing, emphasizes the development of communicative competence for professional and academic contexts. It integrates language competence, sociolinguistic competence, discourse competence, and strategic competence into a structured learning pathway supported by AI tools.

Students will increase the language competence through grammar, vocabulary, pronunciation, and listening/reading practice, ensuring accuracy and clarity in technical communication. The students optimize their sociolinguistic competence by learning how to adapt communication styles to varied professional settings, group discussions, and intercultural contexts.

The course further develops discourse competence by training students to organize ideas cohesively in presentations, reports, and technical papers, while mastering conventions of academic and workplace documentation. The strategic competence is also enhanced through problem-solving in communication, effective use of AI-assisted tools, and adaptive strategies for interviews, teamwork, and collaborative projects.

By progressing through the modules, the students will gain the ability to communicate technical ideas clearly and persuasively in oral and written formats. The modules enable the students to apply critical thinking and professional etiquette in diverse workplace scenarios.



The assignments provide a practice to produce structured academic and technical documents with appropriate referencing. The students will learn to leverage AI responsibly to optimize communication, research, and documentation. The students get familiarized with the employability skills through resumes, cover letters, and interview performance.

The course provides an opportunity for human-AI synergy, where students use AI tools for grammar enhancement, presentation design, and research writing, while simultaneously cultivating soft skills such as emotional intelligence, teamwork, and adaptability. The dual focus ensures graduates are prepared for both technical precision and human-centred professional interaction in the professional context of engineering domain.

Detailed Course Syllabus

The course consists of five modules. Each module integrates:

- Core topics
- Applications
- AI integration
- Authentic Intelligence reflection

Course Modules

Module 1: LANGUAGE COMPETENCE AND AI TOOLS (06 Hours)

Listening vs. hearing; Intensive/extensive reading; Pronunciation guidelines; Grammar and vocabulary building.

Applications: Everyday professional discussions, peer interactions, collaborative initiatives.

Module 2: ADVANCED FRAMEWORKS FOR PROFESSIONAL DISCOURSE (06 Hours)

Communication skills and critical thinking; Presentation skills; Group discussion techniques.

Applications: Organizing ideas in group settings, overcoming expression inhibitions, improving nonverbal communication.

Module 3: ACADEMIC/RESEARCH/TECHNICAL WRITING (06 Hours)

Narrowing research topics; Writing minutes, memos, and emails; Academic report writing; Referencing conventions; Technical paper writing.

Applications: Workplace documentation, professional email etiquette, legally bound reports.

Module 4: EMPLOYABILITY SKILLS IN THE CONTEXT OF AI (06 Hours)

Drafting CVs and resumes; Cover letters; Interview techniques; Professional etiquette.

Applications: Presenting achievements effectively, showcasing personality in job applications.

Module 5: WORKPLACE COMMUNICATION AND SOFT SKILLS (06 Hours)

Emotional intelligence; Teamwork; Problem-solving through communication; Group project planning and execution.



Applications: Team collaboration and maintaining human connections while using AI tools.

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

○ AI Integration:

- **Module-1:** The AI tools used to develop the language competence are Grammarly, QuillBot, BBC Learning English App ELSA Speak and Otter.ai, specifically for LSRW skills and grammatical competence.
- **Module -2:** ChatGPT (brainstorming), Canva AI (design), Kialo Edu (debates and Group discussions. Systematic approach using AI tools to optimize the communicative skills mainly focusing on organising the concepts and ideas and context specific presentation.
- **Module-3:** Grammarly, Hemingway Editor, Explainpaper. Email etiquettes, modifying the written expressions/writing in different cultural, academic and professional contexts.
- **Module-4:** The tools like VMock AI, ChatGPT, Peergrade are used by students to familiarize with employability skills and to meet the requirement of their specific professional domain.
- **Module-5:** Role-play, Peergrade

○ Authentic Intelligence Reflection:

- **Module-1:** Revising grammar optimizes competence; pronunciation improves clarity; reading enhances thought coherence.
- **Module -2:** GDs build cohesive idea presentation, self-confidence, and general awareness and improve the communication skills of the students.
- **Module-3:** Email etiquette builds trust; technical writing tailor the content to audiences; ethics ensure appropriate referencing
- **Module-4:** Tailored resumes highlight specific skills; cover letters reveal formal personality.
- **Module-5:** Fosters emotional intelligence, teamwork, and efficient problem-solving for real-world projects.

Research Integration Teaching

Students get familiarized with Multidisciplinary Research using AI Tools and formal academic writing through their Assignment project and reports.



Assessment Scheme

Criteria	Weightage of Marks
Continuous Assessment	20
Computer based Test (CBT)	40
Mini Project and Written Assignment	40
Total	100 marks

Recommended Tools

Writing & Grammar: Grammarly, QuillBot, Hemingway Editor

Speech & Listening: ELSA Speak, Otter.ai, BBC Learning English App

Research & Design: Explainpaper, ChatGPT, Canva AI, Kialo Edu

Employability: V Mock AI, Peergrade

Suggested Reading

1. Dhanavel, S.P. "English and Communication Skills for Students of Science and Engineering". Orient Blackswan Pvt. Ltd., 2009.
2. Meenakshi Raman and Sangeetha Sharma. "Technical Communication- Principles and Practice". 3rd Edition, Oxford University Press, 2009.
3. Murphy R. "English Grammar in Use", Cambridge University Press, 2012.
4. N. Krishnaswamy and T. Sri Raman. "Creative English for communication", Macmillan Publication, 2005
5. Day, R.A. "Scientific English: A Guide for Scientists and Other Professionals". 2nd Edition, Universities Press, 2000.

E-Resources:

- [Technical Communication Book](#)
- [HKS Communications Slides](#)
- [YouTube:CommunicationSkills](#)

Assignments

Assignment 1: Campus to Community: Needs analysis of local issues (e.g., waste management) followed by digital storytelling campaign

Community Immersion & Needs Analysis. The project provides an opportunity for the students to explore local communities to identify pressing issues (e.g., waste management, education gaps, elder care). Problem-Based Learning (PBL) Identify the problem, research solutions, and present actionable plans. Digital Storytelling & Campaigning involves Creating posters or social media campaigns/ posts to raise awareness.

Assignment 2: – Clayconomy: Sculpting Expression through Ideas &Innovations

Mind mapping social issues and translating them into tangible products or patent proposals.



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Ideation & Theme Setting, Brainstorm themes, Mind Mapping (Students sketch ideas linking social issues to sculptural forms), Sculpting Expression includes the translation of ideas into tangible form/ product, the reflection & Exhibition, Presentation & Reflection Journals should be produced through PPT slides. The students need to create a Product/ App/ Presentation/ Writing a Proposal for Patenting.

For example

- A new product (e.g., eco-friendly drone, smart helmet)
 - A new app (e.g., campus navigation, mental health tracker)
 - A new process (e.g., automated attendance system, waste segregation method)
- Materials- Clay/ colours/ papers, Area- Problem statement- Target users- Proposed Solution- Basic technical feasibility-Process description- Product

Assignment 3: Bumps, Breakthroughs & Becoming: The Plays That Shape Us

The students have to perform a specific scene live, using only the texts provided. Live performance of literary scenes to build confidence in communication and group coordination. Selected texts are: 4 Shakespearean plays, 2 Contemporary plays and 3 Plays from Indian literature in English. The permitted time for the Performance will be (5- 7 minutes). (Costumes optional)

Technical/Academic Research Report Writing for the Assignments 1/2/3

Students will follow the format given below for the written assignments.

Abstract, Objectives of the project, Research questions or guiding problem statements. Literature Review, (Brief overview of existing studies, theories, or practices related to the project). Methodology includes the approach taken, Data collection (surveys, interviews, observation, brainstorming). Tools used (AI tools, communication frameworks, design software). Steps followed in project execution need to be mentioned highlighting the relevance of the methods. Implementation and detailed description of how the project was carried out with the relevant diagrams, flow charts and visuals. Results and Discussion must include Presentation of findings, outcomes, or products of the project. Conclusion includes how the research can be taken forward. Project 3 involves the textual analysis of the chosen text. References and Appendices need to be included. (survey forms, scripts, campaign posters, or mind maps).



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Table: Mapping Levels of COs to POs / PSOs													
COs	Program Outcomes (POs)											PSOs	
	1	2	3	4	5	6	7	8	9	10	11	1	2
C01	-	-	-	-	-	-	-	-	-	2	-	-	-
C02	-	-	-	-	-	-	-	-	-	2	-	-	-
C03	-	-	-	-	-	-	-	-	-	2	-	-	-
C04	-	-	-	-	-	-	-	-	-	2	-	-	-
C05	-	-	-	-	-	-	-	-	-	2	-	-	-



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

BIOLOGY FOR ENGINEERS

Course Code: 26XXXXXX	Semester: I/II
Programs Offered: All Engineering Programs	Prerequisites: Basic Biology
Course Type: Multidisciplinary- Basic Science Course (BSC)	
L – T – P – J: 2-0-0-0 (2 Credits Theory)	Credits: 02

Course Vision

To cultivate engineering graduates with a strong foundation in biological principles and the ability to integrate AI to address complex biological-engineering problems.

Course Objectives

Students will:

1. Acquire an understanding on basic modern biological concepts with an emphasis on how bio-processes are analogous to engineering field, as a multidisciplinary field.
2. Develop an understanding of fundamental engineering principles as they operate within physiological systems, and apply this knowledge to engineering design and problem-solving using functional examples from the human body.
3. Explain how biologically derived solutions can serve as a foundation for designing and developing improved processes and products, ultimately contributing to enhanced quality of life.

Course Outcomes (CO)

CO	Outcome	Bloom's Taxonomy Level
CO1	Apply and utilize essential knowledge of the biological mechanisms of living organisms from the perspective of engineers and find solutions to solve bio-engineering problems with appropriate tools	L3
CO2	Distinguish and make use of optimal designs in engineering that are bio-mechanical in nature and build and use by observing and	L3



CO	Outcome	Bloom's Taxonomy Level
	understanding bio-physiological processes involved in sensing, locomotion, and knowledge application of range of bio-chemicals.	
CO3	Demonstrate that bio-chemical, bio-sensory, bio-processes could be path-finders to optimize similarities for functional aspects of electronic, computer, mechanical, electrical machines.	L3
CO4	Understand and apply electronic principles to biological systems for monitoring, diagnosis, and therapeutic intervention at cellular and physiological levels.	L2
CO5	Develop and produce safe, effective therapeutic products using biological systems for the prevention, diagnosis, and treatment of diseases.	L4

Broad Course Description

Biology for Engineers is a multidisciplinary course that bridges biological sciences with engineering and artificial intelligence. The course focuses on understanding how biological systems function and how these principles can be translated into engineering solutions.

Students will explore concepts such as biomimetics, bioenergy, biomechanics, bioelectronics, and biopharma, along with their real-world applications. The course integrates AI-driven approaches for modeling biological systems, analyzing complex datasets, and designing intelligent solutions.

A key focus is on **Authentic Intelligence**, where students critically evaluate AI outputs, combine human reasoning with machine intelligence, and ensure ethical applications in healthcare, biotechnology, and engineering systems.

By the end of the course, students will be able to apply biological inspiration and AI tools to solve real-world engineering challenges.

Detailed Course Syllabus

The course consists of five modules. Each module integrates:

- Core topics
- Engineering applications
- AI integration
- Authentic Intelligence reflection



Course Modules

Module 1: BIOMIMETICS (06 Hours)

Topics: Biomimetics & related branches (Bioinformatics, Bioelectronics, Biomechanics) - Biofluid-Blood, heart mechanics & blood pressure - Life molecules (water, carbohydrates, proteins, lipids, nucleic acids)- Biological data (genomics, proteomics, metabolomics)- Biomimetic AI & bio-inspired robotics

Engineering Applications:

- Healthcare (diagnosis, heart monitoring) - AI-driven modeling of biological systems for optimization
- Design of energy-efficient systems inspired by nature (e.g., bird flight → aerodynamics)
- Biological data analysis

Module 2: BIOENERGY (06 Hours)

Topics: Unit of life: Human and Plant cell- Metabolism: Enzymes as Bio-catalysts and physiological entities, Anabolism- Bioenergy from Sun-Photosynthesis, catabolism, AI in Modeling metabolic and biochemical pathways

Engineering Applications:

- Energy production in living systems
- Photosynthesis in plants and food production
- Digestion and respiration in humans
- Medical and agricultural improvements

Module 3: BIOMECHANICS (05 Hours)

Topics: Biomechanics: Human body movement mechanics - Force and vector of body movement - Movement angles and joint motion - Muscle contraction and relaxation - Posture: static and dynamic - Ideal vs abnormal posture - Practical movements: stepping, lifting, sitting, standing - AI-based smart prosthetic limbs

Engineering Applications:

- Improving posture and movement efficiency
- Injury prevention and rehabilitation
- Sports performance enhancement
- Design of ergonomic systems
- Development of prosthetic limbs

Module 4: BIOELECTRONICS (06 Hours)

Topics: Bioelectronics: Brain-computer interaction - Artificial Neural Networks (pattern & image recognition) - Biosensors and AIoT in healthcare - Bionic eye (mechanism of vision) - Electronic nose (bio-olfactory mechanism) - Nerve and cardiac impulses - Biological clock and circadian rhythm

Engineering Applications:

- Brain-computer interfaces in medical technology
- Disease diagnosis using biosensors



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

- Smart healthcare systems (AIoT)
- Vision restoration (bionic eye)
- Smell detection (electronic nose)
- Monitoring heart and nerve activity

Module 5: Biopharma (05 Hours)

Topics: Metabolic syndromes - Cancer and its diagnostics - Lab-on-a-chip technology - Drug development and discovery - AI in drug discovery and vaccine development.

Engineering Applications:

- Early diagnosis and monitoring of diseases
- Personalized medicine for metabolic disorders and cancer
- High-throughput screening in drug development
- Point-of-care testing with lab-on-a-chip devices

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

- **Module-1:** This module reflects authentic intelligence by helping students analyse biological datasets, predict molecular structures, model metabolic pathways and use biomimetic algorithms & robotics.
- **Module-2** This module reflects authentic intelligence by aiding students in Modelling metabolic pathways, predicting enzyme activity, analysing biochemical reactions, disease prediction and drug design.
- **Module-3:** This module reflects authentic intelligence by encouraging students to partake in the creation of smart prosthetic limbs with adaptive movement, motion analysis and posture correction, predictive models for muscle and joint behaviour and rehabilitation using AI-assisted devices.
- **Module-4:** This module reflects authentic intelligence by promoting the development of Neural networks for image and pattern recognition, AI-based biosensors for real-time health monitoring, AIoT systems for smart healthcare, signal processing of nerve and cardiac impulses and modelling circadian rhythms and biological patterns
- **Module-5:** This module reflects authentic intelligence by emphasizing on predicting drug-target interactions, designing new drug molecules and vaccines, modeling metabolic pathways for therapy optimization, enhancing diagnostic accuracy and speed.

Authentic Intelligence Reflection:

- **Module-1:** This module reflects authentic intelligence by the integration of human understanding with AI in biology and engineering, helps interpret data from genomics, proteomics, and metabolic pathways, supports decision-making in biomimetics and bio-inspired systems, ensures ethical and effective use of AI in healthcare, biotechnology, and engineering.
- **Module-2:** This module reflects authentic intelligence by Combining human



understanding with AI analysis, interpreting metabolic and biochemical data and making ethical and accurate biological decisions.

- **Module-3:** : This module reflects authentic intelligence by focusing on Combining human understanding with AI for movement analysis, interpreting body mechanics for better prosthetic design and ensuring safe and ethical use in healthcare and rehabilitation
- **Module-4:** This module reflects authentic intelligence by encouraging the design of AI in bioelectronics systems, interpreting neural and sensory data effectively and ensuring ethical use in healthcare technologies
- **Module-5** module reflects authentic intelligence by promoting the development of AI for drug and vaccine development, interpreting complex biological data responsibly and making informed decisions in therapy and diagnostics

Research Integration Teaching

Students review at least one topic:

- AI in genomics and precision medicine
- AI-driven drug discovery
- Bio-inspired robotics
- AI in healthcare diagnostics

ASSESSMENT SCHEME

Definitions

1. **“Continuous Internal Assessment (CIA)”** with a weightage of **60%**. It includes:
 - a. **Mid Semester Examination (MSE):** One descriptive examination to evaluate conceptual understanding of AI fundamentals and problem-solving approaches.
 - b. **Continuous Assessment Tools (CAT – Six Components):** Includes activities such as quizzes, surprise tests, case studies, group discussions, project-based learning, seminars, industrial visits, reflection notes, model building, e-course certifications, or open book assignments.



B. Professional Core Courses (Theory)

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT) – Quiz, Assignments, Write ups	60
Mid Semester Examination (MSE)	40
Total	100 marks

Recommended Tools

- **MATLAB** – modeling biomechanics, system simulations
- **ANSYS** – biomimetic design testing
- **PyMOL** PyMOL – protein structure visualization
- **MATLabVIEW** – biomedical signal acquisition
- **BioSPPY** – ECG/PPG analysis
- **TensorFlow** – disease prediction models
- **AlphaFold** – protein folding prediction
- **AutoDock**– drug design
- **BLAST** – gene/protein comparison
- **Rosetta** – protein engineering
- **Python** – core tool for AI and bioinformatics
- **R (programming language)** – statistical biology

Suggested Reading:

1. "Biology for Engineers" by Arthur T. Johnson: published by CRC Press in 2010.
2. *Biology for Engineers*. Author *Dr. M. Srinivasan & Dr. Santosh Kumar*; Khanna Publishers. ISBN: 978-93-92549-01-4
3. Campbell, N. A.; Reece, J. B.; Urry, Lisa; Cain, M.L.; Wasserman, S. A.; Minorsky, P. V.; Jackson, R. B. Pearson. "Biology: A global approach", , Global Edition, 10/E, 2014
4. David Nelson, Michael Cox. "Lehninger Principles of Biochemistry". W H Freeman & Company, Seventh Edition, 2017.
5. Janine M Benvus. "Biomimicry: Innovation inspired by Nature". William Morrow Paperbacks, 2002.
6. John Holland (1992). *Adaptation in Natural and Artificial Systems*. MIT Press.
7. Melanie Mitchell (1998). *An Introduction to Genetic Algorithms*. MIT Press.
8. Lecture Notes, PPT slides by course instructor.



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11		1	2
C01	2	1	0	0	0	2	0	0	2	0	0		3	0
C02	2	1	0	0	0	2	0	0	2	0	0		3	0
C03	2	1	0	0	0	2	0	0	2	0	0		3	0
C04	2	1	0	0	0	2	0	0	2	0	0		3	0
C05	3	1	0	0	0	2	0	0	2	0	0		3	0



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

KANNADA KALI (ಕನ್ನಡ ಕಲಿ)

Course Code : 26XXXXXX	Semester : I / II
Programs Offered : B.Tech Programs	
Prerequisites : Basic understanding of regional languages and communication skills	
Course Type : Ability Enhancement Course (AEC)	
L – T – P – J : 1 – 0 – 0 – 0	Total Hours : 15 Hours
	Credits : 01

Course Vision

To enable non-Kannada learners to acquire foundational Kannada communication skills for academic, social, and cultural integration while promoting responsible usage of AI-enabled language learning tools and authentic communication abilities.

Course Objectives

Students will:

1. Understand the origin and development of Kannada language and Karnataka culture.
2. Learn Kannada alphabets, numbers, pronunciation, and vocabulary.
3. Develop basic grammar and sentence construction skills.
4. Communicate effectively in day-to-day situations using Kannada.
5. Utilize AI tools such as translators, chatbots, and voice assistants responsibly for language learning.

Course Outcomes (CO)

CO	Outcome	Bloom's Taxonomy Level
CO1	Students will be able to describe the origin of Kannada language and Karnataka culture.	L2
CO2	They will be able to identify Kannada alphabets, numbers, pronunciation patterns, and vocabulary.	L1
CO3	Students will gain the ability to apply grammar concepts such as	L3



CO	Outcome	Bloom's Taxonomy Level
	pronouns, verbs, and tenses in sentence construction.	
CO4	They will be able to construct simple Kannada sentences and communicate in everyday situations.	L3
CO5	Students will be able to demonstrate conversational Kannada skills using appropriate vocabulary and cultural expressions.	L4

Broad Course Description

Kannada Kali is an ability enhancement course designed for non-Kannada learners to develop foundational communication skills in Kannada language. The course introduces students to the origin, culture, alphabets, grammar, pronunciation, vocabulary, and conversational aspects of Kannada.

The course integrates AI-based tools such as translation systems, voice assistants, OCR systems, and NLP chatbots to enhance learning experiences. Simultaneously, it promotes Authentic Intelligence by encouraging students to critically evaluate AI-generated outputs and develop real-life communication skills independently.

The course also emphasizes practical learning through activity-based teaching methods, conversational practice, pronunciation exercises, and cultural interaction.

Detailed Course Syllabus

The course consists of five modules. Each module integrates:

- Core language concepts
- Communication applications
- AI integration
- Authentic Intelligence reflection

Course Modules

Module 1: INTRODUCTION TO KANNADA AND CULTURE (03 Hours)

Topics: Origin and evolution of Kannada language – Karnataka geography and culture – Heritage and traditions – Importance of Kannada language – Cultural identity and social integration.

Applications:

- Cultural understanding and integration
- Communication readiness in Karnataka
- Social interaction in regional environments
- Language appreciation and heritage awareness



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakagalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Module 2: BASICS OF KANNADA LANGUAGE (03 Hours)

Topics: Kannada alphabets (Varnamale) – Swaragalu and Vyanjanagalu – Numbers – Basic pronunciation – Reading and writing practice.

Applications:

- Reading Kannada signboards and notices
- Writing simple Kannada words
- Basic pronunciation and listening skills
- AI-assisted pronunciation learning

Module 3 : GRAMMAR AND VOCABULARY (06 Hours)

Topics: Vocabulary building – Pronouns – Verbs – Tenses – Sentence formation – Everyday vocabulary – Basic grammar rules.

Applications:

- Constructing simple Kannada sentences
- Daily communication
- Understanding grammar usage
- Vocabulary enhancement for practical speaking

Module 4 : COMMUNICATION SKILLS AND CONVERSATION (03 Hours)

Topics: Greetings – Self-introduction – Asking questions – Daily conversations – Phrase making – Colloquial Kannada communication.

Applications:

- Classroom communication
- Market and travel conversations
- Interpersonal communication
- Practical spoken Kannada usage

Module 5 : AI TOOLS FOR LANGUAGE LEARNING (03 Hours)

Topics: Google Translate – Voice Assistants – NLP Chatbots – OCR tools – AI-based pronunciation systems – Ethical use of AI tools.

Applications:

- AI-assisted language learning
- Pronunciation improvement
- Translation support
- OCR-based text learning
- Responsible AI usage in communication



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakagalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Assessment Scheme

Definitions: Continuous Internal Assessment (CIA)

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT) – Viva and Assignment (AI)	25
Mid Semester Examination (CBT)	15
Attendance	10
Total	50 Marks

Activity Based Learning (Suggested Activities in Class)

- Alphabet writing and pronunciation practice
- Vocabulary games and quizzes
- Role-play conversations
- Listening and speaking exercises
- AI-assisted translation activities
- Group discussions in simple Kannada
- Sentence formation exercises
- Short presentations in Kannada
- Cultural story telling activities
- Reflection logs (AI vs real communication)

Recommended Tools

Language Learning Tools

- Google Translate
- Google Assistant
- OCR Applications

AI & NLP Tools

- ChatGPT
- NLP Chatbots
- Voice Recognition Systems

Learning Platforms

- YouTube Kannada Learning Channels
- Duolingo
- Digital Kannada Learning Platforms



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Suggested Reading

Primary Textbooks

1. Kannada Kali – Dr. Lingadevaru Halemane
2. Kannada Paatagalu – Dr. Chandrashekara Kambara
3. Basic Grammar – SLN Sharma & K Shankaranarayana
4. Spoken Kannada – Kannada Sahitya Parishat

Additional Resources

1. Kannada Learning Materials by Kannada Sahitya Parishat
2. Basic Conversational Kannada Guides
3. Digital Language Learning Resources
4. AI-based Language Learning Platforms



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

KANNADA MANASU (ಕನ್ನಡ ಮನಸ್)

Course Code : 26XXXXXX	Semester : I / II
Programs Offered : B.Tech Programs	
Prerequisites : Basic understanding of Kannada language, literature, and communication skills	
Course Type :	Ability Enhancement Course (AEC)
L – T – P – S : 1 – 0 – 0 – 0 (1 Credit)	Credit: 1

Course Vision

To promote Kannada language, culture, literature, and ethical values among students through literary appreciation, communication skills, and responsible usage of AI-enabled learning tools while developing authentic understanding and cultural awareness.

Course Objectives

Students will:

1. Understand Kannada language, culture, and literary heritage.
2. Develop appreciation towards Kannada literature and value-based learning.
3. Analyze social harmony, coexistence, and ethical themes in literary texts.
4. Apply Kannada communication skills in practical and real-life contexts.
5. Utilize AI tools for language and literature learning while maintaining authentic understanding.

Course Outcomes (CO)

CO	Outcome	Bloom's Taxonomy Level
CO1	Students will be able to describe the origin and development of Kannada language and culture.	L2
CO2	They will be able to interpret literary texts and identify cultural and moral values.	L2
CO3	Students will gain the ability to analyze themes such as social harmony and coexistence in Kannada literature.	L4



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakagalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

CO	Outcome	Bloom's Taxonomy Level
CO4	They will be able to apply Kannada cultural understanding and communication skills in real-life situations.	L3
CO5	Students will be able to compose formal and informal letters in Kannada effectively.	L3

Broad Course Description

Kannada Manasu is an ability enhancement course aimed at promoting Kannada language, culture, literature, and ethical values among students. The course introduces learners to selected literary texts, moral values, social harmony, and cultural identity through Kannada literature.

The course integrates AI-based tools such as translation systems, OCR tools, voice assistants, and NLP chatbots to enhance understanding and interaction. Simultaneously, it promotes Authentic Intelligence by encouraging students to critically evaluate AI-generated outputs and develop independent interpretation skills.

The course also emphasizes communication, cultural awareness, literary appreciation, ethical learning, and practical Kannada writing skills.

Detailed Course Syllabus

The course consists of five modules. Each module integrates:

- Literary concepts
- Cultural understanding
- AI integration
- Authentic Intelligence reflection

Course Modules

Module 1: CULTURE (ಸಂಸೃತಿ) – (03 Hours)

Topics: Kannada cultural heritage – Works of Purandaradasa – Chidananda Murthy writings – Karnataka traditions – Cultural identity and values.

Applications:

- Understanding Karnataka culture and heritage
- Literary appreciation
- Cultural awareness and social integration
- Value-based learning



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakagalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Module 2: VALUES (& (ಗ*) – (03 Hours)

Topics: Moral values in Kannada literature – Ethical reasoning – Folklore and poems – Life lessons through literature.

Applications:

- Ethical understanding
- Value-based thinking
- Personality development
- Social responsibility

Module 3: CO-EXISTENCE (ಸಹ, -.) – (03 Hours)

Topics: Social harmony – Human relationships – Coexistence in society – Literary interpretation of social themes.

Applications:

- Social awareness
- Human values and relationships
- Community understanding
- Peaceful coexistence

Module 4: LANGUAGE & IDENTITY (/ 0 - 1 2 3 ೦೦೩5) – (03 Hours)

Topics: Kannada language pride – Linguistic identity – Classical and modern Kannada texts – Importance of regional language.

Applications:

- Language identity and preservation
- Communication skills
- Literary understanding
- Regional language awareness

Module 5: LETTER WRITING (ಪತ7 8 ಂ ಖನ) – (03 Hours)

Topics: Formal letters – Informal letters – Letter writing formats – Practical Kannada writing skills.

Applications:

- Practical communication
- Official letter writing
- Personal communication skills
- Written Kannada proficiency



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Assessment Scheme

Definitions

Continuous Internal Assessment (CIA)

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT) – Viva and Assignment (AI)	25
Mid Semester Examination (CBT)	15
Attendance	10
Total	50 Marks

Recommended Tools

Language Tools

- Google Translate
- Google Assistant
- OCR Applications

AI & NLP Tools

- ChatGPT
- NLP-based Translation Tools
- Voice Recognition Systems

Learning Platforms

- YouTube Kannada Learning Channels
- Online Language Learning Platforms
- Kannada Literature Digital Archives

Suggested Reading

Primary Textbooks

1. Kannada Manasu – Engineering Kannada Text
2. Selected Kannada Literary Works
3. Works of Purandaradasa and Kuvempu
4. Kannada Sahitya Parishat Publications



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakagalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Additional Resources

1. Kannada Literature Resources
2. Digital Kannada Learning Materials
3. AI-based Language Learning Platforms
4. Regional Language Preservation Resources



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Engineering Chemistry for Computer Science Cluster

[As per Choice Based Credit System (CBCS) scheme]

Course Code: 26XXXX	Semester: I/II
Programs Offered: BTech – CSE	Prerequisites: Nil
Course Type: Basic Science (BSC).	Hours/Week: 4 hours (2 T + 2 L)
Total Hours: 60 (30 T + 30 P)	Credits: 03
L-T-P-J: 2-0-2-0	

Course Vision

To impart strong fundamentals in chemical sciences and advanced materials, integrating computational tools, modern electronic systems, and sustainable energy technologies, enabling students to innovate and contribute effectively to contemporary engineering solutions.

Course Objectives

At the end of course students will able to:

1. Understand atomic structure, quantum principles, and chemical bonding theories to explain molecular properties.
2. Understand computational tools for molecular modeling, structure building, and basic property prediction.
3. understand principles and applications of sensors, electrochemical systems, and modern energy devices.
4. Understand polymer properties and explore green energy technologies including photovoltaics and hydrogen fuel.
5. Understand electronic memory systems, optoelectronic materials, and their advanced applications.



Course Outcomes (CO)

At the end of the course, students are able to

CO	Outcome	Bloom's Taxonomy Level
CO1	Apply the principles of atomic structure, quantum mechanics, chemical bonding, and solution chemistry to explain molecular properties, predict structures, and solve concentration and solubility-related problems.	L1
CO2	Apply computational chemistry tools to model molecular structures, and construct Z-matrices, evaluate chemical properties.	L3
CO3	Apply principles of sensors, electrochemical systems, and energy devices to analyze sensing mechanisms, battery technologies, and quantum dot solar cell applications.	L3
CO4	Apply the concept of polymerization for the synthesis of polymers and study their structure-property relationship for commercial applications.	L2
CO5	Apply concepts of electronic memory and optoelectronic materials to understand device functioning and material applications.	L2

Broad Course Description

This course provides a comprehensive foundation in engineering chemistry by integrating fundamental concepts of atomic structure, chemical bonding, and molecular properties with modern applications. It introduces students to polymers, electrochemical systems, sensors, and advanced materials, including optoelectronic and electronic memory devices. The course also emphasizes sustainable technologies such as green fuels, photovoltaics, and hydrogen energy, as well as computational tools for molecular modeling, enabling students to apply chemical principles to solve real-world engineering problems.

Course Modules

Module 1: Quantum and Molecular Foundations for Computing Materials (06 Hours)

Atomic Structure -de Broglie's theory, Bohr's atomic model-Sommerfeld's extension of atomic structure; Electronic configuration and Quantum numbers; Shapes of s, p, d, f orbitals - Pauli's exclusion principle - Hund's Rule of maximum multiplicity- Aufbau principle. Heisenberg's uncertainty principle – wave nature of electron – Schrodinger wave equation (No derivation). Chemical bonding and hybridization of atomic orbitals involving s, p and d orbitals.



Solutions: Types of solutions and classification (solid, liquid and gaseous solutions). Concentration units—molarity, molality, normality, mole fraction and mass percentage. Preparation of solutions and numerical problems involving molarity. Solubility and factors affecting solubility.

Engineering Applications: Semiconductors and electronic devices, prediction of molecular behavior and material properties, Development of nanotechnology and functional materials, Optimization of chemical and industrial processes, Environmental engineering, Corrosion control.

Module 2: Molecular Modelling and Cheminformatics (06 Hours)

Scope of Computational Chemistry: Quantum Mechanics vs Molecular Mechanics vs Experimental Chemistry, Essential Chemical Data Formats for Small Molecules (Z-matrix, PDB, XYZ, MOL, SMILES), DFT Basics: Functionals, Basis Sets, and Electron Correlation, Band Gap Engineering, Global Reactivity Descriptors for Drugs and Materials Screening, IR Spectroscopy & Thermochemistry in Computational Chemistry, Fundamentals of Classical MD Simulations and Thermodynamic Ensembles in MD Simulations.

Hands-On: Molecular Modeling with Avogadro, QM Software: DFT Calculations with GAMESS, MD Software: Biomolecular Simulations with NAMD.

Engineering Applications: Drug design, material development, molecular modeling, computational prediction, industrial process optimization.

Module 3: Sensor Technologies and Electrochemical Energy Systems (06 Hours)

Electrochemical Cells: Galvanic (voltaic) cells and electrolytic cells, electrode potential, standard electrode potential, cell reactions, EMF, and its significance. **Sensors:** Introduction, classification (physical, chemical, biological sensors), Transducer, Actuator, and Sensor: Definitions, functions, and distinctions. **Colorimetric Sensors:** Principle (Beer–Lambert law), instrumentation, and applications in diagnostics. **Engineering Applications:** Potentiometric and Conductometric Sensors: Working principles and applications. **Electrochemical Sensing Systems:** Three-electrode system (working, reference, and counter electrodes) principles, construction, and working. **Cyclic Voltammetry** (principle and sensing applications). **Disposable Sensors:** Definition, advantages.

Energy Systems: Introduction to batteries, construction, working, and applications of Lithium-ion and Sodium-ion batteries. **Quantum Dot: Sensitized Solar Cells (QDSSC's)-** Principle, Properties and Applications.

Engineering Applications: Engineering Applications: Sensors, electrochemical analysis, pollution monitoring, batteries, renewable energy devices



Module 4: Polymeric Materials and Sustainable Energy Technologies (06 Hours)

Polymers: Introduction, Molecular weight - Number average, weight average, and numerical problems. Conducting polymers – Preparation, properties, and commercial applications of conducting polymers: PANI (polyaniline), PPy (polypyrrole), Polythiophene, PEDOT. Specialty of Polymers for designing computers: Preparation, Properties, and Applications: Acrylonitrile-butadiene-styrene (ABS), polycarbonate (PC), and PVC. Green Fuels: Introduction, Power alcohol, Biodiesel. Photovoltaic cells: Introduction, definition, importance, working of PV cell. Solar-grade silicon physical and chemical properties relevant to photovoltaic, doping of silicon by the diffusion technique. Green hydrogen: Introduction, electrolysis of water, and hydrogen fuel cell.

Engineering Applications: Flexible electronics, design of plastic components in computers, clean energy, and sustainable energy.

Module 5: Electronic Memory and Optoelectronic Materials (06 Hours)

Memory Devices: Introduction, Basic concepts of electronic memory, History of organic/polymer electronic memory devices. Classification of electronic memory devices (Transistor-Type, Capacitor-Type, Resistor-Type and Charge transfer type Electronic Memory devices). Display Systems: Photoactive and electroactive materials - Definition and principle. Optoelectronic devices: Definition, working principle. Nanomaterials (Silicon Nanocrystals) and organic materials (Light absorbing materials - Polythiophenes (P3HT), Light emitting materials – Poly [9-vinylcarbazole] (PVK)). Properties of optoelectronic devices.

Engineering Applications: Memory devices, OLEDs, sensors, nanoelectronics.

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

- **AI Integration:**

Module-1: Students explore and visualize atomic and molecular structures using interactive platforms like Atom Animation and MolView, enabling them to understand spatial arrangement, bonding, and properties of molecules in solutions through dynamic 2D/3D models, thereby connecting theoretical chemistry with real-time digital learning and AI-assisted visualization.

Module-2: Conceptual overview of AI and Machine Learning models interpreting SMILES for property prediction and drug discovery (Python for chemistry using Google Colab and RDKit for programmatic handling of molecules and automated data extraction).



Module-3: Collect, visualize, and analyze sensor data using MATLAB and AI tools for real-time IoT-based projects and predictive analytics. <https://thingspeak.mathworks.com/>

Module-4: Explore, analyze, and predict materials for energy systems such as batteries and fuel cells using computational and AI-driven approaches. (<https://next-gen.materialsproject.org/>)

Module-5: AI-assisted analysis of optoelectronic materials for OLEDs, sensors, nanoelectronics, and memory devices.

● **Authentic Intelligence Reflection:**

Module-1: Students connect theory with real-world chemistry, enhancing spatial understanding and critical thinking. AI-assisted exploration fosters analysis of structure–property relationships, promoting problem-solving skills aligned with modern computational and research-driven scientific practices.

Module-2: This module builds authentic intelligence by enabling students to apply AI and machine learning for interpreting SMILES in property prediction and drug discovery. Using Python, Google Colab, and RDKit, learners gain hands-on experience in molecular data handling, fostering computational thinking, automation skills, and real-world problem-solving in modern cheminformatics and pharmaceutical research.

Module-3: This module fosters authentic intelligence by engaging students in real-time IoT data collection, visualization, and analysis using MATLAB and AI tools. By working with platforms like ThingSpeak, learners develop data-driven decision-making and predictive analytics skills, connecting theoretical knowledge with practical applications in smart systems, automation, and modern engineering problem-solving environments.

Module-4: This module develops authentic intelligence by enabling students to explore and predict materials for energy systems using computational and AI-driven approaches. Using platforms such as the Materials Project, learners analyze structure–property relationships, fostering data-driven reasoning, critical evaluation, and innovation skills aligned with modern materials discovery and sustainable energy research.

Module-5: This module develops authentic intelligence by enabling students to relate the properties of electronic memory and optoelectronic materials with their applications in OLEDs, sensors, and nanoelectronic devices, fostering analytical thinking, material understanding, and application-oriented problem-solving skills.



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Laboratory Component

The lab emphasizes **hands-on Experience and visualization.**

Lab Experiments

Volumetric and Instrumental Analysis

1. Potentiometric estimation of FAS using standard potassium dichromate.
2. Conductometric estimation of H^+ in a weak and strong acid using standard sodium hydroxide solution.
3. Determine pKa of a weak acid using standard sodium hydroxide with a pH meter.
4. Synthesis of Nylon 6,10 (Condensation Polymer).
5. Colorimetric estimation of copper in a given solution.
6. Electrolysis of water.

Computational experiments

7. Computational Investigation of Ethylene Carbonate-Li⁺ Complexes: Structural and Vibrational Properties in Lithium-Ion Batteries.
8. Molecular Dynamics Simulation and Structural Analysis of Solvated Lysozyme.
9. Computational IR Spectroscopy of Pharmaceuticals: Analyzing Paracetamol and Phenacetin.
10. Computational Studies of Chain Length Effects on the Optoelectronic Properties of PEDOT.
11. Computational Investigation of the Pharmacophore Quinoline: Electronic and Thermodynamic Profiling.
12. Computational Conformational Analysis of Substituted Cyclohexanes.
13. Cheminformatics and ADME Profiling of Drug Candidates.

Recommended Tools

MolView, Avogadro, PubChem, ChemDraw, Google Colab, Python, RDKit, Jupyter Notebook, MATLAB, ThingSpeak, Arduino IDE, Plotly, Materials Project, VESTA, Gaussian, OriginPro, ImageJ, NanoHub, TensorFlow, PyTorch.



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakagalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Assessment Scheme

Integrated Professional Core Courses (Theory + Practical)

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT) every 2 weeks (Minimum 06 assessments)	20
Mid Semester Examination (MSE)	20
Lab Record	10
Lab Examination	10
Semester end exam	40
Total	100 marks

Suggested Reading

Primary Textbooks

1. Engineering Chemistry – I: Concepts and Applications, Jit Chakraborty, Wiley India Pvt. Ltd.
2. Engineering Chemistry, Shikha Agarwal, Cambridge University Press.
3. Molecular Modelling for Beginners, Alan Hinchliffe, Wiley, 2008.
4. Engineering Chemistry, Jain & Jain, Dhanpat Rai & Co.
5. Dayananda Sagar University laboratory manual.

Additional Resources

1. NPTEL: Database Management Systems – IIT Madras / IIT Kharagpur
2. Coursera: Database Systems Specialization by University of Michigan
3. NPTEL: Engineering Chemistry – IIT Madras
4. NPTEL: Electrochemistry / Sensors – IIT Kharagpur
5. NDLI: National Digital Library of India – IIT Kharagpur
6. LibGen: Online Academic Resource Repository
7. YouTube: Engineering Chemistry Lectures – NPTEL / IIT Channels
8. YouTube: Electrochemistry and Sensors – Educational Lectures
9. YouTube: Polymers and Materials Science – Concept Videos
10. YouTube: Photovoltaic Cells and Green Energy – Lecture Series
11. YouTube: Optoelectronic Devices and Nanomaterials – Tutorials



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Strength of CO Mapping to PO/PSOs:

Table: Mapping Levels of Cos to Pos / PSOs													
COs	Program Outcomes (POs)											PSOs	
	1	2	3	4	5	6	7	8	9	10	11	1	2
C01	3	2	1	-	-	-	-	-	-	-	-	-	-
C02	2	3	2	-	-	-	-	-	-	-	-	-	-
C03	1	2	3	1	-	-	-	-	-	-	-	-	-
C04	2	2	3	-	-	-	-	-	-	-	-	-	-
C05	2	2	2	3	-	-	-	-	-	-	-	-	-
Avg	0.66	0.66	0.66	0.26	-	-	-	-	-	-	-	-	-

Legend: 3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Engineering Chemistry for Electronics cluster

Course Code: 26XXXX	Semester: I/II
Programs Offered: B-Tech ECE	Prerequisites: Nil
Course Type: Basic Science (BSC)	Hours/Week: 4 h (2 Theory Lab + 2 Lab)
Total Hours: (30 T + 30 P)	Credits: 3
L – T – P – J: 2-0-2-0	

Course Vision

To enable students to acquire knowledge on principles of chemistry for electrical engineering applications. To enable students, use the computational tools for molecular modelling, structure building, and basic property prediction. To develop an intuitive understanding of chemistry by emphasizing the related branches of engineering. To provide students with a solid foundation in analytical reasoning required to solve societal problems.

Course Objectives

Students will acquire knowledge on principles of chemistry for engineering applications and use the computational tools for molecular modelling, structure building, and basic property prediction.

It develops an intuitive understanding of chemistry by emphasizing the related branches of engineering, which provide students with a solid foundation in analytical reasoning required to solve societal problems.

Course Outcomes (CO)

CO	Outcome	BTL
CO1	To understand the fundamentals of polymers, energy band theory, semiconductors and PCB electroless plating processes used in electronic materials.	L2
CO2	To learn the basics of computational chemistry and perform molecular modelling, structural optimization and HOMO–LUMO analysis using software tools.	L3



C03	To study optoelectronic materials and display technologies such as LCD, OLED and QLED with emphasis on the role of band gap in light emission	L2
C04	To understand electrochemical systems and apply electrode principles for electrochemical measurements and sensor-based analytical techniques.	L3
C05	To gain knowledge of energy storage and conversion systems including batteries, fuel cells, solar cells and hydrogen energy technologies.	L2

Broad Course Description

This course introduces fundamental principles of engineering chemistry with emphasis on electronic materials, semiconductors, polymers, and electrochemical systems. Engineering Chemistry is a foundational course that provides students with essential chemical principles relevant to engineering applications. It covers polymers, conducting materials, graphene oxide, semiconductor fundamentals, energy band theory, and electroless copper plating for PCB fabrication. The course introduces computational chemistry tools for molecular modelling, structure optimization, Z-matrix construction, and prediction of electronic properties such as HOMO–LUMO energy gap. It also includes optoelectronic materials and display technologies like liquid crystals, OLEDs, and QLEDs, emphasizing the role of band gap. Electrochemical systems, galvanic cells, electrodes, and sensor technologies are studied along with analytical techniques. The course further addresses energy storage and conversion systems including modern batteries, fuel cells, solar PV cells, organic solar cells, and hydrogen energy production methods.

Course Modules

Module 1: Electronic Materials and Semiconductor Fundamentals (06 Hours)

Polymers: Introduction, Molecular weight - number average, weight average and numerical problems. Conducting polymers, Conducting mechanism of polyacetylene. (PPy, PANI, PEDOT: PSS – synthesis, properties and applications.

Electrical Engineering Materials: Introduction, classification of materials based on band gap (conductors, semiconductors and insulators), principle of conductors, insulators and semiconductors with examples. Energy Band Theory: Valence band, conduction band and band gap;

Semiconductors: Introduction, production of electronic grade silicon, Czochralski process (CZ) and Float Zone (FZ) methods. **PCB:** Electroless plating – Introduction, Electroless plating of copper in the manufacture of double-sided PCB.

Engineering Applications

Material selection for semiconductors and conductors in microelectronics. Use of conducting polymers in flexible sensors and electromagnetic shielding. Production of electronic grade silicon wafers for IC fabrication. Copper plating in PCB fabrication for consumer and industrial



electronics.

Module 2: Computational Chemistry and Electronic Structure Modelling (06 Hours)

Chemical Bonding Fundamentals and Bonding Models (Lewis Structures, VSEPR, VBT, and MOT). Scope of Computational Chemistry: Quantum Mechanics vs Molecular Mechanics vs Experimental Chemistry. Essential Chemical Data Formats for Small Molecules (Z-matrix, PDB, XYZ), DFT Basics: Functionals, Basis Sets, and Electron Correlation, Band Gap Engineering, Global Reactivity Descriptors for Drugs and Materials Screening, IR Spectroscopy & Thermochemistry in Computational Chemistry, Fundamentals of Classical MD Simulations and Thermodynamic Ensembles in MD Simulations.

Hands-on experience:

Hands-On: Molecular Modeling with Avogadro, QM Software: DFT Calculations with GAMESS, MD Software: Biomolecular Simulations with NAMD.

Engineering Applications

Computational prediction of molecular geometry and electronic structure of materials. Modeling HOMO–LUMO gaps to estimate electrical/optical behavior of semiconductors. Simulation of combustion thermodynamics for engine design optimization.

Module 3: Optoelectronic Materials and Display Technologies (06 Hours)

Display Systems: Photoactive and electroactive materials used in optoelectronic devices-organic and nanomaterials. Jablonski Diagram.

Liquid crystals (LC's): Introduction, classification, properties and applications in Liquid Crystal Displays (LCD's).

Organic Light Emitting Diodes (OLED's) – Properties, working principle and application of OLED's and Quantum Light emitting diodes (QLED's)- Properties, working principle and application of QLED's. Role of band gap in optoelectronic materials: Influence of band gap on light absorption and emission in OLEDs and QLEDs

Engineering Applications

Use of liquid crystals in LCD displays for TVs, monitors and instrumentation. Design of OLED and QLED displays in consumer electronics. Engineering materials with tuned band gaps for photonic and optoelectronic devices

Module 4: Electrochemical Systems and Sensor Technologies (06 Hours)

Electrode System: Introduction, types of electrodes. Ion selective electrode – definition, construction, working and applications of glass electrode. Determination of pH using glass electrode. Galvanic Cells and its types. Numerical problems based on EMF of a cell. Thermodynamic basis of electrochemical cells: Relationship between EMF, Gibbs free energy and equilibrium constant. **Sensors:** Introduction, working principle, types and applications of



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Electrochemical sensors. **Analytical Techniques:** Introduction, principle and instrumentation of Colorimetric sensors; its application in the estimation of copper. Potentiometric sensors; its application in the estimation of iron. Conductometric sensors; its application in the estimation of weak acid.

Engineering Applications

Design of pH meters and ion-selective electrodes for process control. Sensors for metal ion estimation (Cu, Fe) in industrial and environmental monitoring. Conductometric and potentiometric sensors in automation and quality testing.

Module 5: Energy Storage and Conversion Systems for Electronics (06 Hours)

Batteries: Introduction, classification of batteries. Components, construction, working and applications of modern batteries; Na-ion battery and Lithium ion batteries. **Fuel Cells:** Introduction, construction, working and applications of methanol-oxygen fuel cell, advantages and disadvantages of fuel cells. **Solar Energy:** Introduction, working principle, application of solar PV cell. Introduction, construction and working of organic solar cell (OSC). **Green fuel:** Definition, Hydrogen energy- types and method of production of hydrogen (Photo catalytic water splitting).

Engineering Applications

Development of Li-ion and Na-ion batteries for portable electronics and EVs. Implementation of fuel cells for clean energy and backup power systems. Solar PV systems for renewable electricity generation. Hydrogen production for green fuel and energy storage systems.

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

Module-1: Use ML regression to predict number-average (M_n) and weight-average (M_w) molecular weights from polymerization conditions. Solve numerical problems using AI-assisted tools (Python notebooks, symbolic solvers). Predict polydispersity index (PDI) trends based on synthesis parameters. Classify materials (conductors, semiconductors, insulators) using ML based on band gap datasets. Predict band gap using AI models trained on materials databases. Visualize direct vs indirect band gaps and correlate with optoelectronic properties.

Module-2: Use AI to map theoretical models → computational simulations → experimental predictions. Automate generation of Lewis structures, bonding types, and valence electron counts using cheminformatics tools like MarvinSketch and Avogadro. Apply ML models to predict molecular stability and geometry from simple descriptors. Compute HOMO-LUMO gaps using computational tools and correlate with material band gaps. Use ML regression to predict electronic properties from molecular descriptors.



Module-3: Use ML models to predict optical absorption, emission wavelength, and charge mobility of organic semiconductors and nanomaterials. Screen materials (polymers, quantum dots, perovskites) using AI for desired optoelectronic performance. Analyze structure–property relationships (conjugation length, defects, size effects). Use ML regression to correlate band gap $\leftrightarrow$ absorption/emission wavelength. Predict material suitability for OLEDs/QLEDs based on band gap values. Analyze direct vs indirect band gap impact on light emission.

Module-4: Simulate electrode potentials and pH response curves using ML regression. Use AI-enabled calibration of glass electrodes (drift correction, noise filtering). Predict pH from sensor voltage data using data-driven models. Use ML to correlate EMF $\leftrightarrow$ Gibbs free energy $\leftrightarrow$ equilibrium constant. Solve numerical problems using AI-assisted computation tools. Predict feasibility of reactions based on thermodynamic data. Use AI to analyze absorbance data and generate calibration curves. Predict concentration using regression models.

Module-5: Predict capacity, cycle life, and degradation of Li-ion/Na-ion batteries using ML on charge–discharge data. Optimize electrode materials (graphite, LFP, NMC, hard carbon) via data-driven screening. Use AI to model SEI formation and thermal behavior, Screen donor–acceptor pairs by predicting HOMO–LUMO levels, absorption spectra, and charge mobility. Use ML to correlate morphology (phase separation) with efficiency. Predict electrocatalyst/photocatalyst activity (HER/OER) using ML on materials descriptors. Model photo-catalytic water splitting efficiency via band alignment and kinetics. Optimize electrolysis conditions for yield and energy efficiency.

Authentic Intelligence Reflection:

Module-1: AI integration enables prediction of polymer properties, electronic behavior, and material performance through data-driven models. AI² approaches extend this to autonomous materials discovery, smart manufacturing, and real-time process optimization. This connects chemistry with electronics, nanotechnology, and industry, enabling efficient design of advanced materials for next-generation technological applications.

Module-2: AI integration enhances computational chemistry by enabling rapid structure generation, optimization, and property prediction. AI² approaches extend this to surrogate quantum models, automated simulations, and materials discovery. This bridges theoretical chemistry with real-world applications in energy, environment, and electronics, fostering computational thinking and data-driven scientific problem-solving skills.

Module-3: AI integration enables prediction of optical properties, simulation of photophysical processes, and optimization of optoelectronic materials. AI² approaches extend this to



generative material design, band gap engineering, and intelligent device optimization. This connects chemistry with electronics and photonics, enabling efficient development of advanced display and energy technologies.

Module-4: AI integration enhances electrochemistry and analytical techniques by enabling accurate prediction of pH, EMF, and analyte concentrations through data-driven models. AI² approaches extend this to smart sensors, automated analysis, and real-time monitoring systems, connecting chemistry with healthcare, environmental sensing, and industrial process optimization for efficient and intelligent solutions.

Module-5: AI enables predictive modeling of batteries, fuel cells, solar devices, and hydrogen systems, improving efficiency and lifetime. AI² extends this to generative materials design, real-time control, and closed-loop optimization. The outcome is a scalable, data-centric pathway to cleaner energy technologies, integrating chemistry, materials science, and intelligent systems for sustainable deployment.

Laboratory Component

The lab emphasizes **hands-on Experience and visualization.**

Lab Experiments

1. Potentiometric estimation of FAS using standard Potassium dichromate solution.
2. Determination of the viscosity of a given polymer.
3. Colorimetric determination of copper.
4. Conductometric estimation of an acid mixture using standard NaOH solution.
5. Electrolysis of water.

Computational experiments

6. Computational Investigation of Ethylene Carbonate-Li⁺ Complexes: Structural and Vibrational Properties in Lithium-Ion Batteries
7. Molecular Dynamics Simulation and Structural Analysis of Solvated Lysozyme
8. Computational IR Spectroscopy of Pharmaceuticals: Analyzing Paracetamol and Phenacetin
9. Computational Studies of Chain Length Effects on the Optoelectronic Properties of PEDOT
10. Computational Investigation of the Pharmacophore Quinoline: Electronic and Thermodynamic Profiling
11. Computational Conformational Analysis of Substituted Cyclohexanes
12. Computational Benchmarking of Thermodynamic Properties in Alkane Combustion



**SCHOOL OF
ENGINEERING**

Recommended Tools

Molecular Modeling with Avogadro, QM Software: DFT Calculations with GAMESS, MD Software: Bio molecular Simulations with NAMD

Assessment Scheme

Integrated Professional Core Courses (Theory + Practical)

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT) every 2 weeks (Minimum 06 assessments)	20
Mid Semester Examination (MSE)	20
Lab Record	10
Lab Examination	10
Semester end exam	40
Total	100 marks

Primary Textbooks

1. The Engineering Chemistry by Prasanta Rath, Engineering Chemistry by P.C. Jain & M. Jain, Physical Chemistry of Semiconductor Materials and Processes by Sergio Pizzin)
2. Frank Jensen, Introduction to Computational Chemistry: Chapter 1 & 2, Christopher Cramer, Essentials of Computational Chemistry, Chapter 4)
3. S. O. Pillai, Solid State Physics and Material Science.
4. Electrochemical System (3rd Edition) - John Newman and Karen E. Thomas-Alyea.
5. Engineering Materials for Efficient Energy Storage and Conversion by Shukla et al., Fundamentals of Biofuels Engineering and Technology (Green Energy and Technology) by Cataldo De Blasio.

Additional Resources

[illegible]



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka - 562 112

C05	2	2	2	3	-	-	-	-	-	-	-	-	-
Avg	0.66	0.66	0.66	0.26	-	-	-	-	-	-	-	-	-

1. NPTEL: Fundamental Engineering Chemistry – IIT Madras / IIT Kharagpur

3: Substantial (High)

2: Moderate (Medium)

1: Poor (Low)



Engineering Chemistry for Mechanical Cluster

Course Code: 26XXXX	Semester: I/II
Programs Offered: BTech -Mechanical and Aerospace Engineering	
Prerequisites: Nil	Course Type: Basic Science (BSC)
Hours/Week: 4 h (2 Theory Lab + 2 Lab)	Total Hours: (30 T + 30 P)
L – T – P – J: 2-0-2-0	Credits: 3

Course Vision

To develop engineers with strong chemical foundations and AI-enabled problem-solving skills for energy, materials, and environmental applications. The course envisions integrating theoretical knowledge with computational tools to design sustainable technologies, optimize industrial processes, and foster innovation in mechanical and aerospace systems through data-driven and interdisciplinary approaches.

Course Objectives

Students will:

1. Develop a strong foundation in fundamental chemistry concepts relevant to mechanical engineering applications such as energy systems, materials, corrosion, and environmental processes.
2. Enable students to understand and apply chemical principles in solving real-world engineering problems related to fuels, materials performance, and analytical techniques.
3. Introduce modern energy systems including batteries, fuel cells, and renewable energy technologies with a focus on sustainability.
4. Provide knowledge of material science, polymers, and surface engineering for industrial and technological applications.
5. Integrate AI and computational tools to model, simulate, and analyze chemical systems and engineering processes.
6. Enhance analytical, problem-solving, and critical thinking skills through numerical problems, experimental techniques, and data interpretation.



Course Outcomes (CO)

CO	Outcome	Bloom's Taxonomy Level
C01	Explain fundamental concepts of green fuels and energy devices (Li-ion, fuel cells), and solve numerical problems on calorific value and combustion stoichiometry.	L2
C02	Apply principles of computational chemistry tools (DFT, MD) to simulate molecular structures, thermodynamic behavior, and analyze properties of engineering systems.	L3
C03	Solve analytical problems related to water hardness and COD, and evaluate the synthesis, properties, and applications of industrial and conducting polymers.	L2
C04	Evaluate structure-property relationships in materials, and assess processing techniques for metals, ceramics, composites, and smart materials.	L2
C05	Analyze electrochemical corrosion mechanisms and evaluate surface modification techniques like electroplating and anodization for industrial corrosion control.	L2

Broad Course Description

This Engineering Chemistry course provides foundational knowledge of chemical principles tailored to mechanical and aerospace engineering applications. It covers energy systems (fuels, batteries, fuel cells, photovoltaics), corrosion science, polymers and advanced materials, analytical techniques, and e-waste management, emphasizing real-world industrial relevance. The course integrates computational chemistry, data analytics, and machine learning tools to model, simulate, and optimize chemical systems. Students gain hands-on experience with Python-based platforms, molecular modeling software, and AI-driven prediction techniques to analyse material properties, energy device performance, and environmental processes.

A strong focus is placed on problem-solving, numerical analysis, and experimental validation supported by AI tools. Through AI-assisted learning, virtual labs, and mini-projects, students develop critical thinking and interdisciplinary skills required for modern engineering challenges. The course ultimately bridges chemistry with emerging technologies, enabling students to design sustainable solutions in energy, materials, and manufacturing systems while understanding the capabilities and limitations of AI in scientific applications.



Course Modules

Module 1: Energy Source, Conversion and Storage (06 Hours)

Fuels: Introduction, calorific value, determination of calorific value using bomb calorimeter, numerical problems on GCV and NCV.

Combustion of fuels: Definition, complete and incomplete combustion, combustion reactions of fuels. Stoichiometric air-fuel ratio, excess air and their significance. Numerical problems on stoichiometric calculations of combustion reactions.

Green fuels: Introduction, power alcohol, synthesis and applications of biodiesel.

Hydrogen Fuel: Introduction, Blue Hydrogen, Green Hydrogen and other types of Hydrogen fuel

Energy devices: Introduction, construction, working, and applications of Li-ion, Na-ion, Li-MnO₂ battery, methanol-oxygen fuel cell and new technologies.

Engineering Applications: Energy systems form the backbone of mechanical engineering applications including IC engines, hybrid vehicles, power plants, and renewable energy systems. Understanding calorific values enables efficient fuel selection and engine design. Hydrogen fuel technologies are directly linked to next-generation clean energy vehicles and industrial decarbonization. Batteries and fuel cells are critical in electric vehicles, grid storage, and portable electronics. Photovoltaic technologies are essential for solar power generation systems used in smart grids and sustainable infrastructure.

Module 2: Introduction to Computational Chemistry (06 Hours)

Chemical Bonding Fundamentals and Bonding Models (Lewis Structures, VSEPR, VBT, and MOT), Scope of Computational Chemistry: Quantum Mechanics vs Molecular Mechanics vs Experimental Chemistry, Essential Chemical Data Formats for Small Molecules (Z-matrix, PDB, XYZ), DFT Basics: Functionals, Basis Sets, and Electron Correlation, Band Gap Engineering, Global Reactivity Descriptors for Drugs and Materials Screening, IR Spectroscopy & Thermochemistry in Computational Chemistry, Fundamentals of Classical MD Simulations and Thermodynamic Ensembles in MD Simulations.

Hands-on experience:

- (a) Molecular Modeling with Avogadro, QM Software: DFT Calculations with GAMESS, MD Software: Biomolecular Simulations with NAMD.
- (b) Simulation of combustion of petrol and diesel using the ChemCompute online tool.

Engineering Applications: Computational chemistry enables engineers to model molecular structures, predict thermodynamic behavior of fuels, and analyze electronic properties such as HOMO-LUMO and band gaps. These insights support design of efficient combustion systems,



advanced materials, semiconductors, and energy devices, reducing experimental cost and enabling faster innovation in mechanical and aerospace engineering applications.

Module 3: Macromolecules for Engineering Applications (06 Hours)

Polymers: Introduction, types of polymerization (Addition and Condensation). Techniques of addition and condensation polymerization, molecular weight; number average and weight average, numerical problems. Synthesis, properties and industrial applications of polyvinylchloride (PVC) and polystyrene. Conducting polymers – synthesis, mechanism and applications of P3HT, Polyaniline, Polypyrrole and PEDOT:PSS. **Plastics:** Introduction, synthesis, properties and industrial applications of poly(methyl methacrylate) (PMMA) and Teflon. **Polymers for water treatment:** Introduction, sources and nature of impurities of water, hardness of water, types of hardness, determination of temporary, permanent and total hardness and numerical problems. Determination of total hardness by EDTA method. Softening of water by Lime-Soda Process, determination of COD, numerical problems.

Engineering Applications: Polymers are widely used in automotive components, aerospace materials, packaging, biomedical devices, and water purification systems. Conducting polymers are essential in sensors, flexible electronics, and energy storage devices. Membrane technologies are critical in desalination plants and wastewater treatment.

Module 4: Materials Science and Engineering (06 Hours)

Introduction: Definition, Importance and classification of materials science and engineering (metals, ceramics, polymers, composites). **Structure of Materials:** Crystal structures, Imperfections in crystals, Amorphous structures, Relationship between structure and properties. **Material Processing:** Metal casting, forging, and machining. Ceramic processing (sintering, pressing) Polymer processing (molding, extrusion) Composite fabrication **Advanced Materials and Applications:** Smart materials (shape memory alloys, piezoelectric materials).

Engineering Applications: Material selection is critical in mechanical design, manufacturing, and product development. Understanding structure-property relationships helps in designing stronger, lighter, and more durable materials. Smart materials are used in sensors, actuators, and adaptive systems in robotics and aerospace.

Module 5: Corrosion Science and Engineering (06 Hours)

Corrosion: Introduction, electrochemical theory of corrosion, types of corrosion differential metal and differential aeration stress corrosion. Factors affecting corrosion (EMF, Temperature, pH, relative area of anode and cathode and polarization). **Corrosion control:** Corrosion control – Metal coating-anodic coating- galvanization, Inorganic coating-anodization and Cathodic protection-sacrificial anode and impressed current method. Corrosion testing by weight loss method. **Surface modification techniques:** Introduction, technological



importance. Anodization of metal alloys, Electroplating and Electroless plating: Definition, Electroplating of chromium and electroless plating of nickel, Advantages of electroplating techniques in Automobile and Aerospace industries.

Engineering Applications: Corrosion affects mechanical components such as pipelines, turbines, marine structures, and automotive parts. Corrosion prevention techniques are critical in aerospace, oil & gas, and infrastructure industries. Surface coatings improve durability, reduce maintenance costs, and enhance material performance under harsh environmental conditions.

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration

Module-1 (Energy Source, Conversion and Storage): AI and machine learning models can be used to predict calorific values and optimize stoichiometric air-fuel combustion ratios. Students can explore AI-driven battery management systems (BMS) that predict the state of health and degradation of Li-ion and Na-ion batteries, as well as AI applications in discovering novel catalysts for Green Hydrogen production.

Module-2 (Introduction to Computational Chemistry): Integration of AI with computational tools like GAMESS and NAMD to accelerate the prediction of global reactivity descriptors, band gaps, and thermodynamic properties. Students explore how machine learning algorithms screen massive chemical databases for drug and material design, drastically reducing the time needed for traditional DFT and MD simulations.

Module-3 (Macromolecules for Engineering Applications): AI-based prediction of polymer properties (such as molecular weight distribution and conductivity of polymers like P3HT and PEDOT:PSS). Students can also explore data-driven models and AI IoT sensors used in modern water treatment plants to monitor water hardness, COD levels, and optimize the lime-soda softening process in real-time.

Module-4 (Materials Science and Engineering): Explore, analyze, and predict materials using computational and AI-driven platforms like the Materials Project (<https://next-gen.materialsproject.org/>). AI algorithms are used to model structure-property relationships, predict the behavior of imperfections in crystals, and optimize processing parameters for metal casting, ceramic sintering, and polymer molding.

Module-5 (Corrosion Science and Engineering): Utilizing computer vision and AI-driven image analysis to detect, classify, and predict the progression of localized corrosion (like stress corrosion or pitting). Students explore predictive modeling tools that calculate corrosion rates based on environmental variables (EMF, pH, temperature) and AI systems that optimize electroplating and electroless plating parameters for aerospace and automotive industries.



Authentic Intelligence Reflection:

Module-1: Students connect theoretical combustion and thermodynamics to the real-world global energy transition. By evaluating traditional fuels against green hydrogen and advanced batteries, learners develop critical thinking and decision-making skills essential for designing sustainable, decarbonized mechanical and automotive systems.

Module-2: This module builds authentic intelligence by transitioning students from traditional wet-lab chemistry to modern computational problem-solving. By utilizing tools like Avogadro, GAMESS, and NAMD, learners develop computational thinking, data interpretation skills, and the ability to critically evaluate the accuracy of virtual models against experimental reality.

Module-3: Fosters practical, environmentally conscious engineering intelligence. Students learn to critically evaluate and select appropriate polymers based on their distinct synthesis and properties for applications ranging from flexible electronics to packaging. Furthermore, tackling water hardness and COD calculations grounds their knowledge in vital real-world public health and environmental sustainability practices.

Module-4: Cultivates deep engineering intuition by teaching students to link invisible atomic structures and imperfections to macroscopic, tangible material performance. It encourages innovative and adaptive thinking, enabling students to logically justify the selection of advanced and smart materials for complex aerospace, robotics, and manufacturing challenges.

Module-5: This module develops real-world engineering foresight by focusing on material degradation, safety, and lifecycle economics. Students build analytical reasoning by diagnosing the root causes of electrochemical corrosion and strategically selecting the most effective surface modification and prevention techniques, directly mirroring the responsibilities of industrial and infrastructure engineers.

Laboratory Component

The lab emphasizes **hands-on Experience and visualization.**

Lab Experiments

1. Evaluation of the quality of water in terms of total hardness by complex metric titration.
2. Determination of the viscosity of a given polymer.
3. Determination of alkalinity of the given water sample
4. Electrolysis of water.
5. Estimation of the weight loss of iron by corrosion of an iron nail.



Computational experiments

6. Computational Investigation of Ethylene Carbonate-Li⁺ Complexes: Structural and Vibrational Properties in Lithium-Ion Batteries
7. Molecular Dynamics Simulation and Structural Analysis of Solvated Lysozyme
8. Computational IR Spectroscopy of Pharmaceuticals: Analyzing Paracetamol and Phenacetin
9. Computational Studies of Chain Length Effects on the Optoelectronic Properties of PEDOT
10. Computational Investigation of the Pharmacophore Quinoline: Electronic and Thermodynamic Profiling
11. Computational Conformational Analysis of Substituted Cyclohexanes
12. Computational Benchmarking of Thermodynamic Properties in Alkane Combustion

Recommended Tools

Molecular Modeling with Avogadro, QM Software: DFT Calculations with GAMESS, MD Software: Biomolecular Simulations with NAMD

Assessment Scheme

Integrated Professional Core Courses (Theory + Practical)

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT) every 2 weeks (Minimum 06 assessments)	20
Mid Semester Examination (MSE)	20
Lab Record	10
Lab Examination	10
Semester end exam	40
Total	100 marks

Primary Textbooks

1. A Text book of Engineering Chemistry, SS Dara & Dr. SS Umare, S Chand & Company Ltd., 12th Edition, 2011.
2. A Text Book of Engineering Chemistry, R.V. Gadag and Nityananda Shetty, I. K. International Publishing house. 2nd Edition, 2016.
3. Text Book of Polymer Science, F.W. Billmeyer, John Wiley & Sons, 4th Edition, 1999.



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

4. Nanotechnology: A Chemical Approach to Nanomaterials, G.A. Ozin & A.C. Arsenault, RSC Publishing, 2005.
5. Corrosion Engineering, M. G. Fontana, N. D. Greene, McGraw Hill Publications, New York, 3rd Edition, 1996.
6. Electrochemical System (3rd Edition) - John Newman and Karen E. Thomas-Alyea.
7. Engineering Materials for Efficient Energy Storage and Conversion by Shukla et al., Fundamentals of Biofuels Engineering and Technology (Green Energy and Technology) by Cataldo De Blasio.
8. Frank Jensen, Introduction to Computational Chemistry: Chapter 1 & 2, Christopher Cramer, Essentials of Computational Chemistry, **Chapter 4**)

Additional Resources

1. NPTEL: Fundamental Engineering Chemistry – IIT Madras / IIT Kharagpur



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakagalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Engineering Chemistry for Computer Science and Medical Engineering

Course Code: 26XXXX	Semester: I/II
Programs Offered: Computer Science and Medical Engineering	
Prerequisites: Nil	Course Type: Basic Science (BSC)
Hours/Week: 4 h (2 Theory Lab + 2 Lab)	Total Hours: (30 T + 30 P)
L – T – P – J: 2-0-2-0	Credits: 3

Course Vision

To develop a strong foundation in biochemistry by understanding the structure, properties, and biological roles of carbohydrates, proteins, lipids, and nucleic acids, while integrating modern chemoinformatics, molecular modelling, and AI/ML approaches. The course aims to equip students with the ability to analyse biomolecular systems, visualize and model molecular structures, and apply computational tools in drug design and materials discovery, preparing them for interdisciplinary research and innovation in life sciences and pharmaceutical sciences.

Course Objectives

Students will:

1. Acquire knowledge on the basics of biochemistry for medical engineering applications.
2. Develop an intuitive understanding of biochemistry by emphasizing the related branches of engineering.
3. Be provided with a solid foundation in modelling of molecules.
4. Understand the Principles of Drug Design and scoring function of small molecules.



Course Outcomes (CO)

CO	Outcome	Bloom's Taxonomy Level
CO1	Identify the terms and principles of Biochemistry involved in Medical Engineering and Technology	L1
CO2	Explain the phenomenon of Biochemistry to describe the methods of Synthesis and applications in daily life	L1
CO3	Solve the problems of Biochemistry that are pertinent in socio-economic and environmental issues.	L3
CO4	Apply the basic concepts of Biochemistry in quantification of industrially relevant parameters.	L3
CO5	Analyse the structure, properties and processes associated with chemical substances in multidisciplinary situations	L3

Broad Course Description

This course provides a comprehensive foundation in biochemistry integrated with modern chemoinformatics and computational approaches. It begins with the study of carbohydrates, covering their structural diversity, stereochemistry, functional roles, and significance as energy sources and informational molecules in biological systems. The course then advances to amino acids, proteins, lipids, and nucleic acids, emphasizing their chemical properties, structural organization, and biological functions, including protein folding, enzyme behaviour, and genetic information flow.

In parallel, the course introduces key concepts of molecular modelling, including the use of structural databases such as PDB and CSD, molecular visualization, and analysis of three-dimensional biomolecular structures. Students gain practical exposure to computational tools used in modelling and interpreting molecular interactions.

The course further explores the principles of drug discovery and design, including pharmacokinetics, molecular docking, pharmacophore modelling, and strategies for lead optimization, along with an understanding of drug delivery systems and challenges such as toxicity and resistance.

Finally, the course highlights the growing role of artificial intelligence and machine learning in drug and materials design, introducing concepts such as structure–activity relationships,



virtual screening, and predictive modelling techniques. Overall, the course bridges fundamental biochemical knowledge with advanced computational and AI-driven methodologies, preparing students for applications in pharmaceutical research, biotechnology, and materials science.

Course Modules

Module 1: Carbohydrate Structure, Function, and Biological Information Systems (06Hours)

Monosaccharides - structure of aldoses and ketoses, ring structure of sugars, conformations of sugars, mutarotation, anomers, epimers and enantiomers, structure of biologically important sugar derivatives, oxidation of sugars. Formation of disaccharides, reducing and non-reducing disaccharides.

Polysaccharides – homo- and heteropolysaccharides, structural and storage polysaccharides. Structure and role of proteoglycans, glycoproteins and glycolipids (gangliosides and lipopolysaccharides). (definition, structure, functions, examples only) Carbohydrates as informational molecules, working with carbohydrates (applications of carbohydrate)

Engineering Applications: Biomaterials and Tissue Engineering, Drug Delivery Systems, Food and Fermentation Technology, Biodegradable Materials and Packaging, Glycoengineering and Diagnostics, Energy and Biofuel Production, Pharmaceutical Formulations, Surface Coatings and Hydrogels, Nanotechnology Applications, Environmental Engineering

Module 2: Biomolecular Structure and Function: Proteins, Lipids, and Nucleic Acids (06 Hours)

Classification, chemical and physical properties of amino acids. Peptide bond: planarity and dihedral angles, Ramachandran plot. Structural hierarchy of proteins: primary, secondary, super-secondary, tertiary and quaternary structures. Classification & properties of proteins. Globular and fibrous proteins.

Fatty acids: saturated, unsaturated and essential fatty acids; rancidity, saponification number, iodine number, acid number and Reichert–Meissel number.

Chemistry of purine and pyrimidine, nucleosides and nucleotides. Types of DNA: structure and properties of A-, B- and Z-DNA.

Nucleotides - structure and properties. Nucleic acid structure – Watson-Crick model of DNA.

Structure of major species of RNA. Nucleic acid chemistry - UV

absorption, effect of acid and alkali on DNA. Other functions of nucleotides - source of energy, component of coenzymes, second messengers (examples & functions only)

Engineering Applications: genetic engineering, gene therapy, CRISPR technologies, DNA fingerprinting, forensic analysis, mRNA vaccines, therapeutic development, tissue engineering, biodegradable polymer synthesis, production of pharmaceuticals.



Module 3: Molecular Data Systems and Computational Modelling (06 Hours)

Cambridge Structural Database (CSD), Protein Data Bank (PDB), File format and information in PDB and CSD databases; Molecular Modelling: Molecular Graphics, 3-D Models of Organics and Biomolecules, Potential Energy Surfaces;

Engineering Applications: Protein Structure Analysis, Computational Chemistry, Material Science and Nanotechnology, Drug Design and Discovery, Molecular Visualization, Chemical and Process Engineering, Environmental Engineering

Module 4: Computational Drug Design and Molecular Docking (06 Hours)

The Principles of Drug Design: The Drug Discovery Pipeline and Costs of Drug Discovery, Lipinski's Rules of 5, Drug Metabolism, Toxicity and Side Effects, ADMET (Absorption, Distribution, Metabolism, Excretion, Toxicity) Models, Drug resistance and promiscuity, Drug Delivery systems, Pharmacophore Modeling and alignment, Lead Optimization, Ligand Based and Structure-Based Design; Molecular Docking.

Engineering Applications: Rational Drug Design, Molecular Docking and Simulation, ADMET and Pharmacokinetics, Controlled Drug Delivery Systems, Nanotechnology in Drug Delivery, Industrial and Pharmaceutical Applications

Module 5: AI/ML-Driven Molecular Design and Predictive Modelling (06 Hours)

Application of AI/ML in Drug and Materials Design: Introduction to Linear Free Energy Relationships, Structure-Activity Relationships and the Similarity Principle, Virtual High throughput Screening, Feature selection – Genetic Algorithms, Model Validation.

Engineering Applications: Structure-Activity Relationship (SAR) Modelling, De Novo Drug Design, Materials Design and Discovery, Feature Selection and Optimization, Personalized Medicine.

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

• AI Integration:

Module-1: Students can integrate AI tools to understand carbohydrate structures, properties, biological roles, and applications. Use MolView, ChemDraw, Avogadro, or AI-supported molecular viewers to visualize aldoses, ketoses, cyclic sugar forms, anomers, epimers, and enantiomers. Use AI-assisted concept maps to relate carbohydrates to energy storage, cell recognition, immunity, signalling, glycoproteins, glycolipids, and proteoglycans. Introduce AI-based analysis of carbohydrate databases to study glycan structures, disease biomarkers, and carbohydrate-protein interactions.



Module-2: Use AI-enabled molecular visualization tools (e.g., PyMOL, AlphaFold) to explore amino acid structures, zwitterions, and peptide bond planarity. Predict protein folding and stability using AI (AlphaFold-based structure prediction). Analyse Ramachandran plots via ML tools to understand allowed conformations. Classify proteins (globular/fibrous) using AI clustering techniques. Visualize DNA/RNA structures using UCSF Chimera or Jmol. Use AI models to compare A-, B-, Z-DNA conformations. Simulate denaturation/renaturation and UV absorption profiles. Analyse nucleotide sequences using AI-based bioinformatics tools.

Module-3: Explore structures from Cambridge Structural Database and Protein Data Bank using AI-assisted search (structure similarity, substructure matching). Use AI tools to interpret PDB file formats (ATOM/HETATM records, coordinates, occupancy, B-factors). Automated extraction of structural parameters (bond lengths, angles, torsions) using ML scripts (Python + RDKit/BioPython).

Module-4: Use AI platforms (e.g., DeepChem, KNIME) to simulate the drug discovery pipeline and estimate timelines/costs. Apply ML models to evaluate Lipinski's Rule of Five and predict drug-likeness. Predict ADMET properties (absorption, metabolism, toxicity) using AI-driven QSAR models. Analyse drug metabolism pathways and toxicity risks via predictive models. Model drug release kinetics and delivery efficiency using ML. Predict drug resistance mechanisms via pattern recognition in biological data.

Module-5: Use ML regression to model Linear Free Energy Relationships (LFER) (e.g., Hammett σ vs. reaction rates/equilibria). Build Structure–Activity Relationship (SAR) models linking molecular descriptors to activity (IC_{50} , EC_{50}). Apply similarity metrics (Tanimoto on fingerprints via RDKit) to cluster compounds and infer activity. Random Forest (RF): robust, interpretable (feature importance), handles nonlinearities. Partial Least Squares (PLS): effective for collinear descriptors; standard in QSAR. Support Vector Machine (SVM): strong performance with kernels for nonlinear SAR. Neural Networks (NN): capture complex patterns in large datasets.

.

Authentic Intelligence Reflection:

Module-1: AI integration helps students visualize complex carbohydrate structures, compare stereochemical relationships, and understand biological functions of glycans. By using AI tools for prediction, classification, and data analysis, learners connect carbohydrate chemistry with real-world applications in medicine, nutrition, biotechnology, diagnostics, and biomaterials, improving conceptual understanding and problem-solving skills.



Module-2: AI integration enables students to visualize biomolecular structures, predict protein folding, and analyse nucleic acid behaviour. AI² approaches extend learning to modeling mutations, predicting lipid stability, and simulating biological systems. This bridges chemistry with bioinformatics, drug discovery, food science, and systems biology, enhancing analytical thinking and interdisciplinary application skills.

Module-3: AI integration enables efficient exploration of structural databases, visualization of complex biomolecules, and prediction of molecular properties. AI² approaches extend this by enabling machine-learning-based energy predictions, structural analyses, and drug design. This transforms traditional molecular modelling into a data-driven, computationally intelligent framework for advanced research and interdisciplinary applications.

Module-4: AI integration accelerates drug discovery by enabling prediction of ADMET properties, molecular docking, and pharmacophore modelling. AI² approaches extend this to generative drug design, intelligent screening, and automated workflows. This transforms traditional medicinal chemistry into a data-driven, efficient, and scalable process with applications in pharmaceuticals, biotechnology, and personalized medicine.

Module-5: AI enables quantitative links between structure and function via LFER/SAR, accelerates virtual screening, and ensures rigorous validation. AI² extends this to automated pipelines, generative design, and interpretable ensemble models. The result is a scalable, data-centric framework for discovering drugs and materials with optimized performance and reduced experimental burden.

Laboratory Component

The lab emphasizes hands-on Experience and visualization.

Lab Experiments

1. Qualitative Tests for Carbohydrates by DNS method
2. Isolation of Genomic DNA from Plant Cells
3. Estimation of Protein by Lowry's Method
4. Separation and identification of amino acids by Paper chromatography
5. Study of enzyme activity and kinetics on effect of temperature and pH



Computational experiments

6. 3D Visualization and Structural Analysis of Carbohydrates Using Avogadro
7. In Silico Prediction of ADMET Properties Using Online Tools
8. Molecular Docking of Aspirin with COX Enzyme Using PyRx
9. Analysis and Comparison of Molecular File Formats: PDB, MOL, and CIF
10. Exploration of Protein Structures Using the Protein Data Bank (PDB) and PyMOL

Assessment Scheme

Integrated Professional Core Courses (Theory + Practical)

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT) every 2 weeks (Minimum 06 assessments)	20
Mid Semester Examination (MSE)	20
Lab Record	10
Lab Examination	10
Semester end exam	40
Total	100 marks

Primary Textbooks

1. Lehninger: Principles of Biochemistry (2013) 6th ed., Nelson, D.L. and Cox, M.M., W.H. Freeman and Company (New York), ISBN: 13: 978-1-4641-0962-1 / ISBN: 10:1-4292-3414-8.
2. Computational Chemistry: Introduction to the Theory and Applications of Molecular and Quantum Mechanics (2016) 3rd ed., E. G. Lewars. Springer Nature. ISBN: 978-3-319-30916-3.
3. An introduction to medicinal chemistry (2017) 6th ed., Patrick, G. L. Oxford University Press. ISBN-13: 9780198749691
4. Molecular descriptors for chemoinformatics (2009) 2nd ed., Wiley-VCH. ISBN: 978-3527318520

Additional Resources

1. NPTEL: Database Management Systems – IIT Madras / IIT Kharagpur
2. Coursera: Database Systems Specialization by University of Michigan



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka - 562 112

CO - PO / PSO Mapping

CO - PO / PSO Mapping													
COs	1	2	3	4	5	6	7	8	9	10	11	PO1	PO2
C01	3	2	1	-	-	-	-	-	-	-	-	-	-
C02	2	3	2	-	-	-	-	-	-	-	-	-	-
C03	1	2	3	1	-	-	-	-	-	-	-	-	-
C04	2	2	3	-	-	-	-	-	-	-	-	-	-
C05	2	2	2	3	-	-	-	-	-	-	-	-	-
Avg	0.66	0.66	0.66	0.26	-	-	-	-	-	-	-	-	-

3: Substantial (High)

2: Moderate (Medium)

1: Poor (Low)



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Engineering Graphics & Generative Design For Computer Science Cluster

(As per Choice Based Credit System)

Course Code: 26XXXXX	Semester: I/II
Programs Offered: B.Tech (CSE Cluster)	Prerequisites: NIL
Course Type: Engineering Science Course (ESC)	Software Platform: Autodesk Fusion 360
L – T – P – J: 2 – 0 – 2 – 0 (2 Credit Theory + 1 Credit Lab)	Programming Language: Python
Contact Hours: 60 Hours (Laboratory)	Credits: 3

Course Vision

This course introduces engineering graphics, parametric modeling, and generative design integrated with computational thinking and automation. It enables CSE students to bridge design and programming through CAD tools, AI-assisted workflows, and Fusion API-based automation.

Course Objectives

Students will be able to:

- To develop fundamental understanding of engineering graphics and visualization
- To enable creation of parametric 2D and 3D CAD models
- To introduce AI-driven generative design concepts
- To apply programming concepts for CAD automation using Fusion API
- To build computational design and automation skills



Course Outcomes (COs)

CO No.	Description	Bloom's Taxonomy Level
CO1	Apply engineering graphics principles and CAD tools to create accurate 2D sketches and technical drawings	L3
CO2	Develop parametric 3D models and assemblies using feature-based CAD modeling techniques	L3
CO3	Analyze and interpret generative design outputs for optimized and lightweight engineering solutions	L4
CO4	Implement CAD automation solutions using Python scripting and Fusion API tools	L3
CO5	Design and apply computational workflows for automated and data-driven CAD design systems	L3

Broad Course Description

This course introduces the fundamentals of engineering graphics, parametric modeling, and generative design with a strong emphasis on computational thinking and automation. It is specifically designed for CSE cluster students to bridge the gap between design and programming by integrating CAD tools, AI-assisted workflows, and API-based development.

Students begin with engineering drawing principles and 2D sketching, progressing to parametric 3D modeling and assemblies. The course then explores generative design, where AI-driven techniques are used to create optimized and innovative design solutions based on constraints and objectives. Further, students learn to program and automate CAD tasks using Fusion API and Python, enabling the development of intelligent design systems.

By the end of the course, students will be able to create, analyze, and automate design processes, develop data-driven and parametric models, and apply computational approaches to real-world engineering design problems. The course emphasizes both **AI integration** and **authentic intelligence**, ensuring students not only use intelligent tools but also critically interpret and validate their outputs.



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Detailed Course Syllabus

Module 1: Engineering Graphics & 2D Sketching (12 Hours)

Engineering graphics fundamentals, BIS standards, drawing conventions, Orthographic projections, sectional views, isometric views, CAD interface and sketch environment, 2D sketching: tools, constraints, dimensioning

Engineering Applications: Technical drawing and documentation

Module 2: 3D Modeling (12 Hours)

Parametric and feature-based modelling, Solid modeling: extrude, revolve, sweep, loft, Feature modifications and assemblies

Engineering Applications: Product design and digital prototyping

Module 3: Generative Design (12 Hours)

Generative design concepts and workflows, Constraints, materials, loads, Optimization and validation.

Engineering Applications: Lightweight and optimized structures

Module 4: Fusion API & CAD Programming (12 Hours)

Introduction to APIs and Fusion API architecture, Python scripting fundamentals for CAD, Accessing and modifying sketches, features, and parameters, Automating 2D/3D model creation using scripts, Event handling and UI basics in Fusion, Debugging and execution workflows

Engineering Applications: Automated CAD model generation, parametric script-based design, Custom CAD tool development

Module 5: Design Automation & Computational Design (12 Hours)

Design automation workflows and pipelines, Parametric and rule-based design systems, Generating design variants using code, Data-driven design (inputs → geometry), Introduction to add-in development, CAD + data + automation integration

Engineering Applications: Mass customization systems, automated design exploration tools
Smart CAD applications



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakagalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module-1: Use AI-assisted CAD and drafting tools such as Autodesk Fusion 360 and AutoCAD to generate, validate, and optimize engineering drawings, projections, and 2D sketches.

Module-2: Apply AI-enabled parametric modelling and assembly tools for intelligent feature creation, constraint management, interference detection, and rapid digital prototyping.

Module-3: Use AI-powered generative design platforms for topology optimization, lightweight structure development, material selection, and performance-based design validation.

Module-4: Integrate AI coding assistants and Python scripting tools to automate CAD modelling, feature generation, parametric modifications, and Fusion API workflows.

Module-5: Apply AI and computational design techniques to develop rule-based automation systems, generate design variants, and create intelligent CAD-driven workflows.

Authentic Intelligence Reflection:

Module-1: AI integration improves visualization, drafting accuracy, and technical documentation skills through intelligent CAD-based learning approaches.

Module-2: AI-assisted modelling enhances product development, assembly analysis, and digital prototyping skills aligned with modern engineering industries.

Module-3: AI integration enables data-driven optimization, sustainable design exploration, and computational engineering problem-solving.

Module-4: AI-supported CAD programming strengthens automation, scripting, computational thinking, and Industry 4.0 design capabilities.

Module-5: AI-driven computational design promotes smart automation, mass customization, and intelligent engineering workflow development for advanced digital manufacturing.

Laboratory Component (1 Credit)

The laboratory component focuses on Autodesk Fusion 360 and Python-based automation.



Lab Experiments (Indicative):

1. **2D Sketching:** Creating sketches with auto-dimensioning and sketch recognition tools.
2. **3D Modeling:** Developing parametric 3D models using extrude, revolve, and assembly constraints.
3. **Generative Design:** Applying AI-driven exploration based on materials, loads, and constraints.
4. **Fusion API Scripting:** Writing Python scripts to access and modify CAD parameters and automate model creation.
5. **Design Automation:** Building rule-based design systems and parametric product configurators.
6. **Advanced Automation:** Using AI-assisted code generation and debugging to optimize Python-based CAD scripts.

AI-Integrated Lab Exercises (Indicative)

- **Exercise 1 (Visualization):** Use AI-based visualization for orthographic, sectional, and isometric views.
- **Exercise 2 (Optimization):** Use cloud-based generative design tools for performance prediction and structural optimization.
- **Exercise 3 (Automation):** Implement AI-assisted code generation for CAD scripting and automate parametric modeling workflows.
- **Exercise 4 (Validation):** Critically evaluate AI-generated designs against engineering standards (e.g., BIS standards) and manufacturability.

The course emphasizes that students must **debug AI-generated scripts**, validate automation logic, and analyze trade-offs between performance and complexity

Assessment Scheme

Definitions

1. Internal Assessment (CIA) – 60 Marks

- a. One **Mid Semester Examination (MSE)** (CAD Tool-based, Laboratory Exercises),
- b. One **Continuous Assessment Tool (CAT)**: It may include: Surprise test (Computer-based), Quiz, Project based learning, Reflection Note on Guest Lecture, Industrial visit, E-course certification, Building models, Group discussion, Case study, Assignment, Seminar, or Open Book Assignment.
- c. One **AI-powered Assignment (APA)**: Use of AI tools for problem-solving while simultaneously developing Authentic Intelligence through critical evaluation, ethical reasoning, and responsible decision-making.



OR

d. **Mini Project:** Lightweight generative structural component design.

2. **“External Assessment”** refers to the Semester End Examinations (SEE). The weightage of this component is **40%**. At the end of semester, the final examination is conducted to evaluate a student's performance.

1. Internal Assessment (CIA) – 60 Marks

Component	Marks	Modules
One MSE	30	Module 1 & 2
CAT - Assignment on Generative Design	10	Module 3
AI-powered Assignment (APA) OR Mini Project	20	All Modules
Total	60	—

2. External Assessment (SEE) – 40 Marks

Component	Marks	Module
Exercise 1 Using CAD Tool	20	Module 4
Exercise 2 Using CAD Tool	20	Module 5
Total	40	

Recommended Tools

- CAD Software: Autodesk Fusion
- Simulation: Generative design tools, CAD simulation modules
- Programming: Python (Fusion API)
- AI Tools: ChatGPT, CAD plugins

Suggested Reading

Textbooks

1. K. Venugopal and V. Prabhu Raja, *Engineering Drawing and Graphics*, New Age International Publishers.
2. Sham Tickoo, *Autodesk Fusion 360 for Designers*, CADCIM Technologies.
3. Randy H. Shih, *Parametric Modelling with Autodesk Fusion 360*, SDC Publications.
4. Eric P. Sarriegu, *Fusion 360 API Programming*, Packt Publishing.



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Reference Books

1. Ibrahim Zeid, *CAD/CAM: Theory and Practice*, McGraw Hill.
2. Ian Stroud and Henry Nagy, *3D Modelling and CAD Systems*, Springer.
3. P. N. Rao, *CAD/CAM: Principles and Applications*, McGraw Hill Education.
4. Art Wetmore, *Mastering Autodesk Fusion 360*, Autodesk Press.
5. Wassim Jabi, *Parametric Design for Architecture*, Laurence King Publishing.
6. John M. Snyder and Alan H. Barr, *Ray Tracing Complex Models Containing Surface Tessellations*, ACM Press.

Additional Learning Resources

- NPTEL – Engineering Graphics and CAD
- Autodesk Learning Resources (Fusion 360 official tutorials)
- Coursera / edX / Udemy MOOCs on CAD, Generative Design, and Computational Design
- Research papers from IEEE Xplore / ScienceDirect (for advanced exploration)

Mapped NVIDIA DLI Course

Course Title: Fundamentals of Deep Learning OR Computer-Aided Engineering (Simulation & Digital Twins)

Focus: Understanding how neural networks and parallel processing (CUDA) accelerate design validation and generative iterations.

Relevance: This course provides the foundational knowledge needed to understand the "AI Logic" (Input → Weighted Sum → Activation) mentioned in your syllabus's AI Integration section.

DLI Learning Path for Engineering Design

This path is designed to transition students from traditional CAD to AI-accelerated generative workflows.



Strength of CO Mapping to PO/PSOs:

Table: Mapping Levels of Cos to POs / PSOs													
COs	Program Outcomes (POs)											PSOs	
	1	2	3	4	5	6	7	8	9	10	11	1	2
CO1	3	2	1	-	1	-	-	-	-	2	-	2	1
CO2	3	3	2	-	1	-	-	-	-	1	-	2	1
CO3	2	1	2	-	3	-	-	-	1	2	-	3	2
CO4	2	2	3	2	3	1	2	-	1	1	-	3	3
CO5	1	-	1	-	2	-	-	1	2	3	1	1	2
AVG	11/15	8/15	9/15	2/15	10/15	1/15	2/15	1/15	4/15	9/15	1/15	11/15	9/15

3: Substantial (High)

2: Moderate (Medium)

1: Poor (Low)



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Engineering Graphics & Generative Design For Electronics & Communication Engineering

(As per Choice Based Credit System)

Course Code: 26XXXXX	Semester: I/II
Programs Offered: B.Tech (ECE)	Prerequisites: NIL
Course Type: Engineering Science Course (ESC)	Software Platform: Autodesk Fusion 360
L – T – P – J: 2 – 0 – 2 – 0 (2 Credit Theory + 1 Credit Lab)	Programming Language: Python
Contact Hours: 45 Hours (Laboratory)	Credits: 3

Course Vision

To equip ECE students with strong visualization, CAD modeling, PCB design, and system integration skills, enabling them to design and develop intelligent electronic products using AI-assisted tools and modern engineering workflows.

Course Objectives

Students will be able to:

- Understand engineering graphics and drawing standards
- Develop 2D and 3D CAD models for electronic product design
- Apply generative design for optimized and lightweight structures
- Integrate ECAD and MCAD systems for real-world applications
- Design and develop PCB layouts for embedded and IoT systems



Course Outcomes (COs)

CO No.	Description	Bloom's Taxonomy Level
CO1	Apply engineering graphics principles, BIS standards, and CAD tools to create accurate 2D engineering sketches and technical drawings	L3 – Apply
CO2	Develop parametric 3D models, feature-based components, and assemblies for electronic and mechatronic product applications	L3 – Apply
CO3	Apply generative design workflows, material constraints, and optimization techniques to develop lightweight and efficient design solutions	L3 – Apply
CO4	Analyze and integrate ECAD and MCAD environments for embedded system design, electronic enclosure development, and co-design workflows	L4 – Analyze
CO5	Design, validate, and develop PCB layouts incorporating component placement, routing, design constraints, and manufacturing outputs	L3 – Apply

Broad Course Description

This course introduces engineering graphics fundamentals and progresses towards CAD-based product development tailored for ECE applications. Students begin with 2D sketching and visualization, followed by 3D modeling of enclosures and assemblies.

The course further explores generative design techniques for optimization and lightweight electronic structures. ECAD–MCAD integration enables students to design complete embedded systems by combining PCB layouts with mechanical enclosures.

Finally, students learn PCB design fundamentals including schematic capture, layout design, and manufacturing outputs. The course integrates AI tools for design automation, verification, and optimization, while emphasizing Authentic Intelligence for critical evaluation and decision-making.



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakagalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Detailed Course Syllabus

Module 1: Engineering Graphics & 2D Sketching (12 Hours)

Engineering graphics fundamentals, BIS standards, drawing conventions, Orthographic projections, sectional views, isometric views, CAD interface and sketch environment, 2D sketching: tools, constraints, dimensioning

Engineering Applications: Technical drawing and documentation

Module 2: 3D Modeling (12 Hours)

Parametric and feature-based modelling, Solid modeling: extrude, revolve, sweep, loft, Feature modifications and assemblies

Engineering Applications: Product design and digital prototyping

Module 3: Generative Design (12 Hours)

Generative design concepts and workflows, Constraints, materials, loads, Optimization and validation.

Engineering Applications: Lightweight and optimized structures

Module 4: ECAD-MCAD Integration (12 Hours)

Introduction to ECAD and MCAD environments, Schematic capture and component libraries, 3D PCB modeling and enclosure design, Component placement within assemblies, Interference checking and validation, STEP/IDF data exchange, Co-design workflow for embedded systems.

Engineering Applications: Embedded systems design, IoT product development, mechatronic system integration, and electronic enclosure development.

Module 5: PCB Design & Layout (12 Hours)

PCB design fundamentals, Schematic design and circuit connectivity, Component footprint selection, PCB layout: placement, routing, and multilayer PCB basics, Design rules and constraints, Signal integrity fundamentals, Gerber file generation and validation.

Engineering Applications: PCB development for embedded systems, electronic product prototyping, and intelligent electronic hardware design.

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module-1: Use AI-assisted CAD and drafting tools such as Autodesk Fusion 360 and AutoCAD to generate, validate, and optimize engineering drawings, projections, and 2D sketches according to BIS standards.



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Module-2: Apply AI-enabled parametric modelling and assembly tools for intelligent feature creation, geometric optimization, interference analysis, and rapid digital prototyping.

Module-3: Use AI-powered generative design platforms for topology optimization, material selection, load analysis, and development of lightweight optimized engineering structures.

Module-4: Integrate AI-assisted ECAD–MCAD co-design tools for schematic capture, PCB–enclosure integration, component placement optimization, interference checking, and STEP/IDF data exchange validation in embedded and IoT systems.

Module-5: Apply AI-enabled PCB design and electronic automation tools for intelligent component placement, routing optimization, signal integrity analysis, multilayer PCB validation, and Gerber file generation.

Authentic Intelligence Reflection:

Module-1: AI integration improves visualization, drafting accuracy, and technical documentation skills through intelligent CAD-assisted engineering graphics workflows.

Module-2: AI-assisted 3D modelling enhances product development, assembly validation, and digital prototyping capabilities aligned with modern manufacturing practices.

Module-3: AI integration enables computational optimization, sustainable design exploration, and data-driven engineering decision-making for advanced structural design applications.

Module-4: AI-supported ECAD–MCAD integration strengthens interdisciplinary design skills in embedded systems, IoT product development, and intelligent mechatronic system integration.

Module-5: AI-driven PCB design and layout workflows improve electronic product development, circuit optimization, and smart hardware prototyping capabilities for modern electronic systems.

AI-Integrated Lab Exercises (Indicative)

Exercise 1: Engineering Graphics & 2D Sketching

Utilize AI tools for sketch recognition, auto-dimensioning, and automated visualization of orthographic, sectional, and isometric views.

Exercise 2: 3D Modeling

Apply AI-assisted feature recognition for solid modeling (extrude, revolve, loft) and implement



real-time AI feedback for parametric constraints and assemblies.

Exercise 3: Generative Design

Execute AI-driven design exploration and performance prediction based on specific constraints, materials, and structural loads using cloud-based optimization tools.

Exercise 4: ECAD-MCAD Integration

Leverage AI for automated component placement, smart interference detection, and seamless PCB-enclosure integration validation.

Exercise 5: PCB Design & Layout

Implement AI-assisted auto-routing, layout optimization, and AI-based Design Rule Checking (DRC) for signal integrity and layout validation.

Laboratory Component (1 Credit)

The lab emphasizes hands-on CAD modeling, PCB design, and system integration.

Lab Experiments (Indicative):

1. **2D Sketching:** Creating sketches with auto-dimensioning and sketch recognition tools.
2. **3D Modeling:** Developing parametric 3D models using extrude, revolve, and assembly constraints.
3. **Generative Design:** Applying AI-driven exploration based on materials, loads, and constraints.
4. **PCB Schematic Design:** Create logical circuit representations by defining nets and symbols to ensure electrical connectivity and establish the foundation for physical layout.
5. **PCB Layout and Routing:** Convert schematics into physical boards by placing footprints and routing traces while managing layer stacks and ground planes for signal integrity.
6. **Gerber Generation & Validation:** Export industry-standard manufacturing files (RS-274X, Excellon) and perform Design Rule Checks (DRC) to validate alignment and precision before fabrication.



Assessment Scheme

Definitions

1. Internal Assessment (CIA) – 60 Marks

- One **Mid Semester Examination (MSE)** (CAD Tool-based, Laboratory Exercises),
- One **Continuous Assessment Tool (CAT)**: It may include: Surprise test (Computer-based), Quiz, Project based learning, Reflection Note on Guest Lecture, Industrial visit, E-course certification, Building models, Group discussion, Case study, Assignment, Seminar, or Open Book Assignment.
- One **AI-powered Assignment (APA)**: Use of AI tools for problem-solving while simultaneously developing Authentic Intelligence through critical evaluation, ethical reasoning, and responsible decision-making.

OR

- Mini Project**: Lightweight generative structural component design.
2. “**External Assessment**” refers to the Semester End Examinations (SEE). The weightage of this component is **40%**. At the end of semester, the final examination is conducted to evaluate student’s performance.

1. Internal Assessment (CIA) – 60 Marks

Component	Marks	Modules
One MSE	30	Module 1 & 2
CAT - Assignment on Generative Design	10	Module 3
AI-powered Assignment (APA) OR Mini Project	20	All Modules
Total	60	—

2. External Assessment (SEE) – 40 Marks

Component	Marks	Module
Exercise 1 Using CAD Tool	20	Module 4
Exercise 2 Using CAD Tool	20	Module 5
Total	40	

Recommended Tools

- CAD Software: Autodesk Fusion
- Simulation: Generative design tools, CAD simulation modules
- Programming: Python (Fusion API)
- AI Tools: ChatGPT, CAD plugins



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Suggested Reading

Textbooks:

1. K. Venugopal and V. Prabhu Raja, *Engineering Drawing and Graphics*, New Age International Publishers.
2. Sham Tickoo, *Autodesk Fusion 360 for Designers*, CADCIM Technologies.
3. Simon Monk, *Make Your Own PCBs with EAGLE: From Schematic Designs to Finished Boards*, McGraw Hill Education.

Reference Books:

1. Ibrahim Zeid, *CAD/CAM: Theory and Practice*, McGraw Hill.
2. Ian Stroud and Henry Nagy, *3D Modelling and CAD Systems*, Springer.
3. Chris Schroeder, *Printed Circuit Board Design Using Autodesk EAGLE*, Newnes/Elsevier.
4. Kraig Mitzner, *Complete PCB Design Using OrCAD Capture and PCB Editor*, Newnes.
5. Randy H. Shih, *Parametric Modelling with Autodesk Fusion 360*, SDC Publications.

Additional Sources:

- NPTEL – Engineering Graphics & CAD
- Autodesk Learning Resources
- Online MOOC platforms

Mapped NVIDIA DLI Course

Course Title: Fundamentals of Deep Learning OR Computer-Aided Engineering (Simulation & Digital Twins)

Focus: Understanding how neural networks and parallel processing (CUDA) accelerate design validation and generative iterations.

Relevance: This course provides the foundational knowledge needed to understand the "AI Logic" (Input → Weighted Sum → Activation) mentioned in your syllabus's AI Integration section.

DLI Learning Path for Engineering Design

This path is designed to transition students from traditional CAD to AI-accelerated generative workflows.



Strength of CO Mapping to PO/PSOs:

Table: Mapping Levels of Cos to POs / PSOs													
COs	Program Outcomes (POs)											PSOs	
	1	2	3	4	5	6	7	8	9	10	11	1	2
CO1	3	2	1	-	1	-	-	-	-	2	-	2	1
CO2	3	3	2	-	1	-	-	-	-	1	-	2	1
CO3	2	1	2	-	3	-	-	-	1	2	-	3	2
CO4	2	2	3	2	3	1	2	-	1	1	-	3	3
CO5	1	-	1	-	2	-	-	1	2	3	1	1	2
AVG	11/15	8/15	9/15	2/15	10/15	1/15	2/15	1/15	4/15	9/15	1/15	11/15	9/15

3: Substantial (High)

2: Moderate (Medium)

1: Poor (Low)



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakagalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Engineering Graphics & Generative Design For Mechanical Cluster

(As per Choice Based Credit System)

Course Code: 26XXXXX	Semester: I/II
Programs Offered: B.Tech (ME & Aerospace)	Prerequisites: NIL
Course Type: Engineering Science Course (ESC)	Software Platform: Autodesk Fusion 360
L – T – P – J: 2 – 0 – 2 – 0 (2 Credit Theory + 1 Credit Lab)	Programming Language: Python
Contact Hours: 45 Hours (Laboratory)	Credits: 3

Course Vision

To develop strong digital design skills integrating engineering graphics, 3D modeling, generative design, and additive manufacturing for intelligent mechanical and aerospace product development.

Course Objectives

Students will be able to:

- To develop engineering drawing and visualization skills
- To create parametric 2D and 3D CAD models
- To understand product design and optimization concepts
- To develop assembly modeling skills for engineering systems
- To apply additive manufacturing for rapid prototyping

Course Outcomes (COs)

CO No.	Description	Bloom's Taxonomy Level
CO1	Apply engineering graphics principles and interpret engineering drawings using orthographic, sectional, and isometric projection techniques	L3



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

C02	Develop parametric 2D sketches and feature-based 3D CAD models using modern CAD tools	L3
C03	Apply generative design concepts, constraints, and optimization techniques to develop efficient engineering solutions	L3
C04	Analyze and create assemblies of mechanical and aerospace components using mating conditions, constraints, and interference validation techniques	L4
C05	Design and fabricate engineering prototypes using additive manufacturing principles and 3D printing workflows	L3

Broad Course Description

This course provides a strong foundation in engineering graphics, 3D modeling, product assembly, and additive manufacturing for Mechanical and Aerospace Engineering students. It integrates traditional engineering drawing principles with modern parametric CAD tools, generative design techniques, and 3D printing technologies.

Students will learn to visualize engineering components, create parametric 3D models, develop assemblies of mechanical and aerospace systems, and convert digital designs into physical prototypes using additive manufacturing. The course emphasizes computational thinking, design validation, and AI-assisted engineering workflows.

AI tools are integrated throughout the course to assist in sketching, modeling, generative design, and manufacturability analysis, while students critically evaluate outputs using engineering judgment and design principles.

Detailed Course Syllabus

Module 1: Engineering Graphics & 2D Sketching (12 Hours)

Engineering graphics fundamentals, BIS standards, drawing conventions, Orthographic projections, sectional views, isometric views, CAD interface and sketch environment, 2D sketching: tools, constraints, dimensioning

Engineering Applications: Technical drawing and documentation

Module 2: 3D Modeling (12 Hours)

Parametric and feature-based modelling, Solid modeling: extrude, revolve, sweep, loft, Feature modifications and assemblies

Engineering Applications: Product design and digital prototyping



Module 3: Generative Design (12 Hours)

Generative design concepts and workflows, Constraints, materials, loads, Optimization and validation.

Engineering Applications: Lightweight and optimized structures

Unit 4: Assembly of Mechanical & Aerospace Components (12 Hours)

Assembly modeling fundamentals, Mating conditions and mechanical constraints, Hierarchical assembly structure, Interference detection and fit analysis. Subsystem assembly of mechanical and aerospace systems.

Engineering Applications: Mechanical assemblies (gear systems, shafts, and brackets), Aerospace subsystem integration (frames, supports, linkages).

Unit 5: Design for Additive Manufacturing & 3D Printing (12 Hours)

Fundamentals of additive manufacturing, Design for Additive Manufacturing (DFAM principles), STL generation and slicing workflow, 3D printing technologies (FDM, SLA, SLS overview), Post-processing techniques

Engineering Applications: Rapid prototyping of mechanical and aerospace components, Lightweight structural part fabrication.

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module-1: Use AI-assisted CAD and drafting tools such as Autodesk Fusion 360 and AutoCAD to generate, validate, and optimize engineering drawings, orthographic projections, sectional views, and 2D sketches according to BIS standards.

Module-2: Apply AI-enabled parametric modelling and assembly tools for intelligent feature creation, geometric optimization, interference analysis, and rapid digital prototyping of engineering components.

Module-3: Use AI-powered generative design platforms for topology optimization, lightweight structure development, material selection, load analysis, and performance-driven design validation.

Module-4: Integrate AI-assisted assembly modelling tools for automatic mating suggestions, interference detection, fit analysis, and subsystem integration of mechanical and aerospace assemblies.



Module-5: Apply AI-enabled additive manufacturing and DFAM tools for intelligent part optimization, STL validation, slicing parameter prediction, print quality analysis, and lightweight structural fabrication.

Authentic Intelligence Reflection:

Module-1: AI integration improves visualization, drafting accuracy, and technical documentation skills through intelligent CAD-assisted engineering graphics workflows.

Module-2: AI-assisted 3D modelling enhances product development, assembly validation, and digital prototyping capabilities aligned with modern manufacturing and aerospace industries.

Module-3: AI integration enables computational optimization, sustainable design exploration, and data-driven engineering decision-making for advanced structural applications.

Module-4: AI-supported assembly modelling strengthens system integration, mechanical fit analysis, and aerospace subsystem development skills required in complex engineering environments.

Module-5: AI-driven additive manufacturing workflows improve rapid prototyping, lightweight component fabrication, and smart manufacturing capabilities for advanced mechanical and aerospace applications.

AI-Integrated Lab Exercises (Indicative)

1: Engineering Graphics & 2D Sketching

Utilize AI tools for sketch recognition, auto-dimensioning, and automated visualization of orthographic, sectional, and isometric views.

2: 3D Modeling

Apply AI-assisted feature recognition for solid modeling (extrude, revolve, loft) and implement real-time AI feedback for parametric constraints and assemblies.

3: Generative Design

Execute AI-driven design exploration and performance prediction based on specific constraints, materials, and structural loads using cloud-based optimization tools.

4: Assembly of Mechanical & Aerospace Components

Leverage AI for automated part mating, smart interference detection, and AI-driven tolerance analysis within complex hierarchical assembly structures.

5: Design for Additive Manufacturing & 3D Printing

Implement AI-based orientation optimization, automated support structure generation, and AI-driven slicing workflows to enhance printability and minimize material waste.



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakagalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Laboratory Component (1 Credit)

The lab emphasizes hands-on CAD modeling, assembly design, generative design, and additive manufacturing for mechanical and aerospace systems.

Lab Experiments

1. **2D Sketching:** Creating sketches with auto-dimensioning and sketch recognition tools.
2. **3D Modeling:** Developing parametric 3D models using extrude, revolve, and assembly constraints.
3. **Generative Design:** Applying AI-driven exploration based on materials, loads, and constraints.
4. **Assembly & Interference Checking:** Combine parts using mechanical joints and constraints, utilizing interference detection to ensure physical compatibility of subsystems.
5. **Parametric Modeling:** Establish mathematical relationships between dimensions to create "intelligent" models that update all related components automatically when variables change.
6. **Design Validation:** Apply material properties, structural loads, and constraints to simulate and validate design performance before physical prototyping.
7. **STL Generation:** Convert CAD geometry into high-fidelity STL mesh files, optimizing tessellation and orientation for structural integrity in additive manufacturing.
8. **3D Printing Workflow:** Execute the end-to-end process from slicing models into G-code to hardware management and post-processing like support removal.

Assessment Scheme

Definitions

1. Internal Assessment (CIA) – 60 Marks

- a. One **Mid Semester Examination (MSE)** (CAD Tool-based, Laboratory Exercises),
- b. One **Continuous Assessment Tool (CAT):** It may include: Surprise test (Computer-based), Quiz, Project based learning, Reflection Note on Guest Lecture, Industrial visit, E-course certification, Building models, Group discussion, Case study, Assignment, Seminar, or Open Book Assignment.
- c. One **AI-powered Assignment (APA):** Use of AI tools for problem-solving while simultaneously developing Authentic Intelligence through critical evaluation, ethical reasoning, and responsible decision-making.

OR

- d. **Mini Project:** Lightweight generative structural component design.



2. **“External Assessment”** refers to the Semester End Examinations (SEE). The weightage of this component is **40%**. At the end of semester, the final examination is conducted to evaluate student’s performance.

1. Internal Assessment (CIA) – 60 Marks

Component	Marks	Modules
One MSE	30	Module 1 & 2
CAT - Assignment on Generative Design	10	Module 3
AI-powered Assignment (APA) OR Mini Project	20	All Modules
Total	60	—

2. External Assessment (SEE) – 40 Marks

Component	Marks	Module
Exercise 1 Using CAD Tool	20	Module 4
Exercise 2 Using CAD Tool	20	Module 5
Total	40	

Recommended Tools

- CAD Software: Autodesk Fusion
- Simulation: Generative design tools, CAD simulation modules
- Programming: Python (Fusion API)
- AI Tools: ChatGPT, CAD plugins

Suggested Reading

Textbooks

1. K. Venugopal and V. Prabhu Raja, Engineering Drawing and Graphics, New Age International Publishers.
2. Sham Tickoo, Autodesk Fusion 360 for Designers, CADCIM Technologies.
3. P. N. Rao, CAD/CAM: Principles and Applications, McGraw Hill Education.
4. Randy H. Shih, Parametric Modeling with Autodesk Fusion 360, SDC Publications.

Reference Books

5. Ibrahim Zeid, CAD/CAM: Theory and Practice, McGraw Hill.
6. Ian Stroud and Henry Nagy, Solid Modelling and CAD Systems: How to Survive a CAD



System, Springer.

7. Andreas Gebhardt and Jan-Steffen Hötter, Additive Manufacturing: 3D Printing for Prototyping and Manufacturing, Hanser Publishers.
8. John M. Duff, Design for Additive Manufacturing and Assembly, McGraw Hill.
9. David Planchard and Marie P. Planchard, Engineering Design with Fusion 360, SDC Publications.

Additional Sources

- NPTEL – Engineering Graphics, CAD and Additive Manufacturing
- Autodesk Learning Resources (Fusion 360 official tutorials)
- Online MOOC platforms (Coursera, edX, Udemy) for CAD and 3D printing
- IEEE Xplore / ScienceDirect for research papers in design and manufacturing

Mapped NVIDIA DLI Course

Course Title: Fundamentals of Deep Learning OR Computer-Aided Engineering (Simulation & Digital Twins)

Focus: Understanding how neural networks and parallel processing (CUDA) accelerate design validation and generative iterations.

Relevance: This course provides the foundational knowledge needed to understand the "AI Logic" (Input → Weighted Sum → Activation) mentioned in your syllabus's AI Integration section.

DLI Learning Path for Engineering Design

This path is designed to transition students from traditional CAD to AI-accelerated generative workflows.

Strength of CO Mapping to PO/PSOs:

Table: Mapping Levels of Cos to POs / PSOs													
COs	Program Outcomes (POs)											PSOs	
	1	2	3	4	5	6	7	8	9	10	11	1	2
CO1	3	2	1	-	1	-	-	-	-	2	-	2	1
CO2	3	3	2	-	1	-	-	-	-	1	-	2	1
CO3	2	1	2	-	3	-	-	-	1	2	-	3	2



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka - 562 112

C04	2	2	3	2	3	1	2	-	1	1	-	3	3
C05	1	-	1	-	2	-	-	1	2	3	1	1	2
AVG	11/15	8/15	9/15	2/15	10/15	1/15	2/15	1/15	4/15	9/15	1/15	11/15	9/15

3: Substantial (High)

2: Moderate (Medium)

1: Poor (Low)



Human-Centered Design, Ethics & Values

Course Code: 26XXXXX	Semester: I/II
Programs Offered: B.Tech	Prerequisites: NIL
Course Type: Engineering Science Course (ESC)	
L – T – P – J: 1 – 1 – 0 – 0 (2 Credit Theory)	Programming Language: NIL
Contact Hours: 30 Hours	Credits: 2

Course Vision: To enable students to develop empathy, design user-centric solutions, and understand the ethical implications of technology on society.

Course Outcomes

1. Differentiate between user-centric and technology-centric approaches.
2. Ability to conduct research to understand user needs and emotional context.
3. Apply moral reasoning to identify ethical issues in engineering scenarios.
4. Integrate safety and environmental stewardship into design solutions.
5. Evaluate the long-term societal impact of technology and design responsibly.

Course Modules:

Module 1: Foundations of Human-Centered Design (HCD) (6 Hours)

Topics: Introduction to HCD vs. Product-Centered Design; The Design Thinking Process (Empathize, Define, Ideate, Prototype, Test); Understanding user experience (UX) and human limitations (human error as a design failure); Cognitive ergonomics and mental models.

Activity: Identifying a "bad design" in daily life and mapping the user journey.

Module 2: Empathy, Research, and User Understanding (6 Hours)

Topics: Research methods: Observation, User Interviews, Focus Groups; Empathy mapping; Creating User Personas; Contextual inquiry; Understanding diverse societal contexts.

Activity: Conducting a 10-minute user interview and empathy map creation.

Module 3: Ethical Frameworks and Values in Engineering (6 Hours)

Topics: Engineering Ethics: Morals, Values, and Ethics; Moral dilemmas, dilemmas in technological design; Introduction to ethical theories (Utilitarianism, Duty Ethics, Virtue Ethics); Professionalism and codes of ethics (IEEE/ACM).

Activity: Case study discussions on ethical dilemmas (e.g., Tesla autopilot, social media algorithms).

Table: Mapping Levels of COs to POs mapping													
COs	Program Outcomes (POs)											PSOs	
	1	2	3	4	5	6	7	8	9	10	11	1	2
C01	1	2	3	1		2		2	2	2		2	2
C02	1	3	2	3	1	2		1	3	2		2	2
C03		2	1	2		3		3	2	2		1	3
C04	1	2	3	2	1	3	3	3	2	1		2	3
C05	1	2	2	1	2	3	2	3	2	3	1	2	3
AVG	4/15	11/15	11/15	9/15	4/15	13/15	5/15	12/15	11/15	10/15	1/15	9/15	13/15
3: Substantial (High)			2: Moderate (Medium)					1: Poor (Low)					



DESIGN STUDIO-I: ENGINEERING PROBLEM SOLVING

Particulars	Details
Course Title:	Design Studio-I: Engineering Problem Solving
Course Type:	Studio / Experiential Learning
Credits:2	L-T-P-J:0-0-0-4
Semester:	II
Prerequisites:	None
Nature:	Common to all B.Tech Programs

Course Vision

Design Studio–It is a foundational experiential learning course designed to transform first-year engineering students from passive learners into active problem solvers, system thinkers, and prototype builders. The course introduces students to:

- Engineering problem-solving,
- Human-centered design,
- Sustainability-driven thinking,
- AI-assisted exploration,
- Teamwork,
- Prototyping, and
- Authentic Intelligence (AI²).

Students work in interdisciplinary teams to identify, analyze, design, prototype, and present solutions to real-world engineering and societal problems.

Course Objectives

The course enables students to:

1. Understand real-world engineering problems through observation and analysis.
2. Apply structured problem-solving approaches.



3. Develop creativity, teamwork, and systems thinking.
4. Build low-fidelity and functional prototypes.
5. Use AI tools responsibly for ideation and analysis.
6. Understand sustainability, ethics, and societal impact.
7. Communicate engineering ideas effectively.

Course Outcomes (COs)

At the end of the course, students will be able to:

CO No.	Course Outcome
CO1	Identify and frame engineering problems considering users, constraints, and context
CO2	Generate multiple design alternatives using engineering and computational thinking
CO3	Develop basic prototypes/models to demonstrate solution concepts
CO4	Use AI tools for ideation, analysis, and design exploration responsibly
CO5	Analyze sustainability, ethics, and societal impact of engineering solutions
CO6	Work effectively in interdisciplinary teams and communicate solutions professionally

Pedagogical Framework (AI² Pedagogy)

The studio follows: **AI² = Artificial Intelligence + Authentic Intelligence**

Students learn:

- How to use AI
- How to evaluate AI outputs
- How to make responsible engineering decisions

MODULE 1: UNDERSTANDING ENGINEERING PROBLEMS (*Weeks 1-2*)

Topics: What is engineering problem solving?, Problem identification & problem framing, Stakeholder analysis, Observation & empathy mapping, Real-world engineering challenges

Studio Activities

- Campus/community observation
- Problem identification workshop



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakagalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

- Team formation

AI Integration: AI-assisted idea exploration and AI-based brainstorming tools

Authentic Intelligence Focus: Difference between solving symptoms vs root causes and Human-centered thinking.

Deliverable: Problem Definition Report

MODULE 2: DESIGN THINKING & IDEA GENERATION (*Weeks 3–4*)

Topics: Design thinking fundamentals, Ideation techniques, Functional decomposition, Systems thinking basics, Engineering constraints.

Studio Activities

- Brainstorming sessions
- Concept sketching
- Decision matrix

AI Integration: AI-generated alternatives and Comparing AI vs human ideas.

Authentic Intelligence Focus: Evaluating practicality & ethics and Trade-off analysis.

Deliverable: Concept Design Portfolio

MODULE 3: MODELING, SIMULATION & PROTOTYPING (*Weeks 5–7*)

Topics: Rapid prototyping, Basic modeling tools, Sensors & simple automation, Introduction to simulations, Low-fidelity & functional prototypes.

Studio Activities

- Prototype development
- Makerspace activities
- Basic simulations / Mockups

AI Integration: AI-assisted coding / design suggestions and AI for visualization.

Authentic Intelligence Focus: Reliability & usability and Failure analysis.

Deliverable: Prototype Version–1

MODULE 4: TESTING, ITERATION & VALIDATION (*Weeks 8–10*)

Topics: Testing methodologies, User feedback, Performance metrics, Iterative improvement, Engineering optimization basics.

Studio Activities



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

- Prototype testing
- Peer reviews
- Design iteration

AI Integration: AI-based analysis & visualization and AI-assisted debugging.

Authentic Intelligence Focus: Biases in AI suggestions and Responsible evaluation.

Deliverable: Prototype Version-2 and Testing Report

MODULE 5: SUSTAINABILITY, ETHICS & IMPACT (*Weeks 11-12*)

Topics: Sustainability in engineering, Lifecycle thinking, Ethical engineering practices, Inclusive technologies, Social impact assessment

Studio Activities

- Sustainability analysis
- Ethical dilemma discussions

AI Integration: AI and sustainability and Environmental impact estimation

Authentic Intelligence Focus: “Can this be built?” vs “Should this be built?”

Deliverable: AI² Reflection & Sustainability Report

MODULE 6: FINAL INTEGRATION & DEMONSTRATION (*Weeks 13-14*)

Topics: Technical storytelling, Engineering presentations, Demo preparation, Innovation pitching.

Studio Activities

- Final prototype demonstration
- Poster presentation
- Jury evaluation

Deliverable: Final Prototype or Demo Video, Presentation and Technical Report.

SAMPLE PROBLEM STATEMENTS

Students may work on:

- Reducing water and energy wastage in hostels
- Smart waste segregation



- Air quality monitoring
- Smart parking system
- Campus accessibility solutions
- Automated irrigation systems
- Pedestrian safety systems
- Smart classroom utilization

CoE & Infrastructure Utilization

Students may utilize:

- AI Makerspace,
- General Makerspace,
- AI & Robotics CoE,
- AR/VR/XR Lab,
- AI Factory (introductory exposure), or
- Any suitable lab.

ASSESSMENT SCHEME

Component	Weightage
Continuous Studio Evaluation	30%
Design Reviews	15%
Prototype Development	20%
AI ² Reflection	10%
Final Demonstration & Viva	15%
Documentation & Presentation	10%

RECOMMENDED TOOLS

- ChatGPT / AI assistants
- Canva / Figma
- Arduino / Raspberry Pi
- Python basics
- Tinkercad
- Simulation tools

Note: “Design Studio–I ensure that every first-year engineering student at SoE-DSU learns engineering not by memorizing concepts, but by solving real problems, building prototypes,



using AI responsibly, and understanding the societal impact of technology.”

Laboratory Component

1. Identify a **real-world problem** through field observation.
2. Conduct **user interviews and create personas**.
3. Develop **problem statements and design challenges**.
4. Perform **ideation sessions (brainstorming, SCAMPER)**.
5. Create **low-fidelity prototypes (paper/wireframes)**.
6. Develop **digital prototypes using design tools**.
7. Conduct **user testing and collect feedback**.
8. Perform **basic patent search for the idea**.
9. Prepare **IP strategy for the solution**.
10. Final **design showcase + innovation pitch**.

Assessment Scheme

Internal (100 Marks – No End Semester Exam Recommended)

Component	Marks
Continuous Assessment (Activities)	30
Design Challenge / Assignments	20
Prototype Development	20
IP Report / Patent Search	10
Final Project + Presentation	20
Total	100

Recommended Tools

- **Design Tools:** Figma, Canva, Adobe XD
- **Prototyping:** Balsamiq, Tinkercad
- **AI Tools:** ChatGPT, Ideation tools, UI generators
- **IP Tools:** Google Patents, WIPO databases

Integrated Tools for Studio

- **Software:** Figma, Canva, Tinkercad



- **Platforms:** Google Forms (user feedback), Miro (brainstorming)
- **AI Tools:** Generative AI tools for ideation and prototyping
- **IP Platforms:** Google Patents, WIPO

Suggested Reading

Primary Textbooks

1. Change by Design – Tim Brown
Covers fundamentals of design thinking, innovation strategies, and human-centered design (CO1, CO2, CO3).
2. Sprint: How to Solve Big Problems and Test New Ideas in Just Five Days – Jake Knapp
Focuses on rapid prototyping, ideation, and solution validation
3. WIPO Intellectual Property Handbook – World Intellectual Property Organization
Provides comprehensive understanding of IPR concepts, laws, and protection mechanisms (CO1, CO5).

REFERENCE BOOKS

1. IDEO Design Thinking Resources
Practical toolkits, case studies, and real-world applications of design thinking (CO2, CO3, CO4).
2. Stanford school Materials
Structured methodologies, design frameworks, and collaborative innovation practices
3. The Design of Everyday Things – Don Norman
Enhances understanding of user experience and usability principles (CO2, CO3).
4. Creative Confidence – Tom Kelley, David Kelley

E-RESOURCES

1. World Intellectual Property Organization – Official Learning Resources
<https://www.wipo.int>
(IPR tutorials, case studies, global standards)
2. IDEO U – Design Thinking Courses
<https://www.ideo.com>
(Hands-on innovation and design thinking modules)
3. Stanford University – d.school Open Resources
<https://dschool.stanford.edu/resources>
(Design thinking toolkits and guides)
4. Coursera / edX



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka - 562 112

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)											PSOs	
	1	2	3	4	5	6	7	8	9	10	11	1	2
CO1	2	2	3	3	1	2	1	1	2	2	1	2	1
CO2	2	3	3	2	2	1	1	1	2	2	1	2	1
CO3	2	2	3	1	3	1	1	1	3	2	2	2	2
CO4	1	2	2	2	3	1	1	2	2	2	1	2	2
CO5	1	2	2	2	1	3	3	3	2	3	1	2	3
CO6	1	1	2	1	2	2	1	2	3	3	3	2	2
AV G	9/1 8	12/1 8	15/1 8	11/1 8	12/1 8	10/1 8	8/1 8	10/1 8	14/1 8	14/1 8	9/1 8	12/1 8	11/1 8
3: Substantial (High)			2: Moderate (Medium)					1: Poor (Low)					



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Introduction to C Programming

(As per Choice Based Credit System)

Course Code: 26XXXXXX	Semester: I/II
Programs Offered:	All programs of B.Tech.
Prerequisites: Fundamentals of computers	
Course Type: Core	Hours/Week: 04 Hrs (02 + 02)
Total Hours: 30 Hrs (L) + 30 Hrs (P)	Credits: 03
L – T – P – J: 2 – 0 – 2 – 0 (2 Credits Theory + 1 Credit Lab)	

Course Vision

This course develops the analytical skills, algorithmic thinking, learn C programming fundamentals to write clean, modular and optimized code. The learners can gain confidence in identifying errors and validating solutions for practical problems in mathematics, science and everyday scenarios using C. The course is extended to implement CUDA programming techniques to support parallel execution for massive datasets.

Course Objectives

Students will be able to:

1. Understand programming constructs of C language to solve the real problems
2. Explore simple data structures like arrays and strings to solve real world problems
3. Demonstrate modular codes using pointers and functional programming concepts
4. Demonstrate the concept that binds heterogeneous data using structures and unions constructs
5. Introduce Nvidia CUDA parallel code execution and device memory management methods



Course Outcomes (CO)

CO	Outcome	BTL
CO1	Apply fundamental programming constructs such as data types, operators, expressions, control flow, and loops to write basic C programs.	L3
CO2	Analyze the ability of one-dimensional and two-dimensional arrays to perform search, sort, matrix operations and manipulate strings using standard C functions.	L3
CO3	Develop functional codes with pointers including recursion, call-by-value/address, and arrays as arguments to build modular C programs.	L3
CO4	Develop programs using structures, unions and files to represent and manage compound data types for real-world applications like student or employee databases.	L3
CO5	Implement Nvidia CUDA dynamic memory allocation techniques for device memory management and parallel execution for massive datasets.	L3

Broad Course Description

This course introduces students to the fundamentals of programming and computational problem-solving using the C language. It focuses on building strong skills in logical thinking, algorithmic design, and structured programming techniques to address real-world challenges.

Learners will gain hands-on experience in writing, compiling, debugging, and executing C programs while exploring essential concepts such as Data types and control structures, Arrays and pointers, Functions, structures, and unions, File operations and dynamic memory allocation techniques.

Beyond C programming, the course provides a foundation in CUDA programming, a powerful framework for harnessing the computational capabilities of NVIDIA GPUs. It includes the topics namely, Installation and setup of the NVIDIA CUDA Toolkit, Parallel execution of large-scale data operations, CUDA compiler usage, Memory management across CPU and GPU, Profiling CUDA programs to observe performance gains.



Through a blend of theory and practical exercises, the course nurtures the ability to translate everyday problems into computational models and implement efficient solutions.

Detailed Course Syllabus

The course consists of five modules. Each module integrates:

- Core topics
- Engineering applications
- Programiz Pro C Code Visualizer too

Course Modules

Module 1: Basics and overview of C

(6 Hours)

Introduction to Problem Solving using Algorithms and Flowchart: Key features of Algorithms: Sequence, Decision, Repetition with examples. Background, structure of C program, keywords, Identifiers, Data Types, Variables, Constants, Input / Output statements, Operators (Arithmetic, relational, logical, bitwise etc.), Expressions, Precedence and Associativity, Expression Evaluation, Type conversions. Conditional Branching Statements-if and switch statements, iterative statements (loops)-while, for, do-while statements, Loop examples, Nested loops, break, continue, go to statement.

Engineering Applications: Industrial cooling systems, Automotive Engine Control Units

Module 2: Arrays and Strings

(6 Hours)

1D-Array: Definition, declaration and initialization of one-dimensional array, reading and writing array elements, Operations on 1D-Array: traversal, searching (Linear and Binary search), sorting (Bubble sort and Selection Sort). 2D-Array: Declaration and Initialization of two-dimensional arrays. Operations on 2D-Array: Matrix Operations - addition, subtraction, multiplication and transpose operations. Strings/Character Arrays: Definition, declaration, initialization, and representation of strings, Input/Output operations on strings, String handling functions and character handling functions.

Engineering Applications: Multi-Sensor Monitoring Dashboard, True Caller Simulation.

Module 3: Pointers and User defined functions

(6 Hours)

Pointers: Definition, declaration and initialization of pointers. Accessing values using pointers. Accessing array elements using pointers, Double pointers. User defined functions: Built-in and user defined functions, Definition, declaration and call of user-defined functions. Categories of functions with example. Pointers as function arguments, Array as function argument and return value, Pass-by-value and Pass-by-address, Recursion.

Engineering Applications: Banking transaction systems, Sensor data Analysis.



Module 4: Dynamic Memory allocation & Introduction to CUDA (6 Hours)

Dynamic Memory allocation in C programs: Dynamic memory allocation, Host memory allocation process, allocating a block of memory, releasing the used space, altering the size of allocated memory. Introduction to CUDA: Heterogeneous Computing, Simple processing flow, CUDA toolkit installation, NVCC, Device memory management, Parallel program execution, Blocks, Threads and Indexing concepts.

Engineering Applications: Operating system workflow, Real-time environmental monitoring systems.

Module 5: Structures, Unions and Files (6 Hours)

Structures: Purpose and usage of structures. Declaration of structures. Assignment with structures. Structure variables and arrays. Nested structures. Student and employee database implementation using structures. Unions: Declaration and initialization of a union. Difference between structures and unions. Example programs. Files: Defining, open, read, write, seek and closing of both textual and random files.

Engineering Applications: Student/Employee records, Memory optimization in embedded systems, Configuration management.

AI Integration (AI Pedagogy) AI

Integration:

AIM: Understand how foundational Neural networks makes decisions using **input** → **weighted sum** → **activation** → **output** mirrors AI logic.

Example: Implement the foundational perceptron learning algorithm to represent the AND Logic Gate using Arrays, Loops and Functions:

- Develop a Weighted sum function that multiplies inputs by weights and sums them.
- Test with sample inputs and weights.
- Develop a threshold activation and connect it to the weighted sum to Produce binary outputs.
- Write a loop that compares perceptron output to expected output. Adjust weights slightly if the output is wrong.
- Run multiple iterations (epochs).
- Laboratory Component (1 Credit)

The following lab experiments can be carried out using **GCC, Nvidia CUDA Toolkit and Nvidia Nsight Visualization tool:**



Lab Experiments

1. Develop a program for an industrial cooling system that turns on a fan when temperature exceeds 70°C.
2. Develop a program for an automotive ECU that adjusts fuel injection based on RPM.
3. Create a program that simulates a True Caller application by matching input phone numbers with stored names.
4. Build a multi-sensor monitoring dashboard where commands are given as strings (SHOW, ALERT, RESET).
5. Write a program to access sensor data using pointers.
6. Implement double pointers to manage dynamic sensor arrays.
7. Create a function to calculate banking transaction balances.
8. Implement recursion to analyze sensor data trends.
9. Develop a code that allocates memory dynamically for storing environmental sensor data.
10. Develop a program to resize memory blocks as new data arrives.
11. Develop a simple Nvidia CUDA program to process large-scale environmental monitoring data.
12. Implement Nvidia CUDA parallel matrix multiplication to simulate operating system workflow.
13. Implement a student database using structures.
14. Create an employee record management system.
15. Implement reading and writing of sensor logs for configuration management.

AI-Integrated Lab Exercises (Indicative)

- **Exercise 1:** Make use of Nvidia CUDA Toolkit and Nsight systems to visualize the parallel execution cycle that adds 1 million elements.
- **Exercise 2:** Develop a CUDA program that performs element-wise addition of two arrays. The program should allocate memory on the GPU, transfer the input arrays from the host to the device, perform the addition in parallel using a CUDA kernel, and then transfer the result back to the host. Make sure to handle memory allocation and deallocation properly. Also visualize the same with Nvidia Nsight systems.
- Assessment Scheme



Definitions

1. **“Internal Assessment”** refers to the Continuous Internal Assessment (CIA). The weightage of this component is **60%**. It includes:
 - a. One **Mid Semester Examination (MSE)** (Descriptive-based),
 - b. Six **Continuous Assessment Tool (CAT)**: It may include: Surprise test (Computer-based), Quiz, Project based learning, Reflection Note on Guest Lecture, Industrial visit, E-course certification, Building models, Group discussion, Case study, Seminar, or Open Book Assignment
 - c. One **AI-powered Assignment (APA)**: Use of AI tools for problem-solving while simultaneously developing Authentic Intelligence through critical evaluation, ethical reasoning, and responsible decision-making.
2. **“External Assessment”** refers to the Semester End Examinations (SEE). The weightage of this component is **40%**. At the end of semester, the final examination is conducted to evaluate student’s performance.

Integrated Professional Core Courses (Theory + Practical)

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT) every 2 weeks (Minimum 05 assessments)	10
Mid Semester Examination (MSE)	20
Assignment on CUDA Programming	10
Lab Record	10
Lab Examination	10
Semester End Exam	40
Total	100 marks

Recommended Tools

- Programming: GCC, NVCC
- Visualization: Programiz Pro C Code Visualizer tool
- AI Tools: Nvidia CUDA Toolkit, Nvidia Nvprof, Nsight systems



Suggested Reading

Primary Textbooks

1. Behrouz A. Forouzan, Richard F. Gilberg, "Computer Science - A Structured Approach Using C", Cengage Learning, 2007.

Reference Textbooks

1. Reema Thareja, "Programming in C". Oxford University Press, Second Edition, 2016.
2. Brian W. Kernighan and Dennis M. Ritchie, (2012) "The C Programming Language", 2nd Edition, PHI.
3. Vikas Gupta, "Computer Concepts and C Programming", Dream tech Press 2013.

Additional Resources

1. [NOC: Problem Solving Through Programming In C, IIT Kharagpur - NPTEL](#)
2. [C Tutorial](#)
3. [CUDA C/C++ BASICS - IIT Kanpur](#)

Strength of CO Mapping to PO/PSOs:

Table: Mapping Levels of Cos to POs / PSOs													
COs	Program Outcomes (POs)											PSOs	
	1	2	3	4	5	6	7	8	9	10	11	1	2
CO1	3	2	2		2								2
CO2	3	2	3		2								2
CO3	2	3	3		2								2
CO4	2	3	3		2								2
CO5	2	3	3		2								2
AVG	12/15	13/15	14/15		10/15								10/15

3: Substantial (High)

2: Moderate (Medium)

1: Poor (Low)



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakagalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

INTRODUCTION TO ELECTRONICS ENGINEERING

Course Code: 26XXXXX	Semester: 1/2
Programs Offered: B Tech -Electronics and Communication Engineering	Prerequisites: Basic Mathematics, Physics
Course Type: Foundation Engineering for first year students	L – T – P – J: 3 – 0 – 0 – 0 (<i>3 Credits Theory</i>)

Course Objectives

This course enables students to

1. Understand the physics of semiconductor devices including diodes, transistors, and operational amplifiers.
2. Analyse and design analog electronic circuits from first principles.
3. Explore digital logic and sequential circuits as the foundation of digital systems.
4. Use LTspice to simulate, validate, and troubleshoot electronic circuits.
5. Evaluate AI-assisted simulation outputs and develop authentic engineering judgment.

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
CO1	Explain semiconductor device physics, diode construction, and V-I characteristics	L2
CO2	Analyze transistor operation, biasing circuits, and amplifier configurations	L2
CO3	Design and analyze operational amplifier circuits for signal processing applications	L2
CO4	Apply digital logic and Boolean algebra to design combinational and sequential circuits	L3
CO5	Use LTspice to simulate, verify, and critically evaluate electronic circuits	L3



Course Modules

MODULE 1: Semiconductor Diodes and Applications

Topics: Semiconductor types: intrinsic and extrinsic (n-type and p-type); diode construction; diode under no-bias, forward bias and reverse bias; V-I characteristics of diode; simplified equivalent circuit of practical diode and ideal diode; diode specifications: peak inverse voltage (PIV), reverse leakage current and maximum forward current; Zener diode; Series diode configuration with DC input; Rectifier circuits: Half-wave and Full-wave – Bridge Network and Centre-tapped; Reservoir and smoothing circuits;

Engineering Applications: Applications: DC power supply design

MODULE 2: Transistors and Biasing

Topics: Construction of NPN and PNP BJT transistors; transistor operation; input and output characteristics of CB and CE configurations; significance of different regions of operation: active, cut-off and saturation (transistor as a switch); BJT amplification; transistor amplifying action; numerical on current relations and amplification; need for biasing: Q-point; types of biasing: fixed and voltage divider; Introduction to Field Effect Transistor; Construction and characteristics of JFET; MOSFET (Depletion and Enhancement) types; Transfer characteristic.

Engineering Applications: Switching circuits in logic gates, audio pre-amplification, signal conditioning, motor driver circuits, and analog control systems.

MODULE 3: Operational Amplifiers

Topics: Op-amp symbols, terminals and operation: single mode, differential mode and common mode; basic properties of ideal and practical Op-amp: input offset voltage, input resistance, output resistance, gain, bandwidth, CMRR, slew rate; basic Op-amp applications: inverting amplifier, non-inverting amplifier, summing amplifier, differential amplifier, differentiator and integrator; Op-amp comparator; feedback: positive and negative feedback; criteria for stability and oscillations (Barkhausen criterion); RC phase shift oscillators.

Engineering Applications: Signal amplification, active filters, waveform generation, instrumentation amplifiers, and control system feedback circuits.

MODULE 4: Digital Electronics

Topics: Binary number system: conversion and representation; logic levels: high and low; Boolean algebra: operators and DeMorgan's law; Boolean expressions; logic gates with truth-table and representation: AND, OR, NOT, XOR, NAND, NOR; combination of gates and associated numerical problems; sequential logic circuits: SR latch using NAND/NOR gate, SR Flip-Flop, D Flip-Flop, J-K Flip-Flop.

Engineering Applications: Digital logic design, memory elements in registers and counters, control systems, digital signal processing, and embedded systems.



MODULE 5: Microprocessor Systems

Topics: Overview of microprocessor architecture: ALU, registers, data/address/control buses, fetch-decode-execute cycle; overview of microcontroller architecture: comparison with microprocessor, on-chip peripherals; interface between digital electronics and microprocessor systems.

Engineering Applications: Embedded systems design, IoT devices, real-time control systems, and interfacing digital circuits with processor-based platforms.

NVIDIA DLI Courses (Relevant to this Module): Getting Started With AI on NVIDIA Jetson Nano.

https://learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+S-RX-02+V2

AI Integration:

1. LTspice simulation of diode V-I curves; AI-assisted output verification for rectifier and clipper circuits.
2. LTspice simulation of BJT characteristic curves and biasing Q-points; AI comparison of fixed vs. voltage divider bias stability.
3. LTspice AC and transient analysis of Op-amp circuits; AI-assisted gain and frequency response verification.
4. LTspice digital simulation of combinational gates and flip-flop timing diagrams; AI truth table verification.
5. AI-generated microprocessor architecture diagrams and conceptual summaries with student critique.

Authentic Intelligence:

1. Students evaluate whether LTspice-simulated V-I characteristics match theoretical predictions and identify sources of discrepancy.
2. Students critique AI-generated bias circuit designs and assess thermal stability and load sensitivity.
3. Students assess the reliability of AI-generated Op-amp design recommendations against ideal vs. practical Op-amp models.
4. Students evaluate AI-simplified Boolean expressions, verify minimization, and discuss implications for circuit complexity.
5. Students analyse the trade-offs between microprocessors and microcontrollers for a given embedded application.



Suggested Reading

TEXT BOOKS:

1. R. L. Boylestad and L. Nashelsky, “Electronic Devices and Circuit Theory”, 11th Edition, Pearson Education, 2013.
2. M. Moris. Mano and Michael D. Ciletti, “Digital Electronics”, 4th Edition, Pearson Education, 2006.
3. Michael Tooley, “Electronic Circuits: Fundamentals and Applications”, 5th Edition, Routledge, 2006.

REFERENCE BOOKS:

1. David A Bell, “Electronic Devices and Circuits”, 5th Edition, Oxford university press, 2008.
2. Millman & Halkias, “Electronics Devices and Circuits”, 2nd Edition, McGraw Hill, 2010.

E-Resources:

1. NPTEL- <https://nptel.ac.in/courses/122/106/122106025>
2. Virtual Labs- <http://vlabs.iitkgp.ac.in/be/>
3. <https://nptel.ac.in/courses/117104072>

Table: Mapping Levels of COs to POs / PSOs

COs	PO											PSO	
	1	2	3	4	5	6	7	8	9	10	11	1	2
CO1	3	2	1	-	-	-	-	-	-	-	2	3	-
CO2	3	3	2	1	-	-	-	-	-	-	2	3	-
CO3	3	3	2	2	-	-	-	-	-	-	2	3	2
CO4	3	3	2	2	-	-	-	-	-	-	2	3	2
CO5	3	3	3	1	-	-	-	-	-	-	3	3	3

3: Substantial (High)

2: Moderate (Medium)

1: Poor (Low)



Environmental Science

Course Code: 26XXXX	Semester: I/II
Programs Offered: All Engineering Programs	Prerequisites: Life Science
Course Type: Multidisciplinary- Mandatory Course (MC)	L – T – P – J: 2 – 0 – 0 – 0 (2 Credits Theory)
Credits: 02	

Course Vision

To equip engineering students with environmental science knowledge and AI-driven analytical skills to develop innovative, data-driven solutions for sustainable and resilient ecosystems.

Course Objectives

Students will:

1. Understand the evolution and diversity in nature.
2. Understand the concepts of environment, pollution, energy resources.
3. Learn water as a resource, rain water harvesting as a method of conservation of water
4. Understand solid waste and its management
5. Learn environmental Protection Act laws, environmental Impact Analysis and air Monitoring

Course Outcomes (CO)

CO	Outcome	Bloom's Taxonomy Level
C01	Critically elucidate the basic concepts that govern environmental quality, ambient air quality standards	L4
C02	Compare different Energy resource and their environmental implications.	L4
C03	Identify different types of pollution, waste streams.	L1
C04	Identify different natural and manmade disasters and prevention.	L2
C05	Apply the process of environmental impact assessment and implications of Indian Environment Laws	L3



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakagalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Broad Course Description

Environmental Science for Engineers is a multidisciplinary course that focuses on ecosystems, pollution, energy resources, disaster management, and environmental regulations along with their engineering applications. The course emphasizes environmental sustainability, resource management, and pollution control through practical and real-world approaches. By the end of the course, students will be able to understand environmental challenges and develop sustainable engineering solutions for society and industry.

Detailed Course Syllabus

The course consists of five modules. Each module integrates:

- Core topics
- Engineering applications
- AI integration
- Authentic Intelligence reflection

Course Modules

Module 1: ECOSYSTEMS and BIODIVERSITY (06 Hours)

Topics: Ecosystem – structure and function – Producers, consumers, decomposers – Energy flow in ecosystem – Food chains, food webs, ecological pyramids – Environment: definition, scope and importance – Pollutants and xenobiotics (natural & anthropogenic) – Eco-kinetic and bio-kinetic properties – Dose-response relationships – 3 T's (Time, Type, Threshold) – Chronic and acute health effects – Sustainable Development Goals (SDGs).

Engineering Applications:

- Monitoring and managing ecosystems
- Biodiversity conservation and restoration
- Assessing pollutant impacts on humans and wildlife
- Environmental planning and sustainability studies
- Addressing SDG 15 – Life on Land

Module 2: POLLUTION AND MANAGEMENT (06 Hours)

Topics: Air pollution and criteria pollutants – Ozone (O₃), Particulate Matter (PM), Carbon Monoxide (CO), Nitrogen Oxides (NO_x), Sulphur Dioxide (SO₂), Lead (Pb) – Acid rain cycle – Greenhouse effect and global warming – Water resources: lentic and lotic systems – Rainwater harvesting – Water pollution and management strategies – Noise pollution: sources and effects – Municipal solid waste, hazardous waste, electronic waste, biomedical waste – Solid waste management techniques: landfills and composting.

Engineering Applications:

- Monitoring air, water, and soil quality
- Planning sustainable water and waste management systems



- Reducing noise and environmental hazards in urban and industrial areas
- Ensuring safe disposal of hazardous and biomedical waste
- Addressing SDG 14 – Life Below Water, SDG 6 – Clean Water and Sanitation, SDG 11 – Sustainable Cities and Communities and SDG 13 – Climate Action

Module 3: ENERGY (05 Hours)

Topics: Types of energy – Conventional energy sources: fossil fuels, coal, solar, wind – non-conventional energy sources: biofuels, biomass, biogas – Energy production, conversion and utilization – Renewable and sustainable energy sources and their role in environmental sustainability.

Engineering Applications:

- Renewable energy systems (solar panels, wind turbines, bioenergy plants)
- Energy efficiency and sustainability planning
- Develop and optimize systems for biofuel production (fermentation, transesterification, anaerobic digestion)
- Integrate biofuels into hybrid energy systems alongside solar, wind, and smart grids.
- Addressing SDG 7 – Affordable and Clean Energy and SDG 12 – Responsible Consumption and Production

Module 4: DISASTERS AND MANAGEMENT (05 Hours)

Topics: Environmental and societal impacts of energy production – Ethical decision-making in energy planning and consumption – Balancing sustainability with technological and economic considerations.

Engineering Applications:

- Understanding environmental and societal impacts of energy production
- Ethical decision-making in energy planning and consumption
- Balancing sustainability with technological and economic considerations

Module 5: ENVIRONMENTAL IMPACT ASSESSMENT (04 Hours)

Topics: Environmental Impact Assessment (EIA) – purpose, process and importance – Air pollution monitoring and Ambient Air Quality Standards (AAQS) – Environmental legislation: Environment Protection Act, 1986; Water (Prevention and Control of Pollution) Act; Wildlife Protection Act; Forest Conservation Act – Issues in enforcement of environmental laws – Public awareness and participation in environmental protection.

Engineering Applications:

- Planning and regulating developmental project
- Environmental monitoring and compliance
- Policy-making for sustainable development
- Educating the public on environmental responsibility



AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

- **Module-1:** This module integrates AI-based environmental monitoring systems to support analysis of ecosystem patterns and biodiversity datasets. It enables predictive modelling of environmental changes and trends, helping in data-driven interpretation of ecological systems.
- **Module-2:** This module integrates AI-powered sensors for real-time monitoring of air, water, and soil quality, enabling continuous environmental data collection. It also uses remote sensing technologies for tracking large-scale environmental parameters and predictive models to analyse pollution trends and assess environmental risks. Additionally, AI-based decision support systems assist in efficient waste and resource management.
- **Module-3:** This module integrates AI technologies in modern electricity systems for smart grid management, enabling predictive energy distribution based on demand patterns. It supports AI-assisted optimization of renewable energy output from sources such as solar and wind, and uses predictive maintenance techniques to monitor and improve the performance of power plants and energy infrastructure.
- **Module-4:** This module integrates AI techniques for analysing satellite and climate data to support disaster prediction and environmental forecasting. It enables prediction of solar radiation, wind speed, and extreme weather events using data-driven models. It also incorporates AI-based early warning systems and predictive risk assessment tools for scenario simulation and hazard preparedness.
- **Module-5:** This module integrates AI-based modelling techniques for predictive environmental impact assessment of development projects. It supports real-time monitoring of air and water quality using intelligent systems and enables data-driven decision support for regulatory compliance. It also includes simulation tools to evaluate possible environmental outcomes of engineering and industrial activities.

Authentic Intelligence Reflection:

- **Module-1:** This module reflects authentic intelligence by enabling students to interpret ecosystem and pollution data in a responsible manner, ensuring accurate understanding of environmental conditions. It develops awareness of human impacts on biodiversity and ecological balance. It also promotes ethical and informed decision-making for environmental protection, encouraging sustainable practices and responsible engineering actions.
- **Module-2:** This module reflects authentic intelligence by enabling students to interpret



environmental data ethically and accurately for meaningful analysis. It develops understanding of the long-term impacts of pollutants on ecosystems and human health, supporting deeper awareness of environmental consequences. It also promotes informed decision-making in sustainable urban and industrial planning, encouraging responsible and future-oriented engineering solutions.

- **Module-3:** This module reflects authentic intelligence by enabling students to understand the environmental and societal impacts of energy production systems. It develops ethical decision-making skills in energy planning and consumption, ensuring responsible use of resources. It also encourages balancing sustainability with technological advancement and economic feasibility, promoting holistic and informed engineering solutions for energy management.
- **Module-4:** This module reflects authentic intelligence by enabling students to make ethical and responsible decisions in disaster management scenarios. It develops an understanding of the human and environmental consequences of natural and man-made disasters. It also encourages prioritizing safety, preparedness, and sustainability while designing effective response and mitigation strategies for disaster risk reduction.
- **Module-5:** This module reflects authentic intelligence by enabling students to make ethical and informed decisions in environmental management practices. It develops the ability to balance developmental needs with ecological conservation, ensuring sustainable use of natural resources. It also promotes public awareness and responsibility toward environmental protection, encouraging active participation in building a sustainable and eco-friendly society.

Research Integration Teaching

Students review at least one topic:

- AI in climate change prediction
- Smart cities and sustainable development
- AI in biodiversity conservation
- AI-based pollution monitoring
- Renewable energy optimization using AI

Students present a **short research review**.

Assessment Scheme

Definitions

2. **“Continuous Internal Assessment (CIA)”** with a weightage of **60%**. It includes:
 - a. **Mid Semester Examination (MSE):** One descriptive examination to evaluate conceptual understanding of AI fundamentals and problem-solving approaches.
 - b. **Continuous Assessment Tools (CAT – Six Components):** Includes activities such as quizzes, surprise tests, case studies, group discussions, project-based learning, seminars, industrial visits, reflection notes,



model building, e-course certifications, or open book assignments.

C. Professional Core Courses (Theory)

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT) – Quiz, Assignments, Write ups	60
Mid Semester Examination (MSE)	40
Total	100 marks

Recommended Tools

- Atmo→ High-accuracy weather and climate prediction for disaster management
- Envirosuite→ Real-time air, water, and noise pollution monitoring
- Google Earth Engine→ Satellite data analysis for deforestation, land use, climate studies
- iNaturalist→ AI-based species identification
- eBird→ Tracks bird populations and migration
- MegaDetector→ Detects animals using camera trap images
- Climate FieldView→ AI-based crop monitoring and yield prediction
- IBM Watson→ Used for environmental risk prediction and sustainability analytics
- AMP Robotics→ Automated waste sorting and recycling
- Kayros Methane Watch→ Tracks methane emissions globally

Suggested Reading

1. Benny Joseph (2005). "Environmental Studies", Tata McGraw – Hill Publishing Company Limited, New Delhi.
2. R. J. Ranjit Daniels and Jagadish Krishnaswamy (2014). "Environmental Studies" (2014), Wiley India Pvt Limited, New Delhi.
3. P. Aarne Vesilind, Susan M. Morgan, Thomson (2008). "Introduction to Environmental Engineering" (2008), Thomson learning, Second Edition, Boston.
4. R. Rajagopalan (2005). "Environmental Studies – From Crisis to Cure" Oxford University Press, New Delhi.
5. Zhou, K., Fu, C., & Yang, S. (2016). Big data driven smart energy management. Renewable and Sustainable Energy Reviews.



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Table: Mapping Levels of COs to POs / PSOs														
COs	Program Outcomes (POs)											PSOs		
	1	2	3	4	5	6	7	8	9	10	11	1	2	3
C01	2	0	0	0	0	0	2	0	0	0	0	3	0	2
C02	2	0	0	0	0	0	2	0	0	0	0	3	0	2
C03	2	0	0	0	0	0	2	0	0	0	0	3	0	2
C04	2	0	0	0	0	0	2	0	0	0	0	3	0	2
C05	2	0	0	0	0	0	2	0	0	0	0	3	0	2



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

INDIAN CONSTITUTION AND LAW

Course Code: xxxx	Semester: I/II
Programs Offered: B. Tech Program of Humanities	Course Type: Ability Enhancement Course (AEC) Humanities and Social Science
L – T – P – J: 1– 0 – 0– 0 (<i>1 Credits Theory</i>)	

Course Vision

The study of the Constitution of India helps engineering students understand the basic principles governing society, governance, and individual rights. It also nurtures ethically responsible engineers by creating legal awareness and encouraging professional integrity, social responsibility, and respect for constitutional values.

Course Objectives

Students will:

1. Understand the basic principles of the Constitution of India.
2. Explain fundamental rights, duties, and directive principles.
3. Apply legal concepts to real-world situations.
4. Analyse the role of law in governance and society.
5. Develop awareness of ethical and legal responsibilities.

Course Outcomes (CO)

CO	Outcome	Bloom's Taxonomy Level
CO1	Outline the structure and features of the Constitution of India	L1
CO2	Interpret key constitutional and legal concepts	L2
CO3	Examine the functioning of government institutions	L4
CO4	Relate legal principles to engineering and societal issues	L3
CO5	Assess ethical, legal, and professional responsibilities	L5

Broad Course Description



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakagalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

This course provides a comprehensive introduction to the Constitution of India and the basic principles of law that govern society and the nation. It explains the fundamental features, structure, and functioning of the Indian legal and constitutional system, including Fundamental Rights, Duties, Directive Principles, and the role of the Judiciary. The course also explores the importance of constitutional values such as justice, equality, liberty, and secularism in maintaining democracy and social harmony. Further, it examines the relevance of legal awareness and ethical responsibility in professional and social life, especially for students and future professionals.

Detailed Course Syllabus

The course consists of five modules. Each module integrates:

- M Core topics
- Engineering applications
- AI integration
- Authentic Intelligence reflection

Course Modules

Module 1: FOUNDATIONS OF THE INDIAN CONSTITUTION (03 Hours)

Topics: Meaning of the Constitution and Constitutionalism – Indian Constitution and its Salient Features – Sources of the Indian Constitution and Global Constitutional Influences – The Preamble of the Constitution.

Engineering Applications:

- Interpreting the Preamble and constitutional values in governance and public policy
- Relating constitutional principles to ethical engineering practices
- Evaluating laws and policies in technology-driven environments
- Understanding the role of AI integration within constitutional frameworks

Module 2: FUNDAMENTAL RIGHTS, DIGITAL RIGHTS & DUTIES (03 Hours)

Topics: Importance and Features of Fundamental Rights – Types of Fundamental Rights –



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakagalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Right to Privacy, Data Protection and Digital Rights in the AI Era – Directive Principles of State Policy – Fundamental Duties.

Engineering Applications:

- Protecting individual rights in real-life and digital environments
- Understanding data privacy, cyber security, and digital rights in the AI era
- Applying Directive Principles in governance and public welfare systems
- Promoting justice, equality, and ethical responsibility in technological innovation

Module 3: JUDICIARY, TECHNOLOGY AND JUDICIAL REVIEW (03 Hours)

Topics: Independence of the Judiciary in Democratic Governance – Judicial Review and Judicial Activism – Role of Courts in Regulating AI and Digital Platforms – Public Interest Litigation.

Engineering Applications:

- Applying judicial review in regulating AI and emerging technologies
- Understanding legal accountability of digital platforms and online systems
- Using Public Interest Litigation to address social and technological issues
- Promoting justice and rights protection in a digital society

Module 4: AMENDMENT AND EMERGENCY PROVISIONS (03 Hours)

Topics: Amendment of the Constitution: Power and Procedure – Historical Perspective of Constitutional Amendments in India – Emergency Provisions – National Emergency – State Emergency – Financial Emergency.

Engineering Applications:

- Understanding constitutional changes affecting technology and governance
- Evaluating emergency measures in digital and cyber-security contexts
- Analysing the role of constitutional safeguards during national crises
- Studying the balance between state power and individual rights

Module 5: INTELLECTUAL PROPERTY RIGHTS, INNOVATION & AI (03 Hours)

Topics: Overview and Importance of Intellectual Property Rights – Importance of Human Creativity in the Present Scenario – Patents – Trademarks – Copyrights – Artificial Intelligence and Emerging Technologies in Innovation.

Engineering Applications:

- Protecting technological innovations through patents, trademarks, and copyrights
- Encouraging ethical and creative use of AI technologies
- Applying IPR laws in engineering and research innovations
- Managing ownership, originality, and legal rights in AI-driven creations

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

- **Module-1:** This module reflects AI integration by helping students understand the foundations of the Indian Constitution, constitutional values, and governance principles,



while analysing the role of AI-driven policies, digital governance, and constitutional interpretation in modern society.

- **Module-2:** This module reflects AI integration by emphasizing the protection of fundamental rights, digital rights, privacy, and data protection in the AI era, enabling students to understand ethical and legal challenges arising from artificial intelligence and emerging technologies.
- **Module-3:** This module reflects AI integration by focusing on the role of the judiciary and judicial review in regulating AI systems, digital platforms, and online governance, while promoting accountability, transparency, and protection of rights in technology-driven environments.
- **Module-4:** This module reflects AI integration by encouraging students to examine constitutional amendments, emergency provisions, and governance mechanisms in relation to cyber security, digital emergencies, and technology-based administrative systems.
- **Module-5:** This module reflects AI integration by promoting awareness of Intellectual Property Rights, innovation, and AI-based creations, enabling students to understand legal ownership, ethical use of technology, patents, copyrights, and trademarks in the digital age.

Authentic Intelligence Reflection:

- **Module-1:** This module reflects authentic intelligence by encouraging students to understand constitutional values such as justice, liberty, equality, and fraternity, and apply them responsibly in social, professional, and technological contexts.
- **Module-2:** This module reflects authentic intelligence by promoting ethical thinking, protection of human dignity, responsible use of digital technologies, and awareness of individual duties and social responsibilities in democratic society.
- **Module-3:** This module reflects authentic intelligence by developing critical thinking regarding judicial independence, judicial activism, and public interest litigation, helping students evaluate social and technological issues through legal reasoning.
- **Module-4:** This module reflects authentic intelligence by helping students analyze constitutional amendments and emergency powers critically, ensuring balanced understanding between national security, governance, and protection of citizens' rights.
- **Module-5:** This module reflects authentic intelligence by encouraging creativity, innovation, and ethical responsibility in the use of Artificial Intelligence and emerging technologies, while respecting intellectual property and human contribution.



Research Integration Teaching

Students review on **at least one recent research topics** such as:

Data Privacy and Digital Rights in the AI Era

- Role of Judiciary in Regulating Artificial Intelligence
- Cyber Laws and Constitutional Protection in Digital Society
- Intellectual Property Challenges in AI-Generated Content
- Ethical and Legal Issues in Artificial Intelligence and Emerging Technologies
- Judicial Activism and Public Interest Litigation in Technology Governance
- AI-Based Surveillance and Right to Privacy

Students present a **short research review**.

Assessment Scheme

Definitions

1. **“Internal Assessment (IA)”** refers to the Continuous Internal Assessment (CIA) with a weightage of 60%. It includes:
 - a. **Mid Semester Examination (MSE):** One descriptive examination to evaluate students’ understanding of constitutional principles, legal concepts, governance, and contemporary legal issues related to technology and society.
 - b. **Continuous Assessment Tools (CAT – Six Components):** Includes activities such as quizzes, surprise tests, case studies, group discussions, seminars, project-based learning, reflection notes, presentations, assignments, constitutional case analysis, or open book assignments.

A. Professional Core Courses (Theory)

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT) – CBT	30 Marks
Assignment	05 Marks
Notes Submission	05 Marks
Presentation / Seminar	10 Marks
Total	50 Marks

Suggested Reading

Primary Textbooks

1. Introduction to the Constitution of India by D. D. Basu, 25th edition, 2021 published by LexisNexis
2. Indian Constitutional law by M. P. JAIN, 8th edition, 2022 published by LexisNexis
3. The Indian Constitution: Cornerstone of a nation by Granville Austin, 2nd edition, 2000 published by Oxford University Press.

Additional Resources

1. Gautam Bhatia: The Indian Constitution: A Conversation with Power. Oxford University Press, 2019.



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

2. Purvi Pokhariyal, Amit K. Kashyap, and Arun B. Prasad: Artificial Intelligence: Law and Policy Implications. Eastern Book Company, 2021.
3. P. Narayanan: Intellectual Property Law, Latest Edition 2017 Eastern Law House, Latest Edition.



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakagalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka - 562 112

CALCULUS & MATHEMATICAL MODELING

[As per Choice Based Credit System (CBCS) scheme]

Course Code: 26 XXXX	Semester: II
Programs Offered: CSE	Prerequisites:
Course Type:	Hours/Week: 03 Hrs
Total Hours: 45 hours (L)	Credits: 3
L - T - P - J: 3-0-0-0(3 Credits Theory)	

Course Vision

To develop strong mathematical foundations in modelling, multivariable calculus, vector calculus, and artificial intelligence, so that students can analyze complex engineering problems and design intelligent data-driven systems responsibly. .

Course Objectives

This course will enable students to:

1. **Understand** the principles of mathematical modelling for representing real-world systems.
2. **Apply** multivariable calculus to analyze engineering and computational problems.
3. **Evaluate** multiple integrals and differential equations for modelling dynamic systems.
4. **Use** vector calculus for modelling engineering and scientific systems.
5. **Discuss** artificial intelligence search algorithms and reasoning methods used in intelligent decision-making systems.



Course Outcomes (CO)

At the end of the course, students are able to

CO	Outcome	Bloom's Taxonomy Level
CO1	Understand mathematical modelling concepts including deterministic, probabilistic, static, and dynamic models and apply function fitting techniques for real-world and AI data modelling.	L2
CO2	Apply multivariable calculus including partial derivatives, Jacobians, Taylor expansion, and optimization methods such as Lagrange multipliers in AI and engineering problems.	L3
CO3	To evaluate double and triple integrals and apply coordinate transformations in computational modelling and probabilistic AI systems.	L4
CO4	Apply vector calculus concepts such as gradient, divergence, curl, and vector integrals to solve engineering and computational problems.	L3
CO5	Design and apply search algorithms, logical reasoning, and probabilistic inference techniques in artificial intelligence systems and intelligent decision-making models.	L6

Broad Course Description

The course Calculus & Mathematical Modeling provides a comprehensive foundation in mathematical techniques and computational approaches used for solving real-world engineering and intelligent system problems. It covers mathematical modelling, multivariable calculus, multiple integrals, differential equations, Laplace transforms, and introductory



artificial intelligence concepts including search algorithms and logical reasoning. The course emphasizes analytical thinking, optimization, and data-driven problem solving through practical applications in engineering, robotics, control systems, economics, and AI-based decision-making systems. Laboratory exercises using MATLAB/Python enable students to visualize mathematical concepts, implement computational models, and develop problem-solving skills relevant to modern engineering and intelligent technologies.

Detailed Course Syllabus

The course consists of five modules. Each module integrates:

- Core topics
- Integrated Lab Component
- (Nvidia DLI mapping)

Course Modules

Module 1: Introduction to Mathematical Modelling- 9 Hrs

Introduction to Mathematical Modelling: Definitions, importance of modelling in real-world problems, types of mathematical models, deterministic models, probabilistic models, static and dynamic models, functions and their role in modelling, function fitting.

Engineering Applications: Engineering system modelling, Population growth prediction, Economic forecasting, Climate modelling

Module 2: Multivariable Differential Calculus- 9 Hrs

Functions of two or more variables, level curves, limits and continuity, partial derivatives, chain rules, total differential, Jacobian, Taylor's expansion for two variables, linear approximation and increment estimation, maxima, minima and saddle points, constrained maxima and minima, Lagrange's multiplier method.

Engineering Applications: Engineering design optimization, Economics and production optimization, Surface modelling in computer graphics, Robotics motion planning

Module 3: Multiple Integrals- 9 Hrs

Evaluation of double integrals, change of order of integration, change of variables between Cartesian and polar coordinates, evaluation of triple integrals, change of variables between Cartesian coordinates, error function and its properties.

Engineering Applications: Volume and area computation in engineering systems, Probability distribution modelling, Fluid dynamics simulations, Heat distribution modelling



Module 4: Vector Calculus - 9 Hrs

Introduction to vector calculus, scalar and vector valued functions, gradient, tangent plane, directional derivative, divergence and curl, scalar and vector potentials, Statement of vector identities and simple problems, Line, surface and volume integrals, Statement of Green's theorem, Stokes' theorem and Gauss divergence theorem.

Engineering Applications: Fluid flow analysis, Electromagnetic field modelling, Robotics motion planning, Computer graphics and physical simulations

Module 5: Artificial Intelligence and Reasoning- 9 Hrs

Introduction to machine learning, AI search algorithms: informed and uninformed search, adversarial search, Introduction to logic: propositional logic and predicate logic, reasoning under uncertainty, conditional independence representation.

Engineering Applications: Game playing AI systems, Autonomous navigation systems, Expert systems for decision support, Intelligent recommendation systems

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

- Mathematical models used in machine learning systems
- Gradient computation in machine learning
- Numerical computation in AI models
- Dynamic system modelling in AI robotics
- Search algorithms used in robotics and navigation.

Authentic Intelligence Reflection:

- Understanding limitations of models in real-world applications
- Ethical interpretation of optimization results in AI systems
- Correct interpretation of numerical results in data-driven systems
- Understanding limitations of dynamic models
- Ethical implications of automated decision making

Laboratory Component (1 Credit)

S.No	Exercise Title	Aim / Objective	Tools	Hrs	CO
MODULE I: Exercises 1–3: Introduction to Mathematical Modelling (03 Hours)					
S.No	Exercise Title	Aim / Objective	Tools	Hrs	CO



1	Curve Fitting Techniques.	Implement curve fitting techniques on real-world datasets.	MATLAB/PYTHON	1 h	C01
2	Function Plotting & Analysis	Plot and analyze different types of mathematical functions	MATLAB/PYTHON	1 h	C01
3	Polynomial Regression	Implement polynomial regression to approximate nonlinear data patterns.	MATLAB/PYTHON	1 h	C01
MODULE II: Exercises 4–6 : Multivariable Calculus and Optimization (03 Hours)					
4	Tangent & Rate of Change	Visualize functions and their rates of change using tangent lines.	MATLAB/PYTHON	1 h	C02
5	Single and multi-variable function	Evaluate maxima and minima (extrema) of a single and multi-variable function and Plot 3D surfaces	MATLAB/PYTHON	1 h	C02
6	Lagrange Multipliers	Apply Lagrange Multiplier method	MATLAB/PYTHON	1 h	C02
MODULE III: Exercises 7–9: Multiple Integrals (03 Hours)					
7	Integration & Area	Understand integration as an area under the curve through visualization.	MATLAB/PYTHON	1 h	C03
8	Volume Under Surfaces	Evaluate volume under surfaces using integration techniques.	MATLAB/PYTHON	1 h	C03
9	Triple Integrals	Compute triple integrals for volume calculations	MATLAB/PYTHON	1 h	C03



S.No	Exercise Title	Aim / Objective	Tools	Hrs	CO
MODULE IV: Exercises 10–12: Vector Calculus (03 Hours)					
10	Gradient Computation	Introduction to computational tools and compute gradient of scalar functions	MATLAB/PYTHON	1 h	C04
11	Tangent Plane & Directional Derivative	Compute divergence and curl of vector fields, Visualize tangent planes and directional derivatives	MATLAB/PYTHON	1 h	C04
12	Line Integrals	Implement numerical computation of line integrals	MATLAB/PYTHON	1 h	C04
MODULE V: Exercises 13–15: Artificial Intelligence and Reasoning (03 Hours)					
13	Understanding the Search algorithm	Implement informed search algorithm for pathfinding problems	MATLAB/PYTHON	1 h	C05
14	AI Techniques Implementation	Implement search algorithms, logic inference, and Bayesian reasoning.	MATLAB/PYTHON	1 h	C05
15	Case Study of AI Search Algorithm	Develop a simple propositional logic inference engine	MATLAB/PYTHON	1 h	C05

Assessment Scheme

Definitions

1. **“Internal Assessment”** refers to the Continuous Internal Assessment (CIA). The weightage of this component is **60%**. It includes:
 - a. One **Mid Semester Examination (MSE)** (Descriptive-based),
 - b. Six **Continuous Assessment Tool (CAT)**: It may include: Surprise test (Computer-based), Quiz, Project based learning, Reflection Note on Guest Lecture, Industrial visit, E-course certification, Building models, Group discussion, Case study, Seminar, or Open Book Assignment



- c. One **Project**: Use of AI tools for problem-solving while simultaneously developing Authentic Intelligence through critical evaluation, ethical reasoning, and responsible decision-making.
2. **“External Assessment”** refers to the Semester End Examinations (SEE). The weightage of this component is **40%**. At the end of semester, the final examination is conducted to evaluate student’s performance.

A. Integrated Professional Core Courses (Theory + Practical)

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT) every 2 weeks (Minimum 06 assessments including one Project)	20
Mid Semester Examination (MSE)	20
Lab Record	5
Lab Examination	15
Total	60 marks

Recommended Tools

- Python
- MATLAB

Suggested Reading

Primary Textbooks

1. Thomas' Calculus, George B. Thomas Jr., Maurice D. Weir, Joel Hass, and Christopher Heil, 15th Edition, Pearson Education, 2023.
2. Advanced Engineering Mathematics, Erwin Kreyszig, 11th Edition, Wiley, 2025.
3. Artificial Intelligence: A Modern Approach, Stuart Russell and Peter Norvig, 4th Edition, Pearson Education, 2020.
4. Discrete Mathematics and Its Applications, Kenneth H. Rosen, 8th Edition, McGraw-Hill Education, 2019.

Additional Resources



1. Strang, Gilbert. Linear algebra and learning from data. Vol. 4. Cambridge: Wellesley-Cambridge Press, 2019.
2. Goodfellow, Ian, et al. Deep learning. Vol. 1. No. 2. Cambridge: MIT press, 2016.
3. Chapra, Steven C. Applied numerical methods. Vol. 22. Columbus: Mcgraw-hill, 2012.

Strength of CO Mapping to PO/PSOs:

Table: Mapping Levels of Cos to Pos / PSOs													
COs	Program Outcomes (POs)											PSOs	
	1	2	3	4	5	6	7	8	9	10	11	1	2
C01	3	3	2	2	2	1			1	1		2	
C02	3	3	3	2	2				1	1		3	1
C03	3	3	2	3	2				1	1		3	2
C04	3	3	3	3	2	1			1	1		3	2
C05	3	3	3	3	3	1		1	2	2	1	2	2
Avg	15/ 15	15/ 15	13/ 15	8/ 15	11/ 15	3/ 15	0/ 15	1/ 15	6/ 15	6/ 15	1/ 15	13/ 15	7/ 15

Legend: 3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)



Applied Statistics & Stochastic Processes

Course Code: 26DS2301

SEMESTER – III

Credits: 03

L-T-P-J: 3-0-0-0

Course Vision

To develop strong foundations in probability, statistical inference, and stochastic processes, enabling students to analyze uncertainty, make data-driven decisions, and explore advanced computational models including AI systems.

Course Objectives

1. Understand the concepts of random variables and probability distributions for modelling uncertainty.
2. Analyze relationships between variables using correlation and regression techniques.
3. Apply statistical inference methods for decision-making under uncertainty.
4. Use parametric and non-parametric tests for hypothesis testing and data analysis.
5. Understand the fundamentals of stochastic processes and their applications in modeling random systems, decision-making, and AI-driven applications.

Course Outcomes (CO)

CO	Description	Bloom's Level
CO1	Apply probability concepts and random variables to analyze uncertainty in engineering and real-world problems.	L3
CO2	Analyze data using correlation and regression techniques to model and interpret relationships between variables.	L4
CO3	Perform hypothesis testing and statistical inference to support data-driven decision making.	L4
CO4	Apply non-parametric tests for categorical data analysis and evaluate goodness of fit.	L3
CO5	Model and analyze stochastic processes such as Markov chains, Poisson processes, and Monte Carlo simulation.	L5



CO-PO mapping

COs	Program Outcomes (POs)											PSOs		
COs	1	2	3	4	5	6	7	8	9	10	11		1	2
CO1	3	2			2					1		2		
CO2	3	3	2	2	3				1	1		3	2	
CO3	3	3	3	2	3				1	2		3	2	
CO4	3	2	2	2	2					1		2	2	
CO5	3	3	2	3	3			1		1	2	3	2	
3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)														

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes:

1. Lecture Method: Concepts, algorithms, and models are explained through structured lectures supported by R Programming demonstrations.
2. Interactive Teaching: Active learning is promoted through discussions, brainstorming, and group problem-solving activities.
3. Demonstration using R Programming and Python: AI techniques like classification and optimization are demonstrated to visualize results and understand implementation.
4. Multimedia Learning: Videos and simulations are used to explain complex concepts and real-world applications.
5. Collaborative Learning: Students work in groups to implement projects and analyze datasets using computational tools.
6. Higher Order Thinking Questions: Analytical questions are used to develop model design, evaluation, and decision-making skills.
7. Problem-Based Learning: Real-world problems are given to apply techniques using Python for practical solutions.



Course Modules

MODULE – I Random Variables and Probability Distributions (09 Hours)

Topics: Introduction to random variables, probability mass function (PMF), probability density function (PDF), distribution functions, joint probability distribution and joint density functions, marginal and conditional distributions, mathematical expectation and its properties, covariance, moment generating function, characteristic function.

Applications: Risk analysis and financial modelling, Reliability engineering, Data analysis and forecasting, Signal processing.

MODULE – II Bi-variate Data, Correlation and Regression Analysis (09 Hours)

Topics: Bi-variate random variables, correlation, types of correlation, measures of correlation, graphical methods (scatter diagram), Karl Pearson's correlation, rank correlation (for repeated and non-repeated ranks), linear regression model, regression lines, least squares method, fitted model.

Applications: Market trend analysis, Economic forecasting, Quality control and process analysis, Data relationship analysis.

MODULE – III Inference for Decision Making – I (09 Hours)

Topics: Introduction to hypothesis, hypothesis testing procedure, p-values and decision making, test of hypothesis for proportion (single and difference of proportions), testing hypothesis about mean (Z-test and t-test), two-sample tests, test for equality of variance (F-test), ANOVA (one-way analysis of variance).

Applications: Quality assurance and control, Clinical trials and testing, Business decision-making, Industrial process evaluation.

MODULE – IV Inference for Decision Making – II (09 Hours)

Topics: Introduction to non-parametric tests, Chi-Square test for goodness of fit, Chi-Square test using probability distributions, Chi-Square test for independence of attributes.

Applications: Survey data analysis, Market research, Pattern recognition, Categorical data analysis.

MODULE – V Stochastic Processes (09 Hours)

Topics: Introduction to Stochastic Processes (SPs), Markov chains, Poisson process, Discrete-time Markov Chains (DTMCs), Continuous-time Markov Chains (CTMCs), Monte Carlo Simulation.

Applications: Queueing systems and waiting line models, Reliability and maintenance systems, Population growth and epidemic modelling, Financial market analysis.

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

Authentic Intelligence:

- Understanding uncertainty in AI predictions
- Avoiding misleading interpretations of correlations
- Interpreting statistical results responsibly
- Ethical handling of categorical data
- Ethical use of simulation-based decision models



AI Integration:

- Feature distribution modelling in data science
- Regression models in machine learning
- Statistical decision-making in AI systems
- Model validation using statistical tests
- Markov models in reinforcement learning and decision processes

Assessment Scheme based on the course

Criteria	Weightage of Marks
Continuous Internal Assessment (CIA) – Formative	60
Semester End Examination (SEE) – Summative	40
Total	100 marks

TEXT BOOKS

1. Mendenhall, William, Robert J. Beaver, and Barbara M. Beaver. Introduction to probability and statistics. Cengage Learning, 2020.
2. Ross, Sheldon, M. (1995). Stochastic Processes (2nd edition), John Wiley.

REFERENCE BOOKS

1. R.E. Walpole, R.H. Mayers, S.L. Mayers and K.Ye, Probability and Statistics for engineers and scientists, 9th Edition, Pearson Education, 2018.
2. Douglas A. Wolfe, Grant Schneider, Intuitive Introductory Statistics, Springer, 2017.
3. Miller & Freund's, Probability and statistics for engineers, 8th edition, Pearson publication, 2018.
4. Richard I. Levin, David S. Rubin, Masood H. Siddiqui, Sanjay Rastogi, Statistics for Management, 8th Edition, Pearson Publications, 2018.
5. S. C. Gupta and V. K. Kapoor, Fundamentals of Applied Statistics, S. Chand, 2006.
6. Mark L. Berenson, David M. Levine, Kathryn A. Szabat, Basic Business Statistics Concept and Application, Thirteenth Edition, Pearson Education Limited, 2015.
7. Bruce L. Bowerman, Richard T. O'Connell, Emily S. Murphree, Business Statistics in Practice, Seventh Edition, McGraw-Hill, 2014.
8. Douglas C. Montgomery, George C. Runger, Applied Statistics and Probability for Engineers, Sixth Edition, John Wiley and Sons, 2014.
9. Noreen R. Sharpe, Richard D. De Veaux, Paul F. Velleman, Business Statistics, Fourth Edition, Pearson Education, 2019.
10. Trivedi, K. S. (2016), Probability & Statistics with Reliability, Queuing & Computer Science Applications (2nd edition), New Jersey, John Wiley & Sons, Inc.



E-RESOURCES

Refer to course portal for updated links to NPTEL, MIT OCW, and relevant open-access resources.

RECOMMENDED TOOLS

- Python
- R Programming
- NumPy
- Pandas
- Matplotlib
- Scikit-learn

Lab Exercises

S. No	Exercise Title	Aim / Objective	Tools	Hrs	CO
MODULE I: Exercises 1–3: Random Variables and Probability Distributions (03 Hours)					
1	Random Variable Visualization	Generate and visualize discrete and continuous random variables	R Programming	1 h	CO1
2	PMF, PDF & CDF	Compute PMF, PDF, and CDF for standard distributions	R Programming	1 h	CO1
3	Joint Distributions	Simulate joint probability distributions	R Programming	1 h	CO1
MODULE II: Exercises 4–6: Bi-variate Data, Correlation and Regression Analysis (03 Hours)					
4	Bivariate Data Visualization	Visualize bivariate data using scatter plots	R Programming	1 h	CO2
5	Correlation Analysis	Compute Karl Pearson's and Spearman's rank correlation	R Programming	1 h	CO2
6	Linear Regression	Fit and evaluate regression model	R Programming	1 h	CO2
MODULE III: Exercises 7–9: Inference for Decision Making – I (03 Hours)					
7	Z-Test & t-Test	Perform Z-test and t-test (single & two-sample)	R Programming	1 h	CO3



8	Proportion Testing	Test for proportion and difference of proportions	R Programming	1 h	CO3
9	F-Test & ANOVA	Perform F-test and one-way ANOVA	R Programming	1 h	CO3
MODULE IV: Exercises 10–12: Inference for Decision Making – II (03 Hours)					
10	Chi-Square Tests	Perform goodness of fit and independence tests	R Programming	1 h	CO4
11	Bayesian Inference	Implement basic Bayesian inference techniques	R Programming	1 h	CO4
12	Data Visualization	Visualize probability distributions and statistical results	R Programming	1 h	CO4
MODULE V: Exercises 13–15: Stochastic Processes (03 Hours)					
13	Monte Carlo Simulation	Perform Monte Carlo simulations for probabilistic models	R Programming	1 h	CO5
14	Markov Chains	Construct transition matrices and simulate state transitions	R Programming	1 h	CO5
15	Exponential Distribution	Study exponential distribution and waiting time analysis	R Programming	1 h	CO5



Data Structures

Course Code: 26DS2302

Semester: III

Programs Offered: B. Tech CSE (Data Science)

Course Type: Core

Credits: 4

L – T – P – J: 3 – 0 – 2 – 0

Prerequisite(s): Basic programming knowledge (C/Python)

Course Vision

To develop strong problem-solving skills by equipping students with fundamental and advanced data structures, enabling efficient data organization, manipulation, and algorithmic design for real-world computing applications.

Course Objectives

The course is designed to:

1. Introduce fundamental concepts of data structures and their role in efficient computation.
2. Develop the ability to analyze time and space complexity of algorithms.
3. Enable implementation and application of linear and non-linear data structures.
4. Provide knowledge of hierarchical and graph-based data structures for complex problem solving.
5. Introduce hashing and advanced data structures used in modern computing systems.

Course Outcomes (CO)

CO	Description	Blooms Level
CO1	Understand and classify data structures and analyze algorithm complexity.	L2
CO2	Implement and apply linear data structures such as arrays, linked lists, stacks, and queues.	L3
CO3	Analyze and implement tree-based structures and their applications.	L4
CO4	Apply graph algorithms for solving real-world problems.	L3
CO5	Design efficient solutions using hashing and advanced data structures.	L6

Broad Course Description

The Data Structures course provides a comprehensive foundation in organizing and managing data efficiently. It introduces both linear and non-linear data structures, focusing on their design, implementation, and application in solving computational problems. Students learn how to evaluate algorithm efficiency using time and space complexity analysis.

The course progresses from basic structures like arrays and linked lists to advanced structures such as trees, graphs, and hashing techniques. Emphasis is placed on practical implementation



and problem-solving skills, preparing students for software development, competitive programming, and advanced computing domains such as AI, databases, and systems design.

CO-PO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
CO1	2	1			1						1	2	1
CO2	2	2			2						1	2	2
CO3	2	3	2	2	2			1			1	2	2
CO4	2	3	2	2	3			1			2	3	2
CO5	2	3	3	2	3						2	3	2
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module-1 Introduction to Data Structures and Arrays

(9 Hours)

Introduction to data structures, Classification: Linear and Non-linear structures, Abstract Data Types (ADT), Time and space complexity basics, Arrays: Representation and operations (insertion, deletion, traversal), Multidimensional arrays, Sparse matrices

Module-2 Linear Data Structures-Linked Lists, Stacks, and Queues

(9 Hours)

Linked Lists: Singly, doubly, and circular linked lists, Operations on linked lists, Stacks: Implementation using arrays and linked lists, Applications of stacks: Expression evaluation, recursion, Queues: Simple queue, circular queue, Priority queue and deque.

Module-3 Non-Linear Data Structures-Trees

(9 Hours)

Tree terminology and representation, Binary trees, Tree traversals: Inorder, preorder, postorder, Binary Search Trees (BST), AVL trees (self-balancing trees), Heap data structure and operations, Priority queues using heap

Module-4 Graphs

(9 Hours)

Graph representation: Adjacency matrix and adjacency list, Graph traversal algorithms: Breadth-First Search (BFS), Depth-First Search (DFS), Connected components, Shortest path basics, Introduction to spanning trees.

Module-5 Hashing and Advanced Data Structures

(9 Hours)

Hashing concepts and hash functions, Collision resolution techniques: Chaining, open addressing, Hash tables and applications, Introduction to advanced data structures: Trie, B-Tree basics.



AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module-1: Students use AI-supported coding environments such as Python (Colab) and visualization tools to understand data structures and analyze time and space complexity. Simulation tools help visualize array operations (insertion, deletion, traversal) and multidimensional data representation. AI-assisted tools are used to analyze performance efficiency and optimize basic algorithms.

Module-2: AI-enabled platforms assist in implementing and visualizing linear data structures such as linked lists, stacks, and queues. Interactive tools help simulate dynamic memory allocation and operations. AI-based debugging and code suggestion tools support students in solving problems like expression evaluation and recursion efficiently.

Module-3: Students use simulation and visualization tools to understand hierarchical data structures such as trees. AI-based tools assist in visualizing tree traversals, balancing techniques (AVL trees), and heap operations. Machine learning concepts may be introduced to analyze search efficiency and optimization in tree-based structures.

Module-4: AI tools and graph visualization platforms are used to simulate graph representations and algorithms such as BFS and DFS. Students explore AI-assisted solutions for real-world problems like shortest path and network analysis. Data-driven approaches help in understanding connectivity and optimization in graphs.

Module-5: Advanced AI tools are used to explore hashing techniques and complex data structures such as Tries and B-Trees. Students utilize intelligent systems to analyze collision handling strategies and optimize search operations. Exposure to AI-driven applications such as search engines, databases, and indexing systems enhances practical understanding.

List of Experiments

No.	Experiment	Module Link
1	Application of Array – single and multi-dimensional Array -application in real-time scenario-vehicle number Validation	Module 1
2	Application of Matrix -Sudoku board	Module 1
3	Application of Structures and Pointers	Module 2
4	Application of concept stack using arrays	Module 2
5	Application of concept of Queue using arrays	Module 3
6	Application of Singly Linked List	Module 3
7	Application of Doubly Lined List	Module 4
8	Application of Circular Linked List	Module 4
9	Application of Binary Search Tree	Module 5
10	Open Ended Questions	



Assessment Scheme

Criteria	Weightage
Lab Reports (experiments)	10
Class Tests /assignments	10
Mid-Semester Examination	30
Hackathon/Technical Competition	10
End-Semester Examination	40
Total	100 Marks

Recommended Tools

- Programming Languages: C, C++, Python
- Platforms: HackerRank, LeetCode, CodeStudio
- Visualization Tools: VisuAlgo

Primary Textbooks

1. Data structures and algorithm analysis in C++ / Mark Allen Weiss, Florida International University. — Fourth edition. Pearson, 2013.
2. Yedidyah, Augenstein, Tannenbaum: Data Structures Using C and C++, 2nd Edition, Pearson Education, 2003.
3. Data Structures, Seynour Lipschutz and GAV Pai, Schaum's Outlines, McGraw Hill, 2008

Additional Resources

1. NPTEL Data Structures Courses
2. MIT OpenCourseWare (Algorithms & Data Structures)
3. GeeksforGeeks, TutorialsPoint



Computer Organization and Architecture

Course Code: 26DS2303

Semester: III

Programs Offered: B. Tech CSE (Data Science)

Course Type: Core

Credits: 3

L – T – P – J: 3 – 0 – 0 – 0

Prerequisite(s): NIL

Course Vision

To develop a strong foundation in computer system design by understanding digital logic, processor organization, memory hierarchy, pipelining, and modern GPU-based computing, enabling students to design efficient and high-performance computing systems aligned with emerging parallel and AI-driven architectures.

Course Objectives

1. Develop a strong understanding of digital logic design and enable students to design and analyze combinational and sequential circuits for computing systems.
2. Build proficiency in arithmetic and logic unit operations and enable analysis of instruction execution using hardware-level concepts and computational tools.
3. Equip students with knowledge of memory hierarchy and organization and apply performance optimization techniques for efficient data access and storage.
4. Provide a strong foundation in pipelining and parallel processing concepts and enable evaluation of system performance using modern architectural techniques.
5. Enable students to understand GPU architecture and NVIDIA-based computing platforms and apply parallel computing concepts for high-performance applications.

Course Outcomes (CO)

CO	Description	Blooms Level
C01	Design digital logic circuits and explain fundamental computer organization concepts.	L6
C02	Analyze arithmetic and logic unit operations and instruction execution processes in computing systems.	L4
C03	Apply memory organization and hierarchy techniques for efficient system performance.	L3
C04	Evaluate system performance using pipelining and parallel processing techniques.	L5
C05	Design modern computing systems including GPU-based architectures and parallel computing platforms.	L6



Broad Course Description

Computer Organization and Architecture is a core engineering course that provides a comprehensive understanding of the internal structure and functioning of computer systems. The course covers digital logic design, arithmetic and logic unit operations, memory organization, pipelining, and modern GPU-based computing. It focuses on how hardware components interact to execute instructions efficiently and how system performance can be optimized using architectural techniques.

The course integrates modern computing concepts such as parallel processing and GPU-based acceleration, enabling students to understand real-world high-performance computing systems. Students explore architectural design principles, performance optimization strategies, and emerging technologies such as CUDA-based computing. The course emphasizes analytical thinking, system design, and performance evaluation, preparing students for advanced studies and industry applications in computer architecture and high-performance computing.

CO-PO mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
C01	2	1	2		1						1	2	1
C02	2	3	2	2	1						1	2	2
C03	2	3	2	2	2						2	2	2
C04	2	3	2	3	2						2	3	2
C05	2	3	3	3	2						2	3	2
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module-1: Digital Logic Design

(9 Hours)

Number systems and conversions, binary arithmetic operations, Boolean algebra and laws, logic gates and truth tables, universal gates NAND and NOR, simplification using Boolean algebra, Karnaugh maps, combinational circuits, half adder and full adder, multiplexer and demultiplexer, encoder and decoder, sequential circuits, flip flops SR JK D T, registers and shift registers, counters synchronous and asynchronous.

Module-2: Arithmetic and Logic Unit

(9 Hours)

Binary arithmetic operations, signed number representation, floating-point representation, arithmetic algorithms including Booth multiplication and division algorithms, fixed-point and floating-point arithmetic, ALU design, logic operations, bitwise operations, overflow and underflow conditions, error detection basics, carry look-ahead adder, performance of arithmetic circuits.



Module-3: Memory Organization

(9 Hours)

Memory hierarchy, cache memory organization, mapping techniques direct associative and set associative, cache coherence and consistency, virtual memory concepts, paging and segmentation, translation lookaside buffer, memory management techniques, performance metrics such as hit ratio and miss penalty, secondary storage systems, RAID concepts, memory interleaving, associative memory.

Module-4: Pipeline Concepts

(9 Hours)

Introduction to pipelining, pipeline architecture and stages, instruction pipeline, arithmetic pipeline, pipeline hazards data structural and control, hazard detection techniques, hazard resolution methods, pipeline performance metrics, speedup and efficiency, instruction-level parallelism, superscalar architecture basics, out-of-order execution basics.

Module-5: Fundamentals of GPU Architecture

(9 Hours)

Introduction to GPU architecture, CPU vs GPU comparison, parallel processing fundamentals, SIMD and SIMT architectures, GPU memory hierarchy, CUDA architecture basics, thread and block organization, GPU programming fundamentals, performance optimization techniques, introduction to NVIDIA platforms, overview of NVIDIA Jetson, GPU-based computing applications.

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

Module 1: Students use digital simulation tools such as Logisim and Python-based logic simulators to design and validate combinational and sequential circuits.

Module 2: AI-assisted tools are used to simulate ALU operations and analyze arithmetic performance in computing systems.

Module 3: Students use simulation platforms to analyze cache performance, memory hierarchy, and access time optimization.

Module 4: Pipeline simulation tools and Python-based models are used to analyze hazards, speedup, and performance improvements.

Module 5: GPU simulation tools and CUDA-based environments are used to explore parallel processing and performance optimization in modern computing systems.

Assessment Scheme

Criteria	Weightage of Marks
Project	10
Mid-Semester Examination	30
Class Tests / Assignment	10
Hackathon	10



End-Semester Examination	40
Total	100 marks

Recommended Tools

- C
- C++
- Python
- Logisim
- Digital simulation tools
- CUDA programming tools
- NVIDIA Nsight tools
- MATLAB optional

Primary Textbooks

1. Hardware/Software Interface, 6th ed., Morgan Kaufmann (Elsevier), 2020.
2. M. Morris Mano, Computer System Architecture, 3rd ed., Prentice Hall, 1993.
3. Andrew S. Tanenbaum, Structured Computer Organization, 6th ed., Pearson Education, 2015.
4. John L. Hennessy and David A. Patterson, Computer Architecture: A Quantitative Approach, 6th ed., Morgan Kaufmann (Elsevier), 2019.
5. David Harris and Sarah Harris, Digital Design and Computer Architecture, 2nd ed., Morgan Kaufmann (Elsevier), 2015.

Additional Resources

1. NVIDIA CUDA Developer Platform – <https://developer.nvidia.com/cuda>
2. NVIDIA Training Portal – <https://www.nvidia.com/en-in/training/>
3. NPTEL Computer Architecture Lectures – <https://nptel.ac.in>
4. Online simulation tools and Python notebooks

Mini Project Component

The following projects are suggested based on course outcomes. Students may attempt one project individually or in pairs.

Project 1 – Cache Memory Simulator Problem: Design a simulator to evaluate cache memory performance Concepts Used: Cache mapping, hit ratio, memory hierarchy Industrial Relevance: Processor design, memory optimization

Project 2 – Pipeline Hazard Analyzer Problem: Analyze pipeline hazards and optimize instruction execution Concepts Used: Pipeline stages, hazard detection, performance metrics Industrial Relevance: CPU design and performance optimization



Project 3 – Instruction-Level Parallelism Visualizer Problem: Visualize parallel instruction execution in modern processors Concepts Used: ILP, superscalar architecture, execution units Industrial Relevance: High-performance computing systems

Project 4 – GPU-Based Parallel Processing Application Problem: Implement a parallel computation task using GPU Concepts Used: CUDA programming, parallel processing, memory hierarchy Industrial Relevance: AI systems, scientific computing, graphics processing

Project 5 – CUDA Performance Optimization Project Problem: Optimize execution of a parallel program using GPU Concepts Used: Thread optimization, memory access, execution efficiency Industrial Relevance: High-performance computing, AI acceleration

AI-Powered Assignments

Assignment 1: AI-Based Digital Logic Design and Validation

Students prompt an AI tool: Design a combinational circuit for a given Boolean function and simplify it. Students must:

1. Analyze the correctness of the generated logic expression and circuit design
2. Identify errors in simplification or gate implementation
3. Validate the design using Karnaugh maps and logic simulation tools
4. Assessment focuses on understanding of Boolean algebra, circuit design accuracy, and validation of AI outputs

Assignment 2: AI-Assisted ALU Simulation and Analysis

Students prompt an AI tool: Simulate an arithmetic logic unit and perform arithmetic and logical operations. Students must:

1. Analyze the correctness of operations such as addition, subtraction, and bitwise logic
2. Identify inconsistencies in overflow handling or operation results
3. Validate results using manual calculations and simulation tools
4. Assessment focuses on ALU design understanding, arithmetic accuracy, and evaluation of AI-generated outputs

Assignment 3: AI-Based Memory Hierarchy Performance Analysis

Students prompt an AI tool: Analyze cache performance and calculate hit ratio and access time. Students must:

1. Analyze the correctness of cache calculations and performance metrics
2. Identify incorrect assumptions in mapping techniques or memory hierarchy
3. Validate results using standard formulas and simulation tools
4. Assessment focuses on memory concepts, performance evaluation, and analytical validation

Assignment 4: AI-Based Pipeline Hazard Detection and Optimization

Students prompt an AI tool: Analyze a pipelined processor and identify hazards and performance improvements. Students must:

1. Analyze correctness of hazard detection and pipeline stages
2. Identify missing hazards or incorrect pipeline behaviour



3. Validate results using pipeline diagrams and performance metrics
4. Assessment focuses on pipeline concepts, hazard analysis, and optimization strategies

Assignment 5: AI-Based GPU Performance Analysis using CUDA

Students prompt an AI tool: Compare CPU and GPU performance for parallel computation tasks. students must:

1. Analyze correctness of performance comparison and execution results
2. Identify inefficiencies in parallel execution or memory usage
3. Validate results using CUDA-based simulations or benchmarking tools
4. Assessment focuses on parallel computing concepts, GPU efficiency, and performance validation.



IoT and Sensor Systems

Course Code: 26DS2304

Semester: III

Programs Offered: B. Tech CSE (Data Science)

Course Type: Core

Credits: 3

L – T – P – J: 2 – 0 – 2 – 0

Prerequisites: NIL

Course Vision

To develop the ability to design intelligent IoT and sensor systems by integrating real-time data acquisition, embedded platforms, sensor communication, and GPU-enabled edge computing for scalable and high-performance smart applications.

Course Objectives

1. Understand IoT architectures, embedded platforms such as Arduino and Raspberry Pi, and sensor interfacing techniques.
2. Analyze sensor data acquisition methods, communication protocols, and wireless sensor networks.
3. Apply intelligent sensor system techniques for real-time monitoring and decision making.
4. Evaluate system performance, latency, scalability, and edge computing techniques in IoT systems.
5. Design intelligent IoT and sensor systems integrated with GPU-based acceleration and real-time processing.

Course Outcomes (CO)

CO	Description	Blooms Level
C01	Illustrate IoT architectures, embedded platforms, and sensor technologies used in intelligent systems.	L2
C02	Analyze sensor data acquisition methods, communication protocols, and wireless sensor networks.	L4
C03	Apply sensor data preprocessing, feature extraction, and time-series analytics techniques.	L3
C04	Implement classification and clustering techniques for intelligent sensor data analysis.	L3
C05	Design intelligent IoT and sensor-based systems integrated with real-time analytics and GPU-enabled edge computing.	L5

Broad Course Description

This course provides a comprehensive understanding of IoT systems and sensor analytics, focusing on data acquisition, processing, and intelligent decision-making. It introduces embedded platforms such as Arduino and Raspberry Pi for building real-world IoT systems and explores how sensor data is collected, processed, and analyzed. The course further integrates advanced sensor analytics techniques including machine learning, time-series analysis, and real-



time processing. It also introduces GPU-based acceleration using NVIDIA platforms to enable high-performance data processing and AI-driven IoT applications.

CO-PO mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
C01	3	2			3	1						2	2
C02	3	3			3	1					1	2	2
C03	3	3	2		3	1					2	2	2
C04	3	3	2		3	1		1	1		2	3	2
C05	3	3	2		3	1		2	2		2	3	2
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module-1: IoT Systems and Embedded Platforms (6 Hours)

Introduction to Internet of Things, IoT architecture and components, embedded systems for IoT, Arduino architecture and programming, Raspberry Pi architecture and interfacing, sensor interfacing with Arduino and Raspberry Pi, IoT communication protocols MQTT HTTP CoAP, wireless communication technologies.

Module-2: IoT Data Acquisition and Communication (6 Hours)

Sensor types and classification, analog and digital sensors, sensor calibration and characteristics, signal conditioning, data acquisition systems, sampling and quantization, noise filtering techniques, wireless sensor networks, IoT gateways, cloud connectivity.

Module-3: Sensor Systems and Sensor Data Processing (6 Hours)

Sensor system architecture, smart sensors, sensor calibration techniques, sensor signal processing, sensor data preprocessing and cleaning, normalization techniques, feature extraction methods, statistical analysis of sensor data, time-series analysis, sensor data visualization.

Module-4: Intelligent Sensor Systems (6 Hours)

Classification and clustering of sensor data, anomaly detection in sensor networks, predictive techniques for intelligent sensors, real-time processing frameworks, latency optimization techniques.

Module-5: GPU Architecture and Real-Time Analytics (6 Hours)

Introduction to GPU computing, CUDA fundamentals, parallel processing concepts, GPU memory hierarchy, NVIDIA Jetson platform, sensor data processing using GPU.



List of Experiments

No.	Experiment	Modules
1	Develop a program to blink 5 LEDs back and forth.	Module I
2	Develop a program to interface a relay with Arduino board.	Module I
3	Develop a program to deploy an intrusion detection system using Ultrasonic and sound sensors.	Module II
4	Develop a program to control a DC motor with Arduino board.	Module II
5	Develop a program to deploy smart street light system using LDR sensor.	Module III
6	Develop a program to classify dry and wet waste with the Moisture sensor (DHT22).	Module III
7	Develop a program to read the pH value of a various substances like milk, lime and	Module IV
8	Develop a program to detect the gas leakage in the surrounding environment.	Module IV
9	Develop a program to demonstrate weather station readings using Arduino	Module V
10	Develop a water level depth detection system using Ultrasonic sensor	Module V

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module 1: Students use Arduino and Raspberry Pi platforms to collect and process real-time sensor data.

Module 2: Simulation tools are used for analyzing sensor signals, wireless communication, and IoT connectivity protocols.

Module 3: AI tools are used for sensor data preprocessing, feature extraction, visualization, and multi-sensor data fusion.

Module 4: Machine learning and anomaly detection models are applied for intelligent sensor systems and predictive decision making.

Module 5: GPU-based tools and CUDA environments are used for high-performance sensor data processing and AI acceleration.

Assessment Scheme based on the course

Criteria	Weightage of Marks
Lab Report (Experiments)	10
Assignment /Class Tests / Quizzes	10
Mid-Semester Examination	30
Hackathon/Technical Competition/ Paper presentation/Global/DLI certifications	10
End-Semester Examination	40
Total	100 marks



Recommended Tools

- Python (for data processing, analytics, and machine learning applications) TensorFlow or PyTorch (for building and deploying machine learning models for sensor analytics)
- Arduino (for IoT prototyping, sensor interfacing, and embedded system development)
- Raspberry Pi (for edge computing, IoT system implementation, and real-time data processing)
- MQTT and IoT Communication Platforms (for lightweight data transmission and device communication)
- NVIDIA Jetson Platform (for edge AI deployment and real-time GPU-based processing)
- CUDA Programming Tools (for parallel computing and GPU acceleration of sensor data processing)

Primary Textbooks

1. J. S. Wilson, Sensor Technology Handbook, Elsevier, 2005.
2. P. Lea, Internet of Things for Architects: Architecting IoT Solutions by Implementing Sensors, Communication Infrastructure, Edge Computing, Analytics, and Security, Packt Publishing, 2018.
3. Tamboli, Build Your Own IoT Platform: Develop a Fully Flexible and Scalable Internet of Things Platform in 24 Hours, Packt Publishing, 2019.
4. Goodfellow, Y. Bengio, and A. Courville, Deep Learning, MIT Press, 2016.
5. Adrian McEwen and Hakim Cassimally, Designing the Internet of Things, Wiley, 2013.

AI-Powered Assignments

Assignment 1: AI-Based IoT System Design using Arduino and Raspberry Pi

o Students prompt an AI tool: Design an IoT system using Arduino and Raspberry Pi for real-time data acquisition.

o Students must:

Analyze the correctness of system architecture and hardware interfacing

Identify errors in sensor integration, communication protocols, or data flow

Validate the design using simulation tools or prototype implementation

o Assessment focuses on IoT architecture understanding, embedded system design, and validation of AI-generated outputs

Assignment 2: AI-Based Sensor Data Preprocessing and Analysis

o Students prompt an AI tool: Clean and preprocess raw sensor data and perform statistical analysis.

o Students must:

Analyze correctness of preprocessing techniques such as normalization and filtering

Identify issues like noise, missing values, or incorrect transformations

Validate results using standard data analysis tools

o Assessment focuses on data preprocessing skills, analytical accuracy, and validation ability

Assignment 3: AI-Based Time-Series Prediction for Sensor Systems

Students prompt an AI tool: Predict future sensor readings using time-series techniques.

Students must:

Analyze correctness of prediction models and output trends.

Identify overfitting, underfitting, or incorrect assumptions.

Validate results using visualization and performance metrics.

Assessment focuses on predictive modeling, time-series analysis, and evaluation of AI outputs.

Assignment 4: AI-Based Anomaly Detection in Sensor Networks



Students prompt an AI tool: Detect anomalies in IoT sensor systems using machine learning techniques.
Students must:

Analyze correctness of anomaly detection results.

Identify false positives and missed anomalies.

Validate using datasets and performance metrics.

Assessment focuses on intelligent sensor systems, anomaly detection, and critical evaluation.

Assignment 5: GPU-Based Sensor Data Processing using CUDA

o Students prompt an AI tool: Implement parallel processing of sensor data using GPU acceleration.

o Students must:

Analyze performance improvements over CPU-based processing

Identify inefficiencies in parallel execution

Validate results using CUDA-based tools or benchmarking

o Assessment focuses on parallel computing concepts, GPU efficiency, and performance evaluation

Mini Project Component

Project 1 - Smart Health Monitoring Sensor System

Problem: Develop an IoT-based intelligent sensor system to monitor patient health parameters in real time.

Concepts Used: Smart sensors, IoT communication, real-time monitoring, anomaly detection.

Industrial Relevance: Healthcare systems, wearable devices, remote patient monitoring.

Project 2 - Intelligent Environmental Monitoring System

Problem: Design a sensor-based system for monitoring environmental conditions and air quality.

Concepts Used: Sensor calibration, data acquisition, wireless sensor networks, sensor processing.

Industrial Relevance: Smart cities, environmental safety, pollution monitoring.

Project 3 - Industrial Predictive Maintenance System

Problem: Predict machine failures using intelligent sensor systems and real-time monitoring.

Concepts Used: Sensor preprocessing, machine learning, anomaly detection, predictive systems.

Industrial Relevance: Manufacturing industries, Industry 4.0, industrial automation.

Project 4 - Intelligent Traffic Monitoring System

Problem: Develop a smart traffic monitoring system using IoT sensors and intelligent processing.

Concepts Used: Wireless sensor networks, sensor fusion, real-time processing, predictive monitoring.

Industrial Relevance: Smart transportation, traffic management, urban analytics.

Project 5 - GPU-Accelerated Sensor Processing System

Problem: Implement high-speed sensor data processing using GPU acceleration.

Concepts Used: CUDA programming, edge AI, parallel processing, GPU optimization.

Industrial Relevance: Real-time AI systems, autonomous systems, high-performance computing.



Database Management System

Course Code: 26DS2305

Semester: III

Programs Offered: B. Tech CSE (Data Science)

Course Type: Core

Credits: 4

L – T – P – J: 3 – 0 – 2 – 0

Prerequisites: Nil

Course Vision

To provide students with knowledge of database management systems in modern software development. To enable the design and implementation of efficient, secure, and scalable solutions for data storage, retrieval, and manipulation.

Course Objectives

1. This course aims to provide students with a strong foundation in the concepts, architecture, and design principles of database management systems.
2. It helps learners understand how relational databases are applied in data storage, query processing, transaction management, and application development.
3. The course enables students to develop and implement database applications using SQL, PL/SQL, and database connectivity with programming languages.
4. It also focuses on evaluating database designs in terms of normalization, functional dependencies, transaction isolation levels, and concurrency control.
5. Finally, the course encourages innovation in creating optimal, secure, and scalable database solutions for real-world applications including healthcare, banking, e-commerce, and enterprise systems.

Course Outcomes (CO)

CO	Description	Blooms Level
C01	Illustrate the fundamental concepts, architecture, and data models of database management systems.	L2
C02	Identify the applications of relational databases, SQL queries, and transaction management in real-world computing problems such as banking systems, healthcare records, and e-commerce platforms.	L2
C03	Develop database schemas using ER modelling, relational mapping, and normalization techniques.	L6
C04	Evaluate database designs with respect to normalization	L5
C05	Apply innovative database solutions to address real-world challenges in healthcare data management, NoSQL databases, big data analytics, and distributed systems.	L3



Broad Course Description

This course provides a comprehensive introduction to Database Management Systems (DBMS) and their critical role in managing structured data across various domains such as banking, healthcare, enterprise systems, and e-commerce. It explains the fundamental concepts of database systems, including data models, database architecture, schemas, and relational database design. The course explores how databases enable efficient storage, retrieval, and management of large volumes of data while ensuring consistency, integrity, and security. It also examines their role in real-world applications such as healthcare information systems, financial transaction systems, and enterprise resource planning platforms.

Students will learn how to design, implement, and query relational databases using Structured Query Language (SQL), including data definition, data manipulation, join operations, nested queries, and transaction control. The course covers essential database design techniques such as Entity-Relationship (ER) modelling, normalization, and functional dependency analysis to create efficient and reliable database schemas. It also introduces transaction management concepts including ACID properties, concurrency control, deadlock handling, and recovery mechanisms to ensure data reliability in multi-user environments. Additionally, the course provides an overview of modern NoSQL databases, including document-based, key-value, column-based, and graph databases, highlighting their importance in handling big data and real-time applications. Emphasis is placed on practical implementation, performance optimization, and secure data management practices for real-world engineering applications.

CO-PO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
C01	2	1									1	2	
C02	2	3			2					1	1	3	1
C03	2	3	3	2	2					1	1	3	2
C04	2	3	3	3	2					2	2	3	2
C05	2	3	3	3	3	1		1		2	2	3	3
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module-1: RELATIONAL DATABASES

(09 Hours)

Purpose of Database System, Views of data, Data Models, Database System Architecture, Database System Applications, Introduction to relational databases, Structure of Relational Databases, Database Schema, Keys, Schema Diagrams.

Module-2: RELATIONAL QUERY LANGUAGE

(09 Hours)

Overview of the SQL Query Language, SQL Data Definition, Basic Structure of SQL Queries, Additional Basic Operations, Aggregate Functions, Nested Subqueries, Join Expressions, Views,



Transactions, SQL Data Types and Schemas, Authorization, Accessing SQL from a Programming Language, Functions and Procedures, Triggers.

Module-3: DATABASE DESIGN

(09 Hours)

Entity-Relationship model, E-R Diagrams, Complex Attribute, Mapping Cardinalities, ER-to-Relational Mapping, Features of Good Relational Designs, Decomposition Using Functional Dependencies, Non-loss Decomposition, First, Second, Third Normal Forms, Dependency Preservation, Boyce/Codd Normal Form, Multi-valued Dependencies and Fourth Normal Form, Join Dependencies and Fifth Normal Form.

Module-4: TRANSACTION MANAGEMENT

(09 Hours)

Transaction Concepts, ACID Properties, A simple Transaction Model, Transaction Atomicity and Durability, Transaction Isolation, Schedules, Serializability, Transactions as SQL Statements, Concurrency control, Lock Based Protocols, Deadlock Handling, Recovery System, Failure Classification, Recovery Algorithm, Buffer Management, ARIES.

Module-5: NOSQL DATABASES

(09 Hours)

NOSQL Databases: Introduction, CAP Theorem, Document-Based NOSQL Systems and MongoDB, CRUD Operations, NOSQL Key-Value Stores, Column-Based or Wide Column NOSQL Systems, NOSQL Graph Databases and Neo4j.

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module 1: This module reflects authentic intelligence by helping students understand the fundamental concepts of relational databases (schemas, keys, architecture) and their meaningful application in solving real-world engineering problems such as banking systems, healthcare records, and enterprise resource planning.

Module 2: This module reflects authentic intelligence by emphasizing logical query design, systematic SQL writing, efficient data retrieval through joins and aggregations, and practical database programming for real-world applications like inventory management, data analytics, and customer relationship management.

Module 3: This module reflects authentic intelligence by focusing on accurate entity-relationship modelling, precise functional dependency identification, systematic normalization (1NF to 5NF), and reliable database design for practical needs including healthcare databases, e-commerce platforms, and university management systems.

Module 4: This module reflects authentic intelligence by encouraging the design of robust transaction management systems, ACID compliance, effective concurrency control protocols, deadlock handling mechanisms, and reliable recovery algorithms for real-world banking transactions, online booking systems, and payment gateways.

Module 5: This module reflects authentic intelligence by promoting the development of scalable, flexible, and high-performance database solutions using NoSQL technologies, appropriate CAP



theorem trade-offs, and optimal data models (document, key-value, column, graph) for real-world big data, IoT, social networks, and recommendation systems.

Authentic Intelligence Reflection:

Module 1: This module reflects authentic intelligence by helping students understand the fundamental concepts of data structures (arrays, pointers, structures) and their meaningful application in solving real-world engineering problems such as inventory management, image processing, and simulation state representation.

Module 2: This module reflects authentic intelligence by emphasizing logical thinking, systematic problem-solving through stack-based expression evaluation, fair resource allocation using queues, and efficient computational workflows for real-world applications like CPU scheduling and undo/redo operations.

Module 3: This module reflects authentic intelligence by focusing on accurate pointer manipulation, reliable dynamic memory usage, and efficient data organization in linked structures for practical applications such as browser history management, polynomial manipulation, and undo/redo functionality.

Module 4: This module reflects authentic intelligence by encouraging the design of hierarchical data representation, optimal search operations, and practical tree-based solutions for real-world needs including database indexing, expression parsing, AI decision making (game trees), and network routing tables.

Module 5: This module reflects authentic intelligence by promoting the development of mathematically sound and optimally efficient algorithms for network planning (MST), GPS navigation (shortest path), data compression (Huffman coding), and real-world infrastructure optimization problems.

List of Experiments

No.	Experiment	Module Link
1	Database Table Creation and Manipulation	Module I
2	Foreign Keys and Referential Integrity	Module I
3	Querying with Conditions and Aggregates	Module II
4	Subqueries and Basic Joins	Module II
5	Advanced Join Operations	Module II
6	User Defined Functions and Stored Procedures	Module II
7	Transactions and Control Commands	Module IV
8	SQL Triggers	Module II
9	Views	Module II
10	NoSQL Database Operations (MongoDB)	Module V



Assessment Scheme based on the course

Criteria	Weightage of Marks
Lab Reports (experiments)	10
Class Tests /Assignments	10
Mid-Semester Examination	30
Hackathon/Technical Competition/ Paper presentation/Global/DLI certifications	10
End-Semester Examination	40
Total	100 marks

Recommended Tools

1. Database Management Systems: MySQL, PostgreSQL, Oracle, Microsoft SQL Server, MongoDB, Neo4j
2. Query Tools: MySQL Workbench, pgAdmin, SQL Server Management Studio (SSMS), MongoDB Compass
3. Programming Languages: SQL, PL/SQL, T-SQL, Python, Java, C/C++
4. IDEs & Editors: VS Code, IntelliJ IDEA, Eclipse, DataGrip, DBeaver
5. Visualization: ERDPlus, Lucidchart, Draw.io, dbdiagram.io, Tableau, Power BI
6. AI Tools: ChatGPT, GitHub Copilot, Cursor AI, Claude AI, Google Bard, AI SQL Assistants
7. NoSQL Tools: MongoDB Atlas, Neo4j Browser, Cassandra Query Language (CQL), Redis CLI

Primary Textbooks

1. Abraham Silberschatz, Henry F. Korth, S. Sudharshan, "Database System Concepts", Seventh Edition, McGraw Hill, 2020.
2. RamezElmasri, Shamkant B. Navathe, "Fundamentals of Database Systems", Seventh Edition, Pearson Education, 2017

Additional Resources

1. NPTEL: Database Management Systems – IIT Madras / IIT Kharagpur
2. Coursera: Database Systems Specialization by University of Michigan
3. GeeksforGeeks: DBMS Tutorials and SQL Library
4. W3Schools: SQL Tutorial and Database Concepts
5. IEEE Transactions on Knowledge and Data Engineering
6. ACM Transactions on Database Systems (TODS)

Project Component

Industry-Driven Projects:

Project 1 – AI-Based Material Selection for Aerospace Components

- Problem: Select optimal material for an aircraft wing based on strength-to-weight ratio



- Concepts Used: Stress–strain, material properties, composites
- Industrial Relevance: Aerospace design (Boeing, Airbus), lightweight structures

Project 2 – Automated Hardness Measurement using Computer Vision

- Problem: Detect indentation in hardness testing images and calculate hardness
- Concepts Used: Vickers/Brinell hardness, image processing
- Industrial Relevance: Quality control in manufacturing and metallurgy

Project 3 – AI-Assisted XRD Pattern Classification

- Problem: Identify crystal structure from XRD data
- Concepts Used: Bragg's law, Miller indices, crystallography
- Industrial Relevance: Materials characterization in semiconductor and metallurgy industries

Project 4 – Simulation of Damped Oscillation in Mechanical Systems

- Problem: Model vibration in an aircraft landing gear system
- Concepts Used: SHM, damping, resonance
- Industrial Relevance: Automotive suspension, aerospace vibration control



Sustainability Engineering and Environmental, Social and Governance (CSE Cluster)

Course Code: 26DS2306

Semester: 3

Programs Offered: B. Tech. (Common to all CSE Branches)

Course Type: Core

Credits: 2

L – T – P – J: 2 – 0 – 0 – 0

Prerequisites: Fundamental understanding of engineering mathematics, Basic awareness of environmental science

Course Vision

This course aims to develop a strong foundation in sustainable engineering, Environmental, Social, and Governance (ESG) principles, and digital sustainability systems, enabling students to Analyze and address real-world challenges related to energy efficiency, environmental impact, ethical computing, and sustainable digital infrastructure. The course integrates modern analytical tools, AI-based decision-making, and data-driven sustainability assessment methods to enhance problem-solving and critical thinking.

Course Objectives

Students will:

1. Understand the fundamental concepts of sustainable engineering, ESG framework, and environmental, social, and governance dimensions of digital and computing systems, including energy usage and environmental impact.
2. Apply sustainability principles, ESG indicators, and basic analytical models to assess environmental performance and resource efficiency in engineering and digital systems.
3. Analyze energy consumption patterns, system efficiency, and environmental impacts in computing infrastructures such as data centers, cloud systems, and software applications.
4. Understand sustainability metrics, lifecycle assessment concepts, and ethical considerations in engineering systems, including responsible use of digital technologies.
5. Evaluate the performance of engineering and digital systems in terms of energy efficiency, sustainability impact, and alignment with ESG goals, using modern analytical and AI-assisted tools.



Course Outcomes (CO)

CO	Description	Blooms Level
CO1	Explain fundamental concepts of sustainability engineering, ESG framework, and Analyze environmental impact factors such as energy consumption and resource usage in digital systems.	L2
CO2	Apply sustainability principles, ESG indicators, and analytical models to evaluate environmental performance and efficiency in computing and engineering systems.	L3
CO3	Evaluate energy usage patterns, system efficiency, and resource optimization in IT infrastructure such as cloud systems, data centers, and software applications.	L5
CO4	Analyze sustainability metrics and lifecycle impacts of engineering systems using ESG reporting frameworks, carbon footprint estimation, and responsible design principles.	L3
CO5	Apply sustainability and ESG principles to assess the performance of digital systems in terms of energy efficiency, environmental impact, and ethical governance using AI- assisted tools.	L3

Broad Course Description

Sustainable Engineering and Environmental Governance is an integrated course that introduces the fundamental principles of sustainability science, environmental impact assessment, and ESG (Environmental, Social, and Governance) frameworks to Analyze and design responsible digital and engineering systems.

The course begins with an understanding of sustainability concepts, ESG principles, and global sustainability goals (SDGs), followed by the evaluation of environmental impacts of computing and engineering systems such as energy consumption, carbon footprint, and resource utilization. Key topics include green computing, lifecycle assessment (LCA), energy efficiency in IT infrastructure, and sustainable system design, with emphasis on applications in cloud computing, data centers, software systems, and smart infrastructure. The course also explores social and ethical dimensions of technology, including digital inclusivity, data privacy, ethical AI, and algorithmic fairness.

The governance aspect introduces ESG reporting frameworks, sustainability metrics, and compliance standards, enabling students to assess and communicate sustainability performance effectively. AI-based tools are integrated throughout the course for sustainability analytics, carbon estimation, ESG reporting, and energy optimization, enhancing data-driven decision-making.

The course combines theoretical concepts with case studies, AI-assisted analysis, and project-based learning, enabling students to develop analytical skills and responsible engineering



judgment required to design and evaluate sustainable digital systems in industries such as IT services, cloud computing, smart cities, and emerging green technologies.

CO-PO mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
C01	3	2	1	1	1							1	
C02	2	3	2	2	2							2	
C03	2	3	2	3	3							2	
C04	2	3	2	3	2							1	
C05	2	2	2	2	3							2	
3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)													

Course Modules

Module 1: Sustainability Fundamentals & ESG (6 Hours)

Topics: Sustainable development, Triple Bottom Line, ESG framework, SDGs, ESG vs CSR, sustainability challenges in digital economy.

Engineering Applications: Green computing, sustainable IT infrastructure, energy-efficient data centers.

Module 2: Environmental Sustainability in Digital Systems (6 Hours)

Topics: Carbon footprint of IT systems, energy consumption in data centers, lifecycle assessment (LCA), e-waste management, green cloud computing.

Engineering Applications: Energy-efficient servers, cloud optimization, sustainable software engineering.

Module 3: Social Sustainability & Ethical Computing (6 Hours)

Topics: Stakeholder analysis, digital inclusivity, data privacy, ethical AI, cybersecurity and society, social impact of automation.

Engineering Applications: AI ethics in recommendation systems, fairness in algorithms, inclusive software design.

Module 4: Governance, ESG Reporting, Sustainable Design & Future Technologies(6 Hours)

Topics: Corporate governance, ESG reporting frameworks (GRI, TCFD), ESG metrics, sustainability KPIs, data visualization, Circular economy, sustainable software design, green AI,



smart cities, Industry 4.0 sustainability, digital twins for sustainability.

Engineering Applications: Corporate ESG dashboards, compliance analytics, risk management systems, Smart grids, intelligent transportation, sustainable IoT systems.

Module 5: Energy Efficiency, Audit & Smart Energy Systems

(6 Hours)

Topics: Energy efficiency metrics, Energy losses in computing and electronic systems, Energy-efficient computing: green computing principles, low-power design, Energy Audit: types (preliminary and detailed audit), steps involved, Energy audit of IT infrastructure, Introduction to smart energy management systems.

Engineering Applications: Energy-efficient data centers and cloud computing systems, Power optimization in embedded systems and IoT devices, Energy audit of IT infrastructure in industries and campuses.

Strength of CO Mapping to PO/PSOs

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module 1 — Sustainability Fundamentals & ESG

- Use AI tools for sustainability trend analysis and ESG scoring based on real-world datasets.
- AI-based visualization of SDGs, ESG indicators, and sustainability performance metrics for organizations.

Module 2 — Environmental Sustainability in Digital Systems

- AI-assisted estimation of carbon footprint and energy consumption in IT systems and data centers.
- Machine learning models for energy prediction, workload optimization, and green cloud computing strategies.

Module 3 — Social Sustainability & Ethical Computing

- AI tools for sentiment analysis and stakeholder impact assessment.
- AI-based detection of bias, fairness issues, and ethical concerns in algorithms and digital systems.

Module 4 — Governance, ESG Reporting, Sustainable Design & Future Technologies

- AI-assisted ESG report analysis, KPI tracking, and sustainability dashboards (Power BI/Python tools).
- AI-based optimization of smart cities, digital twins, Industry 4.0 systems, and sustainable IoT infrastructures.

Module 5 — Energy Efficiency, Audit & Smart Energy Systems

- AI models for energy consumption prediction and anomaly detection in IT infrastructure.
- AI-based optimization of data center energy usage, smart energy monitoring using IoT, and identification of energy loss points.



Assessment Scheme

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT) every 2 weeks (Minimum 06 assessments)	20
Mid Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	10
Semester End Examination	40
Total	100 marks

Recommended Tools

Simulation & Analysis Tools

Energy modeling and system analysis using MATLAB / Python (NumPy, Pandas, Scikit-learn), Lifecycle Assessment (LCA) tools such as OpenLCA / SimaPro.

Visualization & Dashboard Tools

Data visualization using Power BI, Tableau, Python (Matplotlib, Seaborn), ESG dashboards and sustainability reporting tools, Smart city and IoT system visualization platforms.

AI & Digital Tools

AI tools such as ChatGPT for sustainability analysis, report generation, and decision support, Machine learning platforms (TensorFlow, Scikit-learn) for energy prediction and ESG analytics.

Primary Textbooks

1. Sustainable Computing and the Social Sciences – Lorenz M. Hilty, Bernard Aebischer, 1st Edition, 2015
2. Green IT: Reduce Your Information System's Environmental Impact – Jean-Marc Pierson, 1st Edition, 2012
3. Sustainability Engineering: Concepts, Design and Case Studies – Allen, D. T. & Shonnard, D. R., 1st Edition, 2001.

Reference book

1. Environmental Impact Assessment: Theory and Practice – Glasson, J., Therivel, R., Chadwick, A., 5th Edition, 2012
2. Designing for Sustainability: A Guide to Building Greener Digital Products and Services – Tim Frick, 1st Edition, 2014.



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Additional Sources

1. NPTEL – Courses on Sustainable Engineering, Energy Systems, Environmental Studies
2. MIT OpenCourseWare – Courses on Sustainability, Energy, and Smart Systems
3. IEEE, Elsevier, and industry ESG reports for latest research and case studies

DLI Course Mapped

Accelerating End-to-End Data Science Workflows

https://learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+S-DS-01+V1&utm



Differential Equations & Optimization

Course Code: 26DS2401

SEMESTER – IV

Programs Offered: B. Tech. (Common to all CSE Branches)

Course Type: Core

Credits: 03

L–T–P–J: 3–0–0–0

Prerequisites: Nil

Course Vision

To develop strong mathematical and computational foundations in vector calculus, numerical methods and optimization techniques for solving complex engineering problems and designing intelligent AI-driven optimization systems.

Course Objectives

1. Understand and apply vector calculus concepts for modelling engineering and scientific systems.
2. Apply numerical methods for solving ordinary and partial differential equations arising in dynamic systems.
3. Analyze and solve classical optimization problems encountered in engineering applications.
4. Develop problem-solving skills for AI and computational systems.
5. Apply nonlinear optimization algorithms to solve complex real-world optimization problems.

Course Outcomes (CO)

CO	Description	Blooms Level
C01	Apply vector calculus concepts such as gradient, divergence, curl, and vector integrals to solve engineering and computational problems.	L3
C02	Implement numerical methods for solving ordinary and partial differential equations arising in engineering and scientific models.	L3
C03	Formulate and solve classical optimization problems including single and multivariable optimization with constraints.	L6
C04	Develop and solve linear programming models for real-world problems.	L6
C05	Apply nonlinear optimization techniques such as gradient descent, Fletcher–Reeves method, and penalty methods to engineering applications.	L3



CO-PO mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
CO1	2	2			2						1	2	
CO2	2	3			3					1	1	3	1
CO3	2	3	3	2	3					1	1	3	2
CO4	2	3	3	2	2					1	1	3	2
CO5	2	3	2	2	3			1		1	2	2	2
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

MODULE – I Vector Calculus

(9 Hours)

Topics: Introduction to vector calculus, scalar and vector valued functions, gradient, tangent plane, directional derivative, divergence and curl, scalar and vector potentials. Statement of vector identities and simple problems. Line, surface and volume integrals. Statement of Green's theorem, Stokes' theorem and Gauss divergence theorem.

Applications: Fluid flow analysis, Electromagnetic field modelling, Robotics motion planning, Computer graphics and physical simulations.

MODULE – II Numerical Methods for ODE and PDE

(9 Hours)

Topics: Existence and uniqueness of solutions for ordinary differential equations. Solving initial value problems using Taylor series method, Euler's method and modified Euler method, Runge-Kutta methods, multistep methods. Higher-order equations and systems. Stability analysis. Solving boundary value problems using shooting methods, difference methods and variational methods. Introduction to numerical solutions for partial differential equations.

Applications: Weather prediction models, Mechanical system simulations, Control system modelling, Heat transfer and diffusion problems.

MODULE – III Introduction to Optimization (Classical Optimization)

(9 Hours)

Topics: Introduction to classical optimization, engineering applications of optimization, classification of optimization problems, single variable optimization, multivariable optimization without constraints, multivariable optimization with equality and inequality constraints, Kuhn-Tucker conditions.

Applications: Resource allocation problems, Engineering design optimization, Production planning, Logistics and supply chain optimization.

MODULE – IV Linear Programming

(9 Hours)

Topics: Linear programming, standard form of linear programming, geometry of linear programming problems, solution of a system of linear simultaneous equations, pivotal production of general systems of equations, graphical method, simplex algorithms, revised simplex methods, duality in linear programming.



Applications: Resource allocation and production planning, Transportation and logistics optimization, Supply chain management, Financial portfolio optimization, Scheduling and assignment problems.

MODULE – V Nonlinear Optimization

(9 Hours)

Topics: Unconstrained nonlinear optimization, univariate method, gradient of a function, gradient descent method, Fletcher-Reeves method. Constrained nonlinear optimization, characteristics of a constrained optimization problem, cutting plane method, interior and exterior penalty function methods.

Applications: Neural network training, Portfolio optimization, Control system optimization, Engineering system design.

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

Authentic Intelligence:

1. Understanding limitations of mathematical approximations in AI systems
2. Evaluating numerical errors in computational models
3. Ethical considerations in automated optimization systems
4. Ethical decision-making in optimization problems
5. Understanding biases introduced by optimization algorithms

AI Integration:

1. Gradient calculations in machine learning optimization
2. Simulation of dynamic systems in robotics
3. Decision optimization in intelligent systems
4. Linear programming used in decision-making systems
5. Gradient descent optimization algorithms for training neural networks and deep learning

Assessment Scheme based on the course

Criteria	Weightage of Marks
Continuous Internal Assessment (CIA) – Formative	60
Semester End Examination (SEE) – Summative	40
Total	100 marks

TEXT BOOKS

1. Hamdy A. Taha, Operations Research: An Introduction, 2017, 10th Edition, Pearson Education, Inc.
2. Rao, S.S., Engineering optimization: theory and practice, 2019, 5th Edition, John Wiley & Sons, Inc.



REFERENCE BOOKS

1. Arora, R.K., Optimization: algorithms and applications, 2015, 1st Edition, Chapman and Hall/CRC.
2. Deb, K., Optimization for engineering design: Algorithms and examples, 2012, 2nd Edition, PHI Learning Pvt. Ltd.
3. Nocedal, Jorge, and Stephen J. Wright. Numerical optimization. New York, NY: Springer New York, 2006.

E-RESOURCES

Refer to course portal for updated links to NPTEL, MIT OCW, and relevant open-access resources.

RECOMMENDED TOOLS

- Python
- MATLAB
- NumPy
- Pandas
- Matplotlib
- Scikit-learn

Lab Exercises

S.No	Exercise Title	Aim / Objective	Tools	Hrs	CO
MODULE I: Exercises 1–3: Vector Calculus (03 Hours)					
1	Gradient Computation	Introduction to computational tools and compute gradient of scalar functions	MATLAB/Python	1 h	CO1
2	Tangent Plane & Directional Derivative	Compute divergence and curl of vector fields, Visualize tangent planes and directional derivatives	MATLAB/Python	1 h	CO1
3	Line Integrals	Implement numerical computation of line integrals	MATLAB/Python	1 h	CO1
MODULE II: Exercises 4–6: Numerical Methods for ODE & PDE (03 Hours)					
4	Taylor Series and Euler Method	Solve ODE using Taylor series method, Euler method and modified Euler method	MATLAB/Python	1 h	CO2
5	Runge-Kutta and Multistep Method	Implement fourth-order Runge-Kutta method and multistep method	MATLAB/Python	1 h	CO2



6	Shooting and Finite Difference Method	Solve boundary value problems using shooting and finite difference method	MATLAB/Python	1 h	CO2
MODULE III: Exercises 7–9: Introduction to Optimization (03 Hours)					
7	Single Variable Optimization	Implement optimization for single variable functions	MATLAB/Python	1 h	CO3
8	Multivariable Optimization	Perform multivariable optimization and their visualization	MATLAB/Python	1 h	CO3
9	Kuhn-Tucker Conditions	Implement KKT conditions for constrained optimization	MATLAB/Python	1 h	CO3
MODULE IV: Exercises 10–12: Linear Programming (03 Hours)					
10	Formulation of LPP	Formulate and solve linear programming problems (LPP) and implement graphical method	MATLAB/Python	1 h	CO4
11	Simplex Method, Revised Simplex Method	Implement simplex algorithm for LPP, Solve LPP using revised simplex method	MATLAB/Python	1 h	CO4
12	Duality in LPP	Analyze duality concepts in linear programming	MATLAB/Python	1 h	CO4
MODULE V: Exercises 13–15: Nonlinear Optimization (03 Hours)					
13	Fletcher-Reeves Method	Implement Fletcher-Reeves optimization method	MATLAB/Python	1 h	CO5
14	Cutting Plane Method	Apply cutting plane method for constrained optimization	MATLAB/Python	1 h	CO5
15	Interior and Exterior Penalty Method	Implement interior and exterior penalty function method	MATLAB/Python	1 h	CO5



Introduction to Data Science

Course Code: 26DS2402

Semester: IV

Program(s): All CSE Cluster Programs

Course Type: Core

Credits:2

L – T – P – J: 2 – 0 – 0 – 0

Prerequisite(s): Python, Probability and Statistics

Course Vision

To provide foundational knowledge of data science concepts, statistical analysis, data preprocessing, and visualization techniques for solving real-world problems across multiple engineering domains.

Course Objectives

Students will be able to:

1. Understand fundamental data science concepts and workflows
2. Apply statistical techniques and visualization for data analysis
3. Perform data cleaning and preprocessing
4. Use feature selection and dimensionality reduction methods
5. Gain exposure to scalable data analytics techniques

Course Outcomes (CO)

CO	Description	Blooms Level
C01	Explain basic data science concepts and data preprocessing techniques	L2
C02	Apply statistical methods and visualization techniques for data analysis	L3
C03	Perform data cleaning and feature selection for efficient data preparation	L3
C04	Apply data transformation and dimensionality reduction techniques	L3
C05	Understand scalable data analytics concepts using modern tools	L2

Broad Course Description

Introduction to Data Science is a 2-credit core course offered in the 4th Semester across all CSE Cluster Programs. It introduces students to the foundational concepts and workflows of data science, including data collection, preprocessing, statistical analysis, and visualization. The course equips students with practical skills in Python-based data science libraries and prepares them for applying data-driven techniques to real-world engineering problems.

Students explore topics such as descriptive statistics, data visualization, data cleaning and feature selection, dimensionality reduction, and accelerated data analytics using GPU-based tools (NVIDIA RAPIDS). Through hands-on programming exercises and a case study on accelerated



data processing, the course develops both conceptual understanding and practical competence in handling, analyzing, and interpreting datasets for informed decision-making.

CO-PO mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
CO1	2				2						1	3	
CO2	2	2			3						1	3	2
CO3	2	3			3						1	3	2
CO4	2	2			3						1	3	3
CO5	2	1			3						1	2	3
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module 1: Basic Concepts And Python Packages

(6 Hours)

Topics: Predictive Modelling, Data Preparation, Importance of Data Preparation, Data Cleaning, Feature Selection, Data Transformation, Python Packages: NumPy, Matplotlib, Pandas, SciPy, Scikit-learn, DataFrames, Loading Machine Learning Data

Module 2: Descriptive Statistics and Data Visualization

(6 Hours)

Topics: Sampling, Data Analysis Process, Mean, Median, Standard Deviation, Skewness, Kurtosis, Data Visualization Techniques: Box Plot, Scatter Plot, Histogram, Bar Chart, Pie Chart, Heat Map, Correlation Analysis, Pivot Tables, ANOVA, Data Preprocessing, Rescaling, Standardization, Normalization, Binarization, Univariate and Bivariate Analysis, Recursive Feature Elimination, Principal Component Analysis

Module 3: Data Cleaning And Feature Selection

(6 Hours)

Topics: Handling Missing Data, Outlier Detection and Removal, Statistical Imputation, KNN Imputation, Iterative Imputation, Feature Selection Techniques, Methods for Categorical and Numerical Data, Feature Selection for Outputs, Recursive Feature Elimination, Importance of Feature Selection

Module 4: Data Transformation and Dimensionality Reduction

(6 Hours)

Topics: Scaling Techniques, Min-Max Scaling, Standard Scaling, Handling Outliers, Encoding Categorical Data, Transforming Data Distributions, Feature Engineering, PCA, LDA, SVD Techniques

Module 5: Accelerated Data Analytics (NVIDIA DLI)

(6 Hours)

Topics: CPU vs GPU in Data Processing, Need for Fast Data Analytics, Introduction to RAPIDS, GPU-Accelerated Data Workflows, Handling Large Datasets Efficiently, Basic Concept of



Accelerated DataFrame Operations, Overview of Machine Learning Acceleration, Identifying Processing Bottlenecks, Real-World Applications of Accelerated Data Analytics.

Case Study (NVIDIA DGX): Population-scale data analysis for epidemic control, Integration of multiple datasets, Real-time data analysis and iterative model updates, Decision-making using large-scale data insights

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module 1: Students use Python-based data science tools (NumPy, Pandas, Scikit-learn) to load, preprocess, and prepare real-world datasets for AI/ML pipelines. AI tools assist in identifying data quality issues and suggesting preprocessing strategies.

Module 2: AI tools support exploratory data analysis (EDA), automated visualization generation, and statistical pattern recognition in datasets.

Module 3: Students apply AI-assisted imputation tools and automated feature selection methods to clean and prepare datasets for downstream analysis.

Module 4: AI-based dimensionality reduction tools (PCA, LDA) are used to optimize feature spaces for downstream ML applications and improve model efficiency.

Module 5: Students explore GPU-accelerated AI workflows using NVIDIA RAPIDS and DGX systems, running large-scale data analytics pipelines for real-world applications.

Assessment Scheme based on the course

Criteria	Weightage of Marks
Mini Project	10
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination	30
End-Semester Examination	40
Total	100 marks



Recommended Tools

Primary Textbooks

1. Jason Brownlee, “Data Preparation for Machine Learning”, Machine Learning Mastery, 2020
2. Wes McKinney, Python for Data Analysis, O’Reilly Media, 3rd Edition, 2022
3. Jake VanderPlas, Python Data Science Handbook, O’Reilly Media, 2016
4. DLI Link: https://learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+T-DS-03+V1

Additional AI & Computational Resources

1. NPTEL: Introduction to Data Science – nptel.ac.in
2. Kaggle Learn (free data science micro-courses) – [kaggle.com/learn](https://www.kaggle.com/learn)
3. Google Colab Python Notebooks – colab.research.google.com
4. Scikit-learn Examples and Tutorials – scikit-learn.org
5. NVIDIA DLI Course: Accelerating Data Science Workflows with RAPIDS – learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+T-DS-03+V1

Project Component (Sample Projects)

Project 1 – End-to-End Data Analysis on a Real-World Dataset

Problem: Perform complete data analysis on a provided dataset (sales, health, or sensor data)

Concepts Used: Data preprocessing, EDA, visualization, feature selection

Industrial Relevance: Business analytics, healthcare informatics, IoT data processing

Project 2 – Customer Churn Prediction using Feature Engineering

Problem: Identify key features predicting customer churn from telecom data

Concepts Used: Feature selection, encoding, dimensionality reduction, PCA

Industrial Relevance: CRM systems, marketing analytics, customer retention strategies

Project 3 – Outlier Detection in Industrial Sensor Data

Problem: Detect anomalies in manufacturing sensor readings for quality control

Concepts Used: IQR method, z-score, KNN imputation, statistical analysis

Industrial Relevance: Predictive maintenance, smart manufacturing, fault detection

Project 4 – GPU-Accelerated Data Processing using RAPIDS

Problem: Demonstrate performance comparison of CPU (Pandas) vs GPU (cuDF) operations

Concepts Used: RAPIDS cuDF, dataframe operations, benchmarking

Industrial Relevance: High-performance computing, real-time analytics, big data

Project 5 – Epidemic Data Analysis and Visualization Dashboard

Problem: Analyze and visualize population-scale epidemic spread data

Concepts Used: Data merging, normalization, visualization, large-scale analytics



AI-Powered Assignments (Sample Assignments)

Assignment 1: AI-Assisted Data Preprocessing and Quality Evaluation

Students prompt an AI tool: Identify and fix data quality issues (missing values, duplicates, outliers) in this dataset and suggest appropriate preprocessing steps.

- Students must:
- Analyze whether the AI-recommended preprocessing steps are statistically justified
- Identify any incorrect imputations or inappropriate transformations suggested by the AI
- Improve the preprocessing pipeline using standard techniques (IQR, z-score, domain knowledge)
- Assessment focuses on data quality reasoning, preprocessing decisions, and critical evaluation of AI outputs

Assignment 2: AI-Assisted Statistical Analysis and Visualization

- Analyze whether the statistical measures and chart types chosen are appropriate for the data
- Identify incorrect chart choices, misleading visualizations, or statistical errors

Verify and correct using proper statistical formulas and standard visualization best practices.

Assessment focuses on statistical reasoning, visualization design, and interpretation skills

Assignment 3: AI-Based Feature Selection and Dimensionality Reduction

Students prompt an AI tool: Select the most important features from this dataset and reduce its dimensionality using PCA.

- Students must:
- Analyze whether the selected features are statistically significant and PCA is applied correctly
- Identify incorrect variance thresholds, dropped important features, or over-reduction
- Refine using correlation analysis, Recursive Feature Elimination (RFE), and correct PCA implementation

Assessment focuses on feature engineering, PCA interpretation, and model validation.

Assignment 4: AI-Based Accelerated Data Processing Analysis

Students prompt an AI tool: Write a Python script to compare Pandas (CPU) and RAPIDS cuDF (GPU) performance on a large dataset.

- Students must:
- Analyze whether the benchmarking methodology and performance comparison are valid
- Identify incorrect timing methods, unfair comparisons, or hardware constraints overlooked
- Improve the benchmarking script using Python time module and correct cuDF API usage

Assessment focuses on GPU computing concepts, performance benchmarking, and critical evaluation of AI-generated code.



Design and Analysis of Algorithms

Course Code: 26DS2403

Semester: IV

Programs Offered: B.Tech CSE (Data Science)

Course Type: Core

Credits: 4

L – T – P – J: 3 – 0 – 2 – 0

Prerequisites: Basic programming knowledge (C/C++/Python)

Course Vision

To develop strong analytical and problem-solving skills by equipping students with fundamental algorithm design techniques and complexity analysis methods for efficient computation.

Course Objectives

1. Introduce fundamental concepts of algorithm design and analysis.
2. Develop the ability to evaluate algorithm efficiency using time and space complexity.
3. Provide knowledge of classical algorithm design paradigms such as divide and conquer, greedy, and dynamic programming.
4. Enable students to apply graph algorithms and optimization techniques to real-world problems.
5. Introduce computational complexity theory and advanced problem-solving approaches.

Course Outcomes (CO)

CO	Description	Blooms Level
C01	Understand algorithm fundamentals and analyze time and space complexity.	L2
C02	Apply divide and conquer and greedy strategies to solve computational problems.	L3
C03	Design solutions using dynamic programming techniques.	L6
C04	Apply graph algorithms for real-world problem solving.	L3
C05	Analyze computational complexity and apply advanced techniques like backtracking and branch & bound.	L4

Broad Course Description

The Design and Analysis of Algorithms course focuses on developing efficient algorithms for solving computational problems. It introduces students to different algorithmic paradigms such as divide and conquer, greedy methods, and dynamic programming.

The course emphasizes analyzing algorithm efficiency using asymptotic notations and solving recurrence relations. It further explores graph algorithms, optimization problems, and computational complexity theory, including NP-completeness.

Students gain practical experience in designing scalable solutions, preparing them for advanced domains such as artificial intelligence, data science, and software engineering.



CO-PO Mapping

CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2
CO1	3	2			2	1							2	2
CO2	3	3			2	1					1	1	2	2
CO3	3	3	2		2	1		1	1		2	2	2	2
CO4	3	3	2		3	1		1	1		2	2	3	2
CO	3	3	2		3	1		2	2		2	2	3	2
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)														

Course Modules

Module 1: Algorithm Fundamentals and Complexity Analysis (9 Hours)

Introduction to algorithms, Characteristics of algorithms, Performance analysis, Time and space complexity, Asymptotic notations (Big-O, Theta, Omega), Best, average, worst case analysis, Recurrence relations, Solving recurrences (substitution, recursion tree, master theorem), Algorithm efficiency comparison

Module 2: Divide and Conquer & Greedy Algorithms (9 Hours)

Divide and Conquer strategy, Binary search, Merge sort, Quick sort, Strassen's matrix multiplication, Greedy method: Concept, Activity selection, Fractional knapsack, Huffman coding, Minimum spanning trees (Prim's and Kruskal's algorithms)

Module 3: Dynamic Programming (9 Hours)

Dynamic programming concept, Optimal substructure, Overlapping subproblems, Memoization and tabulation, 0/1 Knapsack problem, Longest Common Subsequence (LCS), Matrix chain multiplication, Shortest path algorithms (Bellman-Ford)

Module 4: Graph Algorithms (9 Hours)

Graph representations, Breadth First Search (BFS), Depth First Search (DFS), Topological sorting, Shortest path algorithms (Dijkstra's algorithm), Minimum spanning trees, Network flow basics

Module 5: Advanced Topics and Computational Complexity (9 Hours)

Backtracking: N-Queens, Subset sum, Branch and Bound: Traveling Salesman Problem (TSP), Complexity classes (P, NP, NP-complete, NP-hard), Polynomial-time reductions, Introduction to approximation algorithms

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module-1: Students use AI-assisted tools to analyze algorithm complexity and visualize



asymptotic growth. Tools like Python notebooks and simulators help understand recursion and performance analysis.

Module-2: AI-based visualization platforms assist in understanding divide-and-conquer and greedy strategies. Students use intelligent tools to compare sorting algorithms and optimize greedy solutions.

Module-3: Students utilize AI tools to implement dynamic programming solutions and visualize state transitions. Platforms help in understanding optimization problems like knapsack and LCS.

Module-4: Graph visualization tools powered by AI help simulate BFS, DFS, and shortest path algorithms. Students explore real-world applications such as network routing and recommendation systems.

Module-5: AI tools assist in solving complex optimization problems like TSP and N-Queens. Students explore computational complexity using interactive platforms and analyze NP-complete problems.

List of Experiments

No.	Experiment	Module Link
1	Analysis of time complexity of algorithms	Module 1
2	Implementation of Merge Sort and Quick Sort	Module 2
3	Greedy algorithm – Fractional Knapsack	Module 2
4	Dynamic Programming – 0/1 Knapsack	Module 3
5	Longest Common Subsequence implementation	Module 3
6	Graph traversal using BFS and DFS	Module 4
7	Dijkstra's shortest path algorithm	Module 4
8	Minimum Spanning Tree (Prim's/Kruskal's)	Module 4
9	Backtracking – N-Queens problem	Module 5
10	Branch and Bound – Traveling Salesman Problem	Module 5

Assessment Scheme

Criteria	Weightage of Marks
Lab Reports (experiments)	10
Class Tests / Assignments	10
Mid-Semester Examination	30
Hackathon / Technical Competition	10



End-Semester Examination	40
Total	100 Marks

Recommended Tools

- Programming Languages: C, C++, Python
- Platforms: HackerRank, LeetCode, CodeStudio
- Visualization Tools: VisuAlgo, Algorithm Visualizer

Primary Textbooks

1. Introduction to Algorithms authored by Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, 4th Edition, published in the year 2022 by MIT Press.
2. *Introduction to the Design and Analysis of Algorithms* authored by Anany Levitin, 3rd Edition, published in the year 2012 by Pearson Education
3. *Data Structures and Algorithms* authored by Alfred V. Aho, John E. Hopcroft, and Jeffrey D. Ullman, 1st Edition, published in the year 1983 by Addison-Wesley Publishing Company.

Additional Resources

1. NPTEL – Design and Analysis of Algorithms
2. MIT OpenCourseWare – Algorithms
3. GeeksforGeeks, TutorialsPoint

AI POWERED ASSIGNMENTS

Assignment 1: AI-Assisted Complexity Analysis of Algorithms

Students shall use AI tools to analyze the time and space complexity of algorithms such as Linear Search, Binary Search, Merge Sort, and Quick Sort. The assignment includes comparing best, average, and worst-case complexities and validating AI-generated asymptotic analysis manually.

Assignment 2: Intelligent Implementation of Divide and Conquer & Greedy Algorithms

Students shall implement Divide and Conquer algorithms (Merge Sort, Quick Sort) and Greedy algorithms (Activity Selection, Fractional Knapsack, Huffman Coding) using AI-assisted coding platforms. Students must compare execution efficiency and evaluate AI-generated optimizations.

Assignment 3: AI-Based Dynamic Programming Problem Solver

Students shall develop AI-assisted solutions for Dynamic Programming problems such as 0/1 Knapsack, Longest Common Subsequence, and Matrix Chain Multiplication. The assignment includes comparing memoization and tabulation approaches and analyzing performance improvements.

Assignment 4: AI-Powered Graph Algorithm Visualization and Analysis

Students shall use AI tools to simulate and visualize graph algorithms including BFS, DFS, Dijkstra's Algorithm, and Minimum Spanning Tree algorithms. Students should analyze traversal efficiency, shortest path computation, and network optimization.



Machine Learning

Course Code: 26DS2404

Semester: IV

Programs Offered: B. Tech. all branches

Course Type: Core

Credits: 4

L – T – P – J: 3 – 0 – 2 – 0

Prerequisites: Data Analysis and visualization

Course Vision

To develop the ability to design, analyze, and implement machine learning solutions for engineering problems by integrating data-driven modeling, computational tools, and responsible AI practices.

Course Objectives

- To summarize the basic concepts, Feature Engineering and different types of Machine Learning.
- To make Use of the Supervised Machine Learning techniques for solving appropriate real-world applications.
- To make Use of the Unsupervised Machine Learning techniques, Dimensionality Reduction techniques for solving appropriate real-world applications.
- To utilize the Probabilistic Graphical Models and Ensemble Learning Methods for handling complex real time problems.
- To evaluate the performance and reliability of machine learning models.

Course Outcomes (CO)

CO	Description	Blooms Level
C01	Describe the basic concepts and different types of Machine Learning techniques and to design Well Posed Learning System	L2
C02	Apply the learned concepts of the Supervised learning algorithms for regression and classification problems	L3
C03	Apply Unsupervised Machine Learning Algorithms such as Hard and Soft clustering and Dimensionality Reduction techniques	L3
C04	Make Use of Probabilistic Graphical Models and Ensemble Learning Methods	L3
C05	Evaluate performance and apply Optimization and Regularization Techniques	L5

Broad Course Description

This course provides a comprehensive introduction to machine learning, focusing on the design and implementation of data-driven models to solve real-world engineering problems. It covers core concepts such as supervised and unsupervised learning, feature engineering, and data preprocessing, along with widely used algorithms for classification, regression, and clustering.



The course also explores advanced topics including probabilistic graphical models, ensemble learning, and model optimization techniques. Emphasis is placed on model evaluation, validation, and the development of robust, reliable, and interpretable machine learning systems, while incorporating ethical considerations and responsible AI practices.

CO-PO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
C01	2	1			1						1	2	1
C02	2	3			2						1	3	2
C03	2	3			3						1	3	2
C04	2	3	2	2	3						2	3	2
C05	2	3	2	3	3						2	3	2
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module 1: Foundations of Machine Learning (9 Hours)

Well-posed learning problems, Designing a Learning system, Introduction to Machine learning and types Of Learning: Supervised, Unsupervised, And Reinforcement Learning, Data Pre-processing: Missing Value Imputation, Handling balanced and Imbalanced Data, and Feature Engineering.

Applications: Classification and regression problems in real-world scenarios, predictive analytics in healthcare and finance, feature engineering for improving model performance.

Module 2: Supervised Machine Learning (9 Hours)

Introduction to Supervised Learning: Binary and Multi-Class Classification, Regression - Linear Regression and Nonlinear Regression, Logistic Regression, Naive Bayes Classification, Decision Trees (ID3, CART) and Random Forest, K-Nearest Neighborhood, Support Vector Machine (Linear and Non-linear Kernels).

Applications: house price prediction systems, medical diagnosis systems, fault detection in machines, financial risk prediction.

Module 3: Unsupervised Learning (9 Hours)

Introduction to Unsupervised Learning, Clustering (Hard and Soft Clustering), Hierarchical Clustering: K-Means, Fuzzy C-Means (FCM) Algorithm, Gaussian Mixture Models (GMM), Expectation Maximization Algorithm, Dimensionality Reduction - Linear Discriminant Analysis and Principal Component Analysis.

Applications: customer segmentation in business analytics, anomaly detection in network traffic, pattern discovery in industrial data.



Module 4: Graphical Models & Ensemble Learning

(9 Hours)

Probabilistic Graphical Models - Bayesian Networks, Markov Random Fields, and Ensemble Learning - Bagging Algorithms, Understanding Out-Of-Bag Score, Boosting: AdaBoost, Gradient Boosting Machines (GBM), XGBoost, CatBoost.

Applications: uncertainty modeling using Bayesian Networks, image analysis with Markov Random Fields, prediction tasks using ensemble methods, fraud detection and recommendation systems with boosting algorithms.

Module 5: Validation, Optimization & Regularization

(9 Hours)

Hyper parameter Tuning: Grid Search CV, Random Search CV, Machine Learning Model Validation - Confusion Matrix, Accuracy, Precision, Recall, F1 Score, Cost Function, Cross-Validation, Error Metrics (MSE, RMSE, MAE).

Machine Learning Optimization Algorithms: Gradient Descent Techniques, Bayesian Optimization, Overfitting, Underfitting, Optimal Fit, Bias, Variance, Regularization - L1 Regularization, L2 Regularization, Elastic Net Regularization.

Applications: model tuning for improved prediction accuracy, performance evaluation in classification tasks, optimization of machine learning models in healthcare, finance, and engineering applications.

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

Module 1: Introduction to Python-based machine learning workflow, data preprocessing and feature engineering techniques, usage of libraries like NumPy, Pandas, and Scikit-learn for building and evaluating models.

Authentic Intelligence: Understanding well-posed learning problems and their real-world constraints, challenges in designing efficient learning systems, limitations of deep learning models, risks associated with poor feature selection and sampling bias, ethical considerations and responsible deployment of AI systems.

Module 2: Implementation of supervised learning algorithms using Scikit-learn, training and evaluating models using real-world datasets. Authentic Intelligence: Interpretability of machine learning models, risks of incorrect predictions in automated decision-making systems.

Module 3: Implementing clustering and dimensionality reduction techniques using Scikit-learn, visualization of clusters using Python tools.

Authentic Intelligence: Risks of misinterpreting clusters, responsible interpretation of patterns discovered from data.

Module 4: Implementation of graphical models and ensemble techniques using Scikit-learn, use of bagging and boosting methods, model evaluation with Out-of-Bag score.

Authentic Intelligence: Limitations of graphical and ensemble models, overfitting risks in boosting, interpretability challenges, and importance of responsible model selection.

Module 5: Use of Grid Search and Random Search for hyperparameter tuning, implementation of validation techniques and performance metrics, application of optimization algorithms and regularization methods using Scikit-learn.



Authentic Intelligence: Understanding overfitting and underfitting, bias-variance tradeoff, limitations of optimization methods, and importance of selecting appropriate validation and regularization techniques.

List of Experiments

SL. No	Exercise Title	Aim / Objective	Module Link
1	Data Analysis and Visualization	Perform Exploratory Data Analysis and Visualize Features using the Iris Dataset	MODULE I
2	Handling Missing Values, Scaling Features	Preprocess Data by Handling Missing Values, Encoding Categorical Variables, and Scaling Features on Standard Dataset.	MODULE I
3	Split Dataset, cross validation	Split Dataset into Training and Testing Sets and Apply Cross-Validation.	MODULE I
4	Linear Regression	Implement Linear Regression to Predict House Prices using the Housing Price Dataset.	MODULE II
5	Logistic Regression	Build a Logistic Regression Model to Perform Classification	MODULE II
6	K-Nearest Neighbors (KNN)	Apply the K-Nearest Neighbors (KNN) Algorithm to Classify Flower Species using the Iris Dataset	MODULE II
7	Decision Tree Model	Train and Visualize a Decision Tree Model to Classify Wine Quality using the Wine Quality Dataset.	MODULE II
8	Random Forest Model	Develop a Random Forest Model to Predict Heart Disease using the Heart Disease Dataset	MODULE II
9	Linear and Non-linear Support Vector Machine	Implementation of Linear and Non-linear Support Vector Machine for Classification	MODULE II
10	K-Means Clustering	Apply K-Means Clustering to Perform Customer Segmentation using the Mall Customer Segmentation Dataset.	MODULE III



11	Principal Component Analysis	Apply Principal Component Analysis (PCA) to Reduce Feature Dimensions using the Wine Dataset.	MODULE III
12	Bagging, Boosting	Implement and compare ensemble techniques: Bagging (Random Forest) Boosting (AdaBoost / Gradient Boosting)	MODULE IV
13	Model evaluation	Evaluate Classification Models by Computing Accuracy, Precision, Recall, F1-Score, and Confusion Matrix using the Breast Cancer Dataset.	MODULE V
14	Hyperparameter Tuning	Perform Hyperparameter Tuning using Grid Search with Cross-Validation on the Digits Dataset	MODULE V
15	Machine Learning Model to Solve a Real-World Engineering	Design and Implement a Machine Learning Model to Solve a Real-World Engineering Problem using an Open Engineering Dataset (e.g., Energy Consumption, Traffic Flow, or Predictive Maintenance Dataset)	MODULE V

Assessment Scheme based on the course

Criteria	Weightage of Marks
Lab Reports (experiments)	10
Class Tests /assignments	10
Mid-Semester Examination	30
Project	10
End-Semester Examination	40
Total	100 marks

Recommended Tools

Primary Textbooks

1. Machine Learning – Tom M. Mitchell, McGraw-Hill, 1st Edition, 1997.
2. Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow – Aurélien Géron, O'Reilly Media, 3rd Edition, 2022.

REFERENCE BOOKS:

1. Pattern Recognition and Machine Learning – Christopher M. Bishop, Springer, 1st Edition, 2006.
2. Introduction to Machine Learning with Python – Andreas C. Müller & Sarah Guido, O'Reilly Media, 1st Edition, 2017.



Additional Resources

1. Accelerating End-to-End Data Science Workflows

https://learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+S-DS-01+V2

2. Best Practices in Feature Engineering for Tabular Data With GPU Acceleration

https://learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+S-DS-06+V1

3. Accelerating Clustering Algorithms to Achieve the Highest Performance

https://learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+S-DS-04+V1

4. Exploring Adversarial Machine Learning

https://learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+S-DS-03+V1

5. Accelerate Data Science Workflows with Zero Code Changes

https://learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+S-DS-03+V1

Recommended Tools

- Programming: Python (NumPy, Pandas, Scikit-learn, PyTorch) – Google Colab/ Jupyter Notebook/Visual Studio
- Visualization: Matplotlib, Seaborn, Plotly, UMAP
- Experiment Tracking: MLflow, Weights & Biases
- MLOps & Deployment: Docker, FastAPI, ONNX
- AI Assistants: GitHub Copilot, ChatGPT, Claude

Project Component

1. Ensemble Learning Project (Performance Comparison)

Compare bagging vs boosting vs single models

Tech: Random Forest, XGBoost, AdaBoost

2.. Predictive Maintenance System

Predict machine failure

Tech: Classification + anomaly detection

3. Healthcare Diagnosis Prediction System

Predict diseases using patient data

Tech: Ensemble + feature selection

4. Traffic Flow Prediction System

Predict congestion levels

Tech: Regression + optimization

5. Intrusion Detection System (Cybersecurity)

Detect malicious network activity

Tech: SVM, Random Forest, clustering



Operating Systems

Course Code: 26DS2405

Semester: IV

Programs: B. Tech - CSE (Data Science)

Course Type: Core

Credits: 3

L – T – P – J: 3 – 0 – 0 – 0

Prerequisite: Nil

Course Vision

To provide a strong foundation in operating system concepts such as process management, memory management, file systems, and virtualization, enabling students to design efficient and reliable computing systems. The course aims to bridge theory with real-world systems like Linux, Android, and iOS, and to develop problem-solving skills for modern computing environments and emerging technologies.

Course Objectives

1. Develop a fundamental understanding of operating system concepts, including system structure, services, and system calls, enabling students to analyze the functioning of modern computing systems.
2. Build proficiency in process management and synchronization techniques, allowing students to design efficient CPU scheduling algorithms and handle concurrency, deadlocks, and inter-process communication.
3. Equip students with knowledge of memory management techniques such as paging, segmentation, and virtual memory, enabling efficient resource allocation and performance optimization.
4. Provide a strong foundation in storage management and file systems, focusing on disk scheduling, file organization, access methods, and I/O system operations relevant to real-world computing environments.

Course Outcomes (CO)

CO	Description	Blooms Level
CO1	Illustrate the structure, functions, and services of operating systems and their interaction with computer hardware and users.	L2
CO2	Analyze process management techniques including scheduling, synchronization, inter-process communication, and deadlock handling.	L4
CO3	Apply memory management concepts such as paging, segmentation, virtual memory, and page replacement algorithms to optimize system performance.	L3
CO4	Illustrate storage management and file system concepts, including disk scheduling, file organization, and I/O system operations.	L2
CO5	Evaluate virtualization concepts and the working of modern operating systems such as Android and iOS.	L5



Broad Course Description

This course provides a comprehensive understanding of the fundamental concepts and design principles of operating systems. It covers key topics such as process management, CPU scheduling, synchronization, memory management, virtual memory, file systems, and storage management. The course also introduces virtualization concepts and explores modern operating systems like Linux, Android, and iOS. Emphasis is placed on analyzing system performance, efficient resource management, and understanding the role of operating systems in modern computing environments.

Course Modules

Module - 1: Introduction to Operating Systems

(7 Hours)

Computer system organization, Operating system overview, Objectives and functions, Evolution of operating systems, Operating system structures, OS services, User OS interface, System calls, System programs, OS design and implementation, Structuring methods.

Module - 2: Process Management and Synchronization

(11 Hours)

Process concept, Process scheduling, Operations on processes, Inter-process communication, CPU scheduling criteria and algorithms, Threads and multithreading models, Threading issues, Process synchronization, Critical-section problem, Synchronization hardware, Semaphores, Mutex, Classical synchronization problems, Monitors, Deadlocks: prevention, avoidance, detection, and recovery.

Module - 3: Memory Management

(10 Hours)

Main memory concepts, Swapping, Contiguous memory allocation, Paging, Page table structure, Segmentation, Segmentation with paging, Virtual memory, Demand paging, Copy-on-write, Page replacement algorithms, Frame allocation, Thrashing.

Module - 4: Storage and File Systems

(10 Hours)

Mass storage systems, Disk structure, Disk scheduling, File system interface, File concepts, Access methods, Directory structures, File sharing and protection, File system implementation, Allocation methods, Free space management, I/O systems, I/O hardware, Application I/O interface, Kernel I/O subsystem.

Module - 5: Virtual Machines and Mobile Operating Systems

(7 Hours)

Virtual machines: history, benefits, features, types, implementation, Virtualization concepts, OS components in virtualization, Mobile operating systems: iOS and Android.

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module 1: Introduction to Operating Systems

Students use AI tools such as ChatGPT and system simulators to understand OS architecture, system calls, and OS services. AI-assisted visualization helps in comparing different OS structures and interfaces, enabling better conceptual clarity.



Module 2: Process Management and Synchronization

Students apply AI-based simulations and programming tools like Python to model process scheduling, synchronization, and deadlock scenarios. AI tools help visualize CPU scheduling algorithms and predict system behaviour under different workloads.

Module 3: Memory Management

Students use simulation tools and AI-assisted coding environments to implement paging, segmentation, and page replacement algorithms. AI models assist in analyzing memory usage patterns and predicting optimal allocation strategies.

Module 4: Storage and File Systems

Students utilize AI-supported tools to simulate disk scheduling algorithms and file system operations. AI-based analysis helps evaluate performance metrics such as seek time, throughput, and storage efficiency.

Module 5: Virtual Machines and Mobile Operating Systems

Students explore virtualization using tools like VirtualBox and analyze mobile operating systems such as Android and iOS. AI tools assist in understanding resource allocation and system performance in virtual environments.

Authentic Intelligence Reflection:

Module-1

Students critically evaluate AI-generated explanations of OS structures and compare them with textbook and real system behaviour.

Module-2

Students analyze differences between AI-predicted scheduling outcomes and actual algorithm performance, identifying limitations of AI in concurrency handling.

Module-3

Students reflect on AI-suggested memory allocation strategies versus traditional methods, assessing efficiency and feasibility.

Module-4

Students evaluate AI-based disk scheduling results and compare them with standard algorithms to understand trade-offs in performance.

Module-5

Students assess AI-generated virtualization models and mobile OS insights against real-world system constraints such as scalability, security, and resource management.

Assessment Scheme based on the course

Criteria	Weightage of Marks
Lab Reports (experiments)/Project	10
Class Tests /assignments	10
Mid-Semester Examination	30
Hackathon/Technical Competition/ Paper presentation/Global/DLI certifications	10



End-Semester Examination	40
Total	100 marks

Recommended Tools

Primary Textbooks

1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, Operating System Concepts, 10th Edition, Wiley, 21018
2. Andrew S. Tanenbaum, Modern Operating Systems, 5th Edition, Pearson, 2022
3. William Stallings, Operating Systems: Internals and Design Principles, 7th Edition, 2018
4. Achyut S. Godbole, Atul Kahate, Operating Systems, McGraw Hill, 2016
5. Ramaz Elmasri, A. Gil Carrick, David Levine, “Operating Systems – A Spiral Approach”, Tata McGraw Hill Edition, 2010.

Additional Resources

Additional AI & Computational Resources

1. NPTEL – IIT-based courses on Operating Systems
2. Coursera – OS, Linux, and system programming courses
3. edX – University-level OS and virtualization courses
4. MIT OpenCourseWare – Advanced lectures on operating systems
5. Linux Documentation – Hands-on OS usage and commands
6. VirtualBox – Practice virtualization concepts
7. Docker – Learn OS-level virtualization and containers
8. GeeksforGeeks – Quick reference for OS concepts and problems
9. Stack Overflow – Practical problem-solving discussions
10. GitHub – Open-source OS projects and implementations

CO PO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
C01	3	2	1	1	2							2	3	2
C02	3	3	2	2	3				1			2	3	3
C03	3	3	3	2	3				1			2	3	3
C04	3	2	2	2	3				1			2	3	2
C05	3	3	2	2	3				2	1		3	3	3

3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)

Project Component

Industry-Driven Projects:

Project 1 – AI-Based CPU Scheduling Optimization

- Problem: Optimize CPU scheduling to minimize waiting time and maximize throughput•



Concepts Used: Scheduling algorithms (FCFS, SJF, Round Robin), process management, AI optimization• Industrial Relevance: Cloud computing platforms (Amazon Web Services, Microsoft Azure), server resource management

Project 2 – Intelligent Memory Management System

• Problem: Predict and optimize memory allocation to reduce page faults• Concepts Used: Paging, segmentation, virtual memory, page replacement algorithms• Industrial Relevance: Operating system design, high-performance computing, data centers (Intel, AMD)

Project 3 – Deadlock Detection and Recovery System

• Problem: Detect and resolve deadlocks in multi-process environments• Concepts Used: Resource allocation graph, deadlock prevention, avoidance, detection• Industrial Relevance: Database systems and enterprise applications (Oracle, IBM)

Project 4 – Disk Scheduling Optimization using AI

• Problem: Optimize disk scheduling algorithms to reduce seek time• Concepts Used: Disk scheduling (FCFS, SSTF, SCAN), I/O systems, performance metrics• Industrial Relevance: Storage systems and cloud infrastructure (Seagate Technology, Western Digital)

AI-Powered Assignments

Assignment 1: AI-Based CPU Scheduling Analysis

• Students prompt an AI tool: **Suggest the most efficient CPU scheduling algorithm for a system with mixed interactive and batch processes.**• Students must:

- Analyze the AI-recommended scheduling algorithm using criteria such as waiting time, turnaround time, and response time
- Identify any limitations or unrealistic assumptions (e.g., ignoring process priority or arrival time variability)
- Improve the solution by comparing with standard algorithms (FCFS, SJF, Round Robin) through simulation• Assessment focuses on understanding of scheduling concepts, performance evaluation, and validation of AI outputs

Assignment 2: AI-Assisted Deadlock Detection Strategy

• Students prompt an AI tool: **Propose a method to detect and prevent deadlocks in a multi-process system.**• Students must:

- Evaluate the correctness of AI-suggested methods using resource allocation graphs
- Identify missing factors such as resource preemption and system overhead
- Enhance the solution using Banker's Algorithm or deadlock detection techniques• Assessment focuses on problem analysis, correctness of deadlock handling, and critical evaluation of AI suggestions

Assignment 3: AI-Based Memory Management Optimization

• Students prompt an AI tool: **Recommend the best memory management technique to minimize page faults in a system.**• Students must:

- Analyze AI recommendations using paging, segmentation, and virtual memory concepts
- Identify unrealistic assumptions (e.g., infinite memory, no fragmentation)
- Validate results using page replacement algorithms (FIFO, LRU, Optimal)• Assessment focuses on memory management understanding and performance comparison



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Assignment 4: AI-Based Disk Scheduling Evaluation

- **Students prompt an AI tool: Suggest the optimal disk scheduling algorithm for minimizing seek time.**
- **Students must:**
 - Compare AI recommendations with algorithms like FCFS, SSTF, SCAN
 - Identify trade-offs such as starvation and fairness
 - Validate results through numerical examples or simulation
- **Assessment focuses on performance analysis and algorithm comparison**



Design Studio II – (AI Enabling System)

Course Code: 26DS2406

Semester:

Programs: B. Tech. CSE (Data Science)

Course Type: Core – Skill Enhancement Course Credits: 2

L – T – P – J: 1 – 0 – 2 – 0

Prerequisite: Fundamentals of Python Programming, Data Structures, Basic Machine Learning, Deep Learning Fundamentals

Course Vision

To develop highly skilled AI system engineers capable of designing, optimizing, and deploying next-generation intelligent systems by integrating foundation models, semantic computing, autonomous AI workflows, edge intelligence, and scalable AI infrastructure. The course emphasizes practical implementation, production-ready AI deployment, and sustainable computing practices to address emerging industrial and societal challenges.

Course Objectives

- Understand the architectural principles of modern foundation models, state-space models, and multimodal AI systems for enterprise-scale intelligent applications.
- Design and implement semantic memory systems using high-dimensional embeddings, vector databases, and advanced indexing techniques for efficient knowledge retrieval.
- Develop autonomous AI workflows through stateful orchestration, event-driven execution, distributed task scheduling, and collaborative multi-agent systems.
- Optimize and deploy deep learning models on resource-constrained edge devices using model compression, hardware-specific compilation, and edge-cloud collaborative inference.
- Build scalable, reliable, and sustainable production AI infrastructure by leveraging distributed inference frameworks, GPU resource optimization, throughput engineering, and Green AI principles.

Course Outcomes (CO)

CO	Description	Blooms Level
CO1	Explain the architecture and operational characteristics of foundation models, Mixture-of-Experts, State Space Models, and multimodal AI systems for enterprise applications.	L2
CO2	Design semantic computing systems using embedding models, vector databases, and optimized indexing techniques for scalable knowledge management.	L3
CO3	Develop autonomous AI workflows using stateful orchestration, event-driven processing, and distributed multi-agent architectures.	L3
CO4	Optimize and deploy AI models on embedded and edge	L4



CO	Description	Blooms Level
	computing platforms using hardware-aware optimization and inference acceleration techniques.	
CO5	Evaluate and implement scalable production AI infrastructure with efficient GPU utilization, distributed serving, reliability engineering, and sustainable AI deployment strategies.	L5

Broad Course Description

Artificial Intelligence has evolved from standalone predictive models to intelligent systems capable of reasoning, perception, semantic understanding, autonomous decision-making, and large-scale deployment across cloud, edge, and embedded environments. Modern AI-enabled systems increasingly rely on foundation models, multimodal learning, semantic memory, distributed workflows, and optimized hardware infrastructure to deliver scalable, efficient, and reliable intelligent services. This course introduces students to the principles and engineering practices required to design next-generation AI-enabled systems. It emphasizes foundation model architectures, semantic computing using vector databases, autonomous workflow engineering, edge intelligence, and production AI infrastructure. Students gain practical experience in building enterprise-grade AI applications using modern frameworks and hardware accelerators while understanding performance optimization, scalability, reliability, and sustainable AI deployment. The course bridges the gap between academic AI concepts and industrial AI engineering, preparing graduates for careers in AI systems development, intelligent automation, edge AI, and enterprise AI platform engineering.

Course Modules

Module 1: Next-Generation Foundation Models & System Architecture (6 Hours)

Evolution of Foundation Models and modern AI system architectures; Advanced Generative Architectures; Multimodal AI System Design; Modern AI Infrastructure; Industrial Model Selection

Module 2: High-Dimensional Memory Systems & Semantic Computing (7 Hours)

Embedding Engineering & Representation Learning; Production-Grade Vector Database Systems; Advanced Context Engineering; Semantic Indexing & Knowledge Synchronization.

Module 3: Autonomous AI Agent & Workflow Engineering (6 Hours)

Stateful Workflow Engineering; Event-Driven AI Orchestration; Multi-Agent Service Composition; Distributed Task Scheduling

Module 4: Edge Intelligence & Hardware Optimization (6 Hours)

Model Compression Techniques; Hardware-Specific Compilation; Embedded AI Deployment; Edge-to-Cloud Telemetry.

Module 5: Production AI Infrastructure & Reliability Engineering (5 Hours)

GPU Resource Management; Distributed AI Serving; Performance Optimization; Green AI Engineering.

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)



AI Integration:

Module 1: Next-Generation Foundation Models & System Architecture

AI Integration: Students will employ AI-assisted engineering platforms to explore and benchmark modern foundation models, analyze multimodal architectures, and compare computational performance under different deployment environments. AI copilots will support system configuration, architecture exploration, and infrastructure optimization while enabling rapid experimentation with open-weight foundation models.

Module 2: High-Dimensional Memory Systems & Semantic Computing

AI Integration: Students will build semantic computing pipelines by generating domain-specific embeddings, constructing optimized vector indexes, and implementing high-performance semantic search systems. AI-assisted development environments will be utilized to evaluate embedding quality, optimize retrieval performance, automate indexing workflows, and benchmark different vector database technologies for real-world enterprise applications.

Module 3: Autonomous AI Agent & Workflow Engineering

AI Integration: Students will develop intelligent workflow pipelines capable of coordinating multiple AI services through event-driven execution and collaborative task orchestration. AI-assisted development environments will be used to design workflow graphs, automate service interactions, optimize execution pipelines, and evaluate system scalability under distributed execution scenarios.

Module 4: Edge Intelligence & Hardware Optimization

AI Integration: Students will utilize AI-assisted development environments to optimize deep learning models for deployment on edge and embedded platforms. AI-powered profiling and optimization tools will be employed to recommend efficient quantization, pruning, and hardware-specific compilation strategies while benchmarking inference latency, memory utilization, and energy efficiency. Intelligent development assistants will further support model conversion, deployment automation, and performance analysis across heterogeneous edge devices, enabling students to design scalable and resource-efficient AI solutions for real-world applications.

Module 5: Production AI Infrastructure & Reliability Engineering

AI Integration: Students will optimize and deploy AI applications across heterogeneous hardware platforms using AI-assisted profiling, automated optimization techniques, and distributed inference frameworks. They will benchmark model latency, throughput, GPU utilization, memory efficiency, and energy consumption while exploring scalable deployment strategies suitable for enterprise AI systems.

Authentic Intelligence Reflection:

Module 1: Next-Generation Foundation Models & System Architecture

Authentic Intelligence Reflection: While foundation models demonstrate remarkable capabilities in reasoning and multimodal understanding, successful deployment depends on informed engineering decisions regarding computational efficiency, scalability, and resource utilization. Students will critically evaluate architectural trade-offs between dense and sparse models, analyze infrastructure requirements, and reflect on the balance between model capability, operational cost, environmental sustainability, and real-world deployment



constraints. This reflection encourages responsible engineering practices where human expertise complements AI-driven system design to achieve efficient, reliable, and sustainable intelligent systems.

Module 2: High-Dimensional Memory Systems & Semantic Computing

Authentic Intelligence Reflection: Semantic memory systems significantly enhance the contextual understanding of AI applications; however, their effectiveness depends on careful engineering of embeddings, indexing strategies, and knowledge synchronization processes. Students will critically examine how embedding quality, indexing efficiency, and semantic retrieval influence system accuracy, scalability, and responsiveness. They will also reflect on the importance of maintaining current and trustworthy knowledge repositories while balancing retrieval performance, computational resources, and real-world deployment requirements. This reflection reinforces the role of human expertise in designing robust, efficient, and scalable semantic AI systems suitable for enterprise environments.

Module 3: Autonomous AI Agent & Workflow Engineering

Authentic Intelligence Reflection: Autonomous AI systems extend beyond single-model inference by coordinating multiple intelligent components capable of planning, collaboration, and adaptive execution. Students will critically evaluate the reliability, robustness, and transparency of autonomous workflows while considering human oversight in complex decision-making processes. Reflection will emphasize designing trustworthy AI systems that remain resilient under uncertain environments, unexpected failures, and dynamically changing operational requirements.

Module 4: Edge Intelligence & Hardware Optimization

Authentic Intelligence Reflection: Students will critically evaluate the trade-offs between model accuracy, computational efficiency, energy consumption, and deployment constraints when designing edge AI solutions. They will reflect on the importance of human expertise in selecting appropriate optimization techniques, validating AI system performance, and ensuring reliability across diverse hardware platforms. Through practical deployment and performance analysis, students will develop the ability to make informed engineering decisions that balance resource limitations, operational requirements, and sustainability, reinforcing the role of human intelligence in building efficient, trustworthy, and real-world AI-enabled edge systems.

Module 5: Production AI Infrastructure & Reliability Engineering

Authentic Intelligence Reflection: Efficient AI deployment requires balancing computational performance, operational cost, environmental sustainability, and system reliability. Students will evaluate how optimization techniques influence model performance and resource consumption while recognizing the importance of sustainable computing practices. Reflection will focus on responsible AI engineering, emphasizing efficient infrastructure utilization, energy-conscious deployment, and resilient production systems that support long-term societal and industrial needs.

List of Experiments For Lab Integrated Course

All experiments build a single AI-enabled application of the student's choice (open project brief), carried from data pipeline construction through model development, API integration, and deployment. Each module deepens one stage of the AI development lifecycle using a specific



toolchain.

No.	Experiment	Module Link
1	Explore Mixture-of-Experts (MoE) architectures and compare dense versus sparse routing.	Module 1
2	Benchmark Transformer and Mamba architectures for long-context inference.	Module 1
3	Generate domain-specific embeddings using open-source embedding models.	Module 2
4	Implement semantic similarity search using Milvus and Pinecone.	Module 2
5	Optimize vector indexing using HNSW and Product Quantization (PQ).	Module 3
6	Design semantic indexing pipelines with automated synchronization between SQL and vector databases.	Module 3
7	Develop stateful AI workflows using LangGraph.	Module 4
8	Build collaborative multi-agent systems using CrewAI or Microsoft AutoGen.	Module 4
9	Optimize deep learning models using Quantization-Aware Training (QAT) and TensorRT.	Module 5
10	Deploy optimized AI models on Raspberry Pi or NVIDIA Jetson Nano.	Module 5
11	Configure distributed model serving using Triton Inference Server or vLLM.	Module 5

Assessment Scheme based on the course

Criteria	Weightage of Marks
Lab Reports	10
Mini Project	50
Lab Internal Exam	30
Hackathon / Technical Competition / Paper presentation / Global / DLI certifications	10
Total	100 marks

Recommended Tools & Platforms

Foundation Models & AI Systems: Google Cloud AI Platform, Microsoft Azure Cognitive Services, Amazon Web Services (AWS SageMaker), OpenAI API, Hugging Face (Model Hub & Inference APIs).

Semantic Computing & Memory Systems: Python (pandas, NumPy, matplotlib, seaborn), Google Colab, Jupyter Notebook, scikit-learn, SHAP (Explainable AI), FAISS / Pinecone / Weaviate (Vector Databases).



Autonomous AI Agents & Workflows: H2O.ai AutoML, Google AutoML Tables, TensorFlow, Keras, LangChain, Hugging Face Transformers, Apache Airflow (conceptual), Ray (distributed computing).

Edge Intelligence & Optimization: TensorFlow Lite, ONNX Runtime, Model quantization & pruning tools: Docker, Raspberry Pi / NVIDIA Jetson, Google Cloud IoT / telemetry services.

Production AI Infrastructure: Flask, FastAPI, Google Cloud Run, Kubernetes (conceptual), Git, GitHub, GitHub Copilot, VS Code, Prometheus / Grafana (conceptual), NVIDIA GPU monitoring tools.

Reference Resources

TensorFlow: <https://www.tensorflow.org>

scikit-learn: <https://scikit-learn.org>

Google Cloud AI: <https://cloud.google.com/ai>

Microsoft Azure AI: <https://azure.microsoft.com>

Hugging Face: <https://huggingface.co>

CO PO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PS01	PS02
CO1	3	3	2	2	2	1	1	1	1	1	1	2	2	1
CO2	3	3	3	2	3	1	1	1	2	1	1	2	3	2
CO3	3	3	3	3	3	2	1	2	2	2	2	2	3	3
CO4	3	3	3	3	3	2	2	1	1	2	1	2	3	3
CO5	3	3	3	3	3	2	2	2	2	2	2	3	3	3
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)														

Additional Resources

Additional AI Systems & Infrastructure Resources

Online Documentation

Official documentation for Hugging Face Transformers, LangChain, LangGraph, CrewAI, Microsoft AutoGen, TensorFlow, ONNX Runtime, NVIDIA TensorRT, Triton Inference Server, vLLM, Milvus, Pinecone, FAISS, Docker, FastAPI, Google Cloud AI, Microsoft Azure AI, and NVIDIA Jetson Developer Resources.

E-Learning Platforms

Courses on Coursera, Udemy, DeepLearning.AI, Hugging Face Courses, NVIDIA Deep Learning Institute (DLI), Kaggle Learn, and Microsoft Learn covering Foundation Models, Large Language Models (LLMs), Semantic Search, Retrieval-Augmented Generation (RAG), AI Agents, Edge AI, MLOps, and Production AI Systems.

Coding & Collaboration Tools

Visual Studio Code with GitHub Copilot, Google Colab, Jupyter Notebook, GitHub, Docker Desktop, Postman, and NVIDIA Nsight tools for AI system development, testing, profiling, and



collaborative software engineering.

AI Deployment & Infrastructure Platforms

Docker, Hugging Face Spaces, Google Cloud Run, NVIDIA Triton Inference Server, vLLM, ONNX Runtime, TensorFlow Lite, Raspberry Pi, NVIDIA Jetson Nano, Microsoft Azure AI Services, Amazon SageMaker, and Google Cloud AI Platform for scalable deployment and inference optimization.

YouTube Channels

Channels covering Foundation Models, Large Language Models, Hugging Face, LangChain, LangGraph, CrewAI, NVIDIA Jetson, TensorRT optimization, Vector Databases, Edge AI, MLOps, and Production AI Engineering through practical demonstrations and project-based learning.

Project Component

Students will develop a single AI-enabled intelligent application of their own choice (open project brief) and progressively enhance it throughout the five modules of the course. The project should integrate foundation models, semantic computing, autonomous AI workflows, edge deployment, and production-ready infrastructure. The same project will evolve from concept design through deployment, optimization, and maintenance.

Suggested project themes include:

- Enterprise Knowledge Assistant using RAG
- AI Customer Support Chatbot
- Multimodal Document Intelligence System
- Medical Image Analysis Assistant
- AI-Based Smart Campus Assistant
- Intelligent Research Paper Assistant
- AI Code Generation Assistant
- AI-Powered Recommendation System
- Autonomous Multi-Agent Task Management System
- Any equivalent AI-enabled application approved by the instructor.

AI Development Lifecycle & Deliverables

Stage 1 – Foundation Model Selection & AI System Architecture (Module 1)

Goal

- Design the AI-enabled application architecture and evaluate suitable foundation models, multimodal systems, and deployment environments.
- Deliverables
- AI system proposal including problem statement, objectives, application workflow, expected outputs, and deployment requirements.
- Comparative evaluation of at least two foundation models or cloud AI services based on capability, computational requirements, and suitability.
- System architecture diagram illustrating AI components and infrastructure.
- GitHub repository initialized with project milestones and issue tracking.

Tools

Hugging Face, OpenAI API, Google Cloud AI, Microsoft Azure AI, GitHub, VS Code, ChatGPT or Claude.

Stage 2 – Semantic Computing & Knowledge Engineering (Module 2)



Goal

Develop semantic memory capabilities using embeddings and vector databases for efficient knowledge retrieval.

Deliverables

- Domain-specific embedding generation and evaluation.
- Vector database implementation using Milvus, Pinecone, FAISS, or Weaviate.
- Semantic search pipeline with optimized indexing.
- Performance comparison of different indexing techniques and retrieval accuracy.

Tools

Python, Sentence Transformers, Hugging Face Embeddings, FAISS, Milvus, Pinecone, Google Colab.

Stage 3 – Autonomous AI Workflow Development (Module 3)

Goal

Develop intelligent autonomous workflows capable of coordinating multiple AI services through collaborative agent-based execution.

Deliverables

- Stateful workflow graph implemented using LangGraph.
- Multi-agent system using CrewAI or Microsoft AutoGen.
- Event-driven workflow with automated task execution.
- Workflow performance analysis including latency, scalability, and reliability.

Tools

LangChain, LangGraph, CrewAI, Microsoft AutoGen, Python, GitHub Copilot.

Stage 4 – Edge Intelligence & Model Optimization (Module 4)

Goal

Optimize AI models for deployment on resource-constrained edge devices.

Deliverables

- Model optimization using Quantization, Pruning, or TensorRT.
- ONNX or TensorFlow Lite model conversion.
- Edge deployment on Raspberry Pi or NVIDIA Jetson Nano (or simulation if hardware is unavailable).
- Comparative evaluation of inference latency, memory consumption, throughput, and energy efficiency before and after optimization.

Tools

TensorFlow Lite, ONNX Runtime, TensorRT, NVIDIA Jetson SDK, Raspberry Pi, Docker.

Stage 5 – Production AI Deployment & Reliability Engineering (Module 5)

Goal

Deploy the AI-enabled application using scalable production infrastructure and monitor its performance.

Deliverables

- REST API developed using Flask or FastAPI.
- Containerized application using Docker.
- Distributed inference using Triton Inference Server or vLLM.
- Public deployment on Hugging Face Spaces or Google Cloud Run.
- Performance benchmarking including latency, throughput, GPU utilization, memory efficiency, and reliability.
- One maintenance iteration (feature enhancement, model update, or optimization) with



redeployment documentation.

Tools

FastAPI, Flask, Docker, Triton Inference Server, vLLM, Hugging Face Spaces, Google Cloud Run, Prometheus, Grafana.

Final Capstone Deliverable

- A complete, scalable, and publicly deployed AI-enabled system demonstrating the integration of foundation models, semantic computing, autonomous AI workflows, edge intelligence, and production AI infrastructure. The final submission shall include:
- GitHub repository with complete source code, commit history, and issue tracking.
- AI system proposal and architecture documentation.
- Embedding generation and semantic retrieval implementation.
- Autonomous workflow and multi-agent system implementation.
- Optimized edge-deployable AI model.
- REST API and Dockerized application.
- Public deployment link (Hugging Face Spaces or Google Cloud Run).
- Performance evaluation report covering latency, throughput, GPU utilization, memory efficiency, and deployment reliability.
- A reflective report discussing the AI engineering lifecycle, technologies used at each stage, optimization strategies adopted, and the complementary roles of AI-assisted development and human engineering judgment.



RESPONSIBLE AI, ETHICS & POLICY

Course Code: 26DS3501

SEMESTER – V

Programs: For all branches

Course Type: Core

Credits: 02

L-T-P-J: 2-0-0-0

Prerequisites:

Course Vision

To equip Undergraduate students with the knowledge and ethical perspective required to design, develop, and deploy trustworthy, fair, and accountable AI systems, while understanding the societal, legal, and policy implications of artificial intelligence.

Course Objectives

This course will enable students to:

1. Understand ethical frameworks and principles relevant to the design and deployment of artificial intelligence technologies.
2. Analyze the societal, legal, and policy implications of AI adoption across different sectors.
3. Apply the principles of Responsible AI in the development and evaluation of AI systems.
4. Evaluate and design AI solutions that are ethical, transparent, and socially responsible.
5. Design responsible AI strategies and governance mechanisms aligned with global standards and regulatory frameworks.

Course Outcomes (CO)

CO	Description	Blooms Level
C01	Explain key concepts and principles of Responsible AI, including fairness, transparency, accountability, and privacy.	L2
C02	Analyze ethical challenges and societal impacts associated with AI technologies.	L4
C03	Evaluate AI systems for bias, reliability, and ethical risks using appropriate frameworks.	L5
C04	Interpret AI governance models, regulatory guidelines, and policy frameworks related to responsible AI for ethical governance.	L2
C05	Apply responsible AI principles in the design and development of trustworthy AI solutions.	L3



CO-PO mapping

COs	Program Outcomes (POs)											PSOs		
COs	1	2	3	4	5	6	7	8	9	10	11		1	2
CO1	3	2	1			3	1	3				2	1	
CO2	2	3	2	2		3	2	3				2	1	
CO3	3	3	2	2	3	3	1	3	1			2	1	
CO4	2	3	2	2	2	3	2	3	1			2	2	
CO5	3	2	3	3	3	3	2	3	2		2	2	2	
3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)														

Course Modules

MODULE – I Foundation of Responsible and Authentic Intelligence (06 Hours)

Concept of Responsible AI and Authentic Intelligence, Principles of Authentic Intelligence, Human-centred AI, trust, transparency, fairness, accountability, and inclusivity and its societal impact, Ethical frameworks for AI, Human values and cultural contexts in AI systems.

Applications: Ethical decision-making in AI-powered recommendation systems, Responsible deployment of AI assistants and chatbots.

MODULE – II Bias, Fairness, Transparency and Human-Aligned AI (06 Hours)

Sources of bias in AI, data bias, algorithmic bias, societal bias, Fairness metrics and bias mitigation techniques, Algorithmic transparency and explainability, Explainable AI (XAI) models and interpretability methods.

Applications: Transparent AI decision systems in government services, Bias detection in hiring and recruitment AI systems.

MODULE – III Privacy, Safety and Trustworthy AI Systems (06 Hours)

Privacy challenges in AI-driven systems, Privacy-preserving machine learning, differential privacy, federated learning, AI security risks and adversarial attacks, Robustness, safety, and reliability of AI models, Trustworthy AI lifecycle frameworks.

Applications: Privacy-preserving healthcare analytics, Secure AI-based financial fraud detection systems, autonomous vehicles.

MODULE – IV AI Governance, Regulation and Responsible Innovation (06 Hours)

AI governance and regulatory frameworks, Algorithmic accountability and auditing



mechanisms, AI risk assessment and impact evaluation models, Global AI policy initiatives and standards, Responsible innovation in AI development.

Applications: AI compliance frameworks in organizations, Ethical review boards for AI systems.

MODULE – V Designing Authentic and Responsible AI Systems

(06 Hours)

Human-centered AI design methodologies, Responsible AI development lifecycle, Social and economic implications of emerging AI technologies, Sustainable and inclusive AI development.

Applications: Responsible design of generative AI systems, AI solutions for social good, AI deployment in smart cities and governance.

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

1. AI development lifecycle and ethical checkpoints, Ethical considerations in data collection and model design.
2. Fairness evaluation in machine learning pipelines, explainable AI tools for model interpretation, datasets that ensure diversity and representation.
3. Privacy-preserving techniques in ML models, security testing and robustness checks in AI systems, reliability and safe AI systems.
4. Governance mechanisms in AI development processes, Compliance with regulatory standards in AI products, AI auditing tools.
5. Responsible AI frameworks into AI system design, AI models with transparency and accountability mechanisms, monitoring and evaluation of deployed AI systems.

Authentic Intelligence:

1. Human-centered AI design principles, Aligning AI behaviour with human values.
2. Value-aligned AI systems, Ethical dataset design and inclusive AI development.
3. Ethical protection of user privacy and digital rights, trust between humans and AI systems.
4. Ethical leadership in AI development, Accountability and transparency in AI-driven decision systems, AI innovation with societal values and public interest.
5. AI systems that respect human dignity and autonomy, fairness, inclusivity, and sustainability in AI innovation, AI systems remain aligned with human values and societal needs.

Assessment Scheme

Criteria	Weightage of Marks
Continuous Internal Assessment (CIA) – Formative	60
Semester End Examination (SEE) – Summative	40
Total	100 marks



TEXT BOOKS

1. M. Coeckelbergh – AI ethics, 2020, MIT press, ISBN: 978-0262538190.
2. S. Agarwal & S. Mishra – Responsible AI Implementing Ethical and Unbiased Algorithms, 2021, Springer, <https://doi.org/10.1007/978-3-030-76860-7>.

REFERENCE BOOKS

1. S. Voeneky, P. Kellmeyer, O. Mueller, and W. Burgard – The Cambridge handbook of responsible artificial intelligence: interdisciplinary perspectives, eds., 2022, Cambridge University Press, ISBN: 9781009207867 / 9781009207898, DOI: 10.1017/9781009207898.

E-RESOURCES

1. Virginia Dignum – Responsible AI.
<https://people.cs.umu.se/virginia/docs/responsibleAI2018.pdf>
2. Luciano Floridi – The Ethics of AI.
https://www.researchgate.net/publication/385157886_The_Ethics_of_Artificial_Intelligence

RECOMMENDED TOOLS

- Programming: Python (basic ML models)
- AI Libraries: Scikit-learn, TensorFlow (conceptual exposure)
- Explainability Tools: SHAP, LIME
- AI Tools: ChatGPT / AI assistants



Multimodal AI for Data Science

Course Code: 26DS3502

Semester: V

Programs Offered: B. Tech CSE (Data Science)

Course Type: Core

Credits: 3

L – T – P – J: 3-0-0-0

Prerequisites: Python Programming

Course Vision

To equip students with the knowledge, practical skills, and production-level competencies required to design, develop, and deploy multimodal AI systems that integrate text, vision, and audio modalities for real-world data science applications, aligned with responsible AI principles and industry practices.

Course Objectives

1. Understand multimodal data types, representation techniques, and cross-modal learning paradigms
2. Design multimodal learning architectures using transformers, attention mechanisms, and fusion strategies
3. Build end-to-end multimodal AI pipelines for tasks such as captioning, retrieval, and visual question answering
4. Implement observability, evaluation, and performance benchmarking for deployed multimodal systems
5. Architect scalable and production-grade multimodal AI systems with responsible AI considerations
6. Apply GPU-accelerated deep learning for multimodal model training and inference using NVIDIA frameworks

Course Outcomes (CO)

CO	Description	Blooms Level
C01	Explain multimodal AI concepts, data modalities, embedding representations, and cross-modal alignment techniques	L2
C02	Design multimodal learning architectures using transformers, attention mechanisms, and fusion strategies	L6
C03	Build end-to-end multimodal AI pipelines for classification, retrieval, captioning, and visual question answering	L6



C04	Evaluate multimodal AI systems for performance, robustness, bias, and ethical compliance using established metrics	L5
C05	Apply GPU-accelerated deep learning for multimodal training and inference using NVIDIA DLI-aligned framework	L3

Broad Course Description

This course provides a comprehensive foundation in multimodal AI, focusing on the design, development, and deployment of intelligent systems that integrate text, vision, and audio data. It introduces core concepts including multimodal data representation, embedding spaces, cross-modal alignment, and contrastive learning techniques essential for building intelligent data science-applications. The course progresses through transformer-based architectures, fusion strategies, and attention mechanisms that underpin state-of-the-art multimodal models. Students develop hands-on experience building systems for image captioning, visual question answering, semantic search, and emotion recognition using frameworks such as PyTorch, Hugging Face Transformers, and OpenCV.

CO-PO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
C01	2	1			1						1	3	2
C02	2	3	3	2	2						1	3	2
C03	2	3	3	2	3						2	3	3
C04	2	3	3	3	2	1	1				2	3	3
C05	2	3	3	3	3						2	3	3
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module 1: Introduction to Multimodal AI and Representation Learning (9 Hours)

Multimodal data and challenges, Representation learning (embeddings, latent spaces), Cross-modal learning and key tasks, Transformers in multimodal AI, Overview of models (CLIP, ALIGN, BLIP), Applications and AI integration, Bias, interpretability, and ethical issues

Module 2: Multimodal Representation Learning (9 Hours)

Feature extraction (CNNs, transformers, spectrograms), Joint embedding spaces, Contrastive and similarity learning, Self-supervised multimodal learning, Cross-modal retrieval and evaluation, Applications and AI integration, Robustness, fairness, and transparency issues

Module 3: Multimodal Architectures and Fusion Techniques (9 Hours)

Transformer-based architectures (ViT, BERT, cross-attention), Fusion strategies (early, late, hybrid), Co-attention and cross-attention, Multimodal generation (VQA, image captioning),



Overview of models (BLIP, Flamingo, LLaVA)

Module 4: Applications and Evaluation of Multimodal AI

(9 Hours)

End-to-end multimodal pipelines and integration, Evaluation metrics (captioning, retrieval, VQA), Video analytics and temporal modeling, Multimodal sentiment analysis and sensor fusion, Deployment and monitoring (APIs, drift detection), Applications and AI integration, Ethical, bias, and safety considerations

Module 5: GPU-Accelerated Multimodal AI and NVIDIA DLI-Aligned Deployment(9 Hours)

GPU-accelerated training (multi-GPU, mixed precision), Efficient inference (quantization, pruning, distillation), Model export and deployment (ONNX, Triton), Dataset engineering and responsible AI, NVIDIA DLI-based implementation, Performance optimization and benchmarking, Real-time applications and MLOps integration

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module 1: Students explore pre-trained multimodal models and embedding spaces using Hugging Face pipelines and CLIP-based tools. AI-assisted visualisation of embedding clusters is used to understand cross-modal alignment and zero-shot capabilities.

Module 2: Students apply contrastive learning frameworks and fine-tune CLIP-style models on curated datasets. AI-driven retrieval systems are used to Analyze embedding quality and cross-modal similarity across domains.

Module 3: Students build and experiment with transformer-based fusion architectures using Hugging Face and PyTorch. AI-generated captions and VQA outputs are Analyzed and critiqued for correctness, hallucinations, and representational gaps.

Module 4: Students deploy multimodal inference pipelines using FastAPI and LangChain, integrating Whisper for speech input. AI-driven evaluation tools are used to benchmark system outputs across captioning, retrieval, and sentiment tasks.

Module 5: Students design GPU-accelerated multimodal training pipelines and deploy models using NVIDIA Triton Inference Server. AI-driven profiling and benchmarking tools are used to optimize throughput, latency, and energy efficiency aligned with NVIDIA DLI content.

Authentic Intelligence Reflection (AI² Pedagogy):

Module 1: Understanding the limitations of cross-modal alignment and the risk of embedding bias inherited from large-scale training corpora. Students evaluate when pre-trained multimodal models are appropriate versus when domain-specific fine-tuning is required.

Module 2: Recognising fairness issues in similarity learning where demographic or cultural biases are encoded in embedding spaces. Students critically assess contrastive learning objectives and their implications for downstream retrieval fairness.

Module 3: Evaluating hallucinations and factual errors in multimodal generation systems,



including image captioning and VQA. Students develop critical judgment in assessing when AI-generated visual descriptions are unreliable or harmful.

Module 4: Understanding ethical risks in multimodal surveillance, clinical AI, and emotion recognition systems. Students Analyze trade-offs between system accuracy and explainability.

Module 5: Evaluating cost-performance trade-offs of GPU-accelerated multimodal infrastructure, including carbon footprint of large-scale training. Students reflect on governance responsibilities in deploying production-grade multimodal AI systems.

Assessment Scheme:

Criteria	Weightage of Marks
Class Tests/Assignments	10
Mid-Semester Examination	30
Mini Project	10
Hackathons/Research Publications/Certifications	10
End-Semester Examination	40
Total	100 marks

Recommended Tools

The following tools and platforms are recommended to support AI-integrated teaching, learning, and experimentation:

- Python 3.10+
- PyTorch / TensorFlow
- Hugging Face Transformers and Datasets
- OpenCV
- Whisper (OpenAI)
- LangChain
- NVIDIA Triton Inference Server
- NVIDIA TensorRT
- RAPIDS (cuDF, cuML)
- HPC Lecture (GPU computing environment)

Primary Textbooks

1. NVIDIA Corporation, NVIDIA Deep Learning Institute (DLI) Course Materials and Documentation, Latest Edition (Online Resource).
2. Ian Goodfellow, Yoshua Bengio, Aaron Courville, Deep Learning, MIT Press, 1st Edition, 2016.
3. Aurélien Géron, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, O'Reilly Media, 3rd Edition, 2022.
4. François Chollet, Deep Learning with Python, Manning Publications, 2nd Edition, 2021.



5. Aston Zhang et al., Dive into Deep Learning, Cambridge University Press (d2l.ai), Latest Edition.

Additional Resources

1. Hugging Face Documentation – huggingface.co/docs (Transformers, Datasets, BLIP, Whisper)
2. PyTorch Documentation – pytorch.org/docs (DDP, FSDP, TorchScript, ONNX export)
3. NVIDIA Developer Documentation – developer.nvidia.com (Triton, TensorRT, GPU Operator)
4. ACL Anthology – aclanthology.org (Multimodal NLP and Vision-Language research)
5. NeurIPS, CVPR, ICCV, ECCV Proceedings – landmark multimodal AI research papers
6. MS-COCO, Flickr30k, VQA v2, AudioSet – open multimodal benchmark datasets
7. Papers With Code – paperswithcode.com (multimodal model benchmarks and reproducible research)

Mini Project Component

Industry-Driven Projects:

Project 1 – Multimodal Semantic Search Engine

Problem: Design and deploy a search engine that accepts text queries and retrieves semantically relevant images and documents.

Concepts Used: Joint embeddings, contrastive learning, CLIP-based retrieval, cosine similarity, Hugging Face pipelines.

Project 2 – Image Captioning System for Accessibility

Problem: Build an automated image captioning system to generate descriptive text for visual content for accessibility applications.

Concepts Used: Encoder-decoder architectures, cross-attention, BLIP, beam search decoding, BLEU and CIDEr evaluation.

Project 3 – Visual Question Answering (VQA) System

Problem: Develop a VQA system capable of answering natural language questions about input images.

Concepts Used: Multimodal transformers, cross-attention fusion, VQA v2 dataset, VQA Accuracy metric.

Project 4 – Multimodal Emotion and Sentiment Recognition

Problem: Build an emotion recognition system that fuses audio speech features and textual transcripts for sentiment classification.

Concepts Used: Whisper for ASR, text embeddings, feature-level fusion, multimodal sentiment datasets (CMU-MOSI).

Project 5 – Video Analytics with Text Alignment

Problem: Design a pipeline that Analyzes video content by aligning visual frames with speech transcripts for scene understanding.

Concepts Used: Frame sampling, Whisper-based transcription, temporal alignment, multimodal



Project 6 – Healthcare Multimodal Diagnostic Assistant

Problem: Develop a prototype diagnostic assistant that correlates medical imaging data with clinical notes to support diagnosis.

Concepts Used: Vision-language alignment, domain-specific fine-tuning, retrieval-augmented generation (RAG), responsible AI audit.

Project 7 – Multimodal Retrieval-Augmented Generation (RAG) System

Problem: Build a RAG system that retrieves relevant multimodal context (images and text) to augment language model responses.

Concepts Used: LangChain, vector databases, CLIP-based retrieval, prompt engineering, response evaluation.

Project 8 – GPU-Accelerated Multimodal Training and Deployment Pipeline

Problem: Train a multimodal model on a GPU cluster and deploy it as a scalable inference service using NVIDIA Triton.

Concepts Used: PyTorch DDP, mixed-precision training, ONNX export, Triton Inference Server, latency and throughput benchmarking.

Project 9 – Autonomous Perception Prototype Using Sensor Fusion

Problem: Develop a perception prototype that fuses camera image data and textual metadata for object classification in autonomous scenarios.

Concepts Used: Sensor fusion, late fusion strategies, multimodal classification, evaluation under distribution shift.

Project 10 – Responsible AI Audit for a Deployed Multimodal System

Problem: Conduct a comprehensive responsible AI audit of an existing multimodal AI system, evaluating bias, fairness, interpretability, and governance.

Concepts Used: Fairness metrics, GradCAM-based explainability, demographic parity analysis, model card documentation.

AI-Powered Assignments

Assignment 1: AI-Based Multimodal Embedding Analysis

Students prompt an AI tool: Compare text-only and multimodal embedding approaches for a semantic search task and recommend the most suitable strategy.

Students must:

- Analyze the correctness of AI-generated embedding comparisons using retrieval metrics (Recall@K, mAP)
- Identify incorrect assumptions (e.g., ignoring domain shift, overestimating zero-shot capability)
- Improve the analysis using a custom CLIP fine-tuning experiment on a domain-specific dataset

Assessment focuses on representation learning understanding, metric-based reasoning, and critical evaluation of AI outputs.



Assignment 2: AI-Assisted Multimodal Data Preprocessing Pipeline

Students prompt an AI tool: Design an optimised preprocessing pipeline for a multimodal dataset containing images, audio, and text.

Students must:

- Analyze the correctness of preprocessing steps (normalisation, tokenisation, spectrogram extraction)
- Identify inefficiencies or missing steps in the AI-generated pipeline
- Refine the pipeline using PyTorch DataLoader optimisations and benchmarking results

Assessment focuses on data engineering skills, modality-specific preprocessing knowledge, and validation of AI outputs.

Assignment 3: AI-Based Multimodal Architecture Selection and Justification

Students prompt an AI tool: Recommend the best multimodal architecture for an image captioning task and justify the choice.

Students must:

- Analyze the suitability of the recommended architecture (encoder-decoder, cross-attention, BLIP-style)
- Identify incorrect assumptions about dataset size, compute constraints, or task requirements
- Validate and improve results using quantitative evaluation (BLEU, CIDEr, CLIPScore)

Assessment focuses on architecture reasoning, comparative evaluation, and analytical judgment.

Assignment 4: AI-Assisted Fusion Strategy Design

Students prompt an AI tool: Design an optimal fusion strategy for combining audio and text modalities for sentiment classification.

Students must:

- Analyze early, late, and hybrid fusion outputs and compare classification performance
- Identify misinterpretations in AI-recommended fusion approaches (e.g., ignoring temporal alignment)
- Improve the design using ablation experiments across fusion strategies

Assessment focuses on fusion strategy understanding, experimental design, and result interpretation.

Assignment 5: AI-Based Multimodal System Deployment Design

Students prompt an AI tool: Design a scalable deployment architecture for a real-time image captioning API serving high request volumes.

Students must:

- Analyze the feasibility of the proposed architecture (model serving, batching, latency trade-offs)
- Identify bottlenecks such as model size, GPU memory constraints, or lack of monitoring
- Refine the design using FastAPI and NVIDIA Triton Inference Server best practices

Assessment focuses on system design, scalability reasoning, and performance-aware deployment thinking.

Assignment 6: AI-Driven End-to-End Multimodal Pipeline Evaluation

Students prompt an AI tool: Design and evaluate a complete multimodal AI pipeline for a Visual Question Answering task, including data ingestion, model training, and inference.

Students must:

- Analyze the correctness of all pipeline stages (preprocessing → embedding → fusion → decoding → evaluation)



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

- Identify gaps in evaluation strategy (missing metrics, lack of bias audit, no latency profiling)



Big Data Analytics

Course Code: 26DS3503

Semester: V

Programs: B.Tech. CSE (Data Science)

Credits:4

Course Type: Core

L – T – P – J: 3 – 0 – 2 – 0

Prerequisites: Database Management Systems

Course Vision

To provide students with comprehensive knowledge of Big Data ecosystems and distributed computing frameworks, enabling them to design, implement, and optimize large-scale data processing pipelines using Hadoop, Apache Spark, and Hive for real-world analytical applications.

Course Objectives

1. Understand the core concepts of Big Data and the Hadoop v1 architecture including HDFS and MapReduce
2. Explore Hadoop v2 (YARN) enhancements and gain practical experience with ecosystem tools like Pig, Sqoop, and Flume
3. Apply Apache Spark fundamentals for scalable distributed data processing using RDDs and DataFrames
4. Implement advanced Spark modules including Spark SQL, Streaming, and MLlib for analytics and machine learning
5. Construct data warehousing solutions using Apache Hive and execute structured queries with HQL

Course Outcomes (CO)

CO	Description	Blooms Level
C01	Explain the characteristics of Big Data and describe the Hadoop v1 architecture and its limitations	L2
C02	Compare Hadoop v1 and v2 architectures and demonstrate data ingestion and processing using ecosystem tools	L2
C03	Develop distributed data processing applications using Apache Spark RDDs and Data Frames	L6
C04	Apply advanced Spark applications involving SQL, Streaming, and MLlib for real-time and predictive analytics	L3
C05	Construct data warehousing solutions using Hive and execute structured queries for large-scale data analysis	L3



Broad Course Description

Big Data Analytics is a 4-credit Core course (3 Theory + 2 Practical) under the CBCS scheme for CSE(Data Science) students . It covers the complete Big Data processing stack — from foundational Hadoop concepts and HDFS architecture through to distributed in-memory computing with Apache Spark and data warehousing with Apache Hive. The course balances theoretical depth with hands-on lab implementation.

The course spans five units: Introduction to Big Data and Hadoop v1, Hadoop v2 and YARN Architecture, Apache Spark Fundamentals, Advanced Apache Spark (SQL, MLlib, Streaming), and Apache Hive and Data Warehousing. Laboratory sessions (26 hours) provide hands-on experience in setting up Hadoop clusters, writing MapReduce and Spark programs, executing HQL queries, and building end-to-end analytical pipelines on distributed systems.

CO-PO mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
C01	2	1			1						1	2	2
C02	2	3			2						1	3	2
C03	2	3	3	2	3						1	3	3
C04	2	3	3	2	3						2	3	3
C05	2	3	3	3	3						2	3	3
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module 1: Introduction to Big Data and Hadoop v1 (9 Hours)

Introduction to Big Data, Characteristics (Volume, Velocity, Variety, Veracity, Value), Traditional vs Big Data systems, Google File System (GFS) and its architecture, Hadoop Ecosystem Overview, Core Components: HDFS and MapReduce, Hadoop v1 Architecture, HDFS: NameNode and DataNode, MapReduce Programming Model, YARN in Hadoop v1 (early concepts), Limitations of Hadoop v1

Module 2: Hadoop v2 and YARN Architecture (9 Hours)

Introduction to Hadoop v2, YARN Architecture: Resource Manager, Node Manager, Application Master, Comparison of Hadoop v1 vs v2, Enhanced scalability and fault tolerance in v2, HDFS Federation and High Availability, Hadoop Ecosystem tools: Pig, Sqoop, Flume (overview)

Module 3: Apache Spark Fundamentals (9 Hours)

Spark Overview, Resilient Distributed Datasets (RDD), Spark vs MapReduce, Spark Architecture: Driver, Executors, Cluster Manager, Spark Core, Spark SQL, Spark Streaming (overview), DataFrames and Datasets, Lazy Evaluation, Transformations vs Actions



Module 4: Advanced Apache Spark

(9 Hours)

Spark SQL and DataFrames, Schema inference, SQL queries on DataFrames, Working with Parquet, JSON, and ORC files, Spark MLlib, ML Pipeline components: Transformers, Estimators, Pipelines, Classification and Regression using Spark MLlib

Module 5: Apache Hive and Data Warehousing

(9 Hours)

Introduction to Hive, Hive Architecture: Compiler, Execution Engine, Hive Query Language (HQL), Data types, Managed and External tables, Partitions and Buckets, Data loading and schema evolution, Joins, Views, and Indexes in Hive, Hive vs RDBMS

Case Study: End-to-end Big Data pipeline — ingesting large-scale retail transaction data using Sqoop/Flume into HDFS, processing with Spark SQL and MLlib for demand prediction, and warehousing results in Hive for structured business reporting and analytics

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module 1: AI tools are used to explore Big Data characteristics through real-world dataset profiling and simulate HDFS data distribution patterns. Students use AI-assisted log analysis tools to identify NameNode bottlenecks and critically evaluate AI-generated explanations of MapReduce execution flows for correctness and completeness.

Module 2: AI-assisted cluster monitoring dashboards are used to visualize YARN resource allocation and Node Manager utilization patterns. Students critically evaluate AI-generated comparisons of Hadoop v1 vs v2 performance trade-offs and assess the suitability of Pig, Sqoop, and Flume for specific data ingestion scenarios.

Module 3: AI-powered Spark job profiling tools are used to analyze RDD lineage graphs, identify wide vs narrow transformations, and optimize execution plans. Students critically assess AI-generated Spark architecture diagrams and evaluate lazy evaluation strategies for correctness and performance implications.

Module 4: Students use AI-assisted model selection tools within Spark MLlib to evaluate pipeline configurations for classification and regression tasks. AI-generated SQL query plans on DataFrames are critically reviewed for optimization opportunities, and Spark Streaming anomaly detection outputs are validated against expected real-time data patterns.

Module 5: AI tools assist in generating HQL queries for complex joins, partitioning strategies, and schema evolution scenarios. Students critically evaluate AI-generated Hive query execution plans for correctness, partition pruning effectiveness, and performance trade-offs compared to equivalent RDBMS approaches.



List of Experiments

No.	Experiment	Module Link
1	Set up and validate a Hadoop v1 environment in pseudo- distributed mode for simulating a single-node cluster	Unit 1
2	Develop and execute sample programs to explore HDFS operations and basic MapReduce functionalities	Unit 1
3	Install and configure Hadoop YARN; execute MapReduce jobs using YARN Resource Manager on Hadoop v2	Unit 2
4	Set up Apache Hive on a Hadoop v2 cluster and validate by executing basic DDL commands	Unit 2 / Unit 5
5	Develop Hive scripts for basic data manipulation and retrieval using HQL (SELECT, WHERE, GROUP BY)	Unit 5
6	Implement Hive partitioning and bucketing techniques to optimize query performance	Unit 5
7	Write and execute Spark programs demonstrating RDD and DataFrame transformations and actions	Unit 3
8	Implement Spark SQL for structured data queries and build a simple ML pipeline using Spark MLlib	Unit 4

Assessment Scheme based on the course

Criteria	Weightage of Marks
Lab Reports / Practical Experiments	10
Class Tests / Assignments	10
Mid-Semester Examination	30
Mini Project	10
End-Semester Examination	40
Total	100 marks

Recommended Tools

Primary Textbooks

1. Tom White, Hadoop: The Definitive Guide, 4th Edition, O'Reilly Media, ISBN: 9781491901632
2. Jules S. Damji, Brooke Wenig, Tathagata Das, Denny Lee, Learning Spark: Lightning-Fast



Data Analytics, 2nd Edition, O'Reilly Media, ISBN: 9781492050049

3. Dean Wampler and Jason Rutherglen, Programming Hive, O'Reilly Media
4. Apache Spark Documentation – spark.apache.org/docs/latest
5. Apache Hadoop Documentation – hadoop.apache.org/docs
6. Arvind Sathi, Big Data Analytics: Disruptive Technologies for Changing the Game, IBM Press

Additional AI & Computational Resources

1. Apache Hive Documentation – cwiki.apache.org/confluence/display/Hive/Home
2. NPTEL: Big Data Computing – nptel.ac.in
3. Databricks Academy: Apache Spark – academy.databricks.com
4. Cloudera Big Data Platform Tutorials – cloudera.com/tutorials
5. Kaggle Big Data Datasets – kaggle.com/datasets

Mini Project Component

The following projects are suggested based on course outcomes. Students may attempt one project individually or in pairs.

Project 1 – Distributed Log Analysis Pipeline using Hadoop and Hive

Problem: Build an end-to-end pipeline to ingest, process, and analyze web server log data using HDFS for storage, MapReduce for parsing, and Hive for structured querying and reporting

Concepts Used: HDFS operations, MapReduce programming, Hive tables, HQL queries, partitioning, data warehousing

Industrial Relevance: Web analytics, cybersecurity log monitoring, e-commerce clickstream analysis

Project 2 – Real-Time Sales Analytics using Apache Spark

Problem: Develop a Spark-based analytics pipeline that processes a large retail sales dataset, performs SQL-based aggregations using Spark SQL, and applies a classification model via Spark MLlib to predict customer segments

Concepts Used: Spark DataFrames, Spark SQL, Spark MLlib pipelines, Parquet/ORC file handling, transformers and estimators

Industrial Relevance: Retail analytics, customer segmentation, supply chain demand prediction

Project 3 – Big Data Ingestion and Processing with Hadoop Ecosystem Tools

Problem: Design a complete data ingestion system using Sqoop (for RDBMS to HDFS transfer) and Flume (for streaming log collection), followed by Pig or Hive-based processing and analysis

Concepts Used: Sqoop, Flume, Pig, YARN resource management, Hive data loading, ecosystem integration

Industrial Relevance: Enterprise data lake construction, ETL pipelines, batch data integration



AI-Powered Assignments

Assignment 1: AI-Assisted Hadoop Architecture Design Evaluation

Students prompt an AI tool: Design a Hadoop v2 cluster architecture for a healthcare organization managing 5 TB of daily patient data, specifying NameNode configuration, YARN resource allocation, and fault tolerance mechanisms.

Students must:

- Evaluate whether the proposed NameNode configuration adequately handles the data volume and accounts for HDFS Federation requirements
- Identify single points of failure and assess whether the YARN resource allocation strategy is appropriate for the given workload mix
- Improve the design by applying HDFS High Availability, appropriate replication factors, and capacity planning based on standard guidelines

Assessment focuses on: Hadoop architecture reasoning, fault tolerance design, and critical evaluation of AI-generated cluster configuration decisions

Assignment 2: AI-Generated Spark Optimization Analysis

Students prompt an AI tool: Optimize a Spark job that processes a 100 GB e-commerce transaction dataset to compute top-selling products by region, identifying bottlenecks and improving execution time.

Students must:

- Analyze whether the suggested partitioning strategy, caching decisions, and shuffle operations are appropriate for the data size and cluster configuration
- Identify wide transformation bottlenecks, data skew issues, or inefficient join strategies in the AI-proposed solution
- Refine the Spark job using proper broadcast joins, partition tuning, and DataFrame API optimizations; validate improvements using the Spark UI

Assessment focuses on: Spark optimization understanding, distributed computing reasoning, and validation of AI-generated performance tuning strategies

Assignment 3: AI-Based Hive Query Optimization Critique

Students prompt an AI tool: Write HQL queries to analyze a 10 GB sales dataset stored in Hive, computing monthly revenue trends by product category with efficient use of partitioning and indexing.

Students must:

- Evaluate whether the AI-generated HQL uses correct partitioning keys and whether partition pruning is effectively applied to reduce full table scans
- Identify inefficient join strategies (e.g., missing map-side joins for small tables) or incorrect bucketing that may degrade query performance
- Improve the queries by applying proper partition design, bucketed joins, and ORC file format with statistics collection for optimal query execution

Assessment focuses on: Hive query design, partition and optimization understanding, and critical evaluation of AI-generated HQL for correctness and performance



Deep Learning Essentials

Course Code: 26DS3504

Semester: V

Program(s): B.Tech CSE (Data Science)

Course Type: Core

Credits: 4

L – T – P – J: 3 – 0 – 2 – 0

Prerequisite(s): Machine Learning.

Course Vision

To provide foundational and advanced knowledge of deep learning, including neural networks, CNNs, sequence models, and modern practical deep learning techniques for real-world applications.

Course Objectives

1. Understand fundamental concepts and architectures of deep learning
2. Analyze and apply neural network optimization techniques
3. Explore convolutional and recurrent neural networks for real-world tasks
4. Implement deep learning models using modern frameworks such as PyTorch
5. Apply transfer learning, data augmentation, and model deployment techniques

Course Outcomes (CO)

CO	Description	Blooms Level
C01	Explain fundamental deep learning concepts and architectures	L2
C02	Apply optimization techniques for training deep learning models	L3
C03	Analyze CNN and RNN architectures for different applications	L4
C04	Implement deep learning models using modern frameworks	L3
C05	Develop and evaluate real-world deep learning applications	L6

Broad Course Description

Deep Learning is a 4-credit Core course in the 6th Semester. It provides a thorough grounding in the theory and practice of deep neural networks — from perceptrons and MLPs to convolutional, recurrent, and transformer-based architectures. Students gain both conceptual depth and hands-on implementation skills. The course spans five modules: Foundations of Deep Learning, Optimization and Computation, CNNs and Computer Vision, Sequence Models and Recurrent Networks, and Practical Deep Learning with NVIDIA DLI Integration. Laboratory sessions reinforce each module with PyTorch-



based implementations, covering image classification, NLP tasks, transfer learning, and real-world model deployment.

CO-PO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
CO1	2	1			1						1	2	2
CO2	2	3			2						1	3	2
CO3	2	3			3						1	3	2
CO4	2	3	3	2	3						2	3	3
CO5	2	3	3	3	3						2	3	3
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module 1: Foundations of Deep Learning

(9 Hours)

History: Evolution from the McCulloch-Pitts neuron to the Deep Learning explosion, Transitioning from biological neurons to artificial nodes, The Perceptron: Architecture of a single-layer network, Perceptron Convergence Theorem, Perceptron Learning Algorithm, Neural Network Architectures & Mathematics: Multilayer Perceptrons (MLPs), Representation Power of MLPs, Universal Approximation Theorem, Training Dynamics, VC Dimension, Deep Network Optimization & Regularization: Deep vs. Shallow Networks, Batch Normalization, Hyperparameter Foundations, Generative Adversarial Networks (GAN) Basics, Semi-supervised Learning.

Module 2: Optimization and Computation in Deep Learning

(9 Hours)

The Mechanics of Neural Computation: Computational Graphs(DAG) representation of neural operations, Forward Propagation, Backward Propagation, Computational Graphs, Layers and Blocks, Activation Functions: (Sigmoid, Tanh, ReLU, Leaky ReLU, and Softmax), Optimization Algorithms: Gradient Descent (GD) and Stochastic Gradient Descent (SGD), Accelerated Convergence: Momentum and Nesterov Accelerated Gradient (NAG), Adaptive Learning Rates: Evolution of optimizers from AdaGrad and RMSProp to the industry-standard Adam (Adaptive Moment Estimation), Stability, Generalization, and Ensembles: Weight Initialization, Batch Normalization, Regularization Techniques: Dropout, Ensemble Methods (Bagging and Boosting basics).

Module 3: Convolutional Neural Networks and Computer Vision

(9 Hours)

CNN Fundamentals: layers, feature maps, and Pooling CNNs, Specialized Architectures: Deep Dive into VGG, Inception, ResNet (Residual Learning), and DenseNet. Emerging Trends: Vision Transformers (ViTs) and their self-attention mechanisms. Image Segmentation: The U-Net architecture, encoder-decoder, Object Detection: Real-time detection principles using YOLO, Visualizing CNN: Guided Backpropagation and Deep Dream, Applied Computer Vision: Image Classification using MNIST and American Sign Language datasets, covering data preparation, model building, training, and evaluation.



Module 4: Recurrent Foundations and Temporal Dynamics (9 Hours)

Sequence Modeling, Recurrent Neural Networks (RNNs), Training Dynamics: Backpropagation Through Time (BPTT): Backpropagation Through Time (BPTT), The Stability Problem: Vanishing and Exploding Gradients and the use of Truncated BPTT, Gated Mechanisms for Long-Term Memory: Long Short-Term Memory (LSTM), Gated Recurrent Units (GRUs), The Attention Paradigm and Transformers: The Bottleneck Problem, Attention Mechanism, Attention over images, Hierarchical Attention, the Transformer Architecture: Multi-headed Self-Attention, Cross Attention.

Module 5: Practical Deep Learning and NVIDIA DLI Integration (9 Hours)

Image Pre-processing & Augmentation Strategies, and techniques, Synthetic Data Generation, Transfer Learning using pre-trained models, Model deployment and inference, working with PyTorch framework, Pre-trained model fine-tuning, NLP using deep learning: BERT basics, Tokenization, Text segmentation, Question answering systems, Text classification and Named Entity Recognition (NER).

Case Study: End-to-end implementation of image classification and NLP models using pre-trained architectures and transfer learning, demonstrating the complete pipeline from data preparation to model deployment

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module 1: Students use AI tools to explore the evolution of deep learning architectures and visualize perceptron decision boundaries. AI-generated explanations of GAN outputs and semi-supervised learning scenarios are critically evaluated for conceptual accuracy.

Module 2: AI-assisted optimization dashboards are used to visualize gradient descent trajectories, compare optimizer behaviours (Adam vs SGD), and analyze batch normalization effects. Students critique AI-generated convergence Analyzes against theoretical expectations.

Module 3: Students use AI-powered visualization tools (Grad-CAM, guided backpropagation) to interpret CNN feature maps. AI-generated image classification outputs on MNIST and ASL datasets are evaluated for accuracy, bias, and generalization capability.

Module 4: AI tools are used to visualize LSTM cell state dynamics and attention weight distributions. Students critically assess AI-generated sequence predictions and transformer attention maps against ground truth for reliability and interpretability.



Module 5: Students use NVIDIA DLI resources and PyTorch to implement transfer learning pipelines. AI-assisted deployment tools are evaluated for production readiness, and BERT-based QA outputs are critically reviewed for context-awareness and response quality.

List of Experiments

No.	Experiment	Module Link
1	Implement a Perceptron and MLP from scratch using Python/NumPy and evaluate on a classification dataset	Module 1
2	Compare gradient descent optimizers (SGD, Adam, RMSProp) on a deep network and visualize convergence curves	Module 2
3	Implement batch normalization and dropout; analyze their effect on training stability and accuracy	Module 2
4	Build and train a CNN (LeNet/AlexNet style) for MNIST digit classification using PyTorch	Module 3
5	Implement image classification on the American Sign Language (ASL) dataset using a custom CNN	Module 3
6	Build an LSTM model for time-series prediction; compare with vanilla RNN	Module 4
7	Implement a sequence-to-sequence model with attention mechanism for a simple translation task	Module 4
8	Apply transfer learning using a pre-trained ResNet/VGG model for a custom image classification task	Module 5
9	Fine-tune a pre-trained BERT model for text classification or named entity recognition	Module 5

Assessment Scheme based on the course

Criteria	Weightage of Marks
Lab Reports / Project	10
Class Tests / Assignments	10
Mid-Semester Examination	30
Hackathon / Technical Competition / Paper Presentation / DLI Certifications	10
End-Semester Examination	40
Total	100 marks



Recommended Tools

Primary Textbooks

1. Aston Zhang, Zack C. Lipton, Mu Li, Alex J. Smola, Dive into Deep Learning, Amazon Science, 2020 – d2l.ai
2. Ian Goodfellow, Yoshua Bengio, Aaron Courville, Deep Learning, MIT Press, 2016
3. NVIDIA Deep Learning Institute Course Materials – learn.nvidia.com
4. Sebastian Raschka, Yuxi Liu, Vahid Mirjalili, Machine Learning with PyTorch and Scikit-Learn, Packt, 2022
5. François Chollet, Deep Learning with Python, Manning Publications, 2nd Edition, 2021
6. PyTorch Documentation – pytorch.org/docs

Additional AI & Computational Resources

1. NPTEL: Deep Learning – nptel.ac.in
2. Fast.ai: Practical Deep Learning for Coders – fast.ai
3. Andrej Karpathy: Neural Networks Zero to Hero – karpathy.ai
4. Papers With Code – CV & DL Benchmarks – paperswithcode.com
5. Google Colab / Jupyter with GPU – colab.research.google.com

Mini Project Component

The following projects are suggested based on course outcomes. Students may attempt one project individually or in pairs.

Project 1 – Real-Time Object Detection System

Problem: Build a real-time object detection system using YOLO or a fine-tuned ResNet model on a custom dataset

Concepts Used: CNN architectures, transfer learning, data augmentation, model evaluation

Industrial Relevance: Smart surveillance, autonomous vehicles, manufacturing quality control

Project 2 – Sentiment Analysis using BERT Fine-Tuning

Problem: Fine-tune a pre-trained BERT model for sentiment classification on a domain-specific dataset

Concepts Used: Transformer architecture, BERT tokenization, fine-tuning, evaluation metrics

Industrial Relevance: Customer feedback analysis, social media monitoring, brand management

Project 3 – Time-Series Anomaly Detection using LSTM

Problem: Develop an LSTM-based model to detect anomalies in IoT sensor or financial time-series data

Concepts Used: LSTM, sequence modeling, threshold-based detection, BPTT

Industrial Relevance: Predictive maintenance, fraud detection, network intrusion detection



AI-Powered Assignments

Assignment 1: AI-Assisted Neural Network Design Evaluation

Students prompt an AI tool: Design a deep learning architecture for classifying handwritten digits with at least 98% accuracy and explain your design choices.

Students must:

- Analyze whether the proposed architecture is appropriate for the task and dataset size
- Identify potential overfitting risks and evaluate the regularization strategies suggested
- Improve the design by applying dropout, batch normalization, and correct weight initialization

Assessment focuses on: architectural reasoning, regularization understanding, and critical evaluation of AI-generated designs

Assignment 2: AI-Based Optimization Strategy Analysis

Students prompt an AI tool: Compare Adam, SGD with momentum, and RMSProp for training a CNN on CIFAR-10 and recommend the best strategy.

Students must:

- Evaluate the correctness of convergence claims and learning rate recommendations
- Identify scenarios where each optimizer may fail or underperform
- Validate by running experiments and comparing loss curves against AI predictions

Assessment focuses on: optimizer understanding, experimental validation, and critical assessment of AI-generated recommendations

Assignment 3: AI-Generated Transfer Learning Pipeline Critique

Students prompt an AI tool: Create a transfer learning pipeline using ResNet-50 for a medical image classification task.

Students must:

- Assess whether the frozen vs fine-tuned layer choices are appropriate for the target domain
- Identify data leakage risks, class imbalance issues, or inappropriate pre-processing steps
- Improve the pipeline using proper train/val/test splits and domain-specific augmentation

Assessment focuses on: transfer learning understanding, pipeline design, and validation of AI-generated implementations



Theory of Computation with Compiler Design

Course Code: 26DS3505

Semester: V

Programs Offered: B.Tech. CSE (Data Science)

Course Type: Core

Credits: 3

L – T – P – J: 3 – 0 – 0 – 0

Prerequisites: NIL

Course Vision

To develop a strong theoretical foundation in formal languages, automata theory, and compiler design for analyzing computation models and designing efficient language processors.

Course Objectives

1. Introduce formal language theory and computational models.
2. Develop the ability to design and analyze finite automata and grammars.
3. Provide knowledge of regular and context-free languages.
4. Enable understanding of compiler phases and parsing techniques.
5. Apply theoretical concepts to practical compiler construction.

Course Outcomes (CO)

CO	Description	Blooms Level
C01	Understand formal languages, automata, and their representations.	L2
C02	Apply regular expressions and finite automata for language recognition.	L3
C03	Analyze context-free grammars, PDA, and Turing machines.	L4
C04	Understand compiler structure and design lexical analyzers.	L2
C05	Apply parsing techniques for syntax analysis in compilers.	L3

Broad Course Description

This course introduces the theoretical foundations of computation through automata, formal languages, and grammar theory. It explores deterministic and non-deterministic finite automata, regular expressions, and their equivalence.

The course further covers context-free grammars, pushdown automata, and Turing machines as models of computation. Additionally, it provides an introduction to compiler design, including lexical analysis and parsing techniques. Students gain both theoretical understanding and practical exposure to language processing systems.



CO-PO mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
CO1	2	1									1	2	
CO2	2	2			1						1	2	1
CO3	2	3			1						1	2	1
CO4	2	2	2	2	2						1	2	1
CO5	2	3	2	2	2						2	2	2
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module 1: Introduction to Formal Languages and Automata (9 Hours)

Introduction to formal languages and automata, Alphabets, strings, and languages, Deterministic Finite Automata (DFA): definition and representation, Non-Deterministic Finite Automata (NFA), Equivalence of DFA and NFA, Finite automata with ϵ -transitions

Module 2: Regular Expressions & Regular Languages (9 Hours)

Regular expressions: definition and operations, Conversion: Regular Expression $\rightarrow$ NFA $\rightarrow$ DFA, DFA minimization (basics of Myhill-Nerode theorem), Pumping Lemma for Regular Languages, Closure properties of regular languages

Module 3: Context-Free Grammars, PDA & Turing Machines (9 Hours)

Context-Free Grammar (CFG): definition and derivations, Parse trees and ambiguity, Normal forms (Chomsky Normal Form, Greibach Normal Form), Pushdown Automata (PDA), Relationship between CFG and PDA, Turing machine model, representation, language acceptability, design techniques, variants of Turing machines

Module 4: Introduction to Compiler Design & Lexical Analysis (9 Hours)

Overview of compiler structure, Phases of compiler: lexical, syntax, semantic, optimization, code generation, Lexical analysis: tokens, lexemes, patterns, Design of lexical analyzer, Role of finite automata, Tools: Lex/Flex (introduction).

Module 5: Syntax Analysis (Parsing Techniques) (9 Hours)

Role of parser in compiler, Top-down parsing: Recursive descent parser, Predictive parsing (LL(1)), Bottom-up parsing: Shift-reduce parsing, LR parsing (SLR concepts), Error handling in parsing.



AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

- Module-1: Students use AI tools to simulate DFA/NFA behaviour and visualize state transitions.
- Module-2: AI-based tools assist in converting regular expressions into automata and minimizing DFA.
- Module-3: Students explore grammar parsing and Turing machine simulations using intelligent platforms.
- Module-4: AI-assisted tools help design lexical analyzers and tokenize real programming languages.
- Module-5: Students use AI-based parser generators and visualization tools to understand LL and LR parsing.

Assessment Scheme

Criteria	Weightage of Marks
Assignments / Tests	20
Mid-Semester Exam	30
Hackathon/Technical Competition	10
End-Semester Exam	40
Total	100 Marks

Recommended Tools

- Programming Languages: C, C++, Python
- Tools: Lex/Flex, Yacc/Bison
- Platforms: JFLAP, Automata Simulator

Primary Textbooks

1. Introduction to Automata Theory, Languages, and Computation authored by John E. Hopcroft, Rajeev Motwani, and Jeffrey D. Ullman, 3rd Edition, published in the year 2006, Pearson Education
2. Compilers: Principles, Techniques, and Tools authored by Alfred V. Aho, Monica S. Lam, Ravi Sethi, and Jeffrey D. Ullman, 2nd Edition, published in the year 2006, Pearson Education.

Additional Resources

- NPTEL – Theory of Computation
- MIT OpenCourseWare – Automata Theory
- GeeksforGeeks, TutorialsPoint



Design Studio– III: Innovation, Design Thinking & Intellectual Property

Course Code: 26DS3506

SEMESTER – V

Programs Offered: For all branches

Course Type: Core

Credits: 02

L–T–P–J: 1–0–0–2

Prerequisites: Basic Problem Solving, Creativity, Communication Skills

Course Vision

This course empowers students to become innovative problem-solvers by integrating Design Thinking, creativity, rapid prototyping, and Intellectual Property awareness to develop user-centric and ethically responsible solutions for real-world challenges.

Course Learning Objectives

At the end of the course students will be able to:

1. Understand the principles and stages of Design Thinking and innovation processes.
2. Apply user-centric approaches to identify, analyze, and solve real-world problems.
3. Generate innovative ideas and develop prototypes using modern design and AI-assisted tools.
4. Understand Intellectual Property Rights (IPR), patent processes, and innovation protection mechanisms.
5. Develop collaborative, interdisciplinary, and entrepreneurial problem-solving skills.

Broad Course Description

This course introduces students to the principles and practices of Innovation, Design Thinking, and Intellectual Property with a strong emphasis on user-centric problem-solving and experiential learning. The course combines creativity, empathy-driven design methodologies, rapid prototyping, collaborative innovation, and AI-assisted ideation tools to help students solve real-world societal and technological challenges.

Students will engage in a semester-long integrated design project where they move through the complete innovation lifecycle: identifying problems, understanding user needs, ideating creative solutions, prototyping concepts, validating solutions through user testing, and finally understanding how innovations can be protected using Intellectual Property Rights (IPR). The curriculum integrates modern AI tools for brainstorming, design generation, sentiment analysis, wireframing, patent exploration, and prototype refinement while emphasizing “Authentic Intelligence” — the human ability to critically evaluate AI-generated outputs, ensure ethical design, avoid bias, and preserve originality.



The course fosters creativity, innovation, teamwork, communication, and entrepreneurial thinking, preparing students to become future-ready innovators and responsible technology creators.

Course Outcomes (CO)

CO	Course Outcomes (COs)	Bloom's Level
CO1	Apply Design Thinking methodologies to identify user needs and formulate problem statements.	L3
CO2	Generate innovative solutions using ideation, creativity, and prototyping techniques.	L3
CO3	Use modern design and AI-assisted tools for rapid prototyping and validation.	L3
CO4	Analyze user requirements and evaluate alternative design solutions critically	L4
CO5	Demonstrate collaborative innovation practices and present solutions with IPR awareness.	L3

CO-PO mapping

COs	Program Outcomes (POs)											Program Specific Outcomes (PSOs)	
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO 1	PSO 2
CO1	1	3	2	2	2	2	-	2	2	2	3	-	3
CO2	1	2	3	2	3	2	-	2	3	2	3	-	3
CO3	1	2	3	2	3	1	-	1	2	2	3	-	3
CO4	1	1	2	1	2	2	1	3	1	2	3	-	3
CO5	1	2	3	1	2	2	-	2	3	3	3	-	3
3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)													

Detailed Course Syllabus



This course follows a project-based learning model where each week consists of:

- **1-hour lecture** introducing concepts and methodologies.
- **2-hour studio/project task** for hands-on implementation.

Students will progressively build a single integrated innovation project across all 15 weeks.

Phase 1 — Foundations of Innovation & Design Thinking (Weeks 1–3)

Authentic Intelligence Layer

Before creating solutions, students must understand the importance of empathy, originality, ethical innovation, and critical thinking. AI-generated ideas must always be evaluated for relevance, inclusivity, practicality, and authenticity.

Week 1: Introduction to Innovation & Design Thinking

LECTURE (1 HR)

Introduction to innovation and human-centered design. Design Thinking stages: Empathize, Define, Ideate, Prototype, and Test. Types of innovation: product, service, process, and social innovation. Role of creativity in entrepreneurship and engineering.

PROJECT TASK (2 HRS)

Identify 3 campus or community problems through observation. Prepare an “Innovation Opportunity Map” highlighting user pain points, existing solutions, and improvement areas.

AI Integration

Use AI tools such as ChatGPT or Gemini for idea generation and problem exploration.

Authentic Intelligence Check

Compare AI-generated ideas with real observations. Identify at least 2 assumptions made by AI that do not reflect actual user experiences.

Week 2: Empathy & User Research

LECTURE (1 HR)

Understanding user empathy. User research methods: interviews, observation, surveys, shadowing, and contextual inquiry. Persona development and empathy mapping.

PROJECT TASK (2 HRS)

Conduct user interviews with at least 5 users related to the chosen problem domain. Create empathy maps and two detailed user personas.

AI Integration

Use AI-assisted sentiment analysis tools to summarize interview responses and identify emotional patterns.

Authentic Intelligence Check

Evaluate whether AI summaries missed emotional nuances, contextual details, or cultural aspects. Document human corrections

Week 3: Problem Definition & Design Challenges



LECTURE (1 HR)

Problem framing techniques. Writing effective problem statements and “How Might We” questions. Understanding constraints and opportunity spaces.

Develop formal problem statements and create at least 5 “How Might We” questions. Present problem framing to peers for feedback.

AI Integration

Use AI tools to refine problem statements and identify possible opportunity areas.

Authentic Intelligence Check

Critically evaluate whether AI-generated problem statements are generic or truly user-centric.

Phase 2 — Ideation, Creativity & Concept Development (Weeks 4–7)

Authentic Intelligence Layer

Innovation requires originality and contextual thinking. AI tools can assist creativity, but students must critically examine whether generated ideas are feasible, ethical, and genuinely innovative.

Week 4: Ideation & Creative Thinking Techniques

LECTURE (1 HR)

Brainstorming methods, SCAMPER technique, mind mapping, lateral thinking, and collaborative ideation frameworks.

PROJECT TASK (2 HRS)

Generate 50+ ideas for the selected problem using brainstorming and AI-assisted ideation. Cluster ideas and shortlist top 3 concepts.

AI Integration

Use Generative AI tools for concept expansion and alternate solution generation.

Authentic Intelligence Check

Identify duplicated or derivative AI-generated ideas and refine them into original concepts.

Week 5: Concept Evaluation & Feasibility Analysis

LECTURE (1 HR)

Concept screening, feasibility analysis, value proposition, innovation matrix, and stakeholder mapping.

PROJECT TASK (2 HRS)

Evaluate shortlisted concepts using feasibility, desirability, and viability criteria. Select one concept for prototype development.

AI Integration

Use AI-based comparative analysis tools to evaluate solution alternatives.

Authentic Intelligence Check

Assess whether AI recommendations align with actual user needs and project constraints.

Week 6: Storyboarding & User Journey Mapping



LECTURE (1 HR)

User journey mapping, service blueprints, experience design, and storytelling for innovation.

PROJECT TASK (2 HRS)

Develop a visual storyboard and end-to-end user journey for the selected solution.

AI Integration

Use AI-assisted design tools for creating visual journey maps and flow diagrams.

Authentic Intelligence Check

Verify whether generated flows consider accessibility, inclusivity, and real user behavior.

Week 7: Low-Fidelity Prototyping

LECTURE (1 HR)

Low-fidelity prototyping methods: paper prototypes, sketches, wireframes, and MVP concepts.

PROJECT TASK (2 HRS)

Create low-fidelity prototypes using paper sketches, Figma, Canva, or Balsamiq.

AI Integration

Use AI-enabled UI generators to create alternative layouts.

Authentic Intelligence Check

Evaluate AI-generated designs for usability, accessibility, and cultural appropriateness.

Phase 3 — Prototyping, Validation & Innovation Development (Weeks 8–12)

Authentic Intelligence Layer

AI-assisted design and prototyping tools improve speed, but human validation is essential to ensure functionality, ethics, usability, and innovation authenticity.

Week 8: Digital Prototyping & UI/UX Design

LECTURE (1 HR)

High-fidelity prototyping, UI/UX principles, design systems, accessibility standards, and interaction design.

PROJECT TASK (2 HRS)

Develop digital prototypes using Figma, Adobe XD, or Canva. Include navigation flows and interactive elements.

AI Integration

Use AI-assisted UI/UX tools for color palettes, layouts, and component generation.

Authentic Intelligence Check

Audit generated interfaces for accessibility issues and inclusive design practices.

Week 9: Backend Logic & API Integration with AI [Coding]

LECTURE (1 HR)

PROJECT TASK (2 HRS)



Usability testing techniques, heuristic evaluation, feedback collection, and iterative improvement.

Conduct usability testing with at least 5 users. Document observations, pain points, and improvement opportunities.

AI Integration

Use AI tools to summarize user feedback and detect common patterns.

Authentic Intelligence Check

Compare AI-generated feedback summaries with real human observations.

Week 10: Innovation for Entrepreneurship

LECTURE (1 HR)

Innovation ecosystems, startup mindset, business model basics, value proposition canvas, and entrepreneurial innovation.

Develop a basic business model canvas for the proposed solution.

AI Integration

Use AI tools for market analysis and competitor research.

Authentic Intelligence Check

Validate whether AI-generated market insights are realistic and locally relevant

Week 11: Intellectual Property Rights (IPR) Fundamentals

LECTURE (1 HR)

Introduction to Intellectual Property Rights: Patents, Copyrights, Trademarks, Trade Secrets, Industrial Designs, and Licensing.

PROJECT TASK (2 HRS)

Identify the appropriate IP protection strategy for the student project. Create an IP classification chart.

AI Integration

Use AI tools to explore patent databases and novelty analysis.

Authentic Intelligence Check

Verify originality manually and identify risks of plagiarism or copyright violations.

Week 12: Patent Search & Innovation Protection

LECTURE (1 HR)

Patent filing process, prior art search, patent databases, technology transfer, and startup innovation protection.

PROJECT TASK (2 HRS)

Conduct a prior art search using Google Patents and WIPO databases. Prepare a novelty assessment report.

AI Integration

Use AI-assisted search tools to identify related patents and innovation gaps.

Authentic Intelligence Check

Critically evaluate whether AI-identified patents are genuinely relevant to the proposed innovation.

Phase 4 — Integration, Presentation & Innovation Showcase (Weeks 13–15) **Authentic Intelligence Layer**



Innovative solutions must be socially responsible, ethical, and human-centered. Students must evaluate whether their solutions genuinely solve meaningful problems and can be responsibly deployed.

Week 13: Prototype Refinement & Innovation Strategy

LECTURE (1 HR)

Iterative design refinement, innovation strategy, scalability, sustainability, and ethical innovation.

PROJECT TASK (2 HRS)

Refine the prototype based on user feedback and prepare the final innovation strategy document.

AI Integration

Use AI-assisted analytics and visualization tools for refinement insights.

Authentic Intelligence Check

Identify possible ethical concerns, bias, misuse scenarios, and social impact.

Week 14: Innovation Pitch & Storytelling

LECTURE (1 HR)

Presentation skills, storytelling for innovation, pitch deck preparation, and communicating value propositions.

PROJECT TASK (2 HRS)

Prepare a complete innovation pitch including problem statement, research, prototype, impact, and IP strategy.

AI Integration

Use AI tools for presentation refinement and visual enhancement.

Authentic Intelligence Check

Ensure the final pitch reflects genuine contributions and not only AI-generated content.

Week 15: Final Innovation Showcase & Reflection

LECTURE (1 HR)

Peer evaluation, reflective learning, ethical innovation review, and future roadmap planning.

PROJECT TASK (2 HRS)

Present the final prototype and innovation pitch in a 10-minute showcase. Submit a reflective report discussing learnings, challenges, and future improvements.

Mandatory Reflection Question

“Should this innovation be implemented in the real world? Why or why not?”

Students must justify: 1. Social impact and benefits. 2. Possible risks or misuse. 3. Ethical considerations. 4. Sustainability and scalability. 5. Human value beyond technology.



Laboratory / Studio Component (15-Week Activity Schedule)

- Week 1: Team formation and identification of innovation problem areas
- Week 2: Conduct user interviews and create empathy maps
- Week 3: Define problem statements and “How Might We” questions
- Week 4: Perform brainstorming and ideation sessions.
- Week 5: Evaluate concepts and select final solution
- Week 6: Create user journey maps and storyboards
- Week 7: Develop low-fidelity wireframes and prototypes
- Week 8: Build high-fidelity digital prototypes
- Week 9: Conduct usability testing and feedback analysis
- Week 10: Prepare business model and entrepreneurial strategy.
- Week 11: Explore Intellectual Property Rights applicable to the project
- Week 12: Conduct patent search and novelty analysis
- Week 13: Refine prototype and prepare innovation strategy
- Week 14: Prepare innovation pitch deck and presentation
- Week 15: Final innovation showcase and demonstration

Assessment Scheme (100% CIA) -Integrated Evaluation (Theory + Practical)

Component	Description	Weightage
Weekly Studio Activities	15 activity submissions and design exercises	40%
Mid-Term Review	Prototype and design review presentation	20%
Final Innovation Showcase	Final project demonstration and pitch	25%
Reflection & IP Report	Reflection document + Patent/IP analysis	15%

Tools & Technology Stack

Category	Tools
Design & Prototyping	Figma, Canva, Adobe XD, Balsamiq
Collaboration	Miro, FigJam, Google Forms
AI Tools	ChatGPT, Gemini, Claude, Ideation AI Tools
Prototyping Platforms	Tinkercad, Figma, Canva



Patent & IP Tools	Google Patents, WIPO Database
Documentation	Notion, Google Docs, GitHub

Text Books

1. Tim Brown – *Change by Design*
2. Jake Knapp – *Sprint: How to Solve Big Problems and Test New Ideas in Just Five Days*
3. WIPO – *Intellectual Property Handbook*
4. Don Norman – *The Design of Everyday Things*
5. Tom Kelley & David Kelley – *Creative Confidence*

Reference Books

1. IDEO Design Thinking Resources
2. Stanford d.school Materials
3. Innovation and Entrepreneurship – Peter Drucker
4. Design Thinking for Strategic Innovation – Idris Mootee
5. Human-Centered Design Toolkit – IDEO

E-Resources

1. WIPO Learning Resources – <https://www.wipo.int>
2. IDEO U Courses – <https://www.ideou.com>
3. Stanford d.school Resources – <https://dschool.stanford.edu/resources>
4. Google Patents – <https://patents.google.com>
5. Coursera / edX Innovation Courses

DLI Integration

DLI Learning Pathway: Building LLM Applications with Prompt Engineering

Students will use AI-enabled tools and Large Language Models (LLMs) for ideation, research synthesis, design generation, feedback analysis, and innovation storytelling while critically evaluating reliability, originality, and ethical implications.

Expected Learning Outcomes of the Studio

By the end of the course, students will:

- Develop empathy-driven innovation skills.
- Build user-centric prototypes for real-world challenges.
- Integrate creativity with technology and entrepreneurship.
- Understand the fundamentals of Intellectual Property protection.
- Demonstrate collaborative design and innovation practices.
- Critically evaluate AI-assisted innovation processes.



Sustainable Software Engineering

Course Code: 26DS3601

Semester: VI

Programs Offered: All CSE cluster

Course Type: Core

Credits: 2

L – T – P – J: 2 – 0 – 0 – 0

Prerequisite(s): NIL

Course Vision

To enable learners to design and develop energy-efficient, environmentally responsible, and scalable software systems by integrating sustainability principles into the software development lifecycle. The course aims to foster awareness of the environmental impact of computing and empower students to create innovative solutions that optimize resource usage, reduce carbon footprint, and support sustainable digital transformation across industries.

Course Objectives

1. To provide a strong foundation in the principles of sustainable and green software engineering.
2. To enable students to analyze the energy consumption and environmental impact of software systems.
3. To develop the ability to design and implement energy-efficient and resource-optimized software solutions.
4. To explore tools, metrics, and techniques for measuring and improving software sustainability.
5. To evaluate the trade-offs between performance, cost, and sustainability, and apply ethical practices in software development.

Course Outcomes (CO)

CO	Description	Blooms Level
C01	Explain the principles and importance of sustainable and green software engineering.	L2
C02	Analyze the energy consumption and environmental impact of software systems.	L4
C03	Design and develop software solutions that optimize resource usage and improve energy efficiency.	L6
C04	Apply tools and techniques to measure, monitor, and enhance software sustainability.	L3
C05	Evaluate trade-offs among performance, cost, and sustainability while addressing ethical and environmental considerations.	L5



Broad Course Description

This course provides a comprehensive understanding of how software systems impact energy consumption, resource utilization, and the environment, and how these impacts can be minimized through sustainable design and development practices. It introduces the principles of green computing and sustainable software engineering across the entire Software Development Life Cycle (SDLC), including requirements, design, implementation, testing, deployment, and maintenance.

Students will learn techniques to measure and analyze the energy efficiency and carbon footprint of software systems, along with strategies to optimize performance while reducing resource usage. The course covers energy-aware programming, efficient algorithms, cloud and distributed system optimization, and sustainable architecture patterns. It also explores modern approaches such as green AI, carbon-aware computing, and hardware–software co-design for energy efficiency.

In addition, the course examines tools and frameworks for monitoring sustainability metrics, and discusses real-world applications in data centers, cloud platforms, mobile computing, and large-scale software systems. Ethical considerations, regulatory aspects, and the role of software engineers in achieving global sustainability goals are emphasized. Through a combination of theory, case studies, and practical exercises, students will develop the ability to design, evaluate, and implement software solutions that balance performance, cost, and environmental impact, contributing to sustainable digital transformation.

CO PO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
CO1	2	1				2	1				1	1	
CO2	2	3			1	2	1				1	2	
CO3	2	3	2	3	2	2					2	2	
CO4	2	3	2	3	1	2	1			1	2	2	
CO5	2	3	2	2	2	2	1			1	2	2	1
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module-1 Introduction to Sustainable Software Engineering (6 hours)

Sustainability concepts, Triple Bottom Line (environmental, social, economic), Green computing fundamentals, Software lifecycle and sustainability, Carbon footprint of software, Energy consumption in IT systems.

MODULE-2 Energy-Efficient Software Design (6 Hours)

Energy-aware programming, efficient algorithms and data structures, Resource optimization, Memory and CPU utilization, Sustainable architecture patterns, Low-power design principles.

Module-3 Sustainable Software Development Lifecycle (SSDLC) (6 Hours)

Sustainability in SDLC phases, Requirements engineering for sustainability, Green software design principles, Sustainable testing practices, Continuous integration & deployment (CI/CD)



Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112
with sustainability focus.

Module-4 Software Sustainability Metrics & Evaluation (6 Hours)

Energy metrics, Carbon emission measurement, Performance vs energy trade-offs, Software sustainability indicators, Monitoring tools, Lifecycle assessment (LCA).

Module-5 Green AI and Energy-Efficient Systems (6 Hours)

Green computing principles, Carbon footprint of software systems, Energy-aware software design, Green AI concepts, Measuring energy consumption in software, Hardware-software co-design for efficiency, GPU acceleration and energy efficiency trade-offs, Model optimization techniques (pruning, quantization, distillation), Efficient training strategies, Carbon-aware scheduling, Sustainable cloud computing, Lifecycle assessment of AI systems.

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module-1: Foundations of Sustainable Software Engineering

Students utilize AI-driven analytics tools and platforms (Python/Colab, energy profiling libraries) to analyze the environmental impact of software systems. AI-based dashboards are used to visualize energy consumption patterns and carbon footprint, enabling learners to understand sustainability metrics and software lifecycle impacts.

Module-2: Energy-Efficient Software Design and Development

Students employ AI-assisted coding tools to design optimized algorithms and energy-efficient code. Tools such as static analyzers and AI-based code optimizers help identify inefficient patterns, while simulation tools are used to compare performance and energy trade-offs in different software designs.

Module-3: Sustainable Computing in Cloud and Distributed Systems

Students use AI-enabled cloud platforms to monitor resource utilization and optimize workloads. AI tools assist in workload scheduling, virtualization efficiency, and carbon-aware computing strategies, helping students design scalable and energy-efficient distributed systems.

Module-4: Monitoring, Metrics, and Sustainable Practices

Students leverage AI frameworks to evaluate and optimize machine learning models for energy efficiency (e.g., model pruning, quantization). AI tools are used to measure training energy consumption and improve model performance while minimizing computational cost and environmental impact.

Module-5: Green AI and Data-Intensive Applications

Advanced AI platforms such as digital twins, robotics simulation tools, physics-informed AI models, and edge AI systems are used to simulate and analyze real-world engineering systems. Students explore applications in robotics, aerospace, and smart manufacturing.



Assessment Scheme

Criteria	Weightage of Marks
Assignment/Project	15
Class Tests / Quizzes	10
Mid-Semester Examination	30
AI Mini-Project (Group)	5
End-Semester Examination	40
Total	100 marks

Recommended Tools

Primary Textbooks

1. Calero, C., & Piattini, M. (Eds.). (2021). Software Sustainability. Springer Nature. <https://doi.org/10.1007/978-3-030-69970-3>
2. Dick, M., Naumann, S., & Kuhn, N. (Eds.). (2023). Sustainable Software Engineering: Principles and Practices. Springer. <https://link.springer.com/book/10.1007/978-3-031-25695-0>
3. Scuri, S. (2024). Green Computing: Techniques and Architectures for Sustainable Software. Academic Press.
4. Larsson, S. (2022). The Green Tech Guide: Energy-Aware Programming for a Sustainable Future. O'Reilly Media.
5. Penzenstadler, B., & Betz, S. (2023). Requirement Engineering for Sustainability: A Guide for Practitioners. CRC Press.
6. Schwartz, J. (2025). Cloud-Native Sustainability: Optimizing Data Centers and AI with Green Architecture. Wiley.
7. Hussain, F., & Al-Turjman, F. (2024). Green AI: Hardware-Software Co-Design for the Next Generation of Intelligence. Elsevier

Additional Resources

Additional AI & Computational Resources

1. Green Software Foundation (GSF). (n.d.). The Green Software Training Course. <https://learn.greensoftware.foundation/>
2. Green Software Foundation. (2024). Software Carbon Intensity (SCI) Specification. <https://grnsft.org/sci>
3. Greenly. (2026). The Guide to Sustainable Software Development & Green IT. <https://www.greenly.earth/>
4. Principles. Green. (n.d.). The 8 Principles of Green Software Engineering. <https://principles.green/>



5. CodeCarbon-Python package to track CO2 emissions of AI model training. <https://codecarbon.io/>
6. Cloud Carbon Footprint-Multi-cloud (AWS/Azure/GCP) energy and carbon monitoring. <https://www.cloudcarbonfootprint.org/>
7. ScaphandreMetrology agent for monitoring energy consumption of processes. <https://github.com/hubblo-org/scaphandre>
8. Green Metrics ToolComprehensive measurement of energy metrics for software products. <https://green-coding.io/>

AI-Powered Assignments

Assignment 1: AI-Based Energy Consumption Analysis of Software

Students prompt an AI tool: Analyze the energy consumption of a given software application and suggest improvements.

Students must:

Evaluate the correctness of AI-generated analysis using performance and energy metrics.

Identify missing factors (hardware usage, workload variability, background processes).

Propose optimized solutions using energy-efficient coding practices.

Assessment focuses on understanding energy metrics and validating AI outputs

Assignment 2: AI-Assisted Code Optimization for Sustainability

Students prompt an AI tool: Optimize a given program to reduce execution time and energy consumption.

Students must:

Analyze AI-generated optimized code for correctness and efficiency

Identify trade-offs between performance and readability

Compare original and optimized code using benchmarking tools

Assessment focuses on coding efficiency and performance evaluation

Assignment 3: AI-Based Sustainable System Design

Students prompt an AI tool: Design a sustainable architecture for a cloud-based application.

Students must:

Evaluate AI-suggested architecture (resource allocation, scalability, energy use)

Identify missing considerations (cost, redundancy, carbon footprint)

Improve the design using green computing principles

Assessment focuses on system design and sustainability awareness.

Assignment 4: AI-Assisted Green AI Model Optimization

Students prompt an AI tool: Suggest methods to reduce energy consumption in training a machine learning model.

Students must:

Analyze AI recommendations (pruning, quantization, efficient architectures)

Validate feasibility using experiments or case studies

Propose an optimized model with reduced computational cost

Assessment focuses on AI model efficiency and sustainability.



Assignment 5 AI-Based Carbon Footprint Estimation

Students prompt an AI tool: Estimate the carbon footprint of a software system or data center workload.

Students must:

Verify AI-generated estimates using standard metrics

Identify assumptions and limitations in calculations

Suggest strategies to reduce carbon emissions

Assessment focuses on environmental impact analysis and critical thinking.

Assignment 6: AI-Assisted Sustainable Software Lifecycle Evaluation

Students prompt an AI tool: Evaluate sustainability practices across the software development lifecycle (SDLC).

Students must:

Analyze AI recommendations for each SDLC phase

Identify gaps in sustainability considerations

Propose improvements to make the lifecycle more sustainable

Assessment focuses on holistic understanding and application of sustainable practices

Mini Project Component

Industry-Driven Projects:

Project 1 – Energy-Efficient Data Processing Pipeline

Problem: Design a data processing pipeline that minimizes energy consumption for large datasets

Concepts Used: Data preprocessing, parallel processing, resource optimization

Industrial Relevance: Big data platforms (Hadoop, Spark), cloud analytics systems

Project 2 – Carbon Footprint Analysis of Machine Learning Models

Problem: Measure and reduce the carbon footprint of training ML models

Concepts Used: Green AI, energy metrics, model optimization (pruning, quantization)

Industrial Relevance: AI companies, research labs, sustainable AI initiatives

Project 3 – Sustainable Recommendation System

Problem: Develop a recommendation system optimized for both accuracy and energy efficiency

Concepts Used: Collaborative filtering, model optimization, performance trade-offs

Industrial Relevance: E-commerce, streaming platforms, personalization systems

Project 4 – AI-Based Workload Optimization in Cloud Systems

Problem: Optimize scheduling of data science workloads to reduce energy usage in cloud environments

Concepts Used: Cloud computing, resource allocation, AI-based scheduling

Industrial Relevance: Cloud service providers (AWS, Azure), enterprise IT systems

Project 5 – Efficient Data Storage and Retrieval System

Problem: Design a data storage system that reduces energy consumption while maintaining performance

Concepts Used: Data structures, indexing, compression techniques

Industrial Relevance: Data centers, database systems, enterprise storage solutions



CUDA for High Performance Computing

Course Code: 26DS3602

Semester: VI

Programs Offered: B. Tech CSE (Data Science)

Course Type: Core

Credits: 4

L – T – P – J: 3 – 0 – 2 – 0

Prerequisite(s): C/C++ programming, basic computer architecture

Course Vision

To develop proficiency with the knowledge and practical skills to harness GPU-based parallel computing using CUDA. Enabling the design and development of high-performance, scalable, and energy-efficient solutions for complex computational problems across scientific, engineering, and data-intensive domains computing solutions to real-world problems.

Course Objectives

1. Develop a strong foundation in GPU architecture and parallel computing concepts, enabling learners to understand how CUDA leverages massive parallelism for high-performance execution compared to traditional CPU-based systems.
2. Empower students with practical programming skills in CUDA using C/C++, including writing kernels, managing threads and blocks, and effectively utilizing different GPU memory hierarchies.
3. Enable the design and optimization of parallel algorithms, focusing on performance improvement techniques such as memory coalescing, shared memory usage, synchronization, and minimizing latency.
4. Provide hands-on experience in implementing real-world high-performance applications, such as numerical computations, image processing, and AI/ML workloads using GPU acceleration.
5. Build the ability to analyze, evaluate, and optimize computational performance, using profiling tools and benchmarking techniques to achieve scalable and efficient solutions.

Course Outcomes (CO)

CO	Description	Blooms Level
C01	Analyze GPU architecture and the CUDA programming model, including threads, blocks, grids, and execution hierarchy, to differentiate GPU and CPU computing for high-performance applications.	L4
C02	Develop CUDA programs using kernel functions and memory management techniques, including device-host interaction, synchronization, and debugging, to implement basic parallel algorithms.	L6
C03	Apply CUDA memory hierarchy concepts and optimization strategies such as shared memory usage, coalesced access, and occupancy tuning to improve the performance of data-parallel applications.	L3



C04	Implement advanced CUDA features and libraries like streams, multi-GPU programming, cuBLAS, cuDNN, Thrust to design scalable and efficient solutions for real-time and AI-driven applications.	L3
C05	Evaluate and optimize CUDA-based applications for AI/ML workflows, using profiling and benchmarking tools (e.g., NVIDIA Nsight) to achieve high performance in real-world scenarios and deployments.	L5

Broad Course Description

This course provides a comprehensive introduction to GPU-based parallel computing using NVIDIA's CUDA platform, focusing on the design, development, and optimization of high-performance applications. It begins with foundational concepts of high-performance computing, including the differences between CPU and GPU architectures and the evolution of parallel processing. Students are introduced to the CUDA programming model, execution hierarchy, and kernel-based computation. The course emphasizes hands-on programming using CUDA C/C++, covering thread organization, memory management, synchronization, and debugging techniques. Learners explore the GPU memory hierarchy and apply optimization strategies such as coalesced memory access, shared memory utilization, and performance tuning using profiling tools.

Advanced topics include asynchronous execution, CUDA streams, multi-GPU programming, and the use of optimized libraries such as cuBLAS, cuDNN, and Thrust. The course also integrates AI/ML workflows, demonstrating how GPU acceleration enhances deep learning, data processing, and real-time systems. Through practical exercises and case studies, students gain the ability to analyze performance bottlenecks, apply optimization techniques, and develop scalable, efficient CUDA-based solutions for scientific computing, data analytics, and artificial intelligence applications.

CO PO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
C01	2	2			2						1	2	2
C02	2	3	3	2	3						1	3	2
C03	2	3	3	2	3						2	3	2
C04	2	3	3	3	3						2	3	3
C05	2	3	3	3	3						2	3	3
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module-1 INTRODUCTION TO GPU COMPUTING AND CUDA

(09 Hours)

History of supercomputing, Overview of High-Performance Computing (HPC), Understanding Parallelism with GPUs, CPU vs GPU architecture, Evolution of GPU computing, CUDA platform and ecosystem, CUDA programming model, Threads, blocks, grids, Kernel functions, Execution hierarchy, CUDA runtime API, Compilation using NVCC.



Module-2 CUDA PROGRAMMING FUNDAMENTALS

(09 Hours)

Structure of a CUDA program, Writing basic CUDA kernels Compilation using NVCC, CUDA kernel programming, Thread indexing, Memory allocation (cudaMalloc, cudaMemcpy), SIMT (Single Instruction Multiple Threads), model Device vs host memory, Synchronization (syncthreads()), Error handling, Debugging CUDA programs – Vector addition using CUDA

Matrix addition, Basic parallel loop implementation, Performance comparison: CPU vs GPU Introduction to CUDA streams.

Module-3 MEMORY HANDLING WITH CUDA AND OPTIMIZATION

(09 Hours)

Memory hierarchy: Global, shared, constant, and texture memory, Memory Access Optimization Techniques– Coalesced memory access, Bank conflicts, Shared memory optimization, Register usage, Performance Optimization Strategies– Occupancy and performance tuning, Profiling tools (Nsight), Practical Applications–Matrix multiplication optimization using shared memory, Image filtering using texture memory, Reduction algorithms optimization, Real-time data processing using optimized memory usage.

Module-4 CUDA LIBRARIES AND SDK

(09 Hours)

CUDA Libraries Overview, Importance of GPU-accelerated libraries, NPP (Nvidia Performance Primitives), Introduction to- cuBLAS (Basic Linear Algebra Subprograms), cuFFT (Fast Fourier Transform), cuDNN (Deep Neural Networks), Thrust (Parallel algorithms library), CUDA Computing SDK, Integration of libraries in CUDA programs, API usage and examples.

Module-5 GPU DEPLOYMENT, MODEL OPTIMIZATION AND EDGE AI

(09 Hours)

AI Model Deployment-From training to inference pipeline, Challenges in deploying ML/DL models, GPU role in inference acceleration, NVIDIA Inference Optimization Tools, TensorRT for high-performance inference, Edge AI and Embedded Deployment- edge computing, Deployment on NVIDIA Jetson platform, Resource-constrained optimization techniques, Model conversion and optimization workflow, Scalable Deployment Strategies-Cloud-based GPU deployment, Containerization (Docker basics for GPU workloads).

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module 1: This module reflects AI integration by establishing the foundational shift from serial processing to the massive parallelism required to train modern neural networks, enabling students to bridge the gap between CPU-based logic and GPU-accelerated AI workloads.

Module 2: This module reflects AI integration by providing the technical expertise to write and execute parallel kernels, forming the basis for high-speed tensor operations and vector transformations that underpin all deep learning frameworks.

Module 3: This module reflects AI integration by mastering memory hierarchy and coalescing, which are critical for handling the massive datasets (Big Data) used in AI training, ensuring that memory bottlenecks do not stifle the performance of complex AI models.



Module 4: This module reflects AI integration by leveraging specialized libraries like cuDNN and cuBLAS, which are the industry-standard "engines" used by PyTorch and TensorFlow to perform optimized convolutions and matrix math for deep neural networks.

Module 5: This module reflects AI integration by focusing on the end-to-end AI lifecycle, utilizing TensorRT and NVIDIA Jetson to transition from large-scale model training to efficient, real-world deployment at the edge and in the cloud.

Lab Experiments

No.	Experiment	Module
1	Install and configure CUDA Toolkit and verify GPU using device query	Module I
2	Write and execute first CUDA program (Hello GPU)	Module I
3	Compare CPU vs GPU execution for simple computations	Module I
4	Demonstrate thread, block, and grid indexing	Module I
5	Implement vector addition using CUDA kernels	Module II
6	Perform matrix addition on GPU	Module II
7	Implement matrix multiplication using CUDA	Module II
8	Demonstrate memory allocation using cudaMalloc and cudaMemcpy	Module II
9	Implement synchronization using __syncthreads()	Module II
10	Debug CUDA programs and handle runtime errors	Module II
11	Explore global, shared, and constant memory usage	Module III
12	Optimize matrix multiplication using shared memory	Module III
13	Analyze coalesced vs non-coalesced memory access	Module III
14	Measure performance using CUDA events	Module III
15	Perform kernel optimization (block size and occupancy tuning)	Module III
16	Implement parallel reduction algorithm	Module IV
17	Use CUDA streams for asynchronous execution	Module IV
18	Profile GPU applications using NVIDIA Nsight tools	Module IV
19	Integrate CUDA with Python using Numba / CuPy	Module IV
20	Accelerate a simple ML task using GPU (e.g., vector operations)	Module IV

AI-Integrated Laboratory Experiments

GPU-Accelerated Data Preprocessing using CuPy: Perform large-scale data normalization and transformation on GPU and compare with CPU-based preprocessing.

- Deep Learning Inference Acceleration using CUDA: Implement a simple neural network (e.g., image classifier) and measure inference speed improvement using GPU.



- Parallel Feature Extraction for Image Data: Use CUDA kernels to extract features (edges, filters) from images and analyze performance gains.
- Time-Series Analysis using GPU (cuFFT Integration): Apply Fast Fourier Transform on large time-series datasets for frequency analysis using GPU acceleration.
- AI Model Training Optimization using GPU Streams: Optimize batch training of a simple ML model using CUDA streams and evaluate performance improvements.

Assessment Scheme

Criteria	Weightage of Marks
Lab Reports (experiments)/Project	15
Class Tests / Quizzes	10
Mid-Semester Examination	30
AI Mini-Project (Group)	5
End-Semester Examination	40
Total	100 marks

Recommended Tools

- Programming: /C++,Python (optional)
- Libraries: CUDA Toolkit, cuBLAS, cuDNN, Thrust
- Platforms: NVIDIA GPU-enabled systems, Google Colab (GPU),NVIDIA Nsight Tools

Primary Textbooks

1. Cook, S. (2013). CUDA programming: A developer's guide to parallel computing with GPUs. Morgan Kaufmann.
2. Han, J., & Sharma, B. (2019). Learn CUDA programming: A beginner's guide to GPU programming and parallel computing with CUDA 10.x and C++. Packt Publishing.
3. Kirk, D. B., & Hwu, W. W. (2022). Programming massively parallel processors: A hands-on approach (4th Ed.). Morgan Kaufmann.
4. Sanders, J., & Kandrot, E. (2010). CUDA by example: An introduction to general-purpose GPU programming. Addison-Wesley Professional.
5. Wilt, N. (2013). The CUDA handbook: A comprehensive guide to GPU programming. Addison-Wesley Professional.

Additional Resources

1. NPTEL: GPU Computing and CUDA Programming – IIT Madras / IIT Delhi
2. Coursera: Introduction to Parallel Programming with CUDA by NVIDIA
3. Udemy: CUDA Programming Masterclass for High Performance Computing
4. NVIDIA Developer Zone: CUDA Toolkit Documentation and SDK Guides
5. GeeksforGeeks: CUDA Programming and Parallel Computing Tutorials



6. IEEE Transactions on Parallel and Distributed Systems
7. ACM Transactions on Architecture and Code Optimization (TACO)
8. https://learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+S-AC-04+V2
9. https://www.nvidia.com/en-us/training/self-paced-courses/?tab=generative-ai-llm§ion=self-paced-courses&utm_source=chatgpt.com

Mini Project Component

Project 1 – GPU-Accelerated Image Processing System

Problem: Develop a real-time image filtering system using CUDA
Concepts Used: CUDA kernels, shared memory, texture memory, parallel processing
Industrial Relevance: Computer vision applications (autonomous vehicles, surveillance systems)

Project 2 – High-Performance Matrix Computation Engine

Problem: Optimize matrix multiplication using CUDA for large datasets
Concepts Used: Thread hierarchy, shared memory optimization, coalesced access
Industrial Relevance: Scientific computing, engineering simulations, ML computations

Project 3 – Real-Time Video Analytics using GPU

Problem: Implement GPU-based object detection preprocessing pipeline
Concepts Used: CUDA streams, parallel execution, memory optimization
Industrial Relevance: Smart surveillance, traffic monitoring, security systems

Project 4 – GPU-Based Data Preprocessing Framework

Problem: Accelerate large-scale data cleaning and transformation
Concepts Used: CUDA kernels, memory management, parallel loops
Industrial Relevance: Big data analytics, ETL pipelines, data engineering

Project 5 – Deep Learning Acceleration using CUDA

Problem: Speed up neural network training using GPU parallelism
Concepts Used: SIMT model, CUDA libraries (cuDNN), optimization techniques
Industrial Relevance: AI systems, recommendation engines, NLP applications



Cloud Data Engineering

Course Code: 26DS3603

Semester: VI

Programs Offered: B.Tech CSE (Data Science)

Course Type: Core

Credits: 3

L – T – P – J: 3 – 0 – 0 – 0

Prerequisites: Big Data Analytics

Course Vision

To develop skilled data engineers capable of designing, building, and optimizing scalable, secure, and real-time data pipelines using AWS cloud technologies aligned with global industry and certification standards.

Course Objectives

1. To provide foundational knowledge of cloud-based data engineering using AWS core services.
2. To enable learners to design and manage scalable data lakes using Amazon S3.
3. To develop skills in building ETL pipelines using AWS Glue for batch data processing.
4. To introduce real-time data streaming concepts using Amazon Kinesis services.
5. To equip students with the ability to design end-to-end data pipelines incorporating data warehousing, security, monitoring, and optimization using AWS services.

Course Outcomes (CO)

CO	Description	Blooms Level
CO1	Understanding cloud data engineering concepts and architecture.	L2
CO2	Apply scalable batch and streaming data pipelines.	L3
CO3	Apply cloud services for data storage and processing.	L3
CO4	Design the data transformation and integration	L6
CO5	Analyze performance, cost, and reliability of data systems, and use AI tools for automating data engineering workflows	L4

Broad Course Description

The course “AWS Data Engineer: S3, Glue, Kinesis & Redshift” is designed to provide a comprehensive understanding of modern cloud-based data engineering practices using industry-leading services from Amazon Web Services. It aligns with globally recognized certification standards such as the AWS Certified Data Engineer – Associate and focuses on developing both theoretical knowledge and hands-on expertise in building scalable data solutions.

The course begins with foundational concepts of cloud data engineering, introducing learners to data lake architectures using Amazon S3, where they will explore storage optimization



techniques, data formats, and efficient data ingestion strategies. It then progresses to data integration and transformation using AWS Glue, enabling students to design serverless ETL pipelines, manage metadata through the Data Catalog, and perform large-scale data processing using Apache Spark.

A significant component of the course focuses on real-time data processing using Amazon Kinesis services, including Kinesis Data Streams and Firehose, allowing learners to build streaming pipelines for applications such as log processing, IoT data ingestion, and event-driven architectures. The course further explores data warehousing concepts through Amazon Redshift, covering data modelling, query optimization, and integration with data lakes for advanced analytics.

In the final phase, the course emphasizes the design and implementation of end-to-end data pipelines by integrating multiple AWS services. Students will gain practical experience in workflow orchestration, monitoring using CloudWatch, implementing security best practices with IAM and encryption, and optimizing system performance and cost.

CO-PO mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
C01	2	1			2						1	2	2
C02	2	3			3						1	3	2
C03	2	3			3						1	3	2
C04	2	3	3	2	3						2	3	3
C05	2	3	3	3	3						2	3	3
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module-1: Foundations of Cloud Data Engineering (9 hours)

Introduction to Cloud Data Engineering, AWS Global Infrastructure & IAM basics, Amazon S3 fundamentals, Buckets, objects, storage classes, Versioning, lifecycle policies, Data lake architecture (S3-based), Data formats: CSV, JSON, Parquet, ORC, Partitioning strategies and performance optimization, Data ingestion into S3.

Module-2: Data Integration & ETL using AWS Glue (9 Hours)

ETL concepts in cloud, AWS Glue architecture, Glue Crawlers & Data Catalog, Glue Jobs (Spark-based ETL), DynamicFrames vs DataFrames, Schema discovery & metadata management, Data transformation pipelines, Incremental processing (job bookmarks), Integration with S3 and Redshift.

Module-3: Real-Time Data Streaming with Kinesis (9 Hours)

Streaming data fundamentals, Amazon Kinesis ecosystem, Kinesis Data Streams (real-time ingestion) Kinesis Data Firehose (delivery to S3/Redshift), Kinesis Data Analytics (SQL/Flink processing), Producers and consumers, Stream processing with Lambda.



Module-4: Data Warehousing & Analytics with Redshift (9 Hours)

Data warehousing concepts, Amazon Redshift architecture, Cluster, nodes, distribution styles, Columnar storage & compression, Loading data (COPY from S3), Redshift Spectrum (query S3 data), Query optimization & performance tuning, Security & access control.

Module-5: End-to-End Data Pipelines, Security & Monitoring (9 Hours)

Designing end-to-end pipelines: S3 → Glue → Kinesis → Redshift, Workflow orchestration (Step Functions), Monitoring: CloudWatch logs & metrics, CloudTrail auditing, Security: IAM roles & policies, Encryption (KMS), Cost optimization & best practices.

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module-1: Foundations of Cloud Data Engineering

Students utilize AI-powered learning platforms and cloud simulators (e.g., AWS Educate, Google Colab, ChatGPT) to explore cloud computing concepts, analyze data lifecycle stages, and compare cloud service models (IaaS, PaaS, SaaS). AI-based assistants are used to generate architectural diagrams and evaluate cloud platform selection based on use cases.

Module-2: Cloud Data Storage & Organization

Students use AI-enabled cloud monitoring and analytics tools to evaluate storage performance, optimize storage selection (object, block, file), and simulate data redundancy and replication strategies. AI-driven tools assist in automating metadata extraction, data classification, and lifecycle policy recommendations for efficient storage management.

Module-3: Data Processing & Pipeline Engineering

Students implement AI-assisted data pipelines using platforms like Apache Spark, Kafka, and Python (Colab). AI tools help in automating ETL/ELT workflows, anomaly detection in streaming data, and optimizing data ingestion strategies. Intelligent assistants are used to design scalable pipelines and analyze performance bottlenecks.

Module-4: AWS Data Engineering Services & System Design

Students leverage AI-powered AWS tools and services (AWS Glue, Lambda, Kinesis, Redshift) to build and manage data pipelines. AI-based cloud monitoring (CloudWatch insights) is used for system performance analysis, anomaly detection, and cost optimization. Students also use AI assistants to design secure and fault-tolerant architectures aligned with AWS best practices.

Module-5: Advanced AWS Data Engineering & Certification Readiness

Students apply AI tools to design event-driven architectures, perform real-time analytics, and optimize data workflows using AWS services (Athena, QuickSight, EventBridge). AI-driven platforms assist in certification preparation through adaptive learning, practice tests, and scenario-based problem solving. AI is also used to automate CI/CD pipelines and improve decision-making in cloud data engineering solutions.



Assessment Scheme

Criteria	Weightage of Marks
Project	10
Class Tests /assignments	10
Mid-Semester Examination	30
Hackathon/Technical Competition/ Paper presentation/Global/DLI certifications	10
End-Semester Examination	40
Total	100 marks

Recommended Tools

Primary Textbooks

1. Wufka, M., & Canonico, M. (2026), Introduction to cloud computing. Springer Cham. <https://doi.org/10.1007/978-3-032-07151-4>
2. Mittal, M., Balas, V. E., Goyal, L. M., & Kumar, R. (Eds.). (2019). Big data processing using Spark in cloud. Springer Singapore. <https://doi.org/10.1007/978-981-13-0550-4>
3. Haines, S. (2022), Modern data engineering with Apache Spark: A hands-on guide for building mission-critical streaming applications. Apress. <https://doi.org/10.1007/978-1-4842-7452-1>
4. Kukreja, M. (2021), Data engineering with Apache Spark, Delta Lake, and Lakehouse: Create scalable data pipelines. Packt Publishing.

Additional Resources

AWS AI & Data Engineering Services

- AWS S3 Docs: <https://docs.aws.amazon.com/s3/>
- Data Lake Guide: <https://aws.amazon.com/big-data/datalakes-and-analytics/>
- Free Training: <https://explore.skillbuilder.aws/learn>
- <https://www.datameer.com/>

Big Data & Distributed Tools

- <https://spark.apache.org/>
- <https://kafka.apache.org/>
- <https://airflow.apache.org/>

Cloud Labs & Practice Platforms

- <https://aws.amazon.com/training/awsacademy/>
- <https://aws.amazon.com/education/awseducate/>
- <https://colab.research.google.com/>
- <https://www.kaggle.com/>

Monitoring & Optimization

- <https://aws.amazon.com/cloudwatch/>



AI-Powered Assignments

Assignment 1: AI-Based Cloud Architecture Design

Students prompt an AI tool: Design a scalable cloud data architecture for an e-commerce platform handling real-time and batch data.

Students must:

Analyze the correctness of the AI-generated architecture (data ingestion, storage, processing layers).

Identify missing components (security, fault tolerance, cost optimization).

Refine the architecture using AWS best practices (data lake, pipeline layering).

Assessment focuses on: system design skills, architectural understanding, and validation of AI outputs.

Assignment 2: AI-Assisted Data Storage Optimization

Students prompt an AI tool: Recommend optimal cloud storage solutions for large-scale video streaming data with cost constraints.

Students must:

Evaluate the suggested storage types (object, block, archive tiers).

Identify inefficiencies in cost or performance trade-offs.

Improve storage selection using lifecycle policies and redundancy strategies.

Assessment focuses on: storage optimization, decision-making, and critical evaluation

Assignment 3: AI-Driven Data Pipeline Development

Students prompt an AI tool: Generate a data pipeline for ingesting, processing, and storing IoT sensor data in real time.

Students must:

Analyze pipeline components (Kafka, Spark, ETL/ELT stages).

Detect bottlenecks or scalability issues.

Enhance pipeline design for fault tolerance and performance.

Assessment focuses on: pipeline design, scalability analysis, and AI validation.

Assignment 4: AI-Based AWS Service Selection

Students prompt an AI tool: Suggest appropriate AWS services for building a real-time analytics system.

Students must:

Evaluate the relevance of recommended services (Kinesis, Lambda, Glue, Redshift).

Identify incorrect or redundant service usage.

Optimize service selection based on workload and architecture.

Assessment focuses on: AWS knowledge, service mapping, and architectural reasoning.

Assignment 5 AI-Assisted Performance & Cost Optimization

Students prompt an AI tool: Optimize a cloud data pipeline for performance and cost efficiency.

Students must:

Analyze AI recommendations (auto-scaling, partitioning, caching)

Identify unrealistic assumptions or missing metrics

Propose improved optimization strategies using monitoring insights



Assessment focuses on: performance tuning, cost analysis, and critical thinking.

Assignment 6: AI-Based Cloud Security & Governance Evaluation

Students prompt an AI tool: Design a secure cloud data engineering solution for handling sensitive healthcare data.

Students must:

Evaluate security measures (IAM roles, encryption, access control)

Identify gaps in compliance, governance, or data privacy

Improve the design using best practices for secure cloud systems

Assessment focuses on: security awareness, governance understanding, and validation of AI outputs.

Mini Project Component

Industry-Driven Projects:

Project 1 – AI-Based Cloud Data Architecture for E-Commerce Analytics

Problem: Design a scalable cloud data architecture for real-time and batch analytics in an e-commerce platform.

Concepts Used: Data lifecycle, cloud architecture, AWS services (S3, Lambda, Redshift), ETL pipelines.

Industrial Relevance: Amazon, Flipkart, Walmart data platforms for customer behaviour analytics.

Project 2 – Intelligent Data Lake Design Using AWS

Problem: Build an optimized data lake for storing structured and unstructured enterprise data

Concepts Used: Data lakes, object storage, metadata management, data partitioning

Industrial Relevance: Banking and telecom sectors for centralized data storage systems

Project 3 – Real-Time Streaming Data Pipeline for IoT Systems

Problem: Develop a real-time data pipeline for processing IoT sensor data

Concepts Used: Apache Kafka, Spark Streaming, stream processing, data ingestion

Industrial Relevance: Smart cities, industrial IoT monitoring, predictive maintenance systems

Project 4 – AWS-Based Scalable ETL Pipeline for Healthcare Data

Problem: Design an ETL pipeline for processing large-scale healthcare records

Concepts Used: AWS Glue, ETL vs ELT, data transformation, cloud data processing

Industrial Relevance: Healthcare analytics platforms, hospital data management systems

Project 5 – Serverless Data Engineering Using AWS Lambda

Problem: Create a serverless data processing system for event-driven applications

Concepts Used: AWS Lambda, event-driven architecture, API integration, automation

Industrial Relevance: Real-time notifications, fintech transaction systems



Computer Networks

Course Code: 26DS3604

Semester: VI

Programs: B. Tech CSE (Data Science)

Course Type: Core

Credits: 3

L – T – P – J: 3 – 0 – 0 – 0

Prerequisite: NIL

Course Vision

To impart comprehensive knowledge of computer network fundamentals, including protocol layering, transport mechanisms, routing strategies, and data communication techniques, along with an introduction to distributed computing paradigms. The course aims to cultivate the ability to evaluate network performance, ensure quality of service, and design robust distributed systems using modern tools and technologies, thereby preparing students to address contemporary challenges in networking and large-scale system design.

Course Objectives

1. To understand the fundamentals of data communication, network architectures, protocol layering, and application layer protocols such as HTTP, FTP, DNS, and email services.
2. To analyze the transport layer mechanisms including TCP and UDP, focusing on connection management, flow control, congestion control, and Quality of Service (QoS).
3. To develop an understanding of network layer concepts such as IP addressing, subnetting, IPv4/IPv6, and routing algorithms including distance vector, link state, and path vector protocols.
4. To study data link and physical layer functionalities including framing, error control, MAC protocols (Ethernet, WLAN), transmission media, and switching techniques.
5. To introduce the concepts of distributed computing, including system models, communication techniques (RPC, message passing), middleware, and challenges in distributed systems.

Course Outcomes (CO)

CO	Description	Blooms Level
CO1	Understand basic concepts of computer networks and application layer protocols.	L2
CO2	Explain the working of transport layer protocols like TCP and UDP.	L2
CO3	Apply IP addressing and routing concepts in networking.	L3
CO4	Analyze data transmission in data link and physical layers.	L4
CO5	Describe basic concepts of distributed systems and communication methods.	L2



Broad Course Description

This course provides a fundamental understanding of computer networks and distributed computing systems. It introduces the concepts of data communication, network architectures, and protocol layering through models such as OSI and TCP/IP. Students learn about various network layers including application, transport, network, data link, and physical layers, along with their respective protocols like HTTP, TCP, IP, and Ethernet.

The course also covers important topics such as routing, congestion control, Quality of Service (QoS), and network performance. In addition, it introduces the basics of distributed systems, including communication models, system types, and middleware. Overall, the course equips students with the knowledge required to analyze, design, and understand modern networked and distributed systems used in real-world applications.

CO PO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
C01	2	1			1						1	2	1
C02	2	2			1						1	2	1
C03	2	3			2						1	2	1
C04	2	3	2	2	2						2	2	2
C05	2	2	2	2	1						2	2	2
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module - 1: Introduction Computer network And Application Layer (10 Hours)

Data Communication -Networks & Network Types-Protocol Layering-TCP/IP Protocol Suite-OSI Model- Introduction to Sockets, Application Layer Protocols: HTTP- FTP- Email (SMTP, POP3, IMAP)-DNS-SNMP.

Module - 2: Transport Layer & QoS (9 Hours)

Transport Layer Introduction: UDP -TCP: Connection Management-Flow Control-Congestion Control- Congestion Avoidance (DECbit, RED)-SCTP, Quality of Service (QoS).

Module - 3: Network Layer and Routing (8 Hours)

Packet Switching, Internet Protocol: IPv4 -IP Addressing & Subnetting-IPv6, Network Layer Protocols: ARP, RARP, ICMP, DHCP, Routing: Unicast Routing, Distance Vector (RIP), Link State (OSPF), Path Vector (BGP), Multicast Routing (DVMRP, PIM)

Module - 4: Data Link and Physical Layer (9 Hours)

Data Link Layer: Framing-Flow Control-Error Control, Protocols: HDLC- PPP, Media Access Control: Ethernet Basics, CSMA/CD- Virtual LAN-Wireless LAN (802.11), Physical Layer: Data & Signals-Performance Metrics - Transmission Media - Switching (Circuit Switching)



Module - 5: Introduction to Distributed Computing

(9 Hours)

Introduction to Distributed Systems, Goals and Characteristics, Types of Distributed Systems (Client-Server, Peer-to-Peer), Advantages and Challenges, Basic Communication Models (Message Passing, RPC), Introduction to Middleware.

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module-1: Introduction and Application Layer

Students utilize AI-powered tools such as ChatGPT and Wireshark to simulate application layer protocols including HTTP, FTP, SMTP, DNS, and SNMP. AI-assisted visualization tools are used to demonstrate OSI and TCP/IP models, socket programming, and real-time client-server communication. Students analyze packet flows and protocol behaviour using both simulated and real network data.

Module-2: Transport Layer and QoS

Students use AI-supported simulation platforms such as MATLAB and Python-based environments to model TCP/UDP communication, congestion control mechanisms, and QoS techniques. AI tools help visualize flow control, congestion avoidance (RED, DECBT), and SCTP behaviour. Machine learning concepts are introduced to predict network congestion patterns and optimize performance.

Module-3: Network Layer and Routing

Students employ AI-integrated network simulation tools like Cisco Packet Tracer to design and analyze IP addressing, subnetting, and routing protocols such as RIP, OSPF, and BGP. AI models assist in route optimization, traffic prediction, and dynamic network configuration. Students simulate real-world routing scenarios and evaluate efficiency.

Module-4: Data Link and Physical Layer

Students apply AI-based simulation tools to study framing, error control, and flow control mechanisms. AI visualization is used to demonstrate Ethernet operations, CSMA/CD, VLANs, and wireless LAN (802.11). At the physical layer, AI models simulate signal transmission, noise effects, and switching techniques, enabling students to analyze performance metrics and communication reliability.

Module-5: Introduction to Distributed Computing

Students use advanced AI platforms and distributed system simulators to design and analyze client-server and peer-to-peer architectures. AI tools assist in modeling communication mechanisms such as message passing and RPC, and in simulating middleware behaviour. Students explore AI-driven optimization of distributed systems, including scalability, fault tolerance, and resource allocation in real-world applications.

Authentic Intelligence Reflection:

Module 1: Students critically evaluate AI-generated protocol simulations against real packet-level data and textbook concepts, identifying gaps between theoretical and practical networking behaviour.



Module 2: Students compare AI-predicted congestion and QoS performance with standard TCP models, reflecting on the reliability and limitations of AI in dynamic network environments.

Module 3: Students analyze differences between AI-generated routing decisions and traditional routing algorithms, assessing the effectiveness of AI in network optimization.

Module 4: Students reflect on the ability of AI models to represent real-world physical layer phenomena such as noise, interference, and transmission errors, identifying limitations.

Module 5: Students evaluate AI-generated distributed system designs against real-world constraints such as scalability, latency, and fault tolerance, emphasizing the role of human judgment in system design.

Assessment Scheme based on the course

Criteria	Weightage of Marks
Lab Reports (experiments)/Project	10
Class Tests /assignments	10
Mid-Semester Examination	30
Hackathon/Technical Competition/ Paper presentation/Global/DLI certifications	10
End-Semester Examination	40
Total	100 marks

Recommended Tools

Primary Textbooks

1. James F. Kurose, Keith W. Ross, Computer Networking, A Top-Down Approach Featuring the Internet, Eighth Edition, Pearson Education, 2021.
2. Behrouz A. Forouzan, Data Communications and Networking with TCP/IP Protocol Suite, Sixth Edition TMH, 2022
3. Andrew S. Tanenbaum, Maarten van Steen - Distributed Systems: Principles and Paradigms, 2nd Edition, Prentice-Hall, 2006.
4. Andrew S. Tanenbaum, David.J. Wetherall, - Computer Networks||, Prentice-Hall, 6th Edition, 2022.
5. Andrew S. Tanenbaum, Maarten van Steen - Distributed Systems: Principles and Paradigms, 2nd Edition, Prentice-Hall, 2006.



Additional Resources

Additional AI & Computational Resources

1. NPTEL – IIT-based courses on Computer Networks & Distributed Systems
2. Coursera – Industry-oriented networking and cloud courses
3. edX – University-certified programs on networking concepts
4. Cisco Networking Academy – Practical labs and certification courses
5. Virtual Labs (IIT) – Online networking experiments
6. IEEE – Research papers and standards in networking
7. ACM Digital Library – Journals and conference publications
8. arXiv – Latest research preprints in networking and AI
9. Wireshark – Packet-level analysis and protocol learning
10. Cisco Packet Tracer – Network design and simulation tool

Project Component

Industry-Driven Projects:

Project 1 – AI-Based Network Traffic Anomaly Detection

Problem: Detect abnormal traffic patterns and potential cyber-attacks in a network

Concepts Used: Packet analysis, TCP/IP, anomaly detection, machine learning

Industrial Relevance: Cybersecurity systems (Cisco, Palo Alto Networks), intrusion detection systems

Project 2 – Intelligent Routing Optimization using AI

Problem: Optimize routing paths dynamically based on network traffic conditions

Concepts Used: Routing algorithms (RIP, OSPF, BGP), graph theory, AI optimization

Industrial Relevance: Internet service providers, cloud networking (Google, Amazon Web Services)

Project 3 – Network Congestion Prediction System

Problem: Predict congestion in a network and adjust flow control dynamically

Concepts Used: TCP congestion control, QoS, data analytics, machine learning

Industrial Relevance: Telecom networks, streaming services (Netflix, Akamai Technologies)

Project 4 – Distributed File Sharing System

Problem: Design a peer-to-peer file sharing system with fault tolerance

Concepts Used: Distributed systems, P2P architecture, RPC, consistency models

Industrial Relevance: Cloud storage systems (Dropbox, Google Drive)

AI-Powered Assignments

Assignment 1 – AI-Assisted Protocol Analysis

- Task: Use AI tools to analyze HTTP, DNS, and SMTP protocol behaviour
- Concepts Used: Application layer protocols, TCP/IP, packet structure
- AI Tools: ChatGPT, Wireshark
- Outcome: Students compare AI explanations with real packet captures



Assignment 2 – TCP Congestion Control Simulation using AI

- Task: Simulate TCP congestion control mechanisms and analyze performance
- Concepts Used: Flow control, congestion control, QoS
- AI Tools: Python, Google Colab
- Outcome: Predict congestion patterns using AI-based models

Assignment 3 – AI-Based Subnetting and IP Allocation

- Task: Use AI tools to generate subnetting schemes for given network requirements
- Concepts Used: IPv4 addressing, subnetting, network design
- AI Tools: ChatGPT, Python
- Outcome: Validate AI-generated subnetting with manual calculations

Assignment 4 – Intelligent Routing Analysis

- Task: Compare AI-generated routing decisions with traditional algorithms
- Concepts Used: RIP, OSPF, BGP, routing optimization
- AI Tools: Cisco Packet Tracer, Python
- Outcome: Evaluate efficiency of AI in routing optimization



Entrepreneurial Finance & Deep-Tech Start-ups

Course Code: 26DS4701

Semester – VII

Programs: For all branches

Course Type: core

Credits: 02

L–T–P–J: 2–0–0–0

Prerequisites: Basics of Economics, AI/ML Fundamentals, Data Science

Course Learning Objectives

1. Understand fundamentals of entrepreneurial finance and start-up ecosystems.
2. Analyze valuation methods and funding strategies for AI-driven start-ups.
3. Apply AI tools for financial modelling, forecasting, and market analysis.
4. Develop skills to design, pitch, and scale deep-tech ventures.
5. Evaluate ethical, regulatory, and societal implications of AI entrepreneurship.

Broad Course Description

Entrepreneurial Finance & AI-Driven Deep-Tech Start-ups is an interdisciplinary course that integrates financial literacy, entrepreneurship, and advanced technology domains such as Artificial Intelligence and deep-tech innovation. The course equips students with the knowledge and skills required to build, evaluate, and scale start-ups in emerging technology sectors.

Students will explore start-up ecosystems, financial planning, valuation techniques, and funding mechanisms including venture capital, angel investment, and government grants. The course emphasizes AI-driven decision-making for market analysis, forecasting, and business strategy development. It also introduces business models tailored for AI and deep-tech domains such as robotics, IoT, and quantum technologies.

Learners will engage in hands-on activities including financial modelling, pitch deck creation, and start-up simulations using modern tools like Excel, Power BI, and AI platforms. Ethical considerations, legal frameworks, intellectual property rights, and sustainability in AI-driven entrepreneurship are integral components of the course.

By the end of the course, students will be capable of conceptualizing, validating, and presenting scalable and responsible deep-tech start-up ideas.

Course Outcomes (CO)

CO	Description	Bloom's Level
CO1	Apply financial concepts such as cash flow, unit economics, and valuation to evaluate start-up performance.	L3



C02	Design AI-driven business models for deep-tech ventures including AI, robotics, IoT, and quantum technologies.	L3
C03	Analyze start-up funding strategies including venture capital, angel investment, and government funding schemes.	L4
C04	Develop financial projections, pitch decks, and capitalization tables for scalable start-up ventures.	L6
C05	Build scalable and ethical AI start-ups aligned with legal frameworks, IPR, and responsible innovation principles.	L3

CO-PO mapping

CO	Program Outcomes (POs)												
COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
C01	3	2	2	2	2				3	3	3	3	2
C02	3	3	2	2	3	2			3	3	3	3	2
C03	3	3	2	3	3	2			3	3	3	3	2
C04	2	3	2	3	3	2			3	3	3	3	2
C05	2	3	2	3	3	2			3	3	3	3	2

Course Modules

UNIT – I Foundations of Entrepreneurial Finance

(06 Hours)

Topics: Start-up ecosystem overview, types of start-ups (SaaS, AI, Deep-Tech), financial statements (Profit & Loss, Balance Sheet, Cash Flow), unit economics (Customer Acquisition Cost, Lifetime Value), break-even analysis.

UNIT – II Start-up Valuation & Funding Strategies

(06 Hours)

Topics: Start-up valuation methods (DCF, Comparable, Scorecard), capitalization tables, equity vs debt financing, funding stages (Seed, Series A/B/C), venture capital, angel investors, government funding schemes.

UNIT – III AI-Driven Business Models & Deep-Tech Ventures

(06 Hours)

Topics: AI business models (platform, API-based, subscription), deep-tech domains (AI, robotics,



IoT, quantum computing), product-market fit, competitive analysis, monetization strategies.

UNIT – IV Financial Planning, Forecasting & Scaling

(06 Hours)

Topics: Financial projections, burn rate, runway, scenario planning, scaling strategies, growth metrics, exit strategies (IPO, M&A).

UNIT – V Legal, Ethical & Future of AI Entrepreneurship

(06 Hours)

Topics: Start-up legal frameworks, intellectual property (patents, copyrights), AI regulations, data privacy laws, ESG principles, future trends in AI entrepreneurship.

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

- AI-Augmented Financial Analysis: Use AI tools for financial dashboards, forecasting, and automated analysis.
- AI-Assisted Start-up Ideation: Leverage generative AI for idea validation, market research, and MVP design.
- Human vs AI Decision Making: Compare human intuition vs AI-driven insights in financial and strategic decisions.
- Data-Driven Entrepreneurship: Use AI tools for market intelligence and predictive analytics.
- Ethical AI Entrepreneurship: Address bias, fairness, and responsible innovation in start-up ecosystems.
- Experiential Learning: Build real-world AI start-up projects with AI-assisted tools.

Applications:

1. AI-based start-up financial planning
2. Venture capital investment analysis
3. Deep-tech innovation ecosystems
4. FinTech and AI-driven business models
5. Start-up valuation and fundraising
6. Smart city and healthcare start-up solutions

Mini Project Component

Industry-Driven Projects

Project 1 – AI Startup Financial Model

Problem: Build financial projections for an AI-based startup.

Concepts Used: Unit economics, forecasting, valuation.

Industry Relevance: Startup financial planning.

Project 2 – Deep-Tech Startup Pitch

Problem: Develop and pitch a scalable AI startup idea.

Concepts Used: Business model, funding strategy, AI integration.

Industry Relevance: Venture capital ecosystem.



Assessment Scheme based on the course

Criteria	Weightage of Marks
Continuous Internal Assessment (CIA) – Formative	60
Semester End Examination (SEE) – Summative	40
Total	100 marks

Recommended Tools

- Excel, Power BI, Tableau
- Python, SQL
- AI Tools (ChatGPT, AutoML, forecasting tools)
- Startup tools (Pitch deck, Cap table tools)

TEXT BOOKS

1. Brad Feld & Jason Mendelson – Venture Deals.
2. Eric Ries – The Lean Start-up.

REFERENCE BOOKS

1. Ash Maurya – Running Lean.
2. Scott Kupor – Secrets of Sand Hill Road.
3. Harvard Business Review – Entrepreneurship Articles.
4. NASSCOM & Start-up India Reports.

E-RESOURCES

1. Coursera – Entrepreneurship & Finance.
2. edX – Start-up & Innovation Courses.
3. NPTEL – Entrepreneurship.

Activity Based Learning (Suggested Activities in Class)

1. Start-up Pitch Simulation
2. Financial Modelling Workshop
3. Case Study Discussions (AI Start-ups)
4. Ethical AI Entrepreneurship Debate



Principles of DevOps & MLOps

Course Code: 26DS4702

Semester: VII

Programs Offered: B. Tech CSE (Data Science)

Course Type: Core

Credits: 4

L – T – P – J: 3 – 0 – 2 – 0

Prerequisite(s): Software Engineering fundamentals, Data Science basics

Course Vision

To equip students with modern software engineering practices by integrating DevOps and MLOps principles, enabling them to design, deploy, and manage scalable, automated, and intelligent systems for real-world applications.

Course Objectives

1. Introduce DevOps culture, tools, and lifecycle practices.
2. Develop skills in continuous integration and deployment pipelines.
3. Provide knowledge of containerization, orchestration, and cloud-native systems.
4. Enable understanding of MLOps lifecycle and ML system deployment.
5. Equip students with Infrastructure as Code (IaC) for scalable and reproducible environments.

Course Outcomes (CO)

CO	Description	Blooms Level
C01	Understand DevOps principles, lifecycle, and version control systems.	L3
C02	Design and implement CI/CD pipelines using modern tools.	L6
C03	Apply containerization, orchestration, and microservices architecture.	L3
C04	Analyze and implement MLOps pipelines for ML lifecycle management.	L4
C05	Design automated infrastructure and deployment using IaC tools.	L6

Broad Course Description

The Principles of DevOps & MLOps course introduces students to modern software development and deployment practices that bridge development, operations, and machine learning workflows. It covers DevOps culture, CI/CD pipelines, containerization, and orchestration using industry-standard tools. The course extends into MLOps by addressing machine learning lifecycle management, model deployment, monitoring, and continuous training. Students also explore Infrastructure as Code (IaC) for automated environment provisioning and reproducibility. The



course emphasizes hands-on learning using tools like Git, Docker, Kubernetes, Jenkins, and MLflow, preparing students for cloud-native and AI-driven software engineering roles.

CO-PO Mapping Table

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
CO1	2	1			1						1	2	1
CO2	2	3	3	2	3						1	3	2
CO3	2	3	3	2	3						2	3	2
CO4	2	3	3	3	3						2	3	3
CO5	2	3	3	3	3						2	3	3
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module 1: Introduction to DevOps & Version Control (9 Hours)

Evolution of software development practices, DevOps principles and culture, DevOps lifecycle and workflow, Agile, Scrum, and Lean practices, Version control systems (Git Commands), Introduction to DevOps tools and ecosystem.

Module 2: Containerization, Orchestration & Infrastructure (9 Hours)

Virtualization vs containerization, Docker architecture, images and containers, Writing Dockerfiles, Kubernetes orchestration, Scaling and load balancing, Microservices architecture.

Module 3: Continuous Integration & Continuous Deployment (CI/CD) (9 Hours)

CI/CD concepts and importance, Build automation and testing, CI/CD tools (Jenkins, GitHub Actions, GitLab CI), Automated testing frameworks, Pipeline as Code, Deployment strategies (Blue-Green, Canary releases).

Module 4: Introduction to MLOps & ML Lifecycle (9 Hours)

Introduction to MLOps, DevOps vs MLOps, ML lifecycle (data → model → deployment), Data and model versioning, Experiment tracking (MLflow), Reproducibility, Model deployment (REST APIs, batch), Flask/FastAPI, Docker for ML, Continuous Training pipelines, Model monitoring (drift detection), CI/CD for ML.

Module 5: MLFLOW DEPLOYMENT (9 Hours)

MLflow Model Deployment, Cloud Deployment options, Serving models using MLflow Models and REST APIs, Scoring and batch inference, MLflow Advanced Features, Hyperparameter tuning with MLflow, MLflow with Spark and Databricks, End-to-End MLOps Pipeline with MLflow, Building an end-to-end ML pipeline using MLflow, Automating model training, evaluation, and deployment, Best practices for MLflow in production environments.



AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module-1:

Students use AI-powered tools such as GitHub Copilot for intelligent code suggestions, commit generation, and repository management. AI-assisted analytics help monitor team collaboration, workflow efficiency, and project progress in Agile environments. Students also explore automated code review, documentation generation, and smart version control practices.

Module-2:

Students integrate AI-enabled CI/CD tools such as Jenkins and GitHub Actions for build, testing, and deployment automation. AI tools help predict build failures, optimize test execution, and automate security scanning in DevOps pipelines. Intelligent dashboards are used to monitor deployment performance and improve software delivery reliability.

Module-3:

Students use AI-assisted monitoring with Docker and Kubernetes to optimize container deployment and orchestration. AI systems analyze logs, resource utilization, and traffic patterns for autoscaling and fault prediction. Students also study AI-driven optimization for microservices and cloud-native application performance.

Module-4:

Students use MLflow for AI-assisted experiment tracking, model versioning, and deployment management. AI tools support automated model monitoring, drift detection, and retraining workflows for reliable ML systems. Students deploy models using APIs and monitor prediction performance through intelligent

Module-5:

Students build scalable ML pipelines using MLflow integrated with AI-based cloud optimization tools. AI systems recommend infrastructure scaling, workload balancing, and deployment cost optimization strategies. Students automate training, evaluation, deployment, and monitoring processes for enterprise-grade MLOps environments.

List of Experiments

No.	Experiment	Module Link
1	To understand Git fundamentals, repository management, branching strategies, and collaborative development using GitHub/GitLab	Module 1
2	To design and implement a Continuous Integration pipeline using Jenkins.	Module 1



Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

3	To implement a complete CI/CD pipeline for automated deployment.	Module 2
4	To containerize applications using Docker and manage container networking.	Module 2
5	To deploy and manage containerized applications using Kubernetes.	Module 3
6	To automate infrastructure provisioning using Terraform.	Module 3
7	To automate server configuration and deployment using Ansible.	Module 4
8	To implement an end-to-end MLOps workflow for machine learning model development and deployment.	Module 4
9	To implement continuous training pipelines and monitor deployed ML models.	Module 5
10	To design and deploy a complete DevOps and MLOps integrated cloud-native application.	Module 5

Assessment Scheme

Criteria	Weightage of Marks
Class Tests	10
Mid-Semester Examination	30
Mini Project	10
Assignments	10
End-Semester Examination	40
Total	100 marks



Recommended Tools

- Version Control: Git, GitHub
- CI/CD: Jenkins, GitHub Actions, GitLab CI
- Containerization: Docker
- Orchestration: Kubernetes
- MLOps Tools: MLflow, Kubeflow
- IaC Tools: Terraform, Ansible

Primary Textbooks

1. Gene Kim, Jez Humble, Patrick Debois, and John Willis, *The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations*, IT Revolution Press, 2021.
2. Kief Morris, *Infrastructure as Code: Managing Servers in the Cloud*, O'Reilly Media, 2020.
3. Mark Treveil and Alok Shukla, *Introducing MLOps: How to Scale Machine Learning in the Enterprise*, O'Reilly Media, 2020.

Reference Books

1. Jez Humble and David Farley, *Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation*, Pearson Education, 2011.
2. Sam Newman, *Building Microservices*, O'Reilly Media, 2021.
3. Bass, Weber, and Zhu, *DevOps: A Software Architect's Perspective*, Addison-Wesley Professional, 2015.

Additional Resources

- NPTEL – DevOps and Cloud Computing
- Coursera – MLOps Specialization
- Kubernetes Documentation
- Docker Documentation
- MLflow Official Docs



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakagalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Mini Project Component

Project 1 – CI/CD Pipeline Automation

- Automate build, test, and deployment of a web application

Project 2 – Containerized Microservices Application

- Deploy scalable application using Docker & Kubernetes

Project 3 – End-to-End MLOps Pipeline

- Build ML pipeline with training, deployment, and monitoring

Project 4 – IaC-based Cloud Deployment

- Provision infrastructure using Terraform and deploy application



Leadership, Policy & Authentic Intelligence

Course Code: 26DS4801

Semester: VIII

Programs Offered: For all branches

Course Type: Core

Credits: 2

L – T – P – J: 2 – 0 – 0 – 0

Prerequisites: No prior policy or ethics coursework assumed; technical competence and comfort with ambiguity are essential.

Course Vision

This condensed 2-credit version of LPAI retains the full conceptual scope of the original 5-unit course while compressing delivery into 10 weeks and 30 contact hours. The three core lenses — technical leadership, technology policy, and AI ethics — are preserved and integrated from the outset, with a streamlined capstone replacing the extended Unit 5 sprint.

Course Objectives

- Understand the fundamentals of engineering and society leadership
- Apply tools of AI ethics, policy
- Develop governance frameworks
- Analyze ethical, societal, and reliability issues in AI systems

Course Outcomes (CO)

CO	Description	Blooms Level
CO1	Apply authentic intelligence, systems thinking, and leadership models in socio-technical contexts. (Module 1)	L3
CO2	Demonstrate communication and teamwork skills for presenting technical ideas to stakeholders. (Module 1)	L2
CO3	Implement AI policy frameworks and governance principles in real-world scenarios. (Module 2)	L3
CO4	Apply ethical principles to address bias, fairness, and accountability in AI systems. (Module 2)	L3
CO5	Develop solutions using leadership, policy, and ethical considerations for real-world challenges. (Module 3)	L6



CO-PO mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
C01	1	1				1	1	2	1	1	2	1	
C02	1	1						3	3	1	2	1	
C03	2	2				2	2			1	2	2	
C04	2	3				2	3				2	2	
C05	2	3	2	2		2	2		2	1	2	2	1
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Broad Course Description

This course introduces students to the rapidly evolving field of AI and ethics. It covers foundational concepts such as Policy, regulatory, and governance. Students will gain hands-on experience in analyzing and comparing various regulatory frameworks on AI applications. The course integrates NVIDIA DLI resources to expose students to Trustworthy AI workflows in deep learning. Ethical concerns such as bias, hallucination, and responsible AI deployment are also emphasized.

Course Modules

The course consists of three modules. Each module integrates:

- Core topics on leadership
- Governance, policy and ethics
- Case study
- Authentic Intelligence reflection

Module 1: Foundations + Leadership 10 Hours | Weeks 1-4

Topics covered

- What is authentic intelligence? Beyond artificial intelligence
- Engineering in society: roles, responsibilities, and power
- Systems thinking for engineers — feedback loops and stakeholder mapping



- History of technology governance and the leadership-policy-AI nexus
- Leadership models: transformational, servant, and distributed
- Decision-making under uncertainty and high-stakes environments
- Managing diverse, interdisciplinary teams
- Communicating technical ideas to non-technical stakeholders
- Ethical leadership: when to dissent, when to escalate

Teaching Methods

- Socratic seminar and concept mapping (Week 1)
- Guest talk: industry or government practitioner (Week 2)
- Leadership style self-assessment and small group case discussion (Week 3)

Project Deliverable

Deliverable: Project brief (individual) — 1-page problem framing with stakeholder map

Format: Individual · 1 page

Due: End of Week 4

Weight: 15% of final grade

Module 2: Policy, Governance & AI Ethics 10 Hours | Weeks 5–7

Topics covered

- How policy is made: legislative, regulatory, and standards processes
- Key frameworks: EU AI Act, India's DPDP Act 2023, IEEE standards
- Engineers as policy actors: testimony, consultation, design advocacy
- Comparative case: Aadhaar — governance lessons for engineers
- Public interest technology and responsible innovation
- Ethical theories applied to AI: consequentialism, deontology, virtue ethics
- Bias, fairness, and representation in algorithmic systems
- Authentic intelligence: human judgment augmented by AI
- Accountability gaps: who is responsible when AI fails?
- Designing for human dignity, autonomy, and informed consent

Teaching Methods

- Policy memo writing workshop and India AI regulatory landscape mapping (Week 5)
- Mock regulatory hearing and comparative analysis paired activity (Week 6)
- Ethics case carousel (3 cases, 3 groups) and bias detection exercise (Week 7)
- AI ethics canvas workshop and Socratic philosophical dialogue (Week 7)

Project Deliverable

Deliverable: Policy & ethics brief (team) — 2-page governance recommendation + 1-page ethics audit card

Format: Team · 3 pages total + verbal debrief

Due: End of Week 7

Weight: 20% of final grade



Module 3: Capstone: Integrated Design Challenge 10 Hours | Weeks 8–10

Topics covered

- Revisiting your Unit 1 problem brief — what has changed?
- Integrating leadership, policy, and ethics into one design response
- Stakeholder consultation and co-design methods
- Prototyping governance structures and technical safeguards
- Writing for public audiences: op-eds and policy explainers
- Structuring a policy pitch for decision-makers
- Visual communication: slides, infographics, and data stories
- Structured peer review and iteration under time pressure

Teaching Methods

- Problem reframing workshop and capstone kick-off Q&A (Week 8)
- Co-design simulation and structured peer consultation (Week 9)
- Pitch rehearsal and writing workshop (Week 9)
- Structured peer review with instructor micro-consultations (Week 10)
- Public showcase with external jury panel: industry, policy, and academia (Week 10)

Project Deliverable

Deliverable: Capstone portfolio (team) — 6-page written portfolio + 8-minute pitch to external Jury.

Individual component: 300-word individual reflection submitted alongside the team portfolio.

Due: End of Week 10 (final showcase)

Weight: 50% capstone + 15% individual reflection

10-Week Detailed Schedule

Each week = 3 contact hours. Total: 30 hours across 10 weeks.

Wk	Topic	Methods & Notes
1	What is authentic intelligence?	Icebreaker: 'AI decision you made today' · Concept mapping · Short provocation reading · Course overview: the leadership-policy-AI triangle
2	Systems thinking & the technology-society interface	Guest speaker: policy or industry practitioner · Stakeholder mapping workshop · Feedback loops and power mapping
3	Leadership models & self-awareness	Leadership style self-assessment · Small group case discussion · Cases: engineering leaders who shaped policy (Grace Hopper, Sundar Pichai)



4	Decision-making, teams & communication	Role-play simulation: high-stakes engineering decision · Peer feedback · When to dissent: ethical leadership and escalation · Project brief due
5	How policy is made	Policy memo writing workshop · Mapping India's AI regulatory landscape · Regulatory bodies vs. standards organisations (IEEE, ISO, BIS)
6	Comparative governance & public interest tech	Mock regulatory hearing · Comparative analysis: EU AI Act, US, India, China · Case: Aadhaar — governance lessons for engineers
7	AI ethics: frameworks, bias & human dignity	Ethics case carousel (3 cases, 3 groups) · Bias detection hands-on exercise · AI ethics canvas workshop · Socratic dialogue: where should humans always decide? · Policy & ethics brief due
8	Capstone launch & stakeholder consultation	Problem reframing workshop · Team formation or reconfirmation · Capstone kick-off Q&A · Co-design simulation · Draft capstone outline due (1 page)
9	Public communication & peer review	Writing workshop: op-ed drafting · Pitch rehearsal in pairs · Structured peer review of capstone drafts · Instructor micro-consultations · Full capstone draft due
10	Final showcase & course reflection	Final capstone presentations (8 min + 5 min Q&A per team) · External jury panel: industry/policy/academia · Individual reflection · Course close

DLI Resources as Case Studies

These are resources you can Analyze to understand how authentic intelligence principles play out in practice.

Case Study 1 — Deloitte + DLI (enterprise AI governance at scale)

NVIDIA collaborated with Deloitte to shape a DLI curriculum customised to the needs of AI professionals and projects, enabling Deloitte to tap into the power of their own NVIDIA DGX system to explore novel AI use cases. This is a case study in how a major consulting firm institutionalised AI literacy as a governance function — connecting technical training directly to client-facing AI deployments in regulated industry.

Case Study 2 — Booz Allen Hamilton + US Air Force

DLI teamed with Booz Allen Hamilton to train employees and government personnel, including members of the U.S. Air Force, providing hands-on training for data scientists to solve challenging



problems in healthcare, cybersecurity and defence. This illustrates authentic intelligence in high-stakes, mission-critical settings where explainability and accountability are non-negotiable — the military context demands AI that can be audited and overridden by humans.

Case Study 3 — KENET University Faculty Training (Global South capacity building)

The Fundamentals of Deep Learning course provided faculty from different KENET member universities a great opportunity to get up-to-date training. KENET is Kenya’s national research and education network. This is a case study in authentic intelligence as inclusion — building AI capacity in African universities so that the Global South participates in defining how AI works, not just consuming AI built elsewhere.

Case Study 4 — Generative AI Teaching Kit with Dartmouth (academic-industry co-creation)

The Generative AI Teaching Kit, co-developed with Assistant Professor Sam Raymond from Dartmouth College, explores practical implementation of Generative AI through LLMs, diffusion models, multimodal architectures, distributed training, and accelerating inference. The co-development model itself is a case study — when universities co-author industry training content, it creates a feedback loop where academic critical thinking (including ethics and bias analysis) shapes what engineers learn to build.

Case Study 5 — NeMo Guardrails as governance in code

The inclusion of NeMo Guardrails in the Generative AI Teaching Kit curriculum is a case study in authentic intelligence by design. Rather than treating safety as a policy document, NeMo Guardrails allows developers to write topical rails, dialogue flows, and output filters in configuration — making ethical constraints enforceable at the inference layer. Analysing how Guardrails works, what it can and cannot prevent, and who controls the rail definitions is a rigorous case study in the limits and possibilities of technical AI governance.

Assessment Framework

All assessments are project-based. No written exams. Work builds progressively toward the capstone.

Assessment	Unit	Format	Marks
Project brief	U1	Individual · 1 page	20
Policy & ethics brief	U2	Team · 2 pages	10
Capstone pitch	U3	Team · 8-min pitch + 6-page portfolio	40
DLI Certification (NVIDIA)	—	NVIDIA DLI certificate — 50% weightage considered for assessment	—
Individual reflection	U3	Individual · 200 words	30



Assessment Principles

- Each assessment is scaffolded — the project brief frames the policy brief, which frames the capstone.
- Team assessments use a peer contribution rubric; individual grades may be adjusted by $\pm 10\%$ based on peer feedback.
- The capstone is assessed by an external jury using a shared rubric covering technical rigour, policy coherence, and ethical consistency.

Teaching Strategies

Four evidence-based strategies selected for the compressed format. Each is designed to maximise engagement within 3-hour weekly sessions.

Think-Pair-Share

Activate thinking before full group discussion. Give students 60–90 seconds of silent thinking, 2 minutes of pair discussion, then invite pairs to share. Cold-call pairs rather than volunteers to reduce bias. Use a timer to keep it tight.

Socratic Seminar

Assign shared text or prompt in advance. Open with one big question that has no single right answer. Facilitate: redirect questions, do not answer them. Students build on, challenge, or question each other. Close with a written exit ticket. Arrange seating in a circle and norm: speak to the group, not just the teacher.

Jigsaw Activity

Divide complex content into 3–4 chunks. Assign each student to an expert group to study their chunk. Regroup so each new team has one expert per chunk; each expert teaches their piece. Keep expert groups small (3–5) and provide a clear ‘what to teach’ checklist.

Gallery Walk

Post work or prompts around the room. Students rotate in pairs (2–3 minutes per station), leaving a sticky note or written comment at each. Debrief by asking what patterns or surprises came up. Use a signal — bell or music — to rotate, and assign roles: reader, writer, timekeeper.

Assessment Scheme based on the course

Definitions

1. "Internal Assessment" refers to the Continuous Internal Assessment (CIA). The weightage of this component is 60%. It includes:
 - a. One Mid Semester Examination (MSE) (Descriptive-based),
 - b. Six Continuous Assessment Tool (CAT): Quiz, Project based learning, Reflection Note on Guest Lecture, Industrial visit, E-course certification, Building models, Group discussion, Case study, Seminar, or Open Book Assignment.
 - c. One AI-powered Assignment (APA): Use of AI tools for problem-solving while simultaneously developing Authentic Intelligence through critical evaluation, ethical reasoning, and responsible decision-making.



2. "External Assessment" refers to the Semester End Examinations (SEE). The weightage of this component is 40%. At the end of semester, the final examination is conducted to evaluate student's performance.
- If any theory course consists of Mini project component, then the total marks of 10 (Ten marks) shall be evaluated for that mini project and hence there is no APA for such courses.
 - If any theory course does not consist of mini project, then APA is mandatory awarded for 10 marks.

Criteria	Weightage of Marks
Continuous Internal Assessment (CIA) – Formative	60
Semester End Examination (SEE) – Summative	40
Total	100 marks

Recommended Tools

- Programming: Python
- Visualization: Transformers, LangChain
- Platforms: NVIDIA DLI, Google Colab
- AI Tools: CUDA, RAPIDS

Additional Resources

NVIDIA Learning Resources

- NVIDIA Deep Learning Institute (DLI)
- Generative AI Learning Path

Recommended Courses

- Getting Started with Deep Learning
- Accelerating Data Science Workflows

CUDA Python Fundamentals



Skill Enhancement Courses

Java Programming

Course Code: 26DS2308

Semester: III

Programs: B. Tech. CSE(Data Science)

Course Type: Core - Skill Enhancement Course I

Credits: 2

L – T – P – J: 1 – 0 – 2 – 0

Prerequisite: Basic Programming (C/Python)

Course Vision

To equip students with strong practical skills in Java programming by emphasizing object-oriented design, logical problem-solving, and real-world application development, enabling them to build efficient, scalable, and user-centric software systems aligned with modern industry and AI-driven requirements.

Course Objectives

1. To understand the fundamentals of Java programming, including syntax, data types, operators, and control structures.
2. To develop problem-solving skills using arrays and methods in Java.
3. To apply object-oriented programming concepts such as classes, inheritance, polymorphism, abstraction, and interfaces.
4. To implement exception handling mechanisms and understand the use of packages and access control in Java applications.
5. To perform file handling operations and utilize collection frameworks for efficient data management.

Course Outcomes (CO)

CO	Description	Blooms Level
CO1	Understand and apply fundamental concepts of Java programming including data types, operators, and control structures.	L2
CO2	Develop Java programs using arrays and methods to solve basic computational problems.	L6
CO3	Design and implement object-oriented solutions using classes, inheritance, polymorphism, abstraction, and interfaces.	L6
CO4	Apply exception handling mechanisms and organize code using packages and access modifiers.	L3
CO5	Perform file handling operations and utilize collection frameworks for efficient data storage and manipulation.	L6



Broad Course Description

This course introduces the fundamentals of Java programming and provides a strong foundation in problem-solving and object-oriented design. It covers essential concepts such as Java syntax, data types, control structures, arrays, and methods, enabling students to develop basic computational solutions. The course emphasizes object-oriented programming principles including classes, inheritance, polymorphism, encapsulation, abstraction, and interfaces for building modular and reusable applications. It also focuses on exception handling, package organization, and access control mechanisms to ensure robust and maintainable code. Further, students gain practical experience in file handling and the use of collection frameworks for efficient data storage and manipulation. Overall, the course equips learners with the necessary skills to design, implement, and manage Java-based applications in real-world scenarios.

Course Modules

Module 1: Introduction to Java

(6 Hours)

Java basics, JVM, JDK, JRE, Data types, Variables, Operators, Control statements

Module 2: Arrays and Basic Programming

(7 Hours)

Arrays (1D and 2D), Array operations, Introduction to methods, Basic problem-solving in Java

Module 3: Object-Oriented Programming

(6 Hours)

Classes and Objects, Constructors, Method overloading, Inheritance, Polymorphism, Encapsulation, Abstraction, Interfaces

Module 4: Exception Handling and Packages

(6 Hours)

Exception handling (try-catch, throw, throws), Built-in exceptions, User-defined exceptions, Packages, Access modifiers

Module 5: File Handling and Collections

(5 Hours)

File I/O, Streams, Collection framework (ArrayList, HashMap, Set), Iterators

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module 1: Introduction to Java

AI Integration: Introduction to AI concepts, Role of Java in AI applications, Overview of intelligent systems

Module 2: Arrays and Basic Programming

AI Integration: Problem-solving using AI thinking, Introduction to algorithmic logic used in AI, Simple data handling for AI applications

Module 3: Object-Oriented Programming

AI Integration: Designing AI systems using OOP concepts, Object modeling for intelligent agents, Reusability in AI-based applications



Module 4: Exception Handling and Packages

AI Integration: Error handling in AI systems, Managing exceptions in data-driven applications, Modular design for AI projects

Module 5: File Handling and Collections

AI Integration: Data handling for AI (datasets), Using collections for data storage and preprocessing, Reading and managing datasets for machine learning

Authentic Intelligence Reflection:

Module 1: Introduction to Java

Authentic Intelligence Reflection: Reflect on how Java enables real-world intelligent systems, compare human problem-solving vs. programmed logic, Identify ethical considerations in basic computing

Module 2: Arrays and Basic Programming

Authentic Intelligence Reflection: Analyze how data organization impacts decision-making, Reflect on efficiency of different problem-solving approaches, Relate structured data handling to real-life scenarios

Module 3: Object-Oriented Programming

Authentic Intelligence Reflection: Examine how OOP models real-world entities, Reflect on abstraction and its similarity to human thinking, Evaluate design choices in building intelligent systems

Module 4: Exception Handling and Packages

Authentic Intelligence Reflection: Reflect on handling unexpected situations in real life vs. programs, Analyze robustness and reliability in systems, Consider ethical implications of failure handling in software

Module 5: File Handling and Collections

Authentic Intelligence Reflection: Reflect on how data storage influences intelligent decision-making, Analyze importance of data integrity and privacy, Connect data handling practices to real-world AI applications

List of Experiments For lab integrated course

No.	Experiment	Module Link
1	Write a Java program to display “Hello World” and perform arithmetic operations	Module 1
2	Implement control statements (if-else, loops) and operators	Module 1
3	Implement arrays (1D/2D) and perform matrix operations	Module 2



4	Write programs using methods for basic problem-solving	Module 2
5	Create classes and objects with constructors and methods	Module 3
6	Demonstrate inheritance and polymorphism (overloading/overriding)	Module 3
7	Implement encapsulation and abstraction using classes and interfaces	Module 3
8	Handle built-in and user-defined exceptions using try-catch	Module 4
9	Create packages and demonstrate access control mechanisms	Module 4
10	Perform file handling using FileReader and FileWriter	Module 5
11	Implement collection framework using ArrayList, HashMap, and Set	Module 5
12	Use iterators to traverse collection elements	Module 5

Assessment Scheme based on the course

Criteria	Weightage of Marks
Lab Reports	10
Mini Project	50
Lab Internal Exam	30
Hackathon/Technical Competition/ Paper presentation/Global/DLI certifications	10
Total	100 marks

Recommended Tools

Primary Textbooks

1. Herbert Schildt, Java: The Complete Reference, 11th Edition, McGraw Hill Education, 2019.
2. Kathy Sierra and Bert Bates, Head First Java, 3rd Edition, O'Reilly Media, 2022.
3. Paul Deitel and Harvey Deitel, Java: How to Program, 11th Edition, Pearson Education, 2018.
4. Y. Daniel Liang, Introduction to Java Programming and Data Structures, 12th Edition, Pearson Education, 2019.
5. [Oracle Corporation – Official Java Documentation]
(https://docs.oracle.com/javase/?utm_source=chatgpt.com), Oracle Corporation, accessed 2026.



CO PO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PS01	PS02
C01	3	2			2							1	2	1
C02	3	3	2		2							1	3	2
C03	3	3	3	2	3				1	1		2	3	3
C04	2	3	2	2	3							2	2	3
C05	2	2	3	1	3				1	2		2	2	3
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)														

Additional Resources

Additional AI & Computational Resources

- Online Documentation:**
Official Java documentation by Oracle Corporation for detailed API references and programming guides.
- E-Learning Platforms:**
Courses on Coursera, Udemy, and edX for structured Java programming tutorials and certifications.
- Coding Practice Platforms:**
Practice Java programming on HackerRank, LeetCode, and CodeChef to enhance problem-solving skills.
- Development Tools:**
Use IDEs like IntelliJ IDEA and Eclipse IDE for efficient coding, debugging, and project development.
- YouTube Channels:**
Educational content from channels like freeCodeCamp and ProgrammingKnowledge for visual and practical learning.
- Books for Reference:**
Standard textbooks such as *Java: The Complete Reference* by Herbert Schildt and *Head First Java* by Kathy Sierra & Bert Bates for in-depth understanding.

Project Component

Industry-Driven Projects:

Project 1 – Smart Student Management System

- Problem: Develop a system to manage student records, attendance, and performance
- Concepts Used: OOP (classes, objects), file handling, collections
- Industrial Relevance: Academic ERP systems, educational software solutions



Project 2 – Banking Application with Transaction Management

- Problem: Build a secure banking system for account management and transactions
- Concepts Used: OOP, exception handling, file processing
- Industrial Relevance: FinTech applications, digital banking systems

Project 3 – GUI-Based Inventory Management System

- Problem: Design an application to track and manage product inventory
- Concepts Used: Java Swing/AWT, event handling, collections
- Industrial Relevance: Retail management, warehouse and supply chain systems

Project 4 – Log File Analyzer for System Monitoring

- Problem: Analyze system log files to detect errors and usage patterns
- Concepts Used: File handling, collections, string processing
- Industrial Relevance: IT operations, cybersecurity, system monitoring tools



Business Intelligence Tools (Power BI)

Course Code: 26DS2309

Semester: III

Programs: B. Tech CSE (Data Science)

Course Type: Skill Enhancement Course (SEC-I)

Credits: 2

L – T – P – J: 1 – 0 – 2 – 0

Prerequisite(s): Nil

Course Vision

To provide practical knowledge of Business Intelligence tools, enabling students to analyze data, build interactive dashboards, and support data-driven decision-making using Microsoft Power BI and Tableau.

Course Objectives

1. Understand fundamentals of Business Intelligence and the Power BI ecosystem
2. Perform data cleaning and transformation using Power Query in Power BI
3. Design and develop interactive dashboards and visualizations using Power BI and Tableau
4. Apply DAX functions, data modeling, and star schema concepts for business calculations
5. Deploy and share BI reports and dashboards for real-world decision-making applications

Course Outcomes (CO)

CO	Description	Blooms Level
CO1	Explain BI concepts, Power BI fundamentals, and data loading techniques	L2
CO2	Apply data transformation and cleaning techniques using Power Query	L3
CO3	Develop interactive dashboards and visualizations using Power BI and Tableau	L6
CO4	Implement data modeling, star schema design, and DAX calculations	L3
CO5	Deploy and share BI solutions for business reporting and decision-making	L3



Broad Course Description

Business Intelligence Tools (Power BI) is a 2-credit Skill/Practical Oriented course for CSE / AI / DS students in the 3rd Semester. It provides hands-on training in the complete BI workflow — from data ingestion and transformation to interactive dashboard design and deployment — using Microsoft Power BI and Tableau as primary platforms.

The course is structured across five modules: Power BI Data Preparation and Modeling, Tableau Visualization and Analytics, Advanced DAX and Power BI Service, Tableau Advanced Features, and BI Integration with Case Study and Mini Project. Laboratory sessions reinforce each module through hands-on exercises covering Power Query transformations, DAX measures, relationship modeling, interactive dashboards, and an end-to-end BI project.

CO-PO mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
C01	2	1			2						1	2	2
C02	2	3			3						1	3	2
C03	2	3	3	2	3						1	3	3
C04	2	3	3	2	3						2	3	3
C05	2	3	3	3	3					1	2	3	3
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module 1: Power BI – Data Preparation and Modeling (9 Hours)

Introduction to Business Intelligence (BI), Overview of Microsoft Power BI, Power BI components: Power BI Desktop and Power BI Service, Connecting to data sources (Excel, CSV, databases, web), Data loading techniques (Import vs DirectQuery), Data Transformation using Power Query, Data cleaning: removing duplicates, handling missing values, data type conversion, filtering, sorting, grouping, Data shaping and transformation, Data Modeling: tables, relationships, star schema basics, Creating and managing relationships, Calculated columns vs measures, Introduction to DAX (SUM, COUNT, CALCULATE)

Module 2: Tableau – Visualization and Analytics (9 Hours)

Introduction to Tableau, Tableau interface and components, Connecting to data sources, Data preparation in Tableau, Data blending and joins, Creating visualizations: bar charts, line charts, pie charts, maps, Filters and parameters, Dashboards and storyboards, Interactive visual analytics, Calculated fields, Level of Detail (LOD) expressions (basic), Formatting and design principles, Publishing and sharing dashboards



AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module 1: AI-assisted data profiling tools are used to identify data quality issues (missing values, duplicates, type mismatches) in raw datasets before Power Query transformation. Students critically evaluate AI-generated cleaning recommendations against Power Query best practices and validate data model integrity after transformation.

Module 2: Students use AI tools to generate initial Tableau dashboard layouts and chart type recommendations for a given dataset. AI-suggested visualizations are critically evaluated for design effectiveness, chart type appropriateness, and whether they support clear data storytelling and decision-oriented communication.

List of Experiments For lab integrated course

No.	Experiment	Module Link
1	Import data from Excel/CSV and perform basic cleaning using Power Query	Module 1
2	Connect to multiple data sources and perform data transformation in Power BI	Module 1
3	Create relationships between tables and implement a star schema in Power BI	Module 1
4	Develop a basic Power BI dashboard using charts, tables, and filters	Module 1
5	Implement slicers, KPI cards, and interactive visuals in Power BI	Module 1
6	Create DAX measures: SUM, COUNT, percentage change, Year-on-Year calculations	Module 1
7	Create visualizations in Tableau: bar chart, line chart, pie chart, and maps	Module 2
8	Apply filters, parameters, and calculated fields in Tableau	Module 2
9	Design interactive dashboards and storyboards in Tableau	Module 2
10	Mini Project: End-to-end dashboard (data preparation → visualization → insights) using Power BI or Tableau	Module 1&2



Assessment Scheme based on the course

Criteria	Weightage of Marks
Practical Experiments	30
Hackathon/Certificate	10
Lab Record	10
Project	50
Total	100 marks

Recommended Tools

Primary Textbooks

1. Microsoft Power BI Documentation – learn.microsoft.com/en-us/power-bi
2. Tableau Documentation and Learning Resources – help.tableau.com
3. Alberto Ferrari & Marco Russo, The Definitive Guide to DAX, Microsoft Press, 2nd Edition, 2019
4. Rutger Blom, Leila Etaati, Expert Data Modeling with Power BI, Packt Publishing, 2022
5. Evan Stubbs, Delivering Business Intelligence with Microsoft SQL Server, McGraw-Hill
6. Devin Knight et al., Microsoft Power BI: Business Intelligence Cookbook, Packt Publishing

Additional AI & Computational Resources

1. Microsoft Power BI Learning Path (free) – learn.microsoft.com/en-us/training/powerbi
2. Tableau Public Gallery – public.tableau.com
3. SQLBI DAX Resources – sqlbi.com
4. Guy in a Cube (Power BI YouTube tutorials) – youtube.com/c/GuyinaCube
5. Kaggle Business Datasets for BI Practice – kaggle.com/datasets

Mini Project Component

The following projects are suggested based on course outcomes. Students may attempt one project individually or in pairs.

Project 1 – Sales Performance Dashboard (Power BI)

Problem: Build an interactive Power BI dashboard analyzing sales performance across regions, products, and time periods using a retail dataset with at least 3 DAX measures and drill-through functionality

Concepts Used: Power Query transformation, star schema modeling, DAX measures (YoY growth, % share), slicers, KPI cards, drill-through reports

Industrial Relevance: Retail analytics, sales team reporting, executive decision dashboards



Project 2 – Customer Analytics Storyboard (Tableau)

Problem: Design a Tableau storyboard analyzing customer behaviour, purchase patterns, and churn indicators using a dataset of at least 5,000 customer records with LOD expressions and interactive filters

Concepts Used: Tableau data blending, LOD expressions, calculated fields, storyboards, interactive filters, design best practices

Industrial Relevance: Customer experience analytics, CRM reporting, marketing insights

AI-Powered Assignments

Assignment 1: AI-Assisted Power Query Transformation Evaluation

Students prompt an AI tool: Clean and transform a raw sales dataset with missing values, duplicate rows, incorrect data types, and inconsistent category names using Power Query steps. Students must:

- Evaluate whether the AI-suggested cleaning steps handle all data quality issues correctly without inadvertently removing valid data
- Identify incorrect data type conversions, missed duplicates, or filtering steps that may distort the downstream analysis
- Improve the transformation by applying proper null handling, conditional columns, and documented step naming in Power Query

Assessment focuses on: data cleaning reasoning, Power Query step validation, and critical evaluation of AI-generated transformation logic

Assignment 2: AI-Generated DAX Measure Critique

Students prompt an AI tool: Create DAX measures for total revenue, monthly growth rate, top 5 products by sales, and year-to-date (YTD) revenue in a Power BI sales report.

Students must:

- Analyze whether each AI-generated DAX measure correctly handles filter context and row context for the intended calculation
- Identify errors in time intelligence functions (e.g., incorrect DATEADD parameters or missing ALL/ALLEXCEPT for context modification)
- Refine the measures by applying correct DAX patterns and validating outputs against manually calculated expected values in sample data

Assessment focuses on: DAX formula correctness, filter context understanding, and critical validation of AI-generated business calculations

Assignment 3: AI-Based Tableau Dashboard Design Review

Students prompt an AI tool: Design a Tableau dashboard to present quarterly KPIs for a healthcare organization, including patient volume, average wait time, and revenue trends.

Students must:

- Evaluate whether the AI-suggested chart types are appropriate for each KPI and support clear stakeholder communication
- Identify misleading visualization choices (e.g., inappropriate use of pie charts, truncated axes, poor color encoding for accessibility)
- Improve the dashboard by applying design best practices: clear titles, consistent color palette, appropriate chart types, and mobile-responsive layout

Assessment focuses on: visualization design reasoning, dashboard usability evaluation, and critical assessment of AI-generated BI report quality



No Coding / Low Coding Development

Course Code: 26DS2310

Semester: III

Programs: B. Tech. CSE (Data Science)

Course Type: Core – Skill Enhancement Course

Credits: 2

L – T – P – J: 1 – 0 – 2 – 0

Prerequisite: Basic Programming (Python)

Course Vision

To equip students with the practical ability to take a software application through its complete development lifecycle — from planning and design to building, testing, deployment, and maintenance — using modern No-Code, Low-Code, and AI-assisted tools, enabling them to rapidly deliver and sustain real-world web applications aligned with current industry practices, while applying their Python skills strategically where custom logic is required.

Course Objectives

- To understand the software development lifecycle (SDLC) and the role of No-Code, Low-Code, and AI tools at each stage.
- To plan and design applications using AI-assisted requirement engineering and visual design tools.
- To build functional web applications using a low-code, Python-based web development approach.
- To apply AI-assisted coding, version control, and testing tools to accelerate and improve software quality.
- To deploy, operate, and maintain applications using containerization and cloud platforms.

Course Outcomes (CO)

CO	Description	Blooms Level
C01	Understand the SDLC and the role of No-Code, Low-Code, and AI tools across its stages.	L2
C02	Plan and design web applications using AI-assisted requirement engineering and visual design tools.	L4
C03	Build functional web applications using a low-code, Python-based development approach.	L6
C04	Apply AI-assisted coding, version control, and testing to accelerate development and improve quality.	L3
C05	Deploy, operate, and maintain applications using containerization and cloud platforms.	L6



Broad Course Description

This course teaches students how to deliver and maintain a complete web application using modern No-Code, Low-Code, and AI-assisted tools, organised around the software development lifecycle (SDLC). Students, who already know Python, carry a single application from idea to a live, maintained product. The course covers AI-assisted planning and design, low-code web development with Python (Flask), AI-assisted coding and version control, testing and debugging, containerised deployment with Docker, cloud hosting on platforms such as Hugging Face Spaces, and ongoing maintenance and operations. Rather than surveying many tools superficially, students gain hands-on depth with a focused toolchain at each lifecycle stage, learning to use AI to accelerate work while retaining engineering discipline. By the end, learners can independently plan, build, deploy, and maintain a working web application, reflecting real industry workflows.

Course Modules

Module 1: Software Lifecycle & AI-Assisted Planning and Design (6 Hours)

Overview of the SDLC and No-Code/Low-Code/AI paradigms; AI-assisted requirement engineering; user stories and specifications; system architecture; visual UI/UX design and wireframing; project setup and version control basics.

Module 2: Low-Code Web Application Development (7 Hours)

Web application fundamentals; building web apps with a low-code Python approach (Flask); routing, templates, and forms; working with a database (SQLite); displaying and managing data; AI-assisted feature development.

Module 3: AI-Assisted Coding, Version Control & Collaboration (6 Hours)

AI pair-programming with coding copilots; prompting and refining AI-generated code; Git and GitHub workflows; branching, commits, and issues; collaborative development and code review with AI assistance.

Module 4: Testing, Debugging & Quality (6 Hours)

Software testing fundamentals; AI-assisted test generation; debugging with AI tools; handling errors and edge cases; basic continuous integration; improving reliability and maintainability.

Module 5: Deployment, Operations & Maintenance (5 Hours)

Containerisation with Docker; cloud deployment (Hugging Face Spaces); configuration and environment management; monitoring and logging; performing updates, bug fixes, and maintenance using AI tools.

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module 1: Software Lifecycle & AI-Assisted Planning and Design

AI Integration: Using AI assistants to generate requirements, user stories, and architecture; AI-supported design decisions.



Module 2: Low-Code Web Application Development

AI Integration: AI-assisted generation of routes, templates, and database code; embedding AI features within the web app.

Module 3: AI-Assisted Coding, Version Control & Collaboration

AI Integration: AI pair-programming, AI-generated commits and documentation, AI-supported code review.

Module 4: Testing, Debugging & Quality

AI Integration: AI-generated test cases, AI-assisted debugging and error diagnosis, AI-driven code-quality suggestions.

Module 5: Deployment, Operations & Maintenance

AI Integration: AI assistance for Dockerfiles and deployment configuration, AI-supported log analysis and maintenance.

Authentic Intelligence Reflection:

Module 1: Software Lifecycle & AI-Assisted Planning and Design

Authentic Intelligence Reflection: Reflect on the value and limits of AI-generated requirements, evaluate design trade-offs, consider ethics of AI in planning.

Module 2: Low-Code Web Application Development

Authentic Intelligence Reflection: Analyse where low-code accelerates work versus where hand-written code is needed, reflect on maintainability of generated code.

Module 3: AI-Assisted Coding, Version Control & Collaboration

Authentic Intelligence Reflection: Reflect on developer responsibility for AI-generated code, evaluate correctness and originality, consider collaboration ethics.

Module 4: Testing, Debugging & Quality

Authentic Intelligence Reflection: Examine the reliability of AI-generated tests, reflect on the importance of human verification, consider quality and safety.

Module 5: Deployment, Operations & Maintenance

Authentic Intelligence Reflection: Reflect on operational responsibility, analyse security and privacy in deployment, consider sustainable maintenance practices.

List of Experiments For lab integrated course

All experiments build a single web application of the student's choice (open project brief), carried from planning through deployment and maintenance. Each module deepens one stage of the lifecycle using a specific toolchain.



No.	Experiment	Module Link
1	Using an AI assistant (ChatGPT/Claude), convert a one-line project idea into a structured Software Requirements Specification (SRS) with functional requirements and 8–10 user stories; critique and correct at least two AI errors.	Module 1
2	Generate a system-architecture description and block diagram for the app (frontend, backend, database); map each component to the tool that will build it. Create the GitHub repository, README, and log user stories as GitHub Issues.	Module 1
3	Design low-fidelity wireframes for three screens (e.g., home, data-entry form, dashboard) using a visual design tool (Figma) and produce a clickable prototype link.	Module 1
4	Scaffold a Flask web application: set up the project structure, create the first route, and render an HTML template. Run the app locally.	Module 2
5	Build a data-entry form in Flask, connect it to a SQLite database, and store submitted records (Create and Read operations).	Module 2
6	Implement Update and Delete operations and a listing/dashboard page that displays stored data in a table.	Module 2
7	Use an AI coding copilot (GitHub Copilot in VS Code) to add a new feature (e.g., search or filter); review, refine, and document the AI-generated code.	Module 3
8	Apply a Git/GitHub workflow: create a feature branch, commit changes, open a pull request, and merge; close the corresponding issue.	Module 3
9	Generate unit tests for the application logic using an AI tool, run them, and fix any failing cases.	Module 4
10	Use AI-assisted debugging to diagnose and fix an intentionally introduced bug; handle errors and edge cases gracefully.	Module 4
11	Containerise the application by writing a Dockerfile (AI-assisted), build the image, and run the container locally.	Module 5
12	Deploy the containerised application to Hugging Face Spaces; verify it runs publicly, then perform one maintenance update (bug fix or small feature) and redeploy.	Module 5

Assessment Scheme based on the course

Criteria	Weightage of Marks
Lab Reports	10
Mini Project	50
Lab Internal Exam	30
Hackathon / Technical Competition / Paper presentation / Global /	10



DLI certifications	
Total	100 marks

Recommended Tools & Platforms

1. Planning & Design: ChatGPT / Claude (AI assistants), Figma (wireframing).
2. Build (Web Development): Python, Flask, SQLite, VS Code.
3. AI-Assisted Coding & Version Control: GitHub Copilot, Git, GitHub.
4. Testing & Debugging: pytest, AI-assisted debugging tools.
5. Deployment & Operations: Docker, Hugging Face Spaces.

Reference Resources

1. Official documentation: Flask, SQLite, Docker, GitHub, Hugging Face Spaces.
2. Miguel Grinberg, Flask Web Development, O'Reilly, current edition.
3. GitHub Copilot and Git/GitHub official guides and tutorials.
4. Online courses on Coursera, Udemy, and edX covering web development, Git, Docker, and AI-assisted development.

CO PO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	2	2			2							1	2	1
CO2	3	3	2	2	2				1	1		2	2	2
CO3	3	3	3	2	3				1	1		2	3	3
CO4	2	3	2	2	3				1	2		2	2	3
CO5	2	2	3	2	3			2	1	2	2	2	3	3
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)														

Additional Resources

Additional AI & Computational Resources

Online Documentation:

Official documentation for Flask, SQLite, Docker, Git/GitHub, and Hugging Face Spaces.

E-Learning Platforms:

Courses on Coursera, Udemy, and edX for web development, version control, containerisation, and AI-assisted coding.

Coding & Collaboration Tools:

VS Code with GitHub Copilot, GitHub for version control and project management.

Deployment Platforms:

Docker for containerisation and Hugging Face Spaces for free public hosting.

YouTube Channels:

Channels covering Flask, Git, Docker, and AI-assisted development for practical learning.

Project Component

Students work on a single web application of their own choice (open project brief) and carry it through the complete software development lifecycle across the five modules. The same project is progressively built, tested, deployed, and maintained. Suggested themes include a feedback/analytics portal, a student or inventory management system, an expense tracker, or



any equivalent CRUD-based web application approved by the instructor.

Lifecycle Stages & Deliverables

Stage 1 – Plan & Design (Module 1)

Goal: Define the problem and turn it into a working specification and design.

Deliverables:

- AI-generated and student-refined SRS with functional requirements and user stories.
- System-architecture block diagram mapping components to tools.
- Figma wireframes for the core screens plus a clickable prototype.
- Initialised GitHub repository with README and user stories logged as Issues.

Tools: ChatGPT/Claude, Figma, GitHub.

Stage 2 – Build (Module 2)

Goal: Develop the working web application with persistent data.

Deliverables:

- Flask application with routing and HTML templates.
- Data-entry forms connected to a SQLite database.
- Full CRUD (Create, Read, Update, Delete) functionality.
- A listing/dashboard page displaying stored records.

Tools: Python, Flask, SQLite, VS Code.

Stage 3 – Code Faster & Collaborate (Module 3)

Goal: Accelerate development with AI assistance and manage code professionally.

Deliverables:

- At least one feature added with GitHub Copilot, reviewed and documented.
- Git workflow with feature branches, commits, pull requests, and merges.
- Issues tracked and closed against completed features.

Tools: GitHub Copilot, Git, GitHub.

Stage 4 – Test & Debug (Module 4)

Goal: Verify quality and improve reliability.

Deliverables:

- AI-generated unit tests for core logic, executed and passing.
- At least one bug diagnosed and fixed using AI-assisted debugging.
- Graceful error handling for invalid input and edge cases.

Tools: pytest, AI debugging assistants.

Stage 5 – Deploy & Maintain (Module 5)

Goal: Ship the application and keep it running.

Deliverables:

- Dockerfile and a working container image of the app.
- Application deployed publicly on Hugging Face Spaces.
- One maintenance update (bug fix or small feature) applied and redeployed.
- Brief operations note covering logging/monitoring and the update performed.

Tools: Docker, Hugging Face Spaces.

Final Capstone Deliverable

A complete, publicly deployed, and maintained web application, accompanied by its GitHub



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

repository (with commit history and issues), SRS and design artifacts, test results, Dockerfile, the live Hugging Face Spaces link, and a short report documenting the lifecycle, the AI tools used at each stage, and reflections on where AI accelerated the work versus where human engineering judgement was essential.



R Programming

Course Code: 26DS2408

Semester: IV

Programs Offered: B. Tech CSE (Data Science)

Course Type: Skill Enhancement Course (SEC-II)

Credits: 2

L – T – P – J: 1 – 0 – 2 – 0

Prerequisites: Basic programming knowledge, statistics fundamentals,

Course Vision

To develop strong analytical and programming skills using R for data analysis, visualization, and statistical modeling, enabling students to solve real-world data-driven problems using modern computational tools and AI-assisted techniques.

Course Objectives

1. Introduce R programming for statistical computing and data analysis.
2. Develop skills in data manipulation, cleaning, and transformation.
3. Enable visualization and exploratory data analysis using R libraries.
4. Provide knowledge of statistical modeling and inference using R.
5. Equip students with tools for real-world data analytics and AI integration.

Course Outcomes (CO)

CO	Description	Blooms Level
CO1	Understand R programming fundamentals and data structures.	L3
CO2	Apply data manipulation and preprocessing techniques using R.	L3
CO3	Perform exploratory data analysis and visualization.	L3
CO4	Apply statistical methods and modeling techniques in R.	L3
CO5	Develop data-driven solutions using R for real-world applications.	L6

Broad Course Description

R Programming for Data Analytics is a core course designed to introduce students to statistical computing and data-driven problem-solving using the R programming language. The course focuses on data manipulation, visualization, and statistical modeling techniques essential for data analytics.

Students gain hands-on experience using R libraries such as dplyr, ggplot2, and tidyr to analyze and interpret complex datasets. The course integrates AI-enabled tools and platforms to enhance learning and automate data analysis workflows. With an emphasis on practical implementation



and real-world applications, this course prepares students for careers in data science, analytics, and research domains.

CO-PO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
C01	2	1			2						1	2	1
C02	2	3			3						1	3	2
C03	2	3			3						1	3	2
C04	2	3	2	2	3						2	3	3
C05	2	3	2	2	3						2	3	3
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module 1: Introduction to R Programming & Data Structures (6 Hours)

Introduction to R and RStudio, Installing packages, Basic syntax, Variables and data types, Operators, Control structures (if, loops), Functions in R, Data structures (vectors, lists, matrices, arrays, data frames), Input/output operations.

Module 2: Data Manipulation and Preprocessing (6 Hours)

Data import/export (CSV, Excel, JSON), Data cleaning, Handling missing values, Data transformation, Data filtering and sorting, Introduction to dplyr and tidyr, Data merging and reshaping.

Module 3: Data Visualization using R (6 Hours)

Introduction to data visualization, Base R plotting, ggplot2 package, Bar plots, histograms, box plots, scatter plots, Line charts, Customization and themes, Data storytelling.

Module 4: Statistical Analysis using R (6 Hours)

Descriptive statistics, Probability distributions, Correlation and covariance, Hypothesis testing (t-test, chi-square), ANOVA, Regression basics, Introduction to statistical modelling.

Module 5: Predictive Analytics and Machine Learning in R (6 Hours)

Introduction to machine learning in R, Classification and clustering basics, Data analytics workflows, Real-world datasets, Report generation using R Markdown, Integration with AI tools.



AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module-1:

Students use AI tools to understand R syntax and generate code snippets for problem-solving.

Module-2:

AI-assisted tools help automate data cleaning and preprocessing tasks.

Module-3:

Students utilize AI-powered visualization tools to generate and interpret graphs.

Module-4:

AI models assist in statistical analysis and predictive modeling.

Module-5:

Students integrate AI tools with R for advanced analytics and automated reporting.

Assessment Scheme based on the course

Criteria	Weightage of Marks
Practical Experiments	30
Hackathon/Certificate/ Assignments	10
Lab Record	10
Project	50
Total	100 marks

Recommended Tools

- R, RStudio
- Libraries: dplyr, ggplot2, tidyr, caret
- Platforms: Kaggle, Google Colab

Primary Textbooks

1. R for Data Science authored by Hadley Wickham and Garrett Golemund, 2nd Edition, published in the year 2023 by O'Reilly Media.
2. The Art of R Programming authored by Norman Matloff, 1st Edition, published in the year 2011 by No Starch Press.
3. An Introduction to Statistical Learning with Applications in R authored by Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani, 2nd Edition, published in the year 2021 by Springer.

Additional Resources

- NPTEL – Data Analytics with R
- CRAN Documentation
- Kaggle datasets and tutorials



Project Component (if applicable)

Project 1 – Exploratory Data Analysis on Real Dataset

- Analyze and visualize dataset using R

Project 2 – Predictive Modeling using R

- Build regression/classification model

Project 3 – Data Visualization Dashboard

- Create interactive visualizations using R

Assignment Questions:

- Import a dataset (CSV/Excel) into R and display its structure.
- Create the following plots using R (preferably using ggplot2):
 1. Bar chart
 2. Histogram
 3. Box plot
 4. Scatter plot
- Conduct a hypothesis test (t-test or chi-square test) and interpret the results.
- Write R code snippets for each operation and explain the output.



MATLAB for Data Science

Course Code: 26DS2409

Semester: IV

Programs Offered: B. Tech CSE (Data Science)

Course Type: Skill Enhancement Course II

Credits: 2

L – T – P – J: 1 – 0 – 2 – 0

Prerequisites: Mathematics, Fundamental programming concepts.

Course Vision

To develop analytical and computational skills using MATLAB for solving real-world data science problems through data processing, visualization, machine learning, and AI-enabled decision-making techniques.

Course Objectives

1. Develop proficiency in MATLAB for data handling, preprocessing, and analysis.
2. Apply statistical and visualization techniques for extracting insights from data.
3. Build and evaluate machine learning models using MATLAB toolboxes.
4. Utilize parallel computing and GPU acceleration with NVIDIA support for scalable data processing.
5. Solve real-world data science problems through case studies with ethical and data-driven decision-making.

Course Outcomes (CO)

CO	Description	Blooms Level
CO1	To apply MATLAB for advanced data analysis and preprocessing	L3
CO2	To perform EDA and statistical modeling on real-world datasets	L3
CO3	To develop machine learning and computer vision models	L6
CO4	To implement AI and deep learning solutions using MATLAB	L3
CO5	To utilize GPU-based computing for high-performance AI applications	L3

Broad Course Description

MATLAB for CSE (Data Science) – Engineering Cluster is a foundational course offered in the second year of the B.Tech program, designed to build essential computational and analytical skills for engineering students. The course introduces core concepts of data science using



MATLAB, including data handling, matrix operations, statistical analysis, data visualization, and basic machine learning techniques, with an emphasis on real-world applications. It strengthens students' understanding of data-driven problem solving, numerical computation, and analytical modeling, which are vital for modern engineering systems.

The course integrates computational methods and Artificial Intelligence (AI) tools to enhance learning and bridge the gap between theory and practice. Students engage with MATLAB-based simulations, data analysis tasks, and visualization techniques to interpret datasets and solve engineering problems effectively. With a focus on application-oriented and interdisciplinary learning, the course connects data science concepts to emerging technologies such as intelligent systems, predictive analytics, and automation. Overall, it develops analytical thinking, programming proficiency, and problem-solving abilities required for effective engineering practice, innovation, and research.

CO-PO mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
C01	2	2			3						1	2	2
C02	2	3			3						1	3	2
C03	2	3	3	2	3						2	3	3
C04	2	3	3	2	3						2	3	3
C05	2	3	3	3	3						2	3	3
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module 1: Introduction to MATLAB for Data Science

(6 Hours)

MATLAB environment (Command Window, Workspace, Editor), Variables, data types, and operators, Data structures: arrays, tables, timetables, Data import/export (CSV, Excel, JSON, images), Basic data preprocessing techniques, MATLAB toolboxes overview.

Case Studies: Student Performance Analysis System, Financial Dataset Preprocessing.

Module 2: Data Preprocessing & Exploratory Data Analysis

(6 Hours)

Data cleaning and missing value handling, Data transformation and normalization, Exploratory Data Analysis (EDA), Data visualization (histogram, scatter, boxplot, heatmap), Feature selection basics

Case Studies: COVID-19 Trend Analysis, Customer Behaviour Analytics



Module 3: Statistical Modeling and Analysis

(6 Hours)

Probability distributions, Hypothesis testing (t-test, ANOVA, chi-square), Correlation and covariance, Regression (linear and multiple), Model validation

Case Studies: Healthcare Risk Prediction, Sales Forecasting System

Module 4: Machine Learning using MATLAB

(6 Hours)

Supervised learning (classification, regression), Unsupervised learning (clustering, dimensionality reduction), Algorithms: k-NN, Decision Trees, SVM, K-means, Introduction to Computer Vision in MATLAB, Feature extraction (texture, shape, edges), Model training, testing, and evaluation.

Case Studies: Image Classification using Machine Learning (Handwritten Digit / Object Recognition), Face Detection using Computer Vision Techniques

Module 5: Advanced Data Science & AI Applications (Mapped with NVIDIA)

(6 Hours)

Time series analysis and forecasting models, Advanced image processing and computer vision techniques, Deep learning architectures (CNN, transfer learning, basic RNN concepts), Feature extraction methods (LBP, HOG, Gabor filters), Big data handling and parallel computing in MATLAB, Model deployment and real-time inference systems, Introduction to edge AI and real-time analytics.

Case Studies: GPU-Based Facial Emotion Recognition System (Image Processing + CNN), Object Detection System using Deep Learning, Medical Image Analysis (Tumor Detection / MRI Classification).

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module 1: This module introduces the fundamentals of AI workflows in MATLAB, focusing on data preparation techniques for machine learning models and basic feature extraction methods. Students explore the MATLAB Machine Learning Toolbox and learn how to handle GPU-ready data to improve computational efficiency, building a strong foundation for AI-based problem solving.

Module 2: This module emphasizes advanced data preprocessing through feature engineering and normalization techniques suitable for deep learning models. Students learn to derive meaningful insights using visualization techniques, prepare datasets for AI applications, and design interactive AI dashboards for better data interpretation and decision-making.

Module 3: This module develops the statistical foundation required for machine learning by introducing key concepts such as model evaluation metrics and predictive analytics pipelines. Students gain hands-on experience with the MATLAB Statistical Toolbox to analyze data, validate models, and support data-driven decisions.



Module 4: This module focuses on machine learning model development using MATLAB, integrating tools such as the Computer Vision Toolbox. Students apply advanced feature extraction techniques including LBP, HOG, and Gabor filters, and learn model optimization, tuning strategies, and the basics of GPU acceleration for performance enhancement.

Module 5: This module advances into deep learning concepts, where students develop CNN-based models for image classification, object detection, and recognition tasks. It also covers real-time AI system implementation using the MATLAB Deep Learning Toolbox, preparing students for practical deployment of intelligent systems.

List of Experiments

No.	Experiment	Module Link
1	Perform matrix addition and multiplication using arrays in MATLAB and compare execution time with CPU	Module I
2	Import large dataset and generate plots (line, histogram, scatter) using MATLAB processing	Module II
3	Implement parallel for-loop and compare with normal loop performance	Module V
4	Perform linear regression and analyze speed improvement using GPU acceleration	Module III
5	Apply K-means clustering using MATLAB GPU functions and visualize output	Module IV
6	Train Neural Network/CNN using MATLAB Deep Learning Toolbox with NVIDIA GPU support and evaluate performance	Module V

Laboratory Experiments - AI Integrated (Additional)

1. PCA: Reduce data dimensionality and visualize key features using MATLAB.
2. Anomaly Detection: Identify outliers in datasets using MATLAB techniques.
3. Time Series Forecasting: Predict future trends using MATLAB models.

Assessment Scheme

Criteria	Weightage of Marks
Practical Experiments	30
Hackathon/Certificate/ Assignments	10
Lab Record	10
Project	50
Total	100 marks

Recommended Tools

Primary Textbooks

1. Stephen J. Chapman, MATLAB Programming for Engineers, 6th Edition, Cengage Learning, 2019.
2. Rudra Pratap, Getting Started with MATLAB: A Quick Introduction for Scientists and Engineers, Revised Edition, Oxford University Press, 2010.
3. Holly Moore, MATLAB for Engineers, 5th Edition, Pearson Education, 2018.

Additional Resources

1. Ian Goodfellow, Yoshua Bengio, Aaron Courville, *Deep Learning*, MIT Press.
2. MathWorks Documentation – Official MATLAB Learning Resources (<https://www.mathworks.com/help>)
3. Cleve Moler, *Numerical Computing with MATLAB*, SIAM.
4. NPTEL / SWAYAM Courses on MATLAB, Data Science, and Machine Learning.
5. MATLAB Machine Learning Toolbox Documentation (<https://www.mathworks.com/products/statistics.html>)
6. MATLAB Deep Learning Toolbox Documentation (<https://www.mathworks.com/products/deep-learning.html>)
7. MATLAB Parallel Computing Toolbox (GPU Acceleration) (<https://www.mathworks.com/products/parallel-computing.html>)
8. NVIDIA Documentation – GPU Computing & Deep Learning (<https://docs.nvidia.com>)
9. Kaggle Datasets for real-world data analysis (<https://www.kaggle.com>)
10. MATLAB Online Platform for cloud-based experimentation (<https://matlab.mathworks.com>)

AI-Powered Assignments

Assignment 1: AI-Based Data Preprocessing and Cleaning

Students prompt an AI tool: Suggest steps to clean and preprocess a dataset with missing values, noise, and inconsistent formats.

Students must:

- Analyze the AI-recommended preprocessing techniques using MATLAB
- Identify incorrect or unnecessary steps suggested by AI
- Improve the preprocessing pipeline with proper normalization and handling methods Assessment focuses on data understanding, preprocessing accuracy, and validation of AI outputs
- **Assignment 2: AI-Based Exploratory Data Analysis (EDA)**

Students prompt an AI tool: Generate insights and visualization strategies for a given dataset.

Students must:

- Validate AI-generated insights using MATLAB visualizations
- Identify misleading or incomplete interpretations
- Enhance analysis with appropriate plots and feature selection

Assessment focuses on data interpretation, visualization skills, and critical evaluation

Assignment 3: AI-Assisted Model Selection

Students prompt an AI tool: Recommend a suitable machine learning model for a given dataset.

Students must:

- Analyze the suitability of suggested models using dataset characteristics
- Identify overfitting/underfitting risks in AI suggestions
- Implement and compare models in MATLAB

Assessment focuses on model understanding, evaluation metrics, and decision-making

Assignment 4: AI-Based Feature Extraction and Optimization

Students prompt an AI tool: Suggest feature extraction methods for image or structured data.

Students must:

- Evaluate AI-suggested techniques (e.g., PCA, LBP, HOG)
- Identify redundant or irrelevant features
- Optimize feature selection using MATLAB tools

Assessment focuses on feature engineering, optimization, and validation

Assignment 5: AI-Based Predictive Modeling and Evaluation

Students prompt an AI tool: Build a predictive model for a dataset and suggest evaluation metrics.

Students must:

- Implement the AI-suggested model in MATLAB
- Validate results using appropriate metrics (accuracy, RMSE, etc.)
- Improve model performance through tuning and optimization

Assessment focuses on model implementation, performance evaluation, and critical analysis of AI outputs

Mini Project Component

The following projects are suggested based on course outcomes. Students may attempt one project individually or in pairs.

Project 1 – Healthcare Data Analysis for Disease Prediction

Problem: Predict disease outcomes using patient datasets

Concepts Used: Data preprocessing, classification, statistical analysis

Industrial Relevance: Healthcare analytics, clinical decision support systems

Project 2 – Financial Fraud Detection System

Problem: Detect fraudulent transactions using historical financial data

Concepts Used: Anomaly detection, clustering, classification

Industrial Relevance: Banking, fintech, cybersecurity

Project 3 – Smart Agriculture Yield Prediction

Problem: Predict crop yield based on environmental and soil parameters

Concepts Used: Regression, data visualization, feature selection

Industrial Relevance: Precision agriculture, agri-tech solutions

Project 4 – Traffic Flow Prediction and Analysis

Problem: Analyze and predict traffic patterns using real-time data

Concepts Used: Time series forecasting, data visualization

Industrial Relevance: Smart cities, transportation systems

Project 5 – Customer Churn Prediction

Problem: Predict customer churn based on usage and behavioural data

Concepts Used: Classification models, feature engineering

Industrial Relevance: Telecom, e-commerce, subscription services

Project 6 – Energy Consumption Forecasting

Problem: Forecast energy usage based on historical consumption data

Concepts Used: Time series analysis, regression

Industrial Relevance: Smart grids, energy management systems

Project 7 – Image-Based Defect Detection in Manufacturing

Problem: Detect defects in products using image data

Concepts Used: Computer vision, feature extraction, CNN

Industrial Relevance: Quality control, industrial automation

Project 8 – Sentiment Analysis on Social Media Data

Problem: Analyze public sentiment from social media datasets

Concepts Used: Text analytics, classification, NLP basics

Industrial Relevance: Marketing analytics, brand monitoring

Project 9 – Stock Market Trend Prediction

Problem: Predict stock price trends using historical market data

Concepts Used: Time series models, regression, forecasting

Industrial Relevance: Financial analytics, investment strategies

Project 10 – Real-Time Emotion Recognition System

Problem: Detect and classify human emotions using facial images

Concepts Used: LBP, HOG, Gabor filters, CNN

Industrial Relevance: Mental health monitoring, HCI, AI-driven therapy systems



Research Ethics and Technical Writing

Course Code: 26DS2410

Semester: IV

Programs Offered: B. Tech (Data Science)

Course Type: Skill Enhancement Course (SEC-II)

Credits: 2

L – T – P – J: 1 – 0 – 2 – 0

Prerequisites: Basic knowledge of research methodology

Course Vision

To equip learners with a strong foundation in ethical research practices and effective technical communication skills, enabling them to conduct responsible, high-quality research and present their ideas clearly and professionally.

Course Objectives

1. To understand and apply research ethics in academic and professional work.
2. To develop awareness of plagiarism, IPR, and research misconduct.
3. To build strong technical writing and documentation skills.
4. To prepare structured research papers, reports, and proposals.
5. To use LaTeX for professional scientific documentation.

Course Outcomes (CO)

CO	Description	Blooms Level
CO1	Apply ethical principles in research practices by analyzing real-world scenarios, evaluating ethical dilemmas, and ensuring compliance with established research guidelines and standards.	L3
CO2	Identify and prevent plagiarism and intellectual property rights (IPR) violations by using appropriate tools, applying proper citation techniques, and ensuring originality in academic and research work.	L2
CO3	Demonstrate effective technical writing skills by structuring documents clearly, maintaining academic tone, and presenting ideas with clarity, coherence, and precision.	L2
CO4	Develop structured research documents and presentations by organizing content logically, integrating visuals and references, and effectively communicating research findings.	L6



C05	Create professional documents using LaTeX by applying advanced formatting, managing references, and producing publication-quality technical reports and research papers.	L6
------------	--	----

Broad Course Description

Research Ethics and Technical Writing is designed to equip students with essential knowledge of ethical principles in research and effective scientific communication. The course covers topics such as plagiarism, academic integrity, responsible conduct of research, and proper citation practices, along with structured technical writing for reports, research papers, and documentation. It also introduces tools like LaTeX, Microsoft Word, and Grammarly to enhance clarity, formatting, and professionalism in academic and technical writing.

CO-PO mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
C01	1	1					2		1		1	1	
C02	1	2					2		1		1	1	
C03	1	1			1		1		2		1	1	
C04	1	2	2	1	1		1		2		2	1	
C05	1	2	2	1	2		1		2		2	1	
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module 1: Research Ethics – Foundations

(6 Hours)

Introduction to research ethics and its significance, Core ethical principles: honesty, integrity, transparency, accountability, Responsible Conduct of Research (RCR), Types of research misconduct: fabrication, falsification, plagiarism, Ethical issues in data collection, storage, and reporting, Informed consent and confidentiality, Role of Institutional Ethics Committees, Conflict of interest in research, Ethical issues in collaborative research, Data sharing policies and reproducibility

Module 2: Research Ethics – IPR and Plagiarism

(6 Hours)

Topics: Intellectual Property Rights (IPR): patents, copyrights, trademarks, trade secrets, Patent filing process and basics, Copyright laws and fair use policies, Plagiarism: types (self, mosaic, accidental) and consequences, Citation styles: APA, IEEE, MLA, Chicago, Paraphrasing and



summarization techniques, Reference management tools (Mendeley, Zotero, EndNote), Open-source and licensing, Authorship guidelines and publication ethics, Avoiding predatory journals

Module 3: Technical Writing – Introduction (6 Hours)

Introduction to technical writing and communication, Characteristics: clarity, precision, coherence, conciseness, Structure of research papers, Writing abstract, introduction, and literature review, Technical vocabulary and style guidelines, Writing for different audiences (academic, industry), Use of tables, figures, and charts, Basics of data visualization, Avoiding ambiguity and redundancy, Proofreading and grammar essentials

Module 4: Technical Writing –Documentation (6 Hours)

Research methodology and experimental design, Writing methodology, results, and discussion sections, Thesis and dissertation writing structure, Literature survey and systematic review techniques, Data interpretation and result presentation, Technical reports, white papers, and case studies, Presentation skills and scientific communication, Editing, revising, and proofreading techniques, Use of graphs and visualization, Documentation standards and formatting guidelines

Module 5: LaTeX for Technical Writing (6 Hours)

Introduction to LaTeX and typesetting systems, Installation (TeX Live, Overleaf) and environment setup, Basic syntax and document structure, Sections, subsections, and formatting, Mathematical equations and symbols, Tables, figures, and cross-referencing, Bibliography management using BibTeX, Templates for journals, conferences, and theses, Debugging LaTeX errors and best practices, Custom commands and packages

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module 1: AI tools and Turnitin support understanding of ethical principles by analyzing case studies, detecting plagiarism, and promoting academic integrity. They also help identify ethical risks and simulate real-world scenarios for better decision-making.

Module 2: AI enables bias detection, data validation, and authenticity checks to ensure transparency and reliability in research. It assists in monitoring data integrity, detecting anomalies, and maintaining reproducible and well-documented research workflows.

Module 3: AI writing tools like Grammarly to help in drafting, editing, and summarizing technical content. They improve clarity, coherence, grammar, and assist in structuring research papers effectively.

Module 4: AI-integrated platforms such as LaTeX and Zotero support automated formatting, citation management, and structured documentation. They also enable collaborative writing, version control, and consistent document styling.

Module 5: AI tools assist in selecting appropriate journals, generating review feedback, and enhancing research visibility. Platforms like Canva help create impactful presentations, while AI supports visual storytelling and effective communication of research outcomes.



List of Experiments

No.	Experiment	Module Link
1	Advanced Ethical Case Analysis and Compliance Mapping	Module I
2	Deep Plagiarism and Semantic Similarity Analysis using Turnitin	Module II
3	AI-Driven Systematic Literature Review and Research Gap Identification using Google Scholar	Module III
4	Professional Research Paper Writing and Enhancement using Grammarly	Module IV
5	LaTeX-Based Journal-Ready Document Preparation using Overleaf	Module V
6	Intelligent Reference Management and Citation Automation using Zotero	Module IV
7	AI-Based Research Quality and Bias Assessment	Module II

Assessment Scheme

Criteria	Weightage of Marks
Practical Experiments	30
Hackathon/Certificate/ Assignments	10
Lab Record	10
Project	50
Total	100 marks

Recommended Tools

The following tools and platforms are recommended to support AI-integrated teaching, learning,

Primary Textbooks

1. C. R. Kothari and Gaurav Garg, Research Methodology: Methods and Techniques, 4th Edition, New Age International Publishers, 2019.
2. Carol Roberts and Laura Hyatt, The Dissertation Journey: A Practical and Comprehensive Guide to Planning, Writing, and Defending Your Dissertation, 3rd Edition, Corwin Publishers, 2018.
3. Meenakshi Raman and Sangeeta Sharma, Technical Communication: Principles and Practice, 3rd Edition, Oxford University Press, 2015.
4. Frank Mittelbach, Michel Goossens, Johannes Braams, David Carlisle, and Chris Rowley, The LaTeX Companion, 2nd Edition, Pearson Education/Addison-Wesley, 2004.
5. James D. Lester Jr. and James D. Lester Sr., Writing Research Papers: A Complete Guide, 15th Edition, Pearson Education, 2018.

Additional Resources



Additional AI & Computational Resources

1. NPTEL Lectures: Research Ethics & Technical Writing – nptel.ac.in
2. Google Colab Python Notebooks – colab.research.google.com (for AI-assisted text analysis, NLP, and data-driven research tasks)
3. Grammarly Writing Assistant – grammarly.com (AI-powered grammar checking and readability enhancement)
4. Turnitin & iThenticate – turnitin.com / ithenticate.com (plagiarism detection and academic integrity tools)
5. Zotero & Mendeley Reference Managers – zotero.org / mendeley.com (citation management and literature organization)
6. Overleaf & LaTeX – overleaf.com (professional document preparation and formatting for research papers)
7. Semantic Scholar & Google Scholar – semanticscholar.org / scholar.google.com (AI-powered academic search and literature review)
8. NVIDIA Deep Learning Institute (DLI) — learn.nvidia.com (Recommended for AI-enabled research writing, NLP applications, and ethical AI practices in documentation and analysis)

AI-Powered Assignments

Assignment 1: AI-Based Ethical Analysis and Research Validation

Students prompt an AI tool: Analyze a research scenario involving plagiarism, data fabrication, or authorship conflict and suggest ethical actions.

Students must:

- Analyze the correctness of AI-generated ethical decisions using standard research guidelines (COPE, institutional policies)
- Identify any missing ethical considerations (data privacy, consent, bias, conflict of interest)
- Refine the response with proper justification and real-world ethical frameworks

Assessment focuses on ethical reasoning, critical evaluation, and validation of AI outputs

Assignment 2: AI-Assisted Plagiarism Detection and Content Improvement

Students prompt an AI tool: Evaluate this research draft for plagiarism and suggest improvements for originality and citation.

Students must:

- Verify similarity reports using Turnitin and compare with AI suggestions
- Identify incorrect paraphrasing or missing citations
- Improve the document with proper paraphrasing and referencing techniques

Assessment focuses on academic integrity, paraphrasing skills, and validation of AI-assisted corrections

Assignment 3: AI-Driven Literature Review and Research Gap Identification

Students prompt an AI tool: Summarize recent research papers on a selected topic and identify potential research gaps.

Students must:

- Validate summaries using Google Scholar and Semantic Scholar
- Identify inaccuracies or missing key papers in AI output
- Develop a refined literature review with clearly defined research gaps

Assessment focuses on analytical skills, literature synthesis, and reliability of AI-generated



Assignment 4: AI-Enhanced Technical Writing and Document Structuring

Students prompt an AI tool: Generate a structured research paper including abstract, introduction, and methodology.

Students must:

- Evaluate the coherence, clarity, and academic tone of AI-generated content
 - Identify inconsistencies, redundancies, or lack of technical depth
 - Refine the document using Grammarly and domain knowledge
- Assessment focuses on writing quality, logical structuring, and improvement of AI-generated drafts

Assignment 5: AI-Supported LaTeX-Based Research Paper Preparation

Students prompt an AI tool: Generate a LaTeX template for a research paper with sections, references, and figures.

Students must:

- Validate formatting and structure using Overleaf and LaTeX
- Identify formatting errors or missing components (figures, citations, bibliography)
- Enhance the document to meet journal/conference submission standards

Assessment focuses on technical documentation skills, formatting accuracy, and effective use of AI tools

Mini Project Component

Project 1 – AI-Based Research Ethics Evaluation System

Problem: Develop an AI-assisted system to evaluate research documents for ethical compliance (plagiarism, citation accuracy, authorship, and data integrity).

Concepts Used: Research ethics principles, plagiarism detection, citation standards, bias detection.

Industrial Relevance: Academic publishing, universities, and ethical compliance bodies (e.g., Committee on Publication Ethics).

Project 2 – AI-Assisted Technical Writing and Review Platform

Problem: Build a system that generates, reviews, and refines technical documents using AI.

Concepts Used: NLP, grammar analysis, summarization, document structuring.

Industrial Relevance: Publishing and corporate documentation (e.g., Elsevier, Springer).

Project 3 – AI-Based Literature Review and Research Gap Analyzer

Problem: Design a system to summarize papers and identify research gaps.

Concepts Used: Text mining, clustering, semantic analysis.

Industrial Relevance: Research labs, universities, innovation centers.

Project 4 – Intelligent Reference Management System

Problem: Develop an AI tool for citation generation and reference organization.

Concepts Used: Metadata extraction, citation formatting.

Industrial Relevance: Research platforms (e.g., Zotero, Mendeley).



Project 5 – AI-Based Research Paper Quality Assessment

Problem: Evaluate research papers for clarity, originality, and structure.

Concepts Used: NLP, scoring models, readability analysis.

Industrial Relevance: Journals and conferences.

Project 6 – AI-Based Plagiarism Detection Enhancement Tool

Problem: Build a system to detect paraphrased and semantic plagiarism.

Concepts Used: Semantic similarity, NLP, text comparison.

Industrial Relevance: Academic integrity tools (e.g., Turnitin).

Project 7 – AI-Powered Abstract and Summary Generator

Problem: Generate concise abstracts from full research papers.

Concepts Used: Text summarization, transformer models.

Industrial Relevance: Research indexing platforms.

Project 8 – AI-Based Journal Recommendation System

Problem: Suggest suitable journals based on research content.

Concepts Used: Text classification, recommendation systems.

Industrial Relevance: Publishing industry (e.g., IEEE, Elsevier).

Project 9 – AI-Based Bias Detection in Research Content

Problem: Identify bias in datasets and research writing.

Concepts Used: Bias detection algorithms, fairness metrics.

Industrial Relevance: Ethical AI and research validation.

Project 10 – AI-Driven Research Presentation Generator

Problem: Automatically generate presentations from research papers.

Concepts Used: NLP, content summarization, visualization.

Industrial Relevance: Academic and corporate communication



Full Stack Development

Course Code: 26DS2411

Semester: IV

Programs: B. Tech / CSE – Data Science

Course Type: Core - Skill Enhancement Course II

Credits: 2

Prerequisite: Basic Programming, HTML Fundamentals

L – T – P – J: 1 – 0 – 2 – 0

Course Vision

To empower students with comprehensive knowledge and hands-on expertise in full-stack web development, enabling them to design, develop, and deploy scalable, secure, and high-performance web applications using modern frontend and backend technologies. The course aims to bridge the gap between theoretical computing concepts and real-world software development practices by integrating user-centric design principles, efficient server-side architectures, and robust database management systems. It emphasizes the development of problem-solving skills, collaborative development using version control systems, and deployment of applications in cloud-based environments. Additionally, the course envisions preparing students to build intelligent and data-driven applications by incorporating emerging trends such as AI-integrated interfaces, API-driven services, and real-time data processing systems. By the end of the course, learners will be equipped to contribute effectively to industry-grade projects, startup ecosystems, and advanced research in full-stack and web-based technologies.

Course Objectives

The course is designed to:

1. To understand the fundamental concepts of frontend and backend web development.
2. To design and develop responsive and interactive user interfaces using modern web technologies.
3. To build server-side applications and RESTful APIs using appropriate frameworks.
4. To integrate databases with web applications for efficient data storage and retrieval.
5. To deploy and manage full-stack applications using version control and cloud/hosting platforms.

Course Outcomes (CO)

CO	Description	Blooms Level
CO1	Design and develop responsive and user-friendly frontend interfaces using HTML, CSS, and JavaScript.	L6
CO2	Develop server-side applications and RESTful APIs using modern backend technologies.	L6
CO3	Analyze oscillations and wave phenomena in engineering systems using analytical methods and computational tools.	L4
CO4	Integrate frontend and backend components to build complete full-stack web applications.	L5
CO5	Deploy full-stack applications using version control systems and hosting platforms.	L6



Broad Course Description

This course introduces students to the fundamentals of full-stack web development, covering both frontend and backend technologies. It focuses on building responsive user interfaces, developing server-side applications and APIs, integrating databases, and deploying complete web applications. The course emphasizes hands-on learning, real-world application development, and the use of modern tools and frameworks to prepare students for industry-ready full-stack development.

Course Modules

Module - 1: Frontend Fundamentals (7 Hours)

HTML5, CSS3, Responsive design, Flexbox, Grid, JavaScript basics, DOM manipulation

Module - 2: Advanced Frontend Development (6 Hours)

Introduction to server-side programming, Node.js basics, Express.js, REST APIs

Module - 3: Backend Development (6 Hours)

Introduction to server-side programming, Node.js basics, Express.js, REST APIs

Module - 4: Database Integration (6 Hours)

Database basics, CRUD operations, MongoDB/MySQL integration

Module - 5: Deployment and Version Control (5 Hours)

Git & GitHub, Deployment basics, Hosting platforms

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module 1: Frontend Fundamentals

- AI-powered UI components (e.g., smart forms, auto-suggestions)
- Chatbot interface design for web applications
- Personalization of UI using basic user behavior data

Module 2: Advanced Frontend Development

- Data visualization dashboards for AI/ML outputs
- Frontend interfaces for machine learning applications
- Real-time data updates using AI-driven APIs

Module 3: Backend Development

- Development of APIs for serving ML models
- Integration of AI services (e.g., recommendation systems, NLP APIs)
- Handling real-time data processing for intelligent applications

Module 4: Database Integration

- Data storage and management for AI/ML datasets
- Designing schemas for structured and unstructured data
- Supporting real-time data pipelines for AI applications

Module 5: Deployment & Version Control

- Deployment of AI-enabled web applications



- Model serving using backend frameworks
- CI/CD pipelines for AI-integrated applications

Authentic Intelligence Reflection:

Module 1: Frontend Fundamentals

- Emphasis on user-centric design and usability
- Ensuring accessibility and cross-browser compatibility
- Designing intuitive interfaces for diverse user needs

Module 2: Advanced Frontend Development

- Efficient handling of dynamic content and UI updates
- Component-based design for scalability and maintainability
- Enhancing user experience through responsive interactions

Module 3: Backend Development

- Designing scalable and secure backend architectures
- Following best practices in API design and error handling
- Ensuring reliability and performance of server-side systems

Module 4: Database Integration

- Maintaining data consistency and integrity
- Efficient query design and optimization
- Secure handling of user and application data

Module 5: Deployment & Version Control

- Adoption of DevOps practices for continuous improvement
- Effective collaboration using version control systems
- Ensuring reliability, maintainability, and scalability in deployed systems

List of Experiments

No.	Experiment	Module Link
1	Create a static webpage using HTML & CSS	Module I
2	Design a responsive layout using Flexbox and Grid	Module I
3	Implement JavaScript for DOM manipulation and event handling	Module I
4	Develop a dynamic webpage using Fetch API	Module II
5	Build a simple React component for UI rendering	Module II
6	Create a REST API using Node.js and Express.js	Module III
7	Perform CRUD operations using MongoDB/MySQL	Module IV
8	Integrate frontend with backend services	Module III
9	Implement user authentication (login/signup system)	Module IV
10	Deploy a full-stack web application using GitHub and hosting platforms	Module V

Assessment Scheme based on the course

Criteria	Weightage of Marks
Lab Reports	10
Mini Project	50
Lab Internal Exam	30
Hackathon/Technical Competition/ Paper presentation/Global/DLI	10



certifications	
Total	100 marks

Recommended Tools

The following tools and platforms are recommended to support AI-integrated teaching, learning,

Primary Textbooks

1. Duckett, J. (2011). HTML and CSS: Design and Build Websites. Wiley Publications.
2. Duckett, J. (2014). JavaScript and JQuery: Interactive Front-End Web Development. Wiley Publications.
3. Brown, E. (2019). Web Development with Node and Express (2nd ed.). O'Reilly Media.
4. Chodorow, K. (2013). MongoDB: The Definitive Guide (3rd ed.). O'Reilly Media.

CO PO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
CO1	3	2	3		3				1	2		2	1
CO2	3	3	3	2	3				1	2		3	2
CO3	3	3	3	2	3				2	2	1	3	3
CO4	3	2	3	2	3				1	1		3	2
CO5	2	2	3		3	1	1	1	2	2	2	2	2
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Additional Resources

Additional AI & Computational Resources

1. **Mozilla Developer Network (MDN)**
Comprehensive reference for HTML, CSS, JavaScript, and web APIs.
2. **W3Schools**
Beginner-friendly tutorials and interactive examples for web technologies.
3. **Node.js Foundation**
Official documentation and guides for backend development using Node.js.
4. **React Documentation**
Official learning resource for building modern frontend applications using React.
5. **GitHub Docs**
Guides for version control, collaboration, and project management.

Project Component

Industry-Driven Projects:

Project 1 – AI-Based Smart E-Commerce Recommendation System

- **Problem:** Recommend personalized products to users based on browsing and purchase history



- **Concepts Used:** Full-stack architecture, REST APIs, recommendation algorithms, user behavior analysis
- **Industrial Relevance:** E-commerce platforms (Amazon, Flipkart), personalized marketing systems

Project 2 – AI-Powered Chatbot for Web Applications

- **Problem:** Develop a chatbot interface to handle user queries in real-time
- **Concepts Used:** Natural Language Processing (NLP), frontend integration, API handling
- **Industrial Relevance:** Customer support automation, service-based industries

Project 3 – Real-Time Crime Detection Dashboard (Multimodal)

- **Problem:** Build a dashboard to visualize and detect suspicious patterns from streaming data
- **Concepts Used:** Data visualization, API integration, real-time data processing
- **Industrial Relevance:** Smart surveillance systems, law enforcement analytics

Project 4 – Full Stack Student Performance Analytics System

- **Problem:** Analyze and visualize student performance data for insights
- **Concepts Used:** CRUD operations, database design, dashboards, data analytics integration
- **Industrial Relevance:** EdTech platforms, academic analytics systems

Project 5 – AI-Based Resume Screening Web Application

- **Problem:** Automatically filter and rank resumes based on job requirements
- **Concepts Used:** NLP, backend APIs, database filtering, frontend dashboards
- **Industrial Relevance:** HR Tech platforms, recruitment automation systems



Open Elective Courses

Statistical Computing for Data Science

Course Code: 23OE0014

Semester: VI

Programs Offered: B. Tech (All Branches except DS)

Course Type: Open Elective I

Credits:3

L – T – P – J: 3 – 0 – 0 – 0

Prerequisite(s): Basic knowledge of Python and R Programming

Course Vision

To develop the ability to apply statistical foundations, data analysis techniques, and machine learning concepts using R and Python programming, enabling data-driven decision-making and analytical problem-solving in real-world applications.

Course Objectives

1. Develop a strong foundation in statistical concepts and probability distributions for data analysis.
2. Build proficiency in hypothesis testing, parametric and non-parametric statistical methods for analytical decision-making.
3. Equip students with knowledge of classification models and machine learning algorithms for predictive analytics.
4. Provide a strong foundation in unsupervised learning techniques such as clustering and principal component analysis.
5. Enable students to apply statistical analysis and visualization using R and Python programming tools for real-world datasets.

Course Outcomes (CO)

CO	Description	Blooms Level
C01	Summarize and visualize data using statistical and probability concepts.	L2
C02	Apply hypothesis testing methods for inference on sample and population data.	L3
C03	Ability to fit, interpret, and evaluate regression and classification models.	L2
C04	Perform statistical analysis using R programming techniques.	L3
C05	Analyze and visualize datasets using R packages and data science tools.	L4



Broad Course Description

Statistical Foundations of Data Science is a comprehensive course that introduces the fundamental concepts of statistics, data analysis, and machine learning techniques for data-driven applications. The course covers data preprocessing, exploratory data analysis, probability distributions, hypothesis testing, regression analysis, classification models, and clustering techniques using R and Python programming. It focuses on developing analytical thinking and statistical reasoning for solving real-world problems.

The course integrates statistical learning methods with practical implementation using modern data science tools and programming libraries. Students learn to analyze datasets, apply visualization techniques, evaluate machine learning models, and interpret statistical outcomes effectively. The course emphasizes hands-on analytical skills, predictive modeling, and practical applications in business analytics, healthcare, finance, and scientific research.

CO-PO mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
C01	2	2			3						1	3	2
C02	2	3			3						1	3	2
C03	2	3	2	2	3						1	3	2
C04	2	3	2	2	3						2	3	3
C05	2	3	2	3	3						2	3	3
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module - 1: Introduction to Data Science Process (9 Hours)

Different types of data, data preprocessing, data cleaning, handling missing and noisy data, data reduction using principal component analysis, normalization, and discretization techniques.

Module - 2: Exploratory Data Analysis (9 Hours)

Central tendency, dispersion, five-number summary, cross-tabulations, histograms, box plots, correlation plots, scatter plots, line charts, bar charts, pie charts, bubble charts, decision trees, and cluster charts.

Module - 3: Data Mining Techniques (9 Hours)

Data mining applications, data objects and attribute types, statistical description of data, frequent pattern analysis, association and correlation mining, pattern evaluation methods, multidimensional pattern mining, and classification using frequent patterns.

Module - 4: Hypothesis Testing (9 Hours)



Confidence intervals, hypothesis construction, null and alternative hypotheses, Type I and Type II errors, Z-test, T-tests, ANOVA, MANOVA, chi-square test, Kruskal-Wallis test, and Wilcoxon sign-rank test.

Module - 5: Overview of R Programming

(9 Hours)

Basic features of R, vectors, matrices, lists, arithmetic operations, plotting techniques, graphical parameters, ggplot2 package, matrix operations, data visualization, reading and writing files.

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module 1: Students use AI tools to perform data preprocessing, cleaning, and normalization on real-world datasets.

Module 2: AI-assisted visualization tools are used to generate exploratory data analysis charts and statistical summaries.

Module 3: Students use AI-based analytical tools to perform pattern mining and classification analysis.

Module 4: AI tools are used to automate hypothesis testing and evaluate statistical significance of datasets.

Module 5: Students apply AI-assisted R programming tools for statistical modeling, visualization, and predictive analysis.

Assessment Scheme

Criteria	Weightage of Marks
Assignment	10
Class Tests / Quizzes	5
Mid-Semester Examination	30
AI Mini-Project (Group)	15
End-Semester Examination	40
Total	100 marks

Recommended Tools

1. R Studio
2. Python
3. Jupyter Notebook
4. ggplot2
5. Scikit-learn
6. Google Colab



Primary Textbooks

1. Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data, EMC Education Services, Wiley.
2. Tilman M. Davies, The Book of R – A First Programming and Statistics, 2016.
3. Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani, An Introduction to Statistical Learning, Springer, 2021.
4. Andy Field, Discovering Statistics Using R, Sage Publications, 2012.
5. Trevor Hastie, Robert Tibshirani, and Jerome Friedman, The Elements of Statistical Learning, Springer, 2008.

Mini Project Component

Industry-Driven Projects

Project 1 – Customer Purchase Behaviour Analysis

Problem: Analyze customer transaction datasets to identify purchasing trends and customer segments.

Concepts Used: Exploratory data analysis, clustering, visualization, statistical summaries.

Industrial Relevance: Retail analytics, customer relationship management, business intelligence.

Project 2 –Healthcare Statistical Analysis System

Problem: Develop a statistical analysis model to identify disease patterns and patient health trends.

Concepts Used: Hypothesis testing, regression analysis, visualization, classification models.

Industrial Relevance: Healthcare analytics, clinical decision support, medical research.

Project 3 –Financial Risk Prediction Model

Problem: Build predictive models to classify financial risk and loan approval categories.

Concepts Used: Logistic regression, decision trees, support vector machines, confusion matrix evaluation.

Industrial Relevance: Banking analytics, fraud detection, financial risk management.

Project 4 – Authentication System using Cryptography

Problem: Analyze social media datasets to detect trends, sentiments, and user behaviour patterns.

Concepts Used: Data preprocessing, clustering, classification algorithms, visualization techniques.

Industrial Relevance: Digital marketing, social media analytics, customer engagement analysis.



Project 5 – Malware Detection System

Problem: Develop a predictive system to evaluate and predict student academic outcomes.

Concepts Used: Regression models, hypothesis testing, PCA, classification algorithms.

Industrial Relevance: Educational analytics, institutional planning, performance monitoring.

AI-Powered Assignments

Assignment 1: AI-Assisted Data Cleaning and Preprocessing

Students prompt an AI tool: Perform preprocessing and cleaning of a real-world dataset. Students must:

- Analyze the correctness of preprocessing steps.
- Identify missing value handling and normalization issues.
- Validate outputs using statistical summaries and visual inspection.
- Assessment focuses on data preprocessing knowledge and validation of AI-generated solutions.

Assignment 2: AI-Based Exploratory Data Analysis and Visualization

Students prompt an AI tool: Generate visualizations and perform exploratory data analysis on datasets.

Students must:

- Analyze trends, distributions, and relationships identified by AI tools.
- Identify misleading visualizations or incorrect interpretations.
- Validate findings using statistical methods and graphical analysis.
- Assessment focuses on analytical reasoning, visualization quality, and interpretation accuracy.

Assignment 3: AI-Assisted Hypothesis Testing and Statistical Analysis

Students prompt an AI tool: Perform hypothesis testing and statistical inference on sample datasets.

Students must:

- Analyze correctness of statistical test selection and outputs.
- Identify incorrect assumptions or significance interpretations.
- Validate results manually or using statistical software tools.
- Assessment focuses on hypothesis testing concepts, inference accuracy, and analytical skills.

Assignment 4: AI-Based Regression and Classification Modeling

Students prompt an AI tool: Build regression and classification models using real-world datasets.

Students must:

- Analyze model performance and prediction accuracy.
- Identify overfitting, underfitting, or incorrect feature selection.



- Validate results using confusion matrix and cross-validation techniques.
- Assessment focuses on predictive analytics, model evaluation, and validation of AI-generated outputs.

Assignment 5: AI-Assisted Clustering and Pattern Mining

Students prompt an AI tool: Perform clustering and frequent pattern mining on datasets.

Students must:

- Analyze clustering quality and discovered data patterns.
- Identify incorrect associations or misleading clusters.
- Validate outputs using statistical measures and visualization tools.
- Assessment focuses on unsupervised learning concepts, clustering analysis, and interpretation of AI-generated insights.



Data Analytics for Decision Making

Course Code:

Semester: VI

Program(s): B. Tech (All Branches except DS)

Course Type: Open Elective (OE-I)

Credits: 03

L – T – P – J: 3 – 0 – 0 – 0

Prerequisite(s): Basics of Statistics.

Course Vision

To equip students with analytical, business intelligence, and decision-making skills using modern tools and AI-driven approaches for solving real-world business problems.

Course Objectives

1. Understand business intelligence foundations, analytics types, and decision support systems
2. Apply data warehousing, ETL processes, and OLAP concepts for enterprise analytics
3. Analyze business data using visualization, KPI dashboards, and performance metrics
4. Implement predictive and prescriptive analytics using data mining and machine learning techniques
5. Utilize GPU-accelerated AI analytics tools including NVIDIA RAPIDS, XGBoost, and LSTM for real-world applications
- 6.

Course Outcomes (CO)

CO	Description	Blooms Level
C01	Illustrate business intelligence concepts, analytics types, and decision support systems	L2
C02	Analyze business data using descriptive statistics and visualization techniques	L4
C03	Apply data warehousing and ETL processes for enterprise data management	L3
C04	Develop predictive and prescriptive analytics models for business decision-making	L6
C05	Implement AI-driven analytics using GPU-accelerated tools and frameworks (NVIDIA DLI)	L3

Broad Course Description

Data Analytics for Decision Making – BI Tools is a 3-credit Core course for CSE DS students in the 6th Semester. It bridges business intelligence theory with modern AI-driven analytics practice — covering decision support systems, data warehousing, visualization, predictive modeling, and GPU-accelerated deep learning for industrial analytics applications.



The course spans five modules: Foundations of Business Intelligence and Decision Making, Data Warehousing and Data Engineering, Business Reporting and Visualization, Predictive and Prescriptive Analytics, and NVIDIA AI-Driven Predictive Analytics and Time-Series Intelligence. The NVIDIA DLI integration in Module 5 provides hands-on GPU computing experience with XGBoost, LSTM, autoencoders, and RAPIDS cuDF for real-world predictive maintenance and anomaly detection use cases.

CO-PO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PS01	PS02
C01	3	2										2	2	
C02	2	3	2	2	2								3	3
C03	2	2	2	2	2								3	2
C04	3	3	3	2	3								3	3
C05	3	3	3	2	3							1	3	3
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)														

Course Modules

Module 1: Foundations of Business Intelligence and Decision Making (9 Hours)

Business Intelligence (BI) and Analytics overview, Decision Support Systems (DSS), Business Pressures–Responses–Support Model, Managerial Decision Making Process, Types of Decisions: Structured, Semi-Structured, Unstructured, DSS Frameworks, Evolution of BI, BI Architecture, Types of Analytics: Descriptive, Predictive, Prescriptive, Introduction to Big Data Analytics

Module 2: Data Warehousing and Data Engineering (9 Hours)

Data Warehouse Concepts and Data Marts, Operational Data Stores, Enterprise Data Warehouse (EDW), Metadata management, Data Warehousing Architectures, ETL: Extraction, Transformation, Loading, Data Integration techniques, OLAP vs OLTP, Data Warehouse Development Approaches, Real-Time Data Warehousing, Scalability and Security in data warehouse systems

Module 3: Business Reporting and Visualization (9 Hours)

Business Reporting Concepts, Data Visualization Principles, Types of Charts and Graphs, Dashboard Design and KPI Metrics, Balanced Scorecards, Visual Analytics, Performance Dashboards, Business Performance Management (BPM), Six Sigma and Analytics, Designing dashboards using Power BI and Tableau

Module 4: Predictive and Prescriptive Analytics (9 Hours)

Data Mining Concepts, CRISP-DM Process, Classification, Clustering, Association Rules, Neural Networks Basics, Support Vector Machines, k-Nearest Neighbours, Text Analytics and Sentiment Analysis, Web Analytics, Social Media Analytics, Optimization Techniques, Decision Trees, What-if Analysis, Simulation Models



Module 5: NVIDIA AI-Driven Predictive Analytics and Time-Series Intelligence (9 Hours)

Introduction to Predictive Maintenance, Time-Series Data Analytics, Anomaly Detection Techniques, GPU-Accelerated Data Processing using RAPIDS (cuDF), XGBoost for Failure Prediction, LSTM Networks for Time-Series Forecasting, Autoencoders for Anomaly Detection, Seq2Seq Models, GAN-based Anomaly Detection, Model Evaluation and Performance Comparison (CPU vs GPU), Deployment of Predictive Models

Case Study (NVIDIA DLI – AI-Driven Predictive Analytics): Training XGBoost models with GPU acceleration for equipment failure prediction, LSTM-based deep learning for time-series forecasting using TensorFlow and Keras, autoencoder-based anomaly detection on industrial sensor data, and hands-on GPU computing for real-time analytics — benchmarked against CPU baselines using RAPIDS cuDF

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module 1: AI tools are used to explore DSS architecture patterns and simulate decision scenarios across structured, semi-structured, and unstructured decision types. Students critically evaluate AI-generated BI architecture recommendations for completeness, scalability, and alignment with enterprise analytics requirements.

Module 2: AI-assisted ETL pipeline design tools are used to model data warehouse architectures and evaluate OLAP cube configurations. Students critically assess AI-generated data integration plans for schema correctness, transformation logic accuracy, and real-time warehousing feasibility.

Module 3: Students use AI-powered dashboard design assistants (Power BI Copilot, Tableau AI suggestions) to generate KPI layouts and chart recommendations. AI-generated dashboards are critically evaluated for visualization best practices, KPI accuracy, Balanced Scorecard alignment, and stakeholder communication effectiveness.

Module 4: AI-assisted model selection tools are used to evaluate classification, clustering, and association rule algorithms for given business datasets. Students critically assess AI-generated CRISP-DM process plans, model selection justifications, and optimization recommendations for correctness and business relevance.

Module 5: Students use NVIDIA DLI GPU environments to build XGBoost, LSTM, and autoencoder pipelines for predictive maintenance. AI-generated model architectures and hyperparameter configurations are critically evaluated for performance, generalization, and practical deployability in real industrial analytics scenarios.



Assessment Scheme

Criteria	Weightage of Marks
Class Tests / Assignments	10
Mid-Semester Examination	30
Hackathon / Technical Competition / Paper Presentation / DLI Certifications	10
Mini Project	10
End-Semester Examination	40
Total	100 marks

Recommended Tools

Primary Textbooks

1. Ramesh Sharda, Dursun Delen, Efraim Turban, Business Intelligence, Analytics, and Data Science: A Managerial Perspective, Pearson, 4th Edition
2. Jiawei Han, Micheline Kamber, Jian Pei, Data Mining: Concepts and Techniques, Morgan Kaufmann, 3rd Edition
3. Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani, An Introduction to Statistical Learning, Springer, 2nd Edition
4. Tableau Documentation and Learning Resources – help.tableau.com
5. Microsoft Power BI Documentation – learn.microsoft.com/en-us/power-bi
6. NVIDIA Deep Learning Institute Course Materials – Predictive Analytics – learn.nvidia.com

Additional AI & Computational Resources

1. NPTEL: Business Analytics and Data Mining – nptel.ac.in
2. Coursera: Business Intelligence and Data Warehousing – coursera.org
3. NVIDIA RAPIDS Documentation – rapids.ai
4. Kaggle Business and Finance Datasets – kaggle.com/datasets
5. Google Colab / Jupyter Notebooks with GPU – colab.research.google.com

Mini Project Component

The following projects are suggested based on course outcomes. Students may attempt one project individually or in pairs.



Project 1 – Business Intelligence Dashboard for Sales Analytics

Problem: Design an end-to-end BI solution for a retail organization: build an ETL pipeline from a normalized RDBMS to a star-schema data warehouse, and create a Power BI or Tableau dashboard with KPIs, trend analysis, and predictive insights

Concepts Used: ETL, data warehousing, OLAP, star schema, KPI dashboards, descriptive and predictive analytics

Industrial Relevance: Enterprise BI platforms, retail analytics, executive reporting

Project 2 – Predictive Maintenance System using Sensor Data

Problem: Develop a predictive maintenance model using industrial IoT sensor data — applying XGBoost for failure classification and LSTM for time-series anomaly detection, with GPU-accelerated processing using RAPIDS cuDF

Concepts Used: XGBoost, LSTM, RAPIDS cuDF, anomaly detection, time-series forecasting, GPU vs CPU benchmarking

Industrial Relevance: Smart manufacturing, oil and gas equipment monitoring, aerospace predictive maintenance

Project 3 – Customer Churn Prediction using CRISP-DM

Problem: Apply the CRISP-DM process to build a customer churn prediction model using classification algorithms (Decision Tree, SVM, k-NN), with visualization of churn patterns in a BI dashboard

Concepts Used: CRISP-DM, classification, feature selection, model evaluation, data visualization, BI dashboard design

Industrial Relevance: Telecom customer retention, banking churn analytics, subscription service management

AI-Powered Assignments

Assignment 1: AI-Assisted Data Warehouse Design Evaluation

Students prompt an AI tool: Design a data warehouse architecture for a hospital network managing patient records, billing, and appointment data from multiple source systems.

Students must:

- Evaluate whether the AI-proposed star or snowflake schema correctly models the business entities and supports efficient OLAP queries for reporting
- Identify ETL pipeline gaps: missing data lineage, incorrect dimension hierarchies, or incomplete slowly changing dimension (SCD) handling
- Improve the design by applying proper fact-dimension separation, SCD Type 2 for historical tracking, and partitioning strategies for query performance

Assessment focuses on: data warehouse design reasoning, ETL correctness, and critical evaluation of AI-generated schema and pipeline decisions

Assignment 2: AI-Generated Predictive Model Selection Analysis

Students prompt an AI tool: Select and implement the best predictive model to forecast quarterly revenue for a retail chain using historical sales, marketing spend, and seasonality data.

Students must:

- Analyze whether the AI-suggested model (regression, decision tree, or neural network) is appropriate for the data size, feature types, and prediction horizon



- Identify overfitting risks, incorrect feature engineering, or inappropriate evaluation metrics in the AI-proposed CRISP-DM workflow
- Refine the approach using proper cross-validation, feature importance analysis, and compare at least two models with justified final selection

Assessment focuses on: predictive modeling understanding, CRISP-DM process application, and critical evaluation of AI-generated model selection rationale

Assignment 3: AI-Based GPU Analytics Pipeline Critique

Students prompt an AI tool: Build a GPU-accelerated pipeline using RAPIDS cuDF and XGBoost to detect anomalies in manufacturing sensor data collected over 6 months at 1-minute intervals.

Students must:

- Evaluate whether the AI-generated data preprocessing pipeline correctly handles time-series stationarity, missing sensor readings, and feature scaling
- Identify incorrect XGBoost hyperparameter choices, inappropriate anomaly thresholds, or missing model evaluation on held-out time windows
- Improve the pipeline by applying proper temporal train-test splits, hyperparameter tuning, and CPU vs GPU performance benchmarking methodology

Assessment focuses on: GPU analytics understanding, time-series modeling correctness, and critical validation of AI-generated anomaly detection pipeline



Smart Cities & Urban Data Analytics

Course Code:

Semester: VI

Program(s): B. Tech (All Branches except DS)

Course Type: Open Elective I

Credits: 3

L – T – P – J: 3 – 0 – 0 – 0

Prerequisite(s): Data Science Basics, Python Programming, Statistics

Course Vision

To develop the ability to design intelligent and sustainable smart city systems by integrating urban data analytics, IoT-based infrastructure, and AI-driven decision-making for efficient urban planning and management.

Course Objectives

1. Develop a strong understanding of smart city concepts, architecture, and urban system components, enabling students to analyze modern urban challenges and design intelligent solutions.
2. Build the ability to analyze urban data collected from IoT systems, sensors, and spatial data sources, enabling effective data-driven decision-making.
3. Equip students with knowledge of data analytics and machine learning techniques for solving urban problems such as traffic prediction, pollution monitoring, and resource optimization.
4. Enable students to design smart city solutions integrating IoT, data analytics, and digital platforms for real-world applications.
5. Develop the capability to evaluate ethical, privacy, and sustainability challenges in smart cities, ensuring responsible use of data and AI technologies.

Course Outcomes (CO)

CO	Description	Blooms Level
CO1	Explain the fundamental concepts of smart cities, including architecture, components, and urban system integration, and understand how technology enables intelligent and sustainable urban development.	L2
CO2	Analyze urban data collected from multiple sources such as IoT devices, sensors, and spatial systems, and evaluate data acquisition techniques for real-time and large-scale urban environments.	L4
CO3	Apply data analytics and machine learning techniques to urban datasets for solving real-world problems such as traffic prediction, environmental monitoring, and urban planning.	L3



C04	Design smart city solutions by integrating IoT systems, data analytics platforms, and intelligent decision-making systems for applications in transportation, environment, energy, and governance.	L6
C05	Evaluate ethical, privacy, security, and sustainability challenges in smart cities, and apply responsible data usage and AI practices in urban systems.	L5

Broad Course Description

This course provides a comprehensive understanding of smart city systems and urban data analytics. It focuses on the integration of IoT, data analytics, and AI technologies to solve urban challenges such as traffic congestion, pollution, energy management, and public safety. Students learn how urban data is collected, processed, and analyzed for intelligent decision-making.

The course also explores modern applications such as smart transportation systems, environmental monitoring, and urban planning analytics. It emphasizes real-world problem solving, system design, and ethical considerations in data-driven urban environments.

CO-PO mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
C01	2	1			1	1					1	2	1
C02	2	3			2	1					1	3	2
C03	2	3	2	2	3	1					2	3	2
C04	2	3	3	2	3	2					2	3	3
C05	2	3	3	3	2	2	2				2	3	3
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module - 1: Introduction to Smart Cities

(9 Hours)

Definition and evolution of smart cities, urbanization challenges, smart city architecture and components, smart governance, smart mobility, smart environment, smart energy systems, global smart city initiatives, India's Smart Cities Mission overview.

Module - 2: Urban Data Sources and IoT Systems

(9 Hours)

Urban data types structured unstructured spatial, IoT architecture sensors gateways communication protocols, data acquisition from traffic sensors environmental sensors GPS mobile data, real-time data streaming, edge vs cloud computing, IoT-based urban systems.

Module - 3: Urban Data Analytics Techniques

(9 Hours)

Data preprocessing and integration, exploratory data analysis, machine learning techniques regression classification clustering, time series forecasting, GIS and spatial analytics, urban pattern analysis, predictive analytics.



Module - 4: Smart City Applications

(9 Hours)

Smart transportation intelligent traffic systems smart parking, smart environment air quality monitoring waste management, smart energy smart grids energy optimization, smart healthcare public safety systems, decision support systems.

Module - 5: Advanced Smart Cities and Future Trends

(9 Hours)

Case studies global smart cities Singapore Barcelona Amsterdam, Indian smart city projects, data privacy and security, ethical issues in surveillance, sustainability and green cities, AI in smart cities, digital twins in urban planning, generative AI for city modeling.

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module 1: AI tools are used to analyze smart city systems and simulate urban models, with students validating real-world applicability.

Module 2: AI is applied in IoT data pipelines for real-time urban data collection and processing, with focus on data reliability and sensor limitations.

Module 3: Machine learning models are used for urban data analytics and prediction, with students evaluating model accuracy and bias.

Module 4: AI-based optimization is applied in smart city applications, with validation of system efficiency and practical constraints.

Module 5: Advanced AI tools are used for smart city simulations and forecasting, with emphasis on ethical and responsible AI usage.

Assessment Scheme

Criteria	Weightage of Marks
Assignment	10
Class Tests	5
Mid-Semester Examination	30
Project	15
End-Semester Examination	40
Total	100 marks

Recommended Tools

1. Python for urban data processing and analytics
2. TensorFlow or PyTorch for machine learning models
3. Jupyter Notebook or Google Colab for experimentation and visualization
4. Cloud platforms (AWS, Azure, GCP) for scalable data processing
5. GIS tools (QGIS, ArcGIS) for spatial data analysis and mapping
6. Data visualization tools for dashboards and decision systems



Primary Textbooks

1. Anthony M. Townsend, Smart Cities: Big Data, Civic Hackers, and the Quest for a New Utopia, W.W. Norton & Company, 2013.
2. Marcus Foth, Urban Informatics, IGI Global, 2008.
3. Government of India, Smart Cities Mission Official Guidelines, Ministry of Housing and Urban Affairs, 2015.
4. IEEE, Smart Cities Resources and Standards, IEEE Publications.
5. World Bank, Urban Development and Smart City Reports, World Bank Publications.

Project Component

Industry-Driven Projects:

Project 1 – Smart Traffic Management System

Problem: Develop an intelligent system to manage traffic flow and reduce congestion using real-time data

Concepts Used: IoT sensors, traffic data analytics, machine learning prediction models

Industrial Relevance: Smart transportation systems, urban mobility planning, traffic optimization

Project 2 – Urban Pollution Monitoring System

Problem: Design a system to monitor and analyze air quality and environmental conditions in urban areas

Concepts Used: Environmental sensors, data analytics, anomaly detection, visualization

Industrial Relevance: Environmental monitoring agencies, smart city governance, sustainability initiatives

Project 3 – Smart Energy Management System

Problem: Develop a system to monitor and optimize energy consumption in urban infrastructure

Concepts Used: Smart grids, IoT data collection, predictive analytics, optimization techniques

Industrial Relevance: Energy management systems, smart grids, sustainable urban development

Project 4 – Smart Waste Management System

Problem: Design an intelligent waste collection and management system using sensor-based monitoring

Concepts Used: IoT sensors, route optimization, data analytics, real-time monitoring

Industrial Relevance: Municipal services, urban waste management, smart infrastructure

Project 5 – Urban Digital Twin Simulation

Problem: Develop a digital twin model of an urban system for simulation and decision-making

Concepts Used: Simulation modeling, IoT data integration, AI-based analytics, system optimization

Industrial Relevance: Urban planning, smart city simulation, infrastructure management



AI-Powered Assignments

Assignment 1: AI-Based Smart City System Design

Students prompt an AI tool to design a smart city model integrating transportation, environment, and governance systems.

Students must:

- Analyze the system architecture, components, and data flow
- Identify limitations in design assumptions and integration challenges
- Validate the model by comparing with real-world smart city implementations
- Assessment focuses on system design understanding, integration of technologies, and validation of AI-generated outputs

Assignment 2: AI-Based Urban IoT Data Analysis

Students use AI tools to process and analyze urban sensor data from sources such as traffic, pollution, and GPS systems.

Students must:

- Analyze data quality, preprocessing techniques, and data reliability
- Identify noise, missing values, or inconsistencies in datasets
- Validate results using statistical methods and visualization tools
- Assessment focuses on data analysis skills, preprocessing accuracy, and validation of AI outputs

Assignment 3: AI-Based Traffic Prediction Model

Students build machine learning models using AI tools for predicting traffic patterns and congestion.

Students must:

- Analyze model performance and prediction accuracy
- Identify overfitting, underfitting, or incorrect assumptions
- Validate results using real datasets and evaluation metrics
- Assessment focuses on predictive analytics, model evaluation, and real-world applicability

Assignment 4: AI-Based Smart City Optimization

Students simulate optimization of urban systems such as traffic flow, energy usage, or waste management using AI tools.

Students must:

- Analyze efficiency improvements and optimization results
- Identify constraints and limitations in optimization models
- Validate outcomes using performance metrics and comparisons
- Assessment focuses on optimization techniques, analytical reasoning, and validation skills

Assignment 5: AI-Based Urban Simulation and Analysis

Students simulate smart city systems using AI-based platforms for urban planning and decision-making.

Students must:

- Analyze system performance and simulation results
- Identify gaps between simulated and real-world scenarios
- Validate results using case studies and data-driven evaluation



R Programming for Data Analytics

Course Code:

Semester: VI

Programs Offered: B. Tech (All Branches except DS)

Course Type:

Credits: 3

L – T – P – J: 3 – 0 – 0 – 0

Prerequisites: Basic programming knowledge, statistics fundamentals,

Course Vision

To develop strong analytical and programming skills using R for data analysis, visualization, and statistical modeling, enabling students to solve real-world data-driven problems using modern computational tools and AI-assisted techniques.

Course Objectives

1. Introduce R programming for statistical computing and data analysis.
2. Develop skills in data manipulation, cleaning, and transformation.
3. Enable visualization and exploratory data analysis using R libraries.
4. Provide knowledge of statistical modeling and inference using R.
5. Equip students with tools for real-world data analytics and AI integration.

Course Outcomes (CO)

CO	Description	Blooms Level
CO1	Understand R programming fundamentals and data structures.	L2
CO2	Apply data manipulation and preprocessing techniques using R.	L3
CO3	Perform exploratory data analysis and visualization.	L3
CO4	Apply statistical methods and modeling techniques in R.	L3
CO5	Develop data-driven solutions using R for real-world applications.	L6

Broad Course Description

R Programming for Data Analytics is a core course designed to introduce students to statistical computing and data-driven problem-solving using the R programming language. The course focuses on data manipulation, visualization, and statistical modeling techniques essential for data analytics. Students gain hands-on experience using R libraries such as dplyr, ggplot2, and tidyr to analyze



and interpret complex datasets. The course integrates AI-enabled tools and platforms to enhance learning and automate data analysis workflows. With an emphasis on practical implementation and real-world applications, this course prepares students for careers in data science, analytics, and research domains.

CO-PO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
CO1	2	1			2						1	2	1
CO2	2	3			3						1	3	2
CO3	2	3			3						1	3	2
CO4	2	3	2	2	3						2	3	3
CO5	2	3	2	2	3						2	3	3
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module 1: Introduction to R Programming

(9 Hours)

Introduction to R environment, Installation and setup, Basic syntax, Data types and variables, Operators, Control structures (if, loops), Functions in R, R data structures: vectors, matrices, arrays, lists, data frames

Module 2: Data Manipulation and Preprocessing

(9 Hours)

Data import/export, Data cleaning techniques, Handling missing values, Data transformation, Data wrangling using dplyr, Filtering, grouping, and summarizing data, String manipulation

Module 3: Data Visualization and Exploratory Data Analysis

(9 Hours)

Data visualization concepts, Graphical techniques, ggplot2 library, Bar charts, histograms, box plots, scatter plots, Data exploration techniques, Correlation analysis, Descriptive statistics

Module 4: Statistical Analysis and Modeling

(9 Hours)

Probability distributions, Hypothesis testing, Regression analysis (linear and multiple), ANOVA, Time series basics, Statistical modeling using R

Module 5: Advanced Analytics and Applications

(9 Hours)

Introduction to machine learning in R, Classification and clustering basics, Data analytics workflows, Real-world datasets, Report generation using R Markdown, Integration with AI tools.



AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module-1:

Students use AI tools to understand R syntax and generate code snippets for problem-solving.

Module-2:

AI-assisted tools help automate data cleaning and preprocessing tasks.

Module-3:

Students utilize AI-powered visualization tools to generate and interpret graphs.

Module-4:

AI models assist in statistical analysis and predictive modeling.

Module-5:

Students integrate AI tools with R for advanced analytics and automated reporting.

Assessment Scheme based on the course

Criteria	Weightage of Marks
Project	10
Class Tests / Assignments	10
Mid-Semester Examination	30
Hackathon / Certification	10
End-Semester Examination	40
Total	100 marks

Recommended Tools

1. R, RStudio
2. Libraries: dplyr, ggplot2, tidyr, caret
3. Platforms: Kaggle, Google Colab

Primary Textbooks

1. R for Data Science authored by Hadley Wickham and Garrett Grolmund, 2nd Edition, published in the year 2023 by O'Reilly Media.
2. The Art of R Programming authored by Norman Matloff, 1st Edition, published in the year 2011 by No Starch Press.
3. An Introduction to Statistical Learning with Applications in R authored by Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani, 2nd Edition, published in the year 2021 by Springer.

Additional Resources



- NPTEL – Data Analytics with R
- CRAN Documentation
- Kaggle datasets and tutorials

Project Component

Project 1 – Exploratory Data Analysis on Real Dataset

Analyze and visualize dataset using R

Project 2 – Predictive Modeling using R

Build regression/classification model

Project 3 – Data Visualization Dashboard

Create interactive visualizations using R

Assignment Questions:

1. Import a dataset (CSV/Excel) into R and display its structure.
2. Create the following plots using R (preferably using ggplot2):
 - Bar chart
 - Histogram
 - Box plot
 - Scatter plot
3. Conduct a hypothesis test (t-test or chi-square test) and interpret the results.
4. Write R code snippets for each operation and explain the output.



Data Exploration using R

Course Code: 230E0015

Semester: VII

Programs Offered: B. Tech (All Branches except DS)

Course Type:

L – T – P – J: 3 – 0 – 0 – 0

Prerequisite(s): Proficiency in a Python and R programming language.

Course Vision

The course envisions empowering students with strong foundations in R programming for effective data exploration and analysis. It aims to develop analytical thinking through data visualization, exploratory data analysis, and statistical modeling techniques. The course also prepares learners to apply machine learning and classification methods for solving real-world data-driven problems.

Course Objectives

1. Understand and Apply R Programming Fundamentals
2. Perform Data Analysis and Visualization
3. Conduct Exploratory Data Analysis (EDA)
4. Implement and Evaluate Statistical and Machine Learning Models

Course Outcomes (CO)

CO	Description	Blooms Level
C01	Apply fundamental R programming concepts, including data types, operators, functions, and data structures, for data manipulation and analysis.	L3
C02	Create meaningful data visualizations and perform data transformation using tools such as ggplot2 and dplyr.	L6
C03	Conduct exploratory data analysis to identify patterns, anomalies, and relationships within datasets effectively.	L3
C04	Implement and evaluate regression models, including linear and non-linear techniques, for predictive data analysis.	L3
C05	Apply and assess classification algorithms such as logistic regression, decision trees, random forests, and support vector machines for data classification tasks.	L3



Broad Course Description

This course provides a comprehensive introduction to data exploration and analysis using the R programming language. It covers the fundamentals of R programming, including data types, operators, functions, and data structures, along with techniques for data handling, transformation, and visualization using popular libraries such as ggplot2 and dplyr. Students will gain practical knowledge in managing datasets, performing exploratory data analysis, and understanding relationships and patterns within data.

The course further emphasizes the application of statistical and machine learning techniques for real-world problem solving. Topics such as regression analysis, classification algorithms, cross-validation, principal component analysis, and model evaluation are introduced to help learners build predictive and analytical skills. By the end of the course, students will be equipped to analyze complex datasets, develop data-driven insights, and apply modern analytical methods using R effectively.

CO-PO mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
C01	3	2	3										2	
C02	3	3	3										2	
C03	3	3	3										2	
C04	3	3	3										2	
C05	2	3	3										2	
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)														

Course Modules

Module - 1: Fundamentals of R Programming: (9 Hours)

Introduction to R Studio, R vs Python, Comments in R, Constants, R Data Types, R Operators, R Conditional constructs, Looping constructs, Unconditional constructs, R Functions, Data structures used in R, Packages in R.

Module - 2: Introduction to Data Analysis (9 Hours)

Overview of data analysis, working with directory in R, Loading and handling data in R, Data Visualization with ggplot2, Data Transformation with dplyr.

Module - 3: Exploratory Data Analysis (9 Hours)

Exploring a new dataset, Anomalies in numerical data, visualizing relations between variables, Assumptions of Linear Regression, Validating Linear Assumption, Missing Values, Covariation, Patterns and Models, ggplot2 Calls.



Module - 4: Regression Analysis

(9 Hours)

Introduction, Types of Regression Analysis Models, Linear Regression, Simple Linear Regression, Non-Linear Regression, Regression Analysis with Multiple Variables, Cross Validation, Principal Component Analysis, Factor Analysis.

Module - 5: Classification:

(9 Hours)

Introduction, Different types of Classification, Logistic Regression, Support Vector Machines, K-Nearest Neighbours, Naïve Bayes Classifier, Decision Tree Classification, Random Forest Classification, Evaluation.

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module 1: Students use AI tools to generate and debug R programs for data manipulation and analysis.

Module 2: AI-assisted visualization tools are used to create graphs and transform datasets efficiently.

Module 3: Students use AI-based analytical tools to identify patterns, anomalies, and correlations in datasets.

Module 4: AI tools are used to build and evaluate regression models for predictive analytics.

Module 5: Students apply AI-assisted machine learning tools to implement and compare classification algorithms.

Assessment Scheme

Criteria	Weightage of Marks
Assignment	10
Class Tests / Quizzes	5
Mid-Semester Examination	30
AI Mini-Project (Group)	15
End-Semester Examination	40
Total	100 marks

Recommended Tools

The following tools and platforms are recommended to support AI-integrated teaching, learning,

Recommended Tools

1. R Studio
2. ggplot2
3. dplyr
4. tidyverse
5. Jupyter Notebook



Primary Textbooks

1. Hadley Wickham and Garrett Grolemund, *R for Data Science*, O'Reilly, 2017.
2. Dr. Bharati Motwani, *Data Analytics using R*, Wiley, 2019.
3. Roger D. Peng, *Exploratory Data Analysis with R*, Leanpub Publications, 2016.
4. Garrett Grolemund, *Hands-On Programming with R*, O'Reilly Media, 2014.
5. Nina Zumel and John Mount, *Practical Data Science with R*, Manning Publications, 2014.

Mini Project Component

Industry-Driven Projects

Project 1 – Exploratory Data Analysis on Sales Dataset

Problem: Analyze customer purchase behaviour and identify sales trends using R.

Concepts Used: Data cleaning, visualization, correlation analysis, ggplot2, dplyr.

Industrial Relevance: Retail analytics, business intelligence, customer behaviour analysis.

Project 2 – Student Performance Prediction System

Problem: Develop a predictive model to analyze and predict student academic performance.

Concepts Used: Regression analysis, data preprocessing, visualization, cross-validation.

Industrial Relevance: Educational analytics, academic performance monitoring, institutional planning.

Project 3 – Healthcare Data Analysis System

Problem: Analyze healthcare datasets to identify patterns in patient health records.

Concepts Used: Exploratory data analysis, missing value handling, regression models, classification.

Industrial Relevance: Healthcare analytics, medical decision support systems, patient monitoring.

Project 4 – Financial Risk Classification

Problem: Build a classification system to identify financial risk categories from banking data.

Concepts Used: Logistic regression, decision trees, random forest, data visualization.

Industrial Relevance: Banking analytics, fraud detection, financial risk assessment.

Project 5 – Social Media Sentiment Analysis

Problem: Analyze social media datasets to identify user sentiment and trends.

Concepts Used: Data transformation, visualization, classification techniques, machine learning models.



Industrial Relevance: Social media analytics, marketing intelligence, customer feedback analysis.

AI-Powered Assignments

Assignment 1: AI-Assisted R Programming and Data Manipulation

Students prompt an AI tool: Generate R programs for data handling and manipulation tasks.

Students must:

- Analyze the correctness of generated R code.
- Identify logical or syntactical errors in the code.
- Validate outputs using manual execution in R Studio.
- Assessment focuses on programming understanding, debugging ability, and validation of AI-generated solutions.

Assignment 2: AI-Based Data Visualization and Transformation

Students prompt an AI tool: Create visualizations and perform data transformation using ggplot2 and dplyr.

Students must:

- Analyze visualization quality and correctness.
- Identify misleading patterns or improper transformations.
- Validate results using different visualization techniques.
- Assessment focuses on visualization design, data interpretation, and analytical reasoning.

Assignment 3: AI-Assisted Exploratory Data Analysis

Students prompt an AI tool: Perform exploratory data analysis on a given dataset.

Students must:

- Analyze detected trends, anomalies, and relationships.
- Identify incorrect assumptions or missing insights.
- Validate analytical findings using statistical methods.
- Assessment focuses on exploratory analysis skills and interpretation of AI-generated insights.

Assignment 4: AI-Based Regression Analysis and Prediction

Students prompt an AI tool: Build regression models and predict outcomes from datasets.

Students must:

- Analyze model accuracy and prediction performance.
- Identify overfitting, underfitting, or incorrect assumptions.
- Validate results using cross-validation techniques.
- Assessment focuses on regression modeling, predictive analytics, and evaluation methods.



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Assignment 5: AI-Assisted Classification and Machine Learning

Students prompt an AI tool: Implement classification algorithms and compare their performance.

Students must:

- Analyze classification accuracy and evaluation metrics.
- Identify false predictions or incorrect classifications.
- Validate results using confusion matrix and performance measures.
- Assessment focuses on machine learning concepts, classification techniques, and evaluation of AI-based outputs.



MATLAB for Data Science

Course Code:

Semester: VII

Programs Offered: B. Tech (All Branches except DS)

Course Type: Open Elective Course – II

Credits: 3

L – T – P – J: 3 – 0 – 0 – 0

Prerequisites: Mathematics, Basic Programming concepts

Course Vision

To develop strong programming and computational skills using MATLAB integrated with NVIDIA GPU acceleration for solving mathematical, engineering, and real-world problems through high-performance computing and efficient data processing.

Course Objectives

1. Develop MATLAB programming skills for problem-solving and data analysis.
2. Apply visualization, signal, and image processing techniques to real-world data.
3. Understand and implement basic AI and machine learning concepts using MATLAB.
4. Utilize GPU-based computing and parallel processing (with NVIDIA support).
5. Design practical solutions through case studies with ethical and application-oriented approaches.

Course Outcomes (CO)

CO	Description	Blooms Level
C01	Apply MATLAB programming and optimization techniques effectively to solve complex computational and engineering problems	L3
C02	Analyze and process signal and image data using MATLAB tools for real-world and research-oriented applications	L4
C03	Develop and implement machine learning and AI-based models for data-driven problem solving	L6
C04	Design and implement computer vision algorithms and intelligent systems for automation and pattern recognition tasks	L6
C05	Utilize GPU-based computing and parallel processing techniques for scalable, high- performance applications	L3



Broad Course Description

This course provides a comprehensive introduction to MATLAB programming for data analysis, scientific computing, and engineering applications. It covers fundamental concepts such as matrix operations, scripting, and function development, along with advanced topics including optimization techniques, signal and image processing, and data visualization. The course further integrates machine learning, computer vision, and GPU-based computing to enable the development of scalable, intelligent systems for real-world problem solving.

C0-PO mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
C01	2	1			2						1	2	2
C02	2	3			2						1	3	2
C03	2	3	3	2	3						2	3	3
C04	2	3	3	2	3						2	3	3
C05	2	3	3	3	3						2	3	3
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module 1: MATLAB Programming Fundamentals & Problem Solving (9 Hours)

MATLAB environment (Command Window, Editor, Workspace), Variables, constants, data types, Operators (arithmetic, relational, logical), Input/output operations and formatted printing, Control structures (if, nested conditions, loops), User-defined functions and modular programming, Arrays, matrices, vectorization techniques, String processing and data structures (cell arrays, structures), Debugging, profiling, and error handling, Introduction to scripts vs functions vs live scripts

Module 2: Data Visualization & Signal Processing (9 Hours)

Advanced plotting (subplots, annotations, interactive plots) File handling (CSV, Excel, real-time streaming data) Signal representation (continuous & discrete signals) Signal transformations and convolution basics, Fourier Transform (FFT) and spectral analysis, Digital filtering (FIR/IIR basics), Noise reduction and smoothing techniques, Time-frequency analysis basics, Signal feature extraction

Module 3: Image Processing & Computer Vision (9 Hours)

Image representation and color models (RGB, HSV, grayscale), Image enhancement (contrast stretching, histogram equalization), Spatial filtering and noise removal, Edge detection (Sobel, Canny, Prewitt), Image segmentation (thresholding, region-based methods), Morphological operations (dilation, erosion), Feature extraction (texture, shape, keypoints), Introduction to object detection and tracking, Basics of image datasets and annotation



Module 4: Machine Learning & AI using MATLAB

(9 Hours)

Supervised learning (classification, regression), Unsupervised learning (clustering, PCA), Feature engineering and dimensionality reduction, Model training, validation, and cross-validation, Performance metrics (accuracy, precision, recall, F1-score), Overfitting and bias-variance trade-off, Introduction to deep learning (ANN basics), Model interpretability basics

Module 5: Advanced AI, Signal & Vision Systems (Mapped with NVIDIA)

(9 Hours)

Integration of signal processing and image processing, Deep learning (CNN, transfer learning basics), Time-series forecasting models, Real-time system development, Parallel computing and distributed processing, Model deployment and inference systems, Introduction to edge AI and IoT integration, Performance optimization techniques

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module 1: AI integration in this module focuses on building a strong foundation for intelligent systems by enabling efficient data handling, preprocessing, and algorithm development using MATLAB. Students learn vectorization and modular programming techniques that are essential for implementing AI workflows, along with debugging and profiling methods to optimize AI code performance.

Module 2: AI integration in this module emphasizes transforming raw signal data into meaningful features for AI models. Techniques such as Fourier Transform, filtering, and noise reduction are used to prepare high-quality inputs for machine learning applications like speech recognition, biomedical signal analysis, and predictive analytics.

Module 3: This module integrates AI through image preprocessing, feature extraction, and segmentation techniques that form the backbone of computer vision systems. Students apply these methods to prepare datasets and enable tasks such as object detection, facial analysis, and automated visual inspection using AI models.

Module 4: AI integration is central to this module, where students develop and evaluate machine learning models for classification, regression, and clustering. Concepts like feature engineering, model validation, and performance metrics help in building reliable AI systems, while introductory deep learning concepts enable understanding of advanced intelligent models.

Module 5: AI integration in this module focuses on developing scalable, real-time intelligent systems by combining deep learning, signal processing, and computer vision. Students explore GPU-based acceleration, edge AI, and deployment strategies, enabling high-performance applications such as smart healthcare systems, autonomous systems, and IoT-based intelligent solutions.



Assessment Scheme based on the course

Criteria	Weightage of Marks
Class Tests	10
Mid-Semester Examination	30
Mini Project	10
Assignments	10
End-Semester Examination	40
Total	100 marks

Recommended Tools

Primary Textbooks

1. Stephen J. Chapman, MATLAB Programming for Engineers, 6th Edition, Cengage Learning, 2019.
2. Stormy Attaway, MATLAB: A Practical Introduction to Programming and Problem Solving, 5th Edition, Butterworth-Heinemann/Elsevier, 2021.
3. Delores M. Etter, Introduction to MATLAB Programming, Pearson Education, 2013.
4. Rafael C. Gonzalez and Richard E. Woods, Digital Image Processing, 4th Edition, Pearson Education, 2018.
5. S. Rajasekaran and G. A. Vijayalakshmi Pai, Neural Networks, Fuzzy Logic and Genetic Algorithms: Synthesis and Applications, PHI Learning Pvt. Ltd., 2017.
6. Steven W. Smith, The Scientist and Engineer's Guide to Digital Signal Processing, 2nd Edition, California Technical Publishing, 1999.

Additional Resources

Additional AI & Computational Resources

1. Attaway, S. – MATLAB: A Practical Introduction to Programming and Problem Solving – Beginner-friendly book covering MATLAB basics and problem-solving techniques.
2. Chapman, S. J. – MATLAB Programming for Engineers – Focuses on engineering applications and numerical methods using MATLAB.
3. Hahn, B. D. & Valentine, D. T. – Essential MATLAB for Engineers and Scientists – Covers scientific computing and data analysis concepts.
4. Moore, H. – MATLAB for Engineers – Provides practical MATLAB applications with engineering examples.
5. Goodfellow, I., Bengio, Y., Courville, A. – Deep Learning – Comprehensive deep learning resource (Free: deeplearningbook.org).
6. NPTEL Course – MATLAB Programming for Numerical Computation – Covers MATLAB



fundamentals, arrays, loops, and functions – nptel.ac.in

7. NPTEL Course – Scientific Computing using MATLAB – Focuses on numerical methods, interpolation, and data analysis – nptel.ac.in
8. NPTEL Course – Simulation of Communication Systems using MATLAB – Includes modeling and simulation techniques – nptel.ac.in
9. NPTEL Course – MATLAB-Based Programming Lab – Hands-on MATLAB problem-solving and implementation – nptel.ac.in

Mini Project Component

The following projects are suggested based on course outcomes. Students may attempt one project individually or in pairs.

Project 1 – AI-Based Medical Image Tumor Detection

Problem: Detect and classify tumors from MRI/CT images

Concepts Used: Image segmentation, CNN, feature extraction

Industrial Relevance: Healthcare diagnostics, AI-assisted radiology systems

Project 2 – Smart Traffic Signal Optimization System

Problem: Optimize traffic signals based on real-time vehicle density

Concepts Used: Image processing, optimization algorithms, ML

Industrial Relevance: Smart city infrastructure and traffic management

Project 3 – Emotion Recognition using Facial Expressions

Problem: Identify human emotions from facial images or video

Concepts Used: Computer vision, deep learning, feature extraction

Industrial Relevance: Mental health monitoring, HCI, surveillance systems

Project 4 – Predictive Maintenance for Industrial Equipment

Problem: Predict machine failures using sensor and time-series data

Concepts Used: Signal processing, time-series forecasting, ML

Industrial Relevance: Manufacturing industries (Siemens, GE), Industry 4.0

Project 5 – GPU-Accelerated Image Classification System

Problem: Speed up deep learning model training using GPU computing

Concepts Used: CNN, parallel computing, CUDA

Industrial Relevance: High-performance AI systems using NVIDIA

AI-Powered Assignments

Assignment 1: AI-Based Material Selection Analysis

Student Prompt to AI Tool: Suggest the most suitable material for an aircraft wing considering strength, weight, and durability constraints.

Students must:

- Analyze the correctness of the AI-recommended materials using material property data (strength, density, toughness).
- Identify any unrealistic selections or missing constraints (cost, temperature stability, fatigue resistance).



- Improve the selection using Ashby charts or standard material selection criteria.

Assessment Focus:

Evaluation is based on engineering judgment, understanding of material properties, and validation of AI outputs.

Assignment 2 – AI-Based Signal Classification

Student Prompt to AI Tool: Recommend a method to classify ECG/speech signals for pattern recognition.

Students must:

- Evaluate the suggested AI approach using signal features (frequency, amplitude, noise levels).
- Identify limitations in preprocessing or feature extraction.
- Improve the model using filtering, FFT, or feature engineering techniques.

Assessment Focus:

Understanding of signal processing, feature extraction, and critical evaluation of AI models.

Assignment 3 – AI-Based Image Enhancement & Detection

Student Prompt to AI Tool: Suggest techniques to enhance image quality and detect objects in low-light images.

Students must:

- Analyze the effectiveness of enhancement methods (contrast, filtering).
- Identify issues such as noise amplification or poor segmentation.
- Improve detection accuracy using appropriate preprocessing and feature extraction methods.

Assessment Focus:

Application of image processing concepts and validation of AI-based solutions.

Assignment 4 – AI Model Selection and Evaluation

Student Prompt to AI Tool: Recommend the best machine learning model for predicting student performance.

Students must:

- Evaluate the suggested model using performance metrics (accuracy, precision, recall).
- Identify issues such as overfitting or bias.
- Improve model performance using tuning, feature selection, or alternative algorithms.

Assessment Focus:

Understanding of ML models, evaluation techniques, and model improvement strategies.

Assignment 5 – AI-Based GPU Optimization Strategy

Student Prompt to AI Tool: Suggest methods to speed up deep learning model training using GPU computing.

Students must:

- Analyze the feasibility of suggested GPU optimization techniques.
- Identify limitations such as memory constraints or inefficient parallelization.
- Improve performance using batching, parallel processing, or optimized algorithms.

Assessment Focus:

Knowledge of GPU computing, performance optimization, and scalability of AI systems.



Social Media Analytics

Course Code:

Semester: VII

Programs Offered: B. Tech (All Branches except DS)

Course Type: Open Elective (OE-II)

Credits: 3

L – T – P – J: 3 – 0 – 0 – 0

Prerequisites: Data science and visualization, Machine learning

Course Vision

To equip learners with advanced AI and deep learning techniques for analyzing, modeling, and interpreting social media data, enabling intelligent, ethical, and context-aware decision-making.

Course Objectives

1. Analyze social media ecosystems using AI-driven frameworks and data-centric approaches.
2. Apply deep learning and graph-based models to uncover patterns in social networks.
3. Utilize large language models for advanced text analytics and content understanding.
4. Design intelligent systems for sentiment analysis, behaviour prediction, and content moderation.
5. Evaluate real-world applications with a focus on ethical, explainable, and human-centric AI.

Course Outcomes (COs)

CO	Description	Blooms Level
C01	Interpret social media ecosystems and AI-driven analytical frameworks	L2
C02	Apply deep learning and graph-based models for social network analysis	L3
C03	Implement LLM-based techniques for social media text analytics and content generation	L3
C04	Develop AI solutions for sentiment analysis, behaviour prediction, and content moderation	L6
C05	Evaluate real-world social media analytics applications with ethical and explainable AI considerations	L5

Broad Course Description

This course introduces advanced AI techniques for analyzing and understanding social media data, focusing on network structures, user behaviour, and content dynamics. It covers deep learning and graph-based models for social network analysis, along with large language models for text analytics and content generation.



The course also explores intelligent systems for sentiment analysis, behaviour prediction, and misinformation detection, while emphasizing ethical considerations, fairness, and explainability. Through real-world applications and case studies, students gain the ability to design scalable and socially responsible social media analytics solutions.

CO-PO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
CO1	2	1			1	1					1	2	1
CO2	2	3			2						1	3	2
CO3	2	3	3	2	3						2	3	2
CO4	2	3	3	2	3						2	3	3
CO5	2	3	3	3	2	2	1				2	3	3
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module 1: AI Foundations for Social Network Analysis

(9 Hours)

Role of AI in social network analytics; structure and dynamics of social networks; spatial and temporal spread of information; characteristics and challenges of social media data; overview of deep learning and LLMs in social media.

Applications: Public opinion tracking, information diffusion analysis, trend monitoring.

Module 2: Deep Learning & Graph-Based Social Network Modeling

(9 Hours)

CNNs and RNNs for social media data representation; Graph Neural Networks (GNNs); community detection and network analysis; influence modeling and propagation; hidden pattern discovery in large-scale networks.

Applications: Influencer identification, community detection, viral content prediction.

Module 3: Large Language Models & Social Media Text Analytics

(9 Hours)

LLMs (GPT-based models) for social media; text generation, summarization, and classification; sentiment analysis and opinion mining; sarcasm detection and code-mixed language processing; multimodal learning.

Applications: Chatbots, content moderation, multilingual sentiment analysis.

Module 4: Intelligent Social Media Systems & Prediction

(9 Hours)



User behaviour prediction; personalized advertising and recommendation; detection of hate speech, offensive content, and misinformation; mental health analytics; explainable AI and transfer learning.

Applications: Targeted marketing, mental health monitoring, fake news detection.

Module 5: Advanced Applications, Case Studies & Ethics

(9 Hours)

Applications in disaster response, digital agriculture, and low-resource language analytics; evaluation and deployment challenges.

Case studies include hate speech detection, depression detection using LLMs, and sarcasm detection in code-mixed data.

Applications: Crisis response, social good analytics, policy decision-making.

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module 1: AI enables large-scale modeling of complex social interactions and information flow.

Authentic Intelligence: Ensures context-aware and meaningful interpretation of social behaviour.

Module 2: AI captures structural and temporal dependencies in social networks using deep learning.

Authentic Intelligence: Ensures responsible interpretation without amplifying bias or misinformation.

Module 3: AI enables contextual understanding and generation of social media content using LLMs.

Authentic Intelligence: Ensures nuanced interpretation of language, emotion, and cultural context.

Module 4: AI enables predictive and adaptive systems for real-time social media insights.

Authentic Intelligence: Ensures ethical, transparent, and user-centric decision-making.

Module 5: AI enables scalable deployment of intelligent social media analytics systems.

Authentic Intelligence: Ensures ethical, inclusive, and socially responsible use of AI outcomes.



Assessment Scheme based on the course

Criteria	Weightage of Marks
AI powered assignments	10
Class Tests /assignments	10
Mid-Semester Examination	30
Hackathon/Technical Competition/ Paper presentation/Global/DLI certifications	10
End-Semester Examination	40
Total	100 marks

Recommended Tools

Primary Textbooks:

1. P. K. Roy, S. Chakraborty, and D. Gupta, Social Network Analytics: Empowering Data Engineering with Deep Learning and LLMs, Routledge, 2024.
2. Reza Zafarani, Mohammad Ali Abbasi, and Huan Liu, Social Media Mining: An Introduction, Cambridge University Press, 2014..

Additional Resources

Additional AI & Computational Resources

1. Python (TensorFlow, PyTorch, Hugging Face Transformers)
2. Network analysis tools (NetworkX, Gephi)
3. Social media datasets (Twitter, Reddit, Instagram APIs)
4. <https://www.coursera.org/specializations/social-media-analytics>

AI-Powered Assignments

1. Sentiment Analysis on Social Media Data (Foundational Lab)

Perform sentiment classification on Twitter/Reddit data

Tech: Logistic Regression / LSTM / BERT

Outcome: Text analytics basics

2. Hashtag Trend Analysis

Analyze trending topics using frequency + time-series patterns

Visualize trend evolution

Outcome: Social media data understanding



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakagalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

3. Social Network Graph Analysis

Construct user interaction graph (followers/mentions)

Perform centrality and community detection Tools: NetworkX / Gephi

Outcome: Graph-based modeling

4. Fake News / Misinformation Detection

Classify posts as real or fake

Tech: NLP + deep learning

Outcome: Content moderation

5. Hate Speech Detection System

Detect toxic or offensive content

Tech: LSTM / BERT

Outcome: Ethical AI applications



Industrial Data Analytics

Course Code:

Semester: VII

Programs Offered: B. Tech (All Branches except DS)

Course Type: Open Elective Course - II

Credits: 03

L – T – P – J: 3-0-0-0

Prerequisites: Python Programming, Basic Machine Learning,

Course Vision

To equip students with the theoretical foundations and practical competencies to Analyze, model, and extract actionable insights from industrial data streams, enabling intelligent decision-making in manufacturing, process industries, and smart infrastructure using open-source analytics frameworks and GPU-accelerated industrial AI platforms.

Course Objectives

1. Understand industrial data characteristics, sensor networks, and time-series data pipelines
2. Apply statistical and machine learning methods for anomaly detection, predictive maintenance, and process optimization
3. Design data-driven condition monitoring and fault diagnosis systems for industrial assets
4. Build real-time industrial analytics pipelines integrating SCADA, IoT, and edge computing frameworks
5. Apply GPU-accelerated analytics and NVIDIA DLI-aligned frameworks for large-scale industrial AI workloads

Course Outcomes (CO)

CO	Description	Blooms Level
C01	Illustrate industrial data types, sensor architectures, and time-series data acquisition and preprocessing pipelines	L2
C02	Apply statistical process control, anomaly detection, and predictive maintenance models to industrial datasets	L3
C03	Design condition monitoring and fault diagnosis systems using machine learning and deep learning techniques	L6
C04	Build real-time industrial analytics pipelines integrating IoT, edge computing, and open-source frameworks	L6
C05	Evaluate industrial AI model performance, reliability, interpretability, and safety in deployment contexts	L5



Broad Course Description

This course provides a comprehensive introduction to data analytics in industrial environments, covering the end-to-end lifecycle of industrial data from acquisition through to actionable insight. Students learn to work with sensor data, time-series signals, and process data from manufacturing, utilities, and smart infrastructure domains.

Core analytical techniques include statistical process control, anomaly detection, predictive maintenance modelling, and deep learning-based fault diagnosis. Students build hands-on experience using open-source frameworks including Python, Pandas, Scikit-learn, PyTorch, and Apache Kafka for real-time streaming analytics.

The course addresses deployment challenges in industrial settings, including edge computing, SCADA integration, model interpretability, and safety-critical constraints. Responsible AI considerations such as explainability of maintenance alerts, fairness in automated quality inspection, and reliability of AI-driven control systems are embedded throughout. The final module is aligned with the NVIDIA Deep Learning Institute (DLI), enabling students to apply GPU-accelerated data processing and deep learning for large-scale industrial AI workloads using RAPIDS, TensorRT, and NVIDIA Triton Inference Server in production-grade industrial pipelines.

CO-PO mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
C01	2	1			1						1	2	2
C02	2	3			2						1	3	2
C03	2	3	3	2	3						2	3	2
C04	2	3	3	2	3						2	3	3
C05	2	3	3	3	3						2	3	3
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module 1: Industrial Data Acquisition and Preprocessing (9 Hours)

Industrial data types (sensor signals, SCADA logs, process variables, event streams), data



acquisition systems (DAQ) and sensor networks, time-series properties (stationarity, seasonality, noise), preprocessing (outlier removal, imputation, normalization, resampling), feature engineering (rolling statistics, frequency-domain and wavelet features), labeling challenges, and datasets (CMAPSS, MIMII, SKAB), along with applications in manufacturing, energy, and machinery monitoring using Pandas, NumPy, Scikit-learn, SciPy, and PyWavelets, and challenges like sensor bias, missing data, labeling uncertainty, and time-series data leakage risks.

Module 2: Statistical Process Control and Anomaly Detection (9 Hours)

Statistical process control (SPC) using control charts (Shewhart, CUSUM, EWMA) and process capability (C_p , C_{pk}), anomaly detection methods (univariate/multivariate, Isolation Forest, One-Class SVM, autoencoders), change point detection (PELT, binary segmentation), alerting systems, evaluation metrics (precision, recall, F1, ROC/PR AUC), and class imbalance handling, with applications in manufacturing, power grids, and water systems using Scikit-learn and PyOD, and challenges like false positives, over-reliance on automation, and explainability of anomaly outputs.

Module 3: Predictive Maintenance and Fault Diagnosis (9 Hours)

Remaining useful life (RUL) prediction (regression, LSTM models), survival and hazard modeling, fault classification (Random Forest, XGBoost, LightGBM, CNN on vibration data), health index and degradation modeling, condition-based maintenance vs reactive/time-based strategies, multi-class imbalanced fault diagnosis, and explainability (SHAP, LIME), with applications in bearings, turbines, and pumps using PyTorch, XGBoost, and SHAP-based dashboards, along with challenges like distribution shift, safety accountability, and risks of false negatives in critical systems.

Module 4: Real-Time Industrial Analytics and IoT Integration (9 Hours)

Real-time streaming architectures (Kafka, MQTT), edge computing for industrial AI (PLCs, edge devices), digital twins and simulation, industrial protocols (OPC-UA, Modbus), SCADA/MES integration, real-time model serving (Flask, FastAPI), model monitoring (drift detection, alerting, retraining), and operator-focused explainability dashboards, with applications in smart factories, predictive energy systems, and autonomous inspection using Kafka pipelines, FastAPI, Grafana, and Streamlit, along with challenges like latency constraints, trust, unsafe automation, and IoT data privacy.

Module 5: GPU-Accelerated Industrial Analytics and NVIDIA DLI-Aligned Deployment

(9 Hours)

GPU-accelerated industrial data processing (RAPIDS: cuDF, cuML), large-scale time-series analytics on GPU clusters, deep learning for fault diagnosis using PyTorch with multi-GPU (DDP), model optimisation for inference (quantisation, pruning, TensorRT), deployment using NVIDIA Triton Inference Server, GPU vs CPU benchmarking, and integration into industrial MLOps pipelines, plus DLI-aligned hands-on work on RAPIDS processing, GPU training, Triton deployment, and end-to-end benchmarking, with applications in predictive maintenance, real-time anomaly detection, and smart manufacturing systems using RAPIDS, TensorRT, Triton, and



Modulus.

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module 1: Students build industrial data preprocessing pipelines using Pandas, SciPy, and PyWavelets. AI-assisted tools are used to explore sensor data patterns, detect stationarity, and engineer time-domain and frequency-domain features.

Module 2: Students apply anomaly detection algorithms from Scikit-learn and PyOD to industrial datasets. AI-generated anomaly scores are critically evaluated against domain knowledge and ground-truth labels.

Module 3: Students train LSTM and CNN models for RUL estimation and fault classification using PyTorch. SHAP-based explainability tools are used to interpret model decisions and validate against engineering intuition.

Module 4: Students build real-time streaming analytics pipelines using Kafka and FastAPI. AI-assisted dashboard design tools are used to create operator-facing monitoring interfaces with interpretable alerts.

Module 5: Students apply NVIDIA RAPIDS for GPU-accelerated industrial data processing and deploy models using Triton Inference Server. NVIDIA DLI-aligned labs are used to benchmark and optimize end-to-end industrial AI pipelines.

Authentic Intelligence Reflection (AI² Pedagogy):

Module 1: Understanding the impact of sensor noise, calibration drift, and missing data on downstream model quality. Students evaluate the ethical dimensions of automated data labelling in fault annotation.

Module 2: Recognising the operational cost of false positives and false negatives in industrial alerting. Students assess when statistical methods are more appropriate than machine learning-based anomaly detection.

Module 3: Evaluating accountability in AI-driven maintenance decisions and the risk of model overconfidence in safety-critical systems. Students reflect on the role of human expertise in validating AI maintenance recommendations.

Module 4: Understanding the risks of automated actuation based on AI recommendations without sufficient human oversight. Students Analyze data privacy and security challenges in industrial IoT deployments.

Module 5: Evaluating the cost and carbon footprint of GPU-accelerated industrial AI infrastructure. Students reflect on governance responsibilities in deploying AI systems in safety-critical industrial environments.



Assessment Scheme based on the course

Criteria	Weightage of Marks
Class Tests	10
Mid-Semester Examination	30
Mini Project	10
Hackathons/Research Publications/Certifications	10
End-Semester Examination	40
Total	100 marks

Recommended Tools

The following tools and platforms are recommended to support AI-integrated teaching, learning, and experimentation:

1. Python 3.10+
2. Pandas, NumPy, SciPy, PyWavelets
3. Scikit-learn, PyOD
4. PyTorch (LSTM, CNN models)
5. XGBoost, LightGBM
6. Apache Kafka, MQTT
7. FastAPI, Streamlit, Grafana
8. NVIDIA RAPIDS (cuDF, cuML)
9. NVIDIA TensorRT
10. NVIDIA Triton Inference Server
11. HPC Lab (GPU computing environment)

Primary Textbooks

1. NVIDIA Corporation, NVIDIA Deep Learning Institute (DLI) Course Materials and Documentation, Latest Edition (Online Resource).
2. Randall Balestriero et al., Industrial Machine Learning, Cambridge University Press, 2023.
3. Peng Gao et al., Deep Learning for Time Series Forecasting, O'Reilly Media, 2020.
4. Ian Goodfellow, Yoshua Bengio, Aaron Courville, Deep Learning, MIT Press, 1st Edition, 2016.
5. Aurélien Géron, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, O'Reilly Media, 3rd Edition, 2022.
6. Ricky Ho, Programming Models for Distributed Computing, O'Reilly Media, Latest Edition.



Additional Resources

1. NVIDIA RAPIDS Documentation – rapids.ai/docs
2. PyTorch Documentation – pytorch.org/docs
3. Apache Kafka Documentation – kafka.apache.org/documentation
4. CMAPSS Turbofan Engine Degradation Dataset – NASA Prognostics Data Repository
5. MIMII Dataset – Malfunctioning Industrial Machine Investigation and Inspection
6. SKAB – Skoltech Anomaly Benchmark Dataset
7. Papers With Code – paperswithcode.com (predictive maintenance and anomaly detection benchmarks)

Mini Project Component

The following projects are suggested based on course outcomes. Students may attempt one project individually or in pairs.

Project 1 – Predictive Maintenance System for Rotating Machinery

Problem: Build a complete predictive maintenance pipeline that estimates RUL and detects faults in rotating machinery using vibration sensor data.

Concepts Used: LSTM-based RUL estimation, CNN fault classification, SHAP explainability, CMAPSS or MIMII dataset.

Industrial Relevance: Automotive, electronics, and pharmaceutical manufacturing automated quality inspection.

Project 2 – Real-Time Anomaly Detection for Industrial Processes

Problem: Design a real-time anomaly detection system for a continuous industrial process using streaming sensor data.

Concepts Used: Autoencoder-based anomaly detection, Apache Kafka streaming, dynamic thresholding, Grafana dashboard.

Industrial Relevance: Co-assembly lines, surgical robot assistance, and warehouse human-robot teaming.

Project 3 – Smart Energy Management Analytics Platform

Problem: Develop an analytics platform that forecasts energy consumption and detects inefficiencies in industrial energy usage.

Concepts Used: Time-series forecasting (LSTM, Prophet), anomaly detection, feature engineering on energy meter data, Streamlit dashboard.

Industrial Relevance: Smart factory adaptive scheduling, personalised production systems, and resilient supply chain management.

Project 4 – Automated Visual Quality Inspection System

Problem: Build an AI-powered visual quality inspection pipeline that classifies surface defects in manufactured components.



Concepts Used: CNN-based image classification, transfer learning, imbalanced dataset handling, model deployment with FastAPI.

Industrial Relevance: Green manufacturing ESG reporting, smart building energy management, and industrial decarbonisation programmes.

Project 5 – Digital Twin Data Analytics for Asset Health Monitoring

Problem: Design a data-driven digital twin prototype that monitors asset health and triggers maintenance alerts using sensor fusion.

Concepts Used: Sensor fusion, health index modelling, degradation curve fitting, real-time dashboard design.

Industrial Relevance: Smart factory digital twin platforms, predictive maintenance integration, and process optimisation systems.

Project 6 – Edge AI Deployment for Industrial Fault Detection

Problem: Deploy a lightweight fault detection model on an edge device and integrate it with an industrial SCADA system.

Concepts Used: Model quantisation, TensorRT optimisation, FastAPI edge serving, MQTT-based SCADA integration.

Industrial Relevance: Occupational health and safety in manufacturing, construction site safety, and logistics worker wellbeing.

Project 7 – GPU-Accelerated Predictive Analytics Pipeline

Problem: Build a GPU-accelerated end-to-end industrial analytics pipeline from raw sensor ingestion to deployed inference using NVIDIA tools.

Concepts Used: NVIDIA RAPIDS (cuDF, cuML), PyTorch DDP, TensorRT, Triton Inference Server, pipeline benchmarking.

Industrial Relevance: Sustainable manufacturing, electronics recycling optimisation, and circular supply chain management.

Project 8 – Responsible AI Audit for an Industrial Predictive Model

Problem: Conduct a responsible AI audit of a deployed predictive maintenance model, evaluating explainability, fairness, and reliability.

Concepts Used: SHAP values, LIME explanations, model drift analysis, reliability metrics, model card documentation.

Industrial Relevance: Factory floor edge AI, real-time HRC safety systems, and GPU-accelerated industrial inspection at scale.

Project 9 – AI-Driven Supply Chain Resilience Analytics

Problem: Design an AI system that detects supply chain disruptions, forecasts demand volatility, and recommends adaptive sourcing strategies.

Concepts Used: LSTM demand forecasting, anomaly detection in supply signals, XGBoost



disruption classifier, supply chain dashboard.

Industrial Relevance: Manufacturing supply chain resilience, pharmaceutical supply management, and Industry 5.0 adaptive sourcing.

Project 10 – Responsible AI Audit for an Industry 5.0 Cobot System

Problem: Conduct a responsible AI audit of a deployed cobot collaboration system evaluating safety, fairness, explainability, and human-centric design.

Concepts Used: Safety standard compliance (ISO/TS 15066), explainability audit, worker demographic fairness analysis, model card documentation.

Industrial Relevance: AI governance in safety-critical industrial robotics, responsible automation deployment, and worker rights in AI-augmented manufacturing.

AI-Powered Assignments

Assignment 1: AI-Based Industrial Sensor Data Quality Analysis

Students prompt an AI tool: Analyze a raw industrial sensor dataset, identify data quality issues, and recommend a preprocessing pipeline.

Students must:

- Evaluate the correctness of AI-identified quality issues (noise, outliers, missing values, drift)
- Identify missing preprocessing steps such as resampling or frequency-domain cleaning
- Improve the pipeline using SciPy and PyWavelets with benchmarking results

Assessment focuses on data engineering skills, signal processing knowledge, and validation of AI-generated preprocessing recommendations.

Assignment 2: AI-Assisted Anomaly Detection Strategy Selection

Students prompt an AI tool: Recommend the most suitable anomaly detection approach for a given industrial time-series dataset and justify the choice.

Students must:

- Analyze the correctness of the AI-recommended method (statistical vs. ML-based vs. deep learning)
- Identify incorrect assumptions about data distribution, seasonality, or label availability
- Validate results using PyOD and evaluate using precision, recall, and F1 on an industrial benchmark

Assessment focuses on analytical reasoning, method selection judgment, and critical evaluation of AI outputs.

Assignment 3: AI-Based Predictive Maintenance Model Design

Students prompt an AI tool: Design a predictive maintenance pipeline for a rotating machinery dataset including feature engineering, model selection, and RUL estimation.

Students must:

- Evaluate the correctness of AI-generated feature sets and model architecture



recommendations

- Identify gaps such as missing degradation indicators or inappropriate model choice for sequential data
- Improve the pipeline using LSTM or CNN models and validate using CMAPSS benchmark metrics

Assessment focuses on predictive maintenance system design, deep learning application, and model interpretability.

Assignment 4: AI-Assisted Fault Diagnosis Classification Pipeline

Students prompt an AI tool: Build a fault classification system for vibration signals and recommend techniques to handle class imbalance.

Students must:

- Analyze AI-suggested class balancing strategies (SMOTE, cost-sensitive learning, data augmentation)
- Identify incorrect handling of temporal dependencies or feature leakage in the AI-generated pipeline
- Improve results using CNN-based classification and SHAP-based interpretability analysis

Assessment focuses on fault diagnosis understanding, imbalanced learning strategies, and explainability.

Assignment 5: AI-Based Real-Time Industrial Pipeline Design

Students prompt an AI tool: Design a real-time industrial analytics pipeline for a streaming sensor data use case.

Students must:

- Analyze the feasibility of the AI-proposed architecture (Kafka ingestion, edge inference, alerting)
- Identify bottlenecks such as latency in model serving, lack of drift monitoring, or missing operator interface
- Refine the design using FastAPI and Kafka with a Grafana-based monitoring dashboard

Assessment focuses on streaming system design, edge deployment reasoning, and real-time pipeline integration.

Assignment 6: AI-Driven GPU-Accelerated Industrial Analytics Evaluation

Students prompt an AI tool: Design and evaluate a GPU-accelerated industrial data analytics pipeline from preprocessing to inference deployment.

Students must:

- Analyze the correctness of all pipeline stages (cuDF preprocessing → cuML model → TensorRT → Triton deployment)
- Identify gaps in deployment strategy (missing latency benchmarking, lack of drift detection, no model monitoring)
- Improve the solution using NVIDIA RAPIDS and Triton Inference Server with throughput and latency profiling



Professional Elective Courses Natural Language Processing (NLP)

Course Code: 26DS3512

Semester: V

Programs: B. Tech CSE (Data Science)

Course Type: Professional Elective (PE-I)

Credits: 03

L – T – P – J: 3 – 0 – 0 – 0

Prerequisites: Machine Learning

Course Vision

To provide foundational and advanced knowledge of Natural Language Processing, including language models, linguistic structures, and modern neural and transformer-based approaches for building intelligent language systems.

Course Objectives

1. Understand fundamental concepts of NLP and text preprocessing techniques
2. Apply statistical and machine learning techniques for language modeling
3. Explore neural network-based models and word embeddings for NLP tasks
4. Analyze transformer-based and large language models and implement linguistic structure analysis
5. Develop real-world NLP applications such as translation, summarization, and speech systems

Course Outcomes (CO)

CO	Description	Blooms Level
CO1	Explain core NLP concepts and text preprocessing techniques	L2
CO2	Apply language modeling and statistical methods for NLP tasks	L3
CO3	Understand word embeddings and neural network models for NLP	L2
CO4	Analyze transformer-based and large language models and implement linguistic structure analysis	L4
CO5	Develop real-world NLP and speech processing applications	L6



Broad Course Description

Natural Language Processing (NLP) is a 3-credit Core course for CSE / AI / DS students in the 6th Semester. It covers the full spectrum of NLP — from foundational text processing and statistical language models to neural embeddings, deep sequence architectures, and transformer-based systems. Students build both theoretical understanding and practical implementation capability.

The course spans five modules: Foundations of NLP and Text Processing, Statistical NLP and Language Modeling, Embeddings and Neural NLP Models, Advanced NLP with Deep Architectures, and NVIDIA DLI Transformer-Based Applications. Students apply modern tools including NLTK, spaCy, and Hugging Face Transformers to build NLP systems for classification, translation, entity recognition, summarization, and question answering.

CO-PO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
C01	2	1			1						1	2	1
C02	2	3			2						1	3	2
C03	2	2			2						1	3	2
C04	2	3	2	2	3						2	3	2
C05	2	3	3	3	3						2	3	3
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module 1: Foundations of NLP and Text Processing

(9 Hours)

Introduction to NLP, Words and tokens, Morphemes and morphology, Unicode and text representation, Regular expressions, Corpora, Tokenization methods, Subword tokenization (Byte Pair Encoding), Minimum edit distance

Module 2: Statistical NLP and Language Modeling

(9 Hours)

N-gram language models, Training and test sets, Evaluation using perplexity, Sampling from language models, Overfitting and generalization, Smoothing techniques, Interpolation, Backoff methods, Logistic regression for NLP, Text classification

Module 3: Embeddings and Neural NLP Models

(9 Hours)



Lexical semantics, Vector semantics, Word embeddings, Word2Vec (CBOW and Skip-gram), Cosine similarity, Evaluation of embeddings, Bias and fairness in embeddings, Feedforward neural networks for NLP, Training neural networks, Embeddings in neural classifiers, GloVe

Module 4: Advanced NLP – Deep Architectures and Applications (9 Hours)

Conditional text generation, Training and evaluation of sequence models, Information retrieval, Retrieval-based models, Machine translation, Encoder-decoder models, Beam search, Recurrent Neural Networks and LSTMs, Sequence modeling techniques

Module 5: NVIDIA DLI – Transformer-Based NLP Applications (9 Hours)

Transformers as building blocks of modern NLP systems, Transformer-based models for text analysis and generation, Working with pre-trained language models, NLP tasks: text classification, Named Entity Recognition (NER), author attribution, summarization, and question answering, Model adaptation and task-specific usage, Evaluation of NLP models in real-world scenarios

Case Study: Implementation of Transformer-based NLP applications for text classification, entity recognition, and question answering using pre-trained models from Hugging Face Transformers

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration

Module 1: Students use AI-assisted tokenization tools (Hugging Face tokenizers, spaCy) to process multilingual text corpora. AI-generated tokenization outputs are critically evaluated for correctness, edge-case handling, and subword segmentation quality.

Module 2: AI tools generate N-gram language model outputs and text classification results. Students critically evaluate perplexity scores, smoothing effects, and classifier performance, identifying overfitting risks and improving model generalization.

Module 3: Students use AI-assisted embedding tools (Word2Vec, GloVe visualizations) to analyze semantic similarity and identify bias in word vector spaces. AI-generated similarity rankings are evaluated for contextual correctness and fairness.

Module 4: AI-powered seq2seq tools are used to explore machine translation and text generation outputs. Students critically analyze beam search results, attention distributions, and LSTM sequence predictions for accuracy and contextual coherence.

Module 5: Students use NVIDIA DLI and Hugging Face pipelines to build transformer-based NLP applications. AI-generated NER, classification, and QA outputs are systematically evaluated for accuracy, context-awareness, and responsible deployment considerations.



Assessment Scheme

Criteria	Weightage of Marks
Class Tests / Assignments	10
Mid-Semester Examination	30
Hackathon / Technical Competition / Paper Presentation / DLI Certifications	10
Mini Project	10
End-Semester Examination	40
Total	100 marks

Recommended Tools

Primary Textbooks

1. Daniel Jurafsky and James H. Martin, Speech and Language Processing, 3rd Edition Draft, Stanford University, 2025. Available at: Speech and Language Processing
2. Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova, BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, arXiv Preprint arXiv:1810.04805, 2018.
3. NLP Libraries Documentation:NLTK Documentation, spaCy Documentation, Hugging Face Documentation
4. Christopher D. Manning and Hinrich Schütze, Foundations of Statistical Natural Language Processing, MIT Press, 1999.
5. NVIDIA Deep Learning Institute – Transformer-Based NLP, NVIDIA Corporation, accessed 2026.
6. Yoav Goldberg, Neural Network Methods for Natural Language Processing, Morgan & Claypool Publishers, 2017.

Additional AI & Computational Resources

1. NPTEL: Natural Language Processing – nptel.ac.in
2. Stanford CS224N: NLP with Deep Learning – web.stanford.edu/class/cs224n
3. Hugging Face NLP Course (free) – huggingface.co/learn/nlp-course
4. Papers With Code – NLP Benchmarks – paperswithcode.com/area/natural-language-processing
5. Google Colab / Jupyter Notebooks – colab.research.google.com



Mini Project Component

The following projects are suggested based on course outcomes. Students may attempt one project individually or in pairs.

Project 1 – Intelligent Text Classification System

Problem: Build a multi-class text classifier (news categories, sentiment, or topic) using Word2Vec embeddings and a neural classifier

Concepts Used: Word embeddings, neural classifiers, tokenization, evaluation metrics

Industrial Relevance: News aggregation, customer support routing, spam filtering

Project 2 – Named Entity Recognition (NER) using Transformers

Problem: Fine-tune a BERT-based model for NER on a domain-specific dataset (medical, legal, or financial)

Concepts Used: Transformer architecture, tokenization, fine-tuning, sequence labeling

Industrial Relevance: Information extraction, contract analysis, clinical record processing

Project 3 – Question Answering System with Pre-trained LLMs

Problem: Implement an extractive QA system using a pre-trained transformer model on a given document corpus

Concepts Used: Transformer-based models, attention mechanism, context grounding, evaluation (F1, EM)

Industrial Relevance: Intelligent search engines, legal document Q&A, enterprise knowledge bases

AI-Powered Assignments

Assignment 1: AI-Assisted Tokenization and Preprocessing Evaluation

Students prompt an AI tool: Preprocess and tokenize this multilingual dataset using BPE and word-level tokenization and compare the resulting vocabularies.

Students must:

- Evaluate whether BPE is appropriate for the language and domain of the dataset
- Identify vocabulary coverage issues, out-of-vocabulary tokens, and segmentation errors
- Improve tokenization by adjusting merge operations and comparing perplexity on held-out data

Assessment focuses on: tokenization reasoning, vocabulary analysis, and critical evaluation of AI-generated preprocessing

Assignment 2: AI-Based Language Model Evaluation

Students prompt an AI tool: Train an N-gram language model on this corpus and evaluate its performance using perplexity. Suggest smoothing techniques.

Students must:

- Verify perplexity calculations and identify where the model fails on rare or unseen n-grams
- Evaluate the suitability of suggested smoothing techniques for the corpus size and domain
- Compare AI-suggested approach against Kneser-Ney and Laplace smoothing empirically

Assessment focuses on: language modeling understanding, smoothing technique evaluation, and empirical validation skills

Assignment 3: AI-Generated NLP Pipeline Critique

Students prompt an AI tool: Build an NLP pipeline for extracting named entities and classifying text from customer reviews.



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Students must:

- Analyze whether the chosen pre-trained model is appropriate for the domain and language
- Identify bias in entity recognition and incorrect classification due to domain mismatch
- Improve the pipeline using domain-specific fine-tuning and evaluation on held-out test data

Assessment focuses on: pipeline design understanding, bias identification, and responsible NLP system evaluation



Optimization Techniques for Data Science

Course Code: 26DS3513

Semester: V

Program(s): B.Tech CSE (Data Science)

Course Type: Professional Elective (PE-I)

Credits: 03

L – T – P – J: 3 – 0 – 0 – 0

Prerequisite(s): Engineering Mathematics (Calculus, Linear Algebra), Basic Programming.

Course Vision

To develop analytical and computational skills in optimization techniques, enabling students to formulate and solve engineering and real-world problems using classical, mathematical, and AI-driven optimization approaches.

Course Objectives

1. Understand the fundamentals of optimization and engineering applications
2. Formulate optimization problems in mathematical programming form
3. Apply linear, nonlinear, and constrained optimization techniques
4. Analyze geometric and dynamic programming methods
5. Develop problem-solving and analytical skills for real-world optimization problems

Course Outcomes (CO)

CO	Description	Blooms Level
C01	Apply the knowledge of design and development of problem-solving skills	L3
C02	Illustrate the principles of optimization	L2
C03	Design and develop analytical skills	L6
C04	Summarize linear, non-linear, and geometric programming techniques	L2
C05	Apply the concept of dynamic programming	L3

Broad Course Description

Geospatial Analytics and GeoAI is an interdisciplinary course that focuses on spatial data processing, analysis, and visualization using modern computational tools. The course introduces students to geospatial data types, coordinate systems, spatial operations, and mapping techniques. It further integrates artificial intelligence with geospatial data to develop intelligent location-based systems. Students gain hands-on experience in solving real-world problems such as urban planning, infrastructure development, and smart city applications through GeoAI techniques.



CO-PO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
CO1	2	1			1						1	2	1
CO2	2	2			1						1	2	1
CO3	2	3	2		2						1	2	2
CO4	2	3	2	2	2						2	2	2
CO5	2	2			1						1	2	1
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module 1: Introduction to Optimization

(9 Hours)

Introduction to optimization; engineering applications of optimization; formulation of structural optimization problems as programming problems; classical optimization techniques; single-variable optimization; multivariable optimization without constraints; unconstrained minimization techniques and algorithms; constrained optimization using penalty function methods; Lagrange multipliers; feasibility techniques.

Module 2: Linear Programming

(9 Hours)

Linear programming; standard form of linear programming; geometry of linear programming problems; solution of linear simultaneous equations; pivot operations for systems of equations; simplex algorithm; revised simplex method; duality in linear programming.

Module 3: Non-Linear Programming

(9 Hours)

Non-linear programming; one-dimensional minimization methods; elimination methods; Fibonacci method; golden section method; interpolation methods including quadratic and cubic; unconstrained optimization methods; direct search methods; random search methods; descent methods.

Module 4: Constrained Optimization Techniques

(9 Hours)

Direct methods; complex methods; cutting plane method; exterior penalty function methods for structural engineering problems; formulation and solution of structural optimization problems using different techniques.

Module 5: Geometric and Dynamic Programming

(9 Hours)

Geometric programming; conversion of nonlinear programming problems into sequences of linear/geometric programming problems; dynamic programming; conversion of nonlinear problems into dynamic programming formulations; applications in engineering systems.



AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration

Module 1: Use of AI-based symbolic tools and optimization libraries to model engineering problems. Visualization of feasible regions and solution landscapes using computational tools. Introduction to convexity detection and automated formulation using AI assistants.

Module 2: Application of linear programming in supervised learning, resource allocation, and classification problems. Use of optimization solvers (SciPy, CVXOPT) in training models and solving real-world datasets.

Module 3: Implementation of gradient descent, stochastic gradient descent (SGD), and Newton-based methods in deep learning. Analysis of convergence, loss surfaces, and optimization stability in neural networks.

Module 4: Use of Lagrangian methods, penalty-based optimization, and constraint satisfaction techniques in AI models. Application in robotics, scheduling, and real-time decision systems.

Module 5: Advanced optimization using Genetic Algorithms (GA), Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), and Reinforcement Learning (RL). Applications in hyperparameter tuning, combinatorial optimization, and dynamic decision-making systems.

Assessment Scheme

Criteria	Weightage of Marks
Class Tests	10
Mid-Semester Examination	30
Mini Project	10
Assignments	10
End-Semester Examination	40
Total	100 marks

Recommended Tools

Primary Textbooks

1. Spunt, "Optimum Structural Design" - Prentice Hall
2. S.S. Rao, "Optimization – Theory and Practice" - Wiley Eastern Ltd.

Additional AI & Computational Resources

1. Uri Krisch, "Optimum Structural Design" - McGraw Hill
2. Richard Bronson, "Operation Research" - Schaum's Outline Series



3. Bhavikatti S.S.- “Structural optimization using sequential linear programming”- Vikas publishing house

Mini Project Component

The following projects are suggested based on course outcomes. Students may attempt one project individually or in pairs.

Project 1 – Supply Chain Optimization System

Problem: Develop an optimization model for logistics and transportation planning to minimize cost and delivery time. Formulate the problem using linear programming and extend it with metaheuristic approaches for large-scale scenarios.

Concepts Used: Linear programming, transportation models, integer optimization, metaheuristics (Genetic Algorithm, Particle Swarm Optimization), constraint handling, operations research techniques

Industrial Relevance: Logistics and supply chain management, warehouse optimization, route planning, inventory control systems

Project 2 – Portfolio Optimization using AI

Problem: Design a portfolio optimization system that balances risk and return using optimization techniques and integrates machine learning predictions for asset performance.

Concepts Used: Mean-variance optimization, risk-return analysis, convex optimization, machine learning for prediction, financial modeling, constraint optimization

Industrial Relevance: Financial analytics, investment advisory systems, wealth management platforms, robo-advisory services

Project 3 – Hyperparameter Optimization for ML Models

Problem: Optimize machine learning model performance by tuning hyperparameters using metaheuristic algorithms such as Genetic Algorithms (GA) and Particle Swarm Optimization (PSO), and compare with traditional grid search.

Concepts Used: Hyperparameter tuning, Genetic Algorithms, Particle Swarm Optimization, search space exploration, model evaluation, cross-validation

Industrial Relevance: AI/ML system optimization, automated machine learning (AutoML), performance tuning in production systems

Project 4 – Smart Traffic Routing Optimization

Problem: Develop an intelligent traffic routing system that finds optimal paths using dynamic programming and AI-based optimization techniques, considering real-time traffic conditions.

Concepts Used: Shortest path algorithms (Dijkstra, A*), dynamic programming, graph theory, optimization techniques, real-time data integration

Industrial Relevance: Smart city systems, navigation applications, traffic management, intelligent transportation systems

AI-Powered Assignments

Assignment 1: Optimization Modeling using Python

Students prompt an AI tool: Formulate a real-world problem as an optimization model and solve it using Python libraries such as SciPy or CVXOPT. Visualize the feasible region and optimal solution.



Students must:

- Evaluate whether the AI-generated formulation correctly defines objective function and constraints
- Identify potential issues: check for incorrect constraint representation, infeasible regions, or solver misuse
- Assess whether the visualization accurately represents the feasible region and optimal solution

Assessment focuses on: Model formulation correctness, solution validity, and visualization accuracy

Assignment 2: Linear Programming for ML Applications

Students prompt an AI tool: Apply linear programming techniques to a machine learning problem (classification or regression) and implement the simplex algorithm.

Students must:

- Evaluate whether the AI-generated LP model correctly represents the ML problem mathematically
- Identify potential misinterpretations: check for incorrect objective formulation or constraint handling
- Assess whether results from simplex implementation align with computational solver outputs

Assessment focuses on: Mathematical modeling accuracy, algorithm implementation, and result validation

Assignment 3: Gradient-Based Optimization Analysis

Students prompt an AI tool: Implement gradient descent and stochastic gradient descent (SGD) for loss minimization and compare with Newton's method.

Students must:

- Evaluate whether the AI-generated implementation correctly computes gradients and updates parameters
- Identify potential issues: check for learning rate instability, divergence, or incorrect convergence behaviour
- Assess whether comparison with Newton's method is valid in terms of convergence speed and computational cost

Assessment focuses on: Optimization correctness, convergence analysis, and comparative evaluation

Assignment 4: Constrained Optimization in AI Systems

Students prompt an AI tool: Solve constrained optimization problems using Lagrange multipliers and penalty function methods and apply them in ML models.

Students must:

- Evaluate whether constraints are correctly incorporated using Lagrangian or penalty methods
- Identify potential misinterpretations: check for violations of feasibility or incorrect handling of constraints
- Assess whether the solution satisfies both optimality and constraint conditions

Assessment focuses on: Constraint handling accuracy, feasibility analysis, and optimization correctness



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Assignment 5: Metaheuristic Optimization

Students prompt an AI tool: Implement optimization using Genetic Algorithm (GA) or Particle Swarm Optimization (PSO) and apply it to real-world datasets.

Students must:

- Evaluate whether the AI-generated implementation correctly applies selection, crossover, mutation (GA) or velocity updates (PSO)
- Identify potential issues: check for premature convergence, poor parameter tuning, or lack of diversity
- Assess whether performance comparison with classical optimization methods is fair and meaningful

Assessment focuses on: Metaheuristic implementation quality, performance evaluation, and comparative analysis



AI for Healthcare

Course Code: 26DS3514

Semester: V

Programs Offered: B. Tech. CSE (Data Science)

Course Type: Professional Elective (PE-I)

Credits: 03

L – T – P – J: 3 – 0 – 0 – 0

Prerequisites: Machine learning and Deep Learning.

Course Vision

To develop intelligent, ethical, and patient-centric healthcare systems using advanced AI techniques, enabling improved diagnosis, treatment, and healthcare delivery.

Course Objectives

1. Apply AI and deep learning techniques to healthcare data and clinical applications.
2. Analyze medical data across specialties for diagnosis and decision support.
3. Design AI-driven solutions for imaging, prediction, and patient monitoring.
4. Integrate emerging technologies such as wearables, VR, and robotics in healthcare.
5. Evaluate ethical, legal, and trust-related challenges in healthcare AI systems.

Course Outcomes (COs)

CO	Description	Blooms Level
C01	Analyze healthcare data systems and AI applications in clinical domains	L4
C02	Apply AI techniques in medical imaging and diagnostics	L3
C03	Develop AI models for clinical decision support and patient monitoring	L6
C04	Design advanced AI-enabled healthcare systems integrating emerging technologies	L6
C05	Evaluate healthcare AI systems considering ethics, trust, and real-world deployment challenges	L5

Broad Course Description

This course provides a comprehensive understanding of how artificial intelligence is transforming the healthcare ecosystem. It introduces students to AI applications across clinical domains such as radiology, oncology, and emergency medicine, along with techniques for medical imaging, predictive analytics, and decision support systems. The course also explores emerging technologies including wearables, robotics, and immersive systems, while emphasizing ethical considerations, trust, and regulatory challenges. Through real-world case studies, students gain insights into building scalable, patient-centric AI solutions for modern healthcare systems.



CO-PO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	2	3			2	1					1	3	2
CO2	2	3			3	1					1	3	2
CO3	2	3	3	2	3	1					2	3	2
CO4	2	3	3	3	3	1					2	3	3
CO5	2	3	3	3	2	2	2				2	3	3
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module 1: AI Applications Across Clinical Domains

(9 Hours)

AI in primary care and preventive healthcare; applications in ophthalmology, oncology, and radiology; AI in emergency medicine and ICU settings; cardiovascular analytics and surgical support systems; data extraction and quality control in Electronic Health Records (EHRs).
Applications: Early disease detection, ICU monitoring, cancer diagnosis, clinical data management.

Module 2: Medical Imaging and Diagnostic Intelligence

(9 Hours)

AI for automated medical image interpretation; deep learning models for radiology and pathology; image segmentation, detection, and classification; multimodal data integration (imaging and clinical data); software systems for diagnostic support.
Applications: Tumor detection, retinal disease diagnosis, radiology automation.

Module 3: Clinical Decision Support & Predictive Healthcare

(9 Hours)

AI-based clinical decision support systems (CDSS); predictive modeling for disease progression and risk assessment; personalized medicine and treatment recommendation systems; healthcare data mining and knowledge discovery; conversational AI and virtual assistants in healthcare.
Applications: Treatment planning, risk prediction, patient triage, virtual health assistants.

Module 4: Emerging Technologies in Healthcare AI

(9 Hours)

Wearable technology and remote patient monitoring; smart implantable and connected healthcare systems; virtual, augmented, and mixed reality in therapy and training; robotics in healthcare (surgery, rehabilitation, assistance); blockchain for healthcare data security and interoperability.
Applications: Remote monitoring, robot-assisted surgery, rehabilitation systems.

Module 5: Future, Ethics, and Case Studies in Healthcare AI

(9 Hours)

Ethical challenges in AI-driven healthcare; legal and regulatory frameworks; building patient trust in AI systems; healthcare finance and impact of AI; future trends including value-based care,



evidence-based medicine, and personalized healthcare. Case studies include AI-based cancer detection systems, remote patient monitoring using wearables, and AI-powered virtual assistants for patient engagement and triage. *Applications:* Policy making, digital health platforms, patient-centric care systems.

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

- **Module 1:** AI enables data-driven clinical insights across multiple medical specialties. **Authentic Intelligence:** Ensures AI supports informed, patient-centric clinical decision-making.
- **Module 2:** AI enhances accuracy and efficiency in medical imaging diagnostics. **Authentic Intelligence:** Promotes explainable and trustworthy AI for clinician interpretability.
- **Module 3:** AI enables predictive analytics and personalized healthcare interventions. **Authentic Intelligence:** Ensures transparent, reliable, and patient-aligned decision support.
- **Module 4:** AI integrates with emerging technologies for continuous and connected healthcare systems. **Authentic Intelligence:** Ensures accessibility, inclusiveness, and real-world usability.
- **Module 5:** AI enables scalable and intelligent healthcare delivery systems. **Authentic Intelligence:** Ensures ethical, trustworthy, and human-centered healthcare solutions.

Assessment Scheme

Criteria	Weightage of Marks
Project	10
Class Tests /assignments	10
Mid-Semester Examination	30
Hackathon/Technical Competition/ Paper presentation/Global/DLI certifications	10
End-Semester Examination	40
Total	100 marks



Recommended Tools

Primary Textbooks:

1. Kilic, A. – *Artificial Intelligence and Machine Learning in Healthcare*, Elsevier.
2. Cannataro, M. – *The Evolution of Artificial Intelligence in Healthcare*, Elsevier.
3. Malawana et al. – *Artificial Intelligence in Healthcare and Biomedical Visualization*, Springer.
4. Adirim, T. – *Digital Health, AI and Generative AI in Healthcare*, Springer.

Additional AI & Computational Resources

1. Python (TensorFlow, PyTorch), MATLAB
2. Healthcare datasets (MIMIC, NIH datasets)
3. NVIDIA AI Platforms for Healthcare and Life Sciences:- MONAI: Medical Open Network for Artificial Intelligence, <https://project-monai.github.io/>
4. <https://www.coursera.org/specializations/ai-healthcare>

Project Component

Project 1- Medical Image Classification (X-ray / MRI)

Detect pneumonia, tumors, or fractures

Tech: CNN (ResNet, EfficientNet)

Dataset: Chest X-ray datasets

Outcome: Imaging AI

Project 2- ECG Signal Classification

Classify normal vs abnormal heart rhythms

Tech: CNN / LSTM

Dataset: MIT-BIH ECG

Focus: Time-series healthcare data

Project 3- Patient Risk Prediction (Time-Series)

Predict ICU mortality or deterioration

Tech: LSTM / GRU

Dataset: MIMIC-III

Outcome: Predictive healthcare

Project 4- Clinical Decision Support System (CDSS)

Suggest treatments based on patient data

Tech: ML + rule-based hybrid

Focus: Practical healthcare AI

Project 5- NLP for Clinical Text Analysis

Extract insights from clinical notes

Tech: NLP (BERT, BioBERT)

Tasks: Named entity recognition, classification



AI for Industry 5.0

Course Code: 26DS3515

Semester: V

Programs Offered: B. Tech CSE (Data Science)

Course Type: Professional Elective (PE-I)

Credits: 3

L – T – P – J: 3-0-0-0

Prerequisites: Python Programming, Basic Machine Learning,

Course Vision

To equip students with the knowledge, practical skills, and ethical reasoning required to design and deploy AI systems for Industry 5.0 environments, where human-centric intelligence, collaborative robotics, and sustainable manufacturing converge, leveraging open-source AI frameworks and NVIDIA DLI-aligned GPU-accelerated platforms for next-generation industrial applications.

Course Objectives

1. Understand the principles of Industry 5.0, human-AI collaboration, and cyber-physical systems integration
2. Apply AI and machine learning techniques for intelligent manufacturing, quality control, and adaptive production systems
3. Design human-robot collaboration (HRC) systems using computer vision, gesture recognition, and intent prediction
4. Build AI-powered sustainability analytics and circular economy monitoring systems
5. Evaluate AI systems in Industry 5.0 contexts for safety, explainability, fairness, and human-centric design
6. Apply GPU-accelerated AI and NVIDIA DLI-aligned frameworks for real-time industrial perception and edge AI deployment.

Course Outcomes

CO	Description	Blooms Level
CO1	Illustrate Industry 5.0 principles, human-AI collaboration models, and cyber-physical systems integration frameworks.	L2
CO2	Apply AI and ML techniques for intelligent manufacturing, adaptive quality control, and process optimization.	L3
CO3	Design human-robot collaboration systems using computer vision, gesture recognition, and AI-based intent prediction.	L6



C04	Build AI-powered sustainability analytics, digital twin systems, and circular economy monitoring platforms.	L6
C05	Apply GPU-accelerated AI and NVIDIA DLI-aligned tools for real-time industrial perception and edge AI deployment	L5

Broad Course Description

This course introduces students to the paradigm of Industry 5.0, which extends Industry 4.0 automation with a focus on human-centric intelligence, collaborative robotics, and sustainable manufacturing. Students learn how AI enables symbiotic human-machine interaction, adaptive production, and resilient supply chains in next-generation industrial environments.

Core topics include AI for intelligent manufacturing and quality control, computer vision for human-robot collaboration, reinforcement learning for adaptive production scheduling, and AI-driven sustainability and circular economy analytics. Students build hands-on skills using open-source frameworks including PyTorch, OpenCV, Gymnasium, ROS 2, and Streamlit.

The course emphasises the human-centric dimension of Industry 5.0, addressing worker safety, operator trust in AI systems, inclusive design of human-robot interfaces, and the ethical implications of AI-driven automation on workforce wellbeing. Responsible AI principles including explainability, fairness, and safety are embedded throughout all modules.

The final module is aligned with the NVIDIA Deep Learning Institute (DLI), enabling students to apply GPU-accelerated computer vision, real-time edge AI inference, and large-scale industrial perception pipelines using NVIDIA Jetson, DeepStream, Triton Inference Server, and Omniverse for Industry 5.0 simulation and deployment.

CO-PO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
C01	2	1			1	1					1	2	2
C02	2	3			3						1	3	2
C03	2	3	3	2	3						2	3	3
C04	2	3	3	2	3	2					2	3	3
C05	2	3	3	3	3						2	3	3
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course-Modules

Module 1: Foundations of Industry 5.0 and Human-AI Collaboration

(9 Hours)



Evolution to Industry 5.0 (human-centric, sustainable, resilient systems), cyber-physical systems (CPS) and digital-physical integration, human-AI collaboration models (tool, collaborator, autonomous agent), human-AI teaming (shared/variable autonomy, human-on-the-loop), cobots and collaborative robotics (ROS 2, safety standards ISO/TS 15066), human factors (trust, cognitive load, situational awareness), and Industry 5.0 applications (smart factories, personalised manufacturing, sustainable supply chains), with AI-enabled workflows (ROS 2 simulation, Python-based modelling) and ethical considerations (reskilling, safety, trust, and human-AI decision accountability).

Module 2: AI for Intelligent Manufacturing and Quality Control (9 Hours)

AI-driven process optimisation (regression models, Bayesian optimisation, Gaussian processes), computer vision for quality inspection (CNNs, transfer learning, self-/few-shot learning), defect and anomaly detection (autoencoders, one-class classifiers), generative AI for synthetic defect data (GANs, diffusion models), multi-sensor data fusion, and model interpretability (Grad-CAM, SHAP). Applications in PCB, pharmaceutical, and automotive inspection, AI pipelines using PyTorch and augmentation tools (Albumentations), and challenges including over-rejection risk, regulatory explainability needs, and dataset bias from limited production data.

Module 3: Human-Robot Collaboration and Adaptive Interaction (9 Hours)

Human-robot collaboration using computer vision (pose estimation and gesture recognition with MediaPipe/OpenPose), intent and activity recognition (LSTM and transformer models), safe HRC systems (collision avoidance, speed and separation monitoring, force limiting), adaptive robot behaviour using reinforcement learning (Stable-Baselines, Gymnasium), multimodal interaction (voice, gaze, haptics), evaluation metrics (task time, error rate, cognitive load, NASA-TLX), and explainable HRC (natural language explanations). Applications in assembly, surgery, and warehouse automation with ROS 2 and AI tools, along with challenges in safety liability, worker psychology, and fairness in collaborative robot systems.

Module 4: AI for Sustainability, Digital Twins, and Circular Economy (9 Hours)

AI for sustainability (energy forecasting, carbon footprint modelling, waste reduction), digital twins for Industry 5.0 (real-time simulation, predictive modelling), physics-informed ML (PINNs), circular economy analytics (material flow, lifecycle and end-of-life prediction), supply chain resilience (demand forecasting, disruption detection, adaptive sourcing), and occupational safety analytics (fatigue, ergonomics, hazard prediction), plus sustainability dashboards. Applications in green manufacturing, supply chains, and smart buildings using tools like Prophet, LSTM, Eclipse Ditto, and Streamlit, with challenges around carbon reporting accuracy, greenwashing risks, and worker surveillance concerns.

Module 5: GPU-Accelerated Industrial AI and NVIDIA DLI-Aligned Edge Deployment

(9 Hours)

GPU-accelerated industrial AI (RAPIDS, TensorRT), real-time computer vision and video analytics for safety (NVIDIA DeepStream), edge AI deployment using NVIDIA Jetson, model optimisation for edge (INT8 quantisation, pruning, TensorRT), multi-model deployment with NVIDIA Triton Inference Server, and digital twin simulation with NVIDIA Omniverse for synthetic data



generation and Industry 5.0 workflows. Hands-on DLI-aligned work covering object detection for HRC safety, edge-based defect inspection, Omniverse-driven synthetic datasets, and industrial AI pipelines, with applications in cobot safety, real-time quality inspection, and digital twin-driven manufacturing systems.

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module 1: Students model human-AI collaboration workflows using ROS 2 cobot simulation and Python-based task planning tools. AI-generated collaboration strategies are critically evaluated against worker safety standards and human factors guidelines.

Module 2: Students build computer vision quality inspection pipelines using PyTorch CNN and transfer learning. AI-generated defect detection results are evaluated for precision-recall trade-offs and bias in training data from limited production runs.

Module 3: Students implement human pose estimation and intent prediction pipelines using MediaPipe and LSTM models. AI-driven robot motion decisions are critically evaluated for safety, operator trust, and explainability.

Module 4: Students build sustainability analytics dashboards and digital twin prototypes using open-source tools. AI-generated sustainability metrics and carbon forecasts are evaluated for accuracy, interpretability, and potential for greenwashing.

Module 5: Students deploy real-time industrial AI models on GPU infrastructure using NVIDIA DeepStream, Jetson, and Triton. NVIDIA DLI-aligned labs are used to benchmark edge AI performance and optimize production-grade Industry 5.0 pipelines.

Authentic Intelligence Reflection (AI² Pedagogy):

Module 1: Understanding the ethical responsibilities in deploying AI that collaborates with human workers, including safety certification, workforce reskilling, and transparent communication of AI capabilities and limitations.

Module 2: Recognising the risks of AI over-rejection or under-rejection in quality control systems and the regulatory and liability implications of AI-driven quality decisions in safety-critical manufacturing.

Module 3: Evaluating psychological and safety impacts on workers collaborating with intelligent robots. Students reflect on inclusive HRC design that accommodates diverse worker demographics and physical abilities.

Module 4: Understanding the risks of AI-driven sustainability metrics being used for greenwashing and the surveillance implications of worker fatigue and ergonomics monitoring systems.

Module 5: Evaluating the governance of real-time AI systems deployed in safety-critical



industrial environments. Students reflect on the accountability of edge AI decisions made without human oversight in high-speed manufacturing settings.

Assessment Scheme

Criteria	Weightage of Marks
Class Tests	10
Mid-Semester Examination	30
Mini Project	10
Hackathon/Research Publication/Certifications	10
End-Semester Examination	40
Total	100 marks

Recommended Tools

1. Python 3.10+
2. PyTorch (deep learning: CNN, LSTM, RL)
3. OpenCV & MediaPipe (computer vision, pose estimation)
4. ROS 2 (cobots & human-robot collaboration)
5. Stable-Baselines3 & Gymnasium (reinforcement learning)
6. Streamlit (industrial dashboards & applications)
7. Eclipse Ditto (digital twins)
8. NVIDIA ecosystem: DeepStream, Jetson, Omniverse, Triton Inference Server
9. HPC / GPU computing environment

Primary Textbooks

1. NVIDIA Corporation, NVIDIA Deep Learning Institute (DLI) Course Materials and Documentation, Latest Edition (Online Resource).
2. European Commission, Industry 5.0: Towards a Sustainable, Human-Centric and Resilient European Industry, Publications Office of the EU, 2021.
3. Bruno Siciliano, Lorenzo Sciavicco, Luigi Villani, Giuseppe Oriolo, Robotics: Modelling, Planning and Control, Springer, 2009.
4. Ian Goodfellow, Yoshua Bengio, Aaron Courville, Deep Learning, MIT Press, 1st Edition, 2016.
5. Aurélien Géron, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, O'Reilly Media, 3rd Edition, 2022.
6. Richard S. Sutton, Andrew G. Barto, Reinforcement Learning: An Introduction, MIT Press, 2nd Edition, 2018.

Additional Resources

1. NVIDIA DeepStream Documentation – developer.nvidia.com/deepstream-sdk



2. NVIDIA Jetson Developer Documentation – developer.nvidia.com/embedded-computing
3. NVIDIA Omniverse Documentation – developer.nvidia.com/nvidia-omniverse-platform
4. ROS 2 Documentation – docs.ros.org
5. Eclipse Ditto Documentation – eclipse.dev/ditto
6. MVTec AD – Industrial Anomaly Detection Dataset
7. Papers With Code – paperswithcode.com (industrial inspection and human-robot collaboration benchmarks)

Mini Project Component

Industry-Driven Projects:

Project 1 – AI-Powered Surface Defect Inspection System

Problem: Build a real-time surface defect detection system for a manufacturing production line using computer vision.

Concepts Used: CNN-based defect detection, transfer learning, GradCAM explainability, MVTec AD dataset, FastAPI deployment.

Project 2 – Human-Robot Collaboration Safety System

Problem: Design an HRC safety system that detects human presence, predicts intent, and adapts cobot speed and trajectory in real time.

Concepts Used: MediaPipe pose estimation, LSTM intent prediction, ROS 2 cobot integration, speed and separation monitoring.

Project 3 – Reinforcement Learning for Adaptive Production Scheduling

Problem: Develop an RL-based production scheduler that adapts to machine failures, demand shifts, and workforce availability.

Concepts Used: Stable-Baselines 3 RL agent, Gymnasium manufacturing environment, reward shaping for production KPIs.

Project 4 – AI-Powered Sustainability and Carbon Analytics Platform

Problem: Build a sustainability analytics platform that forecasts energy consumption, tracks carbon emissions, and identifies reduction opportunities.

Concepts Used: LSTM and Prophet-based energy forecasting, carbon footprint modelling, Streamlit dashboard, open energy datasets.

Project 5 – Digital Twin for Process Monitoring and Predictive Simulation

Problem: Design and prototype a digital twin for a manufacturing process that enables real-time monitoring, anomaly detection, and what-if simulation.

Concepts Used: Eclipse Ditto digital twin, LSTM anomaly detection, predictive simulation, Streamlit operator dashboard.

Project 6 – Worker Wellbeing Monitoring System Using Computer Vision

Problem: Develop an AI system that monitors worker fatigue, ergonomics, and safety compliance



using computer vision with privacy safeguards.

Concepts Used: MediaPipe pose estimation for ergonomics, fatigue detection from facial landmarks, anonymisation techniques, responsible AI audit.

Project 7 – Circular Economy Material Flow Analytics System

Problem: Build an analytics system that tracks material flows across a product lifecycle and identifies circular economy opportunities.

Concepts Used: Material flow analysis, ML-based end-of-life prediction, supply chain analytics, Streamlit circular economy dashboard.

Project 8 – Real-Time Edge AI Deployment for Industrial Perception

Problem: Deploy a real-time computer vision model for industrial safety and inspection on an edge device using NVIDIA Jetson.

Concepts Used: NVIDIA DeepStream, TensorRT optimisation, NVIDIA Jetson deployment, Triton Inference Server, latency benchmarking.

Project 9 – AI-Driven Supply Chain Resilience Analytics

Problem: Design an AI system that detects supply chain disruptions, forecasts demand volatility, and recommends adaptive sourcing strategies.

Concepts Used: LSTM demand forecasting, anomaly detection in supply signals, XGBoost disruption classifier, supply chain dashboard.

Project 10 – Responsible AI Audit for an Industry 5.0 Cobot System

Problem: Conduct a responsible AI audit of a deployed cobot collaboration system evaluating safety, fairness, explainability, and human-centric design.

Concepts Used: Safety standard compliance (ISO/TS 15066), explainability audit, worker demographic fairness analysis, model card documentation.

AI-Powered Assignments

Assignment 1: AI-Based Human-AI Collaboration Workflow Analysis

Students prompt an AI tool: Design an AI-assisted cobot workflow for a manufacturing assembly task and Analyze the optimal level of human-AI shared autonomy.

Students must:

- Evaluate the correctness of AI-generated workflow design against ISO/TS 15066 cobot safety standards
- Identify missing human factors considerations such as cognitive load, trust calibration, and operator override mechanisms
- Improve the workflow design using ROS 2 simulation and human-in-the-loop evaluation metrics

Assessment focuses on human-AI collaboration design, safety standard awareness, and human-centric system thinking.

Assignment 2: AI-Assisted Quality Inspection Model Design

Students prompt an AI tool: Design a computer vision quality inspection pipeline for surface



defect detection and justify the model and augmentation strategy. Students must:

- Analyze the correctness of AI-recommended model architecture (CNN, transfer learning) and data augmentation strategy
- Identify risks of AI over-rejection and the impact of training data bias from limited production runs
- Improve the pipeline using GradCAM explainability and evaluate precision-recall trade-offs on an industrial dataset

Assessment focuses on quality inspection system design, explainability requirements, and regulatory awareness.

Assignment 3: AI-Based Human Intent Prediction for HRC Safety

Students prompt an AI tool: Design an intent prediction system for a human-robot collaboration scenario and evaluate its safety and adaptability.

Students must:

- Analyze AI-generated activity recognition approaches (LSTM vs. transformer-based) for HRC intent prediction
- Identify safety risks if intent prediction fails and propose fallback mechanisms
- Improve the system using LSTM models trained on skeleton sequence data and evaluate using task completion and error rate metrics

Assessment focuses on HRC safety system design, intent modelling, and responsible AI in robotic systems.

Assignment 4: AI-Assisted Sustainability Analytics Dashboard Design

Students prompt an AI tool: Design an AI-powered sustainability analytics dashboard for a manufacturing facility tracking energy consumption and carbon footprint.

Students must:

- Evaluate the correctness of AI-generated energy forecasting approach and carbon metric calculations
- Identify risks of greenwashing driven by inaccurate AI sustainability metrics
- Improve the dashboard using Prophet-based forecasting and Streamlit visualisation with transparent metric definitions

Assessment focuses on sustainability AI design, carbon reporting accuracy, and responsible use of AI for ESG analytics.

Assignment 5: AI-Based Digital Twin Design for Process Optimisation

Students prompt an AI tool: Design a digital twin architecture for a manufacturing process that enables real-time monitoring and predictive simulation.

Students must:

- Analyze the feasibility of AI-proposed digital twin architecture (real-time sync, predictive simulation, anomaly detection)
- Identify bottlenecks such as data latency, model-reality gaps, or insufficient explainability for operators
- Refine the design using Eclipse Ditto and a Python-based predictive simulation layer

Assessment focuses on digital twin architecture reasoning, simulation accuracy, and operator-



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

centric design.

Assignment 6: AI-Driven GPU-Accelerated Industrial Perception Pipeline Evaluation

Students prompt an AI tool: Design and evaluate a GPU-accelerated real-time industrial perception pipeline for HRC safety and quality inspection.

Students must:

- Analyze all pipeline stages (DeepStream video ingestion → TensorRT inference → Triton serving → operator alert)
- Identify gaps in real-time safety guarantees, latency constraints, and edge deployment feasibility
- Improve the solution using NVIDIA Jetson edge deployment and Triton multi-model serving with latency benchmarking

Assessment focuses on real-time industrial AI pipeline design, NVIDIA DLI toolchain integration, and safety-critical deployment reasoning.



Cryptography and Network Security

Course Code: 26DS3516

Semester: V

Programs Offered: B.Tech. CSE (Data Science)

Course Type: Professional Elective (PE-I)

Credits: 3

L – T – P – J: 3 – 0 – 0 – 0

Prerequisite(s): Computer Networks

Course Vision

To develop the ability to design secure communication systems by integrating cryptographic techniques, network security mechanisms, and modern cybersecurity practices, enabling protection of digital data and infrastructure in real-world applications.

Course Objectives

1. Develop a strong understanding of cryptographic fundamentals and enable secure data encryption and decryption techniques.
2. Build proficiency in symmetric and asymmetric cryptography and enable secure key management and authentication.
3. Equip students with knowledge of modern security protocols and enable analysis of network vulnerabilities and attacks.
4. Provide a strong foundation in system security, authentication, and access control mechanisms.
5. Enable students to analyze emerging cybersecurity technologies and apply secure design principles in real-world systems.

Course Outcomes (CO)

CO	Description	Blooms Level
C01	Students will be able to explain cryptographic concepts and security principles.	L2
C02	They will be able to apply symmetric and asymmetric encryption techniques for secure communication.	L3
C03	Students will gain the ability to analyze network security protocols and identify threats.	L4
C04	They will be able to evaluate system vulnerabilities and implement security mechanisms.	L5
C05	They will be able to design secure communication systems using modern cybersecurity techniques.	L6



Broad Course Description

Cryptography and Network Security is a core course that provides a comprehensive understanding of secure communication systems and cybersecurity principles. The course covers classical and modern cryptographic techniques, symmetric and asymmetric encryption, network security protocols, and system security mechanisms. It focuses on how data can be protected using encryption, authentication, and secure communication protocols.

The course integrates modern cybersecurity practices including intrusion detection, secure communication protocols, and emerging technologies such as blockchain and cloud security. Students learn to analyze threats, design secure systems, and evaluate vulnerabilities. The course emphasizes practical understanding, system-level security design, and real-world applications in digital security and cyber defense.

CO-PO mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
C01	2	1			1						1	2	1
C02	2	3			2						1	2	1
C03	2	3			2						1	2	1
C04	2	3	2	3	2						2	2	2
C05	2	3	3	3	2						2	2	2
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module - 1: Foundations of Cryptography

(9 Hours)

Introduction to cryptography, security goals including confidentiality integrity and availability, classical encryption techniques, substitution ciphers, transposition ciphers, Caesar cipher, Playfair cipher, Hill cipher, Vigenère cipher, cryptanalysis techniques, frequency analysis, mathematical foundations of cryptography, modular arithmetic, prime numbers, greatest common divisor, Euler's theorem.

Module - 2: Symmetric Key Cryptography

(9 Hours)

Block cipher principles, Feistel structure, Data Encryption Standard, Advanced Encryption Standard, modes of operation, Electronic Codebook, Cipher Block Chaining, Counter Mode, stream ciphers, key generation and management, random number generation, confusion and diffusion principles.

Module - 3: Public Key Cryptography

(9 Hours)

Public key cryptography concepts, RSA algorithm, key generation encryption and decryption, Diffie-Hellman key exchange, ElGamal cryptosystem, digital signatures, hash functions, message authentication codes, Secure Hash Algorithm, elliptic curve cryptography basics.



Module - 4: Network Security and Protocols

(9 Hours)

Network security concepts, types of attacks, passive and active attacks, denial of service attack, man in the middle attack, firewalls, intrusion detection systems, intrusion prevention systems, Secure Sockets Layer, Transport Layer Security, IPsec, virtual private networks, email security protocols, secure file transfer protocols.

Module - 5: System Security and Emerging Trends

(9 Hours)

Authentication mechanisms, password security, biometric authentication, access control models, malware types including viruses worms trojans and ransomware, security policies, risk management, cyber laws and ethics, digital forensics basics, blockchain security, cloud security, Internet of Things security.

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module 1: Students use AI tools to analyze classical encryption techniques and perform frequency analysis.

Module 2: Simulation tools are used to analyze symmetric encryption algorithms and evaluate performance.

Module 3: AI-assisted tools are used to simulate public key algorithms and digital signatures.

Module 4: Students use AI-based intrusion detection tools to analyze network attacks and security threats.

Module 5: AI-based cybersecurity tools are used to detect malware and analyze security risks in modern systems.

Assessment Scheme

Criteria	Weightage of Marks
Assignment	10
Class Tests / Quizzes	5
Mid-Semester Examination	30
Project	15
End-Semester Examination	40
Total	100 marks

Recommended Tools

The following tools and platforms are recommended to support AI-integrated teaching, learning,

1. Python
2. OpenSSL
3. Wireshark
4. Kali Linux
5. Network simulation tools



Primary Textbooks

1. William Stallings, Cryptography and Network Security: Principles and Practice, Pearson, 2017
2. William Stallings, Network Security Essentials, Pearson, 2017
3. Jonathan Katz and Yehuda Lindell, Introduction to Modern Cryptography, CRC Press, 2020
4. Bruce Schneier, Applied Cryptography, Wiley, 1996
5. William Stallings and Lawrie Brown, Computer Security: Principles and Practice, Pearson, 2018

Mini Project Component

Industry-Driven Projects

Project 1 – Secure Messaging System

Problem: Design a system for secure communication using encryption techniques

Concepts Used: Symmetric and asymmetric encryption, key exchange, authentication

Industrial Relevance: Messaging platforms, secure communication systems, Defense applications

Project 2 – Encrypted File Transfer System

Problem: Develop a system to securely transfer files over a network

Concepts Used: Encryption algorithms, secure protocols, authentication mechanisms

Industrial Relevance: Cloud storage, enterprise data transfer, secure file systems

Project 3 – Network Intrusion Detection System

Problem: Design a system to detect malicious activities in a network

Concepts Used: Network protocols, intrusion detection, attack analysis

Industrial Relevance: Cybersecurity systems, enterprise network monitoring, threat detection

Project 4 – Authentication System using Cryptography

Problem: Develop a secure user authentication mechanism

Concepts Used: Password security, hashing, digital signatures, access control

Industrial Relevance: Banking systems, login authentication, secure access systems

Project 5 – Malware Detection System

Problem: Design a system to detect and classify malware threats

Concepts Used: Malware analysis, pattern recognition, security policies

Industrial Relevance: Antivirus systems, cybersecurity Defense, threat intelligence

AI-Powered Assignments

Assignment 1: AI-Based Cryptographic Algorithm Validation

Students prompt an AI tool: Generate encryption and decryption using classical or modern cryptographic algorithms.

Students must:

- Analyze the correctness of encryption and decryption outputs
- Identify weaknesses or vulnerabilities in the algorithm
- Validate results using manual computation or standard cryptographic tools



- Assessment focuses on understanding of encryption techniques, correctness of outputs, and ability to validate AI-generated solutions

Assignment 2: AI-Assisted Symmetric Encryption Performance Analysis

Students prompt an AI tool: Compare performance of symmetric encryption algorithms such as AES and DES.

Students must:

- Analyze execution time, security strength, and efficiency
- Identify incorrect assumptions in performance comparison
- Validate results using simulation tools or benchmarking
- Assessment focuses on performance evaluation, algorithm comparison, and analytical reasoning

Assignment 3: AI-Based Public Key Cryptography Simulation

Students prompt an AI tool: Implement RSA algorithm and perform key generation, encryption, and decryption.

Students must:

- Analyze correctness of key generation and encryption process
- Identify errors in modular arithmetic or key selection
- Validate results using mathematical calculations and tools
- Assessment focuses on understanding of public key systems, correctness, and validation of AI outputs

Assignment 4: AI-Based Network Attack Detection and Analysis

Students prompt an AI tool: Identify types of network attacks and simulate detection mechanisms.

Students must:

- Analyze correctness of attack classification
- Identify missing threats or incorrect detection methods
- Validate using network simulation tools or datasets
- Assessment focuses on network security understanding, threat analysis, and validation of AI results

Assignment 5: AI-Based Malware Detection and Classification

Students prompt an AI tool: Detect and classify malware types based on behaviour or patterns.

Students must:

- Analyze correctness of classification results
- Identify false positives or missed detections
- Validate using known malware datasets or security tools



Generative AI and Large Language Models

Course Code: 26DS3607

Semester: VI

Program(s): B.Tech. CSE (Data Science)

Course Type: Professional Elective (PE-II)

Credits: 3

L – T – P – J: 3 – 0 – 0 – 0

Prerequisite(s): Machine Learning, Basic NLP

Course Vision

To provide comprehensive knowledge of Large Language Models and Generative AI, focusing on advanced transformer architectures, contextual representations, fine-tuning techniques, and modern generative systems for building scalable AI applications.

Course Objectives

1. Understand the evolution from traditional NLP to modern LLM-based systems
2. Analyze advanced embeddings and transformer architectures
3. Apply LLMs for real-world text-based and generative applications
4. Implement fine-tuning and optimization techniques for LLMs
5. Explore prompt engineering, agent systems, and multimodal generative AI

Course Outcomes (CO)

CO	Description	Blooms Level
C01	Explain advanced concepts of LLMs and Generative AI	L2
C02	Analyze contextual embeddings and transformer architecture	L4
C03	Apply LLMs for advanced text-based and generative applications	L3
C04	Implement fine-tuning and optimization techniques	L3
C05	Design prompt-based and agent-driven AI systems	L6

Broad Course Description

Large Language Models and Generative AI is a 3-credit course for CSE-DS students. It builds a comprehensive understanding of the transformer-based ecosystem — from foundational NLP evolution and contextual embeddings to fine-tuning, prompt engineering, and agent-driven system design. Students engage with both conceptual frameworks and hands-on implementation.



The course spans five modules: Foundations of LLM and Generative AI, Contextual Embeddings and Representation Learning, Transformer Architecture and LLM Internals, Applications with Fine-Tuning and Generative Extensions, and Prompt Engineering with Agents and NVIDIA DLI Case Study. Students develop skills to build, evaluate, and deploy scalable LLM-based applications using modern tools including Hugging Face Transformers, LangChain, and NVIDIA NIM.

CO-PO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
C01	2	1			1						1	3	2
C02	2	3			2						1	3	2
C03	2	3			3						2	3	3
C04	2	3	3	2	3						2	3	3
C05	2	3	3	3	3						2	3	3
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module 1: Foundations of LLM and Generative AI (9 Hours)

Evolution from traditional NLP to Large Language Models, Limitations of traditional NLP approaches, Emergence of transformer-based systems, Overview of Generative AI, Generative vs discriminative models, Applications of LLMs, Responsible AI usage, Open vs proprietary models, Overview of generative models: GANs, diffusion models, transformers, Introduction to multimodal AI systems

Module 2: Contextual Embeddings and Representation Learning (9 Hours)

Contextual embeddings in LLMs, Sentence and document embeddings, Transformer-based embeddings, Contrastive learning for embeddings, Pretrained embedding models, Embeddings for semantic search and retrieval, Embedding pipelines in LLM systems, Bias and fairness in embeddings

Module 3: Transformer Architecture and LLM Internals (9 Hours)

Transformer architecture (advanced view), Input/output processing, Forward pass and decoding strategies, Context window and limitations, Key-value caching, Transformer blocks and efficiency, Efficient attention mechanisms, Positional embeddings (RoPE), Architectural optimizations for large-scale systems



Module 4: Applications, Fine-Tuning and Generative AI Extensions (9 Hours)

LLM-based text classification and clustering, Topic modeling (BERTopic), Embedding pipelines, Fine-tuning techniques: SBERT, Contrastive learning, Supervised and unsupervised fine-tuning, TSDAE, Domain adaptation, LLM fine-tuning: Pretraining, SFT, PEFT, LoRA, QLoRA, Instruction tuning, Quantization, Evaluation metrics and benchmarks, RLHF, DPO, Generative AI extensions: Diffusion models, U-Net role in generation, CLIP model for text-image alignment, Multimodal generation concepts

Module 5: Prompt Engineering, Agents and NVIDIA DLI Case Study (9 Hours)

Prompt engineering fundamentals, Prompt structures, Instruction-based prompting, Few-shot prompting, Chain-of-thought, Self-consistency, Tree-of-thought, Output validation, LangChain and LLM systems: Chains, workflows, memory, conversation handling, Agents, Tool usage, ReAct framework

Case Study (NVIDIA DLI – LLM Application Development): Iterative prompt engineering, Prompt templates, Batch prompting, Streaming responses, LangChain Expression Language (LCEL), Runnables, Chain composition, Parallel execution, Chat-based prompting, Structured output generation (Pydantic, JSON parsing), Document analysis and tagging, Tool integration, Agent workflows, LLM deployment using NVIDIA NIM with Llama 3.1

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module 1: Students use generative AI tools to explore the evolution of NLP systems, compare open vs proprietary LLMs, and critically evaluate responsible AI guidelines. AI-generated summaries of model capabilities are analyzed for accuracy and limitations.

Module 2: AI-assisted embedding visualization tools (e.g., UMAP, t-SNE projections) are used to analyze semantic similarity and bias in embedding spaces. Students evaluate AI-generated embeddings for fairness and contextual correctness.

Module 3: Students use transformer profiling tools and attention visualization frameworks to analyze decoder behaviour, context window effects, and key-value caching. AI-generated architecture diagrams are validated against reference implementations.

Module 4: AI-assisted fine-tuning pipelines (Hugging Face PEFT, LoRA) are used to adapt pretrained models to domain-specific tasks. Students evaluate fine-tuned models using standard benchmarks and critically assess RLHF/DPO alignment techniques.

Module 5: Students build end-to-end LLM applications using LangChain and NVIDIA NIM. Agent workflows are designed, tested, and critically evaluated for output reliability, hallucination control, and deployment readiness.



Assessment Scheme based on the course

Criteria	Weightage of Marks
Mini Project	10
Class Tests / Assignments	10
Mid-Semester Examination	30
Research Paper	10
End-Semester Examination	40
Total	100 marks

Recommended Tools

Primary Textbooks

1. Jay Alammar, Maarten Grootendorst, Hands-On Large Language Models, O'Reilly Media, 2024
2. Carlos Rodriguez, Generative AI Foundations in Python, Packt Publishing, 2024
3. Hugging Face Transformers & Diffusers Documentation – huggingface.co/docs
4. Andrej Karpathy, Neural Networks: Zero to Hero (lecture series) – karpathy.ai
5. NVIDIA Deep Learning Institute – LLM Application Development – learn.nvidia.com
6. LangChain Documentation – python.langchain.com

Additional AI & Computational Resources

1. NPTEL: Natural Language Processing – nptel.ac.in
2. Fast.ai Practical Deep Learning – fast.ai
3. Papers With Code – LLM Benchmarks – paperswithcode.com
4. Hugging Face Model Hub – huggingface.co/models
5. Google Colab / Jupyter Notebooks – colab.research.google.com



Data-Driven Recommendation Systems

Course Code: 26DS3608

Semester: VI

Program(s): B. Tech. CSE (Data Science)

Course Type: Professional Elective (PE-II)

Credits: 3

L – T – P – J: 3 – 0 – 0 – 0

Prerequisite(s): Machine Learning

Course Vision

To develop intelligent systems capable of delivering personalized, relevant, and ethical recommendations by integrating data-driven techniques with human-centric (authentic intelligence) principles for real-world applications.

Course Objectives

1. Understand the fundamentals of recommender systems along with foundational content-based and heuristic-driven techniques.
2. Apply collaborative filtering and matrix factorization techniques for building recommendation models.
3. Explore sequential and context-aware methods for dynamic personalization.
4. Analyze advanced recommendation approaches including text, graph, and reinforcement learning models.
5. Evaluate recommender systems and design AI-driven, user-centric solutions with fairness and diversity considerations.

Course Outcomes (CO)

CO	Description	Blooms Level
C01	Explain core concepts of recommender systems and apply content-based and heuristic- driven algorithms	L2
C02	Apply collaborative filtering and matrix factorization techniques to build recommendation models	L3
C03	Analyze and implement sequential and context-aware recommendation approaches	L4
C04	Evaluate advanced recommendation paradigms including text, graph, and reinforcement learning models	L5
C05	Assess recommender system performance and design user-centric AI-driven solutions	L5



Broad Course Description

This course introduces the fundamental and advanced concepts of recommender systems, focusing on how data-driven techniques are used to personalize user experiences across digital platforms. It covers core methods such as content-based filtering, collaborative filtering, and matrix factorization, along with modern approaches including sequential and context-aware recommendation. The course further explores advanced paradigms such as text-based, graph-based, and reinforcement learning-based recommendation systems. Emphasis is placed on evaluation metrics, fairness, diversity, and ethical considerations, enabling students to design robust and user-centric recommendation solutions for real-world applications.

CO-PO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
CO1	2	1			1						1	3	2
CO2	2	3			2						1	3	2
CO3	2	3			3						2	3	2
CO4	2	3	2	3	3						2	3	3
CO5	2	3	3	3	3						2	3	3
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module 1: Fundamentals of Recommender Systems

(10 Hours)

Introduction to recommender systems; properties of recommendation data; explicit vs implicit feedback; long-tail phenomenon and sparsity; recommendation tasks such as rating prediction and Top-N recommendation; overview of approaches; content-based filtering using vector space models, TF-IDF, and embeddings; rule-based and heuristic methods including association rule mining.

Applications: E-commerce recommendations, movie/music platforms, news feeds.



Module 2: Collaborative Filtering & Latent Factor Models (9 Hours)

Neighborhood-based methods including user-based and item-based KNN and Slope One; model-based methods; matrix factorization techniques such as SVD, FunkSVD, PMF, WRMF, BPR, and SVD++. Applications: Movie recommendation systems, product recommendation platforms.

Module 3: Sequential and Context-Aware Recommendation (8 Hours)

Sequential and session-based recommendation using RNN and CNN models; context-aware recommendation using Factorization Machines, Attentional FM, and Wide & Deep models. Applications: Video recommendations, session-based shopping, location-aware suggestions.

Module 4: Advanced Recommendation Paradigms (9 Hours)

Text-driven recommendation using review-based models; LLM-based recommendation; knowledge-aware recommendation using graph-based models; introduction to reinforcement learning including multi-armed bandits and contextual bandits. Applications: Personalized assistants, knowledge graph-based recommendations.

Module 5: Evaluation of Recommender Systems (9 Hours)

Evaluation protocols and data splitting; metrics for rating prediction such as RMSE and MAE; ranking metrics including Precision@k, Recall@k, MAP, and nDCG; beyond accuracy metrics such as diversity and novelty; classification-based evaluation metrics; case study on evaluating real-world recommendation systems. Applications: Performance evaluation in platforms like streaming and e-commerce systems.

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

- **Module 1:** AI models learn user preferences from interaction data for personalization.
Authentic Intelligence: Ensures recommendations are meaningful, diverse, and aligned with real user needs.
- **Module 2:** AI uncovers latent user-item relationships using collaborative filtering and factorization.
Authentic Intelligence: Ensures recommendations remain interpretable and trustworthy.
- **Module 3:** AI captures temporal and contextual patterns for dynamic personalization.
Authentic Intelligence: Ensures recommendations reflect real-life context and user intent.
- **Module 4:** AI integrates text, graph, and reinforcement learning for advanced personalization.
Authentic Intelligence: Ensures ethical, explainable, and user-aligned recommendation strategies.



- **Module 5:** AI-driven evaluation optimizes recommendation performance and user satisfaction.
Authentic Intelligence: Ensures fairness, diversity, and user-centric evaluation beyond accuracy.

Assessment Scheme

Criteria	Weightage of Marks
Project	10
Class Tests /assignments	10
Mid-Semester Examination	30
Hackathon/Technical Competition/ Paper presentation/Global/DLI certifications	10
End-Semester Examination	40
Total	100 marks

Recommended Tools

Primary Textbooks:

1. Aggarwal, C. C. – *Recommender Systems: The Textbook*, Springer.
2. Aliev, R. – *Recommender Algorithms: Practitioner's Guide*.

Additional Resources

Additional AI & Computational Resources

1. Python (Surprise, TensorFlow, PyTorch)
2. Recommender system libraries and datasets (MovieLens, Amazon Reviews)
3. Jupyter Notebook, Google Colab
4. <https://www.coursera.org/specializations/recommender-systems>
5. <https://www.coursera.org/specializations/packt-recommender-systems>

Project Component

Project 1 - Matrix Factorization Recommender

Implement SVD / FunkSVD

Dataset: MovieLens / Amazon

Focus: Latent factor models



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Project 2 - Hybrid Recommendation System

Combine content-based + collaborative filtering

Focus: Improve recommendation quality

Project 3 - Session-Based Recommendation System

Recommend next item based on user session

Tech: RNN / GRU4Rec

Focus: Sequential recommendation

Project 4 - Context-Aware Recommendation System

Include time, location, device context

Tech: Factorization Machines

Focus: Context-aware modeling

Project 5- News Recommendation System

Recommend articles using user reading history

Tech: NLP + embeddings

Focus: Real-world application



AI for Agriculture

Course Code: 26DS3609

Semester: VI

Program(s): B. Tech. CSE (Data Science)

Course Type: Professional Elective (PE-II)

Credits: 3

L – T – P – J: 3 – 0 – 0 – 0

Prerequisite(s): Machine learning and Deep Learning.

Course Vision

To develop intelligent, sustainable, and data-driven agricultural systems using advanced AI techniques, enabling precision farming and responsible decision-making aligned with real-world agricultural needs.

Course Objectives

1. Apply AI and deep learning techniques to agricultural data and farming systems.
2. Analyze crop, soil, and environmental data for predictive and prescriptive analytics.
3. Design AI-driven solutions for precision agriculture and smart farming.
4. Integrate remote sensing, IoT, and geospatial analytics with AI models.
5. Evaluate agricultural AI systems with sustainability, ethics, and real-world impact.

Course Outcomes (CO)

CO	Description	Blooms Level
CO1	Analyze agricultural data ecosystems and design AI-based representations for farming applications	L4
CO2	Apply AI and deep learning models for crop monitoring and disease detection	L3
CO3	Implement predictive models for yield forecasting and resource optimization	L3
CO4	Develop AI-driven precision agriculture systems integrating IoT and geospatial data	L6
CO5	Evaluate real-world agricultural AI applications considering sustainability and ethical practices	L5

Broad Course Description

This course introduces the application of artificial intelligence in modern agriculture, focusing on precision farming and data-driven decision-making. It covers core concepts such as agricultural



data ecosystems, computer vision for crop monitoring, predictive analytics, and AI-based decision support systems. The course also explores smart agriculture technologies including IoT, remote sensing, and geospatial analytics, while addressing sustainability and ethical considerations. Through case studies and real-world applications, students gain the ability to design scalable and efficient AI-driven agricultural solutions.

CO-PO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
C01	2	3			2	1					1	3	2
C02	2	3			3	1					1	3	2
C03	2	3	3	2	3	1					2	3	3
C04	2	3	3	2	3	2					2	3	3
C05	2	3	3	3	2	2	1				2	3	3
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module 1: AI Foundations in Agriculture

(9 Hours)

Agricultural data ecosystem including crop, soil, weather, and sensor data; challenges in agricultural analytics such as variability, uncertainty, and scalability; data representation and feature engineering; introduction to precision agriculture and smart farming; integration of AI with traditional farming practices.
Applications: Crop monitoring, soil health assessment, weather-based advisory systems.

Module 2: Computer Vision & Deep Learning in Agriculture

(9 Hours)

Image-based crop monitoring using CNNs; plant disease detection and classification; weed detection and segmentation; object detection for fruits and yield estimation; drone and satellite imagery analysis.
Applications: Disease diagnosis, automated harvesting, precision spraying.



Module 3: Predictive Analytics & Decision Support Systems

(9 Hours)

Crop yield prediction using ML/DL models; time-series forecasting for agriculture; soil and irrigation modelling; pest and disease outbreak prediction; recommendation systems for crop and fertilizer selection.

Applications: Yield forecasting, smart irrigation, pest management systems.

Module 4: Smart Agriculture Systems & Integration

(9 Hours)

IoT-based smart farming systems; sensor networks and real-time data analytics; geospatial analytics and remote sensing; edge AI and cloud-based platforms; robotics and automation in agriculture.

Applications: Smart irrigation, automated farming equipment, precision fertilization.

Module 5: Advanced Applications, Case Studies & Sustainability

(9 Hours)

Climate-smart agriculture; AI for supply chain and agri-logistics; ethical and socio-economic issues; evaluation and deployment challenges. Case studies include crop yield prediction using satellite data, AI-based disease detection using mobile applications, and smart irrigation systems using IoT sensors. Applications: Precision farming, climate adaptation, digital agriculture platforms.

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

Module 1: AI enables integration of heterogeneous agricultural data for informed farming decisions.

Authentic Intelligence: Ensures alignment of AI solutions with real farmer needs and environmental conditions.

Module 2: AI automates large-scale visual crop analysis using deep learning techniques.

Authentic Intelligence: Ensures interpretable and actionable insights for practical farming use.

Module 3: AI enables predictive and prescriptive analytics for optimized agricultural productivity.

Authentic Intelligence: Promotes sustainable decisions balancing yield and resource conservation.

Module 4: AI integrates IoT and geospatial data for intelligent and real-time farm management.

Authentic Intelligence: Ensures solutions are practical, affordable, and farmer-centric.

Module 5: AI enables scalable and sustainable agricultural solutions.

Authentic Intelligence: Ensures ethical, inclusive, and environmentally responsible AI usage.



Assessment Scheme

Criteria	Weightage of Marks
Project	10
Class Tests /assignments	10
Mid-Semester Examination	30
Hackathon/Technical Competition/ Paper presentation/Global/DLI certifications	10
End-Semester Examination	40
Total	100 marks

Recommended Tools

Primary Textbooks:

1. Fiaz, S., Nadeem, M. A., Baloch, F. S., & Chung, Y. S. (Eds.). (2026). Artificial intelligence and data sciences for precision agriculture. Springer.
2. Rani, S., Dutta, S., Rocha, Á., & Cengiz, K. (Eds.). (2025). AI and data analytics in precision agriculture for sustainable development. Springer.
3. Raj, P., Gayathri, N., & Kathrine, G. J. W. (Eds.). (2024). Artificial intelligence for precision agriculture. CRC Press.

Additional Resources:

1. Python (TensorFlow, PyTorch), GIS tools (QGIS, ArcGIS)
2. Remote sensing datasets, agricultural IoT platforms.
3. <https://www.oxfordhomestudy.com/courses/ai-courses-online/artificial-intelligence-and-agriculture>

Project Component

Project 1- Precision Agriculture System (End-to-End)

Combine soil data + weather + crop data

Output: Crop, fertilizer, irrigation recommendation

Tech: Hybrid ML system

Focus: System design



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakagalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Project 2- AI-Based Smart Farming Dashboard

Real-time analytics dashboard

Integrate multiple models (yield, disease, irrigation)

Tech: ML + Web dashboard (Streamlit)

Project 3- Drone Image-Based Crop Monitoring

Analyze aerial images for crop stress detection

Tech: CNN + segmentation

Advanced: Multispectral data

Project 4- Explainable AI for Agriculture Decisions

Build model + explain predictions using SHAP/LIME

Focus: Trust and interpretability

Project 5. Climate-Smart Agriculture Model

Predict crop suitability under climate change scenarios

Tech: ML + scenario simulation



Industrial Digital Twin Technology

Course Code: 26DS3610

Semester: VI

Program(s): B. Tech CSE (Data Science)

Course Type: Professional Elective (PE-II)

Credits: 3

L – T – P – J: 3 – 0 – 0 – 0

Prerequisite(s): Data Science Basics.

Course Vision

To develop the ability to design and implement Industrial Digital Twin systems by integrating IoT-enabled data acquisition, real-time analytics, and AI-driven modeling for intelligent, efficient, and scalable industrial environments.

Course Objectives

1. Develop a strong understanding of digital twin architecture, including system modeling, lifecycle management, and integration of physical and virtual systems, enabling students to conceptualize and design digital representations of industrial processes.
2. Build the ability to analyze industrial data collected from IoT devices and sensor-based systems by applying data acquisition techniques, preprocessing methods, and statistical analysis for extracting meaningful insights from real-time and historical datasets.
3. Equip students with knowledge of machine learning and data analytics techniques such as regression, classification, clustering, and time-series analysis, enabling predictive modeling and intelligent decision-making in industrial environments.
4. Develop the capability to evaluate ethical, security, and sustainability challenges in Industry 4.0, including data privacy, cyber threats, responsible AI usage, and energy-efficient system design, ensuring the development of secure and sustainable digital twin systems.

Course Outcomes (CO)

CO	Description	Blooms Level
CO1	Explain digital twin concepts and industrial systems	L2
CO2	Analyze industrial IoT data and data acquisition techniques	L4
CO3	Apply data analytics and machine learning models to industrial data	L3
CO4	Design digital twin solutions for real-world industrial applications	L6
CO5	Evaluate ethical, security, and sustainability aspects in digital twin systems	L5



Broad Course Description

This course introduces the concept of Industrial Digital Twins and their role in Industry 4.0 environments. It focuses on integrating IoT systems, real-time data acquisition, and AI-driven analytics to create virtual replicas of physical systems. Students will learn how digital twins are used for monitoring, prediction, and optimization in industrial applications. The course further explores data analytics, machine learning models, and real-time system integration, along with emerging technologies such as AI-based simulation and autonomous systems. Emphasis is placed on practical applications, system design, and ethical considerations in industrial automation.

CO-PO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
C01	2	1			1						1	2	2
C02	2	3			2						1	3	2
C03	2	3			3						2	3	2
C04	2	3	3	2	3						2	3	3
C05	2	3	3	3	2	1	1				2	3	3
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module 1: Introduction to Digital Twins

(9 Hours)

Definition and evolution of digital twins, Industry 4.0 overview, Digital twin vs simulation vs IoT, types of digital twin's component asset system process, digital twin architecture, lifecycle of digital twin systems, modeling concepts and system representation.

Module 2: Industrial IoT and Data Acquisition

(9 Hours)

Industrial IoT architecture sensors actuators PLC SCADA, communication protocols MQTT OPC-UA, data acquisition systems, real-time data streaming, edge vs cloud computing, IoT-enabled industrial systems, data collection pipelines.

Module 3: Data Analytics and Modeling

(9 Hours)

Data preprocessing and integration, exploratory data analysis, time series forecasting, machine learning techniques regression classification clustering, physics-based vs data-driven models, predictive modeling, fault detection techniques.



Module 4: Digital Twin Applications (9 Hours)

Predictive maintenance, process optimization, energy efficiency, supply chain digital twins, remote monitoring systems, decision support systems, industrial dashboards.

Module 5: Advanced Digital Twins and NVIDIA Integration (9 Hours)

Digital twin platforms and ecosystems, NVIDIA Omniverse basics, GPU acceleration for simulation, real-time synchronization, cloud vs edge deployment, AI-driven digital twins, autonomous industrial systems, scalability and performance optimization.

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module 1: AI tools are used for simulation and conceptual modeling of digital twin systems. Students generate virtual models and validate them against real-world industrial behaviour to understand system accuracy and limitations.

Module 2: AI is applied in IoT data pipelines for real-time data acquisition and processing. Students analyze sensor data quality, detect noise, and validate data reliability in practical scenarios.

Module 3: Machine learning models are used for industrial data analytics and predictive modeling. Students evaluate model accuracy, detect errors, and validate results using real datasets.

Module 4: AI-driven optimization techniques are applied for improving industrial processes and decision systems. Students analyze performance improvements and validate optimization outcomes.

Module 5: GPU-based tools and NVIDIA platforms are used for real-time simulation and high-performance digital twin systems. Students compare CPU vs GPU performance and validate efficiency gains.

Assessment Scheme

Criteria	Weightage of Marks
Project	10
Class Tests / Assignment	10
Mid-Semester Examination	30
Paper presentation	10
End-Semester Examination	40
Total	100 marks



Recommended Tools

1. Python: Used for data processing, industrial analytics, and machine learning model development in digital twin systems.
2. TensorFlow or PyTorch: Used for building and deploying machine learning and deep learning models for predictive maintenance and system optimization.
3. Jupyter Notebook / Google Colab: Used for interactive development, visualization, and experimentation with industrial datasets and AI models.
4. Cloud Platforms (AWS, Azure, GCP): Used for scalable data storage, real-time processing, and deployment of digital twin systems in industrial environments.
5. Simulation Tools (MATLAB, AnyLogic): Used for modeling, simulation, and validation of digital twin systems and industrial processes.
6. NVIDIA Omniverse: Used for real-time digital twin simulation, 3D modeling, and collaborative industrial system visualization.
7. CUDA Programming Tools: Used for GPU-based parallel computing and accelerating simulation and analytics tasks in digital twin systems.

Primary Textbooks

1. Fei Tao, Qinglin Qi, Ang Liu, and Andrew Kusiak, Digital Twin Driven Smart Manufacturing, Academic Press (Elsevier), 2019.
2. Alasdair Gilchrist, Industry 4.0: The Industrial Internet of Things, Apress, 2016.
3. Michael Grieves, Digital Twin: Manufacturing Excellence through Virtual Factory Replication, White Paper, Florida Institute of Technology, 2014.
4. Ian Goodfellow, Yoshua Bengio, and Aaron Courville, Deep Learning, MIT Press, 2016.

Mini Project Component

Industry-Oriented Projects

Project 1 – Smart Factory Digital Twin

Problem: Develop a digital twin model of a smart factory to monitor and optimize operations in real time

Concepts Used: IoT integration, data acquisition, simulation modeling, real-time analytics

Industrial Relevance: Industry 4.0 manufacturing, smart production systems, automation

Project 2 – Predictive Maintenance System

Problem: Design a system to predict equipment failures using sensor data and machine learning

Concepts Used: Time-series analysis, machine learning models, anomaly detection

Industrial Relevance: Manufacturing, energy systems, industrial equipment monitoring

Project 3 – Industrial Process Optimization

Problem: Optimize industrial workflows using digital twin models and analytics

Concepts Used: Process modeling, AI-based optimization, performance analysis

Industrial Relevance: Production systems, supply chain optimization, operational efficiency



Project 4 – Energy Monitoring System

Problem: Develop a system to monitor and optimize energy consumption in industrial environments

Concepts Used: Sensor data analytics, real-time monitoring, optimization techniques

Industrial Relevance: Smart grids, sustainable manufacturing, energy management systems

Project 5 – GPU-Accelerated Digital Twin Simulation

Problem: Implement a high-performance digital twin simulation using GPU acceleration

Concepts Used: CUDA programming, parallel computing, simulation optimization

Industrial Relevance: High-performance computing, real-time simulation, AI-driven industrial systems

AI-Powered Assignments

Assignment 1: AI-Based Digital Twin System Design

Students prompt an AI tool to design a digital twin system for a given industrial process.

Students must:

- Analyze the system architecture and component interactions
- Identify limitations in model representation and assumptions
- Validate the digital twin design by comparing with real-world industrial systems
- Assessment focuses on system design understanding, modeling accuracy, and validation of AI outputs

Assignment 2: AI-Based IoT Data Pipeline Analysis

Students use AI tools to design and simulate industrial IoT data pipelines.

Students must:

- Analyze data flow from sensors to processing systems
- Identify inefficiencies or errors in data transmission
- Validate pipeline performance using real-time or simulated datasets
- Assessment focuses on IoT system understanding, data handling, and performance evaluation

Assignment 3: AI-Based Predictive Maintenance Model

Students generate machine learning models using AI tools for predicting equipment failures.

Students must:

- Analyze prediction accuracy and model performance
- Identify overfitting, underfitting, or incorrect assumptions
- Validate results using industrial datasets and performance metrics
- Assessment focuses on predictive analytics, model evaluation, and industrial applicability

Assignment 4: AI-Based Industrial Optimization

Students simulate optimization problems using AI tools for improving industrial processes.

Students must:

- Analyze system efficiency improvements
- Identify constraints and optimization limitations
- Validate results using performance metrics and comparative analysis
- Assessment focuses on optimization techniques, analytical reasoning, and validation skills



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Assignment 5: GPU-Based Digital Twin Simulation using NVIDIA Tools

Students implement digital twin simulations using GPU acceleration tools such as CUDA or NVIDIA platforms.

Students must:

- Analyze performance differences between CPU and GPU execution
- Identify bottlenecks in simulation or parallel processing
- Validate improvements using benchmarking techniques
- Assessment focuses on parallel computing concepts, performance optimization, and system efficiency



Cloud Data Security

Course Code: 26DS3611

Semester: VI

Program(s): B.Tech. CSE (Data Science)

Course Type: Professional Elective (PE-II)

Credits: 3

L – T – P – J: 3 – 0 – 0 – 0

Prerequisite(s): Computer Networks, Information Security Basics

Course Vision

To provide in-depth knowledge of cloud data security mechanisms, enabling students to design secure, privacy-preserving, and resilient cloud-based systems using modern tools and advanced technologies.

Course Objectives

1. Understand cloud computing security fundamentals and associated risks
2. Analyze data protection and privacy challenges in cloud environments
3. Apply identity and access management mechanisms for secure cloud systems
4. Explore secure storage and distributed security architectures
5. Implement advanced cloud security solutions using NVIDIA platforms

Course Outcomes (CO)

CO	Description	Blooms Level
C01	Illustrate cloud security fundamentals, threat models, and risk management strategies	L2
C02	Analyze data protection, privacy techniques, and risk assessment models in cloud environments	L4
C03	Apply identity and access management mechanisms including authentication and authorization models	L3
C04	Understand secure cloud storage, distributed security, and container-based protection mechanisms	L2
C05	Implement advanced cloud security solutions using NVIDIA platforms for AI-driven security systems	L3



Broad Course Description

Cloud Data Security is an advanced elective course designed to address the growing need for securing data in cloud environments. The course covers fundamental security principles, data protection mechanisms, identity and access management, distributed security architectures, and advanced AI-driven cloud security solutions.

The course is structured into five modules covering cloud security foundations, privacy-preserving data protection techniques, IAM frameworks, secure cloud storage and distributed systems, and NVIDIA-powered advanced security solutions. Students gain exposure to real-world security frameworks, threat modeling techniques, and cutting-edge technologies such as differential privacy, zero-trust architectures, and GPU-accelerated cybersecurity.

This course prepares students for roles in cloud security engineering, cybersecurity analytics, and secure AI system deployment.

CO-PO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
C01	2	1			1						1	2	1
C02	2	3			1		1				1	2	1
C03	2	3			2						1	2	1
C04	2	2			2						2	2	1
C05	2	3	3	3	3						2	3	2
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module 1: Fundamentals of Cloud Security

(9 Hours)

Introduction to cloud computing security, Security challenges in cloud environments, Cloud deployment models and associated risks, Threat models in cloud systems, Confidentiality,



Integrity, Availability (CIA), Shared responsibility model, Security standards and frameworks, Risk assessment and vulnerability management.

Module 2: Data Protection and Privacy in Cloud (9 Hours)

Cloud data lifecycle security (data at rest, in transit, and in use) – advanced protection perspectives, Threat modeling for cloud data (STRIDE, DREAD) and risk assessment strategies, Data privacy challenges in multi-tenant and distributed cloud environments, Data masking (static and dynamic) and tokenization techniques, Data anonymization and de-identification methods: k-anonymity, l-diversity, t-closeness, Differential privacy and its application in cloud-based analytics systems.

Module 3: Identity and Access Management (9 Hours)

Identity and Access Management (IAM), Authentication and authorization, Multi-factor authentication (MFA), Single Sign-On (SSO), Role-Based Access Control (RBAC), Attribute-Based Access Control (ABAC), Federated identity management, Zero Trust security model.

Module 4: Secure Cloud Storage and Distributed Security (9 Hours)

Secure cloud storage architecture, Distributed storage security, Data deduplication security, Virtualization security, Container security, Secure APIs and microservices, Intrusion Detection Systems (IDS), Intrusion Prevention Systems (IPS), Security in distributed and hybrid cloud environments.

Module 5: Advanced Cloud Security using NVIDIA Platforms (9 Hours)

NVIDIA confidential computing for secure AI workloads, NVIDIA BlueField DPU for infrastructure security, GPU-accelerated cybersecurity analytics, NVIDIA Morpheus framework for real-time threat detection, Secure AI model deployment using NVIDIA AI Enterprise, Data center protection using NVIDIA networking security solutions, Zero-trust architecture using accelerated computing, AI-driven security operations using NVIDIA accelerated platforms.

AI Integration & Authentic Intelligence (AI² Pedagogy)

AI Integration

- **Module 1:** Students use AI tools to simulate cloud threat scenarios and evaluate risk assessment strategies.
- **Module 2:** AI-assisted tools are used to implement anonymization and privacy-preserving techniques such as differential privacy and evaluate effectiveness.
- **Module 3:** Students use AI tools to design IAM policies and evaluate authentication mechanisms under simulated attacks.
- **Module 4:** AI tools are used to analyze IDS/IPS logs and detect anomalies in distributed cloud environments.



- **Module 5:** Students use NVIDIA AI platforms to simulate real-time threat detection and evaluate AI-driven security pipelines.

Assessment Scheme

Criteria	Weightage of Marks
Class Tests	10
Mid-Semester Examination	30
Mini Project	10
Assignments	10
End-Semester Examination	40
Total	100 marks

Recommended Tools

- AWS Security Hub / Azure Security Center / Google Cloud Security Tools
- Wireshark, Snort (IDS/IPS)
- Docker & Kubernetes Security Tools
- Open Policy Agent (OPA)
- NVIDIA Morpheus Framework
- NVIDIA BlueField DPU
- SIEM tools (Splunk / ELK Stack)

Primary Textbooks

1. Deepak Dhyani, RAG with Python Cookbook, Packt Publishing, 2024
2. Jay Alammar, Maarten Grootendorst, Hands-On Large Language Models, O'Reilly Media, 2024
3. NVIDIA Deep Learning Institute – RAG and LLM Systems – learn.nvidia.com
4. Lewis et al., Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks, arXiv:2005.11401
5. FAISS Documentation – faiss.ai | ChromaDB Documentation – docs.trychroma.com
6. LangChain Documentation – python.langchain.com

Additional AI & Computational Resources

1. NPTEL: Deep Learning for NLP – nptel.ac.in
2. Hugging Face RAG Model Documentation – huggingface.co/docs
3. Papers With Code – RAG Benchmarks – paperswithcode.com
4. Google Colab / Jupyter Notebooks – colab.research.google.com
5. Pinecone Learning Center – Vector Databases & RAG – pinecone.io/learn



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Mini Project Component

Project 1 – Cloud Data Privacy Framework

Design and implement a privacy-preserving cloud data system using anonymization and differential privacy.

Project 2 – IAM Security Model Implementation

Develop and simulate RBAC/ABAC-based IAM system for a cloud application.

Project 3 – AI-Based Cloud Threat Detection

Implement a real-time threat detection system using NVIDIA Morpheus or similar AI frameworks.

AI-Powered Assignments

Assignment 1: Threat Modeling Analysis

Use AI tools to generate a STRIDE/DREAD threat model and critically evaluate risks.

Assignment 2: Privacy Technique Evaluation

Analyze AI-generated anonymization outputs and assess privacy vs utility trade-offs.

Assignment 3: Zero Trust Architecture Design

Design and evaluate an AI-generated Zero Trust cloud architecture for enterprise use.



Multimodal RAG System

Course Code: 26DS4709

Semester: VII

Program(s): B.Tech. CSE (Data Science)

Course Type: Professional Elective (PE-III)

Credits: 3

L – T – P – J: 3 – 0 – 0 – 0

Prerequisite(s): Large Language Models

Course Vision

To provide advanced knowledge of multimodal Retrieval-Augmented Generation (RAG), focusing on cross-modal retrieval, knowledge grounding, and efficient data pipelines for real-world intelligent applications.

Course Objectives

1. Understand multimodal RAG concepts, architecture, and system evolution
2. Analyze multimodal data processing and knowledge construction techniques
3. Explore advanced retrieval and indexing strategies for multimodal systems
4. Implement context-grounded response generation with hallucination mitigation
5. Develop and deploy agentic multimodal retrieval-based AI systems

Course Outcomes (CO)

CO	Description	Blooms Level
CO1	Explain multimodal RAG concepts and architecture	L2
CO2	Analyze multimodal data processing and knowledge construction	L4
CO3	Apply advanced retrieval and indexing techniques	L3
CO4	Develop multimodal reasoning and response generation systems	L6
CO5	Design and implement agentic multimodal RAG systems	L6

Broad Course Description

Multimodal RAG is a 3-credit advanced course for CSE- DS students. It focuses on Retrieval-Augmented Generation systems that operate across text, image, and other data modalities. Students learn to design robust retrieval pipelines, construct knowledge representations, and build grounded generation systems that minimize hallucination and maximize contextual accuracy. The course covers five modules: Advanced Multimodal RAG Foundations, Multimodal Data Processing and Knowledge Construction, Multimodal Retrieval and Indexing Optimization, Context-Grounded Response Generation, and Retrieval-Augmented AI Pipelines with an NVIDIA



DLI case study. Students gain hands-on experience with LangChain, FAISS, ChromaDB, and cross-encoder re-ranking to build production-grade multimodal RAG systems.

CO-PO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
C01	2	1			1						1	3	2
C02	2	3			2						1	3	2
C03	2	3			3						2	3	2
C04	2	3	3	2	3						2	3	3
C05	2	3	3	3	3						2	3	3
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module 1: Advanced Multimodal RAG Foundations (9 Hours)

Evolution from text-based RAG to multimodal RAG, Limitations of unimodal systems, Multimodal data types (text, image, audio), Use of pretrained embeddings for cross-modal retrieval, Architecture of multimodal RAG systems, Fusion strategies (early, late, hybrid), Context grounding across modalities, Handling modality gaps and noisy data

Module 2: Multimodal Data Processing and Knowledge Construction (9 Hours)

Document ingestion pipelines, Processing multimodal datasets (text, images, PDFs), Metadata enrichment techniques, Adaptive chunking strategies, Hierarchical document structuring, Graph-based knowledge representation, Indexing strategies for multimodal data, Storage optimization for large-scale systems

Module 3: Multimodal Retrieval and Indexing Optimization (9 Hours)

Dense, sparse, and hybrid retrieval methods, Cross-modal similarity search, multi-vector retrieval strategies, Query transformation and expansion, Metadata-aware retrieval, Temporal and streaming retrieval systems, Re-ranking using cross-encoders, ANN indexing and caching, Retrieval performance evaluation metrics

Module 4: Context-Grounded Response Generation (9 Hours)

Context-aware response generation using LLMs, Grounded generation techniques, Structured and citation-aware outputs, multi-step reasoning with retrieved context, Confidence estimation techniques, Hallucination detection and mitigation, Response validation strategies

Module 5: Retrieval-Augmented AI Pipelines – NVIDIA DLI (9 Hours)

Design of retrieval-augmented systems, Tool-based retrieval pipelines, Self-query and adaptive retrieval mechanisms, Conversational systems with retrieval memory, Pipeline orchestration using LangChain and LCEL, LLM inference interfaces, Dialog management and state handling,



Integration with APIs and tools, Embedding-based guardrails, Evaluation and monitoring, Production deployment strategies

Case Study: End-to-end multimodal RAG system — designing a retrieval-based response generation pipeline integrating text and image modalities, FAISS vector store, cross-encoder re-ranking, and LangChain orchestration for a document intelligence application

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module 1: Students use AI tools to compare unimodal vs multimodal RAG architectures and analyze fusion strategy trade-offs. AI-generated architecture diagrams are critically evaluated against reference implementations for correctness and completeness.

Module 2: AI-assisted document ingestion pipelines (LangChain loaders, PDF parsers) are used to process and chunk multimodal datasets. Students evaluate metadata enrichment quality and validate knowledge graph structures for consistency.

Module 3: Students implement and benchmark dense, sparse, and hybrid retrieval strategies using FAISS and ChromaDB. AI-generated query expansion results are critically assessed for retrieval relevance and bias minimization.

Module 4: AI-powered generation pipelines are built using LLMs with retrieved context. Students evaluate grounded outputs for hallucination, citation accuracy, and multi-step reasoning quality using automated validation tools.

Module 5: Students orchestrate full RAG pipelines using LangChain LCEL and NVIDIA DLI resources. Agent-based retrieval workflows are designed, tested for reliability, and critically evaluated for safe and controlled behaviour in production scenarios.

Assessment Scheme based on the course

Criteria	Weightage of Marks
Mini Project	10
Class Tests	10
Assignment	10
Mid-Semester Examination	30
End-Semester Examination	40
Total	100 marks



Recommended Tools

Primary Textbooks

1. Deepak Dhyani, RAG with Python Cookbook, Packt Publishing, 2024
2. Jay Alammar, Maarten Grootendorst, Hands-On Large Language Models, O'Reilly Media, 2024
3. NVIDIA Deep Learning Institute – RAG and LLM Systems – learn.nvidia.com
4. Lewis et al., Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks, arXiv:2005.11401
5. FAISS Documentation – faiss.ai | ChromaDB Documentation – docs.trychroma.com
6. LangChain Documentation – python.langchain.com

Additional AI & Computational Resources

1. NPTEL: Deep Learning for NLP – nptel.ac.in
2. Hugging Face RAG Model Documentation – huggingface.co/docs
3. Papers With Code – RAG Benchmarks – paperswithcode.com
4. Google Colab / Jupyter Notebooks – colab.research.google.com
5. Pinecone Learning Center – Vector Databases & RAG – pinecone.io/learn



Immersive Data Science

Course Code: 26DS4705

Semester: VII

Programs Offered: B.Tech. CSE (Data Science)

Course Type: Professional Elective (PE-III)

Credits: 3

L – T – P – J: 3 – 0 – 0 – 0

Prerequisites: Python Programming Fundamentals

Course Vision

To develop a strong foundation in immersive data science by integrating augmented reality, virtual reality, and mixed reality technologies with data-driven approaches, enabling students to design, analyze, and implement interactive immersive systems for real-world applications.

Course Objectives

1. Develop an understanding of historical and modern perspectives of immersive technologies including AR, VR, and MR.
2. Enable students to understand human perception and sensory interaction in immersive environments.
3. Equip students with knowledge of system architecture and hardware components of VR systems.
4. Provide exposure to software development techniques for AR and VR applications.
5. Enable students to design and evaluate immersive applications for real-world scenarios.

Course Outcomes (CO)

CO	Description	Blooms Level
C01	Identify, examine, and develop software reflecting fundamental techniques for AR/VR design and deployment	L2
C02	Describe the working principles of AR and VR systems	L2
C03	Select and develop appropriate designs for immersive applications	L3
C04	Evaluate the impact of AR/VR techniques on human perception and usability	L5
C05	Analyze state-of-the-art AR/VR applications and propose solutions for real-world problems	L4

Broad Course Description

Immersive Data Science is an advanced course that introduces students to augmented reality, virtual reality, and mixed reality technologies. The course focuses on system design, human perception, hardware, and software development aspects of immersive environments. Students



learn to build, analyze, and evaluate immersive applications using modern tools and frameworks. The course integrates theoretical knowledge with practical exposure through case studies and capstone projects. It enables students to apply immersive technologies in domains such as digital entertainment, healthcare, education, and industrial simulations, thereby enhancing problem-solving and innovation skills.

CO-PO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
C01	2	1			2						1	2	2
C02	2	2			1						1	2	2
C03	2	3	3	2	3						2	3	2
C04	2	3	3	3	2						2	3	2
C05	2	3	3	3	3						2	3	3
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module – 1: Introduction to Immersive Technologies (9 Hours)

Introduction to augmented, virtual, and mixed reality; taxonomy, features, and technology of AR; differences between AR, VR, and MR; challenges in AR; AR system components and functionality; visualization techniques for augmented reality.

Module – 2: Virtual Reality Systems (9 Hours)

VR as a discipline; features of VR systems; architecture of VR systems; VR hardware components including input devices such as tracking systems, motion capture systems, and data gloves; output devices such as visual displays; system integration.

Module – 3: VR Software Development (9 Hours)

Challenges in VR software development; architectural models such as master-slave and client-server; cluster rendering; game engines and SDKs for VR development; development platforms including HTC Vive, Oculus, and Google VR; performance considerations.

Module – 4: AR Software Development (9 Hours)

AR software fundamentals; camera parameters and calibration techniques; marker-based augmented reality; AR Toolkit; implementation techniques and system optimization.

Module – 5: Applications and Case Studies (9 Hours)

Applications of VR in digital entertainment including film, television production, and gaming; VR in physical exercise and simulation-based training; immersive demonstrations.

Case Studies: AR/VR Gaming Experience, Virtual Tour Systems, Healthcare Simulation



AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module-1:

Students utilize AR/VR visualization tools and simulation platforms (Unity, Unreal Engine) to understand immersive environments. AI-assisted tools are used to analyze user interaction patterns and optimize interface design.

Module-2:

AI-enabled tracking and motion analysis tools are used to study VR hardware performance. Students explore sensor data processing and gesture recognition using machine learning techniques.

Module-3

Students use AI-assisted development environments and game engines to design VR applications. Machine learning models are introduced to enhance rendering, object recognition, and user interaction.

Module-4

AI tools are used for camera calibration, marker detection, and real-time AR object placement. Students explore computer vision techniques using OpenCV and deep learning models.

Module-5

Students implement AI-driven immersive applications such as intelligent virtual assistants, adaptive VR environments, and predictive interaction systems for real-world applications.

Assessment Scheme based on the course

Criteria	Weightage of Marks
Internal Assessment	40
Mini Project / Case Study	20
End Semester Examination	40
Total	100 Marks

Recommended Tools

Primary Textbooks

1. George Mather, *Foundations of Sensation and Perception*, Psychology Press, 2009
2. Jason Jerald, *The VR Book: Human-Centered Design for Virtual Reality*
3. Tony Parisi, *Learning Virtual Reality*, O'Reilly
4. Burdea G. C., Coffet P., *Virtual Reality Technology*, Wiley IEEE Press
5. Alan B. Craig, *Understanding Augmented Reality*, Morgan Kaufman



Additional AI & Computational Resources

1. <http://msl.cs.uiuc.edu/vr/>
2. <https://learn.unity.com/>
3. <https://www.coursera.org/>
4. <https://developer.oculus.com/>

Mini Project Component

The following projects are suggested based on course outcomes. Students may attempt one project individually or in pairs.

Project 1 – AR/VR Gaming Experience

Problem: Develop an immersive gaming application

Concepts Used: VR interaction, rendering, game engines

Industrial Relevance: Gaming industry, simulation systems

Project 2 – Virtual Tour System

Problem: Create an AR/VR-based virtual tour of historical sites

Concepts Used: 3D modeling, AR visualization

Industrial Relevance: Tourism, education

Project 3 – VR-Based Training Simulator

Problem: Develop a VR system for skill training

Concepts Used: Simulation, interaction design

Industrial Relevance: Healthcare, defense training

Project 4 – AR Object Recognition System

Problem: Identify and overlay real-world objects using AR

Concepts Used: Computer vision, AR toolkit

Industrial Relevance: Retail, manufacturing

Project 5 – Immersive Data Visualization System

- Problem: Visualize complex datasets in VR
- Concepts Used: Data visualization, VR rendering
- Industrial Relevance: Data analytics, business intelligence

AI-Powered Assignments

Assignment 1: AI-Based AR/VR System Design

Students prompt an AI tool: Design an AR/VR system for an educational application.

Students must:

- Analyze system architecture and feasibility
- Identify limitations in AI-generated design
- Improve the design using standard AR/VR principles, Assessment focuses on system design understanding and validation of AI outputs

Assignment 2: AI-Assisted VR Interaction Analysis

Students prompt an AI tool: Simulate user interaction in a VR environment

Students must:



- Analyze interaction behaviour and response time
- Identify inconsistencies in simulation
- Refine using human perception models

Assessment focuses on usability and interaction design

Assignment 3: AI-Based VR Application Development

Students prompt an AI tool: Generate a VR application prototype

Students must:

- Evaluate performance and rendering quality
- Identify errors in implementation
- Optimize using appropriate frameworks

Assessment focuses on development and optimization

Assignment 4: AI-Based AR Object Detection

Students prompt an AI tool: Detect and overlay AR objects using camera input

Students must:

- Analyze detection accuracy
- Identify limitations in tracking
- Improve using computer vision techniques

Assessment focuses on computer vision and AR implementation

Assignment 5: AI-Driven Immersive System Evaluation

Students prompt an AI tool: Evaluate an immersive system for real-world use

Students must:

- Analyze system effectiveness
- Identify gaps in AI evaluation
- Propose improvements

Assessment focuses on analytical and evaluation skills



AI for FinTech & Risk Analytics

Course Code: 26DS4706

Semester: VII

Program(s): B.Tech CSE (Data Science)

Course Type: Professional Elective (PE-III)

Credits: 3

L – T – P – J: 3 – 0 – 0 – 0

Prerequisite(s): Basic knowledge of finance and computing.

Course Vision

To develop a strong foundation in financial technology by integrating digital financial systems, analytics, and AI-driven decision-making, enabling students to analyze, design, and implement modern FinTech solutions for real-world applications.

Course Objectives

1. Develop an understanding of FinTech evolution and digital financial ecosystems.
2. Enable students to analyze modern payment systems and regulatory technologies.
3. Equip students with knowledge of blockchain, digital wallets, and decentralized systems.
4. Provide understanding of cybersecurity and cryptographic techniques in financial systems.
5. Enable students to evaluate FinTech business models and emerging trends using data-driven approaches.

Course Outcomes (CO)

CO	Description	Blooms Level
CO1	Illustrate FinTech concepts and financial innovations	L2
CO2	Analyze digital payment systems and regulatory technologies	L4
CO3	Illustrate blockchain and digital financial ecosystems	L2
CO4	Evaluate security mechanisms in financial systems	L4
CO5	Apply business models and emerging trends in FinTech	L3

Broad Course Description

Financial Technology (FinTech) is an interdisciplinary course that introduces students to digital financial systems, payment technologies, blockchain, cybersecurity, and analytics-driven financial decision-making. The course explores the transformation of traditional financial



systems into digital ecosystems powered by advanced technologies such as artificial intelligence, big data, and distributed ledgers.

Students gain insights into modern financial infrastructure, regulatory frameworks, and emerging innovations such as decentralized finance, digital lending, and robo-advisory systems. The course emphasizes real-world applications, industry relevance, and analytical thinking required for modern financial services and technology-driven enterprises.

CO-PO mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
C01	2	1			1						1	2	1
C02	2	3			1	1					1	2	1
C03	2	2			2						1	2	1
C04	2	3	2	3	2						2	2	2
C05	2	3	2	2	1	1					2	2	2
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module – 1: Introduction to FinTech (9 Hours)

Definition and evolution of FinTech; traditional vs digital financial systems; drivers of FinTech growth; enabling technologies; financial inclusion and digital transformation; applications in digital banking and online financial services.

Module – 2: Payment Systems, RegTech, and InsurTech (9 Hours)

Digital payment systems including UPI, wallets, and cards; payment gateways and infrastructure; regulatory technology and compliance automation; insurance technology; fraud detection mechanisms; applications in mobile payments and automated compliance systems.

Module – 3: Blockchain, Digital Wallets, and Remittances (9 Hours)

Blockchain fundamentals; cryptocurrencies; smart contracts; digital wallets; cross-border payments; remittance systems; decentralized finance basics; applications in cryptocurrency platforms and global transactions.

Module – 4: Security in Digital Financial Systems (9 Hours)

Cryptography fundamentals; encryption techniques; digital signatures; authentication mechanisms; cybersecurity threats; secure transaction systems; risk and vulnerability management; applications in fraud prevention and secure payments.

Module – 5: FinTech Business Models and Emerging Trends (9 Hours)

FinTech business models; peer-to-peer lending; crowdfunding; robo-advisors; digital lending platforms; investment strategies using analytics; AI in FinTech; open banking; future trends; applications in digital advisory and investment platforms.



Case Studies: UPI ecosystem in India – scalable digital payment infrastructure, PayPal and Stripe – global payment platforms, Bitcoin and Ethereum – decentralized finance systems, Robo-advisory platforms (Wealthfront / Betterment)

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module-1: Students use AI-driven analytics (time-series forecasting, clustering) to analyze financial inclusion, transaction patterns, and digital adoption trends. Exposure to macro-financial datasets and predictive modeling of economic behaviour.

Module-2: Implementation of supervised and unsupervised machine learning models (Logistic Regression, Random Forest, Isolation Forest) for fraud detection. Real-time anomaly detection pipelines and behavioural biometrics analysis for secure transactions.

Module-3: Use of graph analytics and AI models to detect suspicious blockchain activities, transaction clustering, and smart contract vulnerability analysis. Introduction to AI in DeFi risk modeling and tokenomics evaluation.

Module-4: Application of deep learning models (Autoencoders, LSTM) for intrusion detection and financial anomaly prediction. Risk scoring models, adversarial attacks, and secure AI pipelines in financial ecosystems.

Module-5: Development of AI-based recommendation systems for investments, robo-advisory engines, portfolio optimization using reinforcement learning, and explainable AI (XAI) for financial decisions.

Assessment Scheme

Criteria	Weightage of Marks
Mid-Semester Examination	40
Mini Project	10
Assignments	10
End-Semester Examination	40
Total	100 marks

Recommended Tools

Primary Textbooks

1. Dr. Archana, FinTech and Data Analytics, CHYREN Publication, 2026
2. Susanne Chishti & Janos Barberis, The FinTech Book, Wiley



3. David L. Rogers, The Digital Transformation Playbook, Columbia Business School

Additional AI & Computational Resources

1. FinTech research papers and industry reports
2. RBI and global financial regulatory reports
3. Online platforms for blockchain and digital finance

Mini Project Component

The following projects are suggested based on course outcomes. Students may attempt one project individually or in pairs.

Project 1 – Real-Time Fraud Detection System

Problem: Build an end-to-end ML system for detecting fraudulent transactions

Concepts Used: Python, Scikit-learn, Stream processing

Industrial Relevance: Banking security systems

Project 2 – Blockchain Intelligence Dashboard

Problem: Build an end-to-end ML system for detecting fraudulent transactions

Concepts Used: Apply graph analytics and anomaly detection

Industrial Relevance: Crypto exchanges, regulatory bodies

Project 3 – AI-Based Credit Scoring Engine

Problem: Develop a creditworthiness prediction system

Concepts Used: Include explainable AI for transparency

Industrial Relevance: Lending platforms, NBFCs

Project 4 – Algorithmic Trading System

Problem: Build predictive trading models using ML/DL

Concepts Used: Backtesting and strategy optimization

Industrial Relevance: FinTech trading firms

AI-Powered Assignments

Assignment 1: Intelligent Payment Fraud Detection System

Students prompt an AI tool: *Build and evaluate a fraud detection system using a financial transaction dataset. Perform preprocessing, handle class imbalance, and compare AI-based vs rule-based approaches.*

Students must:

- Evaluate whether the AI-generated model correctly handles class imbalance (e.g., SMOTE usage) and whether performance metrics (ROC-AUC, precision-recall) are appropriate for fraud detection
- Identify potential issues: check for data leakage, improper validation techniques, and bias toward majority class predictions
- Assess whether the comparison between rule-based and AI-based systems is meaningful and aligned with real-world fraud detection scenarios

Assessment focuses on: Model evaluation correctness, handling of imbalanced data, and critical analysis of AI-generated fraud detection pipelines

Assignment 2: Blockchain Transaction Intelligence



Students prompt an AI tool: *Analyze blockchain transaction data to identify suspicious wallet behaviour using graph-based techniques and clustering methods.*

Students must:

- Evaluate whether the AI-generated analysis correctly identifies transaction patterns and wallet clusters using graph/network analysis
- Identify potential misinterpretations: check if fraudulent patterns are overgeneralized or lack proper statistical justification
- Assess whether decentralization and trust metrics are accurately computed and meaningfully interpreted

Assessment focuses on: Graph-based analysis accuracy, anomaly detection validity, and interpretation of decentralized systems

Assignment 3: AI-Based Credit Risk Modeling

Students prompt an AI tool: *Develop a credit scoring model and explain predictions using SHAP or LIME for loan approval decisions.*

Students must:

- Evaluate whether the AI-generated model correctly identifies key risk factors and aligns with domain knowledge (income, credit history, etc.)
- Identify potential issues: check if explanations (SHAP/LIME) are consistent, unbiased, and reflect realistic financial decision logic
- Assess whether the comparison between traditional and AI-based credit scoring is valid and practically justified

Assessment focuses on: Model interpretability, fairness and bias detection, and explanation quality

Assignment 4: Financial Time-Series Forecasting

Students prompt an AI tool: *Predict stock or cryptocurrency prices using ARIMA, LSTM, or Prophet models and evaluate forecasting performance.*

Students must:

- Evaluate whether the AI-selected model appropriately captures trend, seasonality, and volatility in financial time-series data
- Identify potential misinterpretations: check for overfitting, improper train-test split, and unrealistic forecasting assumptions
- Assess whether backtesting strategies and error metrics (RMSE, MAE) are correctly applied and interpreted

Assessment focuses on: Forecasting accuracy, model selection validity, and robustness of evaluation techniques



Embedded AI Systems & Edge Deployment

Course Code:26DS4707

Semester: VII

Program(s): B.Tech CSE (Data Science)

Course Type: Professional Elective (PE-III)

Credits:3

L – T – P – J: 3 – 0 – 0 – 0

Prerequisite(s): Deep Learning

Course Vision

To equip students with the knowledge and practical skills required to design, optimize, and deploy AI models on embedded and edge devices, enabling real-time intelligent applications with efficient resource utilization.

Course Objectives

1. Understand the fundamentals of embedded AI systems and edge computing paradigms
2. Develop lightweight AI/ML models suitable for resource-constrained devices
3. Apply model optimization techniques for efficient edge deployment
4. Explore NVIDIA-based embedded AI platforms and deployment pipelines
5. Design and implement real-time edge AI applications with security and scalability

Course Outcomes (CO)

CO	Description	Blooms Level
CO1	Explain the fundamentals of embedded systems, edge computing, and AI integration	L2
CO2	Develop lightweight AI models and preprocess datasets for edge devices	L6
CO3	Apply model optimization and conversion techniques for efficient deployment	L3
CO4	Implement embedded AI applications using NVIDIA platforms and tools	L3
CO5	Design and deploy end-to-end edge AI systems with real-time constraints and security considerations	L6

Broad Course Description

Embedded AI Systems & Edge Deployment is an advanced elective course designed to bridge artificial intelligence with embedded and edge computing environments. The course focuses on



developing efficient AI models and deploying them on resource-constrained devices such as NVIDIA Jetson and Raspberry Pi platforms. The course is structured into five modules covering foundational concepts, model development, optimization techniques, NVIDIA-based embedded AI programming, and real-time deployment strategies. Students gain hands-on exposure to TensorFlow, PyTorch, TensorRT, CUDA, and DeepStream SDK, enabling them to build scalable and efficient edge AI applications. The course emphasizes real-world applications such as smart surveillance, autonomous systems, IoT analytics, and industrial automation, preparing students for emerging roles in Edge AI engineering.

CO-PO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
C01	2	1			1						1	2	2
C02	2	3	3	2	3						1	3	2
C03	2	3	3	2	3						2	3	3
C04	2	3	3	2	3						2	3	3
C05	2	3	3	3	3						2	3	3
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module 1: Introduction to Embedded AI and Edge Computing (9 Hours)

Introduction to Embedded Systems, Edge AI engineering, Edge Computing vs Cloud Computing, AI at the Edge, IoT and Edge AI architecture, Core components of Embedded systems, Latency and real-time constraints, Edge AI use cases, Overview of Jetson platform, Software stack for embedded AI, Developing Edge AI for Embedded Systems, Fusion of Edge Computing in AI-enabled embedded technologies.

Module 2: AI Model Development for Edge Devices (9 Hours)

Data preprocessing for edge AI, Lightweight ML/DL models, CNNs for embedded vision, Transfer learning, Model training using TensorFlow and PyTorch, Dataset optimization techniques, Introduction to ONNX format and interoperability.

Module 3: Model Optimization and Deployment (9 Hours)

Model compression techniques (quantization, pruning), TensorRT optimization, FP16 and INT8 precision, Model conversion (TensorFlow/PyTorch to TensorRT), Pipeline optimization, Benchmarking inference performance, Hardware acceleration for Edge AI, Introduction to RAG-based optimization, Vision-language models at the edge, Physical computing with Raspberry Pi.

Module 4: Embedded AI using NVIDIA Platforms (9 Hours)

NVIDIA Jetson Nano and Xavier architecture, CUDA programming on embedded devices, DeepStream SDK, GStreamer pipelines, Multi-model deployment, Sensor integration (camera and IoT sensors), Edge device programming and performance tuning.



Module 5: Edge AI Deployment and Real-Time Applications

(9 Hours)

End-to-end AI deployment pipeline, Hardware considerations for Edge AI, Algorithms for Edge AI, Real-time inference workflows, Dataset building strategies, Evaluation and deployment of AI models, Edge-cloud integration, Containerization using Docker, Model monitoring and updates, Security in edge AI systems, Deployment using NVIDIA Jetson and DeepStream.

AI Integration & Authentic Intelligence (AI² Pedagogy)

AI Integration

- **Module 1:** Students use AI tools to simulate edge vs cloud latency scenarios and analyze deployment trade-offs.
- **Module 2:** AI-assisted tools are used to generate and optimize lightweight models and datasets for edge constraints.
- **Module 3:** Students leverage AI tools to automate model optimization (quantization/pruning) and compare performance metrics.
- **Module 4:** AI-assisted deployment pipelines are used with NVIDIA DeepStream to build real-time streaming applications.
- **Module 5:** Students use AI tools to design full deployment architectures and evaluate system performance, scalability, and security.

Assessment Scheme

Criteria	Weightage of Marks
Class Tests	10
Mid-Semester Examination	30
Mini Project	10
Assignments	10
End-Semester Examination	40
Total	100 marks

Recommended Tools

- TensorFlow Lite
- PyTorch Mobile
- NVIDIA TensorRT
- NVIDIA DeepStream SDK
- CUDA Toolkit
- Docker
- ONNX Runtime
- Raspberry Pi & Jetson Nano/Xavier



Primary Textbooks

1. Yifeng Zhu, Embedded Systems with ARM Cortex-M Microcontrollers in Assembly Language and C, 3rd Edition, E-Man Press LLC, 2023.
2. Pete Warden and Daniel Situnayake, TinyML: Machine Learning TensorFlow Lite on Arduino and Ultra-Low-Power Microcontrollers, 1st Edition, O'Reilly Media, 2019.

Additional Resources

1. NVIDIA Deep Learning Institute (DLI), Getting Started with AI on Jetson Nano, NVIDIA Corporation, Self-Paced Online Course, Latest Version.
2. NVIDIA Deep Learning Institute (DLI), Optimization and Deployment of TensorFlow Models with NVIDIA TensorRT, NVIDIA Corporation, Online Training Course.
3. Rajkumar Buyya and Satish Narayana Srirama, Edge AI and IoT: Theory and Practice, 1st Edition, Springer, 2023.
4. Jason Sanders and Edward Kandrot, CUDA by Example: An Introduction to General-Purpose GPU Programming, Updated Edition, Addison-Wesley Professional, 2021.
5. Perry Xiao, Designing Embedded Systems and the Internet of Things (IoT) with the ARM mbed, 2nd Edition, Wiley, 2022.
6. Emmanuel Ameisen, Building Machine Learning Powered Applications, 1st Edition, O'Reilly Media, 2020.

Mini Project Component

Project 1 – Smart Surveillance using Edge AI

Develop a real-time object detection system using Jetson Nano and deploy using TensorRT.

Project 2 – Edge-Based Health Monitoring System

Build an IoT-enabled AI system for real-time health parameter prediction on edge devices.

Project 3 – Industrial Defect Detection

Deploy a CNN-based defect detection model on embedded hardware with optimized inference.

AI-Powered Assignments

Assignment 1: Model Optimization Analysis

Use AI tools to quantize a trained model and evaluate performance improvements.

Assignment 2: Edge Deployment Pipeline

Design an AI-generated pipeline for deploying a model on Jetson and analyze bottlenecks.

Assignment 3: Real-Time Edge AI Evaluation

Evaluate AI-generated deployment architectures for latency, scalability, and security.



Information Security

Course Code: 26DS4708

Semester: VII

Programs: B.Tech CSE (Data Science)

Course Type: Professional Elective (PE - III)

Credits: 3

L – T – P – J: 3 – 0 – 0 – 0

Prerequisites: Cryptography & Network Security

Course Vision

To provide foundational and practical knowledge of information security principles, enabling students to design, analyze, and implement secure systems in modern computing environments.

Course Objectives

1. Understand the fundamentals of information security
2. Analyze legal, ethical, and professional issues in security
3. Explore risk management concepts and methodologies
4. Learn security standards and frameworks
5. Understand technological and architectural aspects of security

Course Outcomes (CO)

CO	Description	Blooms Level
CO1	Explain fundamental concepts of information security	L2
CO2	Analyze legal, ethical, and professional issues in security	L4
CO3	Apply risk management techniques for security analysis	L3
CO4	Understand security standards and frameworks	L2
CO5	Design and implement basic security mechanisms	L6

Broad Course Description

Information Security is a 3-credit Professional Elective offered to B.Tech CSE (Data Science) students. It covers the principles, models, and mechanisms that form the foundation of securing information systems. Topics span the full security lifecycle — from understanding threats and legal issues to designing risk-managed, standards-compliant security architectures.

The course is structured across five units: Introduction to Information Security, Security Investigation (threats, policies, and legal issues), Security Analysis (risk management and access control), Logical Design (standards, blueprints, and business continuity), and Physical Design (IDS, cryptography, and access control devices). Students develop the ability to reason about security trade-offs, apply established frameworks, and design practical protective measures for real-world computing environments.



CO-PO mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
CO1	2	1					1				1	2	1
CO2	2	3					2				1	2	1
CO3	2	3			1		1				1	2	1
CO4	2	2			1		1				2	2	1
CO5	2	3	3	3	2		1				2	2	2
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module I: Introduction to Information Security (9 Hours)

History of information security, Definition and importance of information security, Critical characteristics of information (CIA triad), NSTISSC Security Model, Components of an information system, Securing system components, Balancing security and access, System Development Life Cycle (SDLC), Security SDLC

Module: Security Investigation (9 Hours)

Need for security, Business requirements for security, Threats and attacks, Legal, ethical, and professional issues, Overview of computer security, Access Control Matrix, Security policies, Confidentiality policies, Integrity policies, Hybrid policies

Module III: Security Analysis (9 Hours)

Risk management concepts, Identifying and assessing risks, Risk control strategies, Access control mechanisms, Information flow models, Confinement problem

Module IV: Logical Design (9 Hours)

Security blueprint, Information security policy design, Standards and practices, ISO 17799 / BS 7799, NIST security models, VISA International Security Model, Security architecture design, Business continuity planning

Module V: Physical Design (9 Hours)

Security technologies, Intrusion Detection Systems (IDS), Scanning and analysis tools, Cryptography fundamentals, Access control devices, Physical security mechanisms, Personnel security

Case Study: Analysis of a real-world data breach (e.g., ransomware attack on an enterprise system) focusing on intrusion detection, failure of access control mechanisms, and implementation of cryptographic safeguards to prevent future attacks



AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module I: AI tools are used to explore historical attack timelines, model CIA triad violations, and simulate SDLC security checkpoints. Students analyze AI-generated threat scenarios to understand the importance of balanced security design.

Module II: AI-assisted threat intelligence platforms are used to identify attack patterns and classify security policies. Students evaluate AI-generated policy drafts for completeness and ethical soundness.

Module III: Students use AI tools to perform automated risk scoring on IT assets and evaluate access control configurations. AI outputs are critically reviewed against standard risk matrices and control frameworks.

Module IV: AI tools assist in mapping organizational requirements to ISO/NIST standards and generating security blueprint templates. Students verify alignment with actual frameworks and identify gaps in AI-generated designs.

Module V: AI-powered IDS simulation tools and cryptographic analysis platforms are explored. Students assess AI-generated security configurations for physical and personnel security scenarios and validate against real-world deployment considerations.

Assessment Scheme

Criteria	Weightage of Marks
Class Tests	10
Assignments	10
Mid-Semester Examination	30
Hackathon / Technical Presentation	10
End-Semester Examination	40
Total	100 marks

Recommended Tools

Primary Textbooks

1. Michael E. Whitman and Herbert J. Mattord, Principles of Information Security, Vikas Publishing House, New Delhi, 2003
2. Micki Krause, Harold F. Tipton, Handbook of Information Security Management, Vol 1–3, CRC Press LLC, 2004
3. Stuart McClure, Joel Scrambray, George Kurtz, Hacking Exposed, Tata McGraw-Hill, 2003



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

4. ISO/IEC 27001 Information Security Standards – iso.org/isoiec-27001
5. NIST Special Publication 800-53: Security and Privacy Controls – csrc.nist.gov
6. Matt Bishop, Computer Security: Art and Science, Pearson/PHI, 2002

Additional AI & Computational Resources

1. NPTEL: Information Security – nptel.ac.in
2. SANS Institute Reading Room (free security papers) – sans.org/reading-room
3. OWASP (Open Web Application Security Project) – owasp.org
4. NIST Cybersecurity Framework – nist.gov/cyberframework
5. Cybrary Free Security Courses – cybrary.it



Agentic AI

Course Code: 26DS4704

Semester: VII

Programs: B.Tech CSE (Data Science)

Course Type: Professional Elective (PE-IV)

Credits: 3

Prerequisites: Deep Learning

L – T – P – J: 3 – 0 – 0 – 0

Course Vision

To enable students to understand, design, and implement agentic AI systems with a focus on autonomy, reasoning, decision-making, and intelligent system orchestration for real-world applications.

Course Objectives

1. Understand the evolution, concepts, and architecture of AI agents
2. Analyze levels of autonomy and agent decision-making capabilities
3. Explore reasoning, tool-based actions, and error-handling in agent systems
4. Design multi-agent systems with memory, adaptive behaviour, and personalization
5. Develop practical agent-based AI applications using modern frameworks

Course Outcomes (CO)

CO	Description	Blooms Level
C01	Explain concepts, evolution, and architecture of AI agents	L2
C02	Analyze levels of autonomy and agent decision-making capabilities	L4
C03	Understand reasoning, tool-based actions, and reliability in agents	L2
C04	Design multi-agent and adaptive AI systems with memory and learning	L6
C05	Develop practical agent-based AI applications and orchestration systems	L6

Broad Course Description

Agentic AI is a 3-credit Elective course for CSE-Data Science students in the 7th Semester. It focuses on the design and implementation of autonomous AI agents — systems that perceive their environment, reason over goals, execute tool-based actions, and adapt through feedback. The course spans foundational agent theory through to practical multi-agent system orchestration. The course covers five modules: Introduction and Fundamentals of Agentic AI, Agent Autonomy and Decision-Making, Reasoning and Tool-Based Actions, Memory, Learning and Adaptive Agents, and Multi-Agent Systems with an NVIDIA DLI Case Study. Students develop skills to design agent lifecycles, implement goal-driven pipelines, and build scalable multi-agent systems using Python-based agent frameworks and cloud AI platforms.



CO-PO mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
CO1	2	1			1						1	2	1
CO2	2	3			2						1	3	2
CO3	2	2			2						1	3	2
CO4	2	3	3	2	3						2	3	2
CO5	2	3	3	3	3						2	3	3
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module 1: Introduction and Fundamentals of Agentic AI (9 Hours)

Introduction to AI agents, Evolution of agentic AI, Key terminologies and definitions, Limitations of traditional AI systems, Need for agent-based systems, Role of agents in modern AI, Autonomous systems overview, Agent lifecycle and architecture

Module 2: Agent Autonomy and Decision-Making (9 Hours)

Levels of agent autonomy, Capability spectrum from automation to full autonomy, Decision-making models, Planning strategies, Task decomposition, Goal-driven behaviour, Policy-based and rule-based decisions, Evaluation of agent performance

Module 3: Reasoning and Tool-Based Actions in Agents (9 Hours)

Reasoning techniques in AI agents, Structured reasoning and planning, Decision flows, Tool selection strategies, External environment interaction concepts, Action execution workflows, Reliability in agent reasoning, Error handling and fallback strategies

Module 4: Memory, Learning and Adaptive Agents (9 Hours)

Short-term and long-term memory concepts in agents, Context management, Memory architectures, Learning through feedback loops, Adaptive behaviour and personalization in agents, Continuous improvement strategies, Context-aware decision systems

Module 5: Multi-Agent Systems (NVIDIA DLI Mapped) (9 Hours)

Multi-agent system design principles, Task distribution among agents, Coordination strategies and communication protocols, Distributed agent systems, Orchestration concepts, Real-time decision environments, System scalability and performance, Evaluation of multi-agent systems

Case Study (NVIDIA DLI – Agentic AI): Design and implementation of a multi-agent system capable of coordinating tasks, making autonomous decisions, using tool-based actions, and adapting to dynamic environments — demonstrated through an end-to-end orchestrated agent workflow



AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module 1: Students use AI tools to explore agent architecture diagrams and simulate agent lifecycle stages. AI-generated explanations of autonomous system limitations are critically evaluated against real-world deployment constraints and responsible AI guidelines.

Module 2: AI-assisted decision-making simulators are used to compare autonomy levels and test planning strategies across different task scenarios. Students critically evaluate AI-generated task decomposition plans for completeness, feasibility, and alignment with goal-driven behaviour principles.

Module 3: Students use AI-powered reasoning frameworks (ReAct, Chain-of-Thought) to observe structured decision flows and tool selection behaviour. AI-generated action execution traces are critically reviewed for logical consistency, error recovery quality, and fallback strategy effectiveness.

Module 4: AI tools are used to simulate short-term and long-term memory management in agent systems and explore feedback-driven adaptation. Students evaluate AI-generated memory retrieval strategies and adaptive behaviour policies for context retention accuracy and personalization effectiveness.

Module 5: Students design multi-agent systems using orchestration frameworks and NVIDIA DLI resources. AI-generated coordination protocols and task distribution plans are critically assessed for scalability, communication efficiency, and safe operation in real-time dynamic environments.

Assessment Scheme based on the course

Criteria	Weightage of Marks
Class Tests / Assignments	10
Mid-Semester Examination	30
Mini Project	10
Research Paper	10
End-Semester Examination	40
Total	100 marks

Recommended Tools

Primary Textbooks

1. Pascal Bornet et al., Agentic Artificial Intelligence: Harnessing AI Agents to Reinvent Business, Work and Life, Wiley, 2024
2. NVIDIA Deep Learning Institute Resources – Agentic AI – learn.nvidia.com
3. Stuart Russell & Peter Norvig, Artificial Intelligence: A Modern Approach, 4th Edition, Pearson, 2020



4. AutoGen: Enabling Next-Gen LLM Applications via Multi-Agent Conversation – arxiv.org/abs/2308.08155
5. LangChain Agents Documentation – python.langchain.com/docs/modules/agents
6. Michael Wooldridge, An Introduction to MultiAgent Systems, 2nd Edition, Wiley, 2009

Additional AI & Computational Resources

1. NPTEL: Artificial Intelligence – nptel.ac.in
2. LangGraph Documentation – langchain-ai.github.io/langgraph
3. AutoGen Studio – microsoft.github.io/autogen
4. CrewAI Documentation – docs.crewai.com
5. Papers With Code – AI Agents – paperswithcode.com/task/autonomous-agents

Mini Project Component

The following projects are suggested based on course outcomes. Students may attempt one project individually or in pairs.

Project 1 – Autonomous Task Planning Agent

Problem: Design and implement a goal-driven agent that decomposes a complex task (e.g., research report generation) into subtasks and executes them autonomously using a Python agent framework

Concepts Used: Task decomposition, goal-driven behaviour, tool selection, decision flows, ReAct framework

Industrial Relevance: Intelligent RPA systems, automated research assistants, enterprise workflow automation

Project 2 – Adaptive Personalized Learning Agent

Problem: Build a memory-enabled agent that personalizes study recommendations for a student based on interaction history and adaptive feedback loops

Concepts Used: Short-term and long-term memory, context management, adaptive behaviour, feedback-driven learning

Industrial Relevance: EdTech personalization, intelligent tutoring systems, HR training and development

Project 3 – Multi-Agent Coordination System

Problem: Design a multi-agent system with at least three specialized agents (e.g., researcher, planner, executor) that coordinate to solve a complex problem through communication and task distribution

Concepts Used: Multi-agent design, coordination protocols, orchestration, task distribution, scalability evaluation

Industrial Relevance: Autonomous business process automation, AI-powered operations centers, distributed intelligent systems

AI-Powered Assignments

Assignment 1: AI Agent Architecture Evaluation

Students prompt an AI tool: Design an agentic AI system to automate end-to-end customer support, including query classification, response generation, and escalation handling.



Students must:

- Analyze whether the proposed agent lifecycle and decision flow are complete and logically sound
- Identify gaps in error handling, fallback strategies, and edge-case coverage
- Improve the architecture by incorporating appropriate autonomy levels and safety guardrails

Assessment focuses on: agent architecture reasoning, lifecycle design completeness, and critical evaluation of autonomy decisions

Assignment 2: Reasoning and Tool-Use Strategy Analysis

Students prompt an AI tool: Build an agent that answers complex factual questions by selecting and using appropriate tools (web search, calculator, document retrieval).

Students must:

- Evaluate whether the tool selection logic is appropriate for each query type and context
- Identify reasoning failures such as circular logic, hallucinated tool outputs, or incorrect fallback behaviour
- Refine the reasoning chain using structured prompting (Chain-of-Thought, ReAct) and validate against ground-truth answers

Assessment focuses on: reasoning chain quality, tool selection logic, and critical validation of agent-generated outputs

Assignment 3: Multi-Agent System Design Critique

Students prompt an AI tool: Design a multi-agent pipeline for an automated market research system where agents gather data, analyze trends, and generate a final report.

Students must:

- Assess whether the task distribution and coordination protocol support efficient parallel execution
- Identify communication bottlenecks, single points of failure, and unsafe autonomous behaviours
- Improve the design by adding proper orchestration, agent monitoring, and output validation strategies

Assessment focuses on: multi-agent system design understanding, coordination reasoning, and critical evaluation of scalability and safety



AI for Cybersecurity Analytics

Course Code: 26DS4710

Semester: VII

Programs Offered: B. Tech CSE (Data Science)

Course Type: Professional Elective (PE-IV)

Credits: 3

L – T – P – J: 3-0-0-0

Prerequisites: Python Programming, Basic Machine Learning, Computer Networks, Operating Systems Fundamentals

Course Vision

To equip students with the theoretical foundations and practical competencies to apply AI and machine learning techniques for detecting, analyzing, and responding to cybersecurity threats, enabling intelligent and adaptive security operations using open-source analytics tools and GPU-accelerated threat intelligence platforms aligned with industry practices.

Course Objectives

1. Understand cybersecurity data types, threat taxonomies, and the application of AI in security operations
2. Apply machine learning techniques for network intrusion detection, malware classification, and anomaly detection
3. Design AI-powered threat intelligence systems for log analysis, phishing detection, and vulnerability assessment
4. Build end-to-end security analytics pipelines integrating SIEM systems, open-source threat feeds, and ML models
5. Evaluate AI-based security systems for performance, adversarial robustness, and ethical compliance
6. Apply GPU-accelerated machine learning and NVIDIA DLI-aligned frameworks for large-scale cybersecurity analytics

Course Outcomes (CO)

CO	Description	Blooms Level
CO1	Explain cybersecurity data types, threat taxonomies, and the role of AI in security operations and threat detection	L2
CO2	Apply machine learning models for network intrusion detection, malware classification, and anomaly detection in security logs	L3
CO3	Design AI-powered threat intelligence and incident response systems using NLP, graph analytics, and behavioral profiling	L6



CO4	Build end-to-end security analytics pipelines integrating open-source SIEM tools, threat feeds, and ML inference systems	L6
CO5	Apply GPU-accelerated ML and NVIDIA DLI-aligned frameworks for large-scale cybersecurity data analytics and threat detection	L3

Broad Course Description

This course provides a comprehensive foundation in applying artificial intelligence and machine learning to cybersecurity challenges. Students learn to work with diverse security data sources including network traffic logs, system event logs, malware samples, and threat intelligence feeds, applying AI techniques to detect, classify, and respond to evolving threats. Core analytical techniques include supervised and unsupervised learning for intrusion detection, NLP-based phishing and malware analysis, graph-based threat intelligence, and deep learning for user and entity behaviour analytics (UEBA). Students build hands-on skills using open-source tools including Scikit-learn, PyTorch, Hugging Face, NetworkX, and ELK Stack.

The course addresses adversarial machine learning, where attackers adapt to evade AI-based detectors, alongside fairness, privacy, and legal dimensions of AI in security contexts. Responsible AI considerations including explainability of alert systems, bias in threat profiling, and accountability in automated response are embedded throughout.

The final module is aligned with the NVIDIA Deep Learning Institute (DLI), enabling students to apply GPU-accelerated analytics for large-scale log processing, real-time threat detection, and deployment of security AI models using NVIDIA RAPIDS, TensorRT, and Triton Inference Server in production security operations centres.

CO-PO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	2	1			1						1	2	2
CO2	2	3			2						1	3	2
CO3	2	3	3	2	3						2	3	2
CO4	2	3	3	2	3						2	3	3
CO5	2	3	3	3	3						2	3	3
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													



Course Modules

Module 1: Cybersecurity Data and AI Foundations (9 Hours)

Cybersecurity threats overview (malware, phishing, DDoS, insider and APT attacks, zero-day exploits), security data sources (network traffic/PCAP, system logs, DNS, threat intelligence feeds), threat models (MITRE ATT&CK, kill chain), data preprocessing (log parsing, feature extraction, encoding), class imbalance in security datasets, key datasets (NSL-KDD, CICIDS-2017, UNSW-NB15, EMBER), SOC and SIEM applications, AI integration in security workflows (Scikit-learn, Pandas, MISP, OpenCTI), and challenges in authentic intelligence (label noise, adversarial poisoning, ethical and legal issues).

Module 2: Network Intrusion Detection and Anomaly Detection (9 Hours)

Intrusion detection systems (signature, anomaly, hybrid), ML for IDS (decision trees, random forest, XGBoost, SVM), anomaly detection (Isolation Forest, One-Class SVM, autoencoders), deep learning for IDS (LSTM models), network flow feature engineering, IDS evaluation metrics (detection rate, FPR, F1, MCC), concept drift, applications in network/cloud/IoT security, AI tools and frameworks (Scikit-learn, PyTorch, Suricata, Zeek, CICFlowMeter), and challenges (false positives, adversarial attacks, privacy, drift).

Module 3: Malware Analysis and Threat Intelligence (9 Hours)

Malware analysis (static: PE features; dynamic: API/system call traces), ML for malware detection (random forests, CNN on binaries, LSTM sequences), NLP for cybersecurity (phishing detection with TF-IDF/transformers, threat report NER), graph-based threat intelligence (attack graphs, malware clustering using NetworkX), UEBA (behaviour baselining and anomaly detection), explainability (SHAP, attention maps), applications (EDR, phishing detection, CTI enrichment), AI tools (Hugging Face, NetworkX, EMBER pipelines), and challenges (adversarial evasion, false positives, attribution bias, dual-use risks).

Module 4: Security Analytics Pipelines and Incident Response (9 Hours)

Security analytics pipeline (log ingestion, enrichment, correlation, alerting, response), SIEM and ELK Stack (Elasticsearch, Logstash, Kibana), ML-based alert triage and prioritisation, SOAR and automated incident response, threat hunting with ML, model monitoring (drift detection, alert fatigue), evaluation metrics (MTTD, MTTR, detection coverage), federated learning for privacy-preserving analytics, applications in SOC/cloud security, AI tools (ELK, FastAPI, SOAR platforms like Shuffle and The Hive), and challenges (accountability, explainability, bias, legal constraints in automated response)

Module 5: GPU-Accelerated Cybersecurity Analytics and NVIDIA DLI-Aligned Deployment (9 Hours)

GPU-accelerated security analytics using NVIDIA RAPIDS (cuDF, cuML), large-scale network traffic processing on GPU clusters, deep learning for IDS and malware detection using PyTorch with multi-GPU (DDP), model optimization for real-time inference (quantization, pruning,



TensorRT), deployment with NVIDIA Triton Inference Server, benchmarking GPU vs CPU pipelines, and SOC MLOps integration. Hands-on NVIDIA DLI-aligned workflows covering RAPIDS-based log processing, GPU model training, Triton deployment, and end-to-end security pipeline benchmarking, with applications in real-time SOC analytics, large-scale malware detection, and GPU-accelerated threat intelligence systems.

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module 1: Students parse and preprocess security logs using Pandas and Scikit-learn pipelines. AI-assisted tools are used to explore MITRE ATT&CK mappings and enrich threat data with open-source CTI feeds.

Module 2: Students build network intrusion detection systems using Scikit-learn and PyTorch. AI-generated IDS outputs are critically evaluated for false positives, evasion vulnerabilities, and concept drift under evolving traffic patterns.

Module 3: Students train malware classifiers and phishing detectors using static features and Hugging Face transformers. SHAP-based explainability tools are used to interpret model decisions and validate against analyst knowledge.

Module 4: Students build end-to-end security analytics pipelines using ELK Stack and FastAPI. AI-assisted alert triage and automated response tools are critically evaluated for accountability and operator trust.

Module 5: Students apply NVIDIA RAPIDS for GPU-accelerated security log processing and deploy threat detection models using Triton Inference Server. NVIDIA DLI-aligned labs are used to benchmark and optimize production-grade SOC AI pipelines.

Authentic Intelligence Reflection (AI² Pedagogy):

Module 1: Understanding the legal and ethical constraints on security data collection including privacy laws, data sovereignty, and surveillance boundaries. Students evaluate risks of adversarial data poisoning in security training datasets.

Module 2: Recognizing that adversarial attackers actively adapt to evade ML-based detectors. Students assess the arms-race dynamic between AI-based detection and adversarial evasion techniques.

Module 3: Evaluating dual-use risks of malware analysis tools and bias in threat actor attribution systems. Students reflect on responsible disclosure and the ethics of automated threat intelligence sharing.

Module 4: Understanding accountability and legal liability in automated incident response. Students Analyze the risk of AI-driven blocking decisions causing collateral damage to legitimate users or services.



Module 5: Evaluating the governance and auditability of GPU-accelerated security AI systems deployed in production SOC environments. Students reflect on the carbon cost of large-scale security AI infrastructure and responsible deployment practices.

Assessment Scheme based on the course

Criteria	Weightage of Marks
Class Tests	10
Mid-Semester Examination	30
Mini Project	10
Assignments	10
End-Semester Examination	40
Total	100 marks

Recommended Tools

1. Python 3.10+
2. Pandas, NumPy, Scikit-learn
3. PyTorch (LSTM, CNN, autoencoder models)
4. CICFlowMeter, Suricata, Zeek
5. FastAPI (alert triage API deployment)
6. NVIDIA RAPIDS (cuDF, cuML)
7. NVIDIA TensorRT

Primary Textbooks

1. NVIDIA Corporation, NVIDIA Deep Learning Institute (DLI) Course Materials and Documentation, Latest Edition (Online Resource).
2. Ali Dehghantanha, Kim-Kwang Raymond Choo, Reza M. Parizi, Handbook of Big Data Privacy, Springer, 2020.
3. Clarence Chio, David Freeman, Machine Learning and Security, O'Reilly Media, 1st Edition, 2018.
4. Ian Goodfellow, Yoshua Bengio, Aaron Courville, Deep Learning, MIT Press, 1st Edition, 2016.
5. Aurélien Géron, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, O'Reilly Media, 3rd Edition, 2022.



Additional Resources

1. MITRE ATT&CK Framework – attack.mitre.org
2. NVIDIA RAPIDS Documentation – rapids.ai/docs
3. Hugging Face Documentation – huggingface.co/docs
4. CICIDS-2017 and NSL-KDD Cybersecurity Datasets – University of New Brunswick
5. EMBER Malware Dataset – [endgameinc/ember](https://github.com/endgameinc/ember) on GitHub

Mini Project Component

Industry-Driven Projects:

Project 1 – AI-Powered Network Intrusion Detection System

- Problem: Build a complete ML-based IDS that detects network intrusions in real-time using network flow data.
- Concepts Used: Random forest and XGBoost for classification, autoencoder for anomaly detection, CICFlowMeter, CICIDS-2017 dataset.

Project 2 – Deep Learning Malware Classification Engine

- Problem: Develop a malware classification system using static PE features and CNN-based binary visualisation.
- Concepts Used: PE file feature extraction, CNN on binary images, EMBER dataset, SHAP-based explainability.

Project 3 – Phishing Detection and Threat Intelligence System

- Problem: Build a phishing email and URL detection pipeline using NLP and integrate it with an open-source CTI feed.
- Concepts Used: Hugging Face transformer-based phishing classifier, URL feature extraction, MISP CTI integration.

Project 4 – User and Entity Behaviour Analytics (UEBA) Platform

- Problem: Design a UEBA system that detects insider threats and compromised accounts using behavioural baseline models.
- Concepts Used: Baseline modelling, Isolation Forest, LSTM-based sequence anomaly detection, CERT insider threat dataset.

Project 5 – Threat Intelligence Graph Analytics System

- Problem: Build an attack graph system that maps threat actor TTPs to the MITRE ATT&CK framework using graph analytics.
- Concepts Used: NetworkX attack graph construction, graph-based malware clustering, NER for threat report parsing.



AI-Powered Assignments

Assignment 1: AI-Based Cybersecurity Dataset Quality and Bias Analysis

Students prompt an AI tool: Analyze a cybersecurity dataset for quality issues, class imbalance, and potential labelling bias, and recommend a preprocessing strategy.

Students must:

- Evaluate the correctness of AI-identified data quality issues (label noise, class imbalance, feature leakage)
- Identify missing preprocessing steps such as flow aggregation or categorical encoding for network attributes
- Improve the pipeline using SMOTE, cost-sensitive learning, and CICIDS-2017 benchmark validation

Assessment focuses on security data engineering, bias awareness, and critical evaluation of AI-generated preprocessing strategies.

Assignment 2: AI-Assisted Intrusion Detection System Design

Students prompt an AI tool: Recommend the best ML approach for detecting network intrusions in a given traffic dataset and justify the choice.

Students must:

- Analyze the correctness of the AI-recommended approach (supervised vs. anomaly-based vs. hybrid)
- Identify incorrect assumptions about data availability, label quality, or attack diversity
- Validate results using NSL-KDD or CICIDS-2017 and evaluate using detection rate, F1, and false positive rate

Assessment focuses on IDS design reasoning, method selection, and critical evaluation of AI-generated detection strategies.

Assignment 3: AI-Based Malware Classifier Design and Evaluation

Students prompt an AI tool: Design a malware classification pipeline using static features and justify the model and feature selection choices.

Students must:

- Evaluate the correctness of AI-recommended feature sets (PE headers, import tables, entropy) and model architecture
- Identify adversarial evasion risks in the proposed classifier (e.g., feature manipulation, obfuscation)
- Improve the pipeline using EMBER dataset and SHAP-based explainability analysis

Assessment focuses on malware analysis knowledge, adversarial robustness awareness, and model interpretability.

Assignment 4: AI-Assisted Phishing Detection System Using NLP

Students prompt an AI tool: Build a phishing email detection system using NLP techniques and recommend the most appropriate model architecture.



Students must:

- Analyze AI-suggested NLP approaches (TF-IDF vs. transformer-based classifiers) for phishing detection
- Identify risks of false positives in legitimate email filtering and bias in training corpora
- Improve results using Hugging Face transformers and evaluate precision-recall trade-offs

Assessment focuses on NLP for security, false positive management, and responsible automated filtering decisions.

Assignment 5: AI-Based Security Analytics Pipeline Design

Students prompt an AI tool: Design an end-to-end security analytics pipeline for a SOC environment including log ingestion, ML-based triage, and alerting.

Students must:

- Analyze the feasibility of the AI-proposed architecture (ELK ingestion, ML triage, automated response)
- Identify bottlenecks such as alert fatigue, lack of explainability, or missing drift monitoring
- Refine the design using ELK Stack and FastAPI with an analyst-facing explainability dashboard

Assessment focuses on SOC pipeline design, alert triage reasoning, and operator-centred system thinking.

Assignment 6: AI-Driven GPU-Accelerated Threat Detection Pipeline Evaluation

Students prompt an AI tool: Design and evaluate a GPU-accelerated cybersecurity analytics pipeline from log preprocessing to deployed threat detection inference.

Students must:

- Analyze the correctness of all pipeline stages (cuDF log processing → cuML feature extraction → TensorRT → Triton deployment)
- Identify gaps in deployment strategy (missing latency benchmarking, lack of model monitoring, no explainability for SOC analysts)
- Improve the solution using NVIDIA RAPIDS and Triton Inference Server with throughput and latency profiling

Assessment focuses on GPU pipeline design for security, NVIDIA toolchain integration, and critical evaluation of AI-generated SOC workflows.



Geo-Spatial AI

Course Code: 26DS4711

Semester: VII

Programs Offered: B.Tech CSE (Data Science)

Course Type: Professional Elective (PE-IV)

Credits :3

L – T – P – J: 3 – 0 – 0 – 0

Prerequisites: Python Programming Fundamentals

Course Vision

To develop a strong foundation in geospatial analytics and GeoAI by integrating spatial data analysis, visualization, and AI-driven techniques, enabling students to solve real-world location-based problems using modern tools and intelligent systems.

Course Objectives

1. Develop understanding of geospatial data concepts and coordinate systems.
2. Enable students to perform spatial data ingestion and preprocessing.
3. Equip students with techniques for spatial operations and analysis.
4. Provide knowledge of geospatial visualization and mapping techniques.
5. Enable students to build GeoAI-driven applications for real-world scenarios.

Course Outcomes (CO)

CO	Description	Blooms Level
CO1	Explain geospatial data concepts and coordinate systems	L2
CO2	Perform geospatial data ingestion and preprocessing	L3
CO3	Apply spatial operations such as joins, buffers, and overlays	L3
CO4	Visualize spatial data using maps and analytical tools	L3
CO5	Develop GeoAI-based analytical solutions	L6

Broad Course Description

Geospatial Analytics and GeoAI is an interdisciplinary course that focuses on spatial data processing, analysis, and visualization using modern computational tools. The course introduces students to geospatial data types, coordinate systems, spatial operations, and mapping techniques. It further integrates artificial intelligence with geospatial data to develop intelligent location-based systems. Students gain hands-on experience in solving real-world problems such as urban planning, infrastructure development, and smart city applications through GeoAI techniques.



CO-PO mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
CO1	3	2	1	1	2	1			1	1		2	2
CO2	3	3	2	2	3	1			1	1		2	3
CO3	3	3	3	2	3	1			1	1		2	3
CO4	2	2	2	2	3	1			2	2		2	2
CO5	3	3	3	3	3	2	1	1	2	2	2	3	3

Course Modules

Module – 1: Introduction to Geospatial Data (9 Hours)

Introduction to geospatial data; types of spatial data including vector and raster; coordinate reference systems; map projections; GeoDataFrame concepts; reading shapefiles and GeoJSON; applications in GIS and urban planning.

Module – 2: Geospatial Data Ingestion and Processing (9 Hours)

Data ingestion techniques; GeoPandas basics; data cleaning; CRS transformations; handling projection mismatches; data validation and preprocessing; applications in data pipelines and location analytics.

Module – 3: Spatial Operations and Analysis (9 Hours)

Overlay operations; spatial joins; buffer analysis; attribute aggregation; intersection and union operations; proximity analysis; applications in land use analysis and infrastructure planning.

Module – 4: Geometric Computations and Visualization (9 Hours)

Area and distance calculations; planar vs geodesic measurements; CRS impact on computations; map visualization using Matplotlib; cartographic elements; applications in thematic mapping and distance-based services.

Module – 5: GeoAI and Interactive Mapping (9 Hours)

Introduction to GeoAI concepts; spatial feature engineering; basic geospatial modeling; interactive maps using Folium; visualization of spatial insights; applications in smart cities and location intelligence systems.

Case Studies:

- Smart city traffic analysis using GeoAI
- Disaster management using geospatial data
- Urban land-use planning using GIS
- Location-based service optimization



AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module-1: Application of unsupervised learning (K-Means, DBSCAN) and spatial clustering techniques to identify geographic patterns, hotspots, and spatial correlations. Use of geospatial big data for urban analytics and environmental monitoring.

Module-2: Use of machine learning models for automated data cleaning, anomaly detection, and CRS correction. Implementation of spatial ETL pipelines with AI-assisted preprocessing and data validation frameworks.

Module-3: Application of supervised learning models (Regression, Random Forest, Gradient Boosting) for predicting spatial phenomena such as traffic density, land-use classification, and environmental risks.

Module-4: Automated cartographic design using AI, intelligent map generalization, and visualization optimization. Integration of computer vision techniques for satellite imagery interpretation.

Module-5: Development of GeoAI systems using deep learning (CNN for satellite imagery, LSTM for spatio-temporal data). Applications in smart cities, disaster prediction, and location intelligence platforms with explainable AI (XAI).

Assessment Scheme based on the course

Criteria	Weightage of Marks
Mid-Semester Examination	40
Mini Project	10
Assignments	10
End-Semester Examination	40
Total	100 marks

Recommended Tools

Primary Textbooks

1. Edgar Milvus, *Geospatial AI (GeoAI) with Python Programming*, 2026
2. Standard GIS and Spatial Data Science textbooks

Additional Resources

1. GeoPandas Documentation
2. GDAL/OGR Documentation
3. Research papers in GeoAI and GIS
4. GeoPandas, QGIS, ArcGIS, Python, Folium, Google Earth Engine



Mini Project Component

The following projects are suggested based on course outcomes. Students may attempt one project individually or in pairs.

Project 1 – Smart Traffic Prediction System

Problem: Develop a spatio-temporal machine learning model to forecast traffic patterns using historical and real-time geospatial data. Implement sequence models such as LSTM and preprocess spatial data using GeoPandas.

Concepts Used: Spatio-temporal modeling, time-series forecasting, LSTM networks, geospatial feature engineering, traffic flow analysis, data preprocessing

Industrial Relevance: Smart city traffic management, intelligent transportation systems, congestion prediction, route optimization

Project 2 – Disaster Prediction & Risk Mapping System

Problem: Design a GeoAI-based system to predict disaster-prone zones (floods or earthquakes) using satellite imagery and environmental datasets, and generate spatial risk maps.

Concepts Used: Geospatial analysis, remote sensing, predictive modeling, spatial interpolation, risk mapping, environmental data analysis

Industrial Relevance: Disaster management agencies, early warning systems, climate risk assessment, emergency planning

Project 3 – Land Use & Land Cover Classification

Problem: Apply deep learning models (CNN) to classify satellite imagery into land-use and land-cover categories and generate accurate thematic maps.

Concepts Used: Convolutional Neural Networks (CNN), image classification, satellite image processing, feature extraction, accuracy assessment (confusion matrix, IoU)

Industrial Relevance: Urban planning, agriculture monitoring, environmental conservation, land resource management

Project 4 – Location-Based Recommendation System

Problem: Develop a recommendation system using spatial data to suggest locations (restaurants, tourist spots, services) based on user preferences and geographic proximity.

Concepts Used: Spatial clustering, similarity measures, recommendation systems, geospatial analytics, user profiling, distance metrics

Industrial Relevance: E-commerce platforms, tourism applications, smart city services, location-based marketing

AI-Powered Assignments

Assignment 1: Spatial Clustering & Hotspot Detection

Students prompt an AI tool: Perform clustering on a geospatial dataset using K-Means or DBSCAN, identify spatial hotspots, and evaluate clustering performance.

Students must:

- Evaluate whether the AI-generated clustering correctly identifies spatial patterns and hotspot regions based on geographic distribution
- Identify potential misinterpretations: check if clustering ignores spatial autocorrelation, density variations, or parameter sensitivity (e.g., epsilon in DBSCAN)
- Assess whether clustering evaluation metrics are appropriate for spatial data (e.g., silhouette vs spatial validity measures)



Assessment focuses on: Spatial clustering correctness, hotspot detection validity, and evaluation metric suitability

Assignment 2: Geospatial Data Preprocessing Pipeline

Students prompt an AI tool: Build an AI-driven ETL pipeline using GeoPandas for preprocessing geospatial datasets.

Students must:

- Evaluate whether the pipeline correctly handles CRS transformations, missing values, and spatial inconsistencies
- Identify potential issues: check for incorrect coordinate transformations, data loss, or improper anomaly handling
- Assess whether ML-based validation methods effectively detect outliers and inconsistencies in spatial data

Assessment focuses on: Data preprocessing accuracy, spatial data integrity, and robustness of validation mechanisms

Assignment 3: GeoAI-Based Predictive Modeling

Students prompt an AI tool: Develop predictive models for spatial applications such as traffic prediction, pollution estimation, or land-use classification.

Students must:

- Evaluate whether the AI-generated models correctly capture spatial dependencies and feature relationships
- Identify potential misinterpretations: check for overfitting, lack of spatial features, or ignoring temporal-spatial interactions
- Assess whether feature importance explanations are meaningful and aligned with real-world geographic patterns

Assessment focuses on: Model accuracy, spatial reasoning, and interpretability of predictions

Assignment 4: Satellite Image Analysis using Deep Learning

Students prompt an AI tool: Apply CNN-based models to satellite imagery for land classification or object detection tasks.

Students must:

- Evaluate whether the AI-generated model correctly identifies land cover classes or spatial objects from satellite images
- Identify potential issues: check for data imbalance, resolution limitations, or misclassification due to similar textures
- Assess whether evaluation metrics (accuracy, IoU, F1-score) reflect true spatial performance

Assessment focuses on: Deep learning model performance, spatial accuracy, and robustness in image-based analysis



Scalable Enterprise AI

Course Code: 26DS4712

Semester: VII

Programs Offered: B. Tech CSE (Data Science)

Course Type: Professional Elective (PE-IV)

Credits: 3

L – T – P – J: 3 – 0 – 0 – 0

Prerequisites: Basic knowledge of programming (Python), machine learning fundamentals, data structures, and familiarity with cloud computing and data handling concepts.

Course Vision

To provide learners with the knowledge and skills required to design, develop, and deploy scalable artificial intelligence solutions for enterprise environments, this course aims to prepare industry-ready professionals capable of building robust, secure, and high-performance AI systems using modern cloud platforms, distributed computing, and big data technologies. It fosters innovation through the exploration of advanced AI architectures, MLOps practices, and real-time intelligent systems, while enabling learners to integrate AI into enterprise workflows, ensure ethical and responsible AI deployment, and support data-driven decision-making at scale in real-world organizational contexts.

Course Objectives

1. Understand enterprise AI ecosystems and scalability challenges.
2. Design scalable data pipelines for enterprise AI systems
3. Develop scalable and efficient machine learning models.
4. Implement scalable, secure, and responsible enterprise AI solutions.
5. Deploy AI models in real-world enterprise environments
6. Build generative AI and LLM-based applications.

Course Outcomes (CO)

CO	Description	Blooms Level
CO1	Explain enterprise AI architecture and scalability principles	L2
CO2	Design and implement scalable data pipelines	L3
CO3	Design scalable data pipelines and distributed training systems	L6
CO4	Implement MLOps practices for production AI systems	L6
CO5	Deploy enterprise AI applications including LLM-based systems	L3

Broad Course Description

This course provides a comprehensive foundation for designing, developing, and deploying scalable artificial intelligence solutions within enterprise environments. It begins with an introduction to enterprise AI, digital transformation, and the challenges of scaling AI systems,



including data, infrastructure, and governance, while exploring modern AI architectures and distributed systems. The course then focuses on scalable data engineering practices, covering enterprise data architectures such as data lakes and warehouses, ETL/ELT pipelines, big data frameworks, real-time data streaming, and data governance with privacy and compliance considerations. Building on this, students will learn scalable machine learning techniques, including distributed training, model optimization, and the use of advanced frameworks to handle large-scale datasets and real-time intelligent applications.

The course further explores enterprise-level applications of large language models (LLMs), including their architecture, deployment, fine-tuning, and operationalization (LLMOps), along with security and governance aspects. Finally, it addresses AI infrastructure and enterprise platforms, focusing on large-scale deployment strategies, GPU-based computing, cloud and hybrid AI architectures, and decision automation across various business domains such as customer intelligence, predictive maintenance, and dynamic pricing. Through real-world applications and case studies, the course equips learners with the skills required to build robust, scalable, and production-ready AI systems that drive innovation and data-driven decision-making in modern enterprises.

CO PO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
C01	2	1			1						1	3	2
C02	2	3	3	2	3						2	3	3
C03	2	3	3	2	3						2	3	3
C04	2	3	3	2	3						2	3	3
C05	2	3	3	3	3						2	3	3
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module-1 Introduction to Enterprise AI & Scalability

(9 hours)

Introduction to Enterprise AI and digital transformation, AI adoption lifecycle in enterprises, Data-driven decision making, Challenges in scaling AI (data, infrastructure, governance), Overview of cloud platforms for AI (Amazon Web Services, Microsoft Azure, Google Cloud Platform), AI architecture patterns (monolithic vs microservices), Overview of distributed systems.

Module-2 Scalable Data Engineering & Pipelines

(9 Hours)

Enterprise data architecture (data lakes, data warehouses, lakehouse), ETL/ELT pipelines and orchestration, Big data frameworks (Apache Hadoop, Apache Spark), Real-time data streaming (Apache Kafka), Data preprocessing, feature engineering at scale, Data Governance, Privacy (GDPR/CCPA), and Compliance.



Module-3 Scalable Machine Learning and Model Development (9 Hours)

Distributed machine learning concepts, Parallel training and model optimization Frameworks for scalable ML (TensorFlow, PyTorch), Transfer learning and pre-trained models, Hyperparameter tuning at scale, Handling large-scale datasets and model versioning.

Module-4 Enterprise LLM Applications (9 Hours)

Enterprise LLM Architecture, Scalable Inference & Deployment, Fine-Tuning & Customization, LLMops, Security & Governance.

Module-5 AI Infrastructure & Enterprise Platforms (9 Hours)

Decision and process automation in enterprise operations, Production operations and IoT, revenue and inventory management, customer intelligence, content intelligence, visual search, product recommendations, knowledge management, synthetic media, demand forecasting, price and promotion optimization, dynamic pricing, inventory optimization, anomaly detection, visual quality control, loss functions, AI infrastructure design, GPU clusters, DGX systems, cloud AI platforms, hybrid AI architecture, scalability strategies.

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module-1: Introduction to Enterprise AI & Scalability

Students utilize AI-powered platforms (e.g., Python/Colab, cloud AI sandboxes, generative AI tools) to analyze enterprise AI use cases and simulate digital transformation strategies. AI assistants are used to design scalable AI architectures (monolithic vs microservices) and evaluate distributed system performance. Intelligent tools help in identifying challenges in scaling AI such as data bottlenecks, infrastructure limitations, and governance issues.

Module-2: Scalable Data Engineering & Pipelines

Students implement AI-assisted data engineering workflows using big data tools (Apache Spark, Hadoop, Kafka) and cloud platforms. AI tools support automated ETL/ELT pipeline generation, feature engineering at scale, and anomaly detection in streaming data. Students also use AI-based governance tools to ensure data privacy, compliance (GDPR/CCPA), and efficient data lifecycle management in enterprise systems.

Module-3: Scalable Machine Learning and Model Development

Students apply AI frameworks (TensorFlow, PyTorch) with AI-assisted coding tools to develop and optimize machine learning models at scale. AI platforms are used for distributed training, hyperparameter tuning, and model versioning. Students leverage pre-trained models and transfer learning techniques, while AI tools assist in handling large-scale datasets and improving model performance for real-time applications..

Module-4: Enterprise LLM Applications

Students explore enterprise-grade large language model (LLM) applications using generative AI platforms and APIs. AI tools are used for building enterprise chatbots, cognitive search systems,



and automated code review solutions. Students experiment with LLM fine-tuning, prompt engineering, and scalable deployment, while also analyzing AI governance, security, and ethical considerations in enterprise environments.

Module-5: AI Infrastructure & Enterprise Platforms

Students utilize AI-enabled cloud platforms and GPU-based systems to design scalable AI infrastructure. AI tools assist in deploying large-scale AI models, optimizing performance, and automating enterprise processes such as demand forecasting, anomaly detection, and recommendation systems. Students also explore hybrid AI architectures, digital twins, and real-time enterprise applications using AI-driven monitoring and optimization tools.

Assessment Scheme

Criteria	Weightage of Marks
Lab Reports (experiments)/Project	10
Class Tests /assignments	10
Mid-Semester Examination	30
Hackathon/Technical Competition/ Paper presentation/Global/DLI certifications	10
End-Semester Examination	40
Total	100 marks

Recommended Tools

Primary Textbooks

1. Sreedhar, S. (2025). Enterprise AI at Scale: Architecting Distributed Machine Learning Systems. O'Reilly Media.
<https://www.oreilly.com/library/view/enterprise-ai-at/9781492090000/>
2. Lakshmanan, S. (2024). Data Engineering for AI: Building Scalable Pipelines with Spark, Kafka, and Lakehouse. Google Cloud Press.
<https://proquest.safaribooksonline.com/book/programming/9781098150000>
3. Rothman, D. (2026). Generative AI for the Enterprise: Implementing LLMs, RAG, and LLMops. Packt Publishing.
<https://www.packtpub.com/product/generative-ai-for-the-nterprise/9781805120000>.
4. Kleppmann, M. (2024). Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems (2nd Ed.). O'Reilly Media.
<https://dataintensive.net/>
5. Viswanathan, V. (2025). Enterprise LLM Implementation: Security, Governance, and Scalability. MIT Press. <https://mitpress.mit.edu/books/enterprise-llm>
6. Treveil, M., et al. (2023). Introducing MLOps: How to Scale Machine Learning in the Enterprise. O'Reilly Media. <https://www.oreilly.com/library/view/introducing-mlops/9781492083283/>



Additional Resources

Additional AI & Computational Resources

1. NVIDIA Technical Blog: Generative AI. (2026). Latest in TensorRT-LLM and Enterprise Deployment. <https://developer.nvidia.com/blog/category/generative-ai/>
2. Databricks Learning. (2026). The Lakehouse Architecture and Large Scale Data Engineering. <https://www.databricks.com/learn/training/lakehouse-fundamentals>
3. AWS Machine Learning Blog. (2026). Scaling AI Workflows on Distributed Infrastructure. <https://aws.amazon.com/blogs/machine-learning/>
4. LangChain Documentation. (2026). Building Enterprise-Grade RAG Pipelines. https://python.langchain.com/docs/use_cases/question_answering/
5. Apache Spark Documentation. (2026). Structured Streaming and MLlib for Scalable Engineering. <https://spark.apache.org/docs/latest/>

AI-Powered Assignments

Assignment 1: AI-Based Enterprise AI Adoption Strategy

- **Students prompt an AI tool:** *Design an AI adoption roadmap for a large enterprise undergoing digital transformation.*
- Students must:
 1. Analyze the AI-generated roadmap for completeness (data readiness, infrastructure, talent, governance)
 2. Identify gaps such as unrealistic timelines, lack of scalability planning, or missing risk factors
 3. Improve the strategy using standard enterprise AI lifecycle models and best practices
- **Assessment focuses on: strategic thinking, understanding of enterprise AI lifecycle, and validation of AI outputs.**

Assignment 2: AI-Assisted Data Pipeline Design

- **Students prompt an AI tool:** *Generate a scalable data pipeline for real-time fraud detection in banking systems.*
- Students must:
 1. Evaluate pipeline components (ingestion, storage, processing, orchestration)
 2. Identify bottlenecks, latency issues, or missing components (data governance, fault tolerance)
 3. Optimize the design using big data frameworks and streaming architectures
- **Assessment focuses on: data engineering skills, scalability analysis, and critical evaluation**

Assignment 3: AI-Driven Distributed Machine Learning Optimization

- Students prompt an AI tool: *Suggest a scalable machine learning solution for training models on large datasets.*
- Students must:
 1. Analyze distributed training strategies (parallelism, resource allocation, frameworks)
 2. Identify inefficiencies in model training, hyperparameter tuning, or dataset handling
- Improve the solution using advanced techniques like transfer learning and model



optimization.

- Assessment focuses on: ML scalability understanding, optimization skills, and AI validation.

Assignment 4: AI-Based Enterprise LLM Application Design

- Students prompt an AI tool: *Design an enterprise chatbot system using large language models for customer support.*
- Students must:
 1. Evaluate LLM architecture, deployment strategy, and scalability
 2. Identify risks related to hallucination, data privacy, and security
 3. Improve the system using LLMOps practices, fine-tuning, and governance strategies
- Assessment focuses on: LLM understanding, enterprise deployment, and ethical evaluation.

Assignment 5 AI-Assisted Infrastructure and Cost Optimization

- Students prompt an AI tool: *Optimize an enterprise AI infrastructure for performance and cost efficiency.*
- Students must:
 1. Analyze recommendations related to GPU usage, cloud scaling, and workload distribution
 2. Identify unrealistic assumptions or missing cost factors
 3. Propose improved strategies using hybrid architectures and resource optimization techniques
- Assessment **focuses on: infrastructure design, cost optimization, and critical thinking.**

Assignment 6: AI-Based Enterprise AI Governance & Security Evaluation

- Students prompt an AI tool: *Design a secure and compliant enterprise AI system handling sensitive data.*
- Students must:
 1. Evaluate security measures (access control, encryption, compliance standards)
 2. Identify gaps in governance, privacy policies, or regulatory compliance
 3. Improve the design using best practices for ethical AI, data protection, and governance frameworks
- Assessment focuses on: security awareness, governance understanding, and validation of AI outputs

Mini Project Component

Industry-Driven Projects:

Project 1 – Enterprise AI Adoption Framework for Digital Transformation

- **Problem:** Design an AI adoption roadmap for a large enterprise transitioning to AI-driven operations
- **Concepts Used:** AI lifecycle, digital transformation, scalability challenges, governance
- **Industrial Relevance:** Consulting firms, enterprise IT transformation (Accenture, Deloitte)

Project 2 – Scalable Data Pipeline for Real-Time Fraud Detection

- **Problem:** Build a scalable data pipeline to detect fraudulent financial transactions in real time



- **Concepts Used:** Data pipelines, streaming (Kafka), ETL/ELT, anomaly detection
- **Industrial Relevance:** Banking and fintech (Visa, Mastercard, PayPal)

Project 3 – Enterprise Data Lakehouse for Retail Analytics

- **Problem:** Develop a unified data architecture for integrating structured and unstructured retail data
- **Concepts Used:** Data lakes, warehouses, lakehouse architecture, data governance
- **Industrial Relevance:** Retail analytics platforms (Amazon, Walmart)

Project 4 – Distributed Machine Learning System for Predictive Maintenance

- **Problem:** Design a scalable ML system to predict equipment failures in industrial environments
- **Concepts Used:** Distributed ML, Spark, model optimization, IoT data processing
- **Industrial Relevance:** Smart manufacturing, Industry 4.0 (Siemens, GE)

Project 5 – Enterprise Chatbot Using Large Language Models (LLMs)

- **Problem:** Develop a scalable enterprise chatbot for customer support automation
- **Concepts Used:** LLM architecture, fine-tuning, prompt engineering, LLMOps
- **Industrial Relevance:** Customer service automation (Zendesk, Salesforce)

Project 6 – AI-Powered Cognitive Search System for Enterprises

- **Problem:** Build an intelligent search system for enterprise knowledge management
- **Concepts Used:** NLP, embeddings, vector search, LLM integration
- **Industrial Relevance:** Enterprise search platforms (Microsoft, Google)

Project 7 – Scalable Recommendation System for E-Commerce Platforms

- **Problem:** Develop a real-time recommendation engine for personalized product suggestions
- **Concepts Used:** Collaborative filtering, distributed systems, real-time analytics
- **Industrial Relevance:** E-commerce platforms (Amazon, Netflix)

Project 8 – AI-Based Demand Forecasting and Inventory Optimization

- **Problem:** Predict product demand and optimize inventory across supply chains
- **Concepts Used:** Time-series forecasting, ML models, data pipelines, optimization
- **Industrial Relevance:** Supply chain and logistics (Amazon, DHL)

Project 9 – Enterprise AI Infrastructure Design with GPU Clusters

- **Problem:** Design a scalable AI infrastructure for training large-scale models
- **Concepts Used:** GPU clusters, cloud AI platforms, distributed training, hybrid architecture
- **Industrial Relevance:** AI labs, cloud providers (NVIDIA, AWS, Google Cloud)



Blockchain Technologies

Course Code: 26DS4713

Semester: VII

Credits: 3

Programs Offered: B. Tech CSE (Data Science)

Course Type: Professional Elective (PE-IV)

L – T – P – J: 3 – 0 – 0 – 0

Prerequisite(s): Basic knowledge of programming, , computer networks, cryptography and Familiarity

Course Vision

To prepare learners with a strong foundation in blockchain technologies and distributed systems, enabling them to design, develop, and deploy secure, transparent, and scalable decentralized applications for real-world challenges. The course aims to foster innovation in areas such as digital finance, supply chain, healthcare, and governance, while emphasizing ethical practices, data integrity, and trust in next-generation digital ecosystems.

Course Objectives

1. To provide a strong foundation in the principles of blockchain technology and distributed ledger systems.
2. To enable students to understand and analyze cryptographic mechanisms such as hashing, digital signatures, and consensus algorithms.
3. To develop the ability to design and implement smart contracts and decentralized applications (DApps).
4. To explore various blockchain platforms and frameworks (e.g., Ethereum, Hyperledger) and their real-world applications.
5. To evaluate the security, scalability, and ethical challenges associated with blockchain systems and propose suitable solutions.

Course Outcomes (CO)

CO	Description	Blooms Level
CO1	Illustrate blockchain fundamentals and cryptographic principles	L2
CO2	Analyze consensus mechanisms and distributed ledger technologies	L4
CO3	Develop smart contracts and decentralized applications	L2
CO4	Evaluate blockchain platforms and enterprise use cases	L5
CO5	Design scalable blockchain solutions integrated with emerging technologies	L5

Broad Course Description

This course offers an in-depth exploration of blockchain technology as a transformative paradigm for building decentralized, secure, and transparent digital systems. It begins with the foundational concepts of distributed computing and peer-to-peer networks, progressing to the



core architecture of blockchain, including blocks, transactions, hashing, and data immutability. The course provides a detailed understanding of cryptographic principles such as hash functions, digital signatures, and public-key cryptography that ensure security and trust in blockchain networks. Students will examine various consensus mechanisms, including Proof of Work (PoW), Proof of Stake (PoS), and Practical Byzantine Fault Tolerance (PBFT), and analyze their trade-offs in terms of scalability, energy efficiency, and security. The course also introduces smart contracts and guides learners through the design and development of decentralized applications (DApps) using popular platforms like Ethereum and Hyperledger, with practical exposure to tools, frameworks, and development environments.

In addition, the course explores advanced topics such as tokenization, decentralized finance (DeFi), NFTs, and permissioned vs. permissionless blockchains. It emphasizes real-world applications across industries including finance, supply chain management, healthcare, digital identity, and e-governance. Key challenges such as scalability, interoperability, privacy, legal considerations, and ethical implications are critically examined. Through a combination of theoretical knowledge, hands-on exercises, case studies, and project work, students will gain the skills needed to design, evaluate, and implement blockchain-based solutions. By the end of the course, learners will be equipped to apply blockchain technologies effectively to solve complex engineering and business problems in modern digital ecosystems.

CO PO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
CO1	2	1			1						1	2	1
CO2	2	3			1						1	2	1
CO3	2	3	3	2	2						2	2	2
CO4	2	3	3	3	1						2	2	2
CO5	2	3	3	3	2						2	2	2
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													
3: Substantial (High)				2: Moderate (Medium)				1: Poor (Low)					

Course Modules

Module-1 Introduction to Blockchain & Cryptographic Foundations (9 hours)

Blockchain basics, evolution (Blockchain 1.0, 2.0, 3.0), distributed ledger technology (DLT), types of blockchains (public, private, consortium), cryptographic hash functions, Merkle trees, digital signatures, public/private key cryptography, wallets, transactions, blocks and chain structure.

Module-2 Consensus Mechanisms & Blockchain Architecture (9 Hours)

Consensus algorithms (PoW, PoS, DPoS, PBFT), mining concepts, forks, network architecture (nodes, peers), Byzantine Fault Tolerance, scalability issues, blockchain performance metrics, Layer 1 vs Layer 2 solutions, sharding, sidechains.

Module-3 Smart Contracts & Decentralized Applications (9 Hours)



Smart contracts fundamentals, Solidity programming basics, Ethereum Virtual Machine (EVM), gas mechanism, contract deployment, testing and debugging, decentralized applications (DApps), Web3.js, Truffle/Hardhat frameworks.

Module-4 Enterprise Blockchain & Use Cases (9 Hours)

Enterprise blockchain platforms (Hyperledger Fabric, Corda, Quorum), permissioned blockchains, identity management, interoperability, blockchain governance, supply chain management, healthcare, finance, digital identity, blockchain regulations and ethics.

Module-5 Scalable Blockchain, AI Integration (9 Hours)

Scalability in blockchain systems, off-chain computation, Layer-2 scaling (Lightning Network, Rollups), blockchain interoperability, integration with AI and IoT, blockchain analytics, GPU acceleration in blockchain (mining, cryptography, hashing), high-performance computing for blockchain, decentralized AI (DAI), edge blockchain systems.

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module-1: Introduction to Blockchain & Cryptographic Foundations

Students utilize AI-powered cryptography simulators and blockchain explorers to visualize hash functions, Merkle trees, and transaction flows. Tools like Python/Colab and OpenSSL libraries are used to implement hashing and digital signatures, while AI assistants help analyze security properties and detect vulnerabilities in cryptographic operations.

Module-2: Consensus Mechanisms & Blockchain Architecture

Students employ AI-based simulation platforms to model consensus algorithms (PoW, PoS, PBFT) and evaluate their efficiency, scalability, and fault tolerance. Machine learning tools are used to analyze blockchain performance metrics and predict network behaviour under varying loads, enabling deeper understanding of scalability solutions like sharding and Layer-2 techniques.

Module-3: Smart Contracts & Decentralized Applications

Students use AI-assisted coding tools to develop, test, and debug smart contracts in Solidity using environments like Remix, Truffle, or Hardhat. AI-based vulnerability detection tools (e.g., smart contract analyzers) are applied to identify security flaws, while Web3 frameworks are integrated to build intelligent decentralized applications (DApps).

Module-4: Enterprise Blockchain & Use Cases

Students leverage AI-driven analytics platforms to study real-world blockchain datasets in domains such as supply chain, healthcare, and finance. AI tools assist in anomaly detection, fraud analysis, and identity verification, while enterprise platforms like Hyperledger are explored with AI-supported workflow optimization and governance modelling.

Module-5: Scalable Blockchain & AI Integration

Students integrate AI/ML models with blockchain systems using Python frameworks to develop decentralized intelligent applications. GPU-based tools are used to accelerate cryptographic computations and blockchain analytics, while AI techniques are applied for predictive analysis, decentralized AI (DAI), and edge-based blockchain solutions in IoT environments.



Assessment Scheme

Criteria	Weightage of Marks
Lab Reports (experiments)/Project	10
Class Tests /assignments	10
Mid-Semester Examination	30
Hackathon/Technical Competition/ Paper presentation/Global/DLI certifications	10
End-Semester Examination	40
Total	100 marks

Recommended Tools

Primary Textbooks

1. Narayanan, A., Bonneau, J., Felten, E., Miller, A., & Goldfeder, S. (2024). Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction (2nd ed.). Princeton University Press. <https://press.princeton.edu/books/hardcover/9780691255011>
2. Antonopoulos, A. M., & Wood, G. (2024). Mastering Ethereum: Building Smart Contracts and DApps (2nd ed.). O'Reilly Media. <https://www.oreilly.com/library/view/mastering-ethereum/9781492045250/>
3. Wattenhofer, R. (2025). Blockchain Science: The Token Economy (3rd Ed.). Inverted Forest Publishing. <https://disco.ethz.ch/courses/blockchain/>.
4. Drescher, D. (2017). Blockchain basics: A non-technical introduction in 25 steps. Apress.
5. Bashir, I. (2023). Mastering blockchain (4th Ed.). Packt Publishing.

Additional Resources

Additional AI & Computational Resources

1. Ethereum.org Developers Portal. (2026). Documentation on Smart Contracts, EVM, and Layer 2 Scaling. URL: <https://ethereum.org/en/developers/>
2. Blockchain Council. (2026). Advanced Blockchain & AI Integration Certifications. URL: <https://www.blockchain-council.org/>
3. Hyperledger Foundation. (n.d.). Hyperledger Fabric & Sawtooth Documentation for Enterprise Blockchains. URL: <https://www.hyperledger.org/use/fabric>
4. Coursera / Princeton University. (2025). Bitcoin and Cryptocurrency Technologies (Updated Specialization).URL: <https://www.coursera.org/learn/cryptocurrency>
5. ConsenSys Academy. (2026). Web3, Solidity, and DApp Development Bootcamp. URL: <https://consensys.net/academy/>

AI-Powered Assignments

Assignment 1: AI-Based Blockchain Use Case Identification

- Students prompt an AI tool: *Suggest a suitable blockchain-based solution for improving transparency in supply chain management.*
- Students must:
 1. Analyze the feasibility of the AI-suggested solution (data flow, stakeholders, trust



- requirements).
- 2. Identify missing elements (scalability, cost, regulatory issues).
- 3. Refine the solution using standard blockchain architecture principles.
- Assessment focuses on problem analysis, system design, and validation of AI outputs

Assignment 2: AI-Assisted Cryptographic Mechanism Evaluation

- Students prompt an AI tool: *Students prompt an AI tool: Explain and compare hashing algorithms and digital signatures used in blockchain.*
- Students must:
 1. Verify correctness of AI explanations using cryptographic standards
 2. Identify weaknesses or oversimplifications in AI responses
 3. Demonstrate implementation using Python or relevant tools
- Assessment focuses on understanding cryptography and validation skills

Assignment 3: AI-Based Consensus Algorithm Analysis

- Students prompt an AI tool: *Recommend the best consensus algorithm for a decentralized financial system.*
- Students must:
 1. Evaluate AI recommendations (PoW, PoS, PBFT) based on scalability, security, and energy efficiency
 2. Identify trade-offs and limitations
 3. Propose an improved consensus strategy for the given scenario
- Assessment focuses on analytical thinking and system-level understanding.

Assignment 4: AI-Assisted Smart Contract Development and Auditing

- Students prompt an AI tool: *Generate a basic Solidity smart contract for a voting system.*
- Students must:
 1. Analyze correctness and functionality of the generated contract
 2. Identify vulnerabilities (reentrancy, overflow, access control issues)
 3. Improve and test the contract using standard tools
- Assessment focuses on coding skills, security awareness, and debugging.

Assignment 5 AI-Based Enterprise Blockchain Solution Design

- Students prompt an AI tool: *Design a blockchain-based healthcare record management system.*
- Students must:
 1. Evaluate AI-generated architecture (privacy, access control, interoperability)
 2. Identify practical challenges (compliance, scalability, cost)
 3. Propose enhancements using enterprise blockchain frameworks
- Assessment focuses on real-world application design and critical evaluation.

Assignment 6: AI-Driven Blockchain Scalability and Integration Study

- Students prompt an AI tool: *Suggest methods to scale blockchain systems and integrate with AI/IoT.*
- Students must:
 4. Analyze AI suggestions (Layer-2, sharding, off-chain computation)
 5. Validate technical feasibility and performance impact
 6. Propose an optimized scalable architecture for a real-world application
- Assessment focuses on innovation, integration skills, and performance analysis



Mini Project Component

Industry-Driven Projects:

Project 1 – Blockchain-Based Supply Chain Traceability System

- **Problem:** Develop a system to track products from manufacturer to consumer ensuring transparency.
- **Concepts Used:** Distributed ledger, smart contracts, hashing, transaction validation
- **Industrial Relevance:** Logistics and supply chain (Walmart, Maersk), anti-counterfeiting systems

Project 2 – Decentralized Voting System

- **Problem:** Build a secure and tamper-proof e-voting platform using blockchain
- **Concepts Used:** Smart contracts, consensus mechanisms, cryptographic signatures
- **Industrial Relevance:** E-governance, digital democracy, secure elections

Project 3 – Blockchain-Based Healthcare Record Management

- **Problem:** Design a secure system for storing and sharing patient medical records
- **Concepts Used:** Permissioned blockchain, encryption, identity management
- **Industrial Relevance:** Healthcare systems, patient data privacy, hospital networks

Project 4 – Cryptocurrency Wallet and Transaction System

- **Problem:** Develop a basic crypto wallet for secure transactions
- **Concepts Used:** Public/private keys, digital signatures, transaction validation
- **Industrial Relevance:** Fintech, digital payments, cryptocurrency platforms

Project 5 – Smart Contract-Based Insurance System

- **Problem:** Automate insurance claim processing using blockchain smart contracts
- **Concepts Used:** Solidity, contract deployment, event handling
- **Industrial Relevance:** Insurance industry, claim automation, fraud reduction



Explainable AI (XAI)

Course Code: 26DS4804

Semester: VIII

Program(s): B.Tech CSE (Data Science)

Course Type: Professional Elective (PE-V)

Credits:3

L – T – P – J: 3 – 0 – 0 – 0

Prerequisite(s): Deep Learning

Course Vision

To provide comprehensive understanding of explainable AI techniques, enabling students to build transparent, interpretable, and trustworthy machine learning systems that meet real-world regulatory, ethical, and user-facing requirements.

Course Objectives

1. Understand the need, importance, and landscape of explainable AI
2. Analyze types of explanations, interpretability methods, and their trade-offs
3. Apply feature attribution and model explanation techniques using SHAP, LIME, and Grad-CAM
4. Explore explainability methods for deep learning, NLP, and multimodal AI systems
5. Evaluate and communicate XAI results responsibly, avoiding common explanation pitfalls

Course Outcomes (CO)

CO	Description	Blooms Level
C01	Illustrate concepts, importance, and landscape of explainable AI	L2
C02	Analyze types of explanations, interpretability methods, and Shapley-based attribution	L3
C03	Apply advanced feature attribution and visual explanation techniques (PDP, LIME, Grad- CAM, XRAI)	L3
C04	Understand explainability methods for NLP, deep learning, and multimodal AI systems	L2
C05	Evaluate and communicate explainability results responsibly following best practices	L5

Broad Course Description

Explainable AI (XAI) is a 3-credit Core/Elective course for CSE-DS students in the 8th Semester. It addresses one of the most critical challenges in modern AI deployment — making machine



learning models transparent, interpretable, and trustworthy. The course covers the full XAI spectrum from foundational concepts and pre-modeling explainability through to advanced gradient-based methods, NLP explainability, and responsible communication of AI explanations.

The course is structured across five modules: Foundations of Explainable AI, Feature Attribution and Model Explanation Techniques, Advanced Explainability Techniques, Explainability in NLP and Multimodal Systems, and Evaluation, Communication, and Responsible Explainability. Students gain hands-on experience with industry-standard tools including SHAP, LIME, Captum, Grad-CAM, and Integrated Gradients, and develop the ability to design, implement, and critically evaluate XAI pipelines for real-world models in healthcare, finance, and legal domains.

CO-PO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
C01	2	1									1	2	
C02	2	3			2						1	3	
C03	2	3	3	2	3						1	3	2
C04	2	2			2						1	3	2
C05	2	3	2	2	1	1	1		2	1	2	3	1
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module 1: Foundations of Explainable AI (9 Hours)

Why Explainable AI, What is Explainable AI, Who needs explainability (practitioners, observers, end users), Challenges in explainability, Evaluating explainability, Applications of XAI and industry use cases, Interpretability vs explainability, Types of explanations, Pre-modeling explainability, Intrinsic vs post-hoc explainability, Local, cohort, and global explanations

Module 2: Feature Attribution and Model Explanation Techniques (9 Hours)

Topics: Feature attributions and surrogate models, Permutation feature importance, Implementation in scikit-learn, Shapley values and the SHAP framework, Local and global feature attribution visualization, Managed Shapley values, Explaining tree-based models, Tree Explainer, Kernel Explainer, Deep Explainer

Module 3: Advanced Explainability Techniques (9 Hours)

Topics: Partial Dependence Plots (PDP), Individual Conditional Expectation (ICE), Accumulated Local Effects (ALE), Integrated Gradients (IG) and baseline selection, Gradient accumulation,



XRAI method and implementation, Grad-CAM, Guided Grad-CAM, Guided backpropagation, LIME and its implementation

Module 4: Explainability in NLP and Multimodal Systems (9 Hours)

Topics: Explainability in text models, Tokenization and embeddings for XAI, LIME for text, Gradient-based attribution for NLP, Layer Integrated Gradients, Layer-wise Relevance Propagation (LRP), Attention-based explanations, Language interpretability tools (Captum, BertViz), Explainability in time-series and multimodal data, Bias detection in language models

Module 5: Evaluation, Communication, and Responsible Explainability (9 Hours)

Topics: Evaluation of explainability techniques (theoretical and empirical), Fidelity, stability, and comprehensibility metrics for XAI, Presenting explanations effectively to different audiences, Clarifying what, how, and why models behave, Accurate representation of explanations, Building on user understanding, Common pitfalls: assuming causality, overfitting intent, over-explanation, Regulatory frameworks for AI explainability (EU AI Act, GDPR), Best practices in XAI

Case Study: End-to-end XAI pipeline — training a classifier on a healthcare or financial dataset, applying SHAP global and local explanations, Grad-CAM for CNN-based predictions, LIME for text model interpretation, and presenting a comprehensive explainability report addressing regulatory compliance requirements

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module 1: Students use AI tools to generate explanations of model decisions on sample datasets and compare intrinsic vs post-hoc explanation styles. AI-generated explanations are critically evaluated for clarity, audience appropriateness, and whether they accurately reflect the underlying model behaviour without overstating certainty.

Module 2: AI-assisted SHAP analysis tools are used to generate global and local feature attribution plots for tree-based and neural models. Students critically evaluate AI-generated SHAP waterfall and summary plots for feature ranking correctness, interaction effects, and whether Shapley value interpretations are contextually valid for the given prediction task.

Module 3: Students use AI tools to generate Grad-CAM heatmaps, LIME explanations, and Integrated Gradient attribution maps for CNN and tabular models. AI-generated visual explanations are critically assessed for faithfulness to the model, baseline sensitivity, and whether gradient accumulation artifacts may mislead the interpretation of feature importance.

Module 4: AI-assisted NLP explainability tools (Captum, BertViz) are used to generate token-level attribution and attention-based explanations for transformer models. Students critically evaluate whether attention weights reflect true model reasoning, identify bias patterns in language model explanations, and assess the reliability of LRP attributions for multimodal inputs.

Module 5: Students use AI tools to generate XAI evaluation reports and explanation narratives for stakeholder communication. AI-generated reports are critically assessed against Module 5



best practices — checking for causality assumptions, overfitting of intent, over-explanation, and alignment with EU AI Act and GDPR transparency requirements.

Assessment Scheme

Criteria	Weightage of Marks
Class Tests	10
Mid-Semester Examination	30
Mini Project	10
Assignments	10
End-Semester Examination	40
Total	100 marks

Recommended Tools

Primary Textbooks

1. David Pitman, Michael Munn, Explainable AI for Practitioners: Designing and Implementing Explainable ML Solutions, O'Reilly Media, 2022
2. Christoph Molnar, Interpretable Machine Learning: A Guide for Making Black Box Models Explainable, 2nd Edition, 2022 – christophm.github.io/interpretable-ml-book
3. Andreas Müller, Sarah Guido, Introduction to Machine Learning with Python, O'Reilly Media (SHAP & feature importance chapters)
4. LIME Documentation – github.com/marcotcr/lime
5. Captum Documentation (PyTorch Explainability) – captum.ai
6. SHAP Documentation – shap.readthedocs.io

Additional Resources

1. NPTEL: Responsible AI – nptel.ac.in
2. Christoph Molnar, Interpretable ML Book (free online) christophm.github.io/interpretable-ml-book
3. Google PAIR Explainability Resources – pair.withgoogle.com
4. EU AI Act and GDPR Explainability Guidelines – digital-strategy.ec.europa.eu
5. Papers With Code – Explainability Benchmarks paperswithcode.com/task/explainability

Mini Project Component

The following projects are suggested based on course outcomes. Students may attempt one project individually or in pairs.

Project 1 – SHAP-Based Model Explanation Pipeline for Tabular Data

Problem: Train a gradient boosting classifier (XGBoost or LightGBM) on a healthcare or finance dataset and build a complete SHAP explanation pipeline with global summary plots, local waterfall plots, and dependence plots for the top 5 features

Concepts Used: SHAP TreeExplainer, global and local attribution, feature interaction,



permutation importance, model debugging

Industrial Relevance: Credit scoring explainability, medical diagnosis transparency, regulatory model auditing

Project 2 – Visual Explainability for CNN Image Classifier

Problem: Train or fine-tune a CNN for image classification and implement Grad-CAM, Guided Grad-CAM, and Integrated Gradients to produce and compare visual explanation maps; evaluate faithfulness using insertion/deletion metrics

Concepts Used: Grad-CAM, Guided backpropagation, Integrated Gradients, XRAI, baseline selection, faithfulness evaluation

Industrial Relevance: Medical imaging AI auditing, autonomous vehicle decision explanation, manufacturing defect inspection

Project 3 – NLP Model Explainability and Bias Audit

Problem: Apply LIME for text, Layer Integrated Gradients, and attention visualization to a fine-tuned BERT-based sentiment or NER model; identify and report potential bias patterns in token-level attributions across demographic groups

Concepts Used: LIME for text, Captum Layer IG, BertViz attention, LRP, bias detection, responsible XAI communication

Industrial Relevance: AI fairness auditing, legal NLP compliance, social media content moderation transparency

AI-Powered Assignments

Assignment 1: AI-Generated SHAP Explanation Evaluation

Students prompt an AI tool: Explain the predictions of a loan approval model using SHAP. Generate global feature importance, a local explanation for one rejected applicant, and identify the top 3 features driving the rejection decision.

Students must:

- Evaluate whether the AI-generated SHAP summary plot correctly represents global feature importance and whether the feature ranking aligns with domain knowledge about loan approval decisions
- Identify potential misinterpretations: check if the local waterfall plot correctly reflects filter context, interaction effects, and base value accuracy
- Assess whether the explanation is suitable for a non-technical audience (applicant) vs a technical auditor, and rewrite the narrative accordingly

Assessment focuses on: SHAP interpretation correctness, audience-aware explanation quality, and critical evaluation of AI-generated attribution narratives

Assignment 2: Grad-CAM Faithfulness and Pitfall Analysis

Students prompt an AI tool: Apply Grad-CAM to a CNN trained on chest X-ray classification to explain which regions the model focuses on when predicting pneumonia vs normal cases.

Students must:

- Critically evaluate whether the AI-generated Grad-CAM heatmaps highlight clinically relevant regions or whether they reflect dataset artifacts (e.g., scanner borders, text annotations)
- Identify the causality assumption pitfall: determine whether the highlighted region causes the prediction or merely correlates with it in the training data
- Improve the explanation by applying Guided Grad-CAM and comparing with Integrated Gradients; report faithfulness differences using insertion/deletion scores



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Assessment focuses on: Grad-CAM faithfulness reasoning, causality pitfall identification, and critical comparison of visual explanation methods

Assignment 3: Responsible XAI Communication Critique

Students prompt an AI tool: Generate a stakeholder explainability report for a recidivism prediction model, explaining model behaviour to a judge, a defendant, and an AI auditor.

Students must:

- Evaluate whether the AI-generated report correctly tailors explanation depth, terminology, and format for each of the three distinct audiences
- Identify over-explanation, false causality claims, or missing uncertainty quantification in the AI-generated narratives
- Rewrite the report sections applying Module 5 best practices: accurate representation, audience calibration, regulatory alignment (EU AI Act), and explicit acknowledgment of model limitations

Assessment focuses on: Responsible XAI communication, regulatory alignment, and critical evaluation of AI-generated explanation reports for fairness and accuracy



Autonomous Systems

Course Code: 26DS4805

Semester: VIII

Programs Offered: B. Tech CSE (Data Science)

Course Type: Professional Elective (PE-V)

Credits: 3

L – T – P – J: 3-0-0-0

Prerequisites: Python Programming, Basic Machine Learning, Computer Vision Fundamentals, Probability and Statistics, Linear Algebra

Course Vision

To equip students with the knowledge, practical skills, and ethical reasoning required to design, implement, and evaluate intelligent autonomous systems that perceive, reason, and act in dynamic real-world environments, leveraging open-source AI and robotics frameworks alongside NVIDIA DLI-aligned GPU-accelerated platforms for autonomous vehicle and robotic perception pipelines.

Course Objectives

1. Understand the architecture, sensing modalities, and decision frameworks of autonomous systems
2. Apply computer vision and sensor fusion techniques for environment perception and object detection
3. Design simultaneous localization and mapping (SLAM), path planning, and motion control systems
4. Build reinforcement learning and behaviour planning pipelines for autonomous decision-making
5. Apply GPU-accelerated perception and NVIDIA DLI-aligned frameworks for real-time autonomous system deployment

Course Outcomes (CO)

CO	Description	Blooms Level
CO1	Illustrate autonomous system architectures, sensing modalities, and decision-making frameworks across land, air, and underwater domains	L2
CO2	Apply computer vision, LiDAR processing, and sensor fusion techniques for environment perception and object detection	L3



C03	Design SLAM, path planning, and motion control systems for autonomous navigation in structured and unstructured environments	L6
C04	Build reinforcement learning and behaviour planning pipelines for autonomous decision-making in dynamic environments	L6
C05	Apply GPU-accelerated perception and NVIDIA DLI-aligned tools for real-time autonomous vehicle and robotic inference deployment	L3

Broad Course Description

This course provides a comprehensive foundation in the design and implementation of autonomous systems, spanning autonomous vehicles, mobile robots, drones, and underwater vehicles. Students learn the full autonomous systems stack from sensing and perception through to planning, control, and deployment in dynamic real-world environments. Core topics include computer vision for object detection and semantic segmentation, LiDAR- based 3D perception, sensor fusion using Kalman filters and deep learning, simultaneous localization and mapping (SLAM), path planning algorithms, and reinforcement learning for autonomous decision-making. Students build hands-on skills using open-source frameworks including PyTorch, OpenCV, Open3D, ROS 2, and Gymnasium. The course emphasises functional safety and responsible deployment of autonomous systems, addressing safety standards (ISO 26262, DO-178C), edge case handling, adversarial robustness, and the ethical implications of autonomous decision-making in safety-critical scenarios including autonomous vehicles, medical robots, and military drones. The final module is aligned with the NVIDIA Deep Learning Institute (DLI), enabling students to implement GPU-accelerated perception pipelines using NVIDIA CUDA, TensorRT, DeepStream, and TAO Toolkit, and to deploy real-time autonomous inference systems using NVIDIA DriveAGX and Triton Inference Server for production-grade autonomous vehicle and robotic applications

CO-PO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
C01	2	1			1						1	2	2
C02	2	3			3						1	3	2
C03	2	3	3	2	3						2	3	3
C04	2	3	3	2	3						2	3	3
C05	2	3	3	3	3						2	3	3
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													



Course Modules

Module 1: Foundations of Autonomous Systems and Perception (9 Hours)

Autonomous system types (vehicles, drones, robots), System architectures (sense-plan-act, reactive, hybrid), Sensors (camera, LiDAR, radar, IMU, GPS), Sensor characteristics and failure modes, Camera calibration fundamentals, ROS 2 basics for robotic systems, Key datasets (KITTI, nuScenes, Waymo, AirSim), Applications and AI integration, Sensor reliability, dataset bias, and ethical safety concerns

Module 2: Environment Perception and Sensor Fusion (9 Hours)

Object detection (YOLO, SSD, Faster R-CNN), Segmentation models (FCN, U-Net, Mask R-CNN), LiDAR processing and 3D detection (PointNet, PointPillars), Depth estimation techniques, Sensor fusion (early, late, deep fusion), State estimation (Kalman, EKF), Multi-object tracking (SORT, DeepSORT), Domain adaptation for perception, Applications and AI integration, Edge-case failures, adversarial risk, and liability in perception systems

Module 3: Localisation, Mapping, and Path Planning (9 Hours)

SLAM (EKF, graph-based, visual SLAM), Localisation (particle filters, MCL), HD and semantic mapping, Path planning (A*, Dijkstra, RRT, RRT*), Trajectory planning and smoothing, Dynamic obstacle avoidance (DWA, TEB), ROS 2 Nav2 navigation stack, Planning under uncertainty, Applications and AI integration, Localisation failures, map staleness, and safety/fairness in navigation systems

Module 4: Behaviour Planning and Autonomous Decision-Making (9 Hours)

Autonomous decision-making (FSMs, behaviour trees, rule-based systems), Reinforcement learning (PPO, SAC, safe RL), Imitation and end-to-end learning, Simulation-to-real transfer, Trajectory prediction of agents, V2X and cooperative driving, Safety validation (scenario testing, ISO 26262), Applications and AI integration, Ethics, accountability, adversarial risk, and fairness in autonomous decision systems

Module 5: GPU-Accelerated Autonomous Perception and NVIDIA DLI-Aligned Deployment (9 Hours)

GPU-accelerated perception (CUDA, TensorRT), NVIDIA TAO Toolkit for model training, Real-time multi-sensor pipelines (DeepStream), Edge optimization (quantization, pruning, FP16/INT8), Deployment on Drive AGX and Jetson, Triton Inference Server, CARLA–Omniverse simulation and synthetic data, Functional safety and fail-safe systems, NVIDIA DLI hands-on implementation, Production autonomous perception and robotics deployment pipelines

Assessment Scheme based on the course

Criteria	Weightage of Marks
Class Tests	10
Mid-Semester Examination	30



Mini Project	10
Assignments	10
End-Semester Examination	40
Total	100 marks

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

- Module 1: Students set up ROS 2 sensor data pipelines and explore autonomous system datasets (KITTI, nuScenes). AI-assisted dataset analysis tools are used to understand sensor characteristics, data distributions, and demographic representation in training data.
- Module 2: Students build object detection and sensor fusion pipelines using PyTorch YOLOv8 and Open3D. AI-generated perception results are critically evaluated for edge case failures, adversarial robustness, and demographic bias in detection performance.
- Module 3: Students implement SLAM and path planning using ROS 2 Nav2 and open-source SLAM tools. AI-generated navigation plans are evaluated for safety in dynamic environments, fairness in path decisions affecting pedestrians, and handling of localisation uncertainty.
- Module 4: Students train RL agents for autonomous decision-making using CARLA simulator and Stable-Baselines3. AI-driven autonomous decisions are critically evaluated for safety guarantees, adversarial robustness, and ethical alignment with autonomous vehicle policy standards.
- Module 5: Students deploy real-time autonomous perception pipelines using NVIDIA TAO, DeepStream, and Triton. NVIDIA DLI-aligned labs are used to benchmark GPU-accelerated autonomous AI performance and validate production-grade deployment safety.

Authentic Intelligence Reflection (AI² Pedagogy):

Module 1: Understanding geographic and demographic bias in autonomous system datasets and the ethical responsibility of dataset curation for safety-critical systems. Students evaluate the risks of deploying autonomous systems trained on non-representative data.

Module 2: Recognising perception failure modes under adverse conditions and adversarial attacks. Students assess the liability and accountability frameworks when AI perception errors cause accidents or harm.

Module 3: Evaluating fairness and safety in path planning systems that share environments with vulnerable road users. Students reflect on the responsibility of autonomous systems to prioritise pedestrian and cyclist safety over efficiency.

Module 4: Understanding the ethical dimensions of autonomous decision-making in life-critical scenarios including the trolley problem, accountability for autonomous harm, and the limits of



RL-based safety guarantees in open-world environments.

Module 5: Evaluating the governance and functional safety requirements for deploying GPU-accelerated AI in safety-critical autonomous systems. Students reflect on certification pathways, fail-safe design, and responsible deployment practices for production autonomous vehicles and robots.

List of Experiments

No.	Experiment	Module Link
1	ROS 2 sensor data pipeline: setting up camera, LiDAR, and IMU data streams	Module I
2	Camera calibration using Zhang method and OpenCV for autonomous system setup	Module I
3	Object detection using YOLOv8 on autonomous driving dataset (KITTI or nuScenes)	Module II
4	Semantic segmentation using U-Net for road scene understanding	Module II
5	3D object detection from LiDAR point clouds using PointPillars and Open3D	Module II
6	Sensor fusion using Extended Kalman Filter for camera and LiDAR state estimation	Module II
7	Multi-object tracking using DeepSORT on autonomous driving video sequences	Module II
8	Visual SLAM implementation using ORB-SLAM3 concepts in ROS 2	Module III
9	Path planning using A* and RRT* in a 2D simulation environment	Module III
10	Autonomous navigation using ROS 2 Nav2 with global and local cost maps	Module III
11	Reinforcement learning for autonomous vehicle lane-following using CARLA and Stable-Baselines3	Module IV
12	Behaviour tree implementation for autonomous agent decision-making using py_trees	Module IV
13	GPU-accelerated object detection using NVIDIA TAO Toolkit on autonomous driving data	Module V
14	Real-time multi-camera perception pipeline deployment using NVIDIA DeepStream	Module V
15	Synthetic data generation for autonomous AI using CARLA simulator and Omniverse concepts	Module V



16	Deployment of autonomous perception models using NVIDIA Triton Inference Server with latency benchmarking	Module V
----	---	----------

Recommended Tools

- Python 3.10+
- PyTorch (YOLOv8, U-Net, PointPillars, RL models)
- OpenCV (camera calibration, computer vision)
- Open3D (LiDAR point cloud processing)
- NVIDIA TAO Toolkit (transfer learning for perception)
- NVIDIA DeepStream (real-time multi-sensor video analytics)
- NVIDIA DriveAGX / Jetson (autonomous edge deployment)
- NVIDIA Omniverse / CARLA-Omniverse (synthetic data generation)
- NVIDIA Triton Inference Server
- HPC Lab (GPU computing environment)

Primary Textbooks

1. NVIDIA Corporation, NVIDIA Deep Learning Institute (DLI) Course Materials and Documentation, Latest Edition (Online Resource).
2. Sebastian Thrun, Wolfram Burgard, Dieter Fox, Probabilistic Robotics, MIT Press, 1st Edition, 2005.
3. Bruno Siciliano, Lorenzo Sciavicco, Luigi Villani, Giuseppe Oriolo, Robotics: Modelling, Planning and Control, Springer, 2009.
4. Richard S. Sutton, Andrew G. Barto, Reinforcement Learning: An Introduction, MIT Press, 2nd Edition, 2018.
5. Ian Goodfellow, Yoshua Bengio, Aaron Courville, Deep Learning, MIT Press, 1st Edition, 2016.
6. Aurélien Géron, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, O'Reilly Media, 3rd Edition, 2022.

Additional Resources

- NVIDIA TAO Toolkit Documentation – developer.nvidia.com/tao-toolkit
- NVIDIA DeepStream Documentation – developer.nvidia.com/deepstream-sdk
- NVIDIA DriveAGX Documentation – developer.nvidia.com/drive
- ROS 2 Documentation and Nav2 – docs.ros.org and navigation.ros.org
- CARLA Autonomous Driving Simulator – carla.org

AI-Powered Assignments

Assignment 1: AI-Based Autonomous System Architecture Analysis

Students prompt an AI tool: Design an autonomous system architecture for a delivery drone and recommend the optimal sensing modality combination for urban navigation.

Students must:



- Evaluate the correctness of AI-recommended sensor suite (camera, LiDAR, radar, GPS) against urban flight constraints
- Identify missing failure mode considerations (GPS denial, sensor occlusion, adverse weather)
- Improve the architecture design using ROS 2 sensor pipeline modelling and failure mode analysis

Assessment focuses on autonomous system architecture reasoning, sensor selection trade-offs, and failure mode awareness.

Assignment 2: AI-Assisted Perception Pipeline Design and Evaluation

Students prompt an AI tool: Design an object detection and sensor fusion pipeline for an autonomous vehicle and recommend strategies for handling edge cases.

Students must:

- Analyze the correctness of AI-recommended perception architecture (camera-LiDAR fusion, model selection)
- Identify edge case failures (occluded pedestrians, night-time performance, adversarial patches)
- Improve the pipeline using YOLOv8 and EKF-based fusion with evaluation on KITTI or nuScenes benchmarks

Assessment focuses on perception pipeline design, edge case analysis, and adversarial robustness awareness.

Assignment 3: AI-Based SLAM and Path Planning System Design

Students prompt an AI tool: Design a SLAM and path planning system for an indoor mobile robot operating in a dynamic environment.

Students must:

- Evaluate the correctness of the AI-recommended SLAM approach (EKF-SLAM vs. graph SLAM vs. visual SLAM) for the given environment
- Identify localisation failure scenarios (feature-sparse areas, dynamic object interference)
- Improve the design using ROS 2 Nav2 and evaluate path planning performance in a simulated dynamic environment

Assessment focuses on SLAM system design, planning algorithm selection, and dynamic environment handling.

Assignment 4: AI-Assisted Reinforcement Learning Policy Design for Autonomous Control

Students prompt an AI tool: Design an RL policy for autonomous vehicle highway lane changing and evaluate its safety guarantees. Students must:

- Analyze the AI-recommended RL algorithm (PPO vs. SAC vs. model-based RL) for the highway scenario
- Identify safety constraint violations in the proposed reward function and training environment design
- Improve the policy using CARLA simulator and Stable-Baselines3 with constrained RL evaluation

Assessment focuses on RL policy design for autonomous systems, safety constraint reasoning,



Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112
and simulation evaluation.

Assignment 5: AI-Based Autonomous System Safety and Ethics Analysis

Students prompt an AI tool: Analyze the safety and ethical implications of deploying an autonomous vehicle decision-making system in a mixed urban traffic environment.

Students must:

- Evaluate AI-generated safety analysis against ISO 26262 ASIL levels and scenario-based testing standards
- Identify ethical issues such as demographic bias in pedestrian detection, trolley problem scenarios, and accountability gaps
- Improve the safety analysis with a structured hazard log and responsible AI deployment framework

Assessment focuses on functional safety awareness, autonomous system ethics, and responsible deployment analysis.

Assignment 6: AI-Driven GPU-Accelerated Autonomous Perception Pipeline Evaluation

Students prompt an AI tool: Design and evaluate a GPU-accelerated real-time autonomous perception pipeline for an autonomous vehicle using NVIDIA tools.

Students must:

- Analyze all pipeline stages (TAO fine-tuning → TensorRT optimisation → DeepStream real-time inference → Triton serving)
- Identify gaps in real-time latency guarantees, fail-safe mechanisms, and edge deployment feasibility on DriveAGX or Jetson
- Improve the solution with NVIDIA DLI-aligned benchmarking and safety validation against autonomous driving standards

Assessment focuses on GPU-accelerated autonomous AI pipeline design, NVIDIA toolchain integration, and functional safety-aware deployment reasoning.

Mini Project Component

Industry-Driven Projects:

Project 1 – Autonomous Vehicle Object Detection and Tracking System

Problem: Build a real-time object detection and multi-object tracking pipeline for autonomous driving using camera and LiDAR data.

Concepts Used: YOLOv8, DeepSORT multi-object tracking, Open3D LiDAR processing, sensor fusion, KITTI/nuScenes dataset.

Industrial Relevance: Autonomous vehicle perception stacks, ADAS systems, and smart intersection monitoring.

Project 2 – LiDAR-Based 3D Perception System for Autonomous Navigation

Problem: Develop a 3D object detection and environment mapping system using LiDAR point cloud data.

Concepts Used: PointPillars 3D detection, Open3D point cloud processing, voxelisation, 3D bounding box evaluation on KITTI.



DAYANANDA SAGAR
UNIVERSITY



SCHOOL OF
ENGINEERING

Devarakaggalahalli, Harohalli, Kanakapura Road, Bengaluru South District, Karnataka – 562 112

Industrial Relevance: Autonomous vehicle LiDAR stacks, autonomous forklift navigation, and outdoor robot mapping.

Project 3 – Visual SLAM and Autonomous Navigation System

Problem: Implement a visual SLAM system integrated with autonomous navigation using ROS 2 and a simulated environment.

Concepts Used: ORB-SLAM3 concepts, ROS 2 Nav2, A* and RRT* path planning, dynamic cost map management.

Industrial Relevance: Indoor mobile robot navigation, autonomous warehouse systems, and drone indoor localisation.

Project 4 – Reinforcement Learning for Autonomous Driving in CARLA

Problem: Train an RL agent for autonomous highway driving using the CARLA simulator with lane-following and obstacle avoidance.

Concepts Used: Stable-Baselines3 PPO/SAC, CARLA environment integration, reward shaping, sim-to-real transfer concepts.

Industrial Relevance: Autonomous vehicle behaviour planning, RL-based adaptive control, and autonomous driving policy research.



Supply Chain AI

Course Code: 26DS4806

Semester: VIII

Programs: B.Tech CSE(Data Science)

Course Type: Professional Elective (PE-V)

Credits:3

L – T – P – J: 3 – 0 – 0 – 0

Prerequisite: Basics of Data Science.

Course Vision

To equip students with the knowledge and skills to apply Artificial Intelligence techniques for optimizing and transforming modern supply chain systems, enabling data-driven decision-making and intelligent automation.

Course Objectives

1. Understand supply chain fundamentals, components, and key challenges
2. Apply AI and machine learning techniques for demand forecasting in supply chains
3. Analyze inventory optimization and warehouse automation methods using AI
4. Explore intelligent logistics, route optimization, and transportation analytics
5. Design AI-driven end-to-end supply chain solutions using real-world case studies

Course Outcomes (CO)

CO	Description	Blooms Level
C01	Explain supply chain concepts and AI application areas	L2
C02	Apply machine learning models for demand forecasting	L3
C03	Analyze inventory and warehouse optimization techniques using AI	L4
C04	Design intelligent logistics and transportation systems	L6
C05	Develop AI-based end-to-end supply chain solutions	L6

Broad Course Description

Supply Chain AI is a 3-credit Core course for CSE - DS students in the 6th Semester. It bridges data science and operations management, focusing on how AI and machine learning can transform supply chain processes - from demand forecasting and inventory control to route optimization, digital twins, and resilient supply chain design.



The course covers five modules: Introduction to Supply Chain and AI, Demand Forecasting Using AI, Inventory and Warehouse Optimization, Logistics and Transportation Analytics, and AI-Driven Supply Chain Systems. Students develop practical skills in applying Python, Pandas, and Scikit-learn to real supply chain datasets, and explore case studies from Amazon, Walmart, and other industry leaders to understand deployment at scale.

CO-PO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
C01	2	1			1					1	1	2	2
C02	2	3			2					1	1	3	2
C03	2	3			2					2	2	3	2
C04	2	3	3	2	3					2	2	3	3
C05	2	3	3	3	3					2	2	3	3
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module 1: Introduction to Supply Chain and AI (9 Hours)

Supply chain fundamentals, Components of supply chain (procurement, production, distribution), Challenges in traditional supply chains, Introduction to AI in supply chain, Role of data in supply chain management, Digital supply chain transformation, Overview of predictive and prescriptive analytics

Module 2: Demand Forecasting Using AI (9 Hours)

Demand forecasting concepts, Time series analysis, Regression models for forecasting, Machine learning models for demand prediction, Seasonal and trend analysis, Deep learning for demand forecasting, Forecast evaluation metrics (MAE, RMSE, MAPE)

Module 3: Inventory and Warehouse Optimization (9 Hours)

Inventory management techniques (EOQ, safety stock, reorder point), AI-based inventory optimization, Warehouse operations and layout planning, Automated storage and retrieval systems, Reinforcement learning in inventory control, Cost optimization strategies

Module 4: Logistics and Transportation Analytics (9 Hours)

Transportation models and optimization, Route optimization algorithms, Vehicle Routing Problem (VRP) and variants, AI in logistics operations, Real-time tracking systems, Last-mile delivery optimization, Use of IoT in supply chain logistics



Module 5: AI-Driven Supply Chain Systems

(9 Hours)

End-to-end supply chain analytics, Digital twins in supply chain management, Risk management using AI, Supply chain resilience and disruption handling, Blockchain in supply chain, Intelligent decision support systems, Case studies of AI in supply chain (Amazon, Walmart)

Case Study: Analysis of AI-driven supply chain transformation at a large-scale retailer — covering demand forecasting pipeline design, inventory optimization, last-mile delivery routing, and resilience planning using digital twin technology

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module 1: AI tools are used to explore digital supply chain dashboards and simulate disruption scenarios. Students critically evaluate AI-generated supply chain diagnostic reports for completeness, data quality assumptions, and alignment with real operational constraints.

Module 2: AI-assisted forecasting tools (Prophet, LSTM-based models) are used to generate demand predictions on real datasets. Students critically evaluate forecast accuracy, seasonal adjustment quality, and model selection appropriateness for the given supply chain context.

Module 3: AI-powered inventory optimization tools and reinforcement learning simulators are used to explore dynamic reorder strategies. Students evaluate AI-generated inventory policies against EOQ baselines and assess cost-availability trade-offs under uncertainty.

Module 4: AI-based route optimization tools (VRP solvers, Google OR-Tools) are used to generate delivery route plans. Students critically assess solution quality, computational feasibility for large fleets, and IoT data integration for real-time logistics decisions.

Module 5: Students use digital twin simulation platforms and AI decision support tools to model end-to-end supply chain resilience. AI-generated risk mitigation strategies and blockchain integration recommendations are critically evaluated against real industry case studies.

Assessment Scheme

Criteria	Weightage of Marks
Class Tests / Assignments	10
Mid-Semester Examination	30
DLI Certifications /Hackathon / Technical Competition /Mini Project	10
Paper Presentation	10
End-Semester Examination	40
Total	100 marks



Recommended Tools

Primary Textbooks

1. Nicolas Vandeput, Data Science for Supply Chain Forecasting, De Gruyter, 2nd Edition, 2021
2. Yossi Sheffi, The Power of Resilience: How the Best Companies Manage the Unexpected, MIT Press
3. Nada R. Sanders, Supply Chain Management: A Global Perspective, Wiley
4. Google OR-Tools Documentation – developers.google.com/optimization
5. Pandas & Scikit-learn Documentation – pandas.pydata.org | scikit-learn.org
6. Hokey Min, Artificial Intelligence in Supply Chain Management: Theory and Applications, FT Press

Additional AI & Computational Resources

1. NPTEL: Supply Chain Management – nptel.ac.in
2. MIT Open Courseware: Logistics and Supply Chain Management – ocw.mit.edu
3. Towards Data Science: Supply Chain AI articles – towardsdatascience.com
4. Kaggle Supply Chain Datasets – kaggle.com/datasets
5. Google Colab / Jupyter Notebooks – colab.research.google.com

Mini Project Component

The following projects are suggested based on course outcomes. Students may attempt one project individually or in pairs.

Project 1 – AI-Based Demand Forecasting System

Problem: Build a demand forecasting model using historical sales data with seasonal decomposition and ML/DL models

Concepts Used: Time series analysis, regression, LSTM, forecast evaluation (MAE, RMSE, MAPE)

Industrial Relevance: Retail inventory planning, FMCG supply chain, e-commerce demand management

Project 2 – Intelligent Route Optimization for Last-Mile Delivery

Problem: Solve a Vehicle Routing Problem (VRP) for a delivery fleet using OR-Tools or a heuristic algorithm

Concepts Used: VRP formulation, optimization algorithms, IoT tracking, cost minimization

Industrial Relevance: E-commerce last-mile delivery, food delivery optimization, fleet management

Project 3 – Supply Chain Resilience Dashboard using Digital Twin Concepts

Problem: Simulate a supply chain disruption scenario and design an AI-based risk mitigation strategy

Concepts Used: Digital twins, risk scoring, decision support systems, data visualization

Industrial Relevance: Manufacturing disruption planning, pharmaceutical supply resilience, global logistics risk management



AI-Powered Assignments

Assignment 1: AI-Assisted Demand Forecasting Model Evaluation

Students prompt an AI tool: Forecast monthly demand for a retail product using this historical sales dataset and recommend an appropriate ML model.

Students must:

- Evaluate whether the chosen model accounts for seasonality, trend, and irregular demand spikes
- Identify data leakage risks, improper train-test splits, or inappropriate evaluation metrics
- Improve the forecast by applying proper time series cross-validation and comparing multiple models

Assessment focuses on: forecasting methodology, model selection reasoning, and critical evaluation of AI-generated outputs

Assignment 2: AI-Based Inventory Policy Optimization Analysis

Students prompt an AI tool: Determine optimal reorder points and safety stock levels for a warehouse managing 50 SKUs under uncertain demand.

Students must:

- Evaluate whether the AI-suggested EOQ and safety stock calculations are correct for the given demand variability
- Identify assumptions that oversimplify real inventory dynamics (lead time variability, stockout costs)
- Improve the policy using service-level constraints and simulation-based validation

Assessment focuses on: inventory management understanding, optimization reasoning, and validation of AI-generated policies

Assignment 3: AI-Generated Route Optimization Critique

Students prompt an AI tool: Generate optimal delivery routes for a fleet of 10 vehicles covering 50 delivery locations, minimizing total distance.

Students must:

- Assess whether the solution respects vehicle capacity, time window, and real road constraints
- Identify unrealistic assumptions such as uniform travel speed or ignoring traffic patterns
- Improve the solution by incorporating dynamic constraints and comparing against baseline heuristics

Assessment focuses on: logistics optimization understanding, constraint modeling, and critical evaluation of AI-generated routing solutions



Enterprise Deployments

Course Code: 26DS4807

Semester: VIII

Program: B. Tech CSE (Data Science)

Course Type: Professional Elective (PE-V)

Credits: 3

L – T – P – J: 3-0-0-0

Prerequisites: DevOps Fundamentals

Course Vision

To equip students with the knowledge, practical skills, and production-level competencies required to design, deploy, manage, and secure enterprise-scale cloud and hybrid infrastructure using modern orchestration, automation, and AI-driven operational frameworks aligned with industry practices.

Course Objectives

1. Understand enterprise cloud architecture patterns, migration strategies, and infrastructure lifecycle management
2. Design scalable, highly available infrastructure using modern orchestration platforms
3. Build automated CI/CD pipelines with integrated security and compliance enforcement
4. Implement observability and intelligent monitoring pipelines for enterprise systems
5. Apply enterprise-grade security practices including Zero Trust and compliance automation

Course Outcomes (CO)

CO	Description	Blooms Level
C01	Explain enterprise deployment architectures, migration strategies, and infrastructure lifecycle management	L2
C02	Design scalable and highly available infrastructure using Kubernetes and enterprise orchestration tools	L6
C03	Build automated CI/CD pipelines integrating security, policy enforcement, and infrastructure-as-code	L6
C04	Implement observability and intelligent monitoring systems for enterprise deployments	L3
C05	Apply enterprise security practices including Zero Trust and compliance frameworks	L3



Broad Course Description

This course provides a comprehensive foundation in enterprise-scale deployment strategies, focusing on the design, implementation, and management of modern cloud-native and hybrid infrastructure. It introduces core architectural principles, migration frameworks, orchestration platforms, and DevOps practices required for deploying scalable, secure, and resilient enterprise systems. The course emphasizes infrastructure automation, container orchestration using Kubernetes, and CI/CD pipelines integrated with security, compliance, and policy enforcement mechanisms. It further explores observability, AIOps-driven monitoring, and intelligent incident management for maintaining high availability and performance in production environments.

Advanced topics include multi-cloud and hybrid deployment strategies, disaster recovery planning, Zero Trust security models, and enterprise governance practices such as FinOps and compliance automation. The course integrates practical learning through real-world deployment scenarios and industry-aligned tools. The final module incorporates NVIDIA Deep Learning Institute (DLI)-aligned content, enabling students to work with GPU-accelerated infrastructure, AI-driven operations, and production-grade deployment pipelines. This equips learners with the skills required to design and manage next-generation enterprise systems that integrate cloud, AI, and high-performance computing.

CO-PO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
C01	2	1			1						1	2	2
C02	2	3	3	2	3						2	3	2
C03	2	3	3	2	3						2	3	3
C04	2	3	3	2	3						2	3	3
C05	2	3	3	3	3		1				2	3	3
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)													

Course Modules

Module - 1: Enterprise Cloud Architecture & Migration Frameworks (9 Hours)

Enterprise cloud lifecycle and migration strategies (6Rs), Cloud migration frameworks, Reference architectures and governance models, Infrastructure design (regions, hybrid connectivity), Workload classification and cost modeling, Applications and AI integration, Vendor lock-in, data residency, and architecture trade-offs

Module - 2: Enterprise Orchestration and Containerized Infrastructure (9 Hours)

Kubernetes architecture and enterprise distributions, Helm and workload orchestration, Multi-tenancy and resource management, Storage and autoscaling (HPA, VPA), Service mesh and observability, Applications and AI integration, Resource contention, fairness, and sustainability

Module - 3: Enterprise CI/CD, GitOps, and Infrastructure-as-Code (9 Hours)

CI/CD pipelines and tools, GitOps principles and workflows, Infrastructure-as-Code and Policy-



as-Code, Supply chain security and secrets management, Deployment strategies and automation, Applications and AI integration, Security and governance considerations

Module - 4: AIOps, Observability, and Intelligent Incident Management (9 Hours)

Observability fundamentals (metrics, logs, traces), Monitoring and tracing tools, AIOps (anomaly detection, root cause analysis), Alerting and incident response, SRE practices and chaos engineering, FinOps and cost monitoring, Applications and AI integration, Explainability, privacy, and human oversight

Module - 5: GPU-Accelerated Enterprise Deployments and AI Infrastructure (NVIDIA DLI Aligned) (9 Hours)

GPU-accelerated enterprise infrastructure, AI workload integration, GPU scheduling and orchestration, High-performance model deployment and inference, Distributed AI pipelines and optimization, Performance benchmarking and profiling, NVIDIA DLI hands-on implementation, Production AI systems and MLOps integration

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Module-1: Students use AI-assisted cloud design tools and simulation platforms to model enterprise architectures, including landing zones and migration strategies. AI-based cost estimation and workload classification tools are used to analyze infrastructure requirements and optimize deployment strategies.

Module-2: Students apply AI-enabled orchestration tools within containerized environments to optimize Kubernetes scheduling, autoscaling, and resource allocation. Intelligent monitoring dashboards assist in understanding cluster behaviour and workload distribution.

Module-3: Students utilize AI-assisted CI/CD platforms to automate pipeline design, testing, and deployment workflows. AI-driven tools are used for code quality analysis, security scanning, and anomaly detection in deployment pipelines.

Module-4: Students implement AI-driven observability and AIOps systems for anomaly detection, root cause analysis, and predictive incident management. Machine learning models are used to analyze logs, metrics, and traces for intelligent alerting and system optimization.

Module-5: Students design GPU-accelerated enterprise deployment pipelines using NVIDIA ecosystem tools. AI-driven infrastructure management, GPU workload optimization, and deployment of AI models in production environments are explored through hands-on labs aligned with NVIDIA DLI content.



Authentic Intelligence Reflection (AI² Pedagogy):

Module-1:

Understanding trade-offs in cloud architecture decisions, including vendor lock-in, cost optimization, and data sovereignty. Students evaluate when automation should be balanced with human decision-making in migration planning.

Module-2:

Recognizing limitations of container orchestration in resource-constrained environments and addressing fairness in multi-tenant clusters. Students analyze sustainability concerns in large-scale infrastructure deployments.

Module-3:

Evaluating risks in fully automated CI/CD pipelines, including security vulnerabilities, misconfigurations, and lack of human oversight. Students develop critical judgment in validating AI-assisted deployment decisions.

Module-4:

Understanding the limitations of AI-driven observability systems, including false positives in anomaly detection and challenges in explainability of automated root cause analysis.

Module-5:

Evaluating cost-performance trade-offs of GPU infrastructure, limitations of accelerated systems, and ethical considerations in deploying large-scale AI workloads in enterprise environments.

Assessment Scheme based on the course

Criteria	Weightage of Marks
Class Tests	10
Mid-Semester Examination	30
Mini Project	10
Assignments	10
End-Semester Examination	40
Total	100 marks



Recommended Tools

Primary Textbooks

1. NVIDIA Corporation, RAPIDS Documentation and User Guide, NVIDIA Developer Documentation, Latest Edition (Online Resource).
2. David B. Kirk, Wen-mei W. Hwu, *Programming Massively Parallel Processors: A Hands-on Approach*, Morgan Kaufmann, 4th Edition, 2021.
3. John Cheng, Max Grossman, Ty McKercher, *Professional CUDA C Programming*, Wrox, 1st Edition, 2014.
4. Aurélien Géron, *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*, O'Reilly Media, 3rd Edition, 2022.
5. Ian Goodfellow, Yoshua Bengio, Aaron Courville, *Deep Learning*, MIT Press, 1st Edition, 2016.
6. François Chollet, *Deep Learning with Python*, Manning Publications, 2nd Edition, 2021.

Additional Resources

1. Thomas Erl, *Cloud Computing: Concepts, Technology & Architecture*, Prentice Hall, 1st Edition, 2013.
2. Kelsey Hightower, Brendan Burns, Joe Beda, *Kubernetes: Up & Running*, O'Reilly Media, 3rd Edition, 2022.
3. Martin Kleppmann, *Designing Data-Intensive Applications*, O'Reilly Media, 1st Edition, 2017.
4. Gene Kim, Jez Humble, Patrick Debois, John Willis, *The DevOps Handbook*, IT Revolution Press, 2nd Edition, 2021.
5. Betsy Beyer, Chris Jones, Jennifer Petoff, Niall Richard Murphy, *Site Reliability Engineering*, O'Reilly Media, 1st Edition, 2016.
6. NVIDIA Corporation, *NVIDIA Developer Documentation (GPU Computing, Deployment, and AI Infrastructure)*, Latest Edition (Online Resource).

Mini Project Component

Industry-Driven Projects:

Project 1 – Enterprise Cloud Landing Zone & Architecture Design

- Problem: Design a secure and scalable enterprise cloud architecture using a landing zone model for a regulated industry.
- Concepts Used: Cloud architecture patterns, landing zones, hub-and-spoke networking, IAM-governance, workload-classification.

Project 2 – Enterprise Cloud Migration Strategy & Execution (6Rs Framework) (Module 1)

- Problem: Analyze and design a migration strategy for a legacy enterprise system using the 6Rs framework.
- Concepts Used: Migration frameworks, cost analysis (TCO), workload assessment, hybrid connectivity-design.



Project 3 – Kubernetes-Based Enterprise Application Deployment (Module 2)

- Problem: Deploy a containerized enterprise application on Kubernetes with high availability and scalability.
- Concepts Used: Kubernetes architecture, deployments, services, autoscaling (HPA/VPA), namespaces.

Project 4: GPU-Accelerated Workload Deployment using Kubernetes & NVIDIA GPU Operator (Module 2)

- Problem: Deploy and manage GPU-enabled workloads in a Kubernetes cluster using automated GPU provisioning.
- Concepts Used: NVIDIA GPU Operator, Kubernetes scheduling, resource quotas, GPU workload isolation.

Project 5 – Enterprise Helm-Based Application Lifecycle Management (Module 2)

- Problem: Design and manage application deployments using Helm charts with versioning and rollback strategies.
- Concepts Used: Helm charts, repositories, release management, configuration templating.

AI-Powered Assignments

Assignment 1: AI-Based Infrastructure Performance Analysis (CPU vs GPU in Enterprise Systems)

Students prompt an AI tool: Compare CPU-based and GPU-based infrastructure performance for enterprise workloads and recommend appropriate deployment scenarios.

Students must:

- Analyze the correctness of AI-generated comparisons using performance metrics (throughput, latency, resource utilization, cost efficiency)
- Identify incorrect assumptions (e.g., ignoring workload type, overestimating GPU benefits, neglecting cost and scalability constraints)
- Improve the analysis using real-world deployment scenarios (e.g., containerized applications, AI inference workloads, cloud-based infrastructure benchmarking)

Assignment 2: AI-Assisted Data Preprocessing Optimization

Students prompt an AI tool: Suggest an optimized data preprocessing pipeline for a large dataset using GPU acceleration.

Students must:

- Analyze the correctness of preprocessing steps (handling missing values, encoding, scaling)
- Identify inefficiencies or unnecessary operations in the AI-generated pipeline
- Refine the pipeline using cuDF-based transformations and benchmarking results

Assessment focuses on data engineering skills, optimization strategies, and validation of AI outputs.

Assignment 3: AI-Based Machine Learning Model Selection and Validation



Students prompt an AI tool: Recommend the best machine learning model for a given dataset and justify the choice.

Students must:

- Analyze the suitability of the recommended model (e.g., regression, classification, clustering)
- Identify incorrect assumptions about data distribution or feature relevance
- Validate and improve results using cuML models and performance metrics

Assessment focuses on model selection, analytical reasoning, and comparative evaluation

Assignment 4: AI-Assisted Clustering and Dimensionality Reduction

Students prompt an AI tool: Perform clustering and dimensionality reduction on a high-dimensional dataset.

Students must:

- Analyze clustering quality and dimensionality reduction outputs (PCA/UMAP/t-SNE)
- Identify misinterpretations or poor parameter choices
- Improve results using GPU-accelerated cuML implementations and visualization

Assessment focuses on unsupervised learning understanding, interpretation, and validation

Assignment 5: AI-Based Distributed Data Pipeline Design

Students prompt an AI tool: Design a distributed data processing pipeline for large-scale datasets using multi-GPU systems.

Students must:

- Analyze the trend predicted by AI for size-dependent properties
- Analyze the feasibility of the proposed architecture (Dask-cuDF, data partitioning)
- Identify bottlenecks such as communication overhead or poor data locality



Adversarial Machine Learning

Course Code:26DS4808

Semester: VIII

Programs Offered: B.Tech. CSE (Data Science)

Course Type: Professional Elective (PE-V)

Credits: 3

L – T – P – J: 3 – 0 – 0 – 0

Prerequisites: Calculus, linear algebra, and machine learning

Course Vision

To develop robust and secure AI systems by understanding adversarial threats, defenses, and privacy risks, while promoting trustworthy and human-centric machine learning practices.

Course Objectives

1. Understand the foundations of adversarial machine learning and threat models.
2. Analyze vulnerabilities in deep learning models and attack strategies.
3. Apply techniques to detect and defend against adversarial and poisoning attacks.
4. Explore privacy attacks and secure learning mechanisms.
5. Design robust and trustworthy AI systems with ethical considerations.

Course Outcomes (COs)

CO	Description	Blooms Level
CO1	Illustrate adversarial machine learning concepts, attack types, and threat models	L2
CO2	Analyze deep learning model behaviour and detection mechanisms in adversarial settings	L4
CO3	Apply test-time evasion attacks and evaluate defense strategies	L3
CO4	Develop techniques to detect and mitigate backdoor and data poisoning attacks	L6
CO5	Evaluate privacy attacks and design robust, secure, and trustworthy ML systems	L5

Broad Course Description

This course provides a comprehensive understanding of adversarial machine learning, focusing on the vulnerabilities of modern AI systems and the techniques used to exploit them. It covers various attack models including evasion, poisoning, and reverse-engineering attacks, along with their impact on real-world applications. The course also emphasizes defense mechanisms, privacy-preserving techniques, and robust learning strategies. Students will explore secure AI



system design while addressing ethical concerns, trust, and reliability, enabling the development of resilient and trustworthy machine learning systems.

CO-PO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
C01	3	3	1		2								3	1
C02	3	2	2		3								3	1
C03	2	3	3		3								3	2
C04	2	2	3		3								3	2
C05	2	3	2		2								2	3
3: Substantial (High) 2: Moderate (Medium) 1: Slight (Low)														

Course Modules

Module 1: Foundations of Adversarial Machine Learning

(9 Hours)

Machine learning threat landscape including training-time and test-time attacks; attacker and defender goals and capabilities; test-time evasion attacks; data poisoning attacks; reverse-engineering attacks; privacy attacks such as membership inference and model inversion; adversarial learning taxonomy and frameworks. Applications: Spam filter evasion, intrusion detection attacks, facial recognition vulnerabilities.

Module 2: Deep Learning Foundations for Adversarial Analysis

(9 Hours)

Deep neural network architectures such as CNNs and ResNets; gradient-based optimization and loss landscapes; overfitting and regularization; model interpretability techniques; statistical learning basics including PCA, K-means, and EM; detection theory fundamentals. Applications: Image classification, anomaly detection, confidence estimation.

Module 3: Test-Time Evasion Attacks and Defenses

(9 Hours)

Adversarial attack methods including FGSM, PGD, BIM, JSMA, and Carlini-Wagner; adversarial training and defensive distillation; statistical detection techniques; GAN-based detection; out-of-distribution detection; robust and certified defenses. Applications: Adversarial image perturbations, real-time detection systems, secure AI deployment.



Module 4: Backdoor Attacks and Data Poisoning Defenses

(9 Hours)

Backdoor attacks and trigger mechanisms; data poisoning strategies; reverse-engineering defenses; universal backdoor detection and mitigation; test-time detection of triggers; secure training pipelines and validation methods. Applications: Supply chain AI security, poisoned dataset cleaning, trigger detection systems.

Module 5: Privacy Attacks and Robust Learning

(9 Hours)

Reverse-engineering attacks; membership inference and model inversion attacks; differential privacy; robust deep regression and active learning; open challenges in adversarial ML. Applications: Privacy-preserving AI systems, secure healthcare models, financial risk prediction.

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

- Module 1:** AI systems incorporate adversarial awareness to ensure robustness against malicious inputs.
Authentic Intelligence: Reveals limitations of AI reliability under adversarial conditions.
- Module 2:** AI models leverage interpretability and statistical methods to analyze vulnerabilities.
Authentic Intelligence: Highlights gaps between machine perception and human understanding.
- Module 3:** AI enables both generation and detection of adversarial examples.
Authentic Intelligence: Demonstrates how minor perturbations can significantly mislead AI systems.
- Module 4:** AI-driven techniques detect hidden malicious patterns in models and datasets.
Authentic Intelligence: Promotes realistic defense strategies under uncertain threat conditions.
- Module 5:** AI integrates privacy-preserving mechanisms to secure sensitive data.
Authentic Intelligence: Ensures ethical AI by protecting user privacy and preventing data leakage.

Assessment Scheme

Criteria	Weightage of Marks
Class Tests /assignments	10
Mid-Semester Examination	30
Hackathon/Technical Competition/Paper presentation/Global/DLI certifications	10
AI Powered Assignment	10



End-Semester Examination	40
Total	100 marks

Recommended Tools

Primary Textbooks:

1. Miller, D. J. et al. – *Adversarial Learning and Secure AI*, Cambridge University Press.
2. Joseph, A. D. et al. – *Adversarial Machine Learning*, Cambridge University Press.

Additional Resources

Additional AI & Computational Resources

- Goodfellow, I. et al. – *Deep Learning*, MIT Press.
- NVIDIA Deep Learning Institute (DLI) — **Exploring Adversarial Machine Learning**, https://learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+S-DS-03+VI

AI-Powered Assignments

1. Adversarial Attack Implementation (Hands-on)

- Implement basic adversarial attacks such as FGSM and PGD on a CNN trained on MNIST/CIFAR-10.
- Compare model accuracy before and after attack.
- *Outcome:* Understand test-time evasion attacks

2. Adversarial Defense Experiment

- Apply adversarial training or defensive distillation on the attacked model.
- Evaluate robustness improvement using accuracy and attack success rate.
- *Outcome:* Analyze defense strategies

3. Black-box Attack Simulation

- Perform a black-box attack (transferability-based attack) without direct model access.
- Compare effectiveness with white-box attacks.
- *Outcome:* Understand real-world attack scenarios

4. Backdoor Attack Detection Project

- Inject a simple trigger into a dataset and train a backdoored model.
- Design a detection mechanism (activation clustering / anomaly detection).
- *Outcome:* Explore data poisoning and backdoor defense.



5. Membership Inference Attack Study

- Implement a membership inference attack on a trained model.
- Analyze privacy leakage and mitigation techniques.
- *Outcome:* Evaluate privacy risks.

6. Explainability under Attack

- Use saliency maps or Grad-CAM to visualize how adversarial examples affect model decisions.
- Compare clean vs adversarial explanations.
- *Outcome:* Link interpretability with adversarial behaviour.

7. Robustness Benchmarking Assignment

- Compare multiple models (CNN, ResNet, etc.) under adversarial attacks.
- Rank models based on robustness metrics.
- *Outcome:* Analyze model vulnerability.

8. GAN-based Adversarial Example Generation

- Use GANs to generate adversarial samples and test model robustness.
- Compare GAN-based attacks with gradient-based attacks.
- *Outcome:* Advanced adversarial generation techniques.