Dayananda Sagar University

School of Engineering

Devarakaggalahalli, Harohalli, Kanakapura Road, Bangalore South Dt., Bengaluru – 562 112

Department of

Computer Science & Technology

B.Tech. PROGRAMME– 2024-2028 BATCH

SCHEME AND SYLLABUS

(1th to 8th semester)

Academic Year 2026-27

Implementation of National Education Policy (NEP) 2020 for the B.Tech students of Batch 2024-2028

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

1. Student Centric flexible curriculum.
2. Inter-disciplinary 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.

PART-A

GOVERNING REGULATIONS FOR BACHELOR OF TECHNOLOGY (B. Tech.) 2024-28

PREAMBLE

The School of Engineering under Dayananda Sagar University (DSU) provides Science & Technology based education leading to the development of high calibre engineers suitable for Industry and Scientific Organization. The curriculum focuses on knowledge-based course work integrated with skill development as a part of training. It equally helps in inculcating the scientific temper for the lifelong processes of learning. At the Undergraduate level, a candidate goes through the foundation courses in Science, Humanities & Engineering. Each department ensures that the courses cover both the core & electives courses, as required. Provision for Institutional elective help the candidates to acquire interdisciplinary knowledge base or specialize significantly in an area outside the parent discipline.

DEFINITIONS OF KEY WORDS

(i) **Academic Year:** Two consecutive odd, even semesters and a summer term for make up if required.

(ii) **Course:** Usually referred to as a subject, a course may consist of any of Lecture/Tutorials/Practical /Seminar/Mini project/Project work.

(iii) **Credit:** A unit by which the course work is measured. One credit is equivalent to one hour of lecture or one hour of tutorial or two hours of laboratory/practical/ workshop practice per week.

(iv) **Credit Point:** It is the product of grade point and number of credits per course.

(v) **Cumulative Grade Point Average (CGPA):** It is the measure of overall cumulative performance over all semesters. It is expressed up to two decimal places.

(vi) **First Attempt:** If a candidate has completed all formalities of academic requirement in a term and become eligible to attend the examinations and attend all the end semester

examinations, such attempt shall be considered as first attempt.

(vii) **Grade Point:** It is a numerical weight allotted to each letter grade on a 10-point scale.

(viii) **Letter Grade:** It is an index of the performance in a said course. Grades are denoted by alphabets.

(ix) **Programme:** An educational activity leading to award a degree or Certificate.

(x) **Semester Grade Point Average:** It is a measure of performance during a semester. It shall be expressed up to two decimal places.

(xi) **Transcript:** Based on the grades earned, a grade certificate shall be issued after every semester to the candidate registered.

(xii) **Failure:** It is the case of appearing for Semester End Examinations but fails to obtain minimum passing marks in Semester End Examinations.

(xiii) **Detain**: It is the case of not satisfying the eligibility criteria w.r.t Attendance /Internal Assessment in each course to appear for Semester End Examination.

(xiv) **Audit Course:** A course to be taken by the student without benefit of a grade or a credit.

(xv) **Not Fit For The Program (NFFTP)**: It is the failure of satisfying the criteria laid down by regulations to continue the program of study, which leads to the termination from the University

RULES AND REGULATIONS

UG 1. All B.Tech programmes offered by the University shall be governed by the DSU B.Tech Rules and Regulations – 2023.

UG 2. The B. Tech. rules and regulations shall be applicable to any new discipline(s) that may be introduced in future.

UG 3. A candidate shall become eligible for the award of the B.Tech. Degree after fulfilling all the academic requirements as prescribed by the B.Tech. Rules and Regulations of DSU.

UG 4. ELIGIBILITY FOR ADMISSION

UG 4.1. Admission to First Year Bachelor of Technology shall be open to candidates who have passed the second year Pre-University or XII standard or equivalent examination recognized by the University.

UG 4.2. The candidate shall have studied and passed English as one of the courses and secured not less than forty five percent (45%) marks in aggregate with Physics and Mathematics as compulsory courses, along with any one of the following courses, namely, Chemistry, Bio-Technology, Computer Science, Biology and Electronics. Eligibility shall be 40% in optional courses in case of candidates belonging to SC/ST and OBC candidates from Karnataka.

UG 4.3. Admission to II year /III Semester Bachelor of Technology under Lateral entry shall be open to the candidates who have passed diploma or equivalent qualification as recognized by statutory and regulatory body.

UG 4.4. For candidates who have completed Diploma from other than State of Karnataka, their eligibility shall be based on the recognition of the Diploma awarding Boards by the University.

UG 4.5. Diploma candidates seeking admission under Lateral entry shall take up bridge courses as prescribed in the Scheme of Teaching.

UG 4.6. Admission to II year /III Semester Bachelor of Technology shall be open to candidates who have passed B. Sc. degree from a recognized University or equivalent as recognized by the University and secure not less than 45% marks in aggregate (including all semesters). Eligibility shall be 40% in case of candidates belonging to SC/ST and OBC candidates from Karnataka.

UG 4.7. B.Sc. Graduates seeking admission under Lateral entry shall take up bridge Courses as prescribed in the Scheme of Teaching.

UG 5. ACADEMIC SESSION

UG 5.1. Each academic session is divided into two semesters of approximately sixteen Weeks duration and a summer term: an odd semester (August -December), an even semester (January - May) and summer term (Make-up term) June-July.

UG 5.2. The approved schedule of academic activities for a session, inclusive of dates for registration, mid-semester and end-semester examinations, vacation breaks, shall be laid down in the Academic Calendar for the session.

UG 6. CHANGE OF BRANCH

UG 6.1. Normally a candidate admitted to a particular branch of the undergraduate programme will continue studying in that branch till completion.

UG 6.2. However, in special cases, the University may permit a candidate to change from one branch of studies to another after the first two semesters. Such changes will be permitted, in accordance with the provisions laid down hereinafter.

UG 6.3. Only those candidates will be considered eligible for change of branch after the second semester, who have completed all the credits required in the first two semesters of their studies in their first attempt, without having to pass any course requirement in the summer term examination.

UG 6.4. Applications for a change of branch must be made by intending eligible candidates in the prescribed form. The academic section will call for applications at the end of second semester of each academic year and the completed forms must be submitted by the last date specified in the notification.

UG 6.5. Candidates may enlist their choices of branch, in order of preference, to which they wish to change over. It will not be permissible to alter the choices after the application has been submitted.

UG 6.6. Change of branch shall be made strictly in the order of merit of the applicants. For this purpose, the CGPA obtained at the end of the second semester shall be considered. In case of a tie, SGPA of second semester followed by SGPA of first semester shall decide the tie.

UG 6.7. The applicants may be allowed a change in branch, strictly in order of merit, course to the limitation that the strength of a branch should not fall below the existing strength by more than ten percent and should not go above the sanctioned strength by more than ten percent. The minimum class strength of 75% should be maintained, while considering the change of branch.

UG 6.8. All changes of branch made in accordance with the above rules shall be effective from the third semester of the applicants concerned. No change of branch shall be permitted after this.

UG 7. COURSE STRUCTURE

UG 7.1. Medium of instruction, examination and project reports shall be in English except in case of any language audit courses.

UG 7.2. Teaching of the courses shall be reckoned in credits: Credits are assigned to the Courses based on the following general pattern:

(a) One credit for each lecture period.

(b) One credit for each tutorial period.

(c) One credit per two hours for each Laboratory or Practical or workshop session.

(d) Credits for seminar, mini project, project as indicated in the scheme/curriculum of teaching.

UG 7.3. In order to qualify for a B. Tech. degree of the University, a candidate is required to complete the credit requirement as prescribed in the scheme/curriculum for a particular programme.

UG 7.4. The program of a study consists of the following components:

(i) Humanities and Social Sciences including Management courses

(ii) Basic Science courses

(iii) Engineering Science courses

(iv) Professional core courses

(v) Open Electives

(vi) Project work, seminar and internship

(vii) Mandatory/Audit Courses

UG 7.5. Every B. Tech. Programme shall have a curriculum and syllabi for the courses approved by the Board of Governors. Board of Studies will discuss and recommend the syllabi of all the undergraduate courses offered by the department from time to time before sending the same to the Academic Council. Academic Council will consider the proposals from the Board of Studies and make recommendations to the Board of Management and Board of Governors for consideration and approval. For all approved courses, the copyright shall be with DSU.

UG 7.6. Faculty Advisor: To help the candidates in planning their courses of study and getting general advice on the academic programme, the concerned department will assign a Faculty Advisor to each candidate.

UG 8. REGISTRATION

UG 8.1. Every candidate is required to register for approved courses through the assigned Faculty Advisor at the end of previous semester or first week of the current semester, as notified by the Academic Calendar.

UG 8.2. The Dean may cancel the registration of one or more courses if they are found to violate some rules or if there are restrictions imposed due to disciplinary reasons.

UG 8.3. The student is permitted to drop a course/s from the registered courses, within 4 weeks after the start of the Semester/Year as notified in the academic calendar, with the permission of Faculty Advisor and Chairperson / Dean / Principal of the respective School/College and no mention will be made in the grade card for dropped courses.

UG 8.4. The student is permitted to withdraw course/s from the registered courses, within 4 weeks before the start of the Semester/Year End Examinations as notified in the academic calendar, with the permission of Faculty Advisor and Chairperson / Dean / Principal of the respective School / College and Grade "W" will be awarded for course/s that were withdrawn.

UG 8.5. For the courses with "W" grade, the students should re-register subsequently when offered, either in MOOCS or in-class or summer term and fulfil the passing criteria to secure a grade in that course for change from "W" grade.

UG 8.6. Only those candidates shall be permitted to register who have:

(a) The academic eligibility to move to higher semesters (UG 9 & UG 11)

(b) Cleared all University, Hostel and Library dues and fines (if any) of the previous semesters,

(c) Paid all required advance payments of University and Hostel dues for the current semester,

(d) Not been debarred from registering on any specific ground.

(e) A minimum CGPA of 4 in the previous semesters

UG 9. EXAMINATION: ASSESSMENT CRITERIA & ELIGIBILITYFOR PROGRESSION

Every student shall be assessed for eligibility to higher semester through Continuous Internal Assessment (CIA) and Semester End Examination (SEE) as prescribed.

UG 9.1. The Continuous Internal Assessment (CIA) shall normally be conducted by the assessment components spread through the running semester; the components of CIA may be tests, mid-term exam, quiz, term paper, simulation based problem solving, open-book test, solving open-end problems, mini- projects, seminars, viva-voce, awarding marks for attendance and such activities that enhance original thinking of students. The Course instructor shall announce the detailed methodology for conducting the various components of CIA together specifying component-wise weightages right in the commencement of each semester.

UG 9.2. The Semester End Examinations (SEE) shall be conducted at the end of each semester. The SEE components may be a closed or open book examination, project demo, viva-voce, and/or a portfolio presentation.

UG 9.3. CIA and SEE shall respectively have 60:40 percent weightage. The Vice-Chancellor, on the recommendations of the Dean of Faculty and Department Chair, in exceptional cases, may approve the variation in this weightage ratio.

UG 9.4. The performance of a student with respect to a course in a semester shall be the combined score of marks/points, he/she secures in CIA and SEE, put together. A minimum of securing 40% marks, combining both the CIA with SEE marks secured with respect to a course, shall entail the student a PASS in the course. The Vice-Chancellor, in such cases where the entire class has fared poorly in the course, upon receiving a representation by the students / department, and based on the recommendations of the committee constituted for the purpose, may review the criterion of 40%.

UG 9.5. ATTENDANCE ELIGIBILITY

UG 9.5.1. Candidates are required to attend all the classes (Lectures, Tutorials, Practical, Workshop Practice, etc.) for which they have been registered.

UG 9.5.2. The candidate shall not be allowed to appear for the end semester examination if his/her attendance falls below 85% in each course and shall be awarded a "NE" grade in that course.

UG 9.5.3. A provision for condonation of 10% of the attendance by the Vice-Chancellor on the specific recommendation of the chairman of the department and Dean, showing reasonable cause such as:

(a) Any medical emergencies/ illness where the candidate requires rest for the specified number of days certified by a Government Doctor only /any death in the

family (near and dear ones).

(b) If the student represents the University in Sports/ Cultural Activities/Extra- curricular activities/Co-curricular activities.

(c) If a student presents a Paper in National/ International Conferences or attends any recognized Workshops/Seminars.

UG 9.5.4. If the period of leave is for a short duration (less than two weeks), prior application for leave shall have to be submitted to the Chairman of the Department concerned stating fully the reasons for the leave requested for along with supporting document(s). Such leave will be granted by the Chairman of the Department. However, the student shall comply with 9.5.2 and 9.5.3. of regulations.

UG 9.5.5. If the period of absence is likely to exceed two weeks, a prior application for grant of leave will have to be submitted through the Chairman of the Department to the Dean with supporting documents in each case. The decision to grant leave shall be taken by the Dean on the recommendation of the Chairman of the Department. However the student shall comply with 9.5.2 and 9.5.3.of regulations.

UG 9.5.6. It shall be the responsibility of the candidate to intimate the concerned course instructor(s) regarding his/her absence before availing the leave.

UG 9.6. CONTINUOUS INTERNAL ASSESSMENT

UG 9.6.1. Candidate shall participate in all components of Continuous Internal Assessment (CIA) to become eligible to take up the Semester End Examination or else 'NE' grade shall be awarded. However, the Vice-Chancellor, under exceptional circumstances on the recommendations of Dean of Faculty and Department Chair, may exempt a student from participation in CIA component/s and permit taking up SEE.

UG 9.6.2. There shall be no marks improvement of Continuous Internal Assessment; however, the withdrawal and re-registering of the course shall be permitted.

UG 9.6.3. Continuous Evaluation consists of:

UG 9.6.3.1. Under normal circumstances for theory courses, total CIA weightage shall be a total of 60%, put together all components with varying weightages; Under exceptional circumstances with the approval of the Vice-Chancellor on the recommendation of Dean of the School, the weightage of CIA may be lower/higher than 60%.

The components of CIA may be tests, mid-term exam, quiz, term paper, simulation based problem solving, open-book test, solving open-end problems, mini-projects, seminars, viva-voce, awarding marks for attendance and such activities that enhance original thinking of students.

UG 9.6.3.2 Under normal circumstances for the practical courses (laboratory, workshops, and any such hands-on activity), total CIA weightage shall be a total of 60%, put together all components with varying weightages; Under exceptional circumstances with the approval of the Vice-Chancellor on the recommendation of Dean of the School, the weightage of CIA may be lower/higher than 60%.

CIA may have components such as conduction of an experiment, record writing, viva-voce, tests, simulation, mid-term exam, quiz, demo, term paper, mini-projects, seminars, marks for attendance and activities which enhances original thinking of students.

UG 10. GRADING

UG 10.1 There shall be continuous assessment of a candidate's performance throughout the semester and grades shall be awarded by the concerned course instructor and/or the appropriate committee appointed for this purpose on the following basis.

UG 10.2 The grading will normally be based on CIA and SEE.

UG 10.3. Practical Courses/ Workshop Practice: The evaluation will be based on instructor's continuous internal assessment, a test and end semester examination.

UG 10.4. The weightage assigned to different components of continuous internal assessment will be announced by the concerned instructor(s) in the beginning of the semester.

UG 10.5. The results of performance of the candidates in the Continuous Internal assessment Test shall be announced by the instructors.

UG 10.6. In case of seminar, evaluation will be as determined by the grade awarding Committee (as per the Program scheme).

UG 10.7. Mini project /projects will be based on Continuous evaluation by Guide(s) and Semester End Examination (as per the Program scheme)

UG 10.8. The results of performance of the candidates shall be announced by the Controller of Examinations.

UG 10.9. METHOD OF AWARDING LETTER GRADES

UG 10.9.1. Relationships among Grades, Grade points and % of marks are listed in Table1.

UG 10.10. DESCRIPTION OF GRADES

UG 10.10.1 Table 1 shows the relationships among the grades, grade points and percentage of marks.

UG 10.10.2. A student will have to ensure a minimum CGPA of 4, to become eligible for the award of the degree.

UG 10.10.3. A candidate shall have to repeat all courses in which he/she obtains 'F' Grades until a passing grade is obtained.

Table 1: Grade, Points, Grade Description and % of marks

GRADE	GRADE POINTS	DESCRIPTION	% MARKS
O	10	Outstanding	90 to 100
A+	9	Excellent	80 to 89
A	8	Very Good	70 to 79
B+	7	Good	60 to 69
B	6	Above Average	55 to 59
C	5	Average	50 to 54
P	4	Pass	40 to 49
F	0	Fail	< 40
AP	-	Audit Pass	-
AF	-	Audit Fail	-
IC	-	In Complete	-
NE	-	Not Eligible	-
W	-	Withdrawn	-

UG 10.10.4. An IC grade denotes incomplete performance in any Theory and/or Practical Assessment. It may be awarded in case of absence on medical grounds or other special circumstances for SEE. Requests for IC grade should be made at the earliest but not later than the last day of SEE.

UG 10.10.5. The student can appear for the course/s with IC grade, when exams are conducted subsequently by the University for those Courses.

UG10.11. EVALUATION OF PERFORMANCE

UG10.11.1. The performance of a candidate shall be evaluated in terms of the Semester Grade Point Average (SGPA) which is the Grade Point Average for a semester, Cumulative Grade Point Average (CGPA) which is the Grade Point Average for all the completed semesters.

UG 10.11.2. The Earned Credits (EC) are defined as the sum of course credits for courses in which candidates have been awarded grades between O to P. (Table 1)

UG 10.11.3. Points earned in a semester = (Course credits X Grade point) for Grades O - P

UG 10.11.4. The SGPA is calculated on the basis of grades obtained in all courses, except audit courses and courses in which F grade or below, registered for in the particular semester.

$$\text{SGPA} = \frac{\text{Points secured in the semester (O – P Grades)}}{\text{Credits registered in the semester, excluding audit}}$$

UG 10.11.5. The CGPA is calculated on the basis of all pass grades, except audit courses.

$$\text{CGPA} = \frac{\text{Cumulative points secured in all the passed courses (O – P Grades)}}{\text{Cumulative registered credits, excluding audit}}$$

UG 10.12. WITHHOLDING OF GRADES

UG 10.12.1. Grades shall be withheld when the candidate has not paid his/her dues or when there is a disciplinary action pending against him/her.

UG 10.13. CONVERSION OF CGPA INTO PERCENTAGE

UG 10.13.1. Conversion formula for the conversion of CGPA into percentage is Percentage of Marks Scored = (CGPA Earned − 0.75) × 10

UG 11. PROMOTION CRITERIA AND ENROLLMENTS TO HIGHER SEMESTERS

UG 11.1. During registration to the higher semesters, the following criteria/conditions for promotion, shall be satisfied.

UG 11.1.1. A student shall 'Not Eligible' (NE) for writing SEE if he/she does not comply to the minimum prescribed attendance in any course that carry a credit.

Students shall register afresh for such course/s, whenever offered next, to meet the attendance requirements and secure a pass grade, subsequently in that course/s.

UG 11.1.2. In a semester (ODD / EVEN), a student is deemed to be Not Eligible (NE) if he/she does not satisfy minimum attendance requirements criteria in a credit course.

If this course happens to be a prerequisite to a connected course in the subsequent semester, then the student shall not be permitted to register for that connected course until he / she secures pass grade in the prerequisite course by complying to the minimum attendance requirement when the prerequisite course is offered next (either during summer term or regular semester).

UG 11.1.3. A student shall be permitted to register for FOUR credited courses or to a total of 16 credits whichever is higher along with pending audit courses, if any, during a summer term by paying the prescribed course registration fee per credit notified by the university from time to time.

UG 11.1.4. The students with NE ('NOT ELIGIBLE' due to shortage in attendance) in any Credit Course/s other than Audit Courses in a semester shall have to secure a pass grade by compliance to minimum attendance requirements in the NE course to register for connected courses if NE course happens to be prerequisite course for those connected courses offered in the subsequent semesters.

UG 11.1.5. Candidates who secure 'F' grade in any courses in regular semester or summer term shall secure PASS grade in such course/s either in the subsequent summer term examination or shall repeat in the next appropriate semester whenever it is/they are offered, i.e. odd semester courses during odd semesters examinations and even semester courses during even semester examinations, respectively.

UG 11.2. In case of failure in Practical/Workshop practice course the candidate in any semester may clear it in the subsequent summer term examination or semester examination.

UG 11.3. In case a candidate fails in Practical/ Workshop practice he/she shall register when it is offered next either in the summer term or subsequent semester, as the case may be.

UG 11.4. Candidates may add and drop course(s) with the concurrence of the Faculty Advisor, and under intimation to the concerned course instructors and the academic section provided this is done within the date mentioned in the Academic Calendar.

UG 11.5. SUMMER TERM & MAKEUP EXAMINATIONS

UG 11.5.1. A summer term program may be offered by a department and with the approval of the Dean.

UG 11.5.2. Summer term courses will be announced by the Academic Affairs Office at the end of the even semester and before the commencement of the end semester examination. A candidate will have to register for summer term courses by paying the prescribed fees within the stipulated time in the announcement.

UG 11.5.3. The total number of contact hours in any summer term program will be the same as in the regular semester course. The assessment procedure in a summer term course will also be like the procedure for a regular semester course.

UG 11.5.4. Candidates granted semester drop by the Board of Governors, on medical ground, shall be allowed to clear the concerned courses in summer term course and subject to conditions as stated under clauses 11.5.1, 11.5.2.and 11.5.3.

UG 11.5.5. The Candidates with "NE" grade shall register for summer term by paying the prescribed fees.

UG 11.5.6. Candidates who are awarded 'F' grades in regular semester examinations have the option to register for the concerned courses in summer term examinations to the conditions as stated under clauses 11.5.1, 11.5.2.and 11.5.3above, or they can re-sit for subsequent semester/summer term examination only.

UG 11.5.7. Provision for make-up exam shall be available to the students who might have missed to attend the Semester / Annual end examinations of one or more courses for exceptional cases arising out of natural calamities / medical emergencies / death of a member in the family, with the permission of Faculty Advisor and Chairperson / Dean / Principal of the respective School/College. All such cases have to be exclusively to be approved by the Vice-Chancellor and ratified in the Academic Council / BOM / BOG. All such courses approved for makeup examinations are awarded a transitory grade "IC" (incomplete grade)

UG 11.5.8. The makeup examinations shall be held as notified in the academic calendar or through an exclusive notification duly approved by the Vice-chancellor.

UG 12. DURATION OF THE PROGRAMME

UG 12.1. Normally a candidate should complete all the requirements for undergraduate programme in four years. However, academically weaker candidates who do not fulfil some of the requirements in their first attempt and have to repeat them in subsequent semesters may be permitted up to eight consecutive years (from the first year of registration) to complete all the requirements of the degree.

UG 12.2. Normally a candidate under lateral entry should complete all the requirements for undergraduate programme in three years. However, academically weaker candidates who do not fulfil some of the requirements in their first attempt and have to repeat them in subsequent semesters may be permitted up to six consecutive years (from the second-year registration) to complete all the requirements of the degree.

UG 13. TERMINATION FROM THE PROGRAMME

UG 13.1. A candidate may also be compelled to leave the Program in the University on disciplinary grounds.

UG 13.2. On having been found to have produced false documents or having made false declaration at the time of seeking admission.

UG 13.3. On having been found to be pursuing regular studies and/or correspondence courses (leading to degree or diploma) in any other college, university, or an educational institution simultaneously.

UG 13.4. On having been found to be concurrently employed and performing duty or carrying out business in contravention to academic schedules of the University and without seeking approval from the University.

UG 13.5. If a student fails to earn a pass grade even after 4 attempts such a student is terminated from the university on the grounds of NOT FIT FOR THE PROGRAM (NFFTP).

UG 13.6. If a student secures a CGPA less than 4.0, 4 times during entire duration of the program of study, such a student is terminated from the university on the grounds of NOT FIT FOR THE PROGRAM (NFFTP).

UG 13.7. However, if the student appeals for reconsideration of termination from the university under NFFTP rule by providing the guanine reasons to the Vice- Chancellor through the Dean of Faculty, then the Vice-Chancellor may consider constituting a committee for the purpose of review and provide 2 additional attempts on the recommendations of the committee.

UG 14. TEMPORARY WITHDRAWAL FROM THE UNIVERSITY

UG 14.1. Candidate who has been admitted to an undergraduate programme of the University may be permitted to withdraw temporarily from the University on the grounds of prolonged illness or grave calamity in the family for a period of one semester or more, provided:

UG 14.1.1. He/she applies to the University within at least 6 weeks of the commencement of the semester or from the date he last attended his/her classes whichever is later, stating fully the reasons for such withdrawal together with supporting documents and endorsement of his/her guardian.

UG 14.1.2. The University is satisfied that, counting the period of withdrawal, the candidate is likely to complete his/her requirements of the B. Tech. Degree within the time limits specified in Clause 12.1 or 12.2 above.

UG 14.1.3. There are no outstanding dues or demands in the University / Hostel / Department / Library.

UG 14.1.4. Normally, a candidate will be permitted only one such temporary withdrawal during his/her tenure as a candidate of the undergraduate programme.

UG 15. TRANSFER OF CANDIDATES

UG 15.1. Transfer of candidates from higher education institutions outside university shall be considered at the beginning of Third and Fifth Semesters but subject to confirmation of equivalence.

UG 15.2. The candidates shall apply for equivalence with the No-objection for admission to DSU from the University where they are perusing their study.

UG 15.3. The candidates must have passed in all courses in the earlier semesters prior to transfer.

UG 16. ELIGIBILITY FOR THE AWARD OF B. Tech. DEGREE

A candidate shall be declared to be eligible for the award of B. Tech. degree if he/she has:

UG 16.1. Completed all the credit requirements for the degree with a CGPA 4.0 or higher at the end of the programme.

UG 16.2. Satisfactorily completed all the mandatory audit courses.

UG 16.3. No dues to the University, Department, Hostels.

UG 16.4. No disciplinary action pending against him/her.

UG 17. AWARD OF DEGREE

The award of B. Tech. degree must be recommended by the Academic Council and approved by the Board of Management and Board of Governors of the DSU.

UG 18. CONDUCT AND DISCIPLINE

UG 18.1. Candidates shall conduct themselves within and outside the precincts of the University in a manner befitting the candidates of an institution of national importance. The University has a separate ordinance Code and Conduct of Candidates which is applicable to all candidates of the University.

UG 19. REPEAL AND SAVINGS

Notwithstanding anything contained in these Regulations, the provisions of any guidelines, orders, rules or regulations in force at the University shall be inapplicable to the extent of their inconsistency with these Regulations. The Academic Council, Board of Management and Board of Governors of Dayananda Sagar University may revise, amend or change the regulations from time to time.

UG 20. INTERPRETATION

Any questions as to the interpretation of these Regulations shall be decided by the University, whose decision shall be final. The University shall have the powers to issue clarifications to remove any doubt, difficulty or anomaly which may arise during the implementation of the provisions of these regulations.

General Instructions:

- **Open Elective Courses:** At least two courses must be provided from each department and the courses shall be general course on emerging areas with broad coverage of syllabus so that student shall chose without any difficulty.
- **Honors Degree:** An Honors degree typically refers to a higher level of academic achievement in the major area. That is, certificate in his/her OWN major for Research orientation. The Credit requirement: **178 credits** (Major worth 160 credits + Honors 18 credits)
- **Minor Degree:** Minor is a secondary concentration of courses that often complements the honors. Minor in any OTHER branch for Improving Employability.
  - Minor is an option rather than a requirement for B. Tech students. They may opt for one of the Engineering or Non-engineering discipline as Minor, earning additional credits ranging from 18. However, students are permitted to choose only one Minor either from engineering or non-engineering discipline.
  - This opportunity is ideal for students who took a Major out of necessity but would still like to pursue their passion in another discipline or to enrich/equip them for a specific profession where greater job opportunities exist. Another advantage of opting for a Major with a Minor is to earn standing credits for pursuing a Master's degree abroad or within India too.
  - Only students who satisfy a set of minimum eligibility criteria set forth by the university and meet certain pre-requisites, will be permitted to opt for a Minor.
  - Credit requirement: **178 credits** (Major worth 160 credits + Minor 18 credits)
  - Degree nomenclature: The degree will contain the Major / Major with Specialization. The Minor pursued by the student will be provided in the transcript along with details on courses completed and associated credits earned.
  - For e.g., For a student who pursued Computer Science and Engineering with a Minor in Industrial Psychology, the degree will read "B. Tech in Computer Science and Engineering", Transcripts of B. Tech will reflect the Minor courses and the Minor certificate in Industrial Psychology will be issued separately.

PART-B

SCHEME AND SYLLABUS FOR B.TECH. CST (FIRST TO FOURTH YEARS) PROGRAMME

Definitions / Descriptions

Definition of Credit:	
1 Hour Lecture (L) Per Week	01 Credit
1 Hour Tutorial (T) Per Week	0.5 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

SCHEME - B.TECH – 2024-25 ONWARDS

I SEM - CHEMISTRY CYCLE

SL	COURSE CODE	COURSE TITLE	SCHEME OF TEACHING					Duration in Hrs Exam	Examination (Maximum Marks)		
			L	T	P	J	C		CIE	SEE	TM
1	24EN1101	LINEAR ALGEBRA AND DIFFERENTIAL EQUATIONS	3	0	0	0	3	3	60	40	100
2	24EN1102	OBJECT ORIENTED PROGRAMMING	2	1	2	0	4	3	60	40	100
3	24EN1103	COGNITIVE AND TECHNICAL SKILLS – I	0	0	4	0	2	1	100	-	100
4	24EN1104	ENGINEERING CHEMISTRY	2	0	2	0	3	3	60	40	100
5	24EN1105	INTRODUCTION TO MECHANICAL ENGINEERING	2	0	2	0	3	3	60	40	100
6	24EN1106	INTRODUCTION TO ELECTRICAL ENGINEERING	2	0	0	0	2	3	60	40	100
7	24EN1107	ENGINEERING MECHANICS	2	0	0	0	2	3	60	40	100
8	24EN1108	TECHNICAL ENGLISH	2	0	0	0	2	1	100	-	100
9	24EN1109	ENVIRONMENTAL SCIENCE	1	0	0	0	0	1	50	-	50
10	24EN1110	KANNADA KALI / MANASU	1	0	0	0	0	1	50	-	50
			17	1	10	0	21		660	240	900

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

SCHEME - B.TECH – 2024-25 ONWARDS

I SEM - PHYSICS CYCLE

SL	COURSE CODE	COURSE TITLE	SCHEME OF TEACHING					Duration in Hrs Exam	Examination (Maximum Marks)		
			L	T	P	J	C		CIE	SEE	TM
1	24EN1101	LINEAR ALGEBRA AND DIFFERENTIAL EQUATIONS	3	0	0	0	3	3	60	40	100
2	24EN1102	OBJECT ORIENTED PROGRAMMING	2	1	2	0	4	3	60	40	100
3	24EN1103	COGNITIVE AND TECHNICAL SKILLS – I	0	0	4	0	2	1	100	-	100
4	24EN1111	ENGINEERING PHYSICS	3	0	2	0	4	3	60	40	100
5	24EN1112	INTRODUCTION TO ELECTRONICS ENGINEERING	3	0	0	0	3	3	60	40	100
6	24EN1113	ENGINEERING GRAPHICS AND DESIGN THINKING	2	0	2	0	3	3	60	40	100
7	24EN1114	BIOLOGY FOR ENGINEERS	2	0	0	0	2	1	100	-	100
8	24EN1115	CONSTITUTION OF INDIA AND PROFESSIONAL ETHICS	1	0	0	0	0	1	50	-	50
			16	1	10	0	21		550	200	750

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

SCHEME - B.TECH – 2024-25 ONWARDS

II SEM - CHEMISTRY CYCLE

SL	COURSE CODE	COURSE TITLE	SCHEME OF TEACHING					Duration in Hrs Exam	Examination (Maximum Marks)		
			L	T	P	J	C		CIE	SEE	TM
1	24EN1201	SINGLE AND MULTIVARIATE CALCULUS	3	0	0	0	3	3	60	40	100
2	24EN1202	C PROGRAMMING FOR PROBLEM SOLVING	2	1	2	0	4	3	60	40	100
3	24EN1203	COGNITIVE AND TECHNICAL SKILLS – II	0	0	4	0	2	1	100	-	100
4	24EN1104	ENGINEERING CHEMISTRY	2	0	2	0	3	3	60	40	100
5	24EN1105	INTRODUCTION TO MECHANICAL ENGINEERING	2	0	2	0	3	3	60	40	100
6	24EN1106	INTRODUCTION TO ELECTRICAL ENGINEERING	2	0	0	0	2	3	60	40	100
7	24EN1107	ENGINEERING MECHANICS	2	0	0	0	2	3	60	40	100
8	24EN1108	TECHNICAL ENGLISH	2	0	0	0	2	1	100	-	100
9	24EN1109	ENVIRONMENTAL SCIENCE	1	0	0	0	0	1	50	-	50
10	24EN1110	KANNADA KALI / MANASU	1	0	0	0	0	1	50	-	50
			17	1	10	0	21		660	240	900

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

SCHEME - B.TECH – 2024-25 ONWARDS

II SEM - PHYSICS CYCLE

SL	COURSE CODE	COURSE TITLE	SCHEME OF TEACHING					Contact Hours /Week	Examination (Maximum Marks)		
			L	T	P	J	C		CIE	SEE	TM
1	24EN1201	SINGLE AND MULTIVARIATE CALCULUS	3	0	0	0	3	3	60	40	100
2	24EN1202	C PROGRAMMING FOR PROBLEM SOLVING	2	1	2	0	4	3	60	40	100
3	24EN1203	COGNITIVE AND TECHNICAL SKILLS – II	0	0	4	0	2	1	100	-	100
4	24EN1111	ENGINEERING PHYSICS	3	0	2	0	4	3	60	40	100
5	24EN1112	INTRODUCTION TO ELECTRONICS ENGINEERING	3	0	0	0	3	3	60	40	100
6	24EN1113	ENGINEERING GRAPHICS AND DESIGN THINKING	2	0	2	0	3	3	60	40	100
7	24EN1114	BIOLOGY FOR ENGINEERS	2	0	0	0	2	1	100	-	100
8	24EN1115	CONSTITUTION OF INDIA AND PROFESSIONAL ETHICS	1	0	0	0	0	1	50	-	50
			16	1	10	0	21		550	200	750

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

III SEMESTER													
S.NO	Course Type	Course Code	Course Name	Teaching Department	Teaching Hours / Week				Examination				Credits
					Lecture	Tutorial	Practical	Project	Duration in Hours	CIE Marks	SEE Marks	Total Marks	
					L	T	P	J					
1	BSC	24CT2301	Transforms and Numerical Techniques	MAT	3	0	0	0	03	60	40	100	**3**
2	IPCC	24CT2302	Data Structures	CST	3	0	2	0	03	60	40	100	**4**
3	IPCC	24CT2303	Digital Logic Design	ECE	3	0	2	0	03	60	40	100	**4**
4	PCC	24CT2304	Discrete Mathematics and Graph Theory	CSE	3	0	0	0	03	60	40	100	**3**
5	PCC	24CT2305	Full Stack Development	CST	3	0	0	2	03	60	40	100	**4**
6	SEC	24CT23XX	Skill Enhancement Course – I	CST	1	0	0	2	01	100	--	100	**2**
7	SEC	24CT2309	Product Marketing	CST	1	0	0	2	01	100	--	100	**2**
8	SEC	24CT2310	Cognitive and Technical Skills – III	Any Dept.	0	0	0	0	00	--	--	--	**P/F**
			Total		**17**	**00**	**04**	**06**	**17**	**500**	**200**	**700**	**22**

L – Lecture, T – Tutorial, P – Practical, J – Project, CIE – Continuous Internal Evaluation, SEE- Semester End Examinations.

Skill Enhancement Course – I		
S.NO	**Course Code**	**Course Name**
1	24CT2306	Design Thinking & Ideation
2	24CT2307	Innovative Businesses & Breakthrough Technologies
3	24CT2308	R for Data Analysis and Application Development

IV SEMESTER													
S.NO	Course Type	Course Code	Course Name	Teaching Department	Teaching Hours / Week				Examination				Credits
					Lecture	Tutorial	Practical	Project	Duration in Hours	CIE Marks	SEE Marks	Total Marks	
					L	T	P	J					
1	BSC	24CT2401	Probability & Statistics	MAT	3	0	0	0	03	60	40	100	**3**
2	IPCC	24CT2402	Design and Analysis of Algorithms	CST	3	0	2	0	03	60	40	100	**4**
3	IPCC	24CT2403	Database Management System	CST	3	0	2	0	03	60	40	100	**4**
4	PCC	24CT2404	Introduction to Artificial Intelligence	CST	3	0	0	0	03	60	40	100	**3**
5	PCC	24CT2405	Computer Organization and Architecture	CST	3	0	0	0	03	60	40	100	**3**
6	PCC	24CT24XX	Skill Enhancement Course – II	CST	1	0	0	2	--	100	--	100	**2**
7	SEC	24CT2410	DevOps Fundamentals	CST	1	0	0	2	01	100	--	100	**2**
8	SEC	24CT2411	Cognitive and Technical Skills - IV	CST	0	0	0	0	--	--	--	--	**P/F**
Total					**17**	**00**	**04**	**04**	**16**	**500**	**200**	**700**	**21**

L – Lecture, T – Tutorial, P – Practical, J – Project, CIE – Continuous Internal Evaluation, SEE- Semester End Examinations.

Skill Enhancement Course – II		
S.NO	**Course Code**	**Course Name**
1	24CT2406	Small E-Business Launch
2	24CT2407	Idea Generation and Validation
3	24CT2408	Technical writing
4	24CT2409	Introduction to IPR & Registration

V SEMESTER														
S.NO	Course Type	Course Code	Course Name	Teaching Department	Teaching Hours / Week				Duration in Hours	CIE Marks	SEE Marks	Total Marks	Credits	
					Lecture	Tutorial	Practical	Project						
					L	T	P	J						
1	PCC	24CT3501	Theory of Computation	CST	3	1	0	0	03	60	40	100	**4**	
2	PCC	24CT3502	AI-Enabled Software Systems and Agile Project Management	CST	3	0	0	0	03	60	40	100	**3**	
3	IPCC	24CT3503	Machine Learning	CST	3	0	2	0	03	60	40	100	**4**	
4	PCC	24CT3504	Product Design and Lean Start-up Strategy	CST	2	0	0	2	03	60	40	100	**3**	
5	IPCC	24CT3505	Operating Systems for GPUs	CST	3	0	2	0	03	60	40	100	**4**	
6	PEC	24CT35XX	Professional Elective Course – I	CST	3	0	0	0	03	60	40	100	**3**	
7	SEC	24CT35XX	Skill Enhancement Course – III	CST	1	0	0	2	01	100	--	100	**2**	
8	SEC	24CT3513	Cognitive and Technical Skills - V	CST	0	0	0	0	00	--	--	--	**P/F**	
Total					**18**	**01**	**04**	**04**	**19**	**460**	**240**	**700**	**23**	

L – Lecture, T – Tutorial, P – Practical, J – Project, CIE – Continuous Internal Evaluation, SEE- Semester End Examinations.

NOTE: For Professional Elective Courses (PEC), the LTPJ division is given on the next page according to the type of course.

Professional Elective Course – I / MOOC/ Certification						
S.NO	**Course Code**	**Course Name**	**L**	**T**	**P**	**J**
1	24CT3506	Data Science & Visualization	2	0	0	2
2	24CT3507	Cyber Security Fundamentals	2	0	0	2
3	24CT3508	Interactive Art and Creative Coding	2	0	0	2
4	24CT3509	Introduction to Quantum Computing	3	0	0	0
5	24CT3510	Product Strategy & Management	3	0	0	0

MOOC Courses

Sl. No.	Course Code	Course Title
1	*CID	Data Analytics and Visualization with MS PowerBI [IIMB] 6 WEEKS Data Analytics with Python [NPTEL] 12 WEEKS
2	*CID	Practical Cyber Security for Cyber Security Practitioners, IIT Kanpur [NPTEL] 12 WEEKS
3	*CID	Quantum Computation and Quantum Information [NPTEL] 12 WEEKS
4	*CID	Product and Brand Management, IIT Roorkee [NPTEL] 12 WEEKS

Note: CID = NPTEL Course ID

Skill Enhancement Course – III		
S.NO	**Course Code**	**Course Name**
1	24CT3511	DevOps Fundamentals & Product Launch
2	24CT3512	AWS Web Services / MOOC

VI SEMESTER													
S.NO	Course Type	Course Code	Course Name	Teaching Department	Teaching Hours / Week				Examination				Credits
					Lecture	Tutorial	Practical	Project	Duration in Hours	CIE Marks	SEE Marks	Total Marks	
					L	T	P	J					
1	HSMC	24CT3601	Supply Chain Management and Operations	CST	2	0	0	0	03	60	40	100	**2**
2	IPCC	24CT3602	Computer Networks and Distributed Systems	CST	3	0	2	0	03	60	40	100	**4**
3	IPCC	24CT3603	Deep learning	CST	3	0	2	0	03	60	40	100	**4**
4	OEC	24OEXXXX	Open Elective – I	All.Depts	3	0	0	0	03	60	40	100	**3**
5	PEC	24CT36XX	Professional Elective Course – II	CST	3	0	0	0	03	60	40	100	**3**
6	PEC	24CT36XX	Professional Elective Course – III	CST	3	0	0	0	03	60	40	100	**3**
7	PROJ	24CT3614	Minor Project	CST	0	0	0	4	02	100	--	100	**2**
8	SEC	24CT3615	Cognitive and Technical Skills- VI	CST	0	0	0	0	00	100	--	100	**P/F**
			Total		**17**	**00**	**04**	**04**	**20**	**560**	**240**	**800**	**21**

L – Lecture, T – Tutorial, P – Practical, J – Project, CIE – Continuous Internal Evaluation, SEE- Semester End Examinations.

NOTE1: For Professional Elective Courses (PEC), the LTPJ division is given on the next page according to the type of course.

Internship Note: All the students admitted to III year shall have to undergo mandatory internship of 4 weeks during the vacation of VI and VII semesters and /or VII and VIII semesters. A university examination shall be conducted during VIII semester and the prescribed credit shall be included in VIII semester. Internship shall be considered as a head of passing and shall be considered for the award of degree. Those, who do not takeup / complete the internship shall be declared fail and shall have to complete during subsequent University examination after satisfying the internship requirements

Open Electives-I		
S.No	**Course Code**	**Course Name**
1	24OEXXXX	Small E-Business Launch
2	24OEXXXX	Product Engineering & Entrepreneurship

Professional Elective Course – II / MOOC/ Certification						
S.NO	**Course Code**	**Course Name**	**L**	**T**	**P**	**J**
1	24CT3604	Computer Vision	3	0	0	0
2	24CT3605	Cryptography & Cyber Security	2	0	0	2
3	24CT3606	Gaming Design Fundamentals	2	0	0	2
4	24CT3607	Quantum Algorithms	2	0	0	2
5	24CT3608	Business Management using Cloud	3	0	0	0

MOOC Courses

Sl. No.	Course Code	Course Title
1	*CID	Modern Computer Vision, IIT Madras [NPTEL] 12 WEEKS
2	*CID	Cryptography and Network Security, IIT Kharagpur [NPTEL] 12 WEEKS
3	*CID	Quantum Computing: Algorithms and Limitations Through the Query Model [NPTEL] 12 WEEKS

Note: CID = NPTEL Course ID

Professional Elective Course – III / MOOC/ Certification						
S.NO	**Course Code**	**Course Name**	**L**	**T**	**P**	**J**
1	24CT4709	Building LLM with Prompt engineering	3	0	0	0
2	24CT4710	IoT-Based Data Analytics for Health Care	2	0	0	2
3	24CT4711	Game Development using Unity/ Game Development Using Godot	2	0	0	2
4	24CT4712	Quantum Machine Learning	2	0	0	2
5	24CT4713	Business to Business Marketing	3	0	0	0

MOOC Courses

Sl. No.	Course Code	Course Title
1	*CID	Introduction To Internet Of Things, IIT Kharagpur [NPTEL]

Note: CID = NPTEL Course ID

VII SEMESTER													
S.NO	Course Type	Course Code	Course Name	Teaching Department	Teaching Hours / Week				Examination				Credits
					Lecture	Tutorial	Practical	Project	Duration in Hours	CIE Marks	SEE Marks	Total Marks	
					L	T	P	J					
1	IPCC	24CT4701	Entrepreneurial Finance & Deep-Tech Startups	CST	3	0	0	0	03	60	40	100	**3**
2	OEC	24OEXXXX	Open Elective – II	All. Dept.	3	0	0	0	03	60	40	100	**3**
3	PEC	24CT47XX	Professional Elective Course – IV	CST	3	0	0	0	03	60	40	100	**3**
4	PEC	24CT47XX	Professional Elective Course – V	CST	3	0	0	0	03	60	40	100	**3**
5	PROJ	24CT4712	Capstone Project-Phase I	CST	0	0	0	8	03	60	40	100	**4**
			Total		**12**	**00**	**00**	**08**	**15**	**300**	**200**	**500**	**16**

L – Lecture, T – Tutorial, P – Practical, J – Project, CIE – Continuous Internal Evaluation, SEE- Semester End Examinations.

Open Electives-II		
S.NO	**Course Code**	**Course Name**
1	24OEXXXX	Lean Startup Methodology
2	24OEXXXX	Product Analytics

Professional Elective Course – IV / MOOC/ Certification						
S.NO	**Course Code**	**Course Name**	**L**	**T**	**P**	**J**
1	24CT4702	Generative AI with Diffusion Models	3	0	0	0
2	24CT4703	Blockchain Technology	2	0	0	2
3	24CT4704	AR/VR & Game-Theoretic/ Algorithmic Game Theory	2	0	0	2
4	24CT4705	Quantum Artificial Intelligence	2	0	0	2
5	24CT4706	Sustainable Innovation Business	3	0	0	0

MOOC Courses

Sl. No.	Course Code	Course Title
1	*CID	Blockchain and its Applications, IIT Kharagpur [NPTEL] 12 WEEKS

Note: CID = NPTEL Course ID

Professional Elective Course – V / MOOC/ Certification						
S.NO	**Course Code**	**Course Name**	**L**	**T**	**P**	**J**
1	24CT4807	Agentic AI and RAG Engineering	3	0	0	0
2	24CT4808	Business Intelligence & Analytics	2	0	0	2
3	24CT4809	Unreal Engine Game Development for Beginners	2	0	0	2
4	24CT4810	Quantum Cryptography	2	0	0	2
5	24CT4811	Digital Business	3	0	0	0

MOOC Courses

Sl. No.	Course Code	Course Title
1	*CID	Quantum Algorithms and Cryptography, IIT Madras [NPTEL] 12 WEEKS
2	*CID	Foundation of Digital Business, IIT Kharagpur [NPTEL] 8 WEEKS Business Intelligence & Analytics[NPTEL] 4 WEEKS

Note: CID = NPTEL Course ID

VIII SEMESTER												
S.NO	Course Type	Course Code	Course Name	Teaching Department	Teaching Hours / Week				Examination			Credits
					Lecture	Tutorial	Practical	Project	CIE Marks	SEE Marks	Total Marks	
					L	T	P	J				
1	PEC	26CT4801	Leadership, Policy & Authentic Intelligence	CST	2	0	0	0	100	--	100	**2**
2	PROJ	24CT4801	Capstone Project-Phase II	CST	0	0	0	20	60	40	100	**10**
3	INT	24CT4802	Internship	--	0	0	0	6	100	--	100	**03**
Total					**00**	**00**	**00**	**30**	**160**	**40**	**200**	**15**

L – Lecture, T – Tutorial, P – Practical, J – Project, CIE – Continuous Internal Evaluation, SEE- Semester End Examinations.

NOTE 1: Internship

Completed during the intervening vacations of VI and VII semesters and /or VII and VIII semesters or VIII semesters

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

I -21 V -23
II -21 VI -21
III -22 VII - 16
IV - 21 VIII – 15

Total=160

Elective courses domain-wise

S. NO	Domain-wise	Domain Clusters	PROFESSIONAL ELECTIVE COURSES				
			PEC-I	PEC-II	PEC-III	PEC-IV	PEC-V
			5th Semester	6th Semester		7th Semester	
			Course Name	Course Name	Course Name	Course Name	Course Name
1	Domain-1	Cognitive AI & Intelligent Systems	Data Science & Visualisation	Computer Vision	Building LLM with Prompt engineering	Generative AI with Diffusion Models	Agentic AI and RAG Engineering
2	Domain-2	Smart Security Systems	Cyber Security Fundamentals	Cryptography & Cyber Security	IoT-Based Data Analytics for Health Care	Blockchain Technology	Business Intelligence & Analytics+
3	Domain-3	Interactive Digital Media	Interactive Art and Creative Coding	Gaming Design Fundamentals	Game Development Using Unity/ Game Development Using Godot	AR/VR & Game-Theoretic/ Algorithmic Game Theory	Unreal Engine Game Development for Beginners
4	Domain-4	Quantum Computing	Quantum Computing & Systems	CUDA Programming	Quantum Algorithms+	Quantum Machine Learning	Quantum Cryptography & Post Quantum Security
5	Domain-5	Technology Innovation & Entrepreneurship	Product Strategy & Management	Business Management using Cloud	Business-to-Business Marketing	Sustainable Innovation Business	Digital Business

TRANSFORMS AND NUMERICAL TECHNIQUES

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

SEMESTER – III

Subject Code	:	24CT2301	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P-J	:	3–0–0-0			

Course Learning Objectives:

This Course will enable students to:

1. **Apply** their knowledge of Laplace transforms and inverse Laplace transforms to proficiently solve linear ordinary differential equations with constant coefficients, facilitating the analysis and modelling of complex systems.
2. **Analyze** periodic functions using Fourier series, assessing the convergence properties and precision of the series expansion, thereby enhancing their ability to understand and manipulate periodic phenomena.
3. **Utilize** complex exponential form, Fourier transforms of basic functions, and Fouriersine and cosine transforms to solve problems involving Fourier integrals, developingproficiency in applying these techniques to various mathematical scenarios.
4. **Employ** numerical methods, including Euler's Method, Runge-Kutta 4th order, Adams-Bashforth, and Adams-Moulton Methods, to solve differential equations and effectively analyze dynamic systems, enabling them to model real-world phenomenaand make accurate predictions**.**
5. **Apply** finite difference methods, including the Crank-Nicolson method and appropriate techniques for hyperbolic PDEs, to effectively solve different types of partial differential equations (PDEs) such as elliptic, parabolic, and hyperbolic equations, enhancing their problem-solving skills in the context of differential equations and their applications

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*** means it includes not only traditional lecture method, but different *type 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, notetaking, 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 analyse information rather than simply recall it.
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.

UNIT – I: Laplace Transform and Inverse Laplace Transform	09 Hours

Laplace Transforms of Elementary functions (without proof),
(Text Book-1: Chapter 6: 203 to 207).
Laplace Transforms of $e^{at}f(t)$, $t^n f(t)$ and $\frac{f(t)}{t}$, Periodic functions, Unit step function and impulse functions
(Text Book-1: Chapter 6:208-230).
Inverse Laplace Transforms- By the method of Partial Fractions, Logarithmic and Trigonometric functions, Convolution Theorem, Inverse Laplace transform using Convolution Theorem ***(Text Book-1: Chapter 6: 238)***.
Solution to Differential Equations by Laplace Transform.
(Text Book-1: Chapter 238-242).

UNIT – II: Fourier Series	09 Hours

Periodic Functions, Trigonometric Series
(Text Book-1: Chapter 11: 495).
Fourier series Standard function, Functions of any Period 2L, Even and Odd functions, Half-range Expansions.
(Text Book-1: Chapter 11: 483-492)
Practical Harmonic analysis (calculate average power and RMS values of periodicwaveforms)

UNIT – III: Fourier Transform	06 Hours

Calculation of Fourier integrals using complex exponential form
(Text Book-1: Chapter 11: 510).
Fourier transform of basic functions ***(Text Book-1: Chapter 11: 510-516).***
Fourier sine and cosine transforms. ***(Text Book-1: Chapter 11: 518-522).***

UNIT – IV: Numerical Methods for Solving Ordinary DifferentialEquations	07 Hours

Euler's Method-Basic principles of Euler's method for solving first-order ODEs ***(Text Book-1: Chapter 1:10-12).***
Runge-Kutta 4th order ***(Text Book-1: Chapter 21:904).***
Multistep Methods-Explanation of multistep methods (Adams-Bashforth, Adams–Moulton Methods) ***(Text Book-1: Chapter 21:911-913).***
Second-Order ODE. Mass–Spring System (Euler Method, Runge–Kutta Methods)
(Text Book-1: Chapter 21:916-918).

UNIT – V: Numerical Methods for Partial Differential Equations	08 Hours

Classification of PDEs (elliptic, parabolic, hyperbolic), ***(Text Book-1: Chapter 21:922-923).*** Finite Difference Methods (Laplace and Poisson Equations), Derivation of finite difference approximations ***(Text Book-1: Chapter 21:923-927).***
Crank–Nicolson Method ***(Text Book-1: Chapter 21:938-941).***
Method for Hyperbolic PDEs ***(Text Book-1: Chapter 21:943-945).***

Course Outcome	Description	Bloom's Taxonomy Level

At the end of the course the student will be able to:		
1	**Apply** Laplace transforms and inverse Laplace transforms tosolve linear ordinary differential equations with constant coefficients, demonstrating proficiency in system analysis andmodelling.	L3
2	**Analyze** periodic functions using Fourier series and evaluate the convergence properties and precision of the seriesexpansion.	L2 & L3
3	**Solve** problems involving Fourier integrals by applying complex exponential form, Fourier transforms of basicfunctions, and Fourier sine and cosine transforms.	L3
4	**Utilize** numerical methods such as Euler's Method, Runge-Kutta 4th order, Adams-Bashforth, and Adams-Moulton Methods to solve differential equations and analyze dynamic systems	L2 & L3
5	**Apply** finite difference methods, including the Crank-Nicolson method and appropriate techniques for hyperbolic PDEs, to solve various types of partial differential equations (PDEs)such as elliptic, parabolic, and hyperbolic equations.	L3

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
CO1	3	2	2	1					1					
CO2	3	2	2						1					
CO3	3	2	2	1					1					
CO4	3	2	2	1					1					
CO5	3	2	2	1					1					

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

TEXT BOOKS:

1. Erwin Kreyszig, Advanced Engineering Mathematics, 2015, 10th Edition, Wiley India.

REFERENCE BOOKS:

1. Higher Engineering Mathematics, B.S. Grewal, 2015, 43rd Edition, Khanna Publishers.
2. Higher Engineering Mathematics, John Bird, 2017, 6 th Edition, Elsevier Limited.

E-Resources:

1. https://nptel.ac.in/courses/111106139
2. https://nptel.ac.in/courses/111101164
3. https://nptel.ac.in/courses/111105038

****

DATA STRUCTURES

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

SEMESTER – III

Course Code	: 24CT2302	**Credits**	: 04
Hours / Week	: 03 Hours	**Total Hours**	: 39(Th)+26(P) Hours
L–T–P-J	: 3–0–2-0		

Prerequisites:

Proficiency in a C programming language.

Course Objectives:

This Course will enable students to:

1. **Understand** the basic approaches for analyzing and designing data structures.
2. **Introduce** dynamic memory allocation and C language concepts required for building data structures.
3. **Develop** essential skills to construct data structures to store and retrieve data quickly and **efficiently**.
4. **Utilize** different data structures that support different sets of operations which are suitable for various applications.
5. **Explore & implement** how to insert, delete, search, and modify data in any data structure-Stack, Queues, Lists, Trees.
6. **Develop** applications using the available data structure as part of the course for mini project.

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. ***Lecture method*** means it includes not only traditional lecture methods, but 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 analyse information rather than simply recall it.
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.

UNIT – I	**08 Hours**

IINTRODUCTION:

Introduction to Data Structure, Classification, C Structure and Union, C Pointers, Array Definition, Representation, Operations (Insertion, Deletion, Search and Traversal), Two/Multidimensional Arrays, Sparse matrix.

TB1: 1.1, 2.2, 2.5 ; TB2: 1.1, 1.2,1.3.1-1.3.4; RB1: 5.1 – 5.12, 6.4

UNIT – II	**08 Hours**

INTRODUCTION TO ADT:

Stack: Definition, Array Representation of Stack, Operations on Stacks.

Applications of Stack: Expression evaluation, Conversion of Infix to Postfix, Infix to Prefix Recursion, Tower of Hanoi **Queue:** Definition, Representation of Queues, Operations of Queues, Circular Queue. **Applications of Queue:** Job Scheduling, A Maze Problem TB1: 3.1,3.2, 3.3,3.4,3.5 ; TB2: 2.1, 2.2, 2.3, 3.2, 3.3
UNIT – III **08 Hours**
DYNAMIC DATA STRUCTURES: **Linked List:** Types, Representation of Linked Lists in Memory. Traversing, Searching, Insertion & Deletion from Linked List. Circular List, Doubly Linked List, Operations on Doubly Linked List (Insertion, Deletion, Traversal). **Applications:** Stack & Queue Implementation using Linked Lists. **Case Study**: Josephus problem. TB2: 4.2,4.3,4.5
UNIT – IV **08 Hours**
TREES: Basic Terminology, Binary Trees and their representation, Complete Binary Trees, Binary Search Trees, Threaded Binary Trees, Operations on Binary Trees (Insertion, Deletion, Search & Traversal). **Applications:** Expression Evaluation **Case Study:** Game Tree TB2: 5.5.3,5.5.4,5.6
UNIT – V **07 Hours**
Efficient Binary Search Trees: Optimal Binary Search Trees, AVL Trees, Red Black Trees, Splay Trees. **Case Study**: B Trees TB1: 10.1,10.2,10.3,10.4, 11.2

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Apply the concepts of pointers, arrays, structures, and unions to address real-world problems and implement the concept in C programming language.	L3
2	Utilize stacks and queue data structures to solve problems such as infix to postfix, infix to prefix conversions, the Towers of Hanoi puzzle, job scheduling and maze navigation.	L3
3	Implement and manipulate singly linked lists, doubly linked lists, and circular linked lists, executing operations such as insertion, deletion, and traversal.	L3
4	Understand the concepts of binary trees, binary search trees, and threaded binary trees, and their associated operations.	L2
5	Understand advanced binary tree structures includes optimal binary search trees, AVL trees, Red-Black trees, and Splay trees.	L2

Table: Mapping Levels of COs to POs / PSOs														
COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
CO1	1	2	3	-	3	-	-	-	-	-	-	-	2	2
CO2	1	2	3	-	3	-	-	-	-	-	-	-	2	2
CO3	1	2	3	-	3	-	-	-	-	-	-	-	2	2
CO4	-	1	2	-	3	-	-	-	-	-	-	-	2	2
CO5	-	1	2	-	-	-	-	-	-	-	-	-	2	2

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

TEXT BOOKS (TB):

1. Ellis Horowitz, Susan Anderson-Freed, and Sartaj Sahni, "Fundamentals of Data structures in C", 2nd Edition, Orient Longman, 2008.
2. A.M. Tannenbaum, Y Langsam, M J Augentien "Data Structures using C", 1st Edition, Pearson, 2019.

REFERENCE BOOKS:

1. Brian. W. Kernighan, Dennis. M. Ritchie, "The C Programming Language", 2nd Edition, Prentice-Hall, 1988.
2. Gilbert & Forouzan, "Data Structures: A Pseudo-code approach with C", 2nd Edition, Cengage Learning,2014.
3. Jean-Paul Tremblay & Paul G. Sorenson, "An Introduction to Data Structures with Applications", 2nd Edition, McGraw Hill, 2013.
4. R.L. Kruse, B.P. Learly, C.L. Tondo, "Data Structure and Program design in C", 5th Edition, PHI ,2009.

E-Resources:

1. https://nptel.ac.in/courses/106102064
2. https://www.coursera.org/learn/data-structures?specialization=data-structures-algorithms
3. https://www.udemy.com/topic/data-structures/free/
4. https://www.mygreatlearning.com/academy/learn-for-free/courses/data-structures
5. https://cse01-iiith.vlabs.ac.in/
6. https://kremlin.cc/k&r.pdf

Activity Based Learning (Suggested Activities in Class)

1. Real world problem solving using group discussion.
2. Role play E.g., Stack, Queue, etc.,
3. Demonstration of solution to a problem through programming.
4. Flip class activity E.g., arrays, pointers, dynamic memory allocation, etc.,

DATA STRUCTURES LABORATORY

Total Contact Hours: 26

Following are experiments to be carried out using either C programming language.

1. To Implement C programs to perform array operations.
2. To determine the validity of a 9x9 Sudoku board (application of 2-dimensional array).
3. To store, retrieve and update the elements in structures (structures and pointers to structures).
4. To implement stack using linked list.
5. To implement a queue data structure using a singly linked list.
6. To implement a singly linked list and its operations.
7. To implement a doubly linked list and its operations.
8. To create a circular queue using a circular linked list data structure
9. To implement binary tree traversal techniques.

OPEN-ENDED EXPERIMENTS

1. Design a web browser history tracker in C. Implement a stack data structure to keep track of visited URLs. Create functions to push new URLs onto the stack as users visit websites and pop URLs when users navigate backward in their browsing history.
2. Imagine you are responsible for designing a queue-based system to manage the queue of regular customers waiting to purchase cinema tickets at a popular movie theatre. Your system should ensure fair and efficient ticket sales for all customers. When a customer's arrive at the cinema, they join the queue. Each customer is represented by his name, age (for record-keeping), and number of tickets needed. When a customer reaches the front of the queue, they are served by the ticketing agent. Implement a ticket sale process where the agent provides the customer with the requested ticket(s). Initialize the total number of tickets and if the tickets are sold, then the ticketing agent should display a houseful message.

****

DIGITAL LOGIC DESIGN

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

SEMESTER – III

Subject Code	:	24CT2303	**Credits**	:	04
Hours / Week	:	03 Hours	**Total Hours**	:	39(T)+26(P) Hours
L–T–P–J	:	3–0–2–0			

Course Learning Objectives:

This Course will enable students to:

1. Translate the elements of digital logic functions to digital system abstractions using Verilog.
2. Illustrate simplification of Boolean expressions using Karnaugh
3. Model combinational logic circuits for arithmetic operations and logical operations
4. Analyze and model sequential elements flip-flops, counter, shift registers.
5. Outline the concept of Mealy Model, More Model and apply FSM to solve a given design problem

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*** means it includes not only traditional lecture method, but different *type 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, notetaking, 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. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I	**08 Hours**

INTRODUCTION:

Number System- Binary, Hexa, Decimal, Octal and its conversion. Canonical Notation - SOP & POS forms, Minimization of SOP and POS forms.

ARITHMETIC CIRCUITS AND VERILOG MODELLING

Adders: Half adder, full adder, Ripple carry adder, parallel adder /subtractor, fast adders-CLA, comparator- 2 bit. Simplification using K-Maps

Introduction to Verilog, Syntax of Verilog coding, Modelling styles in Verilog, Verilog Operators, Test bench for simulation

Text Book-1: Chapter 1: 1.2 to 1.4, Chapter 2: 2.6
Text Book-2: Chapter 5: 5.2, 5.3.3, 5.4,5.5.2, 5.5.3
Text Book-3: Chapter 1: 1.1, 1.2.2, 1.3.1, 1.3.2, 1.3.3, 1.4.2, 1.5.1.2, 1.5.2.2, 1.5.3.2, 1.5.4.2, 1.6.2

UNIT – II	**07 Hours**

Combinational Circuit Building Multiplexers 4:1, 8:1, decoders 3:8, 2:4, demultiplexers 1:4, encoders 8:3, 4:2, code converters- B to G and G to B- Simplification using K-Maps

Verilog for combinational circuits, if else, case-casex, casez, for loop, generate.

Text Book-2: Chapter 6: 6.1, 6.2, 6.3, 6.4, 6.6

UNIT – III	**08 Hours**

Sequential Circuits-1

Basic Latch, Gated latches, Flip Flops SR, D, JK, T, master-slave flip-flops JK, Characteristic equations, 0's and 1's Catching Problem, Race round condition, Switch debounce, shift registers- SISO, SIPO, PISO, PIPO, Setup time, Hold time, Propagation Delay **Text Book-2: Chapter 7: 7.1, 7.2,7.3, 7.4,7.5,7.6, 7.8**

UNIT – IV	**8 Hours**
Sequential Circuits-2 Binary counters – asynchronous and synchronous, mod-n counter, ripple counter- 4 bit. Verilog blocking and non-blocking, Mealy Model, Moore Model, State machine notation, Construction of Finite State Machine. **Text Book-2: Chapter 7: 7.9, 7.11, 7.12.3, 7.12.4, 8.1, 8.2, 8.3, 8.4**	

UNIT – V	**8 Hours**
Introduction to Electronic Design Automation: FPGA Design Flow, ASIC Design flow, architectural design, logic design, simulation, verification and testing, 3000 Series FPGA architecture. **Applications:** Design 4 Bit ALU, 7 Segment display, Vending Machine, 3 Pipeline. **Text Book-4: Chapter 1**	

Laboratory Experiments
Experiments are conducted using Verilog tool /Kits
1. Introduction to Xilinx tool, FPGA flow
2. Adder – HA, FA using data flow and behavior modelling styles
3. Adder – HA, FA using structural modelling style
4. Combinational designs – I (blocking and non-blocking/looping examples) **a.** Multiplexer: 4:1, 8:1 MUX. **b.** De Multiplexer: 1:4, 1:8 DEMUX.
5. Combinational designs – II (different types of case statements) **c.** Encoder with and without Priority: 8:3 and 4:2. **d.** Decoder: 3:8 and 2:4.
6. Design of 4-bit ALU
7. Flip Flop: D FF, T FF, JK FF
8. Design of Mod – n Up/Down Counter with Synchronous reset
9. Design of Mod – n Up/Down Counter with Asynchronous reset.
10. Design of Universal shift Register using FSM

Course Outcome	**Description**	**Bloom's Taxonomy Level**

At the end of the course the student will be able to:		
1	Interpret Boolean Expressions of digital design in simplified form	L2
2	Build the various elements of digital logic system with Verilog	L3
3	Construct Combinational and Sequential logic circuits	L3
4	Apply the hardware model of a digital system at different levels of abstraction in Verilog	L3
5	Apply the functionality of digital design by implementing on FPGA kits	L4
6	Build digital systems using FSM	L3

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
CO1	3		1										2	
CO2	3	1	1		1								2	
CO3	3	1	3										2	
CO4	3	1	2		1							1	2	
CO5	3	2	2		1								2	
CO6	3	2	2										2	

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

TEXT BOOKS:

1. M. Morris Mano Michael D. Ciletti , "Digital Design with an Introduction to the Verilog HDL", 6th Edition, Pearson Education, 2014.
2. Stephen Brown, Zvonko Vranesic, "Fundamentals of Digital Logic with Verilog design", McGraw Hill, 2014.
3. Nazein M. Botros, "HDL programming (VHDL and Verilog)", Dreamtech Press, 2006.
4. Douglas J Smith, "HDL Chip Design", Doone publications 1996.

REFERENCE BOOKS:

1. John M Yarbrough, "Digital Logic Applications and Design", Thomson Learning, 2014.
2. Donald D. Givone, "Digital Principles and Design", McGraw Hill, 2015.
3. Samir Palnitkar, "Verilog HDL: A Guide to Digital Design and Synthesis", Pearson Education, 2016.

E-Resources:

1. https://archive.nptel.ac.in/courses/106/105/106105165/

2. https://nptel.ac.in/courses/117105080

Activity Based Learning (Suggested Activities in Class)

1. Design problem solving and Programming using group discussion. E.g., Traffic light controller, Digital Clock, Elevator.
2. Demonstration of solution to a problem through simulation.

****

DISCRETE MATHEMATICS AND GRAPH THEORY

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

SEMESTER – III

Subject Code	:	24CT2304	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P–J	:	3–0–0–0			

Course Learning Objectives:

This Course will enable students to:

1. **Learn** the set theoretic concept and its application in theory of computation.
2. **Determine** the concepts of mathematical induction, recursive relations and their application.
3. **Illustrate** the association of functions, relations, partial ordered set and lattices with problems related to theoretical computer science and network models.
4. **Discuss** the basics of graph theory and its application in computer networks. Learn the concepts of counting techniques and its application.

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** means it includes not only traditional lecture method, but different type 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, notetaking, 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 analyse information rather than simply recall it.
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 possible, it helps improve the students' understanding.

UNIT – I	**08 Hours**

SET THEORY: Sets and subsets, Operations on Sets: Basic set operations, algebraic properties of sets, The Addition Principle

RELATIONS AND ITS PROPERTIES: Relations and their properties, N-Ary Relations and their applications, Representing relations.

Textbook – 2: 1.1, 1.2 ; Textbook – 1: 7.1., 7.2, 7.3

UNIT – II	**06 Hours**

RELATIONS AND ORDER RELATIONS: Closure of relations, Equivalence Relations, Partial Orderings, Functions, The Growth of Functions.

Self-Study: Transitive Closure and Warshall's Algorithm.

Textbook – 1: 7.4., 7.5, 7.6, 3.2

UNIT – III	**08 Hours**

MATHEMATICAL INDUCTION AND RECURSION: Mathematical Induction, Recurrence Relations: Rabbits and the Fibonacci Numbers, The Tower of Hanoi, Code word Enumeration, Solving Linear Recurrence Relations **Self-Study:** Basic Connectives and Truth Tables ***Textbook-1: 4.1;6.1, 6.2;1.1***
UNIT – IV **09 Hours**
GRAPH THEORY: Graphs and Graph Models. Graph Terminology and Special Types of Graphs: Basic Terminology, Some Special Simple Graphs, Bipartite Graphs, Complete Bipartite Graphs. Representing Graphs and graph isomorphism: Adjacency lists, Adjacency Matrices, Incidence Matrices, Connectivity: Paths, Connectedness in Undirected and Directed Graphs, Vertex and Edge connectivity and their applications. ***Textbook-1: 8.1, 8.2, 8.3, 8.4***
UNIT – V **08 Hours**
GRAPHS AND ITS APPLICATIONS: Euler and Hamilton Paths and their applications, Planar Graphs and their Applications, Graph Coloring and its applications. ***Textbook-1: 8.5, 8.7, 8.8***

Course Outcome	**Description**	**Bloom's Taxonomy Level**
At the end of the course the student will be able to:		
1	**Identify** the membership of the Set, Relations and make use of basic Algebraic properties.	L3
2	**Examine** the steps involved in Mathematical Induction and Linear recurrence-related problems.	L4
3	**Construct** different types of graphs based on the properties and the real-time applications of graph theoretical concepts.	L3
4	**Analyze** the methods for optimizing the solution for graph coloring problem, Eulerian and Hamilton circuits/planes.	L4

Table: Mapping Levels of COs to POs / PSOs

COs	**Program Outcomes (POs)**												**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
CO1	3	1	2					1	1	1		2	1	0
CO2	3	3	2					1	1	1		2	1	0
CO3	3	3	3					1	1	1		1	1	0
CO4	3	3	3					1	1	1		1	1	0

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

TEXT BOOKS:

1. Kenneth H. Rosen, "Discrete Mathematics and its applications", Tata McGraw Hill, 2003.
2. Bernard Kolman, Robert C. Busby, Sharon Ross, "Discrete Mathematical Structures", 3rdEdition, PHI 2001.

REFERENCE BOOKS:

1. Ralph P. Grimaldi, "Discrete and Combinatorial Mathematics", IV Edition, Pearson Education, Asia, 2002.
2. J. P. Tremblay, R. Manohar, "Discrete Mathematical Structures with applications to computer Science", Tata McGraw Hill, 1987.
3. J K Sharma, "Discrete Mathematics", 3rd edition, 2013, Macmillan India Ltd.

E-Resources:

1. Discrete Mathematics with Algorithms by M. O. Albertson, J. P. Hutchinson – J 1988, Wiley.
2. Discrete Mathematics for Computer Science, Gary Haggard, John Schlipf, Sue Whitesides,Thomson Brooks/Cole, 2006.
3. http://ocw.mit.edu/courses/mathematics/
4. http://www.nptelvideos.in/2012/11/discrete-mathematical-structures.html
5. http://cglab.ca/~discmath/notes.html
6. https://www.cs.odu.edu/~toida/nerzic/content/web_course.html

Activity Based Learning (Suggested Activities in Class)

1. Real world problem solving and puzzles using group discussion.
2. Demonstration of solution to a problem using graph theory.

****

FULL STACK DEVELOPMENT

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

SEMESTER – III

Course Code	:	24CT2305	**Credits**	:	04
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P–J	:	3–0–0–2			

Course Learning Objectives:

This Course will enable students to:

1. Develop proficiency in using HTML5 to create well-structured, semantically correct web pages. Incorporate multimedia elements, forms, and hyperlinks to enhance interactivity and accessibility.
2. Apply advanced CSS3 techniques to style web pages, ensuring they are visually appealing and responsive. Master the CSS Box Model, selectors, and new CSS3 features such as shadows, opacity, and rounded corners.
3. Gain a thorough understanding of JavaScript programming to create dynamic and interactive web pages. Focus on syntax, control structures, object-oriented features, and practical applications like game development and DOM manipulation.
4. Acquire skills to set up a Node.js development environment and utilize Node.js modules and NPM. Implement server-side functionality, including file system operations and database connectivity using MongoDB.
5. Develop the ability to create robust web applications using Express.js, focusing on middleware usage. Handle various types of requests and responses, and integrate a basic React.js front-end with RESTful APIs to build a complete web application.

Teaching-Learning Process (General Instructions)

1. These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.
2. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
3. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, focused listening, and formulating questions.
4. To make ***Critical thinking***, ask at least three higher order Thinking questions in the class.
5. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
6. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them
7. Discuss how some ***concept can be applied to the real world*** - and when that's possible.

UNIT – I	**07 Hours**

Markup Language (HTML5): Introduction to HTML and HTML5 - Formatting and Fonts - Commenting Code – Anchors – Backgrounds – Images – Hyperlinks – Lists – Tables – HTML Forms, Audio, Video Tag

(Text Book 1- Chapter-2)

UNIT – II	**08 Hours**

CSS3: Levels of style sheets; Style specification formats; Selector forms; Property value forms; Font properties; List properties; Color; Alignment of text; Background images, Conflict Resolution, CSS Box Model .CSS3 features: Box Shadow, Opacity, Rounded corners, Attribute selector.

(Text Book 1- Chapter-3)

UNIT – III	**08 Hours**

JavaScript: Overview of JavaScript; Object orientation and JavaScript; General syntactic,

characteristics; Primitives, operations, and expressions; Screen output and keyboard input. Control statements; Arrays; Functions, Constructors; A brief introduction on pattern matching using regular expressions, DOM Events, JSON , Case Study -game development **(Text Book 1- Chapter-4,5)**
UNIT – IV **08 Hours**
Node JS: Introduction to NodeJS, Set up Dev Environment, Node JS Modules, Node Package Manager, File System, Events, Database connectivity using Mongo DB. **(Text Book 2- Chapter-1,Chapter 8.2,8.3,8.4)**
UNIT – V **08 Hours**
Express.JS: Introducing Express: Basics of Express, Express JS Middleware: Serving Static Pages, React JS basic, REST API, Your first React Web Application. **(Text Book 2- Chapter-7.1,7.4)**

Course Outcome	Description	Bloom's Taxonomy Level
1	Use the common HTML5 elements(tags) to **develop the** static web pages.	L3
2	**Make use of** Cascading Style Sheets with HTML5 elements for visual presentation and design well-structured web pages.	L3
3	**Implement** the JavaScript programming concepts to develop client-side scripts and display the contents dynamically.	L3
4	**Develop** dynamic server-side applications by employing Node.js event-driven, non-blocking I/O model and its integration with Mongo dB to work with dynamic schemas.	L3
5	Utilize basic concepts React.js and Express.js framework, to **construct** REST web applications.	L3

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
CO1	2	2	2		3				1			2	2	
CO2	2	2	2		3				1			2	2	
CO3	2	2	2		3				2			2	2	
CO4	2	2	2		3				2			2	2	
CO5	2	2	2		3				2			2	2	

TEXT BOOKS:

1. Robert W. Sebesta , "Programming the World Wide Web", 7th Edition, Pearson Education, 2008.
2. Basarat Ali Syed," Beginning Node.js ",Apress ,2014

REFERENCES:

1. Lionel Lopez,React Quickstart Step-by-Step Guide to Learning React Javascript Library
2. Kirupa Chinnathambi, JavaScript Absolute Beginner's Guide, 1st Edition, 2017.
3. Robert W Sebesta, Pearson, Programming the World Wide Web, 7th Edition, 2013.
4. Kirupa Chinnathambi, Learning React, 1 Edition, Addison-Wesley Professional
5. Mark Pilgrim,HTML5Up and Running,O'Reilly, 1st Edition, 2012.

E-Resources:

1. https://www.edureka.co/blog/ebook/web-development-ebook
2. MOOC:https://www.coursera.org/learn/server-side-javascript-with-nodejs

Activity Based Learning (Suggested Activities in Class)

1. Hands-on Lab exercises
2. Mini Project
3. Freecode camp/OpenSource Online Coding Boot Camp
4. Coding platform Challenges – Hackerrank, CodeChef
5. Case based learning

****

DESIGN THINKING AND IDEATION

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

SEMESTER – III

Subject Code	:	24CT2306	**Credits**	:	02
Hours / Week	:	01 Hours	**Total Hours**	:	26 Hours
L–T–P-J	:	1–0–0-2			

Course Learning Objectives:

This Course will enable students to:

1. **Analyze** the historical and cultural context of design thinking.
2. **Generate** innovative ideas using various ideation techniques.
3. **Develop** an empathetic and user-centric mindset.
4. **Analyze** and evaluate case studies and real-world examples of successful design thinking and idea generation projects to extract insights and best practices.

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*** means it includes not only traditional lecture method, but different *type 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, notetaking, 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 analyse information rather than simply recall it.
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.

UNIT – I	**07 Hours**

INTRODUCTION:

Introduction to Design Thinking, Stages, Understanding the Mindsets-Empathy, Optimism, Embrace Ambiguity, Make it, Learn from Failure, Iterate, Create Confidence, Creativity Convergent & Divergent Thinking.

METHODS AND TOOLS FOR EMPATHIZE PHASE:

Methods of Empathize phase: Ask 5 Why/ 5W +H questions, Stakeholders map, Empathy Map, Peer observation, Trend analysis.

Case study:How to apply Empathize tool and create a design for solving real world problems.

UNIT – II	**07 Hours**

METHODS AND TOOLS FOR DEFINE PHASE:

Methods of Define Phase: Storytelling, Critical items diagram, Define success.

Case study: How to leverage methods & tools in the define phase for real world applications.
Methods and Tools for Ideate Phase:
Ideate- Brainstorming, 2x2 matrix, NABC method
Activity: Generate innovative idea and present based on the methods/ tools learnt so far.

UNIT – III	**07 Hours**

Methods and Tools for Prototype & Test phase:
Protypes- Types of prototypes, Methods of prototyping- focused experiment, Exploration map, Minimum Viable Product, Methods of testing, Feedback capture grid. A/B testing

Activity: Iterate and improve the idea(s) using the prototype and testing -methods & tools.

UNIT – IV	**05 Hours**

Create a pitch, plan for scaling up, Road map for implementation. Design Thinking to Business Process modelling.

Course Outcome	**Description**	**Bloom's Taxonomy Level**
At the end of the course the student will be able to:		
1	**Utilize** the core concepts of design thinking stages, including Mindsets- Empathy, Optimism, Embrace Ambiguity, Make it, Learn from Failure, Iterate, Create Confidence, Creativity Convergent & Divergent Thinking to showcase the application of empathize phase methods and tools.	L2
2	**Implement** define tools to frame the problem statement and Ideate tools to generate idea for solving the problem.	L3
3	**Develop** proficiency in prototyping and testing as essential steps in creating successful and user-centered products or solutions.	L6
4	**Apply** effective pitching skills and roadmap for business process modelling.	L3

Table: Mapping Levels of COs to POs / PSOs

COs	**Program Outcomes (POs)**												**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
CO1	3	3	3	3	3							2		3
CO2	3	3	3	3	3	3			3		3	2		3
CO3	3	3	3	3	3	3			3		3	2		3
CO4	3	3	3	3	3	3			3		3	2		3

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

TEXT BOOKS:

1. IdrisMootee," Design Thinking for Strategic Innovation",2013, John Wiley & Sons Inc

REFERENCE BOOKS:

1. Brenda Laurel, "Design Research methods and perspectives", MIT press 2003
2. Terwiesch, C. & Ulrich, K.T., 2009. "Innovation Tournaments: creating and identifying Exceptional Opportunities", Harvard business press.
3. Ulrich &Eppinger, "Product Design and Development", 3rd Edition, McGraw Hill, 2004
4. Stuart Pugh, "Total Design: Integrated Methods for Successful Product Engineering, BjarkiHallgrimsson, Prototyping and model making for product design", 2012, Laurence King Publishing Ltd

E-Resources:

1. https://www.mygreatlearning.com/academy/learn-for-free/courses/design-thinking
2. https://www.edx.org/learn/design-thinking
3. https://www.coursera.org/learn/design-thinking-innovation
4. https://alcorfund.com/insight/ideation-tools-purpose-methods/
5. https://onlinecourses.swayam2.ac.in/aic19_de04/preview

****

INNOVATIVE BUSINESSES & BREAKTHROUGH TECHNOLOGIES

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

SEMESTER – III

Subject Code	:	24CT2307	**Credits**	:	02
Hours / Week	:	01 Hours	**Total Hours**	:	26 Hours
L–T–P-J	:	1–0–0-2			

Course Learning Objectives:

This Course will enable students to:

1. To equip students with the skills and knowledge needed to apply breakthrough technologies strategically in business innovation.
2. To enable students to comprehend the nature, impact, and potential of breakthrough technologies

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*** means it includes not only traditional lecture method, but different *type 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, notetaking, 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 analyse information rather than simply recall it.
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.

CONTENTS	**13 HRS**
The User Innovation Paradigm, Going for User Solutions Instead of User Needs – the Lead User Method, How Patents Discourage Innovation User Innovation Communities, Exploring New Combinations of Customer Needs, Determine Users' Needs Ethnographically and Develop Solutions MIT Media Lab Approach, Design and Manufacture of Mass Customized Products with Toolkits/Platforms for User Innovation	

Hands-on Component	**13 HRS**
1. Brainstorming as a Concept Generation Method 2. Case Studies on Problem-Solving about Analogical Thinking	

Course Outcome	**Description**	**Bloom's Taxonomy Level**
At the end of the course the student will be able to:		
1	Discuss innovative solutions by combining diverse customer needs in producing products or services with unique prepositions.	L2

2	Utilize design product tool kits or platforms to the production of customized products that meet individual user preferences and demands.	L3

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
CO1	2	2	3	3	3	2						2		3
CO2	2	2	3	3	3	2						2		3

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

TEXT BOOKS:

1. MIT Open Course Ware – How to Develop Breakthrough Products and Services

REFERENCE BOOKS:

1. Eric Von Hippel, Democratizing Innovation, The MIT Press, February 2006
2. Eric Von Hippel, Stefan Thomke and Mary Sonnack, Creating Breakthroughs at 3M, Harvard Business Review
3. Glen L. Urban, John R. Hauser, "Listening In" to Find & Explore New Combinations of Customer Needs, Journal of Marketing, Sage Journals, 2004

E-Resources:

1. https://ocw.mit.edu/courses/sloan-school-of-management/15-356-how-to-develop-breakthrough-products-and-services-spring-2012/index.htm

****

R FOR DATA ANALYSIS AND APPLICATION DEVELOPMENT

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

SEMESTER – V

Subject Code	:	24CT2308	**Credits**	:	02
Hours / Week	:	03 Hours	**Total Hours**	:	26 Hours
L–T–P–S	:	1–0–0–2			

Course Learning Objectives:

This Course will enable students to:

1. **Acquire** the concept of data science, data analysis.
2. **Develop** the data analysis graphs and charts using data visualization tools and machine learning with statistics.
3. **Acquire** the basics of R, simulation and usage of built in functions, file processing and graphics programming in applications
4. **Understand** and **acquire** skill on T-Testing and linear optimization designing.
5. **Develop** confidence and efficient writing of R scripts.

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*** means it includes not only traditional lecture method, but different *type 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, notetaking, 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 analyse information rather than simply recall it.
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.

UNIT – I	**5 Hours**
INTRODUCTION: how to run R, R Sessions and Functions, Basic Math, Variables, Data Types, Vectors, Advanced Data Structures, Data Frames, Lists, Matrices, Arrays, Classes. ***(Text Book-1: Chapter 1: 1.1 to 1.4, 1.6, Chapter 2: 2.1,2.2, 2.4)***	
UNIT – II	**5 Hours**
R Programming Structure: Control Statements, Loops, - Looping Over Non-Vector Sets, If-Else, Arithmetic and Boolean Operators, Default Values for Argument, Return Values, Functions with No Pointers in R, Recursion, Sorting and Searching ***(Text Book-1: Chapter 3: 3.1 to 3.3).***	
UNIT – III	**5 Hours**
Plots and Graphs: Graphics, Creating Graphs, The Workhorse of R Base Graphics, the plot () Function Customizing Graphs,Saving Graphs to Files. (***Text Book-2: Chapter 8: 8.1 tow 8.8)***	
UNIT-IV	**5 Hours**
T-Testing: Probability Distributions, Normal Distribution- Binomial Distribution- Poisson Distributions, BasicStatistics, Correlation and Covariance, T-Tests, -ANOVA. ***(Text Book-2: Chapter 10: 10.1 to 10.6)***	

UNIVT-V	6 Hours
ML Model Implementation: Linear Models, Simple Linear Regression and Multiple Regression, Generalized Linear Models, Nonlinear Models, Splines- Decision- Random Forests. (***Text Book-2: Chapter 12: 12.1 to 12.9)***	

Course Outcomes:

At the end of the course the student will be able to:

1. **Understand** the fundamentals, standards of Functions and capabilities of R- Language to **develop** data analytics applications.
2. **Formulate** Learning the basic R-Language Constructs and **apply** it to **build** basic educationalapplications.
3. **Demonstrate** the simulation in R-Language, Math functions and files Processing to read and analysis health care data.
4. **Apply** the Principals of Graphics and R-Base Graphics to plot the real time health care applicationsdata.
5. **Develop** basic health care applications and Performing T-Testing, and **Design** and **build** Linear optimization of health care data by using R-Language concepts.

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
CO1	3	-	-	-	3	2	-	-	2	2	-	2	-	3
CO2	2	-	2	-	2	-	-	-	-	3	-	-	-	-
CO3	2	-	-	-	-	-	-	-	-	3	-	1	-	2
CO4	2	-	1	-	3	-	-	-	-	3	-	2	-	-
CO5	3	-	-	-	2	-	-	-	3	3	-	1	-	2

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

TEXT BOOKS:

1. **The Art of R Programming, Norman Matloff, Cengage Learning: Efficient R Programming: A Practical Guide to Smarter Programming 1st Edition - Colin Gillespie & Robin Lovelace- First Edition.**

2. **Jared P. Lander, R for Everyone: Advanced A nalytics and Graphics, Second Edition 2017.**

REFERENCE BOOKS:

1. **R Cookbook, PaulTeetor, Oreilly: R Cookbook [R CKBK] [Paperback] R Cookbook[RKBK] [Paperback]Mar 31, 2011 by Paul Teetor.**

2. **R in Action,Rob Kabacoff, Manning: R in Action: Data Analysis and Graphics withRNov 5,2018 | Unabridged by Robert Kabacoff and Dale Ogden**

E-Resources:

1. http://nptel.ac.in/courses
2. https://www.listendata.com/p/r-programming-tutorials.html#:~:text=Hands-on%20R%20Programming%20Tutorials%201%20Best%20R%20Tutorials,R%204%20Shiny%20Tutorials%205%20R%20Interview%20Questions
   http://agce.sets.edu.in/cse/ebook/DBMS%20BY%20RAGHU%20RAMAKRISHNAN.pdf

Activity Based Learning (Suggested Activities in Class)

1. Data analysis and data exploration using group discussion.
2. Collaborative Activity is minor project development with a team of 4 students.

****

PRODUCT MARKETING

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

SEMESTER – III

Subject Code	:	24CT2309	**Credits**	:	02
Hours / Week	:	03 Hours	**Total Hours**	:	26 Hours
L–T–P–S	:	1–0–0–2			

Course Learning Objectives:

This Course will enable students to:

1. To put the students on the cutting edge of knowledge in making strong and actionable recommendations to managements in the field of marketing.

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*** means it includes not only traditional lecture method, but different *type 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, notetaking, 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 analyse information rather than simply recall it.
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.
   Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I: **26 Hours**

Introduction to Product Marketing, Visual representation of Product Marketing, building a Customer Persona, Product-Market Fit, Product Life Cycle, Product Marketing vs Regular Marketing, Responsibilities of a Product Marketer, Critical steps of Product Marketing, Measuring the impact of Product Marketing.

Resources:

Video: https://olympus.mygreatlearning.com/courses/91026/pages/introduction-to-product-marketing?module_item_id=3776737

Course Outcomes:

At the end of the course the student will be able to:

1. Develop a deep understanding of product marketing fundamentals, including its role in bridging product development and market demand.
2. Analyze data performance and make data-driven decisions to optimize marketing strategies and achieve better results.

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
CO1	3	3	3				3							2
CO2				3	2					2		2		2

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

REFERENCE BOOKS:

1. Philip Kotler, Kevin Lane Keller, Abraham Koshy and Mithileswar Jha, Marketing Management, Pearson, 14th Edition, 2013.

E-Resources:

1. https://learning.productmarketingalliance.com/on-demand/advanced-product-marketing- certified
2. https://learning.productmarketingalliance.com/leadership-on-demand/product-marketing- certified-leadership#course-overview

Activity Based Learning (Suggested Activities in Class)

1. Real-world case studies of successful and unsuccessful product launches.
2. Brainstorm different messaging strategies for various customer segments.

****

PROBABILITY AND STATISTICS

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

SEMESTER – IV

Course Code	:	24CT2401	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P–J	:	3–0–0–0			

Course Learning Objectives:

This Course will enable students to:

1. **Apply** statistical principles and probability concepts to solve complex problems in real-world scenarios involving uncertainty and randomness.
2. **Evaluate** and select appropriate probability distributions and statistical techniques to analyze and interpret data accurately in various applications.
3. **Justify** the use of estimation methods and hypothesis testing techniques for drawing meaningful inferences about population parameters.
4. **Analyze** and interpret sample test results for different statistical relationships, such as means, variances, correlation coefficients, regression coefficients, goodness of fit, and independence, to make informed decisions.
5. **Identify** sample tests using appropriate statistical procedures to investigate the significance of observed data and communicate findings effectively.

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*** means it includes not only traditional lecture method, but different *type 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, notetaking, 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 analyse information rather than simply recall it.
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.

UNIT – I : Probability	**09 Hours**
Definitions of Probability, Addition Theorem, Conditional Probability, Multiplication Theorem, Bayes' Theorem of Probability	
UNIT – II: Random Variables and their Properties and Probability Distributions	**09 Hours**
Discrete Random Variable, Continuous Random Variable, Joint Probability Distributions Their Properties, Probability Distributions: Discrete Distributions: Binomial, Poisson Distributions and their Properties; Continuous Distributions: Exponential ,Normal, Distributions and their Properties.	
	06 Hours

UNIT – III: Estimation and testing of hypothesis	
Sample, Populations, Statistic, Parameter, Sampling Distribution, Standard Error, Un-Biasedness, Efficiency, Maximum Likelihood Estimator, Notion & Interval Estimation.	
UNIT – IV: Sample Tests-1	**07 Hours**
Large Sample Tests Based on Normal Distribution , Small Sample Tests : Testing Equality of Means, Testing Equality of Variances, Test of Correlation Coefficient	
UNIT – V: Sample Tests-2	**08 Hours**
Test for Regression Coefficient; Coefficient of Association, 2 – Test for Goodness of Fit, Test for Independence.	

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	**Apply** the principles of probability to solve complex problems in various real-world scenarios.	L2 & L3
2	**Solve** and **compare** different probability distributions, including discrete and continuous random variables, in order to make informed decisions and predictions.	L2 & L3
3	**Apply** statistical estimation techniques, such as maximum likelihood estimation and interval estimation, to draw meaningful inferences about population parameters from sample data.	L3
4	**Examine** hypothesis testing methods, including large and small sample tests, to assess the significance of observed data and draw valid conclusions.	L4
5	**Analyze** statistical relationships and perform sample tests to assess the Equality of means in different populations, Correlation coefficients between variables to determine the strength and direction of the relationship. Independence of variables using appropriate statistical tests to assess the absence of any relationship.	L4

Table: Mapping Levels of COs to POs / PSOs														
COs	**Program Outcomes (POs)**												**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
CO1	3	2	2		2				1					
CO2	3	2	2		2				1					
CO3	3	2	2						1					
CO4	3	2	2		2				1					
CO5	3	2	2		2				1					

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

TEXT BOOKS:

1. Probability & Statistics for Engineers and Scientists, Walpole, Myers, Myers, Ye. Pearson Education.

REFERENCE BOOKS:

1. Probability, Statistics and Random Processes T. Veerarajan Tata McGraw – Hill
2. Probability & Statistics with Reliability, Queuing and Computer Applications, Kishor S. Trivedi, Prentice Hall of India ,1999

E-Resources:

1. https://nptel.ac.in/courses/106104233
2. https://nptel.ac.in/courses/117103067
3. https://nptel.ac.in/courses/103106120
4. https://www.coursera.org/learn/probability-intro#syllabus
5. https://nptel.ac.in/courses/111104073

Activity Based Learning (Suggested Activities in Class)

1. Tools like Python programming, R programming can be used which helps student to develop a skill to analyze the problem and providing solution.
2. Regular Chapter wise assignments/ Activity/Case studies can help students to have critical thinking, developing an expert mind set, problem-solving and teamwork.

Following are Assignments/ Activities Can be carried out using either R programming language or Python Programming or excel solver.

1. There are n people gathered in a room. What is the probability that at least 2 of them will have the same birthday? (Use excel solver, R Programming, Python Programming)

a. Use simulation to estimate this for various n., and Produce Simulation Graph.
b. Find the smallest value of n for which the probability of a match is greater than 0.5.
c. Explore how the number of trials in the simulation affects the variability of our estimates.

2. **Case Study 1: Customer Arrivals at a Coffee Shop**
   a. A coffee shop wants to analyze the number of customer arrivals during its morning rush hour (7:00 AM to 9:00 AM). The shop has been recording the number of customer arrivals every 15 minutes for the past month.
   b. Data: The data consists of the number of customer arrivals recorded at the coffee shop during each 15-minute interval for the past month.
   c. Here is a sample of the data:

Time Interval	**stomer Arrivals**
00 AM - 7:15 AM	**6**
15 AM - 7:30 AM	**4**
30 AM - 7:45 AM	**9**
45 AM - 8:00 AM	**7**
00 AM - 8:15 AM	**5**
15 AM - 8:30 AM	**8**
30 AM - 8:45 AM	**10**
45 AM - 9:00 AM	**6**

analyze the customer arrivals and determine the probability distribution that best fits the data. Specifically, explore both discrete and continuous probability distributions, including the binomial, Poisson, exponential, and normal distributions.

3. **Case Study 2: Comparing the Performance of Two Groups**
   a. Suppose you are a data analyst working for a company that manufactures a new energy drink. The marketing team conducted a promotional campaign in two different cities (City A and City B) to determine the effectiveness of the campaign in increasing sales. The sales data for a random sample of customers in each city was collected over a week. Your task is to compare the average sales between the two cities and test whether there is a significant difference in the variance of sales.
   b. **Data**: Let's assume the following sample data for the number of energy drinks sold in each city:

City A: [30, 28, 32, 29, 31, 33, 34, 28, 30, 32]
City B: [25, 24, 26, 23, 22, 27, 29, 30, 26, 24]
perform a two-sample t-test to test the equality of means and a test for equality of variances using Python's SciPy library.

4. **Case study 3:** testing independence between two categorical variables.

a. Data: Sample of 100 employees, and each employee is classified as either Male or Female. They were asked to rate their job satisfaction on a scale of 1 to 5, where 1 represents low satisfaction and 5 represents high satisfaction. The data is as follows:

Employee	**Gender**	**Job Satisfaction**
1	Male	4
2	Female	3
3	Male	2
4	Female	5
...	...	...
100	Female	4

b. Test for independence between gender and job satisfaction, use the chi-squared test in R.

****

DESIGN AND ANALYSIS OF ALGORITHMS

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

SEMESTER – IV

Subject Code	:	24CT2402	**Credits**	:	04
Hours / Week	:	03 Hours	**Total Hours**	:	39(L) + 26(P) Hours
L–T–P-J	:	3–0–2-0			

Course Learning Objectives:

This Course will enable students to:

1. **Analyze** the non-recursive and recursive algorithms and to represent efficiency of these algorithms in terms of the standard Asymptotic notations.
2. **Devise** the Brute Force and Divide and Conquer techniques to design the algorithms and apply these methods in designing algorithms to solve a given problem.
3. **Explain** the Decrease and Conquer, Transform and Conquer algorithm design techniques, and Time versus Space Trade-offs.
4. **Get the idea** of Greedy method and dynamic programming methods and apply these methods in designing algorithms to solve a given problem.
5. **Describe** and illustrate the idea of Backtracking and Branch and Bound algorithm design techniques to solve a given problem.

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. ***Lecture method*** means it includes not only traditional lecture methods, but 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 analyse information rather than simply recall it.
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.

UNIT – I	**08 Hours**

INTRODUCTION:

What is an Algorithm? Fundamentals of Algorithmic Problem Solving. ***(Text Book-1: Chapter 1: 1.1 to 1.2)***

FUNDAMENTALS OF THE ALGORITHMS EFFICIENCY:

Analysis Framework, Asymptotic Notations and Standard notations and common functions ***(Text Book-2: Chapter 3: 3.1, 3.2)***,

Mathematical Analysis of Non-recursive and Recursive Algorithms,

(Text Book-1: Chapter 2: 2.1, 2.3, 2.4,)

UNIT – II	**08 Hours**
BRUTE FORCE: Background, Selection Sort, Brute-Force String Matching. ***(Text Book-1: Chapter 3: 3.1, 3.2)*** **DIVIDE AND CONQUER:** General method, Recurrences: The recursion-tree method, The master method. ***(Text Book-2: Chapter 4: 4.4, 4.5***), Merge sort, Quick sort, Binary Search, Multiplication of large integers, **Case study:** Strassen's Matrix Multiplication. ***(Text Book-1: Chapter 4: 4.1 to 4.3, 4.5)***	
UNIT – III	**06 Hours**
DECREASE & CONQUER: General method, Insertion Sort, Graph algorithms: Depth First Search, Breadth First Search, Topological Sorting **TRANSFORM AND CONQUER:** **Case study:** Heaps and Heap sort. **TIME AND SPACE TRADEOFFS:** Input Enhancement in String Matching: Horspool's algorithm, Hashing: Open and Closed hashing. ***(Text Book-1: Chapter 5: 5.1 to 5.3, Chapter 6: 6.3 to 6.4, Chapter 7: 7.2 to 7.3)***	
UNIT – IV	**09 Hours**
GREEDY TECHNIQUE: General method of Greedy technique, Single-Source Shortest Paths: General method, The Bellman-Ford algorithm, Single-Source Shortest Paths in DAGs, Dijkstra's Algorithm ***(Text Book-2: Chapter 24: 24.1 to 24.3).*** Minimum Spanning Trees: Prim's Algorithm, Optimal Tree problem: Huffman Trees; **Case study:** Kruskal's Algorithm. ***(Text Book-1: Chapter 9: 9.1, 9.2, 9.4).*** **DYNAMIC PROGRAMMING:** General method, The Floyd-Warshall Algorithm, Johnson's algorithm for sparse graphs ***(Text Book-2: Chapter 25: 25.1 to 25.3)***, The Knapsack problem ***(Text Book-1: Chapter 8: 8.4).***	
UNIT – V	**08 Hours**
LIMITATIONS OF ALGORITHMIC POWER P, NP and NP-complete problems ***(Text Book-1: Chapter 11: 11.3***) **BACKTRACKING:** General method, N-Queens problem, Subset-sum problem. ***(Text Book-1: Chapter 12: 12.1***) **BRANCH AND BOUND:** General method, Travelling Salesman problem, Approximation algorithms for TSP. **Case study:** Knapsack Problem. ***(Text Book-1: Chapter 12: 12.2, 12.3***)	

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Analyze the given recursive and non-recursive algorithms for time using step count, substitution method, recurrence tree method and mathematical analysis method. Represent the complexity of the algorithm using asymptotic notation.	L3
2	Solve sorting, searching, matrix multiplication problems based on divide and conquer design technique and implement in C programming language.	L3
3	Apply algorithms for graph-based problems (DFS, BFS and Topological sorting) using decrease and conquer design techniques. Distinguish the trade-offs between space and time complexity.	L3
4	Apply algorithms for finding the shortest path and minimum spanning tree for a given graph using greedy and dynamic programming techniques and implement the 0/1 knapsack problem in C programming language.	L3
5	Apply an efficient algorithm to solve N-Queens problem, Subset-sum problem, Knapsack and Traveling salesman problem using branch and bound and backtracking design technique. Describe the limitation of algorithmic power in terms of P, NP, NP hard and NP complete categorizations.	L3
6	Implement the graph-based algorithms including DFS, BFS, Warshall's Algorithm, Floyd's Algorithm, Kruskal's Algorithm, and Dijkstra's Algorithm in C programming language. (**Lab Experiments**)	L3

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
CO1	2	3	-	-	-	-	-	-	-	1	-	2	3	-
CO2	2	-	-	-	3	-	-	-	-	1	-	2	3	-

CO3	2	2	3	3	-	-	-	-	-	1	-	2	2	-
CO4	2	2	3	-	3	-	-	-	-	1	-	2	3	-
CO5	2	2	3	-	-	-	-	-	-	1	-	2	2	-
CO6	2	2	3	-	3	-	-	-	-	1	-	2	3	-

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

TEXT BOOKS:

1. Anany Levitin, "Introduction to the Design & Analysis of Algorithms", 2nd Edition, Pearson Education, 2011.
2. Thomas H. Cormen, Charles E.Leiserson, Ronald L. Rivest, Clifford Stein, "Introduction to Algorithms", 3rd Edition, PHI, 2014.

REFERENCE BOOKS:

1. Horowitz E., Sahni S., Rajasekaran S, "Computer Algorithms", Galgotia Publications, 2001.
2. R.C.T. Lee, S.S. Tseng, R.C. Chang & Y.T.Tsai, "Introduction to the Design and Analysis of Algorithms: A Strategic Approach", Tata McGraw Hill, 2005.

E-Resources:

1. https://nptel.ac.in/courses/106/101/106101060/
2. http://cse01-iiith.vlabs.ac.in/
3. http://openclassroom.stanford.edu/MainFolder/CoursePage.php?course=IntroToAlgorithms
4. https://www.coursera.org/specializations/algorithms

Activity Based Learning (Suggested Activities in Class)

1. Real world problem solving and puzzles using group discussion. E.g., Fake coin identification, Cabbage puzzle, Konigsberg bridge puzzle etc.,
2. Demonstration of solutions to a problem through programming.

DESIGN AND ANALYSIS OF ALGORITHMS LABORATORY

Total Contact Hours: 26

Following are experiments to be carried out using either C programming language or Object-oriented programming language:

1. Implementation of Binary Search algorithm.
2. Sort a given set of n integer elements using the Merge Sort method and compute its time complexity. Demonstrate this algorithm using the Divide-and-Conquer method.

3. Sort a given set of n integer elements using Quick Sort method and compute its time complexity. Demonstrate this algorithm using the Divide-and-Conquer method.
4. Incorporate the array data structure and demonstrate whether a given unweighted graph is connected or not using the DFS method.
5. Implement the graph traversal technique using the BFS method to print all the nodes reachable from a given starting node in an unweighted graph.
6. Compute the Transitive Closure for a given directed graph using Warshall's algorithm.
7. For a given weighted graph, construct an All-Pairs Shortest Paths problem using Floyd's algorithm and implement this algorithm to find the shortest distance and their shortest paths for every pair of vertices.
8. Implement 0/1 Knapsack problem using Dynamic Programming Memory Functions technique
9. Find Minimum Cost Spanning Tree for a given weighted graph using Prim's and Kruskal's algorithm.
10. From a given vertex in a weighted connected graph, determine the Single Source Shortest Paths using Dijkstra's algorithm.
11. Implement N-Queens problem using Backtracking technique.
12. Case Study

****

DATABASE MANAGEMENT SYSTEMS

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

SEMESTER – IV

Subject Code	:	24CT2403	**Credits**	:	04
Hours / Week	:	03 Hours	**Total Hours**	:	39 (L)+ 26(P) Hours
L–T–P-J	:	3–0–2-0			

Course Learning Objectives:

This Course will enable students to:

1. Acquire the concept of databases and Relational Model., Entity-Relationship Model and relational model for creating and designing databases for the real-world scenario.
2. Develop queries to extract data from the databases using a structured query language.
3. Draw ER Diagrams and Optimize the Database design using Normalization Concepts.
4. Understand the importance of Transaction Management, Concurrency control mechanism and recovery techniques.
5. Understand NoSQL Database concepts and Demonstrate CRUD operations on MongoDB.

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*** means it includes not only traditional lecture method, but different *type 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, notetaking, 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 analyse information rather than simply recall it.
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.

UNIT – I	**08 Hours**

RELATIONAL DATABASES

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.

(Text Book 1: Chapter 1, 2)

UNIT – II	**08 Hours**

RELATIONAL QUERY LANGUAGE

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.
(Text Book 1: Chapter 3, 4, 5)

UNIT – III	06 Hours

DATABASE DESIGN
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
(Text Book 1: Chapter 6, 7)

UNIT – IV	09 Hours

TRANSACTION MANAGEMENT
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
(Text Book 1: Chapter 17, 18, 19)

UNIT – V	08 Hours

NOSQL Databases
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
(Text Book 2 : Chapter 24)

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Understand the basic concepts of database management system and Relational Model	L2
2	Utilize SQL concepts to build and manipulate relational databases for a given schema.	L3
3	Create E-R diagrams and design relational schema, Apply normalization techniques in designing the relational database	L3
4	Understand the Transaction Management, concurrency control and recovery management techniques.	L2
5	Use NoSQL queries and build databases in MongoDB for the given collection.	L3

Table: Mapping Levels of COs to POs / PSOs														
COs	**Program Outcomes (POs)**												**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
CO1	**1**	-	-	-	-	-	-	-	-	-	-	-	-	**2**
CO2	**1**	**2**	**3**	**-**	**3**	**-**	**-**	**-**	**2**	**1**	**-**	**1**	**-**	**3**
CO3	**1**	**1**	**2**	**-**	**-**	**-**	**-**	**-**	**2**	**1**	**-**	**1**	**-**	**3**
CO4	**1**	**1**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**1**	**-**	**1**
CO5	**1**	**2**	**3**	**-**	**3**	**-**	**-**	**-**	**2**	**1**	**-**	**1**	**-**	**3**

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

TEXT BOOKS:

1. Abraham Silberschatz, Henry F. Korth, S. Sudharshan, "Database System Concepts", Seventh Edition, McGraw Hill, 2020.
2. Ramez Elmasri, Shamkant B. Navathe, "Fundamentals of Database Systems", Seventh Edition, Pears on Education, 2017

REFERENCE BOOKS:

1. Raghu Ramakrishnan and Johannes Gehrke, "Database Management Systems", Third Edition, McGraw-Hill, 2003.
2. C.J. Date, A. Kannan, S. Swamynatham: "An Introduction to Database Systems", Eight Edition, Pearson Education, 2012.

E-Resources:

1. https://www.ibm.com/docs/en/zos-basic-skills?topic=zos-what-is-database-management-system
2. **https://www.mongodb.com/resources/basics/database-management-system**
3. **https://www.oracle.com/in/database/what-is-database/**
4. **https://onlinecourses.swayam2.ac.in/cec19_cs05/preview**
5. **https://onlinecourses.nptel.ac.in/noc19_cs46/preview**

Activity Based Learning (Suggested Activities in Class)

1. Real world problem solving and puzzles using group discussion.
2. Demonstration of solution to a problem through programming.

Design and analysis of algorithms Laboratory

Total Contact Hours: 26

1. Database Table Creation and Manipulation - Create a database table and add constraints such as primary key, unique, check, and not null. Insert, update, and delete rows using SQL DDL and DML commands.
2. Foreign Keys and Referential Integrity - Create a set of tables with foreign key constraints to ensure referential integrity between them.
3. Querying with Conditions and Aggregates - Query the database tables using various WHERE clause conditions and implement aggregate functions like SUM, AVG, COUNT, etc.
4. Subqueries and Basic Joins - Perform queries involving subqueries and simple join operations to retrieve data from multiple tables.
5. Advanced Join Operations - Explore and implement natural joins, equijoins, and outer joins in your queries.
6. UserDefined Functions and Stored Procedures - Write and execute userdefined functions and stored procedures in SQL for various operations.
7. Transactions and Control Commands - Execute complex transactions and understand the use of Data Control Language (DCL) and Transaction Control Language (TCL) commands.
8. SQL Triggers - Write SQL triggers for handling insert, delete, and update operations on a database table.
9. Views - Create views to enhance manageability.
10. NoSQL Database Operations - Create and manage document based data using MongoDB database tools.

Open ended Questions

1. GUIBased Database Application - Develop a simple GUIbased database application incorporating features such as table creation, constraints, queries, joins, functions, procedures, transactions, and triggers.
2. Case Study: RealLife Database Application

****

INTRODUCTION TO ARTIFICIAL INTELLIGENCE

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

SEMESTER – IV

Subject Code	:	24CT2404	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P-J	:	3–0–0-0			

Course Learning Objectives:

This Course will enable students to:

1. To explore introductory survey of concepts and techniques in artificial intelligence.
2. To learn about with methods for search, classification, reasoning and machine learning.
3. Familiar with applications including core AI (game, Planning), robotics , computer vision and natural language understanding

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. ***Lecture method*** means it includes not only traditional lecture methods, but 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 analyse information rather than simply recall it.
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.

UNIT – I	**08 Hours**

INTRODUCTION

What is AI? Foundations of artificial intelligence (AI). History of AI; The State of the Art. Agents and Environments, Good Behavior, The Nature of Environments, The Structure of Agents, Problem- solving agents, Example problems, searching for solutions, Uninformed Search Strategies, Informed (Heuristic) Search Strategies, Heuristic Functions

(Text Book 1: Chapter 1 – 1.1-1.4, Chapter 2 – 2.1-2.4, Chapter 3 – 3.1-3.6)

UNIT – II	**08 Hours**

ROBOTICS AND CLASSICAL PLANNING

Classical Planning: Definition, Algorithms for Planning as State-Space Search, Planning Graphs, Other Classical Planning Approaches, Analysis of Planning Approaches, Robotics: Introduction, Robot Hardware, Robotic Perception, planning to move, Planning Uncertain Movements.

(Text Book 2: Chapter 11 – 11.1-11.7 Chapter 26 – 26.1-26.6, Chapter 4 – 4.4.1, 4.4.2, Chapter 6 – 6.3, C12- 12.2, Chapter 13- 13.3.1, 13.3.2, Text Book 1: Chapter 9 – 9.1, 9.2.1, 9.3)

UNIT – III	**08 Hours**
Uncertainty, Naive Bayes and Probabilistic Reasoning Acting Under Uncertainty, Review of Basic Probability, Bayes Theorem, Representing Knowledge in an Uncertain Domain, The Semantics of Bayesian Networks, Exact Inference in Bayesian Networks, Time and Uncertainty, Inference in Temporal Models ***(Text Book 2: Chapter 12.1–12.6 Chapter 13 – 13.1-13.3 Chapter 14-14.1-14.2)***	
UNIT – IV	**08 Hours**
LEARNING Forms of Learning, Supervised Learning, Learning Decision Trees, Evaluating and Choosing the Best Hypothesis, Regression and Classification with Linear Models, Artificial Neural Networks. Reinforcement Learning ***(Text Book 1: Chapter 19 – 19.1-19.6)***	
UNIT – V	**07 Hours**
APPLICATIONS Natural Language Processing: Language Models, Text Classification, Information Retrieval, Information Extraction, Computer Vision: Image Formation, Early Image-Processing Operations, Object Recognition by Appearance, Reconstructing the 3D World, Object Recognition from Structural Information, Using Vision. ***(Text Book 1: Chapter* 23- 23.1 Chapter 25-25.1-25.6)**	

Course Outcome	**Description**	**Bloom's Taxonomy Level**
At the end of the course the student will be able to:		
1	Demonstrate fundamental understanding of artificial intelligence (AI) especially the notion of problem-solving using AI techniques, current scope and limitations thereof.	L2
2	Apply basic principles of AI in solutions that require problem solving, knowledge representation, and learning	L3
3	Demonstrate awareness and a fundamental understanding of applying AI techniques in intelligent agents, expert systems, artificial neural networks and other machine learning models.	L2

Table: Mapping Levels of COs to POs / PSOs

COs	**Program Outcomes (POs)**												**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
CO1	1	2												3
CO2	3	2												3
CO3	2	2												3

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

TEXT BOOKS:

1. Stuart Russell and Peter Norvig. Artificial Intelligence – A Modern Approach, Pearson Education Press, 4th Edition.

REFERENCE BOOKS:

1. Artificial Intelligence by Elaine Rich, Kevin Knight and Shivashankar B Nair, Tata McGraw Hill.
2. Introduction to Artificial Intelligence and Expert Systems by Dan W. Patterson, Pearson Education.

E-Resources:

1. https://nptel.ac.in/courses/106105079
2. https://www.coursera.org/learn/introduction-to-ai

Activity Based Learning (Suggested Activities in Class)

1. Real world problem solving and puzzles using group discussion.
2. Demonstration of solutions to a problem through programming.

****

COMPUTER ORGANIZATION AND ARCHITECTURE

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

SEMESTER – IV

Subject Code	: 24CT2405	**Credits**	: 03
Hours / Week	: 03 Hours	**Total Hours**	: 39 Hours
L–T–P–J	: 3–0–0–0		

Course Learning Objectives:

This Course will enable students to:

1. Understand the Architecture and programming of ARM microprocessor.
2. Develop program using Arm instruction set and appreciate the advanced features provided in the ARM
3. Understand the exception handling techniques.
4. Study in detail the concept of instruction level parallelism and concepts of pipelining.
5. Understand various cache memory mapping techniques and memory Organization.

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*** means it includes not only traditional lecture method, but different *type 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, notetaking, 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. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I	**05 Hours**

An Overview of Computing Systems:
History of Computers, The Computing Device.
The ARM7TDMI Programmers' Model:
Introduction, Data types, Processor Modes, Registers, Program Status Registers, The vector Table.
Assembler Rules and Directives: Structure of Assembly Language Modules, Registers, Directives and Macros.
Loads, Stores and Addressing: LODS and STORES instructions, Operand Addressing, ENDIANNES
Text Book-1: 1.1 to 1.3; 2.1 to 2.3; 4; 5.3, 5.4, 5.5

UNIT – II	**05 Hours**

Constants and Literal Pools:
The ARM Rotation Scheme, Loading Constants and address into Registers
Logic and Arithmetic: Flags and their Use, Compare instructions, Data Processing Instructions
Loops and Branches: Branching, Looping, Conditional Execution, Straight-Line Coding
Subroutines and Stacks: Stack, Subroutines, Passing parameters to subroutines.
Text Book-1: 6.1 to 6.4; 7.1 to 7.4; 8.2 to 8.5; 13.1 to 13.4

UNIT – III	**05 Hours**
Mixing C and Assembly Language: Inline Assembler Embedded Assembler, Calling Between C and Assembly. **Exception Handling:** Interrupts, Error Conditions, Processor Exception Sequence, The Vector Table, Exception Handlers, Exception Priorities, Procedures for Handling Exceptions. **Text Book-1: 18.1 to 18.4; 14.1 to 14.8**	
UNIT – IV	**12 Hours**
Pipelining: Basic and Intermediate Concepts: Introduction, The Major Hurdle of Pipelining, How Pipelining Implemented, what makes Pipelining hard to Implement, Extending the MIPS Pipeline to Handle Multicycle Operations, The MIPS R4000 Pipeline, Crosscutting Issues, score boarding Technique. **Text Book-2: C.1 to C.7**	
UNIT – V	**12 Hours**
Memory Hierarchy: Introduction, Cache Performance, Six basic cache Optimizations, Virtual Memory, Protection and examples of Virtual Memory, Fallacies and Pitfalls. **Text Book-2: B.1 to B.6**	

Course Outcome	**Description**	**Bloom's Taxonomy Level**
At the end of the course the student will be able to:		
1	**Interpret** the knowledge of the internal architecture and organization of ARM microprocessors to understand their components and functionalities.	L2
2	**Apply** the instruction set of ARM Microprocessor by writing Assembly language programs.	L3
3	**Understand** the various exception handling techniques.	L2
4	**Demonstrate** the integration of C and Assembly language to manage function calls between C and Assembly language.	L2
5	**Apply** the concept of instruction-level parallelism and understand the principles of Pipelining techniques.	L3
6	**Understand** memory hierarchy and its impact on computer cost/performance.	L2

Table: Mapping Levels of COs to POs / PSOs

COs	**Program Outcomes (POs)**												**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
CO1	3		2										2	

CO2	3		3		1							2	
CO3	3	1	1									2	
CO4	3	1	1									2	
CO5	3	2	1									2	
CO6	3	2	1									2	

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

TEXT BOOKS:

1. William Hohl, "ARM Assembly Language", 2nd Edition, CRC Press, 2009.
2. John L Hennessy, David A Patterson, "Computer Architecture, A Quantitative Approach", 6th Edition, Morgan Kaufmann publishers, 2019.

REFERENCE BOOKS:

1. David A Patterson, John L Hennessy, "Computer Organization and Design", 4th Edition, Morgan Kaufmann publishers, 2010.
2. Steve Furber, "ARM System-on-chip Architecture", 2nd Edition, Pearson Publications, 2000.
3. Carl Hamacher, Zvonko Vranesic, Safwat Zaky, "Computer Organization", 5th Edition, Tata McGraw Hill, 2002.

E-Resources:

1. https://www.udemy.com/topic/arm-cortex-m/
2. https://www.edx.org/school/armeducation
3. https://onlinecourses.nptel.ac.in/noc22_cs93/preview

Activity Based Learning (Suggested Activities in Class)

1. Mini project implementation using Assembly Language Programming.
2. Demonstration of solution to a problem through programming.

****

SMALL E-BUSINESS LAUNCH

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

SEMESTER – IV

Subject Code	:	24CT2406	**Credits**	:	02
Hours / Week	:	03 Hours	**Total Hours**	:	26 Hours
L–T–P–S	:	1–0–0–2			

Course Learning Objectives:

This Course will enable students to:

1. **Experience** with a set of tools and methods for setting up an online service.
2. **Knowledge** of the role of multiple functions in creating a small business.
3. **Ability** to coordinate multiple, interdisciplinary tasks to achieve a common objective.
4. **Emphasis** of specific knowledge from other courses through practice and reflection in an action-oriented setting.

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*** means it includes not only traditional lecture method, but different *type 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, notetaking, 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 analyse information rather than simply recall it.
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.

UNIT – I	**06 Hours**

INTRODUCTION:

Introduction, Identifying Customer Needs, Opportunity Identification & Planning, Idea Generation and Validation, Entrepreneurial Culture, Digital Business Opportunity

(Text Book-1: Chapter 1: 1.1, 1.4 to 1.9, 1.11 to 1.17, 1.22 to 1.28)

UNIT – II	**06 Hours**

Business Plan & its Components, Business Model and Strategies, Selection of Appropriate Business Model, Business Model Canvas, Feasibility

(Text Book-1: Chapter 3: 3.1 to 3.18)

UNIT – III	**06 Hours**

Lean thinking, Specifications, Wireframe Generation and Validation.

(Text Book-1: Chapter 5: 5.1 to 5.4, Chapter 5: 5.11 to 5.19)

UNIT – IV	**08 Hours**

Service-Market Fit, legal Aspects & Entrepreneurial Law, Logistics of goods delivery, Virtual Trade Shows

(Text Book-1: Chapter 7: 7.2 to 7.17)

Course Outcomes:

At the end of the course the student will be able to:

1. Identify customer needs and pain points.
2. Generate concept and validate idea.
3. Use design thinking to create a wire frame of your online service.
4. Launch your app/website.
5. Register your small business.

List of Laboratory/Practical Experiments activities to be conducted (if any):	**13 Hrs**
Building of Business Models and seminars on deployment tools	

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
CO1	2	3										2		3
CO2	3	2	2									1		2
CO3	2	2	2									1		2
CO4	3	2	2									2		3
CO5	3	2										1		3

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

TEXT BOOKS:

1. Fisher, Steven, Duane, Ja-Nae (2016). Start-up Equation – A Visual Guidebook for Building your Start-up, New Delhi: Tata McGraw-Hill
2. Ulrich, K. T. (2019). Product design and development. Tata McGraw-Hill Education, 7th edition

REFERENCE BOOKS:

1. Holt H., David (2017). Entrepreneurship: New Venture Creation, New Delhi: Pearson Education
2. Zimmerer, W. Thomas and Scarborough, M. Norman and Doug Wilson (2009). Essentials of Entrepreneurship and Small Business Management, 5/e; New Delhi: Prentice Hall India

Activity Based Learning (Suggested Activities in Class)

1. Proposal Handout & Presentation
   a. The Problem
   b. Need
   c. Existing Alternatives
   d. Potential Market

2. Wireframe Sketches, Target Specifications
   a. List Critical Customer Needs
   b. For a., Prepare a List of Target Specifications
   c. Prepare Wireframe

****

IDEA GENERATION & VALIDATION

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

SEMESTER – IV

Subject Code	:	24CT2407	**Credits**	:	02
Hours / Week	:	01 Hours	**Total Hours**	:	26 Hours
L–T–P-J	:	1–0–0–2			

Course Learning Objectives:

This Course will enable students to:

1. Generate, develop, and pitch innovative new venture ideas
2. Evaluate the attractiveness and feasibility of new venture opportunities
3. Analyze industries, markets, and competitors

Teaching-Learning Process (General Instructions)

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

Lecture method means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.

1. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
2. Show ***Video/animation*** films to explain functioning of various concepts.
3. Encourage ***Collaborative*** (Group Learning) Learning in the class.
4. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
5. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
6. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
7. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

CONTENT	**13 Hours**
Process of Idea Generation; Problem Identification, Idea formation, Idea Generation Techniques: Brainstorming, Mind Mapping, Story Boarding, Attribute Listing, Idea Validation and Risk Assessment	

List of Laboratory/Practical Experiments activities to be conducted (if any):	**13 HRS**

1. Get out of the Building Exercise – Meet at least 20 persons in your targeted segment and understand their pain points and propose ideas that may solve their problems.
2. Discuss with mentors and zero in on one idea.
3. Make a hand drawn functional sketch of the proposed solution
4. Idea Validation with potential customers - Meet at least 20 customers and validate the proposed idea with them.
5. Prepare a comprehensive report, highlighting the learnings and the implications on the proposed solution and pivots (if any).

Course Outcome	**Description**	**Bloom's Taxonomy Level**
At the end of the course the student will be able to:		
1	Classify idea generation techniques to validate ideas.	L2
2	Interpret to pitch innovations on new venture ideas.	L3

Table: Mapping Levels of COs to POs / PSOs														
COs	**Program Outcomes (POs)**												**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
CO1	2	2	3	3	3							2		3
CO2	2	2	3	3	3							2		3

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

TEXT BOOKS:

1. Kotter, J.P. and Whitehead, L.A. (2010). Buy-In: Saving Your Good Idea from Getting Shot Down.

REFERENCE BOOKS:

1. Belsky, S. (2010). Making Ideas Happen: Overcoming the Obstacles between Vision and Reality.

****

Technical writing

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

SEMESTER – IV

Course Code	:	24CT2408	**Credits**	:	02
Hours / Week	:	01 Hours	**Total Hours**	:	26 Hours
L–T–P-J	:	1–0–0–2			

Course Learning Objectives:

This Course will enable students to:

1. Teach how to improve writing skills and level of readability
2. Tell about what to write in each section
3. Summarize the skills needed when writing a Title
4. Infer the skills needed when writing the Conclusion
5. Ensure the quality of paper at very first-time submission

Teaching-Learning Process (General Instructions)

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

Lecture method means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.

1. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
2. Show ***Video/animation*** films to explain functioning of various concepts.
3. Encourage ***Collaborative*** (Group Learning) Learning in the class.
4. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
5. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
6. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
7. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

–	UNIT I	6 Hours

INTRODUCTION TO RESEARCH PAPER WRITING:

Planning and Preparation, Word Order, Breaking up long sentences, Structuring Paragraphs and Sentences, Being Concise and Removing Redundancy, Avoiding Ambiguity and Vagueness

UNIT-II	6 Hours

PRESENTATION SKILLS:

Clarifying Who Did What, Highlighting Your Findings, Hedging and Criticizing, Paraphrasing and Plagiarism, Sections of a Paper, Abstracts, Introduction

UNIT-III	5 Hours

TITLE WRITING SKILLS:

Key skills are needed when writing a Title, key skills are needed when writing an Abstract, key skills are needed when writing an Introduction, skills needed when writing a Review of the Literature, Methods, Results, Discussion, Conclusions, The Final Check

UNIT-IV	5 Hours

RESULT WRITING SKILLS:

Skills are needed when writing the Methods, skills needed when writing the Results, skills are needed when writing the Discussion, skills are needed when writing the Conclusions

UNIT-V	5 Hours

VERIFICATION SKILLS:

Useful phrases, checking Plagiarism, how to ensure paper is as good as it could possibly be the first-time submission

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Understand that how to improve your writing skills and level of readability	L2
2	Learn about what to write in each section	L2
3	Understand the skills needed when writing a Title	L2
4	Understand the skills needed when writing the Conclusion	L2
5	Ensure the good quality of paper at very first-time submission	L3

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
CO1	2	2	2	2	2							2	2	
CO2	2	2	2	2	2							2	2	
CO3	2	2	2	2	2							2	2	
CO4	2	2	2	2	2							2	2	
CO5	2	2	2	2	2							2	2	

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

BOOKS:

1. Adrian Wallwork , English for Writing Research Papers, Springer New York Dordrecht Heidelberg London, 2011
2. Day R How to Write and Publish a Scientific Paper, Cambridge University Press 2006
3. Goldbort R Writing for Science, Yale University Press (available on Google Books) 2006
4. Highman N, Handbook of Writing for the Mathematical Sciences, SIAM. Highman's book, 1998.

REFERENCE BOOKS:

1. Adrian Wallwork , English for Writing Research Papers, Springer New York Dordrecht Heidelberg London, 2011

****

INTRODUCTION TO IPR AND REGISTRATION

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

SEMESTER – IV

Course Code	: 24CT24s09	**Credits**	:	02
Hours / Week	: 01 Hours	**Total Hours**	:	26 Hours
L–T–P	: 1-0-0-2			

Course Learning Objectives:

This Course will enable students to:

1. To understand the concepts IPR
2. To understand Trademarks, Trade Secretes and GI of goods.
3. To understand Copyrights, Patents and Industrial Designs.
4. To learn about how to manage IP rights and legal aspects.

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*** means it includes not only traditional lecture method, but different *type 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, notetaking, 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 analyse information rather than simply recall it.
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.

UNIT – I	**04 Hours**

INTRODUCTION TO IPR:

Introduction to Intellectual Property Rights, types of intellectual property, importance of intellectual property rights, Evolution of IP acts and treaties, Agencies responsible for IPR registrations, Role and value of IP in international commerce, Issues affecting IP internationally. ***(Text Book-1: Chapter 1).***

NIT – II	**07 Hours**

Trade Marks:

Purpose and function of trademarks, Acquisition of trade mark rights, transfer of rights, Selecting and evaluating trademark, registration of trademarks, claims.

(Text Book-1: Chapter 2,3,4).

Trade Secrets: Trade secret law, determination of trade secret status, liability for misappropriation of trade secrets, trade secret litigation.

(Text Book-1: Chapter 22).

UNIT – III	**07 Hours**

COPYRIGHTS:

Fundamentals Of Copyright Law, Originality of Material, Right of Reproduction, Right to Perform the Work Publicly, Copyright Ownership Issues, Notice of Copyright. (Text Book-1: Chapter 10,11).

PATENTS: Foundation of Patent Law, Patent Searching Process, Basic Criteria of Patentability Industrial Designs: Kind of Protection Provided in Industrial Design. (Text Book-1: Chapter 17,18).

Managing IP Rights:

Acquiring IP Rights: letters of instruction, joint collaboration agreement, Protecting IP Rights: non-

disclosure agreement, cease and desist letter, settlement memorandum. Transferring IP Rights: Assignment contract, license agreement, deed of assignment. ***(Text Book-1: Chapter 19).***
UNIT – IV | **08 Hours**
DESIGN- Meaning, Definition, Object, Registration of Design, Cancellation of Registration, International convention on design, functions of Design. Semiconductor Integrated circuits and layout design Act- 2000. ***(Text Book-2: Chapter 8).*** **BASIC TENENTS OF INFORMATION TECHNOLOGY ACT-2000 –** IT Act - Introduction E-Commerce and legal provisions E- Governance and legal provisions Digital signature and Electronic Signature. ***(Text Book-2: Chapter 12).***

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	**Demonstrate** the basic concepts of IPR and their role and value internationally.	L2
2	**Interpret** trademark, their values and trade secrets for liability and litigations.	L3
3	**Devise** knowledge on copyrights, patents and how to manage and protect IP.	L3
4	**Examine** convention on design and basic information technology act.	L3

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
CO1	3	3	3	3	3							2		3
CO2	3	3	3	3	3	3			3		3	2		3
CO3	3	3	3	3	3	3			3		3	1		3
CO4	3	3	3	3	3	3			3		3	2		3

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

TEXT BOOKS:

1. Deborah Bouchoux ,"Intellectual Property: The Law of Trademarks, Copyrights, Patents, and Trade Secrets", 4th Edition, Delmar Cengage Learning.
2. Elizabeth Verkey," Intellectual Property: Law and Practice", 2015 Edition, Eastern Book Company

REFERENCE BOOKS:

1. Kompal Bansal , Parikshit Bansal, "Fundamentals of Intellectual Property for Engineers", 2014, BS Publications.
2. Prabuddha Ganguli," Intellectual property right - Unleashing the knowledge economy", Ganguli, 1st edition (1 July 2017), McGraw Hill Education.

E-Resources:

1. Inventing the Future: An introduction to Patents for small and medium sized Enterprises; WIPO publication No. 917. URL: www.wipo.int/ebookshop.
2. Looking Good : An Introduction to Industrial Designs for Small and Medium-sized Enterprises; WIPO publication No.498. URL: www.wipo.int/ebookshop.
3. Creative Expression: An Introduction to Copyright and Related Rights for Small and Medium-sized Enterprises; WIPO publication No. 918. URL: www.wipo.int/ebookshop
4. Making a Mark: An Introduction to Trademarks for Small and Medium-sized Enterprises; WIPO publication No. 900. URL: www.wipo.int/ebookshop

****

DevOps FUNDAMENTALS

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

SEMESTER –IV

Course Code	:	24CT2411	**Credits**	:	02
Hours / Week	:	03 Hours	**Total Hours**	:	26 Hours
L–T–P–J	:	1–0–0–2			

Course Learning Objectives:

This course will enable students to:

1. **To understand** the principles and lifecycle of DevOps and its benefits in modern software development.
2. **To describe** the fundamentals of Git and its role in version control, including common workflows and collaborative practices.
3. **To outline** the concepts of CI/CD, the setup of Jenkins, and the basics of Docker containerization and orchestration.

Teaching-Learning Process (General Instructions)

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

1. ***Lecture method*** means it includes not only the traditional lecture method but a different *type of teaching method* that may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note-taking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain the 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. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the student's understanding.

UNIT – I: Introduction to DevOps:	**06 Hours**

Understanding DevOps, Benefits, DevOps Lifecycle
(Text Book-1: Chapter 1)

Version Control with Git
Git Fundamentals, Workflow, Collaboration with Git
(Text Book-1: Chapter 4)

UNIT – II: Continuous Integration and Continuous Deployment (CI/CD)	**10 Hours**

CI/CD Concepts, Jenkins, Creating and managing CI/CD pipelines with code.
(Text Book-1: Chapter 5)

UNIT – III: Containerization with Docker	**10 Hours**

Introduction to Containers, Learning Docker fundamentals, containers, and images., Exploring container orchestration with Kubernetes.
(Text Book-1: Chapter 8)

Course Outcomes:

At the end of the course the student will be able to:

1. **Apply** Git for version control by creating repositories, managing branches, and collaborating on code using workflows.
2. **Construct** automated CI/CD pipelines using Jenkins and manage them using pipeline-as-code practices.
3. **Analyze** the role of Docker and Kubernetes in containerization and orchestration for scalable and

reliable deployment.

Table: Mapping Levels of COs to POs / PSOs **3:**

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
CO1	3	2	2	2	2	-	-	-	2	2	2	2	-	2
CO2	2	2	2	2	2	-	-	-	2	2	2	2	-	2
CO3	2	2	2	2	2	-	-	-	2	2	2	2	-	2

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

TEXT BOOKS:

1. **"DevOps Automation Cookbook" by Ekambar Kumar Singirikonda**

E-Resources:

1. https://www.ebooks.com/en-us/book/211369550/devops-automation-cookbook/ekambar-kumar-singirikonda/
2. https://www.mygreatlearning.com/academy/learn-for-free/courses/introduction-to-devops1
3. https://intellipaat.com/academy/course/devops-free-course/

Activity Based Learning (Suggested Activities in Class)

1. Development of project using DevOps tools and group discussion.
2. Collaborative Activity is minor project development with a team of 4 students.

****

THEORY OF COMPUTATION

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

SEMESTER – V

Subject Code	: 24CT3501	**Credits**	: 04
Hours / Week	: 04 Hours	**Total Hours**	: 39 Hours (Theory)+13Hours (Tutorial)
L–T–P-J	: 3–1–0-0		

Course Learning Objectives:

This Course will enable students to:

1. **Understand** the Automata Theory and Formal Languages to build efficient design of FA
2. **Identify** Regular Expression and recognize the properties that make a language regular and construct the FA of the language.
3. **Devise** the technique to minimize DFA and understand the importance of minimization in optimizing automata for efficient language recognition.
4. **Get the idea** to Interpret and design different PDA for a given language
5. **Describe** the finite automata and formal languages, equipping them with the knowledge and skills necessary to analyse and design TM for language recognition tasks.

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*** means it includes not only traditional lecture method, but different *type 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, notetaking, 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 analyse information rather than simply recall it.
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.

UNIT – I	**08 Hours**

INTRODUCTION TO FINITE AUTOMATA:

Study and Central Concepts of Automata Theory, Finite Automata -Yet Another Method for Defining Languages, Deterministic and Nondeterministic Finite Automata, Finite Automata with Epsilon – transitions. An Application: Text Search.

(Text Book-1: Chapter 1: 1.1, 1.7, Chapter 2: 2.2 to 2.5)

(Text Book-2: Chapter 5: Page no: 52)

UNIT – II	**08 Hours**

REGULAR EXPRESSIONS AND LANGUAGES:

Regular Expressions, Finite Automata and Regular Expressions, Algebraic Laws of Regular expressions, Applications of Regular Expressions, Properties of Regular Languages - Pumping Lemma, Applications of the Pumping Lemma, Closure Properties of Regular languages, Equivalence and minimization of Automata.

(Text Book-1: Chapter 3: 3.1 to 3.4, Chapter 4: 4.1, 4.2, 4.4)

UNIT – III	09 Hours

CONTEXT – FREE GRAMMARS AND LANGUAGES:
Context-Free Grammars, Parse Trees,Ambiguity in Grammars and Languages, Properties of Context free languages- Normal Forms of Context-Free Grammars, The Pumping Lemma for Context Free Languages, Closure Properties of Context-Free Languages.
(Text Book-1: Chapter 5: 5.1 to 5.2.3, 5.4, Chapter 7: 7.1 to 7.3)

UNIT – IV	06 Hours

PUSHDOWN AUTOMATA:
Definition of the Pushdown automation (PDA), The Language of PDA, Equivalence of PDA's and CFG's-From Grammars to Push Down Automata and PDA to Grammars, Deterministic Pushdown Automata
(Text Book-1: Chapter 6: 6.1 to 6.4)

UNIT – V	08 Hours

INTRODUCTION TO TURING MACHINE:
Problems that Computers Cannot Solve, The Turing Machine, Programming Techniques for Turing Machine,Extensions to the Basic Turing Machine.
(Text Book-1: Chapter 8 : 8.1, 8.4)

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Utilize the basic concepts of formal languages of finite automata techniques such as DFA, NFA and E-NFA	L3
2	Develop Finite Automata for different Regular Expressions and Languages and minimization of Finite Automata to Regular Expression.	L3
3	Analyze context-free grammars, ambiguity and Chomsky normal form grammars to design computer languages	L4
4	Construct context free, regular, Interpret and design different PDA for a given language	L5
5	Design Turing machine to solve problems.	L6

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
CO1	3	3	3									2		3
CO2	3	3	3									2		3
CO3	1	1		3	1							1		2
CO4	2	2	3	3	1							2		3
CO5	1	2	3	3	3							3		3

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

TEXT BOOKS:

1. J.E. Hopcroft, R. Motwani, and J. D. Ullman, Introduction to Automata Theory, Languages andComputation, 3rd Edn. Pearson Education , New Delhi 2008
2. Daniel I. A. Cohen, Introduction to Computer Theory, 2nd Edition, Wiley IndiaStudent Edition, 2008.

REFERENCE BOOKS:

1. K.L.P. Misra and N. Chandrashekaran. Theory of Computer Science- Automata, Languages andComputation, 3rd Edn. PHI, New Delhi, 2007
2. C. Martin - Introduction to Languages and the Theory of Computation 2ndEdn, TMH, NewDelhi, 2000.

Activity Based Learning (Suggested Activities in Class)

3. Real world problem solving and puzzles using group discussion. E.g., Fake coin identification, Cabbage puzzle, Konigsberg bridge puzzle etc.,
4. Demonstration of solution to a problem through programming.

List of Laboratory/Practical Experiments activities to be conducted.
1. Design a Program for creating machine that accepts three consecutive one.
2. Design a Program for creating machine that accepts the string always ending with 101.
3. Design a Program for Mode 3 Machine
4. Design a program for accepting decimal number divisible by 2.
5. Design a program for creating a machine which accepts string having equal no. of 1's and 0's.
6. Design a program for creating a machine which count number of 1's and 0's in a given string.
7. Design a Program to find 2's complement of a given binary number.
8. Design a Program which will increment the given binary number by 1.
9. Design a Program to convert NDFA to DFA.
10. Design a Program to create PDA machine that accept the well-formed parenthesis.
11. Design a PDA to accept WCWR where w is any string and WR is reverse of that string and C is a Special symbol.
12. Design a Turing machine that's accepts the following language an b n c n where n>0.

****

AI-Enabled Software Systems & Agile Project Management

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

SEMESTER – V

Subject Code	:	**Credits**:	03
Hours / Week	: 03 Hours	**Total Hours**:	45 L Hours
L–T–P-J	: 3–0–0-0	**CIE Marks**	: 60 Marks + 40 Marks

Prerequisites: Basics of Programming (Python/Java) , Software Engineering Fundamentals, Introduction to Artificial Intelligence (preferred)

Course Learning Objectives:

This course will enable students to:

1. Understand AI-enabled software architectures and intelligent system design.
2. Apply Agile methodologies for managing AI-driven projects.
3. Develop AI-integrated software solutions using modern tools and frameworks.
4. Analyze lifecycle challenges in AI systems including data, deployment, and ethics.
5. Implement collaborative, iterative, and adaptive development practices

Broad Course Description

This course focuses on the integration of Artificial Intelligence into modern software systems and the application of Agile methodologies for efficient project management. It covers AI-driven system architecture, data-centric development, MLOps pipelines, and real-time deployment strategies. Students will explore Agile frameworks such as Scrum, Kanban, and SAFe, and understand how iterative development aligns with AI experimentation cycles. The course emphasises hands-on learning using industry-relevant tools including cloud AI platforms, Git-based workflows, CI/CD pipelines, and collaboration tools. By bridging AI technologies with Agile practices, the course prepares students to build scalable, ethical, and production-ready intelligent systems in multidisciplinary environments

UNIT – I	**09 Hours**
Foundations of AI-Enabled Software Systems: AI in Software Engineering: Evolution and impact**,** AI system components: Data, models, pipelines**,** Software architecture for AI systems (microservices, APIs) **,** Data-driven development lifecycle**,** Introduction to MLOps concepts *(Text Book 1: Chapters 1,2 | Ref 2,3)*	
UNIT – II	**09 Hours**
Agile Methodologies & Frameworks: Agile Manifesto and principles**,** Scrum framework: roles, artefacts, ceremonies**,** Kanban and Lean development**,** SAFe for scaling Agile**,** Agile vs Traditional project management *(Text Book 2: Chapters 1–4 | Ref 1)*	

UNIT – III	**09 Hours**
AI Development Lifecycle & Tools: Data collection, pre-processing, labelling**,** Model development and evaluation**,** Version control for AI (Git, DVC) **,** CI/CD for AI pipelines**,** Experiment tracking and reproducibility *(Text Book 3: Chapters 3,5 | Ref 2,4)*	
UNIT – IV	**09 Hours**
Deployment, Monitoring & Ethics: Model deployment (cloud, edge, APIs), Monitoring AI systems (drift, performance), Responsible AI: bias, fairness, transparency, Legal and ethical aspects (link to IPR concepts) Security in AI systems *(Text Book 1: Chapter 6 | Ref 3,5)*	
UNIT – V	**09 Hours**
Agile AI Project Execution: Managing AI projects in Agile sprints **,**Product backlog for AI features	

,Risk management in AI projects ,Collaboration and interdisciplinary teams ,Case studies (industry: Google, Amazon, Tesla AI systems)

(Text Book 2,3 | Ref 4,5)

AI Integration & Authentic Intelligence: Reflection (AI² Pedagogy)

- AI-assisted coding using GitHub Copilot
- Experimentation using Google Colab
- Model deployment via Amazon Web Services / Microsoft Azure
- MLOps pipelines using MLflow .
- Agile tracking using Jira

Applications

1. AI-powered enterprise systems
2. Intelligent SaaS platforms
3. Healthcare AI solutions
4. FinTech automation systems
5. Autonomous and smart Applications

Mini Project Component

Project 1 — AI Agile Sprint Simulation Build a small AI product using Scrum methodology

- Deliverables: backlog, sprint reports, working prototype

Project 2 — MLOps Pipeline Development

- Develop and deploy an AI model with CI/CD
- Tools: GitHub, MLflow, Docker

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
CO1	Explain AI-enabled software architectures and Agile principles	L2
CO2	Apply Agile methodologies to manage AI projects	L3
CO3	Develop AI-integrated software systems using modern tools	L3
CO4	Analyze deployment, monitoring, and ethical challenges in AI	L4
CO5	Design scalable AI systems using Agile practices	L6

Table: Mapping Levels of COs to POs / PSOs		
COS	**Program Outcomes (POs)**	**Program Specific Outcomes (PSOs)**

	PO 1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO1 0	PO1 1	PSO 1	PSO 2
CO1	3	2								2		3	2
CO2	2	3	3		2					3		2	3
CO3	2	2	3	2	3					2		2	2
CO4	2	3	2	3	2	2		2		2		2	3
CO5	3	3	3	2	3	2		2	3	3	2	3	3

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

TEXT BOOKS:

1. Designing Machine Learning Systems, Chip Huyen, O'Reilly Media, 1st Edition, 2022.
2. Agile Project Management with Scrum, Ken Schwaber, Microsoft Press, 1st Edition, 2004.
3. Machine Learning Engineering, Andriy Burkov, True Positive Inc., 1st Edition, 2020.

REFERENCE BOOKS:

1. The Lean Startup, Eric Ries, Crown Business, 1st Edition, 2011.
2. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, Aurélien Géron, O'Reilly Media, 2nd Edition, 2019.
3. Site Reliability Engineering, Google, O'Reilly Media, 1st Edition, 2016.
4. Accelerate: The Science of Lean Software and DevOps, Nicole Forsgren, Jez Humble, Gene Kim, IT Revolution Press, 1st Edition, 2018.

NVIDIA Course Mapping (Industry Alignment)

NVIDIA DLI Course / Content	Mapped Units	Relevance
Fundamentals of Accelerated Computing with CUDA C/C++	Unit III & IV	Relates to AI model development, parallel computing, and deployment optimization using GPUs.
Fundamentals of Deep Learning for Multi-GPU and CPU Systems	Unit I & V	Connects AI system architecture with scalable training and Agile-based AI project execution.
Edge AI and Embedded Systems with Jetson	Unit IV	Maps to AI model deployment on edge devices and real-time intelligent systems.
Fundamentals of Accelerated Data Science	Unit I & III	Links data-driven development, preprocessing, and AI pipelines with high-performance computing.
Building Real-Time Video AI Applications	Unit IV & V	Demonstrates deployment, monitoring, and Agile delivery of real-time AI Applications.

NVIDIA DLI Course / Content	Mapped Units	Relevance
		Covers CI/CD pipelines,

MLOps: Production Machine Learning Systems	Unit III & V	experiment tracking, and lifecycle management of AI systems in Agile environments.

E-RESOURCES:

1. NPTEL – AI & Software Engineering courses
2. Coursera – Agile & AI Specialisations
3. edX – MLOps and AI systems
4. Google Cloud AI Learning Paths
5. NVIDIA DLI Courses

Activity-Based Learning

1. Agile sprint simulation (team-based)
2. AI model building hackathons
3. CI/CD pipeline demonstrations
4. Case study discussions (industry AI failures & success)
5. Peer code reviews using Git

****

MACHINE LEARNING

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

SEMESTER - V

Course Code	:		**Credits**	:	04
Hours / Week	:	05	**Total Hours**	:	45L+15P
L–T–P–J	:	3–0–2–0	**CIE+SEE**	:	60+40 Marks

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:

This course will enable students:

1. Understand the types of classifications and dimensionality reduction techniques.
2. To become acquainted with regression, classification, and error functions.
3. To become acquainted with the concepts of ensemble, clustering and reinforcement learning.
4. Show scholarly expertise in the application of and analysis of machine learning algorithms to address various learning challenges.

Broad Course Description

Machine Learning is a core area of Artificial Intelligence that focuses on developing algorithms and models that enable computers to learn patterns from data and make intelligent decisions without being explicitly programmed. This course introduces the fundamental concepts, techniques, and applications of machine learning, including supervised, unsupervised, and reinforcement learning approaches. Students will explore key algorithms such as linear regression, decision trees, support vector machines, and clustering methods.

The course emphasizes data preprocessing, feature selection, model training, and evaluation techniques to build efficient and accurate predictive models. It also covers essential mathematical foundations, including probability, statistics, and optimization methods. Practical implementation using popular tools like Python and libraries such as Scikit-learn and TensorFlow is an integral part of the course.

Students will gain hands-on experience through real-world datasets and projects in domains like healthcare, finance, and natural language processing. The course also introduces concepts of model deployment and ethical considerations in machine learning. By the end of the course, learners will be equipped with the skills to design, implement, and evaluate machine learning solutions for practical problems.

UNIT – I	**09 Hours**

Introduction to Machine Learning

Introduction to Machine Learning:

Introduction to Machine Learning, Types of Machine Learning, Applications of Machine Learning, Issues in machine Learning, Basic Types of Data in Machine Learning, Data Preprocessing: Missing Value Imputation, Handling balanced and Imbalanced Data, and Feature Engineering, Evaluation Metrics(Confusion Matrix, Accuracy, Precision, Recall, F1 Score, Cost Function, Cross- Validation, Error Metrics (MSE, RMSE, MAE)).

Textbook 1: Chapter: 1

Textbook 2: Chapter: 1, 2

UNIT – II	**09 Hours**

Supervised Machine Learning - I

Introduction, Examples of Supervised Machine Learning, Classification Model, Classification Learning Steps, Classification Algorithms: KNN, Naïve Bayes, Support Vector Machine, Decision Tree (ID3), Random Forest, Bagging & Boosting.

Textbook 1: Chapter: 3, 6, 8 Textbook 2: Chapter: 7	
UNIT – III	**09 Hours**
Supervised Machine Learning - II Regression: Introduction to Regression, Example of Regression. Regression Algorithms: Linear Regression, Multiple Linear Regression, Logistic Regression, Overfitting, Underfitting, Bias, Variance. Introduction to Neural Networks, Perceptron, Multi-layer Perceptron, Backpropagation. Textbook 1: Chapter: 4 Textbook 2: Chapter: 8, 10	
UNIT – IV	**09 Hours**
Unsupervised Machine Learning I: Introduction to Unsupervised, Application of Unsupervised, Clustering: K-Means, K-Medoid, Hierarchical, EM algorithm, Density-based methods-DBSCAN. Textbook 2: Chapter: 9	
UNIT – V	**09 Hours**
Unsupervised Machine Learning II: Introduction to Association Analysis, Apriori Algorithm, Advantages and Disadvantages of Apriori Algorithm. Introduction to Dimensionality Reduction, Principal Component Analysis, Linear Discriminant Analysis, Singular Value Decomposition. Textbook 2: Chapter: 9, 11	

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

· **AI-Augmented Concept Learning:** Integrate AI tools to help students explore core concepts of Machine Learning such as supervised and unsupervised learning through interactive simulations, visualizations, and adaptive explanations.

· **Human vs Machine Intelligence Reflection:** Encourage students to critically compare human learning with machine learning, fostering reflection on cognition, decision-making, and limitations of AI systems.

· **AI-Assisted Coding and Model Development:** Use AI-based coding assistants to support implementation of algorithms like KNN, SVM, and neural networks, while ensuring students understand the underlying logic and not just generated code.

· **Data-Centric Thinking with AI Tools:** Promote the use of AI tools for data preprocessing, cleaning, and visualization, helping students understand data quality issues and their impact on model performance.

· **Ethical AI Awareness and Responsible Usage:** Integrate discussions on bias, fairness, transparency, and accountability in machine learning models, especially in classification and clustering applications.

· **Experiential Learning through AI Projects:** Encourage project-based learning where students apply ML techniques to real-world datasets (healthcare, finance, etc.) using AI-assisted platforms.

Applications:

1. Healthcare Diagnosis & Prediction
2. Recommendation Systems
3. Fraud Detection in Banking

4. Email Spam Filtering
5. Stock Market Prediction
6. Customer Segmentation & Marketing
7. Predictive Maintenance in Industry

Laboratory Component (1 Credit)

The lab emphasizes hands-on implementation, visualization, and AI-assisted debugging.

Lab Experiments

1. Develop a program to implement the K-Nearest Neighbor algorithm to classify the iris data set. Print both correct and wrong predictions.
2. Apply the working of Naïve Bayes using a suitable dataset.
3. Analyze the working of perceptron and error functions using suitable datasets.
4. Build an Artificial Neural Network by implementing the Backpropagation algorithm.
5. Construct a code for Linear & Logistic Regression.
6. Apply EM algorithm to cluster a set of data stored in a .CSV file. Use the same data set for clustering using the k-means algorithm. Compare the results of these two algorithms and comment on the quality of clustering.
7. Implement Apriori algorithm by using suitable market basket dataset.
8. Apply PCA and any classification algorithm on suitable datasets.

Mini Project Component

- **Industry-Driven Projects**

Some sample projects described below simulate the **real software engineering problems**.

Project 1 – Student Performance Prediction System

- **Problem:** Predict student grades based on study hours, attendance, and previous scores..
- **Concepts used:** Supervised learning, regression, data preprocessing
- **Industry relevance:** Used in EdTech platforms for personalized learning analytics.

Project 2 – House Price Prediction

- **Problem:** Predict house prices based on features like location, size, and amenities.
- **Concepts used:** Linear regression, feature selection.
- **Industry relevance:** Real estate analytics and pricing tools.

Course Outcome

Course Outcomes (COs)	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Understand fundamental concepts of machine learning and apply basic data preprocessing techniques to prepare datasets for analysis.	L2
2	Apply classification algorithms to build and evaluate predictive models for real-world problems.	L3

3	Develop regression and basic neural network models to analyze and predict continuous and categorical outcomes.	L3
4	Implement clustering techniques to discover hidden patterns and group data effectively.	L3
5	Apply association rule mining and dimensionality reduction techniques to extract meaningful insights from complex datasets.	L3

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
CO1	3	2	2	2	2						2	3	2
CO2	3	3	2	2	3						2	3	2
CO3	3	3	2	3	3						2	3	2
CO4	2	3	2	3	3						2	3	2
CO5	2	3	2	3	3						2	3	2

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

Suggested Tools:

- **Jupyter Notebook.**
- **Google Colab.**
- **Scikit Learn.**
- **Numpy & Pandas.**
- **TensorFlow**

TEXT BOOKS:

1. Tom M. Mitchell, -Machine Learning, McGraw-Hill Education (India) Private Limited, 2013.
2. Machine Learning, Saikat Dutt, Subramanian Chandramouli, Amit Kumar Das, Fifth Edition 2020, Pearson Publisher.

REFERENCE BOOKS:

1. Ethem Alpaydin, “Introduction to machine learning”, second edition, PHI publication, 2010
2. Shai Vaingast, “ Beginning Python Visualization Crafting Visual Transformation Scripts”, Apress, 2nd Edition, 2014.
3. Peter Flach, —Machine Learning: The Art and Science of Algorithms that Make Sense of Data, First Edition, Cambridge University Press, 2012.
4. Stephen Marsland, —Machine Learning – An Algorithmic Perspective, Second Edition, Chapman and Hall/CRC Machine Learning and Pattern Recognition Series, 2014.
5. John L. Semmlow, Benjamin Griffel, Biosignal and Medical Image Processing, 3rd Ed, CRC Press, 2014.
6. Pattern recognition and machine learning by Christopher Bishop, Springer Verlag, 2006
7. Stephen Marsland, - Machine Learning: An Algorithmic Perspective, Second Edition, 2014.

E-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

****

PRODUCT DESIGN & LEAN START-UP STRATEGY

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

SEMESTER – V

Subject Code	:	24CT3504	**Credits**	:	03
Hours / Week	:	02 Hours	**Total Hours**	:	45Hrs (30 Hrs T+30 Hrs P)
L–T–P–S	:	2–0–0–2			

Course Learning Objectives:
This Course will enable students to:
1. To understand the fundamentals of product design and product development processes.
2. To analyze customer needs and develop product concepts and specifications.
3. To apply industrial design and manufacturing considerations in product development.
4. To understand lean start-up principles and business model development strategies.
5. To develop innovative and sustainable start-up solutions using business model design processes.

Broad Course Description

This course introduces students to the principles of product design and lean start-up strategy for developing innovative and sustainable products and businesses. The course covers customer-centric product development, concept generation, industrial design, manufacturing considerations, and lean start-up methodologies. Students will learn business model development, product validation, prototyping, and strategic innovation processes to transform ideas into scalable technology-driven ventures.

UNIT – I	**06 Hours**

Introduction
Introduction to Product Design, Product Development Process, Product Development Organization.
Product Planning, Opportunity Identification, Identifying Customer Needs.

	06 Hours

Product Specifications & Concept Development
Product Specifications, Concept Generation Techniques, Concept Selection, Concept Testing, Product Architecture.

UNIT – III	**06 Hours**

Industrial Design & Manufacturing

Industrial Design, Design for Environment (DfE),Design for Manufacturing (DfM), Supply Chain

Considerations.

UNIT – IV	**06 Hours**

Business Model Part I
Leap, Test, Measure & Pivot, Batch, Grow, Adapt, Innovate, Business Model Definition, 9 Building Blocks, Business Model Canvas, Patterns

UNIT – V	**06 Hours**

Business Model Part II
Design – Customer insights, Ideation, Prototyping, Story Telling
Strategy – BM environment, Evaluating BM's, Manage
Process – Business Model Design Process

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

- **AI-assisted Customer Need Analysis:** Use AI tools to identify customer requirements, analyze market trends, and validate product opportunities while ensuring ethical and unbiased decision-making.

- **Intelligent Product Concept Development:** Apply AI-driven techniques for concept generation, product specifications, and prototype evaluation to improve innovation and design efficiency.
- **AI-enabled Product Design & Manufacturing:** Integrate AI tools in industrial design, Design for Manufacturing (DfM), and supply chain optimization to support sustainable and resource-efficient product development.
- **Smart Business Model Innovation:** Utilize AI-based analytics for Business Model Canvas creation, market fit analysis, and start-up strategy evaluation with transparent and explainable assumptions.
- **Responsible AI in Start-up Ecosystems:** Evaluate ethical, legal, and societal implications of AI in product development and lean start-up processes, ensuring sustainability, privacy, fairness, and digital trust.

Suggested Tools:

- Trello
- Notion
- FigJam
- DALL·E
- Canvanizer
- Figma

Applications:

- **Customer Need Analysis and Market Research:** Using AI and analytics tools to identify customer pain points, market trends, and product opportunities.
- **Product Concept Development and Prototyping:** Designing wireframes, mock-ups, and Minimum Viable Products (MVPs) for validating start-up ideas.
- **Business Model Design and Validation:** Creating and evaluating Business Model Canvas and Lean Start-up strategies for scalable ventures.
- **Product Analytics and Market Fit Analysis:** Tracking customer behaviour, product usage, and feedback to improve product-market fit.
- **Sustainable Product Development:** Applying Design for Environment (DfE) and responsible innovation principles in product design.
- **AI-driven Innovation and Decision Making:** Using Generative AI and predictive analytics for idea generation, product recommendations, and start-up strategy optimization.

Project Component

- Develop a system that analyses customer feedback, market trends, and user preferences to identify potential product opportunities using AI and analytics tools.
- Design and prototype an MVP for a digital product or start-up idea and validate it through customer feedback and usability testing.
- Create a digital platform that assists start-ups in generating and evaluating Business Model Canvas components using AI-based recommendations.

- Design a system that helps start-ups generate pitch decks, value propositions, and strategic insights using Generative AI tools.

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
CO1	Describe lean start-up concepts such as build-measure-learn and pivot strategies	L2
CO2	Apply customer need analysis and concept development techniques for product design.	L3
CO3	Analyze industrial design, manufacturing, and sustainability considerations in product development.	L4
CO4	Develop business models using lean start-up methodologies and business model canvas frameworks.	L5
CO5	Design innovative and scalable start-up solutions through prototyping and business model strategies.	L5

Table: Mapping Levels of COs to POs / PSOs

COS	Program Outcomes (POs)												Program Specific Outcomes (PSOs)	
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO 1	PSO 2
CO1	3	2			3				2	2	2	2		3
CO2	3	2	2		3				2	2	2	2		3
CO3	3	3	3	2	3				2	2	2	2		3
CO4	3	3	3	2	3				2	2	2	2		3
CO5	3	3	3	3	3				2	2	2	2		3

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

Text Books:

1. Karl T. Ulrich, Steven D. Eppinger, Maria C. Yang, Product Design and Development, 7th Edition, McGraw Hill Education, 2020.
2. Eric Ries, The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses, 1st Edition, Crown Business Publishers, 2011.

Reference Books:

1. Alexander Osterwalder, Yves Pigneur, Business Model Generation, 1st Edition, Wiley Publications, 2010.
2. Jake Knapp, John Zeratsky, Braden Kowitz, Sprint: How to Solve Big Problems and Test New Ideas in Just Five Days, 1st Edition, Simon & Schuster, 2016.
3. Donald A. Norman, The Design of Everyday Things, Revised Edition, Basic Books, 2013.

E-Resources:

1. https://pdxscholar.library.pdx.edu/cgi/viewcontent.cgi?article=3218&context=etm_studentprojects
2. https://leanstartup.co/resources/articles/eric-ries-unveils-exclusive-lean-startup-online-course/
3. https://www.coursera.org/learn/wiley-product-management-and-lean-startup-principles
4. https://nptel.ac.in/courses/112107217

Activity Based Learning (Suggested Activities in Class)

1. Deliver a startup pitch including value proposition, business strategy, and prototype demonstration.
2. Real-world case studies for successful start-up companies to understand product development, innovation, and scaling strategies.

****

Operating Systems for GPUs

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

SEMESTER – IV

Subject Code	:	24CT3505	**Credits**	:	04
Hours / Week	:	03 Hours + 02 Hours	**Total Hours**	:	60 Hrs (45 Hrs L + 15 Hrs P)
L–T–P-J	:	3–0–2-0	**CIE Marks**	:	60 + 40 Marks

Prerequisites: Linux Fundamentals, Programming in C Language, Python

Course Learning Objectives:

This Course will enable students to:

1. Explain and differentiate core operating system concepts such as processes, threads, scheduling algorithms, memory management, file systems, and synchronization.
2. Apply scheduling principles to GPU job scheduling, mapping CPU algorithms to CUDA kernel launches and evaluating NVIDIA MPS for concurrent AI workloads.
3. Analyze memory management techniques (paging, segmentation, TLB) in the context of CUDA's memory hierarchy and unified memory for efficient AI training.
4. Evaluate file system structures (ext4, NTFS, journaling, B+ trees) for storing and checkpointing large-scale LLM weights on high-performance storage systems.
5. Implement synchronization mechanisms (mutexes, semaphores, monitors) to design concurrent inference servers and avoid deadlocks in GPU resource contention.
6. Integrate OS-level concurrency and containerization (Python threading, multiprocessing, asyncio, namespaces, cgroups) to build scalable, isolated AI workloads.
7. Utilize observability tools (strace, perf, NVIDIA NSight Systems) to profile, monitor, and optimize end-to-end inference pipelines in distributed AI systems.

Broad Course Description

This course bridges core operating system principles with modern AI infrastructure, showing how processes, threads, scheduling, memory management, file systems, and synchronization directly impact large-scale AI workloads. Students will learn classical concepts such as CPU scheduling algorithms, paging, segmentation, journaling, and concurrency, while applying them to GPU job scheduling, CUDA memory hierarchy, NVMe-based checkpointing, and thread pools for inference servers. The curriculum emphasizes practical tie-ins like simulating GPU schedulers, mapping CPU scheduling to CUDA kernel launches, and analyzing NVIDIA MPS for concurrent streams. It also explores container internals, Python concurrency models, and observability tools for profiling end-to-end inference pipelines. By integrating textbook theory with case studies on LLM batch inference, distributed training, and GPU resource management, the course equips learners to design efficient, scalable, and resilient AI systems grounded in OS fundamentals.

UNIT – I	**09 Hours**

Processes, Threads, Scheduling Concepts: Processes vs. threads, scheduling algorithms (FCFS, SJF, Round Robin, Priority).
Textbook-1 Coverage: 3.1–3.3, 4.1, 4.2, 4.4, 5.1–5.3.
Map CPU scheduling to GPU kernel launches.
Compare Round Robin vs. NVIDIA MPS scheduling for concurrent CUDA streams.
Case study: Batch inference scheduling for LLM workloads.
Textbook-2 Coverage: 3.1–3.2, 3.4, 3.7

UNIT – II	**09 Hours**

Memory Management: Concepts: Paging, segmentation, virtual memory, TLB.

Textbook-1 Coverage: 8.1–8.6. GPU global vs. shared vs. texture memory. CUDA Unified Memory bridging CPU–GPU address spaces. TLB relevance in GPU page migration. Textbook-3 Coverage: 5.4, 5.5	
UNIT – III	**09 Hours**
File Systems Concepts: ext4, NTFS, journaling, B+ tree indexing. Textbook-1 Coverage: 10.1–10.6. Checkpointing large LLM weights (multi-TB). Journaling for crash recovery in distributed training.	
UNIT – IV	**09 Hours**
Synchronization Concepts: Mutex, semaphore, monitors, deadlock. Textbook-1 Coverage: 6.1–6.7. Avoiding deadlocks in GPU resource contention. Monitor-based synchronization for async inference pipelines. Textbook-4 Coverage: 13.6	
UNIT – V	**09 Hours**
OS for AI Infrastructure Concurrency in Python: threading, multiprocessing, asyncio, concurrent.futures → async LLM inference endpoint. Container Internals: namespaces, cgroups, copy-on-write → Docker isolation for AI workloads. GPU Scheduling: NVIDIA MPS enabling concurrent CUDA contexts on DGX B200. Observability: strace, perf, NVIDIA NSight Systems → profiling end-to-end inference workloads. Source: https://learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+S-OV-16+V1	

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

The integration of AI systems is framed through classical operating system concepts. Traditional OS mechanisms—such as processes, threads, scheduling, memory management, file systems, and synchronization—are directly mapped to modern AI infrastructure. For example:

- Scheduling: CPU scheduling algorithms are compared to GPU kernel launches and NVIDIA MPS handling of concurrent CUDA streams.
- Memory management: Paging and TLBs are linked to CUDA's unified memory and GPU page migration.
- File systems: Structures like journaling and B+ trees are contextualized through large-scale LLM checkpointing on NVMe RAID.
- Synchronization: Classical primitives are applied to thread pools serving concurrent inference requests.
- Concurrency & observability: OS concurrency models, container internals, GPU scheduling, and profiling tools are integrated to optimize AI inference workloads.

Applications:

The course bridges academic theory with practical AI infrastructure engineering. Students learn to apply classical Operating systems concepts to modern AI workloads and GPU Scheduling.

Exp. No.	Division of Experiments	List of Experiments
1	Process Management	Write a C program to create a new process using **fork()**, replace the process image using **exec()**, and synchronize parent and child using **wait(),** to demonstrate **process creation vs. thread creation** using **fork()** and **POSIX pthreads**, and compare execution time.
2	CPU Scheduling	Write a C program to simulate **First Come First Serve (FCFS)** CPU scheduling algorithm, **Shortest Job First (SJF)** and **Round Robin** scheduling algorithms and compare their performance metrics.
3	Memory Management	Write a C program to simulate **Least Recently Used (LRU)** page replacement algorithm and compare its performance with **FIFO** page replacement algorithm.
4	File Systems	Write a C program to implement basic file operations (**create, delete, rename, list files**) and demonstrate directory management using Linux system calls and managing files using Linux system calls (**open(), read(), write(), close()**) to copy the contents of one file to another and analyze file access permissions.
5	Process Synchronization	Write a C program to solve the **Producer–Consumer Problem** using POSIX semaphores and threads, also implement the **Reader–Writer Problem** using mutexes and semaphores. Give a choice to the user to choose which one to execute.
6	Deadlock Management	Write a C program to implement the **Banker's Algorithm** for deadlock avoidance.
Suggested Mini Case Study / Demonstration Experiments		
7	Python Concurrency	Develop a Python program to compare **threading, multiprocessing, and asyncio** by executing concurrent computational tasks and measuring execution time.
8	Containerization	Demonstrate **Linux namespaces and cgroups** using Docker containers and observe process isolation, resource allocation, and copy-on-write file system behavior.
9	Performance Monitoring	Demonstrate the use of Linux observability tools (**strace**, **perf**) to analyze system calls and process performance of a sample application.
10	GPU Memory	Study different CUDA memory models (Global, Shared, Unified Memory) using sample CUDA programs and compare execution time. *(Demonstration if GPU is available.)*
11	GPU Scheduling	Study NVIDIA Multi-Process Service (MPS) or CUDA Streams using sample programs to understand concurrent GPU execution. *(Demonstration/Case Study.)*
12	AI Infrastructure	Deploy a simple Python inference application inside a Docker container and observe CPU/memory utilization using Linux monitoring tools.

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
CO1	**Explain** the concepts of processes, threads, and CPU scheduling algorithms, and **apply** them in GPU job scheduling and AI inference workloads.	L3
CO2	**Apply** memory management techniques such as paging, segmentation, virtual memory, and TLB concepts to understand CUDA memory hierarchy and CPU–GPU unified memory management.	L3
CO3	**Demonstrate** modern file system architectures, journaling mechanisms, and indexing techniques, and relate them to checkpointing and storage reliability in large-scale AI training systems.	L3
CO4	**Apply** synchronization mechanisms including mutexes, semaphores, monitors, and deadlock handling techniques to design concurrent and resource-efficient AI applications.	L3

CO5	**Demonstrate** the use of operating system services for AI infrastructure by implementing Python concurrency, containerization, GPU scheduling, and performance profiling tools for AI workloads.	L3

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												Program Specific Outcomes (PSOs)	
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO 1	PSO 2
CO1	3	2	2	3								2	3	
CO2	3	3	3	3								2	3	
CO3	3	3	2	3								2	3	
CO4	3	3	3	3								3	3	
CO5	3	3	3	3								3	3	

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

TEXT BOOKS:

1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, Operating System Principles 8th edition, Wiley-India, 2010
2. NVIDIA Ebook 3rd-Edition-Chapter03-scalable-parallel-execution
3. NVIDIA Ebook 3rd-Edition-Chapter05-performance-considerations
4. NVIDIA Ebook 3rd-Edition-Chapter13-cuda-dynamic-parallelism

E-Resources:

https://learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+T-AC-01+V1

Activity Based Learning (Suggested Activities in Class)
Deadlock Role Play: Assign students the roles of CPU, GPU, memory, and I/O resources. Using resource cards, simulate mutexes, semaphores, and deadlock situations, then demonstrate prevention or avoidance techniques.

****

DATA SCIENCE AND VISUALIZATION

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

SEMESTER – V

Subject Code	:		**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	45 Hrs (30Hrs L + 15Hrs J)
L–T–P-J	:	2–0–0-2	**CIE Marks**	:	60 + 40 Marks

Prerequisites: Python Programming, Mathematics

Course Learning Objectives:

This Course will enable students to:

1. Understand and implement predictive modelling approaches using appropriate Python libraries.
2. Employ descriptive statistical methods to explore and prepare datasets effectively.
3. Design and apply suitable data wrangling techniques to handle and refine raw data.
4. Evaluate data from various sources to identify and develop optimal analytical solutions.
5. Apply data transformation and dimensionality reduction techniques to enhance data analysis and modelling.

Broad Course Description

Data Science and Visualization is an interdisciplinary field that combines statistics, programming, and domain knowledge to extract meaningful insights from structured and unstructured data. It involves the entire data lifecycle—data collection, cleaning, transformation, analysis, and interpretation—to support decision-making. Core concepts include predictive modeling, statistical analysis, and machine learning, where techniques are used to identify patterns, trends, and relationships within data. Tools and libraries such as Python-based ecosystems enable efficient handling of large datasets and implementation of analytical models.

Data visualization plays a crucial role in data science by transforming complex data into intuitive graphical representations such as charts, graphs, and dashboards. Visualization techniques help in exploring data (exploratory data analysis), identifying patterns, detecting anomalies, and communicating insights effectively to both technical and non-technical audiences. Methods like histograms, box plots, scatter plots, and heatmaps provide visual clarity and support better understanding of statistical properties like distribution, correlation, and variance.

UNIT – I	**06 Hours**

Basic Concepts And Python Package

Basic Concepts: Predictive Modelling, Data preparation, Importance of Data preparation, Data Cleaning, Feature selection, Data Transform, Dimensionality reduction, K-fold cross validation, Data Leakage and avoidance measure. Python Packages: NumPy, Matplotlib, pandas, SciPy, scikit, Dataframe, Loading Machine Learning data.

Textbook 1: Chapter 1-7.

UNIT – II	**06 Hours**

Descriptive Statistics And Data Preparation

Descriptive Statistics: Reasons to study Statistics, Sampling, Data Analysis Process, Mean, Median, Standard Deviation, Skewness, Kurtosis, Graphical Representation-Box Plots, Pivot Table, Heat Map Correlation, Statistics–ANOVA. Data Preparation: Need for Data Pre-processing, Data Transforms, and Rescale Data Standardize Data, Normalize Data, Binarize Data, Univariate and Bivariate Data, Recursive Feature Elimination, Principal Component Analysis.

Textbook 2: Chapter 1, 2, 3, 10.

UNIT – III	**06 Hours**

Data Cleaning and Feature Selection

Data Cleaning: Basic data cleaning, Outlier Identification and Removal, how to Mark and Remove Missing Data, Statistical Imputation, KNN Imputation, Iterative Imputation. Feature Selection:

Statistics for feature selection, Methods for categorical input, Methods for Numerical input, Select Features for Numerical Output, RFE for Feature Selection, Significance of feature selection. Textbook 1: Chapter 2, 3, 8.	
UNIT – IV	**06 Hours**
Data Transform and Dimensionality Reduction: Data Transforms: Scaling data source, minmax scalar and standard scaler, Scale data with outliers, encode categorical data, Make Distributions More Gaussian, Approach for Numerical Data Distributions, Deriving new input variables. Dimensionality reduction: Techniques for Dimensionality Reduction, Linear Discriminant Analysis, PCA Dimensionality Reduction, SVD Dimensionality Reduction Textbook 1: Chapter 4, 5.	
UNIT – V	**06 Hours**
Other Transforms Transform numerical to categorical, Transform Numerical and Categorical Data, Transform the Target in Regression, Save and load the transformation, case studies for Binary classification, Multiclassification and Regression. Textbook 1: Chapter: 4, 9, 10.	

AI Integration & Authentic Intelligence: Reflection (AI² Pedagogy)

AI-Assisted Learning with Critical Reflection: Students use AI tools (e.g., code generators, AutoML platforms) to build ML models, but must critically reflect on outputs, assumptions, and limitations to ensure genuine understanding.

- **Ethical AI and Bias Awareness:** Encourages identification of bias in datasets and models, promoting fairness, accountability, and transparency in ML Applications.
- **Integration of AI Tools with Core Concepts:** AI tools are used as support systems (not replacements), ensuring students still learn algorithms like regression, classification, and clustering fundamentally.
- **Transparency in AI Usage:** Students explicitly document where and how AI tools were used in their ML workflow, ensuring academic integrity and authentic learning.

Applications:

1. Finance & Banking
2. E-Commerce & Retail
3. Transportation & Logistics
4. Healthcare & Medical Diagnosis
5. Education

Mini Project Component

- **Industry-Driven Projects**

Some sample projects described below simulate the **real software engineering problems**.

Project 1 – Sales Forecasting System

- **Problem:** Forecast future sales based on historical sales data and trends.
- **Concepts used:** Time Series Analysis, Regression, Data Transformation, Visualization..
- **Industry relevance:** Retail and supply chain management.

Project 2 – Sentiment Analysis on Social Media

- **Problem:** Analyze user sentiments (positive/negative) from social media text data.
- **Concepts used:** NLP, Text Preprocessing, Classification, Visualization.
- **Industry relevance:** Marketing, brand monitoring, and public opinion analysis.

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
CO1	Illustrate predictive modeling techniques with necessary python packages.	L2
CO2	Apply descriptive statistics concepts for data preparation.	L3
CO3	Develop and examine appropriate methods for data wrangling.	L3
CO4	Analyze efficient solution for the given data sources	L4
CO5	Examine the data transformation and dimension reduction techniques on the data source.	L4

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)											Program Specific Outcomes (PSOs)	
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	3		3						2	3	2
CO2	3	2	2		3						2	3	2
CO3	3	2	2		3						2	3	2
CO4	3	2	2		3						2	3	2
CO5	2	2	2		3						2	3	2

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

Suggested Tools:

- Development Environments: Jupyter Notebook / Google Colab.
- Python Libraries: NumPy, Pandas, Matplotlib, Seaborn, SciPy, Sckit-learn.
- Data Visualization Tools: Tableau, Microsoft Power BI, Plotly.

TEXT BOOKS:

1. Jason Brownlee, "Data Preparation for Machine Learning" 2020.
2. Roxy Peck, Chris Olsen and Jay Devore, "Introduction to Statistics & Data Analysis "3rd Edition Thomson Higher Education,2015. ISBN: 1305445961, 9781305445963.

REFERENCE BOOKS:

1. Andrew Park ,"Data Science For Beginners",2021. ISBN: 9781914167997.
2. Norman Matloff, "Probability and Statistics for Data Science", CRC Press,2019. ISBN: 9780429687112.

E-RESOURCES:

1. Data Science for Engineers:

   https://digimat.in/nptel/courses/video/106106179/L01.html
2. Statistics for Data Science: https://www.youtube.com/watch?v=V5fqShLVpoI

Activity Based Learning (Suggested Activities in Class)

1. Student seminars (on topics of the syllabus and related aspects (individual activity).
2. Quiz (on topics where the content can be compiled by smaller aspects and data (Individuals or groups as teams).
3. Study projects (by very small groups of students on selected local real-time problems about the syllabus or related areas. The individual participation and contribution of students shall be ensured (team activity).

****

Cyber Security Fundamentals

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

SEMESTER – V

Subject Code	:		**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	45Hrs (30Hrs L + 15 Hrs J)
L–T–P-J	:	2–0–0-2	**CIE Marks**	:	60 + 40 Marks

Course Learning Objectives:

This Course will enable students to:

1. Understand security principles, CIA triad, and risk management.
2. Recognize threat landscapes—malware, phishing, DoS.
3. Explain network, system, and application security layers.
4. Illustrate authentication, authorization, and access control models.
5. Relate cybersecurity foundations to modern AI-enabled monitoring.

Broad Course Description

Introduces core principles of cybersecurity. Topics include threat taxonomy, security controls, firewalls, crypto basics, and incident response. It lays the groundwork for higher courses on cryptography, IoT security, and blockchain.

UNIT – I	**06 Hours**

INTRODUCTION TO CYBER SECURITY:

Information security fundamentals – CIA triad, authenticity, non-repudiation. (Text Book 1 : Ch. 1 1.1–1.5)

Threats, vulnerabilities, exploits, and attack vectors. Security policies, standards (ISO 27001, NIST). Cyber ethics, legal frameworks (IT Act 2000/Amendments). Cyber warfare and critical infrastructure protection. Security architecture layers – physical, logical, organizational.

(Text Book 2 : Ch. 2 2.1–2.3)

Risk management process – identification, analysis, control.

(Text Book 1 : Ch. 3 3.1–3.3)

Case Study: Analysis of recent ransomware incident.

UNIT – II	**06 Hours**

NETWORK SECURITY & PROTOCOL PROTECTION:

OSI vs TCP/IP models – security implications. Common network attacks: sniffing, spoofing, DoS, DDoS, MITM. (Text Book 2 : Ch. 6–7)

Firewalls: packet-filtering, stateful inspection, proxy and next-gen firewalls.

Intrusion Detection and Prevention Systems (IDS/IPS) – architectures and signatures. Secure network protocols – HTTPS, SSH, TLS, VPNs. Wireless security standards – 802.11i/WPA 2/3, EAP. Network segmentation and zero-trust architecture basics.

(Text Book 1 : Ch. 4-7) (Text Book 2 : Ch. 7-10)

Case Study: Analysis of OWASP Top 10 vulnerabilities.

UNIT – III	**06 Hours**

SYSTEM & APPLICATION SECURITY:

OS security concepts – kernel hardening, patch management. Access control models (DAC, MAC, RBAC, ABAC). Malware taxonomy – virus, worm, ransomware, spyware. Application vulnerabilities – buffer overflow, SQL injection, XSS, CSRF. Secure coding principles and SDLC integration (DevSecOps). Mobile and cloud application security. Virtualization and container security concepts (Docker hardening).

(Text Book 1 : Ch. 8-12) (Text Book 2 : Ch. 11-13)

UNIT – IV	06 Hours
INCIDENT MANAGEMENT & FORENSICS: Incident response lifecycle – preparation, detection, containment, recovery. Log management and SIEM systems (Splunk, ELK stack). Digital forensics – data acquisition, preservation, examination, reporting. Forensic imaging tools (FTK Imager, Autopsy). Legal admissibility of digital evidence. Disaster recovery and business continuity planning. Security metrics and Key Risk Indicators (KRI). (Text Book 1 : Ch. 14-18) (Text Book 2 : Ch. 14-16)	
UNIT – V	**06 Hours**
EMERGING TRENDS AND AI-DRIVEN CYBER DEFENSE: Cloud security models (SaaS, PaaS, IaaS). Identity and Access Management (IAM) with Azure AD / Okta. Secure configuration management and compliance automation. AI in cybersecurity – ML for anomaly detection, phishing identification. Ethical and privacy implications of AI-based defense. Quantum computing impact on cybersecurity. Security in IoT and industrial systems. (Text Book 1 : Ch. 19-25) (Text Book 2 : Ch. 17-18) **Case Study:** AI vs AI attacks (adversarial ML).	

AI Integration & Authentic Intelligence: Reflection (AI² Pedagogy)

- Sustainable, ethical monitoring practices.
- Automation in threat detection using AI models.
- AI-driven threat detection and traffic classification.
- Automated static code analysis using ML classifiers.
- Predictive forensics and threat prioritization using AI

Applications

1. Enterprise risk management and compliance using CIA triad, ISO 27001, and NIST standards.
2. Network defense through firewalls, IDS/IPS, VPNs, and zero-trust architecture.
3. Secure software development with DevSecOps and protection against OWASP Top 10 vulnerabilities.
4. Incident response and digital forensics using SIEM tools, evidence preservation, and recovery planning.
5. AI-driven cyber defense for anomaly detection, phishing prevention, IoT security, and quantum-resistant cryptography.

Mini Project Component

Project 1 – AI-Driven Intrusion Detection System (IDS)

- **Problem:** Configure a simple IDS using Snort/Wireshark and integrate AI² for real-time anomaly detection and traffic classification.**Concepts used:** Packet sniffing, rule-based detection, machine learning classifiers, anomaly detection.

- **Industry relevance:** Network security monitoring and AI-powered SOC platforms (Google Cloud Security, Amazon GuardDuty).

Project 2 – Automated Static Code Analysis for OWASP Top 10

- **Problem:** Perform automated static code analysis using AI² to detect OWASP Top 10 vulnerabilities in Applications.
- **Concepts used:** Secure coding practices, ML-based code parsing, vulnerability classification, DevSecOps automation.
- **Industry relevance:** Application security audits and secure DevOps pipelines (GitHub Advanced Security, AWS CodeGuru).

LAB EXERCISES

1. Installation and configuration of firewalls and packet analyzers.
2. Malware behavior analysis using sandboxing tools.
3. Identify SQL Injection and implement secure input validation.
4. GIAC-style capture-the-flag (CtF) simulation.
5. Report on forensic log analysis for breach incident.

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
CO1	Explain and apply fundamental cybersecurity concepts and policies.	**L2**
CO2	Identify and analyze network/system vulnerabilities.	**L3**
CO3	Demonstrate security mechanisms and mitigation strategies.	**L3**
CO4	Investigate security incidents using forensic approaches.	**L3**
CO5	Employ AI tools for intelligent cyber defense in emerging environments.	**L4**

Table: Mapping Levels of COs to POs / PSOs

CO S	Program Outcomes (POs)											Program Specific Outcomes (PSOs)	
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO1 0	PO11	PSO1	PSO2
CO1	3	2										2	
CO2	2	3	3		2							3	
CO3	2	2	3	2	2							2	
CO4	3	2										2	
CO5	2	2	3		3							3	

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

TEXT BOOKS:

1. William Stallings – *Computer Security: Principles and Practice*, 8e, Pearson, 2023.

2. Mark Ciampa – *Security+ Guide to Network Security Fundamentals*, 7e, Cengage, 2022

REFERENCE BOOKS:

1. Charles Pfleeger and Shari Lawrence Pfleeger – *Security in Computing*, 5e.
2. Matt Bishop – *Computer Security Art and Science*, 2e, Addison Wesley.

NVIDIA Course Mapping (Industry Alignment)

NVIDIA DLI Course / Content	Mapped Units	Relevance
Fundamentals of Cybersecurity Simulations w ith Morpheus	II, V	AI-driven security and threat analyti cs
Deep Learning for Anomaly Detection	V	Behavioral analytics and intrusion patterns
Edge AI with Jetson	V	IoT security gateway intelligence

E-RESOURCES:

1. nptel.ac.in
2. coursera.org
3. edx.org
4. cybrary.it
5. udemy.com
6. developer.nvidia.com
7. cse29-iiith.vlabs.ac.in

Activity-Based Learning (Suggested Activities in Class)

1. Case Study Discussion: Analyze a recent cyber-attack (ransomware or data breach) and identify weaknesses and preventive controls.
2. Threat Modeling Exercise: Students design a simple network architecture and mark pot ential entry points and countermeasures.
3. Hands-on Demo: Capture and inspect network traffic using Wireshark to detect anomal ies or malicious packets.
4. Role Play (SOC Simulation): Groups act as Security Operations Center analysts respon ding to a simulated incident.
5. Ethical Debate: Discuss “AI in Cybersecurity – Balancing Privacy and Protection” to promote responsible security thinking.

****

Interactive Art and Creative Coding

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

SEMESTER – V

Subject Code	:		**Credits**	:	03
Hours / Week	:	04 Hours	**Total Hours**	:	45Hrs (30Hrs L+15 Hrs J)
L–T–P-J	:	2–0–0-2	**CIE Marks**	:	60 + 40 Marks

Course Learning Objectives:

This Course will enable students to:

1. Explore the intersection of art, design, and programming through computational creativity.
2. Develop skills in creative coding using Processing, p5.js, and generative art frameworks.
3. Design interactive audiovisual experiences using code-driven animation and multimodal user interaction.
4. Understand generative art principles and apply them to create expressive, publishable digital artworks.

Broad Course Description

Interactive Art and Creative Coding fuses programming with visual art, design, music, and human-computer interaction. Using Processing and p5.js as primary environments, students create dynamic visual compositions through generative patterns, particle systems, Perlin noise, physics simulations, and algorithmic design. The course covers interactive design with mouse, keyboard, camera, and microphone inputs, AI-based interactions using ml5.js (PoseNet, FaceAPI), sound synthesis with Tone.js, 3D graphics via Three.js, and AI art generation tools (Stable Diffusion, RunwayML). Students build a portfolio of published creative coding projects demonstrating both technical skill and artistic vision.

UNIT – I	**06 Hours**

Foundations of Creative Coding: Introduction to creative coding – history and movements (Casey Reas, Vera Molnár, John Maeda). Processing IDE and p5.js environment setup. Drawing primitives – shapes, colors, coordinate system. Typography and image manipulation in code. Variables, conditionals, loops in a creative context. Functions for reusable generative components. Randomness – random() and noise() for controlled unpredictability. (Ref: Casey Reas & Ben Fry, Getting Started with Processing Ch. 1–4)

Textbook Reference:

• Casey Reas & Ben Fry, Getting Started with Processing, 2nd Ed – Ch. 1–4 (Maker Media, 2015)

• p5.js Reference Documentation (p5js.org/reference)

UNIT – II	**06 Hours**

Animation & Motion Systems: Animation principles in code – frame rate, easing, interpolation. Object-Oriented approach for animated entities. Particle systems – creation, emission, and lifecycle. Physics simulation – gravity, friction, spring forces. Trigonometric motion – circular, pendulum, wave animations. PVector/createVector mathematics for natural movement. Perlin noise for organic motion, terrain, and landscapes. (Ref: Shiffman, The Nature of Code Ch. 1–4)

Textbook Reference:

• Daniel Shiffman, The Nature of Code – Ch. 1–4 (natureofcode.com, 2024)

• Casey Reas & Ben Fry – Ch. 5–7

UNIT – III	**06 Hours**

Interactivity & Input Devices: Mouse and keyboard interaction – events, states, and gestures. Touch interaction for mobile p5.js sketches. Camera input and computer vision with ml5.js – PoseNet, FaceAPI, BodyPix. Microphone input – audio analysis, FFT, frequency visualization. Webcam-based interactive installations. Game controller and MIDI input integration. Building interactive web-based art experiences with A-Frame basics. (Ref: Shiffman, The Nature of Code Ch. 8–9)

Textbook Reference:

• Daniel Shiffman, The Nature of Code – Ch. 8–9

• ml5.js Documentation (ml5js.org)

UNIT – IV	06 Hours

Generative Art & Algorithmic Design: Generative systems – L-Systems, Fractal geometry (Mandelbrot, Julia sets). Cellular Automata – Conway's Game of Life and rule-based systems. Algorithmic patterns – Voronoi diagrams, Truchet tiles, flow fields. Data-driven art – visualizing real datasets creatively. 3D graphics in Processing and WebGL with p5.js. Three.js basics for 3D web art. Introduction to GLSL shaders for GPU-accelerated visuals. (Ref: Matt Pearson, Generative Art Ch. 1–5)

Textbook Reference:

• Matt Pearson, Generative Art: A Practical Guide using Processing – Ch. 1–5 (Manning, 2011)

• Daniel Shiffman, The Nature of Code – Ch. 5–7

UNIT – V	06 Hours

Sound, AI Art & Exhibition: Sound synthesis and music programming with Tone.js. Audiovisual synchronization – music-reactive visuals using FFT. Introduction to AI-generated art – DALL-E, Stable Diffusion, Runway ML. Integrating AI-generated assets into interactive sketches. Creative workflow – ideation, prototyping, refining, and critique. Publishing web-based art to GitHub Pages and OpenProcessing. Ethics of AI art – authorship, originality, attribution, and copyright. (Ref: Tone.js Documentation)

Textbook Reference: Tone.js Official Documentation (tonejs.github.io), Runway ML Documentation (runwayml.com/docs)

NVIDIA DLI Free Course Mapping (Validated & Justified)

Validation Source: NVIDIA DLI Official Training Catalog (August 2024 PDF) + Live URL verification (April 2026). All courses below are confirmed FREE self-paced courses accessible at **learn.nvidia.com**. Only FREE courses are listed; paid instructor-led workshops excluded.

Unit	NVIDIA DLI Free Course	Category	Duration	Cert.	Verified URL	Mapping Justification
Un	**Generative AI Explained**	Generative AI	1 Hr	—	courses.nvidia.com/course-v1:DLI+S-FX-07+V1	Essential conceptual foundation for AI art tools (DALL-E, Stable Diffusion, Runway ML) covered in Unit 5 — students must

it 5						understand how generative AI works before critically using, integrating, and evaluating AI art tools.
Unit 5	**Generative AI With Diffusion Models**	Generative AI / Image	2 Hrs	—	learn.nvidia. com/courses/ course-detail?course _id= course-v1:DLI+T-FX-11+V1	Deep dive into diffusion model-based image generation (DDPM, score matching, DALL-E concepts) — directly supports Unit 5's Stable Diffusion and AI art generation module with hands-on
						implementation context.
Unit 3	**Building a Brain in 10 Minutes**	Deep Learning	10 Min	—	courses.nvidi a.com/ course-v1:DLI+T-FX-01+V1	10-minute neural network introduction — bridges the gap for art students encountering ml5.js AI (PoseNet, FaceAPI) in Unit 3, providing conceptual grounding for how AI interprets body and face inputs.
Units 3–4	**Getting Started With Deep Learning**	Deep Learning	2 Hrs	—	courses.nvidi a.com/ course-v1:DLI+S-FX-01+V1	Introductory deep learning course with TensorFlow/Keras and CNNs — foundational for students wanting to understand how ml5.js models work under the hood and how to fine-tune them for creative installations.
	Synthetic Tabular Data	Generative AI			learn.nvidia. com/cours	Synthetic data generation using Transformer

Unit 4	**Generation Using Transformers**		2 Hrs	—	es/ course-detail?course_id=course-v1:DLI+T-FX-13+V1	models — connects to Unit 4's data-driven art module by showing how AI can generate structured datasets that can be visualized and sonified creatively.

AI Integration & Authentic Intelligence: Reflection (AI² Pedagogy)

- **AI as Creative Collaborator:** Use DALL-E and Midjourney as inspiration sources and integrate AI-generated assets into interactive p5.js sketches.
- **Reflective Authorship:** Compare AI-generated art with student code-generated work; critically examine creativity, intent, originality, and attribution.
- **Data-Driven Storytelling:** Visualize real datasets creatively with p5.js; reflect on how data selection and framing shapes narrative and perception.
- **Computational Aesthetics:** Discuss what makes algorithmic art beautiful; analyze how human aesthetic judgment guides the development and evaluation of AI art outputs.
- **Interactive AI Experiences:** Build web experiences using ml5.js PoseNet responding to body movement; reflect on privacy, consent, and bodily data in interactive installations.

Applications:

- Digital Art Installations and Museum Exhibitions
- Music Visualization and Live Audiovisual Performance
- Interactive Brand Experiences and Advertising
- Data Visualization and Infographic Design
- Game UI/UX Prototyping and Motion Design

Laboratory Component (1 Credit)

The lab emphasizes hands-on implementation, visualization, and AI-assisted debugging.

Lab Experiments

- Create a generative poster using loops, randomness, and algorithmic color palettes in p5.js.
- Build an animated particle system responding to mouse movement with simulated gravity and friction.
- Implement a Perlin noise-based landscape/terrain generator with animated camera movement.
- Create an interactive kaleidoscope pattern responding to mouse position and click events.
- Build a music visualizer reacting to microphone input using FFT frequency band analysis.
- Implement a PoseNet-based sketch where user body movement controls

visual elements in real time.

- Design a Cellular Automata simulation (Conway's Game of Life) with interactive rule modification.
- Create a fractal explorer (Mandelbrot/Julia set) with zoom, color mapping, and parameter controls.
- Build a 3D rotating abstract sculpture using Processing 3D mode or WebGL in p5.js.
- Design and publish a complete interactive web art experience on OpenProcessing or GitHub Pages.

Mini Project Component

- **Industry-Driven Projects**

Some sample projects described below simulate the real software engineering problems.

Project 1 – Generative Art Series

Problem: Five algorithmically generated artworks with a unifying theme, each with distinct interactive elements.

Concepts used: p5.js, Generative algorithms, Color theory, Randomness

Industry relevance: Digital art galleries, NFT platforms, design portfolios.

Project 2 – Music-Reactive Visual Performance

Problem: Real-time audiovisual performance tool generating synchronized visuals from live or recorded music.

Concepts used: Tone.js, FFT analysis, Particle systems, Timing

Industry relevance: Live performance, music production, entertainment.

Project 3 – Body-Tracking Interactive Installation

Problem: Webcam captures body movement via PoseNet, triggering generative visual and sound outputs.

Concepts used: ml5.js PoseNet, Interactive design, Sound synthesis

Industry relevance: Museum installations, interactive advertising, therapy.

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Apply fundamental creative coding concepts to create expressive visual compositions using Processing and p5.js.	L3
2	Design and implement animated motion systems using physics simulation, particle effects, and trigonometric motion.	L3
3	Build interactive experiences that respond to user input from mouse, keyboard, camera, and microphone inputs.	L3
4	Create generative art systems using algorithmic design principles such as fractals, L-systems, and cellular automata.	L4
5	Integrate AI art tools and sound synthesis into creative coding	L5

	projects and present polished interactive artworks.	

Table: Mapping Levels of COs to POs / PSOs													
COS	Program Outcomes (POs)											Program Specific Outcomes (PSOs)	
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO 1	PSO 2
CO1	2	2	3	2	3						2	2	3
CO2	2	3	2	3	3						2	2	3
CO3	3	2	3	3	3						2	3	3
CO4	3	3	3	3	2						2	3	2
CO5	2	3	3	3	3						2	3	3

Suggested Tools:

- Processing 4.x – Creative Coding IDE
- p5.js – JavaScript Creative Library
- ml5.js – Friendly Machine Learning for Web
- Tone.js – Web Audio Framework
- Three.js – 3D JavaScript Library
- OpenProcessing.org – Share and Explore Sketches

TEXT BOOKS:

1. Daniel Shiffman, The Nature of Code, natureofcode.com, 2024.
2. Casey Reas & Ben Fry, Getting Started with Processing, 2nd Ed, Maker Media, 2015.
3. Matt Pearson, Generative Art: A Practical Guide using Processing, Manning, 2011.

REFERENCE BOOKS:

1. Philip Galanter, What is Generative Art?, NYU Research Paper, 2003.
2. John Maeda, Design by Numbers, MIT Press, 2001.

E-RESOURCES:

- https://thecodingtrain.com (The Coding Train by Daniel Shiffman)
- https://openprocessing.org
- https://p5js.org
- https://learn.nvidia.com (NVIDIA DLI Free Courses)

Activity Based Learning (Suggested Activities in Class)

- Weekly creative challenges – 'Sketch of the Week' shared on class gallery board.
- Collaborative generative art exhibition displayed publicly at end of semester.
- Art critique sessions using design vocabulary and aesthetic evaluation rubrics.
- Creative coding live demo: instructor codes a generative sketch from scratch in class.

****

Introduction to Quantum Computing

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

SEMESTER – V

Subject Code	:		**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	45 Hrs (45 hrs T)
L–T–P-J	:	3–0–0-0			

Prerequisites: Python Programming, Mathematics

Course Learning Objectives:

This Course will enable students to:

1. Understand the mathematical foundations required for quantum computing and quantum information theory.
2. Analyze qubits, quantum states, vector spaces, matrices, and operators used in quantum systems.
3. Apply principles of quantum mechanics, tensor products, and quantum state representations.
4. Design and analyze basic quantum gates and quantum circuits.
5. Understand quantum measurement, entanglement, and their applications in modern quantum technologies.

Broad Course Description

The course "Introduction to Quantum Computing" provides a comprehensive foundation in the mathematical and computational principles underlying quantum computation and quantum information processing. The course introduces learners to the concepts of qubits, quantum states, vector spaces, operators, tensor products, quantum gates, quantum circuits, quantum measurement, and entanglement. It bridges the gap between classical computation and emerging quantum technologies by combining mathematical rigor with computational intuition. Students will gain knowledge of the theoretical foundations of quantum mechanics relevant to computing and develop the ability to analyze simple quantum circuits and quantum algorithms. The course also emphasizes practical exposure to quantum programming environments and emerging applications of quantum computing in artificial intelligence, cryptography, optimization, and communication systems.

UNIT – I	**09 Hours**

Mathematical Foundations of Quantum Computing, Qubits, and quantum states

Classical Information, Information Content in a Signal, Entropy and Shannon's Information Theory, Probability Basics. The Qubit, Vector Spaces, Linear Combinations of Vectors, Uniqueness of a Spanning Set, Basis and Dimension, Inner Product, Orthonormality, Gram-Schmidt Orthogonalization, Bra-Ket Formalism, The Cauchy-Schwartz and Triangle Inequalities.

(Text Book-1: Chapter 1, Chapter 2)

UNIT – II	**10 Hours**

Matrices & Operators - Pauli operators, outer products, Representations of Operators Using Matrices, outer products & matrix representation, Matrix Representation of Operators in Two-Dimensional Spaces,

The Pauli Matrices, Hermitian, unitary & normal operators, eigenvalues and eigen vectors, characteristic equation, trace of an operator, expectation value of an operator, projection operators. (Text Book-1: Chapter 3)
UNIT – III | **08 Hours**
Postulates and Tensor Products The Heisenberg Uncertainty Principle, Polar Decomposition and Singular Values, The Postulates of Quantum Mechanics, Representing Composite States in Quantum Mechanics, Computing Inner Products, Tensor Products of Column Vectors, Operators and Tensor Products, Tensor Products of Matrices. (Text Book1: Chapter 3, Chapter 4)
UNIT – IV | **09 Hours**
Quantum Gates and Circuits: classical logic gates and circuits, single qubit quantum gates, the Hadamard gate, two qubit quantum gates- the CNOT gate, three qubit quantum gates- The Fredkin gate, The Toffoli gate, quantum circuits, universal quantum gates. (Text Book1: Chapter 8)
UNIT – V | **09 Hours**
Quantum Measurement and Entanglement: Distinguishing Quantum States and Measurement, Projective Measurements, Measurements on Composite Systems, Generalized Measurements, Positive Operator-Valued Measures, Entanglement: Bell's Theorem, Bipartite Systems and the Bell Basis, When Is a State Entangled?, The Pauli Representation, Entanglement Fidelity, Using Bell States For Density Operator Representation, Schmidt Decomposition (Text Book1: Chapter 6, Chapter 7)

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

- **AI–Quantum Computing Foundations:** Integrate fundamental concepts of artificial intelligence and quantum computing to understand how quantum principles can enhance intelligent computational systems and future AI applications.
- **Quantum Mathematical Intelligence:** Apply linear algebra, vector spaces, matrices, tensor products, and probability theory to model quantum states and computational processes required for intelligent quantum systems.
- **Quantum Information Representation:** Utilize qubits, superposition, and quantum state representations to encode and process information beyond the limitations of classical binary systems.
- **Quantum Circuit-Based Computational Thinking:** Design and analyze quantum gates and quantum circuits to develop logical reasoning and problem-solving abilities for next-generation intelligent computing systems.
- **Simulation-Driven Quantum Learning:** Employ quantum simulation platforms such as Qiskit and IBM Quantum Experience for experimentation, validation, and visualization of quantum operations and circuits.

- **Hybrid Quantum–Classical Learning Environment:** Encourage the integration of classical computational approaches with quantum paradigms to understand emerging hybrid intelligent systems and optimization strategies.
- **Experiential and Inquiry-Based Learning:** Promote active learning through hands-on circuit design, simulation exercises, case studies, and mini projects to strengthen conceptual understanding and innovation skills.
- **Ethical and Responsible Quantum Intelligence:** Develop awareness regarding ethical implications, security considerations, and responsible use of quantum technologies and intelligent computational systems.
- **Research-Oriented Quantum Exploration:** Encourage students to analyze recent advancements in quantum computing, quantum machine learning, and quantum communication through seminars, research reviews, and exploratory activities.
- **Authentic Intelligence Development (AI²):** Foster analytical reasoning, creativity, collaboration, and critical thinking skills to prepare learners for future interdisciplinary research and industry applications in quantum technologies and intelligent systems.

Suggested Tools:

Qiskit
IBM Quantum Lab
PennyLane
Google Cirq

Applications:

Students will explore applications of quantum computing in:

1. Quantum Cryptography
2. Secure Communication Systems
3. Quantum Machine Learning
4. Drug Discovery and Molecular Simulation
5. Financial Modeling and Optimization
6. Natural Language Processing
7. Cybersecurity
8. Supply Chain Optimization
9. Weather Forecasting
10. Smart Healthcare Systems

Mini Project Component

Industry-Driven Projects (IBM / Google Quantum Style)

These projects are designed to simulate real-world quantum computing and intelligent system applications using modern quantum technologies and simulation platforms.

Project 1 – Quantum Gate Simulation System

1. **Problem:** Design and simulate fundamental quantum gates and analyze their behavior on qubit states.
2. **Concepts used:** Qubits, Pauli gates, Hadamard gate, CNOT gate, quantum circuits
3. **Tools/Platforms:** Qiskit
4. **Industry relevance:** Quantum processor design, quantum hardware testing, research laboratories

Project 2 – Quantum Circuit Visualization Platform

1. **Problem:** Develop a system to create and visualize simple quantum circuits for educational and analytical purposes.
2. **Concepts used:** Quantum circuits, tensor products, measurement, quantum states
3. **Tools/Platforms:** IBM Quantum Experience
4. **Industry relevance:** Quantum education technology, simulation platforms, research training

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Explain the mathematical foundations of quantum computing, including qubits, vector spaces, entropy, probability theory, and quantum states.	L2
2	Analyze matrices, operators, eigenvalues, eigenvectors, and quantum state transformations used in quantum systems.	L4
3	Apply tensor products and postulates of quantum mechanics to represent and analyze composite quantum systems.	L3
4	Design and evaluate quantum gates and quantum circuits for solving basic computational and information processing problems.	L5
5	Analyze quantum measurement techniques, entanglement properties, and their applications in quantum information processing systems.	L4

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	3	2	2	3						2	3	2
CO3	2	3	2	2	3						2	3	2
CO4		2				2	3				2	2	
CO5	3	3	3	3	3	2	2	2	2	2	3	3	3

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

TEXT BOOKS:

1. David McMahon, **"Quantum Computing Explained",** IEEE Computer Society, WILEY Interscience
2. Noson S. Yanofsky, Mirco A. Mannucci, "**Quantum Computing for Computer Scientists**", Cambridge University Press

REFERENCE BOOKS:

1. Michael A. Nielsen. & Issac L. Chiang, "Quantum Computation and Quantum Information", South Asia Edition 2013, Cambridge University Press, Reprint 2022.

2. "Quantum Computing, A Gentle Introduction", Eleanor G. Rieffel and Wolfgang H. Polak MIT press (2014).
3. Mikio Nakahara and Tetsuo Ohmi, "Quantum Computing", CRC Press (2008). 3. N. David Mermin, "Quantum Computer Science", Cambridge (2007).
4. An introduction to Quantum Computing, Phillip Kaye, Raymond Laflamme, Muchele Mosca, Oxford University Press, 2007, ISBN-13: 978-0198570493, ISBN-10: 019857049X.

E-Resources:

- https://quantum.ibm.com
- https://qiskit.org
- https://azure.microsoft.com/en-us/products/quantum
- https://nptel.ac.in

Activity Based Learning (Suggested Activities in Class)

Activity 1: Quantum Circuit Design Activity
Activity 2: Mini Project (Quantum random number generator, Quantum password generator, Quantum circuit simulator)
Activity 3: Visualization Activity (Qubit states, Quantum circuits)

****

Product Strategy & Management

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

SEMESTER – V

Subject Code	:		**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	45 Hrs L
L–T–P-J	:	3–0–0-0	**CIE Marks**	:	60 + 40 Marks

Course Learning Objectives:

This Course will enable students to:

1. Develop a strong foundation in product strategy concepts, competitive positioning, and product management principles.
2. Enable students to understand and analyze buyer behavior and its influence on product decisions.
3. Provide knowledge of product life cycle and portfolio management for effective product planning.
4. Equip students with the skills to manage the new product development (NPD) process from idea generation to commercialization.
5. Enhance the ability to evaluate product ideas using concept testing, business analysis, and product testing techniques.

Broad Course Description

This course provides a comprehensive understanding of product strategy and management in competitive markets. It explores the theoretical foundations of product concepts, customer behavior, and market dynamics, along with practical approaches to managing product life cycles and portfolios. The course emphasizes the new product development (NPD) process, including idea generation, screening, concept testing, and commercialization decisions. Through strategic frameworks and real-world Applications, students will learn how to design, evaluate, and manage products to achieve sustainable competitive advantage and market success.

UNIT – I	**07 Hours**
Foundations of Product Strategy Competition and Product Strategy, The Product in Theory and Practice	
UNIT – II	**08 Hours**
Understanding Markets and Customers Buyer Behaviour, Market Needs and Customer Insights	
UNIT – III	**10 Hours**
Product Life Cycle and Portfolio Management Product Life Cycle in Theory and Practice, Innovation and Portfolio Balance	
UNIT – IV	**10 Hours**
New Product Development Process	

Importance, Nature, and Management of NPD, New Product Strategy, Idea Management for NPD.	
UNIT – V	**10 Hours**
Product Development and Evaluation Screening New Product Ideas, Concept Development and Testing, Business Analysis, Product Testing	

AI Integration & Authentic Intelligence: Reflection (AI² Pedagogy)

AI-Enabled Product Ideation & Concept Development:

Use AI tools to generate product ideas, refine concepts, and align them with market needs while ensuring originality and contextual relevance.

• AI-Assisted Market & Customer Analysis:

Leverage AI to analyze customer behaviour, segment markets, and derive insights for product positioning, while critically evaluating the accuracy and bias of AI outputs.

• Responsible & Ethical Use of AI in Product Management:

Ensure ethical use of AI by validating outputs, avoiding over-dependence, maintaining transparency, and reflecting on biases and limitations in AI-supported decision-making.

Suggested Tools:

- Miro
- Figma

Applications:

- Business analysis, forecasting, product performance evaluation
- Product presentations, concept visualization
- Product prototyping and interface design

Project Component

- Identify an unmet customer need in a specific market segment and develop a new product concept, including idea generation, screening, concept testing, and business analysis.
- Analyze the product portfolio of a company and recommend strategies for resource allocation, product addition, or elimination to maximize profitability and growth.
- Use AI tools to generate and refine a product idea, and evaluate its feasibility, customer value, and ethical considerations.

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Explain the concepts of product strategy, competition, and product management.	L2
2	Analyze buyer behaviour and its impact on product decisions.	L4
3	Evaluate product life cycle stages and portfolio strategies.	L5

4	Apply new product development processes and strategies.	L3
5	Assess product ideas using business analysis, testing, and evaluation techniques.	L4

Table: Mapping Levels of COs to POs / PSOs

COS	Program Outcomes (POs)											Program Specific Outcomes (PSOs)	
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO1 1	PSO 1	PSO 2
CO1	3	2	2	2	3			2	2	2	2		2
CO2	3	3	2	2	3			2	2	2	2		2
CO3	2	3	3	2	3			2	2	2	2		2
CO4	2	2	3	3	3			2	2	2	2		2
CO5	3	3	2	2	3			2	2	2	2		2

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

TEXT BOOKS:

1. Michael Baker and Susan Hart, Product Strategy and Management, Pearson Education, 2nd Edition, 2008. ISBN – 9788131724286

REFERENCE BOOKS:

1. Philip Kotler and Kevin Lane Keller, Marketing Management, Pearson, 14th Edition, 2013.
2. Donald Lehmann and Russell Winer, Product Management, McGraw-Hill, 4th Edition, 2005.
3. Marc Annacchino, New Product Development: From Initial Idea to Product Management, Elsevier, 2003.

E-RESOURCES:

- https://hbr.org/topic/product-management
- https://www.productplan.com/learn/
- https://www.interaction-design.org/literature/topics/product-design

Activity Based Learning (Suggested Activities in Class)

- Identify real-world product failures and analyze reasons using product strategy concepts.
- Develop a new product idea and carry out concept testing with peer/customer feedback.
- Conduct product life cycle analysis for existing products in different industries.
- Design a product portfolio strategy for a company using case-based scenarios.
- Perform market research to understand buyer behavior and product positioning.

****

MOOC Courses

Sl. No.	Course Code	Course Title
1	*CID	Data Analytics and Visualization with MS PowerBI [IIMB] 6 WEEKS Data Analytics and Visualization with MS Power BI | SWAYAM https://onlinecourses.swayam2.ac.in/e-learning/preview/imb26_mg110 Data Analytics with Python [NPTEL] 12 WEEKS Preview: Data Analytics with Python | SWAYAM https://onlinecourses.nptel.ac.in/e-learning/preview/noc26_cs86
2	*CID	Practical Cyber Security for Cyber Security Practitioners, IIT Kanpur [NPTEL] 12 WEEKS Practical Cyber Security for Cyber Security Practitioners | SWAYAM https://onlinecourses.nptel.ac.in/e-learning/preview/noc25_cs120
3	*CID	Quantum Computation and Quantum Information [NPTEL] 12 WEEKS Quantum Computation and Quantum Information | SWAYAM https://onlinecourses.nptel.ac.in/e-learning/preview/noc26_ph02
4	*CID	Product and Brand Management, IIT Roorkee [NPTEL] 12 WEEKS Product and Brand Management | SWAYAM https://onlinecourses.nptel.ac.in/e-learning/preview/noc22_mg82

Note: CID = NPTEL Course ID

DevOps Fundamentals & Product Launch

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

SEMESTER – V

Subject Code	:		**Credits**	:	02
Hours / Week	:	04 Hours	**Total Hours**	:	30 Hrs (15HrsL +15Hrs J)
L–T–P-J	:	1–0–0-2	**CIE MARKS**	:	100 Marks

Course Learning Objectives:

This Course will enable students to:

1. Understand the DevOps philosophy, culture, and its role in modern software delivery pipelines.
2. Learn version control best practices, CI/CD pipeline design, and build automation strategies.
3. Master containerization with Docker and container orchestration using Kubernetes.
4. Implement Infrastructure as Code, observability, and cloud deployment using industry-standard tools.

Broad Course Description

DevOps breaks down silos between development and operations, enabling faster, more reliable software delivery. This course covers the complete DevOps lifecycle: Git-based version control with GitFlow and trunk-based development, Jenkins and GitHub Actions CI/CD pipelines, automated testing strategies, Docker containerization with multi-stage builds and Docker Compose, Kubernetes orchestration (Pods, Services, Deployments, ConfigMaps, Helm, HPA), Terraform and Ansible Infrastructure as Code, AWS cloud basics (EC2, S3, RDS, Lambda), and Prometheus/Grafana/ELK observability stack. DevSecOps principles including SAST, DAST, container vulnerability scanning with Trivy, and secrets management with HashiCorp Vault are integrated throughout. Students build complete CI/CD pipelines and production deployment systems applicable to real-world engineering roles.

UNIT – I	**03 Hours**

DevOps Foundations & Version Control: DevOps philosophy – CAMS model (Culture, Automation, Measurement, Sharing), Lifecycle phases, Agile and DevOps alignment, DORA metrics. Git fundamentals – init, add, commit, push, pull, branching and merging. GitHub/GitLab – remote repositories, pull requests, code reviews. Git workflows – GitFlow, GitHub Flow, Trunk-Based Development. (Ref: Gene Kim, The DevOps Handbook Ch. 1–4)

Textbook Reference:

• Gene Kim et al., The DevOps Handbook, 2nd Ed – Ch. 1–4 (IT Revolution, 2021)

• Scott Chacon & Ben Straub, Pro Git – Ch. 1–3 (Apress; free at git-scm.com)

UNIT – II	**03 Hours**

CI/CD Pipelines & Automation: CI concepts – build automation, automated testing, code quality gates. Jenkins – installation, declarative pipeline syntax. GitHub Actions – workflows, triggers, action marketplace. Automated testing in CI – unit tests, integration tests, code coverage reports. Artifact management – Docker Registry, JFrog. Blue-Green and Canary deployment strategies. (Ref: Jez Humble, Continuous Delivery Ch. 1–5)

Textbook Reference:

• Jez Humble & David Farley, Continuous Delivery – Ch. 1–5 (Addison-Wesley, 2010)

• GitHub Actions Official Documentation (docs.github.com/en/actions)

UNIT – III	**03 Hours**

Containerization with Docker: Virtualization vs. Containerization. Docker architecture – Docker

Engine, Images, Containers, Registry. Dockerfile – writing, layering, multi-stage builds. Docker CLI – build, run, stop, exec, logs. Docker Compose – multi-container Applications, networking, and volumes. Container networking modes. Docker Hub and private registries. Security scanning and best practices. (Ref: Nigel Poulton, Docker Deep Dive Ch. 1–8)

Textbook Reference:

• Nigel Poulton, Docker Deep Dive – Ch. 1–8 (Self-Published, 2023)

• Docker Official Documentation (docs.docker.com)

UNIT – IV	03 Hours

Container Orchestration with Kubernetes: Kubernetes architecture – Master/Worker nodes, Pods, Services, Deployments, ReplicaSets. ConfigMaps and Secrets, Namespaces, kubectl CLI. Kubernetes networking – ClusterIP, NodePort, LoadBalancer, Ingress. Persistent Volumes and PVCs. Helm package manager. Horizontal Pod Autoscaler (HPA). Managed Kubernetes – EKS, GKE, AKS. (Ref: Brendan Burns, Kubernetes: Up and Running Ch. 1–8)

Textbook Reference:

• Brendan Burns et al., Kubernetes: Up and Running, 3rd Ed – Ch. 1–8 (O'Reilly, 2022)

• Kubernetes Official Documentation (kubernetes.io/docs)

UNIT – V	03 Hours

IaC, Monitoring & DevSecOps: Terraform – providers, resources, state management, modules. Ansible – playbooks, roles, inventory management. AWS basics – EC2, S3, RDS, Lambda. Monitoring and observability – Prometheus, Grafana dashboards, ELK Stack (Elasticsearch, Logstash, Kibana). Alerting and on-call practices. DevSecOps – SAST, DAST, Trivy container scanning, Vault secrets management. SRE principles – SLIs, SLOs, SLAs. (Ref: Kief Morris, Infrastructure as Code Ch. 1–6)

Textbook Reference:

Kief Morris, Infrastructure as Code, 2nd Ed – Ch. 1–6 (O'Reilly, 2020)

Betsy Beyer et al., Site Reliability Engineering – Ch. 1–4 (O'Reilly, 2016)

NVIDIA DLI Free Course Mapping (Validated & Justified)

Validation Source: NVIDIA DLI Official Training Catalog (August 2024 PDF) + Live URL verification (April 2026). All courses below are confirmed FREE self-paced courses accessible at **learn.nvidia.com**. Only FREE courses are listed; paid instructor-led workshops excluded.

Unit	NVIDIA DLI Free Course	Category	Duration	Cert.	Verified URL	Mapping Justification
Unit 3	**An Even Easier Introduction to CUDA**	Accelerated Computing	2 Hrs	—	learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+T-AC-01+V1	CUDA introduces GPU-accelerated parallel computing — directly relevant to understanding Docker GPU containers

						(NVIDIA Container Toolkit) and GPU-enabled containerized AI workloads in Unit 3.
Units 4–5	**AI Infrastructure and Operations Fundamentals**	Infrastructure / AI Ops	8 Hrs	**Yes**	learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+S-IN-01+V1	Essential 8-hr free course on GPU cluster management, DGX administration, NCCL, and AI Enterprise — directly maps to Kubernetes-based GPU orchestration, IaC, and monitoring in Units 4–5. Certificate awarded.
Unit 5	**Accelerate Data Science Workflows With Zero Code Changes**	Data Science / RAPIDS	2 Hrs	—	courses.nvidia.com/course-v1:DLI+T-DS-03+V1	GPU-accelerated data science pipelines via RAPIDS — reinforces Unit 5 cloud deployment and monitoring of data-intensive AI microservices deployed in GPU-enabled Kubernetes clusters.
Unit 5	**Exploring Adversarial Machine Learning**	Deep Learning / Security	2 Hrs	—	learn.nvidia.com/courses/course-detail?course_id=course-	Adversarial ML covers AI-based threat modeling — maps directly to

					v1:DLI+T-FX-06+V1	DevSecOps (SAST, DAST, AI-powered security pipelines) and security hardening topics in Unit 5.

AI Integration & Authentic Intelligence: Reflection (AI² Pedagogy)AI-Assisted

Pipeline Generation: Use GitHub Copilot to generate CI/CD YAML configurations; evaluate correctness, security, and edge-case handling.

- **Intelligent Monitoring Analysis:** Explore AI-based anomaly detection in Prometheus/Grafana; understand the difference between rule-based and ML-based alerting.
- **Reflective Automation Thinking:** Identify pipeline stages suitable for full automation versus those requiring human approval gates and judgment.
- **AI in DevSecOps:** Evaluate AI-powered security scanning tools (SAST, vulnerability detection) for reliability, false positive rates, and actionability.
- **Infrastructure Cost Optimization with AI:** Evaluate AI recommendations for cloud resource sizing, reserved instance planning, and cost-aware engineering.

Applications:

- Continuous Software Delivery in Enterprise Product Companies
- Cloud-Native Application Deployment and Scaling
- Microservices Infrastructure Management
- Automated Testing and Quality Assurance Pipelines
- Site Reliability Engineering (SRE) at Scale

Laboratory Component (1 Credit)

The lab emphasizes hands-on implementation, visualization, and AI-assisted debugging.

Lab Experiments

- Set up a Git repository; practice branching strategies, pull requests, and structured code reviews.
- Build a Jenkins pipeline to automate build, test, and artifact archiving for a Node.js application.
- Create a GitHub Actions CI/CD workflow that runs tests and deploys to a staging environment.
- Write a multi-stage Dockerfile for Node.js/Python; build, tag, and push to Docker Hub.

- Deploy a multi-container application using Docker Compose (frontend + backend + database).
- Set up local Kubernetes cluster with Minikube; deploy, scale, and expose a containerized application.
- Implement Kubernetes rolling updates, rollbacks, and Horizontal Pod Autoscaler.
- Write Terraform config to provision an AWS EC2 instance with security groups and S3 bucket.
- Set up Prometheus and Grafana to monitor a containerized application with custom alerting rules.
- Implement a DevSecOps pipeline with SonarQube (code quality), Trivy (container scanning), and Vault (secrets).

Mini Project Component

- **Industry-Driven Projects**

Some sample projects described below simulate the real software engineering problems.

Project 1 – Full CI/CD Pipeline for Microservices App

Problem: Complete CI/CD pipeline for a 3-tier microservices application with automated testing and Kubernetes deployment.

Concepts used: GitHub Actions, Docker, Kubernetes, Helm

Industry relevance: Standard DevOps workflow in product-based companies.

Project 2 – Infrastructure Automation with IaC

Problem: Provision cloud infrastructure (VPC, EC2, RDS, Load Balancer) using Terraform and configure with Ansible.

Concepts used: Terraform, Ansible, AWS, IaC principles

Industry relevance: Cloud and platform engineering teams.

Project 3 – Observability Platform Setup

Problem: Build a complete monitoring and logging platform for a distributed application.

Concepts used: Prometheus, Grafana, ELK Stack, Alerting

Industry relevance: SRE and platform engineering roles.

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Understand DevOps philosophy and apply Git version control best practices in team-based development workflows.	L2
2	Design and implement CI/CD pipelines using Jenkins and GitHub Actions for automated build, test, and deployment.	L3

3	Containerize Applications using Docker and manage multi-container deployments with Docker Compose.	L3

4	Deploy, scale, and manage containerized Applications using Kubernetes with networking and storage configurations.	L4
5	Implement IaC using Terraform/Ansible, establish observability, and apply DevSecOps security practices.	L5

Table: Mapping Levels of COs to POs / PSOs

	Program Outcomes (POs)											Program Specific Outcomes (PSOs)	
COs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO 1**	**PSO 2**
CO1	3	2	2	3	2						2	3	2
CO2	3	3	3	3	3						2	3	2
CO3	3	3	2	3	3						2	3	2
CO4	3	3	3	3	3						3	3	3
CO5	3	3	3	3	3						3	3	3
3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)													

Suggested Tools:

- Git, GitHub / GitLab
- Jenkins, GitHub Actions
- Docker, Docker Compose
- Kubernetes, Helm
- Terraform, Ansible
- Prometheus, Grafana, ELK Stack

TEXT BOOKS:

1. Gene Kim et al., The DevOps Handbook, 2nd Ed, IT Revolution Press, 2021.
2. Jez Humble & David Farley, Continuous Delivery, Addison-Wesley, 2010.
3. Brendan Burns et al., Kubernetes: Up and Running, 3rd Ed, O'Reilly, 2022.REFERENCE BOOKS:
4. Kief Morris, Infrastructure as Code, 2nd Ed, O'Reilly, 2020.
5. Betsy Beyer et al., Site Reliability Engineering, O'Reilly, 2016.

E-RESOURCES:

- https://kodekloud.com
- https://training.linuxfoundation.org
- https://www.coursera.org/learn/devops-on-aws

- https://learn.nvidia.com (NVIDIA DLI Free Courses)

Activity Based Learning (Suggested Activities in Class)

- DevOps simulation: build-break-fix pipeline exercises in Minikube sandbox environments.
- Weekly retrospectives simulating Agile stand-up meetings and sprint reviews.
- Guest lectures from DevOps engineers and Site Reliability Engineers at product companies.
- Incident response role-play: teams diagnose and resolve a simulated production outage.

****

AWS WEB SERVICES

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

SEMESTER – V

Subject Code	:	24CT3511	**Credits**	:	02
Hours / Week	:	03 Hours	**Total Hours**	:	26 Hours
L–T–P–S	:	1–0–0–2			

Prerequisites: Fundamentals of Cloud Computing and Cloud Security Services

Course Learning Objectives:

This Course will enable students to:

1. To **understand** fundamental concepts and hands-on knowledge of Cloud Computing using AWS Platform.
2. To **implement** the state-of-the-art AWS Networking, Database, Storage Services in real time projects
3. **Apply** Security and privacy in AWS Cloud across various domains like Storage, Database and applications.

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*** means it includes not only traditional lecture method, but different *type 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, notetaking, 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 analyse information rather than simply recall it.
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.

UNIT – I	**5 Hours**
Introduction What is Cloud Computing? How Does Cloud Computing Work? What is AWS? AWS Free Tier, Compute Services: Amazon EC2, Elastic Load Balancing, Auto Scaling ***(Text Book: Chapter 1, Chapter 2)***	
UNIT – II	**5 Hours**
Networking Services Amazon VPC, Amazon Route 53 ***(Text Book: Chapter 4, Chapter 8)***	
UNIT – III	**5 Hours**
AWS Security Shared Responsibility Model, AWS IAM and KMS ***(Text Book: Chapter 1 , Chapter 6)***	
UNIT-IV	**5 Hours**

AWS Storage Services Amazon S3, Amazon EBS, Amazon EFS, Amazon Glacier, AWS Storage Gateway, Amazon Cloud Front ***(Text Book: Chapter 2 , Chapter 3)***

UNIT-V	**6 Hours**
AWS Database Services, Application Services Amazon RDS, Amazon DynamoDB, Amazon ElastiCache, Amazon Simple Email Service (Amazon SES), Amazon Simple Notification Service (Amazon SNS), Amazon Simple Queue Service (Amazon SQS), Amazon Simple Workflow Service (Amazon SWF) ***(Text Book: Chapter 5 , Chapter 9)***	

Course Outcome	**Description**	**Bloom's Taxonomy Level**
At the end of the course the student will be able to:		
1	**Examine** and practice various AWS services to Configure and deploy, a virtual machine using Amazon EC2.	**L4**
2	**Analyze** different VPC architectures and their use cases in real-world scenarios by Examining the network traffic flow within a VPC and between VPCs.	**L4**
3	**Evaluate** the security and compliance of an AWS environment using AWS KMS for encryption and Assess the impact of key management practices on data security and operational efficiency.	**L5**
4	**Implement** versioning and lifecycle policies to manage data in S3.	**L3**
5	**Analyze** the performance characteristics of different EBS volume types to select the appropriate one for a given workload.	**L6**

Table: Mapping Levels of COs to POs / PSOs

COs	**Program Outcomes (POs)**												**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
CO1	2	2	3	2	3	2	2		2	2		2	2	
CO2	2	2	3	2	3	2	2		2	2		2	2	
CO3	2	2	3	2	3	2	2		2	2		2	2	
CO4	2	2	3	2	3	2	2		2	2		2	2	
CO5	2	2	3	2	3	2	2		2	2		2	2	

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

TEXT BOOKS:

1. Ben Piper, David Clinton, 'AWS certified solutions architect study guide: associate SAA-C02 Exam (Aws Certified Solutions Architect Official: 22 February 2021

REFERENCE BOOKS:

1. John Paul Mueller, "AWS for Developers For Dummies" , Wiley publications.

2. David Clinton and Ben Piper, "AWS Certified Cloud Practitioner Study Guide: CLF-C01 Exam", Wiley publications.

E-RESOURCES :

1. https://aws.amazon.com/education/awseducate/
2. https://www.simplilearn.com/amazon-web-services-getting-started-rar404-article
3. https://www.mygreatlearning.com/academy/learn-for-free/courses/aws-for-beginners1
4. https://www.edureka.co/free-course/learn-aws-basics-online

Activity Based Learning (Suggested Activities in Class)

1. Building a Server less Application
2. Setting Up a CI/CD Pipeline
3. Hands-On Labs
4. Real time application Projects.

****

SUPPLY CHAIN MANAGEMENT & OPERATIONS

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

SEMESTER – VI

Subject Code	: 24CT3601	**Credits**	: 02
Hours / Week	: 02 Hours	**Total Hours**	: 26 Hours
L–T–P–S	: 2–0–0–0		

Course Learning Objectives:

This Course will enable students to:

1. To introduce various concepts related to the supply chain management.
2. To educate the importance of supply chain decision such as design, planning, and operations of a firm.
3. To give the students an understanding of the analytical tools necessary to solve supply chain problems.
4. To showcase the ways by which the companies use different supply chain drivers to improve their performance.

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 means it includes not only traditional lecture method, but different *type 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, notetaking, 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 analyse information rather than simply recall it.
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.

UNIT – I	**05 Hours**

INTRODUCTION TO SUPPLY CHAIN MANAGEMENT: Supply chain historical perspective - objectives importance decision phases process view impellers of supply chain financial measures of performance drivers of supply chain performance framework for structuring drivers facilities inventory transportation information sourcing pricing.

(Text Book-1: Chapter 1: 1.1 to 1.6)

UNIT – II	**05 Hours**

DESIGNING DISTRIBUTION NETWORKS AND NETWORK DESIGN:

Role of distribution factors influencing distribution network design, design options for a distribution network, online sales and the distribution network, distribution channels in Indian agricultural, FMCG and commodity sectors, role of network design, factors influencing network design decisions, framework for network design decisions.

(Text Book-1: Chapter 4: 4.1 to 4.7, Chapter 5: 5.1 to 5.3)

UNIT – III	05 Hours

DEMAND FORECASTING AND MANAGING UNCERTAINTY:
DEMAND FORECASTING: role, characteristics, components, approach, time series, methods, measures of forecast error, role of it. **MANAGING UNCERTAINTY:** safety inventory and its appropriate level, impact of supply uncertainty, aggregation and replenishment policies. (Text Book-1: Chapter 7: 7.1 to 7.6, 7.9, ***Chapter 12: 12.1,12.4, 12.5,12.7)***

UNIT – IV	05 Hours

COORDINATION IN A SUPPLY CHAIN:
Lack of supply chain coordination and the Bullwhip effect, effect on performance, obstacles to coordination, managerial levers, continuous replenishment and vendor- managed inventories, collaborative planning, forecasting and replenishment, Indian experience.
(Text Book-1: Chapter 10: 10.1 to 10.7)

UNIT – V	06 Hours

TRANSPORTATION AND SOURCING:
Role of transportation, modes and their performance, design options and their trade, offs Tailored transportation. Sourcing In-house or Outsource 3rd and 4th PLs, Scoring and assessing suppliers. Case studies in SCM
(Text Book-1: Chapter 14: 14.1 to 14.4, 14.6, 14.7, Chapter 15: 15.2)

Course Outcomes:

At the end of the course the student will be able to:

1. **Understand** the principles and practices of inventory management within a supply chain.
2. Identify and **analyze** factors that influence decisions in network design**.**
3. **Apply** various time series methods for demand forecasting, such as moving averages, exponential smoothing, and trend analysis.
4. **Examine** case studies and examples from the Indian context to understand challenges, successes, and unique approaches to mitigating the Bullwhip Effect and enhancing supply chain coordination**.**
5. **Develop** frameworks and methods for scoring and assessing suppliers based on criteria such as quality, reliability, cost, responsiveness, and sustainability.

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
CO1	3	3	3									2		3
CO2	3	3	3									2		3
CO3	3	3	3									2		2
CO4	3	3	3									2		3
CO5	3	3	3									2		3

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

TEXT BOOKS:

1. Sunil Chopra, Peter Meindl and Kalra D V, Supply Chain Management Strategy, Planning and Operation, Pearson, 5th Edition, 2013.
2. Janat Shah, Supply Chain Management : Text and Cases, Pearson, 1st Edition, 2009.

REFERENCE BOOKS:

1. Donald J Bowersox, David J Closs, Bixby Cooper M, Supply Chain Logistics Management, Tata McGraw Hill, 2nd Edition, 2008.
2. Sople V V, Supply Chain Management Text and Cases, Pearson Education, 2012.
3. Jeremy F Shapiro, Modeling the Supply Chain, Cengage Learning, 2nd Edition, 2007.
4. Coyle, Bardi, Longley, The management of Business Logistics A supply Chain Perspective, Thomson Press, 7th Edition, 2006.

Activity Based Learning (Suggested Activities in Class)

1. Demonstration of solution to a problem through real time examples.

****

Computer Networks & Distributed Systems

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

SEMESTER – VI

Subject Code	:		**Credits**	:	04
Hours / Week	:	03 Hours	**Total Hours**	:	30 Hrs L+15 Hrs P
L–T–P-J	:	3–0–2-0	**CIE Marks**	:	60 + 40 Marks

Prerequisites: Programming in any Language, Networking Basics

Course Learning Objectives:

This Course will enable students to:

1. Understand and compare OSI and TCP/IP models, analyzing their relevance to modern AI API latency and streaming.
2. Apply IP addressing, subnetting, and routing protocols (OSPF, BGP) to design scalable, low-latency cloud AI topologies.
3. Evaluate transport and application protocols (TCP, UDP, HTTP/2, WebSockets) for reliability, efficiency, and secure AI communication.
4. Implement streaming LLM response endpoints using Server-Sent Events (SSE) in Python.
5. Analyze DNS, CDN, and load balancing strategies to optimize global AI API distribution and cache performance.
6. Apply CAP theorem and consensus algorithms (Raft) to distributed ML systems, balancing availability, consistency, and reliability.
7. Benchmark gRPC vs REST communication models for high-performance AI inference APIs under heavy load.
8. Examine GPU communication frameworks (NVIDIA NCCL, NVLink) to understand high-throughput distributed training.

Broad Course Description

This course explores how classical network models and protocols underpin modern AI infrastructure, focusing on latency, scalability, and distributed communication. Students will study OSI and TCP/IP layers, IP addressing, routing protocols, and transport/application mechanisms like TCP, UDP, HTTP/2, and WebSockets, applying them to real-world AI API design and streaming LLM responses. The curriculum extends to DNS, CDN, and load balancing for global distribution, while also examining distributed systems trade-offs through the CAP theorem and consensus algorithms. Finally, learners will compare gRPC and REST for high-performance AI services and analyze GPU communication frameworks such as NVIDIA NCCL and NVLink, connecting textbook theory to practical cloud AI deployment and optimization.

.

UNIT – I	05 Hours

Network Models & Latency: Network Fundamentals & Protocols
OSI and TCP/IP models — connect to AI API network latency analysis; why HTTP/2 multiplexing matters for LLM streaming
Textbook 1: Ch. 1 (Introduction), Ch. 2 (Physical Layer), Ch. 6 (Transport Layer) Textbook 2: Ch. 2 (Data Transmission), Ch. 3 (Data Communication Protocols)

UNIT – II	05 Hours

Addressing & Routing: IP addressing, routing (OSPF, BGP), subnetting — design the network topology for a cloud AI startup (multi-region, low-latency) Textbook 1: Ch. 5 (Network Layer: Routing)

Textbook 2: Ch. 5 (Network Layer: Addressing & Routing)	
UNIT – III	**05 Hours**
Transport & Application Protocols: TCP, UDP, HTTP, HTTPS, WebSockets — implement a streaming LLM response endpoint with SSE (Server-Sent Events) in Python Textbook 1: Ch. 6 (Transport Layer), Ch. 7 (Application Layer) Textbook 2: Ch. 6 (Transport Layer Protocols), Ch. 7 (Application layer)	
UNIT – IV	**07 Hours**
Application Services & Distribution: DNS, CDN, load balancing — design a globally distributed AI API with Cloudflare CDN; analyse cache hit rates Textbook 1: Ch. 7 (Application Layer: DNS, Web, CDN) Textbook 2: Ch. 7 (Application Layer Services)	
UNIT – V	**08 Hours**
Distributed Trade-offs Distributed Systems for AI CAP theorem — understand trade-offs in distributed vector databases (availability vs consistency) NVIDIA NCCL: distributed GPU communication — understand how DGX B200 NVLink 4 enables 900 GB/s all-reduce for distributed training API Communication Models and High Performance GPU Communication gRPC vs REST for AI services — benchmark gRPC vs REST for LLM inference API at 1000 req/s Textbook 1: Ch. 7 Distributed consensus: Raft algorithm — how distributed ML parameter servers stay consistent Textbook 2: Ch. 8 (Network Security & Reliability touches distributed coordination) Textbook 1: Ch. 7 (Application Layer: RPC concepts) Textbook 2: Ch. 7 (Application Layer Services)	

AI Integration & Authentic Intelligence: Reflection (AI² Pedagogy)

- Globally distributed AI API service
- Build: (1) LLM inference endpoint with async streaming (FastAPI + SSE), (2) load balancer (nginx) distributing to 2 model replicas, (3) Redis caching for repeated queries, (4) Prometheus + Grafana monitoring dashboard. Benchmark throughput, latency, cache efficiency. Frame as "infrastructure design for a funded AI startup".

Applications:

- **Latency Models**: Benchmark AI API performance and optimize streaming with modern protocols like HTTP/2.
- **Addressing & Routing**: Guide the design of multi-region cloud topologies to achieve low-latency inference.
- **Transport & Application Protocols**: Enable real-time LLM endpoints using Server-Sent Events (SSE) or WebSockets.
- **DNS, CDN & Load Balancing**: Support globally distributed AI APIs with efficient caching

and traffic management.

- **Distributed Systems Trade-offs**: Concepts like the CAP theorem and Raft consensus underpin vector databases and parameter server consistency.
- **API Communication Models**: Compare gRPC versus REST for efficient communication.
- **GPU Communication Frameworks**: Tools like NVIDIA NCCL drive high-performance inference and training pipelines.

Lab Component

1. **Latency & Congestion Simulation** Implement a three-node point-to-point network with duplex links. Vary bandwidth and queue size to measure packet drops and analyze latency trade-offs.
2. **Routing & Path Optimization** Write a program to compute shortest paths between nodes using the Bellman-Ford algorithm. Extend the experiment to simulate routing in a multi-region AI service topology.
3. **Transport Protocols & Streaming** Using TCP/UDP sockets, build a client-server program where the client requests a file and the server streams its contents. Compare performance with datagram sockets for real-time message delivery.
4. **Wireless & Mobile Systems** Simulate a wireless LAN (ESS) with transmitting nodes and evaluate packet transmission performance. Extend to study GSM/CDMA behavior using NS2/NS3 or equivalent environments.
5. **Reliability & Security in Communication** Implement error detection using CRC codes and a simple RSA algorithm for encryption/decryption. Demonstrate how these ensure integrity and confidentiality in distributed AI APIs.
6. **Congestion Control & Distribution** Implement congestion control using the leaky bucket algorithm. Analyze throughput and packet drops, connecting results to CDN/load balancing strategies in globally distributed systems.

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	**Analyze** network fundamentals and protocol behavior using OSI/TCP-IP models, and evaluate latency impacts in AI API communication, including the role of HTTP/2 multiplexing in LLM streaming.	L2
2	**Design** and configure IP addressing and routing schemes (OSPF, BGP, subnetting) to build multi-region, low-latency network topologies suitable for cloud AI startups.	L3
3	**Implement** and compare transport/application protocols (TCP, UDP, HTTP, HTTPS, WebSockets) by building streaming endpoints (e.g., SSE in Python) and assessing their performance in real-time AI services.	L3
4	**Apply** DNS, CDN, and load balancing mechanisms to design globally distributed AI APIs, and analyze cache hit rates for optimizing service delivery.	L4

5	**Evaluate** distributed system trade-offs using the CAP theorem and consensus algorithms (Raft), and assess consistency vs availability in distributed vector databases and ML parameter servers. Benchmark API communication models (gRPC vs REST) under high request loads, and analyze high-performance GPU communication frameworks (e.g., NVIDIA NCCL, NVLink) for scalable distributed training.	L5

COS	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	2	3							2	3	3
CO2	3	3	3	3							2	3	3
CO3	3	3	2	3							2	3	3
CO4	3	3	3	3							3	3	3
CO5	3	3	3	3							3	3	3

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

TEXTBOOKS

1. Computer Networks - Andrew S Tanenbaum, 5th Edition, Pearson Education.
2. Behrouz A. Forouzan, Data Communications and Networking, Fifth Edition TMH, 2013.

Activity Based Learning (Suggested Activities in Class)

- Globally distributed AI API service
- Build: (1) LLM inference endpoint with async streaming (FastAPI + SSE), (2) load balancer (nginx) distributing to 2 model replicas, (3) Redis caching for repeated queries, (4) Prometheus + Grafana monitoring dashboard. Benchmark throughput, latency, cache efficiency. Frame as "infrastructure design for a funded AI startup".

****

Deep Learning

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

SEMESTER – IV

Subject Code	:		**Credits**	:	04
Hours / Week	:	03 Hours	**Total Hours**	:	60 Hrs (45Hrs L+15 Hrs P)
L–T–P-J	:	3–0–2-0			

Course Vision: This course develops the ability to design, train, and deploy deep learning models using modern AI frameworks, integrating AI Factory workflows, NVIDIA DLI practices, and ethical AI principles for real-world intelligent systems.

Prerequisites: Linear Algebra, Calculus, Probability & Statistics, Python Programming, Machine Learning, Data handling and Preprocessing

Course Learning Objectives:

This Course will enable students to:

1. Describe the fundamental concepts of artificial neural networks, including perceptrons, multilayer perceptrons, activation functions, and evaluation techniques used in deep learning.
2. Illustrate the learning mechanisms of neural networks, including forward and backward propagation, gradient descent variants, and optimization techniques used in training models.
3. Apply regularization techniques, weight initialization methods, and hyperparameter tuning strategies for building well-generalized deep learning models.
4. Analyze major deep learning architectures such as CNNs, RNNs, LSTMs, GRUs, and attention-based models for vision and sequential data applications.
5. Examine advanced deep learning approaches including autoencoders, graph neural networks (GNNs), and explainable AI techniques with awareness of ethical and responsible AI principles.

Broad Course Description

Deep Learning is a core discipline of artificial intelligence that enables machines to learn hierarchical representations from data. This course covers neural network architectures, optimization techniques, and advanced models such as CNNs, RNNs, with emphasis on model design, training, and evaluation for real-world applications.

Students gain hands-on experience in GPU-accelerated training and high-performance computing workflows inspired by the **NVIDIA Deep Learning Institute.** The curriculum is aligned with industry-recognized certifications, enabling learners to pursue **NVIDIA-Certified Associate credentials in Generative AI (LLMs/Multimodal).** By the end of the course, students will be able to build, optimize, and deploy deep learning models while considering scalability, performance efficiency, ethical implications, and societal impact.

UNIT – I	**08 Hours**

Introduction to Neural Networks and Deep Learning:

Introduction to Artificial Intelligence, Machine Learning, and Deep Learning: Differences and Relationships.

Foundations of Neural Networks and Deep Learning: Neural Networks, Activation functions, Loss Functions, Hyperparameter, The Perceptron, Multilayer perceptron (MLP).

Engineering Applications: Image classification and object detection, speech recognition, predictive maintenance and fault diagnosis in industrial systems, medical diagnosis and healthcare analytics, autonomous systems and robotics.

UNIT – II	10 Hours
Mathematical Background, Data Preparation and Training Techniques : Data representation using tensors: Scalars, vectors, matrices, tensor shapes, data batches, image data, and introduction to sequence/time-series data representation. The gears of neural networks: Tensor operations—Element-wise operations, Broadcasting, Tensor product, and Tensor reshaping. Training the Model:Forward and Backward Propagation:forward propagation; backpropagation using the chain rule;Training Loop- epochs and batches, forward pass, loss computation, backpropagation, parameter update and validation steps;Gradient descent methods: batch, stochastic, and mini-batch gradient descent;Optimization algorithms: Momentum and Adam. Model Optimization and Generalization: Weight initialization: overview of Xavier/Glorot and He initialization; Regularization techniques: L2 weight decay, dropout, early stopping based on validation performance, and basic concepts of batch normalization; Data augmentation; Hyperparameter tuning using validation data. **Engineering Applications:** Time-series forecasting and demand prediction, anomaly detection in cybersecurity and network monitoring, energy load forecasting in smart grids, traffic flow prediction and intelligent transportation systems, signal processing and sensor data analysis.	
UNIT – III	**09 Hours**
Convolutional Neural Networks (CNNs) Biological inspiration of CNNs,Convolution operation: kernels/filters, feature maps, stride, and padding (with simple numerical examples). Convolutional layers: parameter sharing, local connectivity, and multiple filters/channels. Pooling layers: max pooling and average pooling; role of batch normalization and dropout in CNNs. CNN architecture: overall structure and working of convolutional neural networks, including input, convolutional, pooling, and fully connected layers. Popular CNN architectures (conceptual overview): LeNet-5 (architecture and idea of early CNN); AlexNet (representation learning—brief—and architecture); VGG (VGG blocks and network design); ResNet (residual blocks and model architecture). Transfer learning: use of pre-trained models (VGG16, ResNet50) for feature extraction and fine-tuning. Object detection (introduction): bounding boxes and YOLO (concept and intuition). **Engineering Applications:** Autonomous driving and ADAS, remote sensing and satellite imagery, biometric authentication (fingerprint/iris recognition), agricultural monitoring using drones, and CNN-based access control systems (e.g., NVIDIA DLI case study: “Personalized Doggy Door”).	
UNIT – IV	**09 Hours**
Recurrent Neural Networks, LSTMs and Attention Mechanism : Need for Sequential Models: limitations of MLP and CNN for sequential data (text, time-series, audio). Recurrent Neural Networks (RNNs): architecture, hidden state, and unrolling through time. Vanishing Gradient Problem: challenges in learning long-term dependencies in RNNs. Long Short-Term Memory (LSTM): cell state, input, forget, and output gates. Gated Recurrent Unit (GRU): simplified gating mechanism and comparison with LSTM. Attention Mechanism: limitations of fixed context vectors in Seq2Seq models; queries, keys, and values; Bahdanau (additive) attention-conceptual understanding. **Engineering Applications:** Machine translation systems, predictive maintenance using sensor sequence data, anomaly detection in financial transactions, human activity recognition using wearable sensors, network traffic analysis and intrusion detection.	
UNIT – V	**09 Hours**
Unsupervised Deep Learning and Applications :	

Unsupervised Pretrained Networks (UPNs): motivation and learning from unlabelled data. Autoencoders: encoder–decoder architecture, bottleneck/latent space, applications in compression and

denoising, Variational Autoencoders, Introduction to Generative Adversarial Networks (GANs) Graph Neural Networks (GNNs): basic concepts of graph-structured data, learning on graphs, and applications.
Explainable AI (XAI): importance of model interpretability; Grad-CAM, LIME, and SHAP concepts. Responsible AI and Ethics: bias in datasets, fairness, data privacy, and societal impact of AI.

Engineering Applications: Graph-based analysis for smart city infrastructure and networked systems, data compression and denoising in communication and signal processing, explainable AI for reliable decision-making in critical systems, and development of fair, unbiased, and privacy-preserving AI solutions.

NVIDIA DLI Course Mapping

1. Getting Started with Deep Learning
2. Fundamentals of Deep Learning(workshop)
3. Introduction to Graph Neural Networks.

Nvidia Learning Path

1. Deep Learning
2. Gen AI LLM/Generative AI Multimodal)

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

- **AI Integration:**
  - **Module 1:** AI-assisted model development and visualization tools.
  - **Module 2:** AI-based hyperparameter tuning, AutoML tools, and AI-assisted performance analysis.
  - **Module 3:** AI-based vision tools, use of pre-trained models, and transfer learning with foundation models.
  - **Module 4:** NLP frameworks (Transformers), introductory LLM APIs, and AI-assisted text generation tools.
  - **Module 5:** AI development and deployment tools, AI factory platforms, and MLOps automation tools.

- **Authentic Intelligence Reflection:**
  - **Module 1:** limitations of neural networks and interpretability challenges.
  - **Module 2:** bias in training data, overfitting risks, and reliability and fairness concerns.

- **Module 3:** ethical issues in facial recognition, dataset bias in vision systems, and risks in safety-critical automated decisions.
- **Module 4:** misuse of generative AI, hallucination and misinformation risks, and ethical concerns in NLP systems.
- **Module 5:** responsible AI deployment, governance and accountability, and safety, transparency, and trustworthy AI systems.

Laboratory Component

Experiments will be conducted using **PyTorch** or **TensorFlow** on **Jupyter Notebook** or **Google Colab**, following workflows inspired by the **NVIDIA Deep Learning Institute (DLI)**. Students will implement models, evaluate performance using metrics such as accuracy and loss, and analyze training behavior. NVIDIA ecosystem tools such as transfer learning toolkits for vision tasks (**e.g., NVIDIA TAO Toolkit**) and language/speech model development frameworks (**e.g., NVIDIA NeMo**) may be used.

Hardware Requirement: Internet-enabled system with access to a GPU-accelerated computing environment (NVIDIA GPU cloud or equivalent).

List of Experiments

1. Create tensors and perform basic operations such as reshaping, indexing, and element-wise computations using a deep learning framework. Demonstrate simple forward computation without training.
2. Implement linear and logistic regression models using gradient descent. Build a basic training loop with forward pass, loss computation, and parameter updates. Evaluate model performance using appropriate metrics.
3. **Training a Basic Neural Network**

   Train a neural network for image classification using the MNIST dataset and evaluate performance using loss and accuracy. Implement the model using NumPy/PyTorch and analyze training behavior using loss curves.
4. **Convolutional Neural Network (CNN)**

   Design and train a Convolutional Neural Network (CNN) for image classification using MNIST or American Sign Language dataset. Compare performance in terms of accuracy and feature learning.
5. **Data Augmentation**

   Apply data augmentation techniques such as rotation, flipping, and scaling to enhance the dataset and evaluate their impact on model generalization and performance.
6. **Optimization and Regularization**

   Train a neural network using different optimizers (SGD, Adam) and apply regularization techniques such as dropout and batch normalization. Compare overfitting using training and validation curves.
7. **Using Pre-trained Models**

Implement an image classification system using a pre-trained model (e.g., ResNet). Perform inference on new images and understand how classification layers work in deep learning models.

8. **Transfer Learning**

   Fine-tune a pre-trained model on a custom dataset and compare performance with training from scratch. Analyze convergence speed and generalization.

9. **Sequence Modeling using RNN/LSTM**

   Train an RNN or LSTM model for sequence prediction (text or time-series). Evaluate performance and understand handling of sequential data.

10. **Natural Language Processing using Pre-trained Models**

    Implement an NLP task such as text classification using a pre-trained language model. Perform tokenization and text segmentation during preprocessing and understand text masking in masked language modeling. Evaluate model performance using accuracy and loss. *(e.g., sequence modeling / attention-based models, machine translation systems, BERT-based NLP models)*

11. **GNN**

    Implement a Graph Neural Network (GNN) for node classification on a graph dataset. Represent data using nodes and edges, train the model efficiently, and evaluate performance. Analyze the impact of graph structure and discuss real-world applications.

Project Component (Indicative, not limited to)

1. Develop an image classification model that interprets color images using deep learning techniques (CNN-based approach).
2. Design and implement a data generator with augmentation techniques to effectively train models on small datasets and improve generalization.
3. Build an efficient image classification system using transfer learning and feature extraction with pre-trained models (e.g., ResNet), and analyze improvements in training speed and accuracy.
4. Implement a deep learning solution that combines data augmentation and transfer learning to optimize model performance on limited data scenarios.
5. Explore and compare advanced neural network architectures (e.g., CNN variants, transformers) for image-based tasks and analyze their performance.
6. Design an end-to-end deep learning pipeline for image-based applications, including data preparation, model training, and performance evaluation.
7. Investigate recent advancements in deep learning architectures and apply one technique to improve model performance on a selected task.

Research Component:

Students will select three recent research papers in the deep learning domain from reputed sources such as IEEE, Springer, or arXiv, and conduct a literature survey by analyzing the problem, methodology, and key findings. A brief report and presentation summarizing insights and comparisons will be submitted.

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
CO1	**Outline** fundamental concepts of deep learning and neural networks.	L2
CO2	**Apply** forward and backward propagation, tensor operations, and optimization techniques for training deep learning models.	L3
CO3	**Construct and develop** deep learning models using regularization, weight initialization, and hyperparameter tuning to improve generalization.	L3
CO4	**Analyze** and compare architectures such as CNNs, RNNs, LSTM, GRU, and attention-based models for solving vision and sequential data problems.	L4
CO5	**Examine** deep learning solutions using autoencoders, GNNs, and explainable AI techniques while considering ethical and responsible AI principles.	L4

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)											Program Specific Outcomes (PSOs)	
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO 1	PSO 2
CO1	3	2	1	1	2	–	–	1	2	1	2	3	2
CO2	3	3	2	1	3	-	–	1	2	2	3	3	2
CO3	2	3	3	2	3	-	1	2	2	2	3	3	2
CO4	2	3	2	2	3	2	-	2	2	2	3	3	2
CO5	1	2	2	2	2	3	3	2	2	2	3	2	2

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

Assessment Scheme: Integrated Professional Core Courses (Theory + Practical)

Criteria	Weightage of Marks
Continuous Assessment Tests (CAT) – includes mini-project/research component	10
industry-oriented certification (NVIDIA DLI or equivalent)	10
Mid Semester Examination (MSE)	20
Lab Continuous Evaluation and Lab Record	10
Lab exam	10
Total	**60 marks**

Suggested Reading

Primary Textbooks

1. Aston Zhang, Zack C. Lipton, Mu Li, Alex J. Smola, "Dive into Deep Learning", Cambridge University Press,ISBN-10 **:** 1009389432,2023**.**
2. Josh Patterson and Adam Gibson, "Deep Learning: A Practitioner's Approach", O'Reilly Media, ISBN:978-1-4919-1425-0, 2017.
3. Ian Goodfellow, Yoshua Bengio, Aaron Courville, "Deep Learning (Adaptive Computation and Machine Learning series", MIT Press, 2017.

Additional Resources

1. Aurelien Geron, "Hands-On Machine Learning with Scikit-Learn, Keras and TensorFlow", O'Reilly, 3rd Edition, 2022.
2. Francois Chollet, "Deep Learning with Python", Manning Publications, 2nd Edition, 2021.
3. Simon Haykin, "Neural Networks: A Comprehensive Foundation", 2nd Edition, Prentice Hall, 2003.
4. Nikhil Buduma, "Fundamentals of Deep Learning", O'Reilly Media, 2017.

E-Resources

1. NPTEL Deep Learning: https://onlinecourses.nptel.ac.in/noc20_cs62/preview
2. Dive into Deep Learning (interactive): https://d2l.ai

NVIDIA Deep Learning Institute: https://learn.nvidia.com/

****

SMALL-E BUSINESS LAUNCH

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

SEMESTER – VI

Subject Code	:	24OEXXXX	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P–S	:	3–0–0–0			

Course Learning Objectives:

This Course will enable students to:

1. **Experience** with a set of tools and methods for setting up an online service.
2. **Knowledge** of the role of multiple functions in creating a small business.
3. **Ability** to coordinate multiple, interdisciplinary tasks to achieve a common objective.
4. **Emphasis** of specific knowledge from other courses through practice and reflection in an action-oriented setting.

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*** means it includes not only traditional lecture method, but different *type 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, notetaking, 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 analyse information rather than simply recall it.
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.

UNIT – I	**10 Hours**

INTRODUCTION:

Introduction, Identifying Customer Needs, Opportunity Identification & Planning, Idea Generation and Validation, Entrepreneurial Culture, Digital Business Opportunity

(Text Book-1: Chapter 1: 1.1, 1.4 to 1.9, 1.11 to 1.17, 1.22 to 1.28)

UNIT – II	**10 Hours**

Business Plan & its Components, Business Model and Strategies, Selection of Appropriate Business Model, Business Model Canvas, Feasibility

(Text Book-1: Chapter 3: 3.1 to 3.18)

UNIT – III	**09 Hours**

Lean thinking, Specifications, Wireframe Generation and Validation.

(Text Book-1: Chapter 5: 5.1 to 5.4, Chapter 5: 5.11 to 5.19)

UNIT – IV	**10 Hours**

Service-Market Fit, legal Aspects & Entrepreneurial Law, Logistics of goods delivery, Virtual Trade Shows ***(Text Book-1: Chapter 7: 7.2 to 7.17***)

Course Outcomes:

At the end of the course the student will be able to:

1. Identify customer needs and pain points.
2. Generate concept and validate idea.
3. Use design thinking to create a wire frame of your online service.
4. Launch your app/website.
5. Register your small business.

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
CO1	2	3										2		3
CO2	3	2	2									1		2
CO3	2	2	2									1		2
CO4	3	2	2									2		3
CO5	3	2										1		3

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

TEXT BOOKS:

1. Fisher, Steven, Duane, Ja-Nae (2016). Start-up Equation – A Visual Guidebook for Building your Start-up, New Delhi: Tata McGraw-Hill
2. Ulrich, K. T. (2019). Product design and development. Tata McGraw-Hill Education, 7th edition

REFERENCE BOOKS:

1. Holt H., David (2017). Entrepreneurship: New Venture Creation, New Delhi: Pearson Education
2. Zimmerer, W. Thomas and Scarborough, M. Norman and Doug Wilson (2009). Essentials of Entrepreneurship and Small Business Management, 5/e; New Delhi: Prentice Hall India

Activity Based Learning (Suggested Activities in Class)

1. Proposal Handout & Presentation
   a. The Problem
   b. Need
   c. Existing Alternatives
   d. Potential Market
2. Wireframe Sketches, Target Specifications
   a. List Critical Customer Needs

b. For a., Prepare a List of Target Specifications

c. Prepare Wireframe

****

PRODUCT ENGINEERING & ENTREPRENEURSHIP

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

SEMESTER – VI

Subject Code	:	24OEXXXX	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P–S	:	3–0–0–0			

Course Learning Objectives:

This Course will enable students to:

1. **Analyze** and understand customer pain points .
2. **Devise** the best possible solution to build PoC to test in agile way.
3. **Explain** the different roles and responsibilities in collaborative product development
4. **Get the idea** of tools and technologies with automation in building product.
5. **Devise** the business plan and business pitch as potential start-ups.

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*** means it includes not only traditional lecture method, but different *type of teaching methods* adopted to develop the course outcomes.
2. ***Interactive Teaching: Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, 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***, by following Idea to PoC cycle leading to development of Business Plan and Pitch to potential investors.
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.

UNIT – I	**08 Hours**
INTRODUCTION, PRODUCT DEVELOPMENT PROCESS & ORGANIZATION, PRODUCT PLANNING, OPPORTUNITY IDENTIFICATION , IDENTIFYING CUSTOMER NEEDS ***(Text Book-1: Chapter 1 - 5)***	
UNIT – II	**08 Hours**
PRODUCT SPECIFICATIONS, CONCEPT - GENERATIONS, SELECTION, TESTING, PRODUCT ARCHITECTURE ***(Text Book-1: Chapter 6 - 10)***	
UNIT – III	**06 Hours**
INDUSTRIAL DESIGN, DESIGN FOR ENVIRONMENT, DESIGN FOR MANUFACTURING & SUPPLY CHAIN, ***(Text Book-1: Chapter 11 - 13)***	
UNIT – IV	**08 Hours**
PROTOTYPING, ROBUST DESIGN, PATENTS & INTELLECTUAL PROPERTY, SERVICE DESIGN ***(Text Book-1: Chapter 14 - 17).***	

UNIT – V	09Hours

BUSINESS PLAN, BUSINESS MODEL CANVAS (Below mentioned are reference links)

(***https://bklynlibrary.libguides.com/planning/businessplans***
https://bklynlibrary.libguides.com/planning/businessmodels
https://bklynlibrary.libguides.com/planning/pitchdecks)

Course Outcomes:

At the end of the course the student will be able to:

1. **Explain** the algorithm design techniques and standard Asymptotic notations. Analyze non-recursive and recursive algorithms to obtain worst-case running times of algorithms using asymptotic analysis
2. **Interpret** the brute-force, divide-and-conquer paradigms and explain when an algorithmic design situation calls for it. Recite algorithms that employ this paradigm. Synthesize divide-and-conquer algorithms. Derive and solve recurrences describing the performance of divide-and-conquer algorithms.
3. **Describe** the Decrease and Conquer, Transform and Conquer algorithm design techniques and analyze the performance of these algorithms.
4. **Identify and explain** the greedy technique, dynamic-programming paradigm as to when an algorithmic design situation calls for it. Recite algorithms that employ this paradigm. Synthesize dynamic-programming algorithms and analyze them.
5. **Describe** the Backtracking, Branch and Bound algorithm design paradigms and explain when an algorithmic design situation calls for it. Recite algorithms that employ these paradigms. **Explain** the limitations of algorithmic power.

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
CO1	3	3										2		3
CO2	3	3	2									2		3
CO3	3	3										1		3
CO4	3	3										2		3
CO5	3	3	2									1		3

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

PROJECT DELIVERABLES

1. Proposal Handout & Presentation
    1. The Problem
    2. Need
    3. Existing Alternatives
    4. Potential Market
2. PoC / Prototype, Target Specifications

1. Core solution
2. Resources (Environment, Automation, Time to Market)
3. Build PoC / Prototype

3. Create Business Plan and Model(s) and the Business Pitch
4. Register your business (Optional)

TEXTBOOKS:

1. Ulrich, K. and Eppinger, S.; Product Design and Development | 7th Edition, McGraw Hill; ISBN -13978-9390113231 (https://pdfhost.io/v/jcYs4cRR8_Karl_T_Ulrich_Steven_D_Eppinger_Maria_C_Yang_Product_Design_and_Development_2019_1pdf.pdf)

REFERENCE BOOKS:

1. Christensen, C.M.; The Innovator's Dilemma; Harvard Business Review Press; ISBN: 978-1-63- 369178-0
2. Chesbrough, H. W.; Open Innovation; Harvard Business Review Press; ISBN: 978-1-42-210283-1

E-Resources:

1. https://onlinecourses.swayam2.ac.in/imb19_mg01/preview
2. https://bklynlibrary.libguides.com
3. https://www.meetthedrapers.com/season5video
4. https://sharktank.co.in/episodes/

Activity Based Learning (Suggested Activities in Class)

1. Real world problem solving as a team building soft skills as well as getting exposure to product development skills and automation as appropriate apart from Business Plan / Model development and build Pitch like Shark tank / Meet the Drapers
2. Demonstration of PoC & Business Pitch

****

Computer Vision

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

SEMESTER – VI

Subject Code	:		**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	45 Hrs L
L–T–P-J	:	3–0–0-0	**CIE Marks**	:	60 Marks + 40 Marks

Prerequisites: Machine Learning, Deep Learning

Course Learning Objectives:

This Course will enable students to:

1. To understand the fundamentals of neural networks and deep learning
2. To develop CNN-based models for image classification, detection, and segmentation
3. To implement sequential models (RNN/LSTM) for vision tasks
4. To apply deep learning techniques to real-world datasets
5. To promote research and innovation in AI-driven vision systems.

Broad Course Description

This course introduces the core concepts of deep learning with a focus on computer vision. It covers neural network training, convolutional architectures, object detection, image segmentation, and sequence modelling using RNNs. Students gain hands-on experience in building, training, and evaluating models for real-world applications such as autonomous systems, healthcare imaging, and video analytics.

UNIT – I	**09 Hours**

MODULE 1: FOUNDATIONS OF NEURAL NETWORKS

Introduction, Feedforward Neural Networks, Activation Functions and Cost Functions, Backpropagation, Stochastic Gradient Descent and its Variants, Need for Regularization, Regularization Techniques (L1, L2, Dropout), Improving Training of Neural Networks, Code Walkthrough: Training and Testing Pipeline with Simple Dataset.
(Text Book-1,2 and 3)

UNIT – II	**09 Hours**

MODULE 2: CNNS FOR IMAGE CLASSIFICATION

Introduction to CNNs, Convolution and Pooling Layers, Backpropagation in CNNs, Data Augmentation Techniques, Transfer Learning and Fine-tuning, Visualization Techniques, Case Studies: VGGNet,
Inception-V3, ResNet-50, MobileNet

(Text Book-1,2 and 3)

UNIT – III	**08 Hours**

MODULE 3: CNNS FOR OBJECT DETECTION

Evaluation Metrics: IoU and mAP, Localization Techniques: Sliding Window and Regression Approach, Single-stage Detectors: YOLO, Two-stage Detectors: Fast R-CNN and Faster R-CNN, Single Shot MultiBox Detector (SSD).
(Text Book-1,2 and 3)

UNIT – IV	**09 Hours**

MODULE 4: CNNS FOR IMAGE SEGMENTATION Segmentation Concepts, Evaluation Metrics for Segmentation, Fully Convolutional Networks (FCN), SegNet Architecture, Dilated Convolutions, Case Studies: U-Net, Mask R-CNN. (Text Book-1,2 and 3)

UNIT – V	10 Hours

MODULE 5: RECURRENT NEURAL NETWORKS FOR VISION Introduction to RNNs, Backpropagation Through Time (BPTT), Long Short-Term Memory (LSTM), Temporal Modeling in Vision, Case Study: Video Understanding using CNNs and RNNs. (Text Book-1,2 and 3)
DLI: Applied Image Classification. Applied Object Detection. Convolutional Neural Networks. and Recurrent Neural Networks.

AI Integration & Authentic Intelligence: Reflection (AI² Pedagogy):

- Optimization algorithms, regularization, model training pipelines
- Transfer learning, feature extraction, pretrained models
- YOLO, Faster R-CNN, SSD models
- Pixel-level classification, U-Net, Mask R-CNN
- LSTM, sequence modeling, temporal learning
- Human learning vs gradient-based learning.
- Visual perception vs learned feature hierarchies
- Attention and localization in human vision vs AI.
- Scene understanding vs pixel-wise reasoning
- Memory and sequence learning in human's vs AI.

Applications:

8) Signal classification, predictive analytics, pattern recognition
9) Face recognition, defect detection, medical image classification
10) Autonomous vehicles, surveillance, robotics vision
11) Medical imaging, satellite image analysis, scene understanding
12) Video analytics, action recognition, gesture recognition

Mini Project Component:

) Real-time object detection system using webcam
a) Medical image segmentation prototype
b) Video-based action recognition system using CNN + LSTMIntelligent safety/security system.
c) Implement object detection using YOLO or SSD on a custom dataset
d) Perform image segmentation using U-Net and analyze results

Course		Bloom's

Outcome	Description	Taxonomy Level
At the end of the course the student will be able to:		
CO1	Explain and implement neural network training mechanisms	L3
CO2	Design CNN models for image classification tasks	L6
CO3	Develop object detection models using modern frameworks	L3
CO4	Apply image segmentation techniques	L4
CO5	Apply RNN/LSTM models for video and sequential vision tasks	L4

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	2	2	3	–	–	–	2	2	–	2	3
CO2	3	3	3	2	3	2	–	–	2	2	–	2	3
CO3	3	3	3	3	3	2	2	2	2	2	2	2	3
CO4	3	3	3	3	3	2	2	2	2	2	2	2	3
CO5	2	3	3	3	3	2	2	2	3	3	2	2	3

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

Textbooks:

1. Deep learning for computer vision, R. Shanmugamani, Packt Publishing, 2018.
2. Deep Learning, I. Goodfellow, Y. Bengio, A. Courville, MIT Press, 2016.

REFERENCE BOOKS:

1. Materials from Hugging Face: https://huggingface.co/learn/computer-vision course/

E resources:

1. IEEE research articles
2. NPTEL course on Deep Learning for Computer Vision by Dr. Vineet Balasubramanian: https://onlinecourses.nptel.ac.in/noc24_cs89/preview

Activity Based Learning (Suggested Activities in Class)

1. Implement and Compare Activation Functions in a Neural Network
2. Build and Train a CNN Model for Image Classification using a Real Dataset
3. Object Detection using Pre-trained YOLO Model on Custom Images
4. Image Segmentation using U-Net Architecture on Medical/Real-world Data
5. Video Classification using CNN + LSTM for Temporal Feature Learning.

****

Cryptography & Cyber Security

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

SEMESTER – VI

Subject Code	:		**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	45Hrs (30Hrs L + 15 HrsJ)
L–T–P-J	:	2–0–0-2			

Course Learning Objectives:

This Course will enable students to:

1. Master principles of symmetric and asymmetric cryptography.
2. Implement digital signatures, message authentication, and hash algorithms.
3. Understand key management, PKI, and internet security protocols.
4. Explore post-quantum, hardware-accelerated, and AI-linked cryptography.
5. Apply cryptographic design for secure enterprise and IoT systems.

Broad Course Description

This course explores mathematical foundations and practical implementations of cryptographic systems that ensure data confidentiality, integrity, and authenticity. Students learn symmetric and asymmetric algorithms, digital signatures, PKI, and secure communication protocols, linking modern cryptography to enterprise and IoT security requirements in the evolving cyber landscape.

UNIT – I	**06 Hours**

FOUNDATIONS OF CRYPTOGRAPHY:

Classical ciphers: substitution, transposition, frequency analysis. Modern cryptographic primitives: confidentiality, integrity, authenticity. Symmetric key concepts and random number generation. Modular arithmetic, Euler's theorem, modular inverses, GCD algorithms. Project Activity: Implement Caesar and Vigenère cipher in C/Python with brute-force attack.
(Text Book 1 : Ch. 1 1.1–1.5; Text Book 2 : Ch. 2 2.1–2.3)

UNIT – II	**06 Hours**

SYMMETRIC CRYPTOGRAPHY:

Feistel network and data encryption standards (DES, 3DES, AES).
Block vs stream ciphers, chaining modes (CBC, CFB, OFB, CTR). Key distribution and session key protocols.
Differential and linear cryptanalysis introduction.
(Text Book 1 : Ch. 3 3.1–3.5)
(Text Book 2 : Ch. 3 3.2–3.5)
Mini Lab: Performance comparison of AES and ChaCha20 (crypto libraries benchmarking).

UNIT – III	**06 Hours**

PUBLIC KEY CRYPTOGRAPHY & HASH FUNCTIONS:

RSA mathematical concept, key generation steps.
Diffie–Hellman key exchange and Elliptic Curve Cryptography (ECC). Message Authentication Codes and Hash functions (MD5, SHA-2/3). Digital signatures and PKI architectures.
(Text Book 1 : Ch. 6 6.1–6.5)
(Text Book 2 : Ch. 5 5.1–5.4)
Activity: Create digital certificate chain using OpenSSL.

UNIT – IV	**06 Hours**

SECURITY PROTOCOLS:

Secure Email (PGP, S/MIME), IPSec architecture and components. SSL/TLS handshake and cipher-suite negotiation.
Kerberos authentication system. Cryptanalysis labs (using Hashcat).

(Text Book 1 : Ch. 8 8.1–8.3) (Text Book 2 : Ch. 7 7.1–7.5) **Project Task:** TLS handshake trace and analysis using Wireshark.	
UNIT – V	**06 Hours**
ADVANCED CRYPTOGRAPHY: Quantum-resistant algorithms (Lattice, Code, Hash based). Blockchain cryptography (review of next domain course links). Hardware crypto modules (HSM, TPM). AI-based key management and anomaly detection. (Text Book 1 : Ch. 10 10.1–10.3) (Text Book 2 : Ch. 9 9.1–9.4) **Capstone:** Implement secure file sharing with AES + RSA hybrid encryption.	

AI Integration & Authentic Intelligence: Reflection (AI² Pedagogy)

1. GPU-accelerated encryption (NVIDIA CUDA Crypto).
2. Machine learning for encryption pattern monitoring.
3. Responsibility – privacy-preserving ML.

Applications:

1. Encrypting sensitive communications in banking and e-commerce systems.
2. Enabling secure web transactions through SSL/TLS protocols.
3. Implementing digital signatures and certificates in public key infrastructure.
4. Protecting IoT devices with lightweight cryptographic algorithms.
5. Designing post-quantum and AI-enhanced encryption models.

Mini Project Component

1. VPN prototype using AES + RSA.
2. Password-less authentication through ECC.
3. Timing attack simulation and mitigation.

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
CO1	Explain crypto principles and mathematical foundations.	L2
CO2	Implement symmetric & asymmetric algorithms practically.	L3
CO3	Apply digital signatures and PKI for security.	L3
CO4	Analyze protocols and demonstrate secure implementations.	L3
CO5	Design hybrid and AI-linked crypto solutions.	L4

Table: Mapping Levels of COs to POs / PSOs

COS	Program Outcomes (POs)											Program Specific Outcomes (PSOs)	
	PO1	PO2	PO3	PO4	PO5	PO 6	PO 7	PO 8	PO 9	PO1 0	PO1 1	PSO 1	PSO 2

CO1	3	2									2	2	
CO2	2	3	2	2	3						2	3	
CO3	2	2	3	2	3						2	3	
CO4	2	3		3	2						2	2	
CO5	2	2	3		3							3	

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

TEXT BOOKS:

1. William Stallings – Cryptography and Network Security, 8e.
2. Behrouz A. Forouzan – Cryptography and Network Security, McGraw-Hill.

NVIDIA Course Mapping (Industry Alignment)

NVIDIA DLI Course / Content	Mapped Units	Relevance
CUDA Accelerated Crypto and Privacy	II, V	Hardware crypto optimization
Morpheus Cybersecurity Framework	III, IV	AI for malware signature detection

E-RESOURCES:

1. nptel.ac.in
2. coursera.org
3. edx.org
4. cryptohack.org
5. developer.nvidia.com

Activity-Based Learning (Suggested Activities in Class)

1. Implement Caesar, Vigenère, and AES encryption using Python.
2. Demonstrate RSA key generation and message encryption.
3. Capture an HTTPS handshake and decode TLS parameters.
4. Create and verify a digital signature using OpenSSL.
5. Debate on the future of "Quantum-Safe Cryptography."

****

Gaming Design Fundamentals

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

SEMESTER – VI

Subject Code	:		**Credits**	:	03
Hours / Week	:	04 Hours	**Total Hours**	:	45Hrs(30Hrs L+15Hrs J)
L–T–P-J	:	2–0–0-2			

Course Learning Objectives:

This Course will enable students to:

1. Understand core principles of game design including the MDA (Mechanics, Dynamics, Aesthetics) framework.
2. Develop skills in game concept development, level design, difficulty tuning, and player experience design.
3. Learn narrative design, character development, branching storytelling, and world-building techniques.
4. Create professional Game Design Documents (GDD) and prototype interactive game concepts.

Broad Course Description

Gaming Design Fundamentals provides a comprehensive foundation in the art and science of designing engaging, playable games. The course approaches game design from both analytical and creative perspectives, examining what makes games compelling and how designers intentionally craft player experiences. Students study the MDA framework, flow theory, Bartle's player taxonomy, game mechanics and economy design, level design principles, narrative design (branching narrative with Twine), world-building, playtesting methodologies, and professional GDD writing. Digital prototyping tools including Figma and GDevelop are used alongside paper prototyping. The course integrates AI tools for concept generation while critically examining responsible and ethical game design practices including addiction mechanics and representation.

UNIT – I	**09 Hours**

Foundations of Game Design: What is a game – definitions, magic circle, play theory. History of game design from board games to video games. MDA Framework – Mechanics, Dynamics, Aesthetics. Core game loops and feedback loops. Game genres and their design conventions. Player psychology – motivation, challenge, and reward. Flow theory (Csikszentmihalyi) applied to game design. Types of fun – Bartle's player taxonomy (Killers, Achievers, Explorers, Socializers). (Ref: Jesse Schell, The Art of Game Design Ch. 1–5)

Textbook Reference:

• Jesse Schell, The Art of Game Design: A Book of Lenses, 3rd Ed – Ch. 1–5 (CRC Press, 2020)

• Richard Rouse III, Game Design: Theory and Practice, 2nd Ed – Ch. 1–3 (Wordware, 2005)

UNIT – II	**09 Hours**

Game Mechanics & Systems Design: Game mechanics – rules, goals, actions, resources. Space and time in games. Randomness and probability – dice, cards, procedural generation. Game economy – resources, sinks, sources, converters. Progression systems – leveling, skill trees, unlocks. Balancing

game systems – tools and iteration techniques. Emergent gameplay vs. scripted experiences. Core mechanic design exercise through paper prototyping. (Ref: Jesse Schell Ch. 8–14)

Textbook Reference:

• Jesse Schell – Ch. 8–14

• Brenda Romero & Ian Schreiber, Challenges for Game Designers – Ch. 1–5 (Cengage, 2009)

UNIT – III	**09 Hours**

Level Design & Spatial Design: Principles of level design – guidance, pacing, discovery, and flow. Tutorial design – teaching through play without explicit instruction. Spatial design considerations for 2D, 2.5D, and 3D environments. Environmental storytelling and atmosphere. Landmark design and wayfinding. Difficulty curves and challenge scaling strategies. Reward placement and collectibles design. Level design analysis using case studies from iconic games. (Ref: Scott Rogers, Level Up! Ch. 1–6)

Textbook Reference:

• Scott Rogers, Level Up! The Guide to Great Video Game Design, 2nd Ed – Ch. 1–6 (Wiley, 2014)
• Jesse Schell – Ch. 18–19

UNIT – IV	**09 Hours**

Narrative Design & World Building: Story vs. Narrative in games – ludonarrative harmony and dissonance. Story structures – three-act structure, Hero's Journey, non-linear narratives. Character design – archetypes, motivation, and backstory. Dialogue writing and branching narrative systems using Twine. World-building – lore, culture, factions, geography, and history. Cutscenes vs. in-engine environmental storytelling. Sound design's role in narrative – ambient, music, and voice acting overview. (Ref: Ernest Adams, Fundamentals of Game Design Ch. 12–14)

Textbook Reference:

• Ernest Adams, Fundamentals of Game Design, 3rd Ed – Ch. 12–14 (New Riders, 2014)
• Twine Documentation (twinery.org)

UNIT – V	**09 Hours**

Prototyping, Playtesting & GDD: Paper prototyping – rapid ideation and iteration. Digital prototyping tools – Figma, GDevelop, Twine. Playtesting methodologies – observational, think-aloud, feedback surveys. Analyzing playtest data and iterating on designs. Game Design Document (GDD) structure – vision, mechanics, narrative, UI, art direction. Pitching game concepts – elevator pitch and presentation. Ethics in game design – addiction mechanics, representation, microtransactions. Career pathways in game design and industry outlook. (Ref: Jesse Schell Ch. 24–26)

Textbook Reference:

Ernest Adams – Ch. 4–6
Jesse Schell – Ch. 24–26

NVIDIA DLI Free Course Mapping (Validated & Justified)

Validation Source: NVIDIA DLI Official Training Catalog (August 2024 PDF) + Live URL verification (April 2026). All courses below are confirmed FREE self-paced courses accessible at **learn.nvidia.com**. Only FREE courses are listed; paid instructor-led workshops excluded.

NVIDIA Deep Learning Institute (DLI) — FREE Self-Paced Course Mapping
Source: NVIDIA DLI Official Catalog (Aug 2024) + Live URL verification (Apr 2026) | Access: learn.nvidia.com | Only FREE self-paced courses included

Unit	NVIDIA DLI Free Course	Category	Duration	Cert.	Verified URL	Mapping Justification
Units 4–5	**Generative AI Explained**	Generative AI	1 Hr	—	courses.nvidia.com/course-v1:DLI+S-FX-07+V1	Essential foundation for AI-assisted concept generation, narrative AI, and procedural content creation tools

						evaluated in Units 4–5 — students must understand GenAI before critically using AI game design tools.
Unit 5	**Generative AI With Diffusion Models**	Generative AI / Image	2 Hrs	—	learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+T-FX-11+V1	Diffusion model image generation directly maps to AI-powered concept art generation in Unit 5's prototyping module — students use AI-generated images as concept art within their GDD and Figma prototypes.
Units 4–5	**Prompt Engineering With Llama 2**	Generative AI / LLM	2 Hrs	—	learn.nvidia.com/courses/course-detail?course_id=course-	Prompt engineering directly applicable to generating branching game dialogue, quest scripts, character
					v1:DLI+T-FX-12+V1	backstories, and world-lore content for narrative design in Units 4–5.
Unit 5	**Synthetic Data Generation for Training CV Models**	Graphics / CV	2 Hrs	—	learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+S-OV-24+V1	Synthetic visual data generation using NVIDIA Omniverse — introduces how game-world assets and AI playtesting datasets can be procedurally generated, extending Unit 5's AI playtesting discussion.

AI Integration & Authentic Intelligence: Reflection (AI² Pedagogy)

- **AI-Assisted Concept Generation:** Use ChatGPT and Midjourney to brainstorm game concepts, character designs, and world lore; critically evaluate and refine AI outputs.
- **Procedural Design Exploration:** Investigate AI-driven procedural content generation and its impact on game replayability, diversity, and designer control vs. automation.
- **Playtesting with AI Agents:** Discuss how AI playtesting agents identify design flaws before human testing and the limitations of AI vs. human player experience.
- **Narrative AI:** Build branching dialogue using AI-generated scripts; critically reflect on authorship, narrative coherence, and player agency in AI-assisted storytelling.

- **Responsible Game Design:** Critically analyze AI-driven monetization mechanics (dynamic pricing, targeted loot boxes) and discuss ethical design obligations.

Applications:

- Video Game Design and Development Studios
- Serious Games for Education, Healthcare, and Military Training
- Gamification of Enterprise Applications and Learning Platforms
- Interactive Narrative and Transmedia Experience Design
- Board Game and Tabletop Game Design and Publishing

Laboratory Component (1 Credit)

The lab emphasizes hands-on implementation, visualization, and AI-assisted debugging.

Lab Experiments

- Analyze and deconstruct core mechanics of 3 different game genres (platformer, puzzle, RPG).
- Create a paper prototype of a 2-player board game; conduct a structured playtest session with documented feedback.
- Design a complete level layout for a 2D platformer annotating challenge points, rewards, and tutorial flow.
- Use Twine to build a 3-scene branching narrative story with at least 3 distinct endings.
- Design a balanced resource economy for a strategy game; verify balance using a spreadsheet simulation.
- Create 3 distinct character concept profiles (protagonist, antagonist, supporting) with full backstory.
- Conduct a guided playtesting session for a peer's game prototype; write a structured feedback report.
- Design a progression system (XP, skill tree) for an RPG mapped to player psychology principles.
- Create a complete UI/UX wireframe for a mobile game HUD using Figma.
- Write a comprehensive 15-page Game Design Document (GDD) for an original game concept.

Mini Project Component

- **Industry-Driven Projects**

Some sample projects described below simulate the real software engineering problems.

Project 1 – Original Board Game Design

Problem: Design, prototype, and playtest a complete 2–4 player board game with unique mechanics and balance.

Concepts used: Game mechanics, Economy design, Paper prototyping, Playtesting

Industry relevance: Tabletop game publishing and educational game design.

Project 2 – Interactive Narrative Game

Problem: Branching narrative adventure game in Twine with complex choices, multiple

endings, and rich world-building.

Concepts used: Narrative design, World-building, Twine, Character design

Industry relevance: Interactive fiction and visual novel studios.

Project 3 – Complete GDD for Digital Game

Problem: Professional-quality Game Design Document for an original video game concept with all systems documented.

Concepts used: GDD writing, Systems design, Level design, UI/UX design

Industry relevance: Game industry pitching, indie development, graduate portfolios.

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Understand and apply foundational game design theories including MDA framework, flow theory, and player psychology.	L2
2	Design engaging game mechanics, balanced systems, and progression structures with documented design rationale.	L3
3	Apply level design principles to create well-paced, accessible, and discovery-driven game spaces.	L3
4	Develop compelling narrative designs and world-building documents with coherent story structures and characters.	L4
5	Create professional GDDs, conduct and analyze playtests, and iterate game designs based on evidence.	L5

Table: Mapping Levels of COs to POs / PSOs

	Program Outcomes (POs)										Program Specific Outcomes (PSOs)	
COs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**
CO1	2	2	2	2	3						2	2	3
CO2	2	3	3	3	2						2	3	2
CO3	2	3	3	3	2						2	3	2
CO4	2	3	3	3	3						2	3	3
CO5	2	3	3	3	3						3	3	3

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

Suggested Tools:

- Twine – Interactive Narrative Tool
- Figma – UI/UX Design and Prototyping
- GDevelop – No-code 2D Game Prototyping
- Miro / Mural – Collaborative Design Boards
- Notion / Google Docs – GDD Writing
- Canva / Midjourney – Concept Art Generation

TEXT BOOKS:

1. Jesse Schell, The Art of Game Design: A Book of Lenses, 3rd Ed, CRC Press, 2020.
2. Richard Rouse III, Game Design: Theory and Practice, 2nd Ed, Wordware, 2005.
3. Scott Rogers, Level Up! The Guide to Great Video Game Design, 2nd Ed, Wiley, 2014.

REFERENCE BOOKS:

1. Brenda Romero & Ian Schreiber, Challenges for Game Designers, Cengage Learning, 2009.
2. Ernest Adams, Fundamentals of Game Design, 3rd Ed, New Riders, 2014.

E-RESOURCES:

- https://gdcvault.com (GDC Vault)
- https://www.youtube.com/@extracredits (Extra Credits)
- https://gamedesignconcepts.wordpress.com
- https://learn.nvidia.com (NVIDIA DLI Free Courses)

Activity Based Learning (Suggested Activities in Class)

- Weekly game analysis assignments – deconstruct design pillars of assigned games.
- Design jams – 2-hour rapid game concept creation in teams of 3 students.
- Guest lectures from game designers at local indie studios or game companies.
- Game exhibition – students play and critique each other's Twine narrative games.

****

Quantum Algorithms

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

SEMESTER – VI

Subject Code	:		**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	45 Hrs (30 hrs L + 15 hrs J)
L–T–P-J	:	2–0–0-2			

Prerequisites: Python Programming, Basics of Quantum Computing

Course Learning Objectives:

This Course will enable students to:

1. Understand principles of quantum algorithms and computation.
2. Analyze major quantum algorithms like Deutsch–Jozsa, Grover's, and Shor's.
3. Explore quantum search techniques and computational complexity.
4. Study quantum noise models and error correction methods.
5. Implement quantum algorithms using Qiskit and evaluate performance.

Broad Course Description

The course *Quantum Algorithms* provides a comprehensive understanding of the principles and design of quantum algorithms that offer computational advantages over classical methods. It covers fundamental concepts such as quantum interference and parallelism, followed by key algorithms including Deutsch, Deutsch–Jozsa, Simon's, Grover's, and Shor's algorithms. The course further explores quantum search techniques, oracle models, and computational limits. It also addresses practical challenges such as quantum noise and error correction. Using Qiskit, students gain hands-on experience in implementing and analyzing quantum algorithms for real-world applications in optimization, cryptography, and data processing.

UNIT – I	**08 Hours**
Quantum Algorithmic Principles: Hadamard Gates, Phase Gate, Matrix Representation of Serial and Parallel Operations, Quantum Interference, Quantum Parallelism, Function Evaluation. (Text Book-1: Chapter 9)	
UNIT – II	**10 Hours**
Quantum Algorithms: Deutsch's Algorithm, Deutsch–Jozsa Algorithm, Simon's Algorithm, Grover's Search Algorithm, Shor's Factoring Algorithm (Text Book-2: Chapter 4)	
UNIT – III	**09 Hours**
Quantum Search Algorithms: Quantum Search Algorithm, Oracle Model, Algorithm Procedure, Geometric Visualization, Quantum Counting, NP-Complete Problems, Unstructured Search, Optimality and Black Box Limits (Text Book 3: Chapter 6)	
UNIT – IV	**09 Hours**
Quantum Noise and Error Correction: Single-Qubit Errors, Kraus Operators, Depolarization Channel, Bit Flip and Phase Flip Errors, Amplitude Damping, Phase Damping, Quantum Error Correction. (Text Book1: Chapter 12)	
UNIT – V	**09 Hours**

Implementation using Qiskit:
Quantum Circuit Implementation, Simulation of Algorithms, Execution on Quantum Simulators, Performance Evaluation

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

- **Quantum-Enhanced Intelligent Computation:** Leverage quantum principles such as superposition, interference, and parallelism to design AI systems capable of solving complex problems more efficiently than classical methods.
- **Algorithmic Intelligence using Quantum Models:** Apply quantum algorithms (Deutsch, Grover, Shor) to enhance search, optimization, and decision-making processes in AI applications.
- **Efficient Search & Optimization:** Utilize quantum search techniques and amplitude amplification to improve performance in large-scale AI problems such as pattern recognition and database search.
- **Noise-Aware AI Systems:** Incorporate quantum noise models and error correction techniques to ensure reliability, robustness, and accuracy in quantum AI computations.
- **Simulation-Driven AI Development:** Use Qiskit-based simulators to design, test, and validate quantum algorithms, enabling reproducible and scalable AI solutions.
- **Responsible & Explainable Quantum AI:** Ensure transparency, fairness, and ethical deployment of quantum algorithms by analyzing their impact, limitations, and computational boundaries.

Suggested Tools:

- Qiskit
- PennyLane
- Google Cirq
- Google Colab

Applications:

1. Quantum search for large-scale data retrieval and recommendation systems
2. Cryptography and cybersecurity using Shor's algorithm
3. Optimization problems in logistics, scheduling, and supply chains
4. Drug discovery and molecular simulation
5. Financial modeling and portfolio optimization
6. AI-based decision systems with quantum speedup
7. Pattern recognition and anomaly detection using quantum algorithms

Mini Project Component

Industry-Driven Projects (IBM / Google Quantum Style)

These projects simulate real-world **quantum algorithm and AI problem-solving scenarios**.

Project 1 – Quantum Search Engine

1. **Problem**: Design a search system using Grover's algorithm
2. **Concepts used**: Oracle model, amplitude amplification
3. **Industry relevance**: Search engines, recommendation systems

Project 2 – Noise-Aware Quantum Algorithm Simulator

1. **Problem:** Analyze the effect of noise and apply error correction
2. **Concepts used**: Kraus operators, noise channels, QEC
3. **Industry relevance**: Reliable quantum computing systems

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Explain quantum algorithmic principles and gate operations.	L2
2	Apply quantum algorithms to solve computational problems	L3
3	Analyze quantum search algorithms and their efficiency.	L4
4	Evaluate quantum noise and error correction techniques.	L5
5	Design and implement quantum algorithms using Qiskit.	L6

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	3	2	2	3						2	3	2
CO3	2	3	2	3	2						2	3	2
CO4	2	2		2	2	2	2	2			2	2	2
CO5	3	3	3	3	3	2	2	2	2	2	3	3	3

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

TEXT BOOKS:

1. David McMahon – *Quantum Computing Explained*, WILEY-INTERSCIENCE A John Wiley & Sons, Inc., Publication
2. Noson S. Yanofsky, Mirco A. Mannucci – *Quantum Computing for Computer Scientists*, Cambridge University Press
3. Michael A. Nielsen & Isaac L. Chuang, "Quantum Computation and Quantum Information" CAMBRIDGE UNIVERSITY PRESS Cambridge, New York, Melbourne, Madrid, Cape Town, Singapore

REFERENCE BOOKS:

1. Eric R. Johnston, Nic Harrigan, and Mercedes Gimeno-Segovia – Programming Quantum Computers, O'Reilly Media, 2019.
2. Maria Schuld and Francesco Petruccione – Machine Learning with Quantum Computers, Springer, 2018.

E-Resources:

- https://qiskit.org
- https://pennylane.ai
- https://quantum.ibm.com
- https://nptel.ac.in

Activity Based Learning (Suggested Activities in Class)

Activity 1: Simulation of quantum gates (Hadamard, Phase) using Qiskit
Activity 2: Implementation of Deutsch–Jozsa and Grover's algorithms
Activity 3: Visualization of quantum search using geometric interpretation
Activity 4: Analysis of quantum parallelism and interference through examples
Activity 5: Simulation of quantum noise and error correction techniques

****

Business Management Using Cloud

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

SEMESTER – VI

Subject Code	:	**Credits** :	03
Hours / Week	: 03 Hours	**Total Hours** :	45Hrs L
L–T–P-J	: 3–0–0-0		

Prerequisites: Basics of Computer Applications, Fundamentals of Management, Introduction to Cloud Computing (preferred)

Course Learning Objectives:

This Course will enable students to:

1. Understand cloud computing concepts and their role in business transformation.
2. Apply cloud platforms for business operations and decision-making.
3. Analyze cloud-based enterprise solutions and services.
4. Evaluate security, cost, and governance in cloud adoption.
5. Develop cloud-enabled business strategies and digital solutions

Broad Course Description

The course focuses on leveraging cloud computing technologies for modern business management and digital transformation. It introduces students to cloud service models, deployment strategies, and enterprise applications such as ERP, CRM, and analytics platforms delivered via the cloud. Students will explore how organizations use cloud platforms like Amazon Web Services, Microsoft Azure, and Google Cloud to optimize operations, enhance scalability, and drive innovation. The course also integrates concepts of cost optimization, cloud governance, data-driven decision-making, and security. Through case studies and practical exposure, students will develop skills required to manage and innovate cloud-based business systems.

UNIT – I	**09 Hours**
Introduction to Cloud Computing in Business: Cloud computing fundamentals, Service models: IaaS, PaaS, SaaS, Deployment models: Public, Private, Hybrid, Role of cloud in digital transformation, Business benefits and challenges *(Text Book 1: Chapters 1,2 | Ref 2)*	
UNIT – II	**09 Hours**
Cloud Platforms & Business Applications: Overview of AWS, Azure, Google Cloud, Cloud-based ERP, CRM, HRM systems, SaaS tools for business operations, Cloud-based collaboration tools Case studies of cloud adoption *(Text Book 2: Chapters 3,4 | Ref 1,3)*	
UNIT – III	**09 Hours**
Cloud Data Management & Analytics: Data storage and databases in cloud, Big data and business analytics, Cloud-based BI tools, Data-driven decision-making, Introduction to AI in cloud business *(Text Book 3: Chapters 5,6 | Ref 2,4)*	
UNIT – IV	**09 Hours**
Cloud Security, Governance & Costing: Cloud security fundamentals, Identity and access management (IAM) , Data privacy and compliance, Cloud governance frameworks, Cost models and optimisation strategies *(Text Book 1: Chapter 7 | Ref 3,4)*	
UNIT – V	**09 Hours**

Cloud Strategy & Digital Business Transformation: Cloud migration strategies**,** Business continuity and disaster recovery**,** DevOps and cloud integration **,** Digital transformation frameworks, Future trends: Serverless, Edge, AI-driven cloud
(Text Book 2,3 | Ref 1,4)

AI Integration & Digital Intelligence

- Business analytics using Google Analytics
- Cloud dashboards using Power BI
- AI services via Amazon Web Services AI tools
- Automation using Zapier
- Collaboration using Microsoft Teams

Applications

1. Cloud-based enterprise resource planning
2. Digital business transformation
3. E-commerce platforms
4. Financial analytics systems
5. Remote business operations

Project Component

Project 1 — Cloud-Based Business Solution

- Develop a cloud-enabled business model (e.g., CRM system)
- Tools: AWS / Azure

Project 2 — Cloud Cost Optimisation Analysis

- Analyse cost models and optimise cloud usage. Deliverables: Report + dashboard

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
CO1	Explain cloud computing concepts and business applications	L2
CO2	Apply cloud platforms for business operations	L3
CO3	Analyze cloud-based enterprise solutions	L4
CO4	Evaluate security, governance, and cost factors	L5
CO5	Design cloud-enabled business strategies	L6

Table: Mapping Levels of COs to POs / PSOs													
COS	Program Outcomes (POs)											Program Specific Outcomes (PSOs)	
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO 1	PSO 2
CO1	3	2								2		3	2
CO2	2	3	3		2					3		2	3
CO3	2	2	3	2	3					2		2	2
CO4	2	3	2	3	2	2		2		2		2	3
CO5	3	3	3	2	3	2		2	3	3	2	3	3

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

TEXT BOOKS:

1. Cloud Computing: Concepts, Technology & Architecture, Thomas Erl, Prentice Hall, 1st Edition, 2013.
2. Cloud Strategy, Gregor Hohpe, O'Reilly Media, 1st Edition, 2021.
3. Architecting the Cloud, Michael J Kavis, Wiley, 1st Edition, 2014

REFERENCE BOOKS:

1. The Lean Startup, Eric Ries, Crown Business, 2011.
2. CompTIA Cloud+ Study Guide, Todd Montgomery, Stephen Olson, Sybex, 2nd Edition, 2018.
3. AWS Certified Solutions Architect Official Study Guide, Ben Piper, David Clinton, Wiley, 2nd Edition, 2019.

E-RESOURCES:

1. NPTEL – Cloud Computing Courses
2. Coursera – Cloud & Business Analytics
3. edX – Digital Transformation
4. Amazon Web Services Training Portal
5. Microsoft Azure Learning

Cloud Industry Mapping (Alignment)

Platform / Certification	Mapped Units	Relevance
AWS Cloud Practitioner	Unit I, II	Cloud fundamentals and services
Azure Fundamentals	Unit I, IV	Governance, security, compliance
Google Cloud Digital Leader	Unit III, V	Analytics and business transformation
AWS Solutions Architect	Unit II, IV	Architecture and cost optimisation

Activity-Based Learning

1. Cloud service comparison (AWS vs Azure vs GCP)
2. Hands-on SaaS tool demonstrations
3. Case study analysis (Netflix, Amazon cloud adoption)
4. Cost estimation exercises
5. Group project on digital transformation

****

MOOC Courses

Sl. No.	Course Code	Course Title
1	*CID	Modern Computer Vision, IIT Madras [NPTEL] 12 WEEKS
2	*CID	Cryptography and Network Security, IIT Kharagpur [NPTEL] 12 WEEKS
3	*CID	Quantum Computing: Algorithms and Limitations Through the Query Model [NPTEL] 12 WEEKS

Note: CID = NPTEL Course ID

Building LLM with Prompt engineering

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

SEMESTER – VI

Subject Code	:		**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	45 Hrs (45Hrs L)
L–T–P-J	:	3–0–0-0			

Prerequisites: NLP with GENAI

Course Learning Objectives:

This Course will enable students to:

1. To understand the fundamental concepts of IoT architecture, M2M communication, and Industry 4.0 systems.
2. To analyze sensor and actuator technologies along with measurement and interfacing techniques.
3. To design and implement embedded systems for IoT applications using modern hardware platforms.
4. To evaluate IoT communication protocols and networking technologies for efficient data transmission.
5. To apply cloud computing, data analytics, and security principles in IoT-based solutions.

Broad Course Description

This course provides a comprehensive foundation in Sensors, Systems, and Internet of Things (IoT), covering the complete IoT stack from sensing and embedded systems to communication protocols and cloud-based analytics. Students will learn the principles of sensor technologies, embedded system design, and IoT networking, along with modern advancements such as edge computing, AI integration, and IoT security. The course emphasizes real-world engineering applications including smart cities, industrial automation, healthcare, and sustainable systems, while fostering critical thinking through AI-assisted learning and research-integrated pedagogy.

UNIT – I	**09 Hours**

Foundations of Large Language Models:

Fundamentals of language models including their history, evolution, and significance in Natural Language Processing (NLP), word embeddings and neural language models, Transformer architecture covering self-attention mechanism and encoder–decoder structure, encoder layers and encoder stack, and core components of Transformers such as positional encoding, batch normalization, and layer normalization.

(Text Book-1: Chapter 1: 1.1, 1.2, 1.3)
(Text Book-3: Chapter 1: 1.1, 1.2)
(Text Book-4: Chapter 1: 1.1)

UNIT – II	**09 Hours**

Architectures of Large Language Models:

Decoder-only Large Language Models including GPT architecture, pre-training vs. fine-tuning, and decoding strategies (greedy, beam, top-k, top-p); encoder-only models including BERT architecture, training objectives, and NLP applications; tokenization techniques such as BPE, WordPiece, and SentencePiece; and an overview of Small Language Models for resource-constrained environments.
(Text Book-1: Chapter 2: 2.1, 2.2, 2.3) (Text Book-2: Chapter 3: 3.1, 3.2) (Text Book-4: Chapter 1: 1.1,
1.2)

UNIT – III	09 Hours
Advanced Architectures and Adaptation of LLMs: Encoder–Decoder models including an introduction to models like BART and T5, understanding the text-to-text framework and zero-shot learning; advanced model components covering pre-training strategies, scaling laws, instruction fine-tuning, advanced attention mechanisms, and the Mixture of Experts approach; parameter-efficient fine-tuning (PEFT) techniques for efficient adaptation and inference along with their advantages; and efficient model adaptation through retrieval and tool augmentation, prompt engineering, chains, memory, and agents. (Text Book-1: Chapter 4: 4.1, 4.2) (Text Book-3: Chapter 2: 2.2, 2.3, 2.4) (Text Book-4: Chapter 2: 2.1, 2.2)	
UNIT – IV	**09 Hours**
Introduction to Prompt Engineering Definition, motivation, and importance of prompt engineering, language model control techniques such as temperature and top-p sampling, applications of prompt engineering, elements and structure of prompts including components like instruction, context, input, and output formatting, and prompt formulation strategies for different tasks. (Text Book-1: Chapter 5: 5.1, 5.3) (Text Book-3: Chapter 4: 4.1, 4.2, 4.3) (Text Book-4: Chapter 3: 3.2, 3.3)	
UNIT – V	**09 Hours**
Prompting for Tasks and Advanced Prompt Engineering Techniques: Applications of prompt engineering including summarization, question answering, classification, code generation, and reasoning; role-play and scenario-based prompts; few-shot and zero-shot prompting techniques; advanced methods such as chain-of-thought (CoT), knowledge-augmented prompting, self-consistency, program-aided language models (PAL), and ReAct frameworks; along with hands-on prompt engineering exercises. (Text Book-1: Chapter 7: 7.1, 7.2) (Text Book-3: Chapter 5: 5.1, 5.2, 5.3) (Text Book-4: Chapter 6: 6.1, 6.3)	
• Building Transformer-Based NLP Applications • Building Conversational AI Applications • Building LLM Applications With Prompt Engineering • Rapid Application Development with Large Language Models	

AI Integration & Authentic Intelligence: Reflection (AI² Pedagogy)

- AI-assisted understanding of language models, word embeddings, and neural LMs for NLP tasks.
- AI-driven implementation of large language models (GPT, BERT), tokenization techniques, and small language model applications.
- AI-enabled design and optimization of encoder–decoder models, advanced architectures, and parameter-efficient fine-tuning techniques.
- AI-guided prompt engineering, including prompt design, control strategies, and task-specific prompt formulation.
- AI-based application of prompting techniques for tasks such as summarization, reasoning, code generation, and advanced frameworks like CoT, PAL, and ReAct.
- Students evaluate limitations, biases, and ethical concerns in language models and NLP systems.

- Students analyze reliability, generalization, and potential biases in large language models and tokenization methods.
- Students assess scalability, efficiency, and trade-offs in advanced architectures and fine-tuning approaches.
- Students examine risks of prompt misuse, bias amplification, and control limitations in prompt engineering.
- Students evaluate trust, interpretability, and ethical implications of advanced prompting techniques and AI-generated outputs.

Applications:

- Language Modeling Systems (Word Embeddings, Neural Language Models, NLP Applications)
- Large Language Models (GPT, BERT, Tokenization Techniques, Small Language Models)
- Advanced NLP Architectures (Encoder–Decoder Models, BART, T5, PEFT, Model Optimization)
- Prompt Engineering Systems (Prompt Design, Control Strategies, Task-specific Prompting)
- AI-driven NLP Applications (Summarization, Question Answering, Code Generation, Reasoning, Advanced Prompting Frameworks)

Mini Project Component

Some sample projects described below simulate the real software engineering problems.

Project 1 – Intelligent Text Summarization System.

- Problem: Develop a system to automatically generate concise summaries from large text documents.
- Concepts used: Transformer models, encoder–decoder architecture (BART/T5), tokenization, fine-tuning.

- Industry relevance: News platforms, research tools, companies like Google, OpenAI.

Project 2 – AI-powered Question Answering System

- Problem: Build a system that can answer user queries based on given context or documents.
- Concepts used: BERT-based models, embeddings, retrieval-augmented generation (RAG), prompt engineering.
- Industry relevance: Chatbots, search engines, virtual assistants (Amazon Alexa, Google Assistant).

Project 3 – Code Generation Assistant

- Problem: Create an AI tool that generates code snippets based on natural language prompts.
- Concepts used: GPT-based models, prompt engineering, decoding strategies, fine-

tuning.

- Industry relevance: Developer tools (GitHub Copilot), software companies, productivity platforms.

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
CO1	Explain fundamentals of language models, including word embeddings, neural LMs, and Transformer architecture.	L2
CO2	Apply concepts of large language models such as GPT and BERT, along with tokenization techniques for NLP tasks.	L3
CO3	Design and utilize advanced architectures including encoder–decoder models and parameter-efficient fine-tuning methods.	L3
CO4	Analyze and implement prompt engineering techniques for controlling and optimizing language model outputs.	L3
CO5	Develop AI-driven NLP applications such as summarization, question answering, and code generation using advanced prompting frameworks.	L3

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	2	2	2				2		2	2	2
CO2	3	2	2	2	2				2		2	2	2
CO3	3	2	2	2	2				2		2	2	2
CO4	3	2	2	2	2				2		2	2	2
CO5	3	2	2	3	2				2		2	2	2

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

TEXT BOOKS:

1. Speech and Language Processing – Daniel Jurafsky and James H. Martin, 3rd Edition (Draft), Pearson.Comprehensive resource covering fundamentals of NLP, language models, and modern deep learning approaches.
2. Transformers for Natural Language Processing – Denis Rothman, 2nd Edition, Packt, 2021.Focuses on Transformer models, BERT, GPT, and real-world NLP applications.
3. Natural Language Processing with Transformers – Lewis Tunstall, Leandro von Werra, and Thomas Wolf, O'Reilly, 2022.Covers Hugging Face ecosystem, fine-tuning, and deployment of transformer-based models.
4. Generative AI with Python and PyTorch – Joseph Babcock et al., Packt, 2023. Introduces generative models, LLMs, and practical implementation using Python.

REFERENCE BOOKS:

1. Deep Learning – Ian Goodfellow, Yoshua Bengio, and Aaron Courville, MIT Press, 2016.
2. Language Models for NLP – Sebastian Raschka (Latest Editions / Resources).

3. Hands-On Large Language Models – Jay Alammar and Maarten Grootendorst, O'Reilly, 2024.
4. Prompt Engineering for Generative AI – James Phoenix and Mike Taylor, O'Reilly, 2024.

E-RESOURCES:

1. NPTEL (Courses on NLP, Deep Learning, and AI by IITs)
2. MIT OpenCourseWare (Machine Learning, NLP, and Artificial Intelligence fundamentals)
3. IEEE research articles on NLP, Large Language Models, and Generative AI

Activity Based Learning (Suggested Activities in Class)

5. Real world problem solving using group discussion. E.g., Designing an NLP-based application such as a chatbot or text summarization system.
6. Role play. E.g., Students act as tokenizer, encoder, decoder, and attention mechanism to simulate Transformer-based data flow.
7. Demonstration of solution to a problem through programming. E.g., Implementing a simple NLP model using Python/Hugging Face for tasks like classification or question answering.
8. Flip class activity. E.g., Prompt engineering techniques, LLM architectures (GPT, BERT), and advanced NLP applications.

****

IoT-Based Data Analytics for Heath Care

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

SEMESTER – VI

Subject Code	:		**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	45 (30Hrs (L) + 15Hrs(J))
L–T–P-J	:	2–0–0-2			

Prerequisites:

Course Learning Objectives:

This Course will enable students to:

1. Understand IoT architecture and standards in healthcare environments.
2. Establish secure data collection and transmission models.
3. Apply AI/ML techniques for health data analytics.
4. Integrate blockchain for data integrity and sharing.
5. Develop sustainable smart health analytics solutions..

Broad Course Description

This course integrates Internet of Things concepts with secure data analytics for healthcare applications. It covers sensor networks, IoT communication protocols, data acquisition pipelines, AI-enabled health analytics, and privacy-preserving data sharing using blockchain, enabling students to design sustainable and secure smart healthcare solutions.

UNIT – I	**06 Hours**

IoT ARCHITECTURE AND HEALTH DEVICES:

Sensing technologies (BP, HR, ECG). IoT stack overview – device, gateway, cloud, application. Communication protocols (MQTT, CoAP, BLE, LoRa). Data formats (JSON, CBOR). Healthcare standards – HL7, FHIR.

(Text Book 1 : Ch. 1 1.1–1.6; Text Book 2 : Ch. 2 2.1–2.3)

UNIT – II	**06 Hours**

IoT NETWORKING AND SECURITY:

End-to-end communication models. Security challenges in IoT and M2M systems. Device authentication. Cloud and edge security. Secure IoT middleware and firmware updates

(Text Book 1 : Ch. 3 3.1–3.5; Text Book 2 : Ch. 4 4.1–4.4)

UNIT – III	**06 Hours**

DATA ACQUISITION AND STORAGE:

Streaming data pipelines (MQTT, Kafka). Time-series databases (InfluxDB, MongoDB). Data cleaning, synchronization, missing value imputation. Data privacy – anonymization and differential privacy.

(Text Book 1 : Ch. 5 5.1–5.5;
Text Book 2 : Ch. 6 6.1–6.3)

UNIT – IV	**06 Hours**

AI-BASED HEALTH DATA ANALYTICS:

Predictive analytics models (SVM, Random Forest, CNN for ECG). Federated learning for distributed data models. Early warning systems and real-time visual dashboards. Performance evaluation metrics (precision, recall, ROC curve). ML algorithms for sensor data.

(Text Book 2 : Ch. 8 8.1–8.5)

UNIT – V	**06 Hours**

BLOCKCHAIN INTERFACING AND ETHICAL USE:

Smart contracts for health data access control. Blockchain data storage formats and timestamping. Audit trails and data verification process. GDPR and HIPAA compliance integration.

(Text Book 1 : Ch. 7 7.1–7.4; Text Book 2 : Ch. 9 9.2–9.4)

AI Integration & Authentic Intelligence: Reflection (AI² Pedagogy)

1. Edge AI for low-power devices.
2. Responsible AI and Bias Mitigation in Clinical Analytics.

Applications:

3. Monitoring patient vitals using connected health sensors.
4. Detecting medical anomalies through AI-based analytics on IoT data.
5. Storing and processing patient data securely on cloud and edge platforms.
6. Integrating blockchain for tamper-proof health records.
7. Building predictive healthcare dashboards for early diagnosis.

Mini Project Component

The following can be considered as the project-based activities in the course:

1. **Project:** Connectivity simulation of multiple sensor nodes (Raspberry Pi / Arduino).
2. **Mini Lab:** Implement token-based device authentication.
3. **Activity:** Design a cloud data pipeline using AWS IoT Core or ThingSpeak.
4. **Mini Project:** Anomaly detection in vital sign datasets.
5. **Capstone:** Prototype "patient-data ledger" with integrity check.

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
CO1	Describe IoT architecture and standards for health systems.	L2
CO2	Implement secure IoT networks using cryptographic techniques.	L3
CO3	Develop data pipelines for health analytics.	L3
CO4	Apply AI models to analyze sensor data.	L3
CO5	Integrate blockchain for privacy-preserving computation.	L4

Table: Mapping Levels of COs to POs / PSOs

COS	Program Outcomes (POs)											Program Specific Outcomes (PSOs)	
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO 1	PSO 2
CO1	3	2			2						2	2	
CO2	2	3	3		3			2			2	2	
CO3	2	3	3	2	3						2	3	
CO4	2	3	3	2	3						2	3	
CO5	2	2	3		3			2			2	3	

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

TEXT BOOKS:

1. Arshdeep Bahga, Vijay Madisetti – IoT: A Hands-On Approach, Universities Press.
2. Dietmar Putz – AI for IoT Applications, Springer.

NVIDIA Course Mapping (Industry Alignment)

NVIDIA DLI Course / Content	Mapped Units	Relevance
Edge AI and Jetson for IoT	III, IV	Edge ML deployments
Deep Learning for Anomaly Detection	IV	Predictive Health analytics

E-RESOURCES:

1. nptel.ac.in
2. coursera.org
3. aws.amazon.com
4. developer.nvidia.com
5. cse29-iiith.vlabs.ac.in

Activity-Based Learning (Suggested Activities in Class)

1. Connect a sensor to an IoT platform (AWS IoT Core / ThingSpeak).
2. Record and visualize temperature or heart-rate data streams.
3. Perform data cleaning on real IoT sensor files using Python.
4. Apply a simple ML algorithm for anomaly detection in vitals.
5. Present a use case on secure IoT ecosystem in healthcare.

****

Game Development using Unity

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

SEMESTER – VI

Subject Code	:		**Credits**	:	03
Hours / Week	:	04 Hours	**Total Hours**	:	30HrsL +15Hrs J
L–T–P-J	:	2–0–0-2			

Course Learning Objectives:

This Course will enable students to:

1. Develop proficiency in Unity for creating 2D and 3D games.
2. Learn C# scripting for game logic, player controls, and game systems.
3. Implement physics, animation, AI behaviors, and audio in Unity.
4. Build and deploy complete games for desktop and mobile platforms.

Broad Course Description

Game Development using Unity is a project-driven course covering Unity editor fundamentals, C# scripting (MonoBehaviour lifecycle), physics, 2D Tilemap, animation (Animator/Blend Trees/Cinemachine), NavMesh AI, Unity UI (uGUI/TextMeshPro), audio management, save systems, Shader Graph, particle systems (Niagara-style), optimization, and deployment to Android/iOS/PC.

UNIT – I	**06 Hours**

Unity Foundations & C# Scripting: Unity Editor – Hierarchy, Inspector, Project, Scene views, GameObjects, Components, Prefabs, MonoBehaviour lifecycle, Input System, Instantiate/Destroy, Tags/Layers, Scene management.

• Harrison Ferrone, "Learning C# by Developing Games with Unity" 7th Ed Ch. 1–5 (Packt, 2023)
• Unity Learn Official Curriculum (learn.unity.com)

UNIT – II	**06 Hours**

Physics, Collision & 2D Game Development: Rigidbody and Collider – 3D and 2D, Physics materials, Collision detection events, Layer-based collision, 2D workflow – Sprites, Sprite Atlas, Tilemap, 2D joints, Coroutines, Object pooling.

• Harrison Ferrone Ch. 7–10
• John P. Doran, "Unity 2022 by Example" Ch. 2–4 (Packt, 2023)

UNIT – III	**06 Hours**

Animation & Character Systems: Animation clips, Animator Controller, Transitions, Parameters, Blend Trees, Root motion vs. in-place animation, Cinemachine, Character Controller vs. Rigidbody, Ragdoll physics, IK basics, Particle System.

• John P. Doran Ch. 5–7
• Unity Animation Documentation (docs.unity3d.com/Manual/AnimationSection.html)

UNIT – IV	**06 Hours**

AI, UI & Game Systems: NavMesh AI – patrol, chase, attack state machines, Scriptable Objects, Unity uGUI – Canvas, Button, Text, Image, Slider, Health bars, score HUD, TextMeshPro, Audio Source/Mixer, Save system – PlayerPrefs/JSON, Game Manager/Singleton.

• Harrison Ferrone Ch. 11–14
• John P. Doran Ch. 8–10

UNIT – V	**06 Hours**

Lighting, Shaders, Optimization & Deployment: Unity lighting – Baked vs. Real-time, Light Probes,

Reflection Probes, Shader Graph, Post-processing, Performance optimization – batching, LOD, Profiler, Mobile build settings, Android/iOS, Unity Asset Store, Publishing to Google Play/App Store/itch.io.

- John P. Doran Ch. 11–13
- Unity Performance Best Practices (docs.unity3d.com/Manual/BestPracticeGuides.html)

NVIDIA Deep Learning Institute (DLI) Course Mapping

The following NVIDIA DLI courses are recommended to supplement the subject. Free courses are accessible at learn.nvidia.com. Paid instructor-led workshops (ILT) are available via NVIDIA partner institutions.

UNIT	NVIDIA DLI Course Name	Type	Duration	URL / Code
Units 3–4	Getting Started With Deep Learning Foundational deep learning – prerequisite for Unity ML-Agents integration and AI NPC development covered in Unit 4.	Self-Paced	2 Hrs	FREE DLI+C-FX-01+V1
Unit 4	Getting Started With AI on Jetson Nano Practical AI deployment on edge hardware – models trained here can be integrated as Unity NPC AI via ONNX export.	Workshop (ILT)	8 Hrs	PAID DLI-ILT-GAIJN
Units 3–5	Build Beautiful, Custom UI for 3D Tools on NVIDIA Omniverse Self-paced UI design for 3D platforms – directly applicable to Unity UGUI/HUD design principles and 3D tool UI patterns.	Self-Paced	2 Hrs	FREE DLI+S-OV-12+V1
Units 4–5	Generative AI Explained GenAI overview – applicable to AI-generated game assets, procedural content, and ML-Agents-based NPC behavior.	Self-Paced	1 Hr	FREE DLI+S-FX-07+V1
Unit 5	Bootstrapping Computer Vision Models with Synthetic Data ILT: use NVIDIA Omniverse to generate synthetic training	Workshop (ILT)	8 Hrs	PAID DLI-ILT-BCVMSD

	data for computer vision – applicable to AI enemy vision systems and game perception.			
Unit 5	Synthetic Data Generation for Training Computer Vision Models Self-paced: generating synthetic game-world data for training vision-based AI models – advanced Unity AI integration topic.	Self-Paced	2 Hrs	FREE DLI-SP-SDGCVM

AI Integration & Authentic Intelligence: Reflection (AI² Pedagogy)

- AI-Assisted Code Generation: Use GitHub Copilot in Unity to generate C# scripts.
- Procedural Content with AI: Implement procedural level generation.
- Behavior AI Reflection: Compare state machine AI with ML-based agents.
- Automated Playtesting: Explore Unity ML-Agents for automated testing.
- AI Art Pipeline: Use AI-generated 2D/3D assets and discuss IP and quality.

Applications

- Mobile and PC Game Publishing
- Serious Games for Healthcare and Education
- Architectural VR Walkthroughs
- Interactive Training Simulations
- AR Applications for Marketing

Laboratory Component (1 Credit)

The lab emphasizes hands-on implementation, visualization, and AI-assisted debugging.

Lab Experiments

- Build 2D platformer with player movement, jump, and obstacle collision.
- Create Tilemap-based level with rule tiles and scrolling camera.
- Implement Animator Controller with 4 states and blend tree transitions.
- Design 3D third-person controller with Cinemachine.
- Build NavMesh-based enemy AI with patrol, detection, chase.
- Create complete HUD with health bar, score, timer, and pause menu.
- Implement save/load system using JSON serialization.
- Build particle effect system for visual feedback.
- Implement mobile touch controls and build for Android/iOS.
- Submit fully playable original game with win/lose conditions and sound.

Mini Project Component

Industry-driven projects that simulate real software engineering problems.

Project 1 – 2D Mobile Platformer

Problem: 2D platformer with 3 levels, enemies, collectibles, score, mobile touch controls.

Concepts used: Unity 2D, Tilemap, Animation, Physics, Mobile deployment

Industry relevance: Mobile game publishing on Google Play / App Store.

Project 2 – 3D Top-Down Shooter

Problem: 3D shooter with AI enemies, weapon systems, power-ups, and progression.

Concepts used: 3D Unity, NavMesh AI, Scriptable Objects, UI

Industry relevance: Action game genre – widely played across platforms.

Project 3 – Puzzle Adventure Game

Problem: 3D puzzle-adventure with physics puzzles, inventory system, and narrative.

Concepts used: Unity Physics, Game Manager, Save system, Lighting

Industry relevance: Indie puzzle games on Steam and itch.io.

Course Outcomes

CO	Description	Bloom's
At the end of the course the student will be able to:		
CO1	Navigate Unity editor proficiently and write C# scripts for game logic, input handling, and scene management.	L3
CO2	Implement 2D/3D physics simulations and collision detection; create complete 2D games using Tilemap workflows.	L3
CO3	Design character animation systems using Animator Controller, Blend Trees, and Cinemachine.	L3
CO4	Build complete game systems: AI pathfinding, UI/HUD, audio management, and persistent save systems.	L4
CO5	Optimize, polish, and deploy a complete original game to target platforms.	L5

Table: Mapping Levels of COs to POs / PSOs

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PSO 1	PSO 2
CO 1	3	2	3	3	3						2	3	3
CO 2	3	3	2	3	3						2	3	2
CO 3	3	3	3	3	2						2	3	2
CO 4	3	3	3	3	3						3	3	3
CO 5	3	3	3	3	3						3	3	3
3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)													

Suggested Tools

- Unity 2022 LTS / 2023
- Visual Studio / Rider
- Cinemachine

- TextMeshPro
- Unity ML-Agents Toolkit
- Blender

TEXT BOOKS:

1. John P. Doran, "Unity 2022 by Example", Packt Publishing, 2023.
2. Jason Gregory, "Game Engine Architecture", 3rd Ed, CRC Press, 2018.
3. Harrison Ferrone, "Learning C# by Developing Games with Unity", 7th Ed, Packt, 2023.

REFERENCE BOOKS:

1. Unity Technologies, Official Unity Manual and Scripting Reference (docs.unity3d.com).
2. Adam Thorn, "The Unity Game Developer's Cookbook", Addison-Wesley, 2013.

E-RESOURCES:

- Unity Learn (learn.unity.com)
- Brackeys YouTube Channel
- Game Dev TV (gamedev.tv)
- NVIDIA DLI: learn.nvidia.com

Activity Based Learning (Suggested Activities in Class)

- Monthly game jams using Unity (48-hour events).
- Portfolio reviews – peer and instructor critique of game builds.
- Publish games to itch.io for community feedback.

****

Quantum Machine Learning

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

SEMESTER – VII

Subject Code	:		**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	45 Hrs (30hrs T + 15hrs J)
L–T–P-J	:	2–0–0-2			

Prerequisites: Linear Algebra, Probability, Machine Learning

Course Learning Objectives:

This Course will enable students to:

1. Understand fundamental concepts of machine learning and their quantum extensions.
2. Learn techniques for encoding classical data into quantum states.
3. Analyze variational quantum circuits as machine learning models.
4. Explore quantum kernel methods and quantum optimization techniques.
5. Develop applications using Quantum Machine Learning frameworks.

Broad Course Description

The course Quantum Machine Learning provides a comprehensive understanding of integrating machine learning techniques with quantum computing principles to develop advanced intelligent systems. It covers fundamental concepts of machine learning, followed by quantum data representation methods such as basis and amplitude encoding. The course explores variational quantum circuits and quantum neural networks as learning models, along with quantum kernel methods and optimization using Ising models. Through real-world case studies, students gain practical insights into applications in healthcare, finance, and image processing, equipping them to design and analyze hybrid quantum–classical machine learning solutions.

UNIT – I	**08 Hours**

Introduction to Machine Learning:

Background, The Toy Example of Quantum Algorithm for classification, Examples of Typical machine learning problems, The Three Ingredients of a Learning Problem, Risk Minimisation in Supervised Learning, Training in Unsupervised Learning, Methods in Machine Learning.

(Text Book-1: Chapter 1, 2)

UNIT – II	**10 Hours**

Representing Data on a Quantum Computer:

Encoding Binary Inputs into Basis States, Arbitrary State Preparation for Amplitude Encoding, Encoding Inputs as Time Evolutions, Encoding a Dataset via the Hamiltonian, Data Encoding as a Feature Map.

(Text Book-1: Chapter 4)

UNIT – III	**09 Hours**

Variational Circuits as Machine Learning Models: How to Interpret a Quantum Circuit as a Model, Which Functions Do Variational Quantum Models Express? Training Variational Quantum Models, Quantum Circuits and Neural Networks. (Text Book 1: Chapter 5)	
UNIT – IV	**09 Hours**
Quantum Models as Kernel Methods and Ising Model: The Connection Between Quantum Models and Kernel, Quantum Computing, Feature Maps and Kernels, Examples of Quantum Kernels, The RKHS of Quantum Kernels, Kernel-Based Training, Quantum Extensions of Ising Models, Quantum Annealing. (Text Book1: Chapter 6: 6.1, 6.2, 6.3, 6.4, 6.5. Chapter 8: 8.1, 8.2)	
UNIT – V	**09 Hours**
Case Studies and Applications of QML: Healthcare Diagnosis using QML, Financial Market Prediction, Image Classification using Quantum Feature Maps, Image Classification using Quantum Feature Maps	

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

- **Quantum-Enhanced Machine Learning:** Integrate classical machine learning concepts with quantum computing principles to design hybrid models that improve learning efficiency and scalability.
- **Advanced Data Representation:** Utilize quantum data encoding techniques (basis states, amplitude encoding, feature maps) to represent complex datasets in high-dimensional spaces for improved model performance.
- **Variational Learning Models:** Apply Variational Quantum Circuits (VQC) and Quantum Neural Networks (QNN) as trainable models, combining quantum circuits with classical optimization techniques.
- **Kernel-Based Quantum Intelligence:** Leverage quantum kernels and feature maps to enhance classification and pattern recognition tasks, enabling more expressive learning models.
- **Optimization and Decision Systems:** Use quantum annealing and Ising models to solve complex optimization problems in AI applications efficiently.
- **Responsible & Explainable Quantum AI:** Ensure fairness, transparency, and interpretability in quantum machine learning models while addressing challenges such as bias, uncertainty, and reproducibility.

Suggested Tools:

- Tensorflow Quantum
- PennyLane
- Google Colab
- IBM Quantum Expereince
- Qiskit Machine Learning

Applications:

- Healthcare diagnosis and disease prediction using quantum ML models
- Financial market prediction and risk analysis
- Image classification and pattern recognition using quantum feature maps

- Optimization problems in logistics and resource allocation
- Fraud detection and anomaly detection systems
- Drug discovery and molecular data analysis
- Smart decision systems using hybrid quantum–classical AI

Mini Project Component

Industry-Driven Projects (IBM / Google Quantum AI Style)

These projects simulate real-world **quantum machine learning and intelligent system applications**.

Project 1 – Quantum Classification Model

- **Problem:** Develop a classification model using Variational Quantum Circuits (VQC)
- **Concepts used:** Feature maps, parameterized circuits, training
- **Industry relevance:** Healthcare, finance, predictive analytics

Project 2 – Case Study-Based QML Application

- **Problem:** Implement a real-world case study (healthcare/finance/image classification)
- **Concepts used:** QML pipeline, hybrid models, performance evaluation
- **Industry relevance**: Industry-ready AI solutions

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Explain machine learning fundamentals and quantum learning concepts	L2
2	Apply data encoding techniques for representing classical data in quantum systems	L3
3	Analyze variational quantum circuits as machine learning models	L4
4	Evaluate quantum kernel methods and quantum optimization approaches.	L3
5	Evaluate quantum kernel methods and quantum optimization approaches.	L6

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	1							2	
CO2	3	3	2	2	3							3	2
CO3	2	3	2	3	2							3	2
CO4	2	3	2	3	2							3	2

CO5	3	3	3	3	3						2	3	3

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

TEXT BOOKS:

1. Maria Schuld · Francesco Petruccione, "**Machine Learning with Quantum Computers",** Second Edition, Springer.

REFERENCE BOOKS:

1. Maria Schuld and Francesco Petruccione – *Supervised Learning with Quantum Computers*
2. Peter Wittek – *Quantum Machine Learning: What Quantum Computing Means to Data Mining*
3. Michael Nielsen and Isaac Chuang – *Quantum Computation and Quantum Information*

E-Resources:

- https://qiskit.org
- https://pennylane.ai
- https://quantum.ibm.com
- https://nptel.ac.in

Activity Based Learning (Suggested Activities in Class)

Activity 1: Quantum data encoding experiment
Activity 2: Quantum circuit based classifier implementation
Activity 3: Hybrid quantum–classical ML model design
Activity 4: Mini project on quantum machine learning application

****

Business to Business Marketing

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

SEMESTER – VI

Subject Code	:	**Credits**	03
Hours / Week	: 03 Hours	**Total Hours** :	45 Hrs L
L–T–P-J	: 3–0–0-0		

Course Learning Objectives:

This Course will enable students to:

1. To understand the fundamentals of B2B marketing and the structure of business markets.
2. To analyze organizational buying behaviour and decision-making processes.
3. To apply strategic marketing concepts for managing products, services, and business portfolios.
4. To evaluate segmentation, targeting, positioning, and pricing strategies in B2B markets.
5. To develop skills in business communication, channel management, and relationship marketing.

Broad Course Description

This course provides a comprehensive understanding of Business-to-Business (B2B) marketing, focusing on how organizations create value and build relationships in industrial and institutional markets. It covers the characteristics of business markets, organizational buying behaviour, and strategic marketing processes, followed by product, service, and channel management in B2B contexts. The course also explores segmentation, targeting, positioning, pricing strategies, and relationship marketing, along with modern digital marketing practices. Emphasis is given to business communication, sales force management, and logistics to help students develop practical skills for managing complex B2B environments and making data-driven marketing decisions.

UNIT – I	**09 Hours**
Introduction to B2B Marketing: Business marketing and Business market customers, Market structure, Environment and Characteristics of Business Marketing, Strategic role of marketing, Commercial enterprises, Commercial and institutional customers.	
UNIT – II	**09 Hours**
Organizational Buying and Buyer Behaviour: Organizational buyers' decision process - A Stepwise Model and A Process Flow Model, Organizational and business markets - Government as a customer - Commercial enterprises - Commercial and institutional Customers.	
UNIT – III	**09 Hours**
B2B Marketing Strategy: Strategy making and strategy management process, Industrial product strategy–Managing Products for Business Markets-Managing Services for Business Markets-Managing Business Market Channels the Growth-Share Matrix, Multifactor Portfolio Matrix, The Balanced Scorecard.	
UNIT – IV	**09 Hours**
B2B Marketing STP: Market Segmentation, basic framework of segmentation, choosing target segments and positioning-Pricing strategies for Business Markets, B2B Advertising, Competitive bidding, Relationship marketing and CRM.	
UNIT – V	**09 Hours**
Business Marketing Communications and Channels- B2B Advertising, Digital marketing, - Trade shows,	

exhibitions, business meets - Managing the sales force - Deployment analysis -Business marketing channels and participants - Channel design and management decisions -B2B logistics management.

AI Integration & Authentic Intelligence: Reflection (AI² Pedagogy)

- **AI-driven Market Intelligence:**
  Use AI tools to analyze B2B market trends, customer needs, and competitor strategies while ensuring ethical data usage and unbiased insights.

- **Intelligent Customer Relationship Management:**
  Apply AI in CRM systems to predict customer behaviour, personalize interactions, and enhance long-term business relationships with transparency.

- **AI-enabled Strategic Decision Making:**
  Utilize AI-based analytics for pricing, segmentation, and channel decisions with explainable and data-driven approaches.

- **Smart Sales and Marketing Automation:**
  Implement AI in digital marketing campaigns, lead generation, and sales forecasting while maintaining privacy and fairness.

- **Responsible AI in Business Marketing:**
  Evaluate ethical, legal, and societal implications of AI in B2B marketing, ensuring trust, accountability, and sustainable practices.

Suggested Tools:

- HubSpot
- SEMrush
- Mailchimp
- Miro
- Notion

Applications:

- Market segmentation and targeting in industrial markets.
- Customer relationship management and key account handling.
- Supply chain and distribution channel optimization.
- Digital marketing and lead generation in B2B platforms.
- Pricing and competitive bidding strategies.
- Sales force automation and performance analysis.

Project Component

- Analyze a selected industry to identify potential B2B opportunities, customer

segments, and competitors.

- Design an effective distribution channel and logistics plan for a B2B product.
- Develop a LinkedIn/email-based marketing campaign for a business product or service.
- Conduct a case study on how a company makes purchasing decisions and map the decision-making process.

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
CO1	Understand the concepts, structure, and environment of B2B marketing.	L2
CO2	Analyze organizational buying behavior and decision-making models in business markets.	L4
CO3	Apply B2B marketing strategies for product, service, and portfolio management.	L3
CO4	Evaluate STP strategies, pricing, and relationship marketing approaches in B2B contexts.	L5
CO5	Design effective B2B communication, sales force, and channel management strategies.	L3

Table: Mapping Levels of COs to POs / PSOs

COS	Program Outcomes (POs)											Program Specific Outcomes (PSOs)	
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO 2
CO1	2	2	2	2	3			2	2	2	2		3
CO2	2	2	2	2	3			2	2	2	2		3
CO3	2	2	2	2	3			2	2	2	2		3
CO4	2	2	2	3	3			2	2	2	2		3
CO5	2	2	2	2	3			2	2	2	2		3

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

TEXT BOOKS:

1. Krishna K. Havaldar, Vasant M. Cavale, *Business Marketing: Text and Cases*, 4th Edition, McGraw Hill Education, 2017.

REFERENCE BOOKS:

1. Michael D. Hutt, Dheeraj Sharma, Thomas W. Speh, B2B Marketing: A South Asian Perspective Cengage 2014, 11th ed.
2. Sharad Sarin, Business Marketing: Concepts and Cases McGraw Hill 2013, 1st ed.
3. James C. Anderson, Das Narayandas, James A. Narus and D.V.R. Seshadri, Business Market Management (B2B): Understanding, Creating, and Delivering Value, Pearson 2010, 3rd ed.
4. Robert Vitale, WaldemarPfoertsch, Joseph Giglierano, Business to Business,

Marketing, Pearson 2011
5. Krishna K Havaldar, Business Marketing: Text and Cases McGraw Hill 2014, 4th ed.
6. Armstrong, Gary and Philip Kotler, "Principles of Marketing", Prentice Hall, New Delhi, 2006

E-RESOURCES:

- https://www.coursera.org/learn/b2b-marketing
- https://onlinecourses.nptel.ac.in/noc23_mg120/preview
- https://www.b2bmarketing.net/resources/
- https://www.forrester.com/b2b-marketing/content-strategy-resources/

Activity Based Learning (Suggested Activities in Class)

- Analyze market structure, customers, and competitive environment, followed by a short presentation.
- Conduct a role-play where students simulate the organizational buying process (buyer, influencer, decision-maker) to understand decision-making dynamics.
- Design a sales force structure and CRM strategy for managing key business clients.

****

MOOC Courses

Sl. No.	Course Code	Course Title
1	*CID	Introduction To Internet Of Things, IIT Kharagpur [NPTEL] 12 WEEKS Introduction To Internet Of Things | SWAYAM https://onlinecourses.nptel.ac.in/e-learning/preview/noc26_cs37

Note: CID = NPTEL Course ID

MINOR PROJECT

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

SEMESTER – VI

Subject Code	:	24CT3612	**Credits**	:	02
Hours / Week	:	04 Hours	**Total Hours**	:	26 Hours
L–T–P–S	:	0–0–0–4			

Course Learning Objectives:

This Course will enable students to:

1. To apply theoretical knowledge to solve practical problems.
2. Develop professional project management and communication skills.
3. Able to work in teams and present the project work.

DESCRIPTION:

1. Each B.Tech Project must be carried out by a group of students at the Institute. To ensure uniform participation of each student, the group size should be preferably at least 3 but not more than 4 students.
2. Each project activity must be supervised by the faculty members of the Institute. These faculty members are termed Project Guides.
3. In case the project is of multi-disciplinary nature, the Project group can be formed consisting of students from other Departments. But there must be at least one student and a project Guide from the Department who is offering the Project.
4. The topic proposed by both the guide and the student team should be approved by the department chairman and the department project coordinator to proceed further. A degree of industrial input and involvement will be encouraged and can be facilitated through existing academic-industrial collaborations or by addressing specific topics that are of interest to industrial partners.
5. All projects will be closely supervised by the Project Guide with ongoing feedback and guidance at all stages of the project from conception to completion.
6. The following criteria will be checked by the department chairman to approve for the project proposal:
    a. Department staff as Project guide
        i. Ability to provide direction to the student in the chosen field of interest to formulate a suitable title of the project.
        ii. Ability to design an appropriate strategy and methodology to carry out the Project by the team.
        iii. Ability to provide and evaluate the strong literature review document for the chosen topic.
        iv. Ability to train students on paper / technical writing skills
    b. Student Team
        i. To be dedicated and committed to work on the project by sharpening the existing and learning new technical skills.
        ii. To be committed to completing the project and participate in hackathons and project exhibitions.
7. Phase-1 comprises of Literature Survey, Problem identification, Objectives and Methodology.
8. There will be CIA evaluation (Project reviews) done by a committee of senior faculty of the Department based on the rubrics

Course Outcomes:

At the end of the course the student will be able to:

1. Demonstrate practical problem-solving skills by analyzing requirements, designing solutions, implementing algorithms, and testing software systems in real-world scenarios.

2. Develop critical thinking skills, enabling them to evaluate the feasibility, effectiveness, and efficiency of proposed solutions and make informed decisions based on evidence and reasoning.

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
CO1	3	3	3	3	3	3	3	3	3	3	3	3	3	3
CO2	3	3	3	3	3	3	3	3	3	3	3	3	3	3

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

****

Entrepreneurial Finance & Deep-Tech Start-ups

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

SEMESTER – VII

Subject Code	:	**Credits** :	02
Hours / Week	: 02 Hours	**Total Hours** :	30 Hrs L
L–T–P-J	: 2–0–0-0		

Prerequisites: - Basics of Economics, AI/ML Fundamentals, Data Science

Course Learning Objectives:

This Course will enable students to:

1. Understand fundamentals of entrepreneurial finance and start-up ecosystems.
2. Analyse 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.

UNIT – I	**05 Hours**

Foundations of Entrepreneurial Finance

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	**05 Hours**

Start-up Valuation & Funding Strategies

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	**05 Hours**

AI-Driven Business Models & Deep-Tech Ventures

AI business models (platform, API-based, subscription), deep-tech domains (AI, robotics, IoT, quantum computing), product-market fit, competitive analysis, monetization strategies

UNIT – IV	05 Hours
Financial Planning, Forecasting & Scaling Financial projections, burn rate, runway, scenario planning, scaling strategies, growth metrics, exit strategies (IPO, M&A).	
UNIT – V	**05 Hours**
Legal, Ethical & Future of AI Entrepreneurship 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: 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: Business model, funding strategy, AI integration
- Industry relevance: Venture capital ecosystem

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
CO1	Apply financial concepts such as cash flow and valuation	L3
CO2	Design AI-driven business models for deep-tech ventures	L3
CO3	Analyze startup funding strategies	L4
CO4	Develop financial projections and pitch decks	L6
CO5	Build scalable and ethical AI startups	L3

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)											Program Specific Outcomes (PSOs)	
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO 1	PSO 2
CO1	3	2	2	2	2				3	3	3	3	2
CO2	3	3	2	2	3	2			3	3	3	3	2
CO3	3	3	2	3	3	2			3	3	3	3	2
CO4	2	3	2	3	3	2			3	3	3	3	2
CO5	2	3	2	3	3	2			3	3	3	3	2

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

Suggested 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

****

LEAN START-UP METHODOLOGY

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

SEMESTER – VII

Subject Code	:	24OEXXXX	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P–S	:	3–0–0–0			

Course Learning Objectives:

This Course will enable students to:

1. **Analyze** the opportunities and identify the potential problem to build a solution
2. **Devise** the solution to solve the given problem and build core PoC and business model(0).
3. **Explain** the business model iterations based on the Field experience with customers.
4. **Get the idea** of acquiring / retaining customers as well basic finance aspects.
5. **Describe** and illustrate the current and future path forward of your (potential) start-up.

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*** means it includes not only traditional lecture method, but different *type of teaching methods* adopted to develop the course outcomes.
2. ***Interactive Teaching: Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, 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***, by following Idea to PoC to deliver to customer(s) leveraging business model(s) and iterate based on their feedback.
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. Encourage students to apply the learnings ***to the real world*** with on field experience in helping to improve their understanding.

UNIT – I	**06 Hours**
VISION: Start, Define, Learn & Experiment. ***(Text Book-1: Chapter 1 - 4)***	
UNIT – II	**06 Hours**
STEER: Leap, Test, Measure & Pivot ***(Text Book-1: Chapter 5 - 8)***	
UNIT – III	**07 Hours**
ACCELERATE: Batch, Grow, Adapt, Innovate ***(Text Book-1: Chapter 9 – 12)***	
UNIT – IV	**10 Hours**
BUSINESS MODEL PART I	

Business Model Definition, 9 Building Blocks, Business Model Canvas, Patterns
(Text Book-2: Page 14 - 125).

UNIT – V	**10 Hours**

BUSINESS MODEL PART II
Design – Customer insights, Ideation, Prototyping, Story Telling
Strategy – BM environment, Evaluating BM's, Manage
Process – Business Model Design Process
(Text Book-2: Page # 126 - 261)

Course Outcomes:
At the end of the course the student will be able to:

1. Explain the principles of the Lean Startup methodology and its application in early-stage business development.
2. Develop a Minimum Viable Product (MVP) using validated learning and measurement techniques.
3. Analyze growth engines and strategies for scaling startups sustainably.
4. Construct a Business Model Canvas incorporating all nine building blocks
5. Evaluate business models using design thinking, prototyping, and strategic assessment.

Table: Mapping Levels of COs to POs / PSOs

COs	**Program Outcomes (POs)**												**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
CO1	2	2												2
CO2	2	2	2	2	2				2	2	2	2		3
CO3	2	3	3	2	2		2		2	3	2	3		3
CO4	2	2	2	2	2				3	3	3	2		3
CO5	2	3	3	3	2		2	2	3	3	3	3		3

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

TEXT BOOKS:

1. Eric Ries "The Lean Startup", ISBN–978-0670921607.

   https://ia801206.us.archive.org/31/items/TheLeanStartupErickRies/The%20Lean%20Startup%20-%20Erick%20Ries.pdf

2. Alex Osterwalder, Yves Pigneur "Business Model Generation Handbook" ISBN-978-0-470-87641-1 http://alvarestech.com/temp/PDP2011/pdf/Business%20Model%20Generation%20(1).pdf

REFERENCE BOOKS:

1. Steve Blank "The Four Steps to the Epiphany: Successful Strategies for Products That Win", ISBN-13 : 978-0989200509

2. The Entrepreneur's Guide to Customer Development: A "Cheat Sheet" to the Four Steps to the Epiphany Paperback – Import, 6 February 2012, **ISBN-13 :** 978-0982743607

E-Resources:

1. https://ia801206.us.archive.org/31/items/TheLeanStartupErickRies/The%20Lean%20Startup%20-%20Erick%20Ries.pdf
2. http://alvarestech.com/temp/PDP2011/pdf/Business%20Model%20Generation%20(1).pdf
3. https://www.getstoryshots.com/books/the-lean-startup-summary/
4. https://www.strategyzer.com/canvas

Activity Based Learning (Suggested Activities in Class)

1. Real world problem solving with on field exposure
2. Build Start-up and test the solution with real customers.

****

PRODUCT ANALYTICS

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

SEMESTER – VII

Subject Code	:	24OEXXXX	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P–S	:	3–0–0–0			

Course Learning Objectives:

This Course will enable students to:

1. **Analyse** and understand User Behaviour .
2. Explore the skills to leverage product analytics for data driven decision making.
3. **Get the idea** of tools and technologies around Product Analytics.
4. **Describe** and illustrate the ways to improve Product performance and optimize user experience..

Teaching-Learning Process (General Instructions)

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* adopted to develop the course outcomes.
2. ***Interactive Teaching: Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Encourage ***Collaborative*** (Group Learning) Learning in the class.
4. Adopt ***experiential Learning*** by following hands-on .
5. Discuss how every ***concept can be applied to the real world*** – and when that's possible, it helps improve the students' understanding.

UNIT – I	**08 Hours**

Introduction to Product Analytics

- Understanding the significance of product analytics in modern business.
- Key concepts: metrics, KPIs, and data-driven decision-making.
- Role of Product Analysts and their responsibilities.

Resources:

Article: "A Beginner's Guide to Product Analytics" by Mixpanel
Video: "Introduction to Product Analytics" by Heap

UNIT – II	**08 Hours**

Data Collection and Tools

- Exploring data sources and collection methods.
- Introduction to Google Analytics for tracking user behaviour.
- Ensuring data accuracy, privacy, and compliance.

Resources:

- Google Analytics Academy (Free online courses)
- Article: "Getting Started with Google Analytics" on Analytics Help

UNIT – III	**07 Hours**

Defining Metrics and KPIs

Defining meaningful metrics and KPIs.
Building a measurement framework for effective analysis.
Practical exercise: Identifying KPIs for a social media platform.

Resources:

Article: "Choosing the Right Product Metrics" by Amplitude

UNIT – IV	**08 Hours**

User Behaviour Analysis

- Mapping user journeys and funnel analysis.
- Cohort analysis to measure user retention.
- Hands-on: Analyzing user behavior using cohort analysis.

Resources:

Article: "Cohort Analysis in Google Analytics" by Loves Data

Article: "User Funnel Analysis: A Step-by-Step Guide" by Mixpanel

UNIT – V	08 Hours

A/B Testing and Experimentation, Customer Segmentation, Personalization, Performance

- Understanding A/B testing principles.
- Practical guide to A/B testing with Google Optimize.
- Hands-on: Designing and running an A/B test.

Resources:
Google Optimize (Free tool for A/B testing)
Article: "A/B Testing: A Comprehensive Guide" by CXL

- Segmentation strategies for targeted analysis.
- Leveraging data for personalized user experiences.
- Case studies of successful personalization.

Resources:
Article: "The Power of Personalization in Product Development" by SmarterHQ.
Case Study: "How Netflix Uses Segmentation for Personalization" by Kissmetrics.

- Measuring product success and health.
- Churn analysis and strategies for reducing churn.
- Sentiment analysis of user feedback.

Resources:
Tool: Amplitude for product health measurement.
Article: "Mastering Customer Churn Analysis" by Mixpanel.

Course Outcomes:

At the end of the course the student will be able to:

1. **Describe** what is product analytics and why needed.
2. **Explain** analyzing user behavior.
3. **Describe** the role of product analytics in data driven decision making.
4. **Identify and explain** the business case with example reflecting product performance improvement and optimize user experience

Table: Mapping Levels of Cos to Pos / PSOs

Cos	Program Outcomes (Pos)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
CO1	3	3										2		3
CO2	3	3	2									2		3
CO3	3	3										1		3
CO4	3	3	2									2		3

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

PROJECT / COURSE DELIVERABLES

1. Presentation
   a. The PoC running
   b. Parameters tracked with alternatives
2. PoC / Prototype, Target Specifications
   a. Using Any tool with few parameters
3. Business Case for Product Analytics

Activity Based Learning (Suggested Activities in Class)

1. Real world problem solving as a team building soft skills as well as getting exposure to product analytics skills as appropriate.
2. Demonstration of PoC

****

Generative AI with Diffusion Models

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

SEMESTER – VII

Subject Code	:		**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	45 Hrs (45 Hrs L)
L–T–P-J	:	3–0–0-0			

Prerequisites: Basic Programming, Mathematics

Course Learning Objectives:

This Course will enable students to:

2. Understand the principles of generative AI, including text, image, and multimodal models.
3. Apply prompt engineering and fine-tuning techniques for domain-specific text generation tasks.
4. Analyze diffusion and GAN-based models for image synthesis and enhancement.
5. Design and deploy generative AI systems using APIs and scalable architectures.
6. Evaluate ethical, bias, and safety issues in real-world AI deployments.

Broad Course Description

This course provides an in-depth understanding of Generative AI, Diffusion Models, and Vision-Language Models, focusing on both theoretical foundations and practical implementation. Students will explore advanced techniques in text generation, image synthesis, and multimodal learning, along with deployment strategies for real-world applications. The course emphasizes hands-on development, ethical considerations, and scalable AI system design, enabling learners to build intelligent systems capable of generating, interpreting, and integrating multimodal data across domains.

UNIT – I	**09 Hours**

ADVANCED TEXT GENERATION AND CUSTOMIZATION

Fine-tuning and customizing large language models for specific domains, Prompt engineering and chaining prompts for complex tasks, Case studies: automated report generation, creative writing, summarization. (Text Book-1)

UNIT – II	**08 Hours**

IMAGE & MULTIMEDIA SYNTHESIS

Using GANs and diffusion models for image creation and editing, Tools for rapid image prototyping and style transfer, Multimedia content generation: audio, video, and cross-modal use cases. (Text Book1)

UNIT – III	**08 Hours**

DEPLOYMENT AND INTEGRATION

Incorporating generative AI into web and mobile applications (e.g., APIs, plugins),Ethics, safety, and bias management in deployment, Scalability and performance considerations. Text Book1)

UNIT – IV	**10 Hours**

DIFFUSION MODELS

Introduction, Denoising Diffusion Probabilistic Models (DDPMs), Classifier and Classifier-Free Diffusion Guidance, Text-conditioned Diffusion Models: Sampling, Prediction Space, Noise Schedules, Architectures. (Text Book1)

UNIT – V	**10 Hours**

VISION-LANGUAGE MODELS

Self-Supervised Learning: Introduction, Simple Framework for Contrastive Learning of Visual Representation (SimCLR), Contrastive Learning, Vision-Language Models, Contrastive Language-Image Pre-training (CLIP). (Text Book1)

- Generative AI Explained
- Introduction to Deploying RAG Pipelines for Production at Scale
- Deploying and Optimizing AI Inference at Scale
- Deploying RAG Pipelines for Production at Scale

AI Integration & Authentic Intelligence: Reflection (AI² Pedagogy)

- This course integrates AI as both a **learning tool and subject of study**. Students will:
- Use AI tools for **code generation, prototyping, and experimentation**
- Critically reflect on **AI-generated outputs vs human intelligence**
- Analyze **bias, hallucination, and ethical risks**
- Develop **authentic solutions**, ensuring:
- Original problem formulation
- Human-in-the-loop validation
- Transparent AI usage documentation
- How AI assisted vs how I improved the solution
- Limitations of AI outputs in my project
- Ethical considerations in my design

AI Tools:

Programming: Python

Libraries: PyTorch, TensorFlow, Hugging Face Transformers

Generative AI Tools: ChatGPT, DALL·E, Stable Diffusion

Vision Tools: OpenCV, CLIP models

Deployment: FastAPI, Flask, Streamlit

Applications:

- Automated report and content generation
- AI-powered chatbots and assistants
- Image generation and editing systems
- Medical image synthesis and analysis
- Cross-modal search engines (text ↔ image)
- Video/audio content generation
- Creative AI (art, storytelling, design)

Mini Project Component

- Some sample projects described below simulate the real software engineering problems.

1.Domain-Specific AI Report Generator

- Fine-tuned LLM for generating technical/research reports

2. Text-to-Image Generator using Diffusion Models

- Prompt-based image synthesis with style control

3. AI-Powered Multimodal Search Engine

- Retrieve images using text queries (CLIP-based)

4. Smart Content Creation Assistant

- Generates blogs, summaries, and creative content

5. Ethical AI Analyzer Tool

- Detects bias and unsafe outputs in generative AI systems

Course Outcome	**Description**	**Bloom's Taxonomy Level**
At the end of the course the student will be able to:		
CO1	Explain concepts of generative AI, GANs, diffusion models, and vision-language models	L3
CO2	Apply prompt engineering and fine-tuning techniques for text generation	L3
CO3	Analyze and implement image synthesis models using GANs and diffusion	L4
CO4	Design and deploy AI-powered applications integrating multimodal models	L6
CO5	Evaluate ethical, bias, and scalability challenges in AI systems	L4

CO S	PO 1	PO 2	PO3	PO4	PO5	PO 6	PO 7	PO 8	PO9	PO 10	PO 11	PSO 1	PSO 2
CO1	3	2	1	-	-	-	-	-	-	-	-	3	2
CO2	2	3	2	1	2	-	-	-	1	-	-	3	2
CO3	2	3	3	2	2	-	-	-	1	-	-	3	2
CO4	2	2	3	3	3	1	-	-	2	2	2	3	2
CO5	2	3	2	2	1	2	1	3	2	2	1	3	2

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

Textbooks:

1. Hands-On Generative AI with Transformers and Diffusion Models, Omar Sanseviero et al., O'Reilly Media, 1st Edition, 2024
2. Applied Generative AI for Beginners, Akshay Kulkarni et al., Apress, 1st Edition, 2023.

REFERENCE BOOKS:

1. Image Generation Models, Vladimir Bok, Manning Publications, 1st Edition, 2025
2. Master Diffusion Models: A Practical Guide to Generative AI, Ashley, Independently Published, 1st Edition, 2025.

E resources:

1. Coursera: Best Generative AI Courses & Certificates
2. Udemy: Top Generative AI (GenAI) Courses Online

Activity Based Learning (Suggested Activities in Class)

- Prompt Engineering and Text Generation Lab
- Text-to-Image Generation using Diffusion Models
- Generative AI Application Deployment Project
- Diffusion Model Visualization and Analysis
- Vision-Language Model (CLIP) Implementation

****

Blockchain Technology

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

SEMESTER – VII

Subject Code	:		**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	45 (30Hrs (L) + 15 hRs(J))
L–T–P-J	:	2–0–0-2			

Course Learning Objectives:

This Course will enable students to:

1. Understand blockchain architecture and consensus mechanisms.
2. Write and deploy smart contracts for secure applications.
3. Integrate IoT devices and data into blockchain systems.
4. Analyze blockchain performance, governance, and ethics.
5. Explore AI synergy and green blockchain designs.

Broad Course Description

This course provides a comprehensive understanding of blockchain architecture, distributed ledgers, and consensus mechanisms. It focuses on smart contract design, decentralized applications, and blockchain integration with IoT and AI systems, highlighting transparency, security, and ethical aspects crucial for next-generation digital trust platforms.

UNIT – I	**06 Hours**

BLOCKCHAIN PRINCIPLES:

Distributed ledger technologies. Blockchain structure, transactions, blocks, hashes. Cryptographic primitives. Peer-to-peer communication and consensus mechanisms. Types of blockchain (public, private, consortium).

(Text Book 1 : Ch. 1 1.1–1.4)

UNIT – II	**06 Hours**

CONSENSUS ALGORITHMS:

PoW, PoS, DPoS, PBFT, Raft. Mining concept and difficulty adjustment. Energy consumption vs efficiency metrics. Smart contract execution model.

(Text Book 1 : Ch. 3 3.1–3.5; Text Book 2 : Ch. 2 2.3–2.5)

UNIT – III	**06 Hours**

SMART CONTRACTS AND DApps :

Ethereum architecture and EVM. Solidity language basics. Smart contract debugging and deployment (Truffle/Ganache). Token creation (ERC-20 / NFT).

Text Book 1 : Ch. 5 5.1–5.4)

UNIT – IV	**06 Hours**

BLOCKCHAIN INTEGRATION:

Blockchain in Healthcare, Supply Chain, Finance. Interoperability and sidechains (Polkadot, Cosmos). Identity and Access Management on blockchain. AI–Blockchain convergence (federated learning audit trail).

(Text Book 2 : Ch. 4 4.1–4.4)

UNIT – V	**06 Hours**

PERFORMANCE, ETHICS & EMERGING TRENDS:

Scalability issues and Layer-2 solutions (Lightning, Plasma). Quantum-resistant blockchains. NFTs, DeFi, and Web3. Sustainability and energy-efficient mining.

(Text Book 1 : Ch. 7 7.1–7.3; Text Book 2 : Ch. 5 5.1–5.2)

AI Integration & Authentic Intelligence: Reflection (AI² Pedagogy)

1. AI driven fraud detection on blockchains.
2. Reinforcement learning for consensus optimization.

Applications:

1. Create and deploy a basic smart contract using Remix/Metamask.
2. Build a simple blockchain prototype in Python to log transactions.
3. Trace and visualize block hash links to explain immutability.
4. Conduct a group discussion on public vs private blockchains.
5. Demonstrate a demo use case of IoT Blockchain integration.

Mini Project Component

The following project-based activities can be conducted in the course:

1. Journal: Draw a transaction lifecycle diagram.
2. Project: Implement PoW simulation in Python.
3. Activity: Case study – IBM Food Trust or Hyperledger Fabric.
4. Capstone: Prototype green blockchain ledger with energy usage estimation.
5. Project: Design a consent management smart contract for IoT data.

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
CO1	Explain the fundamental concepts, architecture, and working of blockchain systems along with cryptographic foundations.	L2
CO2	Analyze and apply various consensus mechanisms such as Proof of Work, Proof of Stake, and PBFT for secure distributed agreement.	L3
CO3	Design and implement smart contracts and decentralized applications (DApps) using suitable development frameworks.	L3
CO4	Integrate blockchain with IoT and AI systems for secure data management and provenance tracking.	L4
CO5	Evaluate blockchain performance, scalability, and ethical implications to propose sustainable, real-world solutions.	L4

Table: Mapping Levels of COs to POs / PSOs

COS	Program Outcomes (POs)											Program Specific Outcomes (PSOs)	
	PO1	PO2	PO3	PO4	PO5	PO 6	PO7	PO8	PO 9	PO1 0	PO1 1	PSO 1	PSO 2
CO1	3	2			2						2	2	
CO2	2	3	2	2	2						2	3	
CO3	2	2	3	2	3						2	3	
CO4	2	2	3	2	3			2			2	3	
CO5	2	3	3	2	2		2	2			2	3	

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

TEXT BOOKS:

1. Imran Bashir – Mastering Blockchain, 3e, Packt.
2. Arvind Narasimha – Blockchain for Enterprise Applications, CRC Press.

NVIDIA Course Mapping (Industry Alignment)

NVIDIA DLI Course / Content	Mapped Units	Relevance
Accelerated Computing for Blockchain Analytics	II–V	GPU-enhanced validation
AI for Fraud Detection	IV–V	Smart contract monitoring

E-RESOURCES:

1. ptel.ac.in
2. coursera.org
3. edx.org
4. ethereum.org
5. developer.ibm.com

Activity-Based Learning (Suggested Activities in Class)

1. Create and deploy a basic smart contract using Remix/Metamask.
2. Build a simple blockchain prototype in Python to log transactions.
3. Trace and visualize block hash links to explain immutability.
4. Conduct a group discussion on public vs private blockchains.
5. Demonstrate a demo use-case of IoT-Blockchain integration.

****

AR/VR & Game-Theoretic

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

SEMESTER – VII

Subject Code	:		**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	30Hrs L+15Hrs J
L–T–P-J	:	2–0–0-2			

Course Learning Objectives:

This Course will enable students to:

1. Understand principles and hardware of AR and VR systems.
2. Develop AR/VR applications using Unity XR, ARCore, ARKit, and WebXR.
3. Introduce mathematical foundations of Game Theory and Nash Equilibrium.
4. Apply Algorithmic Game Theory to mechanism design, auctions, and multi-agent systems.

Broad Course Description

A dual-focus course covering immersive XR development (Unity XR Toolkit, ARCore/ARKit, WebXR, spatial audio, hand tracking, Mixed Reality) and Algorithmic Game Theory (Nash Equilibrium, dominant strategies, mechanism design, auction theory, Price of Anarchy, multi-agent coordination). Students build VR environments, AR mobile apps, and game-theoretic simulations.

UNIT – I	**06 Hours**

Introduction to AR/VR Technologies: History and evolution of AR/VR, VR hardware – HMDs, display technologies, tracking, VR perception – stereoscopic vision, FOV, refresh rates, AR hardware – smartphones, smart glasses, MR and Reality-Virtuality Continuum, Presence, cybersickness, XR design principles.

• Jason Jerald, "The VR Book: Human-Centered Design for VR" Ch. 1–4 (ACM Books, 2016)
• Unity XR Documentation (docs.unity3d.com/Manual/XR.html)

UNIT – II	**06 Hours**

VR Development with Unity XR: Unity XR Toolkit – XR Rig, controllers, input actions, Room-scale vs. stationary VR, Locomotion – teleportation, smooth locomotion, Interaction systems, Spatial audio, Hand tracking, VR optimization, Deploying to Meta Quest.

• Unity XR Interaction Toolkit Documentation (docs.unity3d.com/Packages/com.unity.xr.interaction.toolkit)
• Meta Quest Developer Hub Documentation (developer.oculus.com)

UNIT – III	**06 Hours**

AR Development with ARCore/ARKit & WebXR: ARCore/ARKit fundamentals – plane detection, image tracking, face tracking, Unity AR Foundation, Markerless AR, Marker-based AR with Vuforia, AR UI/UX, WebXR with A-Frame/Three.js, Location-based AR, Deployment.

• AR Foundation Documentation (docs.unity3d.com/Packages/com.unity.xr.arfoundation)
• A-Frame Documentation (aframe.io/docs)

UNIT – IV	**06 Hours**

Foundations of Game Theory: Strategic interactions, Normal/extensive form games, Dominant

strategies, Nash Equilibrium – pure and mixed strategies, Zero-sum games and minimax, Cooperative game theory – coalitions, Shapley value, Sequential games, Evolutionary game theory.

• Martin Osborne & Ariel Rubinstein, "A Course in Game Theory" Ch. 1–6 (MIT Press, 1994 – free at osborne.economics.utoronto.ca)
• Nisan et al., "Algorithmic Game Theory" Ch. 1–3 (Cambridge University Press, 2007)

UNIT – V	06 Hours

Algorithmic Game Theory & Applications: Mechanism design – revelation principle, incentive compatibility, Social choice theory, Arrow's theorem, Auction theory – Vickrey auction, combinatorial auctions, Price of Anarchy, Algorithmic mechanism design, AdWords auctions, P2P networks, Blockchain consensus, Multi-agent AI coordination.

• Nisan et al., "Algorithmic Game Theory" Ch. 4–9
• Tim Roughgarden, "Twenty Lectures on Algorithmic Game Theory" Lectures 1–8 (Cambridge, 2016)

NVIDIA Deep Learning Institute (DLI) Course Mapping

The following NVIDIA DLI courses are recommended to supplement the subject. Free courses are accessible at learn.nvidia.com. Paid instructor-led workshops (ILT) are available via NVIDIA partner institutions.

UNIT	NVIDIA DLI Course Name	Type	Duration	URL / Code
Units 1–2	How to Build a Native OpenUSD XR Application Self-paced: build XR applications using OpenUSD and NVIDIA Omniverse – directly maps to Unit 1–2 XR development principles.	Self-Paced	2 Hrs	FREE DLI+S-OV-17+V1
Unit 2	Build Beautiful, Custom UI for 3D Tools on NVIDIA Omniverse 3D tool UI design –	Self-Paced	2 Hrs	FREE
	directly applicable to VR/XR UI/UX design in Unit 2.			DLI+S-OV-12+V1
Unit 2	Building a 3D Product Configurator With OpenUSD and Omniverse Interactive 3D product visualization – mirrors VR environment interaction and object placement covered in Unit 2.	Self-Paced	2 Hrs	FREE DLI+S-OV-16+V1
Unit 3	Getting Started With Image Segmentation Image segmentation using deep learning – applicable to AR plane detection, object recognition, and occlusion in Unit 3.	Self-Paced	2 Hrs	FREE DLI+C-FX-04+V1
Unit 3	Disaster Risk Monitoring Using Satellite Imagery Real-world computer vision for location-based AR – extends Unit 3 location-based AR with spatial data processing.	Self-Paced	2 Hrs	FREE DLI-SP-DRMS
Units 3–	Building Real-Time Video AI Applications Building real-time video AI pipelines – relevant to AR camera-based	Self-Pac	2 Hrs	FREE DLI-SP-

5	recognition and object tracking in Unit 3.	ed		BRV AA
Units 4–5	Introduction to Graph Neural Networks GNNs for relational data – directly relevant to network game theory, graph-based mechanism design, and multi-agent systems.	Self-Paced	2 Hrs	FREE DLI-SP-IGNN
Units 4–5	Exploring Adversarial Machine Learning Adversarial ML – maps directly to game-theoretic concepts (adversarial players, minimax strategies) in Units 4–5.	Self-Paced	2 Hrs	FREE DLI-SP-EAM

AI Integration & Authentic Intelligence: Reflection (AI² Pedagogy)

- AI in XR Environments: Integrate AI NPCs with natural language in VR using LLM APIs.
- Game-Theoretic AI Analysis: Analyze AlphaGo/AlphaZero strategic reasoning.
- Procedural AR Content: Use AI image generation to create AR target images.
- Reflective Mechanism Design: Design incentive mechanism for AI multi-agent system.
- Ethics of Immersive AI: Surveillance, manipulation, and psychological effects of AI-enhanced XR.

Applications

- Medical Training and Surgical Simulation in VR
- Architectural and Real-Estate VR Walkthroughs
- AR-Based Retail Try-On
- Internet Advertising and Programmatic Auctions
- Blockchain Protocol Design
- Multi-Agent AI Systems

Laboratory Component (1 Credit)

The lab emphasizes hands-on implementation, visualization, and AI-assisted debugging.

Lab Experiments

- Set up Unity XR Toolkit; build VR scene with grabbable objects and teleportation.
- Implement hand tracking in VR for natural interaction.
- Build spatial audio-enhanced VR environment.
- Create marker-based AR app overlaying 3D models on image targets.
- Develop markerless AR app for placing virtual furniture in real room.
- Build WebXR experience in browser using A-Frame.
- Implement and solve Nash Equilibrium for 3x3 payoff matrix.
- Simulate Vickrey auction and verify truthful bidding.
- Model network routing game and compute Price of Anarchy.
- Develop multi-agent simulation using game-theoretic coordination strategies.

Mini Project Component

Industry-driven projects that simulate real software engineering problems.

Project 1 – Immersive VR Museum Experience

Problem: VR museum where users walk around, pick up artifacts, and hear spatial audio narration.

Concepts used: Unity XR, Interaction Toolkit, Spatial Audio

Industry relevance: Cultural heritage, tourism, and educational VR.

Project 2 – AR Product Visualization App

Problem: AR mobile app for placing and visualizing 3D products in real environment.

Concepts used: AR Foundation, ARCore/ARKit, 3D Modeling, UI/UX

Industry relevance: E-commerce, retail, and interior design.

Project 3 – Mechanism Design Simulation

Problem: Design and simulate a spectrum auction using algorithmic mechanism design.

Concepts used: Game theory, Auctions, Algorithm design, Agent modeling

Industry relevance: Telecom auctions, digital advertising, policy design.

Course Outcomes

CO	Description	Bloom's
At the end of the course the student will be able to:		
CO1	Explain hardware, perceptual principles, and design guidelines for AR and VR immersive experiences.	L2
CO2	Build interactive VR applications using Unity XR Toolkit with locomotion, interaction, and spatial audio.	L3
CO3	Develop AR applications for mobile platforms using AR Foundation, ARCore/ARKit, and WebXR.	L3
CO4	Analyze strategic interactions using Nash Equilibrium, dominant strategies, and cooperative game theory.	L4
CO5	Apply algorithmic mechanism design principles to design and evaluate auction systems and multi-agent coordination.	L5

Table: Mapping Levels of COs to POs / PSOs

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PSO 1	PSO 2
CO1	2	2	2	2	3						2	2	2
CO2	3	3	3	3	3						2	3	3
CO3	3	3	3	3	3						2	3	3
CO4	3	3	2	3	2						2	3	2
CO5	3	3	3	3	3						3	3	3
3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)													

Suggested Tools

- Unity 2022 LTS with XR Toolkit
- AR Foundation (ARCore/ARKit)
- Vuforia
- A-Frame / Three.js
- Meta Quest / HTC Vive
- Python / Jupyter for Game Theory

TEXT BOOKS:

1. Jason Jerald, "The VR Book", ACM Books, 2016.
2. Nisan et al., "Algorithmic Game Theory", Cambridge University Press, 2007.
3. Martin Osborne & Ariel Rubinstein, "A Course in Game Theory", MIT Press, 1994.

REFERENCE BOOKS:

1. Tim Roughgarden, "Twenty Lectures on Algorithmic Game Theory", Cambridge, 2016.
2. Unity XR Official Documentation (docs.unity3d.com).

E-RESOURCES:

- Unity Learn – XR Development (learn.unity.com)
- Stanford CS364A – Algorithmic Game Theory (YouTube)
- WebXR Samples (immersive-web.github.io)
- NVIDIA DLI: learn.nvidia.com

Activity Based Learning (Suggested Activities in Class)

- VR Demo Day – experience and critique commercial VR applications.
- Game theory debates – simulate multi-player strategic games.
- Interdisciplinary workshops with economics and design students.

****

Quantum Artificial Intelligence

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

SEMESTER – VII

Subject Code	:		**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	45 Hrs (30hrs T + 15hrs J)
L–T–P-J	:	2–0–0-2			

Prerequisites: Linear Algebra, Probability, Machine Learning

Course Learning Objectives:

This Course will enable students to:

- Understand the principles of Quantum Artificial Intelligence and intelligent computational models.
- Analyze information theory, reversible computation, and probabilistic reasoning for intelligent systems.
- Apply quantum-inspired techniques for problem solving and intelligent decision systems.
- Design and simulate Quantum AI models using modern quantum frameworks.
- Explore ethical, explainable, and emerging trends in Quantum Artificial Intelligence.

Broad Course Description

This course introduces the concepts, principles, and applications of Quantum Artificial Intelligence (QAI) by integrating artificial intelligence paradigms with quantum computational models. The course focuses on quantum-enhanced intelligent systems, probabilistic reasoning, quantum cognition, reversible computation, quantum problem solving, and hybrid quantum–classical AI systems. Students will explore emerging Quantum AI approaches through simulation-driven learning and project-based activities using modern quantum platforms.

UNIT – I	**08 Hours**

Computation and Problem Solving Foundations:

Introduction to Quantum Artificial Intelligence, Artificial Intelligence Foundations, Computational Models, Complexity Theory, Church–Turing Thesis, Classical and Quantum Computation, Knowledge Representation, Production Systems, Deduction Systems, Sub-symbolic Models of Problem Solving.

(Text Book-1: Chapter 1, 2)

UNIT – II	**10 Hours**

Information, Probability, and Reversible Computation:

Information Theory and Thermodynamics, Entropy and Information Measures, Information and Memory, Associative Memory, Sparse Coding, Reversible Computation, Reversible Circuits, Toffoli Gate, Kolmogorov Probabilities, Conditional Probability, Bayes Rule, Bayesian Networks, Markov Chains.

(Text Book-1: Chapter 3, Chapter 4)

UNIT – III	**09 Hours**

Qubit-Based Computation: Unitary Evolution, Hilbert Space, Quantum Time Evolution, Compound Systems, Von Neumann Entropy, Quantum Measurement, Quantum Randomness, Quantum Circuits, Quantum Fourier Transform, Quantum Parallelism. (Text Book 1: Chapter 5, Chapter 6)
UNIT – IV | **09 Hours**
Quantum Search and Quantum Problem Solving: Overview of Grover's Search Algorithm, Quantum Search Space Representation, Periodicity and Quantum Fourier Transform, Quantum Problem-Solving Models, Quantum Production Systems, Quantum Search in AI, Quantum Decision Models, Quantum Associative Memory. (Text Book1: Chapter 7, Chapter 8)
UNIT – V | **09 Hours**
Quantum Cognition and Emerging Quantum AI Approaches: Quantum Cognition, Quantum Decision Theory, Quantum-Inspired Learning Systems, Quantum-Like Bayesian Networks, Hybrid Quantum–Classical AI Models, Explainable and Responsible Quantum AI, Emerging Trends in Quantum Artificial Intelligence. (Text Book1: Chapter 9, Chapter 10)

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

1. **AI–Quantum Convergence:** Integrate classical AI techniques with quantum computational paradigms to develop advanced intelligent systems.
2. **Quantum Data Representation & Learning:** Utilize quantum-inspired information representation for intelligent processing and reasoning.
3. **Quantum Circuit-Based Intelligence:** Apply quantum computational models for intelligent learning and optimization.
4. **Simulation-Driven AI Development:** Use Qiskit and quantum simulators for experimentation and validation.
5. **Hybrid Quantum–Classical AI Systems:** Combine classical optimization algorithms with quantum computational approaches.
6. **Responsible & Explainable Quantum AI:** Promote transparency, explainability, and ethical practices in Quantum AI systems.

Suggested Tools:

- Tensorflow Quantum
- PennyLane
- Google Colab
- IBM Quantum Expereince
- Qiskit Machine Learning

Applications:

- Intelligent decision support systems

- Quantum-inspired optimization systems
- Healthcare analytics and diagnosis
- Cybersecurity and secure communication
- Financial intelligence and forecasting
- Smart robotics and autonomous systems
- Quantum-enhanced recommendation systems
- Quantum Natural Language Processing

Mini Project Component

Industry-Driven Projects (IBM / Google Quantum AI Style)

These projects simulate real-world **quantum machine learning and intelligent system applications**.

Project 1 – Quantum Decision Intelligence System

6. **Problem:** Develop a quantum-inspired intelligent decision-making systems
7. **Concepts used:** Quantum probability, Bayesian reasoning, optimization
8. **Industry relevance:** Smart analytics, finance, healthcare

Project 2 – Hybrid Quantum–Classical AI Framework

1. **Problem:** Build a hybrid intelligent model integrating classical AI with quantum computation
2. **Concepts used:** Hybrid systems, optimization, Qiskit
3. **Industry relevance:** Predictive analytics, automation

Project 3 – Quantum Search-Based Problem Solver

1. **Problem:** Implement a quantum-inspired search framework for intelligent problem solving.
2. **Concepts used:** Grover's search, search space representation
3. **Industry relevance**: Robotics, recommendation systems, AI search applications

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Explain the foundations of Quantum Artificial Intelligence and intelligent computational models.	L2
2	Analyze information theory, reversible computation, and probabilistic reasoning in intelligent systems.	L4
3	Apply quantum-inspired approaches for intelligent problem solving and decision making.	L3
4	Design and simulate Quantum AI systems using quantum computing frameworks.	L5
5	Analyze emerging applications, ethical aspects, and future directions of Quantum Artificial Intelligence.	L4

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	1							2	
CO2	3	3	2	2	3							3	2
CO3	2	3	2	3	2							3	2
CO4	2	3	2	3	2							3	2
CO5	3	3	3	3	3						2	3	3

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

TEXT BOOKS:

1. Andreas Wichert, "**Principles of Quantum Artificial Intelligence**", World Scientific Publishing.

REFERENCE BOOKS:

1. Chris Bernhardt, "Quantum Computing for Everyone", MIT Press.
2. Maria Schuld and Francesco Petruccione, "Supervised Learning with Quantum Computers", Springer.
3. Michael Nielsen and Isaac Chuang, "Quantum Computation and Quantum Information", Cambridge University Press.

E-Resources:

- https://qiskit.org
- https://pennylane.ai
- https://quantum.ibm.com
- https://nptel.ac.in

Activity Based Learning (Suggested Activities in Class)

Activity 1: Quantum Circuit simulation
Activity 2: Analysis of real-world Quantum AI applications in healthcare, finance, cybersecurity, and robotics.
Activity 3: Activities related to entropy, probabilistic reasoning, Bayesian inference, and information measures.

****

Sustainable Innovation Business

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

SEMESTER – VII

Subject Code	:		**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	45 Hrs L
L–T–P-J	:	3–0–0-0			

Prerequisites: basic understanding of management principles, business environment, and organizational behaviour

Course Learning Objectives:

This Course will enable students to:

1. Develop an understanding of fundamental concepts, principles, and processes of innovation.
2. Enable students to analyze the role of innovation in organizational strategy and competitiveness.
3. Familiarize learners with different types and approaches to innovation, including disruptive and open innovation.
4. Provide insights into innovation ecosystems, including policy, funding, and barriers.
5. Equip students with frameworks and tools to manage innovation in real-world contexts, especially emerging markets.

Broad Course Description

This course introduces students to the theory and practice of innovation management in contemporary organizations. It covers the evolution of innovation, its strategic importance, and the processes through which innovation is generated and managed. The course explores different types of innovation, including incremental, disruptive, and design-based innovation, along with modern approaches such as open innovation, reverse innovation, and Jugaad innovation. It also examines the role of government policies, venture capital, and organizational structures in fostering innovation. Through frameworks, models, and real-world examples, the course prepares students to understand, analyze, and apply innovation strategies for sustainable competitive advantage and value creation.

UNIT – I	**09 Hours**

Foundations of Innovation:

Historical Perspectives of Innovation, Innovation Myths and Realities, Challenges, Triggers, and Sources of Creativity & Innovation, Innovation by Individuals, Communities, and Corporations, Innovator Profiles.

UNIT – II	**09 Hours**

Innovation Process and Strategy:

Innovation Cycle and Phases, Structured vs. Unstructured Innovation, Link between Corporate Vision, Strategy, and Innovation, Components of Strategic Innovation, Organizational Architecture for Strategic Innovation.

UNIT – III	**09 Hours**

Innovation Ecosystem:

Role of Government Policy in Innovation, Role of Venture Capitalists and Business Angels, Barriers to Innovation (Eight Barriers), Principles for Breaking Innovation Barriers, Innovation for Sustainable Competitive Advantage, Wealth and Value Generation

UNIT – IV	**09 Hours**

Innovation Types and Approaches:

Incremental, Breakthrough, and Disruptive Innovation, Design-based Innovation, Open vs Closed

Innovation, Kotler's Four Levels of Innovation.	
UNIT – V	**09 Hours**
Innovation Frameworks and Emerging Models: Hansen & Birkinshaw, Treacy & Wiersema, Sawhney & Wolcott, Reverse Innovation: Principles and Practices, Management Transformation, Jugaad Innovation: Principles and Practices, Emerging Market Relevance	

AI Integration & Authentic Intelligence: Reflection (AI² Pedagogy)

- **AI-Enabled Innovation Ideation & Validation:**
  Leverage AI tools to generate, refine, and validate innovative ideas using market insights, user needs, and emerging trends while ensuring originality and contextual relevance.
- **AI-Assisted Strategic Decision Making:**
  Apply AI to analyze data, evaluate innovation opportunities, and support strategic decisions related to product development, market entry, and competitive positioning.
- **Responsible & Ethical Use of AI in Innovation:**
  Ensure ethical use of AI by validating outputs, avoiding over-reliance, maintaining transparency, and critically reflecting on biases, limitations, and the role of human judgment in innovation processes.

Tools:

- Miro
- Notion
- Figma

Applications:

- Product and service innovation
- Business model innovation
- Process optimization and automation
- Customer experience enhancement
- Predictive innovation strategies
- Sustainable and frugal innovation

Project Component

- Develop a disruptive business model for a traditional industry using AI.
- Design a low-cost (Jugaad) innovation for rural or underserved markets.
- Develop a smart workflow system using automation tools.
- Analyze successful reverse innovation cases and propose improvements.

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Explain fundamental concepts, principles, and processes of innovation.	L2
2	Analyze the relationship between innovation, corporate strategy, and organizational performance.	L4
3	Evaluate different types and approaches of innovation in various contexts.	L5
4	Assess the role of innovation ecosystems, including policy,	L4

	funding, and barriers.	
5	Apply innovation frameworks and models to solve real-world business problems.	L3

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	3				2	2	2		2
CO2	3	3	2	2	3				2	2	2		2
CO3	2	3	3	2	3				2	2	2		2
CO4	2	2	3	3	3				2	2	2		2
CO5	3	3	2	2	3				2	2	2		2

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

TEXT BOOKS:

1. Paul Trott, *Innovation Management and New Product Development*, Pearson Education, 6th Edition, 2017

REFERENCE BOOKS:

2. Joe Tidd and John Bessant, Managing Innovation: Integrating Technological, Market and Organizational Change, Wiley, 5th Edition, 2018. ISBN – 978-1118360637.

3. Clayton M. Christensen, The Innovator's Dilemma: When New Technologies Cause Great Firms to Fail, Harvard Business Review Press, 2016. ISBN – 978-1633691780.

4. Vijay Govindarajan and Chris Trimble, Reverse Innovation: Create Far From Home, Win Everywhere, Harvard Business Review Press, 2012. ISBN – 978-1422157640.

E-RESOURCES:

- https://www.strategyzer.com
- https://www.ideou.com/blogs/inspiration/what-is-design-thinking
- https://hbr.org/topic/innovation
- https://online.hbs.edu/blog/post/types-of-innovation
- https://www.nesta.org.uk/toolkit/innovation-methods/
- https://www.wipo.int/about-ip/en/

Activity Based Learning (Suggested Activities in Class)

1. Identify real-world problems and propose innovative solutions with field exposure.
2. Develop and prototype an innovative product/service and test with potential users.
3. Analyze real-world case studies of successful and failed innovations.

****

MOOC Courses

Sl. No.	Course Code	Course Title
1	*CID	Blockchain and its Applications, IIT Kharagpur [NPTEL] 12 WEEKS Blockchain and its Applications | SWAYAM https://onlinecourses.nptel.ac.in/e-learning/preview/noc26_cs34

Note: CID = NPTEL Course ID

Agentic AI and RAG Engineering

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

SEMESTER – VII

Subject Code	:		**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	45 Hrs L
L–T–P-J	:	3–0–0-0			

Prerequisites: Machine Learning, Natural Language Processing.

Course Learning Objectives:

This Course will enable students to:

1. Understand the foundations of Generative AI and LLMs including transformer architecture, prompt engineering techniques, and evaluation metrics for building intelligent systems.
2. Develop Retrieval-Augmented Generation (RAG) pipelines by integrating embeddings, vector databases, and retrieval strategies for knowledge-grounded AI applications.
3. Analyze and implement advanced RAG techniques such as query transformation, re-ranking, self-RAG, and evaluation frameworks to improve accuracy and reliability.

Broad Course Description

Agentic AI refers to intelligent systems capable of acting autonomously to achieve defined goals by perceiving context, reasoning, planning, making decisions, and executing actions with minimal human intervention. Unlike traditional AI models that primarily respond to prompts, Agentic AI systems can operate as goal-driven agents that use tools, APIs, memory, and feedback loops to solve multi-step problems. These systems integrate capabilities such as task decomposition, autonomous decision-making, workflow orchestration, and adaptive learning, enabling applications in automation, software development, intelligent assistants, robotics, cybersecurity, and enterprise operations. Agentic AI represents a shift from reactive AI toward proactive, self-directed, and collaborative intelligent systems.
Retrieval-Augmented Generation (RAG) is an advanced AI framework that enhances large language models by combining information retrieval with generative capabilities. Instead of relying solely on pre-trained knowledge, RAG retrieves relevant information from external data sources such as databases, knowledge graphs, vector stores, or enterprise documents and uses that context to generate more accurate, up-to-date, and domain-specific responses. This approach improves factual grounding, reduces hallucinations, and enables customized knowledge-driven applications such as intelligent search, question answering, chatbots, and decision support systems. RAG typically integrates embeddings, vector databases, retrievers, and generative models to create scalable and trustworthy AI solutions.
Together, Agentic AI and RAG form a powerful foundation for next-generation intelligent systems, where autonomous agents can reason, plan, retrieve relevant knowledge, and execute tasks dynamically in complex environments. Their integration supports advanced applications such as autonomous research assistants, AI copilots, intelligent automation, multi-agent systems, and enterprise knowledge agents, making them central to the future of generative and autonomous AI

UNIT – I	09 Hours

Foundations of Gen AI & LLM:

Introduction to Generative AI: Scope, Applications, and Industry Use Cases, Transformer Architecture:
Encoder-Decoder, Self-Attention, Multi-Head Attention, Tokenization Strategies: BPE, WordPiece, SentencePiece, Pre-training vs Fine-tuning: Concepts and Differences, Prompt Engineering Fundamentals: Zero-shot, Few-shot, Chain-of-Thought, Overview of Popular LLMs: GPT-4, Claude, LLaMA, Mistral, Gemini, LLM Evaluation Metrics: BLEU, ROUGE, Perplexity, Human Evaluation, Limitations of Base LLMs: Hallucinations, Stale Knowledge, Context Window Constraints, Introduction to LangChain and LlamaIndex Frameworks, Hands-on: Setting up LLM API (OpenAI / Anthropic) and basic prompt interaction.
Textbook 1: Chapter: 1, 2, 3

UNIT – II	09 Hours
Retrieval-Augmented Generation (RAG): Motivation for RAG: Solving Hallucination and Knowledge Staleness, RAG Pipeline Architecture: Ingestion → Retrieval → Augmentation → Generation, Document Loaders: PDF, HTML, CSV, DOCX using LangChain, Text Chunking Strategies: Fixed-size, Recursive, Semantic Chunking, Embedding Models: OpenAI Embeddings, Sentence-BERT, INSTRUCTOR, BGE, Vector Databases: FAISS, Chroma, Pinecone, Weaviate – Setup and Comparison, Similarity Search: Cosine Similarity, Euclidean Distance, HNSW Indexing, Retrieval Techniques: Dense Retrieval, Sparse (BM25), Hybrid Retrieval, Context Window Management: Stuffing, Map-Reduce, Refine Strategies, Hands-on: Build an end-to-end RAG pipeline over a PDF knowledge base. Textbook 2: Chapter: 4, 5, 6, 7	
UNIT – III	**09 Hours**
Foundations of Retrieval- Augmented Generation: Naive RAG vs Advanced RAG vs Modular RAG: Architectural Comparison, Query Transformation: HyDE, Step-back Prompting, Multi-query Retrieval, Re-ranking Strategies: Cross-Encoder Re-ranking, Cohere Rerank, ColBERT, Self-RAG: Adaptive Retrieval with Reflection Tokens, Corrective RAG (CRAG): Dynamic Retrieval Correction, Multi-vector Retrieval: Parent-Child Chunks, Summary Indexing, Contextual Compression and LLM-based Filtering, RAG Evaluation Frameworks: RAGAS, TruLens – Faithfulness, Relevance, Groundedness, Guardrails and Output Validation in RAG Systems, Hands-on: Implement Self-RAG and evaluate using RAGAS on a custom dataset. Textbook 4: Chapter: 5, 6, 7	
UNIT – IV	**09 Hours**
Agentic AI – Autonomous Agents & Multi Agent System: Introduction to AI Agents: Definition, Properties, and Cognitive Architecture, ReAct Framework: Reasoning + Acting – Thought-Action-Observation Loop, Tool Use and Function Calling: APIs, Calculators, Web Search, Code Execution, Memory in Agents: Short-term (Conversation Buffer), Long-term (Vector Memory), Episodic Memory, Planning Strategies: Tree-of-Thoughts, Plan-and-Solve, Hierarchical Planning, LangChain Agents: AgentExecutor, Custom Tools, OpenAI Functions Agent, Introduction to Multi-Agent Systems: CrewAI, AutoGen, LangGraph, Agent Roles and Collaboration: Planner, Executor, Critic, Synthesizer, Human-in-the-Loop (HITL): Checkpoints, Interrupts, and Approval Gates, Hands-on: Build a multi-tool Research Agent using LangGraph with web search and RAG. Textbook 4: Chapter: 1, 2, 3, 4, 5	
UNIT – V	**09 Hours**
Production Deployment, Security & Real-World Applications: Agentic RAG: Combining Agents with Dynamic Retrieval for Complex Tasks, Fine-tuning vs RAG vs Agents: When to Use Which Strategy, Productionizing RAG: Caching, Async Processing, Load Balancing, LLMOps: Monitoring, Observability with LangSmith, Phoenix Arize, Security in LLM Systems: Prompt Injection, Jailbreaking, Data Privacy, Responsible AI: Bias Detection, Toxicity Filtering, Output Guardrails (NeMo Guardrails), Cost Optimization: Token Management, Model Routing, Caching Strategies, Real-world Case Studies: Customer Support Bots, Code Assistants, Legal Research, Medical QA, Emerging Trends: Multimodal RAG, Agentic Workflows with Vision Models, Capstone Project: Design and present a full Agentic RAG system for a domain of choice. Textbook 1: Chapter: 8, 9, 10	

AI Integration & Authentic Intelligence: Reflection (AI² Pedagogy)

- **AI-Assisted Prompt Engineering and Agent Design:** Students use generative AI

tools to design prompts, develop agent workflows, and analyze how autonomous agents reason, plan, and execute tasks in real-world scenarios.

- **Retrieval-Augmented Learning with Knowledge Systems:** Students explore RAG frameworks by integrating vector databases, retrievers, and LLMs to understand contextual retrieval, factual grounding, and knowledge-driven response generation.
- **Human-AI Collaboration for Problem Solving:** Learners critically evaluate how humans collaborate with AI agents in decision-making, comparing automated agent outputs with human reasoning to assess accuracy, trust, and reliability..
- **Authentic Reflection on AI Agent Behavior:** Students reflect on agent autonomy, bias, hallucination, and ethical implications while analyzing the strengths and limitations of Agentic AI and RAG systems.
- **Ethical and Responsible AI Reflection:** Students examine privacy, security, governance, and responsible deployment challenges associated with autonomous agents and retrieval-augmented generative systems..

Applications:

1. Autonomous Research Assistants
2. Enterprise Knowledge Assistants
3. AI Copilots for Software Development
4. Healthcare Decision Support Systems
5. Multi-Agent Workflow Automation

Laboratory Component

The lab emphasizes hands-on implementation, visualization, and AI-assisted debugging.

Lab Experiments

1. Design a simple autonomous agent to perform goal-based task execution using prompt engineering.
2. Create a retrieval-augmented generation system using embeddings, vector database, retriever, and LLM.
3. Implement document chunking, embedding generation, and similarity-based retrieval.
4. Build a system that retrieves knowledge from custom documents and generates contextual answers.
5. Design an agent capable of calling APIs or external tools to complete multi-step tasks.

Mini Project Component

- **Industry-Driven Projects**

Some sample projects described below simulate the **real software engineering problems**.

Project 1 – Intelligent Research Assistant using RAG

- **Problem:** Develop a system that retrieves information from research papers and generates contextual answers for academic queries..
- **Concepts used:** Vector embeddings, document indexing, semantic search, retrieval pipelines, LLM prompting, RAG architecture.
- **Industry relevance:** Used in research automation, knowledge management, digital libraries, and academic assistant systems.

Project 2 – Agentic Customer Support Chatbot

- **Problem:** Build an autonomous AI agent that retrieves company policy data and resolves customer queries through multi-step reasoning
- **Concepts used:** AI agents, tool calling, RAG pipelines, memory, prompt engineering, workflow orchestration.
- **Industry relevance:** Applied in customer support automation, enterprise helpdesks, and intelligent service assistants.

Project 3 – Resume Screening Assistant using Agentic RAG

- **Problem:** Create an AI agent that retrieves job descriptions, evaluates resumes, and ranks candidates automatically.
- **Concepts used:** RAG, autonomous agents, embeddings, semantic matching, reasoning workflows, vector databases.
- **Industry relevance:** Used in HR automation, talent analytics, and intelligent recruitment platforms.

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
CO1	Understand the architecture and working principles of Large Language Models, transformer attention mechanisms, and their role in generative AI systems.	L4
CO2	Design and implement a complete RAG pipeline including document ingestion, embedding generation, vector storage, and context-aware response generation.	L5
CO3	Apply advanced RAG strategies such as re-ranking, query transformation, multi-vector retrieval, and evaluation frameworks to build production-grade RAG systems.	L5
CO4	Design, implement, and orchestrate LLM-powered autonomous agents with tool use, memory, planning, and multi-agent coordination capabilities.	L5
CO5	Deploy, monitor, secure, and optimize Agentic AI and RAG systems in real-world production environments while addressing ethical considerations and compliance requirements.	L5

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)											Program Specific Outcomes (PSOs)	
	PO1	PO2	PO3	PO4	PO5	PO 6	PO 7	PO 8	PO 9	PO1 0	PO1 1	PSO 1	PSO 2
CO1	3	2	2	3	2						2	3	2
CO2	3	3	2	3	3						2	3	2
CO3	3	3	2	3	3						2	3	2
CO4	3	3	2	3	3						2	3	2
CO5	2	3	2	3	3						2	3	2

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

Suggested Tools:

- Frameworks & Libraries: LangChain, LlamaIndex, LangGraph, CrewAI, AutoGen, HuggingFace Transformers, RAGAS, TruLens, NeMo Guardrails
- LLM & Embedding APIs: OpenAI GPT-4 / GPT-4o, Anthropic Claude API, Cohere Embed & Rerank, Sentence-Transformers, BGE
- Vector DBs & Observability: FAISS, Chroma, Pinecone, Weaviate, Qdrant, LangSmith, Phoenix Arize, Google Colab / Jupyter

TEXT BOOKS:

1. Building LLM Powered Applications – Valentina Alto, Packt Publishing, 2024
2. Generative AI with LangChain (Second Edition): Build Production-Ready LLM Applications and Advanced Agents using Python, LangChain, and LangGraph – Ben Auffarth and Leonid Kuligin, Packt Publishing, 2025.
3. *Developing Apps with GPT-4 and ChatGPT* – Olivier Caelen & Marie-Alice Blete, O'Reilly Media, 2nd Edition, 2024.
4. *AI Agents in Action* – Micheal Lanham, Manning Publications, 2025. ISBN: 978-1633436343.

E-RESOURCES:

1. https://www.deeplearning.ai/short-courses/building-agentic-rag-with-llamaindex/
2. https://python.langchain.com/docs/
3. https://docs.llamaindex.ai/
4. https://developer.nvidia.com/blog/

Activity Based Learning (Suggested Activities in Class)

1. Student seminars (on topics of the syllabus and related aspects (individual activity).
2. Quiz (on topics where the content can be compiled by smaller aspects and data (Individuals or groups as teams).
3. Study projects (by very small groups of students on selected local real-time problems about the syllabus or related areas. The individual participation and contribution of students shall be ensured (team activity).

****

Business Intelligence & Analytics

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

SEMESTER – VII

Subject Code	:		**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	45 (30Hrs (L) + 15 hRs(J))
L–T–P-J	:	2–0–0-2			

Course Learning Objectives:

This Course will enable students to:

1. Understand BI life cycle and data architecture.
2. Apply data preparation and warehousing concepts.
3. Employ ML and AI for predictive analytics and decision support.
4. Implement secure analytics using blockchain verified data.
5. Develop sustainable and ethical BI dashboards.

Broad Course Description

This course focuses on transforming large data sets into actionable business insights through data warehousing, analytics, and visualization techniques. Students apply machine learning, secure data processing, and explainable AI to build intelligent, ethical, and sustainable decision-support systems for modern enterprises.

UNIT – I	**06 Hours**

BI FOUNDATIONS:

Data driven decision making concepts. BI architecture – data sources, ETL, data marts. Dimensional modeling – Star vs Snowflake schema. OLAP and ROLAP systems.
(Text Book 1 : Ch. 1 1.1–1.4; Text Book 2 : Ch. 2 2.1–2.3)

UNIT – II	**06 Hours**

DATA PREPARATION AND INTEGRATION:

Data cleaning and transformation techniques. Handling missing values, outlier detection. Integration of heterogeneous data sources. Use of ETL tools (Talend, Informatica, Power Query)
(Text Book 1 : Ch. 3 3.1–3.4)

UNIT – III	**06 Hours**

ANALYTICS MODELS AND VISUALIZATION:

Descriptive, Predictive, Prescriptive analytics. ML methods – regression, classification, clustering. Visualization tools – Tableau, Power BI, Matplotlib. KPI design and dashboard principles.
(Text Book 1 : Ch. 5 5.1–5.5; Text Book 2 : Ch. 3 3.1–3.4)

UNIT – IV	**06 Hours**

ADVANCED BI & SECURE ANALYTICS:

Big data platforms (Hadoop, Spark). Real-time analytics and stream processing. Blockchain verified data and data lineage. Explainable AI (XAI) in analytics.
(Text Book 2 : Ch. 4 4.1–4.3)

UNIT – V	**06 Hours**

ETHICAL AND SUSTAINABLE ANALYTICS:

Data ethics and bias awareness. Regulatory compliance (GDPR, HIPAA). Green data centers and energy metrics. AI governance frameworks for analytics solutions.
(Text Book 1 : Ch. 7 7.1–7.4)

AI Integration & Authentic Intelligence: Reflection (AI² Pedagogy)

- Fair and transparent AI models.
- Dashboard interpretability and bias control.

Applications:

- Designing dashboards for executive decision-making and performance tracking.
- Applying predictive analytics for market and risk forecasting.
- Integrating blockchain-verified data into BI pipelines for data integrity.
- Using machine learning for customer segmentation and trend analysis.
- Building ethical, explainable AI-driven business decision systems.

Mini Project Component

- Distributed analytics for multi-hospital data (BI + Blockchain).
- Energy-aware data center monitoring dashboard.
- Project: ETL pipeline building for multi sensor datasets.
- Journal: Draw a BI architecture for hospital IoT use case
- Journal: Design dashboard for IoT health blockchain data
- Project: Integrate BI dashboard with smart contract ledger.
- Capstone: “Responsible AI Business Dashboard” project for smart security ecosystem.

Course Outcome	Description	Bloom’s Taxonomy Level
At the end of the course the student will be able to:		
CO1	Describe architecture and components of BI systems.	L2
CO2	Perform ETL operations and data integration.	L3
CO3	Apply analytics and ML models for decision making.	L3
CO4	Implement secure and auditable analytics using blockchain.	L4
CO5	Design ethical and sustainable BI applications.	L4

Table: Mapping Levels of COs to POs / PSOs

COS	Program Outcomes (POs)											Program Specific Outcomes (PSOs)	
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO 1	PSO 2
CO1	3	2			2						2	2	
CO2	2	3	2	2	3						2	3	
CO3	2	3	3	2	3						2	3	
CO4	2	2	3	2	3			2			2	3	
CO5	2	2	3	2	2		2	2			2	3	

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

TEXT BOOKS:

1. Ramesh Sharda et al., Business Intelligence and Analytics, Pearson.
2. Thomas H. Davenport, Analytics at Work, Harvard Press.

NVIDIA Course Mapping (Industry Alignment)

NVIDIA DLI Course / Content	Mapped Units	Relevance
Accelerated Data Science with RAPIDS	II–IV	GPU ETL and analytics
Responsible AI Development	V	Ethical AI dashboards

E-RESOURCES:

1. nptel.ac.in
2. coursera.org
3. edx.org
4. learn.microsoft.com
5. developer.nvidia.com

Activity-Based Learning (Suggested Activities in Class)

- Design a simple ETL pipeline for sample CSV data using Power Query.
- Create a Tableau dashboard with multiple KPIs and charts.
- Run a prediction model on sales data using Python or R.
- Present ethical considerations of AI-based business decisions.
- Conduct a group project on real-time data analytics case study.

****

Unreal Engine Game Development for Beginners

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

SEMESTER – VII

Subject Code	:		**Credits**	:	03
Hours / Week	:	04 Hours	**Total Hours**	:	45Hrs(30 Hrs L + 30 Hrs J)
L–T–P-J	:	2–0–0-2			

Course Learning Objectives:

This Course will enable students to:

1. Gain familiarity with Unreal Engine 5 editor, workflow, and core systems.
2. Develop game logic using Blueprint Visual Scripting without C++.
3. Create visually stunning environments using Lumen, Nanite, and Quixel Megascans.
4. Build and deploy complete interactive 3D game experiences.

Broad Course Description

Unreal Engine 5 (UE5) is used for AAA games, virtual production, architectural visualization, and automotive design. This course focuses on Blueprint Visual Scripting, UE5 editor (Actors, Components, World Outliner), environment design (Landscape, Foliage, Water, Sky Atmosphere), Quixel Megascans, Lumen GI, Nanite geometry, UMG UI, MetaSound, Animation Blueprint, Sequencer cinematics, Niagara particles, and Windows/Linux deployment.

UNIT – I	**06 Hours**

Unreal Engine 5 Foundations: UE5 Editor – Viewport, Content Browser, World Outliner, Details Panel, Actors, Components, Levels, Coordinate systems, Importing assets, Blueprint fundamentals – Event Graph, variables, execution flow, UE5 Project templates.

- Luis Cataldi & Andi McClure, "Unreal Engine 5 Beginner's Guide" Ch. 1–3 (Packt, 2023)
- Epic Games, Official UE5 Documentation (docs.unrealengine.com)

UNIT – II	**06 Hours**

Blueprint Visual Scripting: Blueprint Class types – Actor, Pawn, Character, GameMode, Blueprint nodes – Events, Functions, Macros, Variables, Blueprint communication – Direct reference, Event Dispatchers, Interfaces, Casting, Enhanced Input System – Input Mapping Context, Blueprint debugging.

• Joeri Ouwerkerk, "Blueprints Visual Scripting for UE5" Ch. 1–5 (Packt, 2022)
• Luis Cataldi Ch. 4–6

UNIT – III	**06 Hours**

Environment Design & Rendering: Landscape – sculpting, painting, erosion tools, Foliage Tool, Water Body system, Sky Atmosphere, Volumetric Clouds, Quixel Megascans via Bridge, Material Editor – PBR texturing, Material Instances, Lumen – dynamic GI and reflections, Nanite – virtualised geometry, TSR.

- Luis Cataldi Ch. 7–9
- Quixel Megascans Documentation (quixel.com/megascans/documentation)
- NVIDIA RTX/Lumen Comparison Documentation (docs.unrealengine.com/5.0/RealTimeLumenGlobalIllumination)

UNIT – IV	**06 Hours**

Gameplay Systems & UI: Collision profiles, Pickup/interaction systems, Inventory basics, Game Framework – GameMode, GameState, PlayerController, UMG – Widget Blueprints, HUD design, Main menu/pause menu, MetaSound, Animation Blueprint – State Machine, Blend Spaces, Aim Offset.

- Joeri Ouwerkerk Ch. 6–9
- UMG UI Designer Documentation (docs.unrealengine.com/5.0/umg-ui-designer)

UNIT – V	06 Hours
Cinematics, Polish & Deployment: Sequencer – cinematic cutscenes, camera animation, Timeline tracks, Cinematic cameras – DoF/focal length, Chaos physics – destructible meshes/cloth, Niagara particle system, Post-process volumes, Performance optimization – LODs, occlusion culling, Packaging for Windows/Linux. • Luis Cataldi Ch. 10–12 • NVIDIA RTX in Unreal Engine (developer.nvidia.com/rtx/unreal-engine) • Epic Games, "Niagara Visual Effects System" Documentation	

NVIDIA Deep Learning Institute (DLI) Course Mapping

The following NVIDIA DLI courses are recommended to supplement the subject. Free courses are accessible at learn.nvidia.com. Paid instructor-led workshops (ILT) are available via NVIDIA partner institutions.

UNIT	NVIDIA DLI Course Name	Type	Duration	URL / Code
Unit 3	How to Build a Native OpenUSD XR Application Self-paced: building XR/3D applications with OpenUSD – directly	Self-Paced	2 Hrs	FREE DLI+S-OV-17+V1
	relevant to UE5's OpenUSD integration and 3D pipeline in Unit 3.			
Unit 3	Getting Started With USD for Collaborative 3D Workflows USD workflows for 3D collaboration – applicable to UE5's Datasmith/OpenUSD import pipeline and environment design in Unit 3.	Self-Paced	2 Hrs	FREE DLI+S-OV-20+V1
Unit 3	Essentials of USD in NVIDIA Omniverse Essential USD concepts in Omniverse – foundational for UE5 environment pipeline (Quixel, Megascans, USD assets) in Unit 3.	Self-Paced	2 Hrs	FREE DLI+S-OV-15+V1
Unit 3	Building a 3D Product Configurator With OpenUSD and Omniverse Interactive 3D configurators with Omniverse/OpenUSD – applicable to UE5 architectural visualization workflows.	Self-Paced	2 Hrs	FREE DLI+S-OV-16+V1
Units 3–5	Essentials of Developing Omniverse Kit Applications Omniverse Kit development – relevant to NVIDIA-powered real-time rendering and physics in UE5 environment workflows.	Self-Paced	4 Hrs	FREE DLI+S-OV-11+V1
Unit 3	Easily Develop Advanced 3D Layout Tools on NVIDIA Omniverse Advanced 3D layout tools – directly applicable to UE5 environment layout, Level design, and spatial design in Unit 3.	Self-Paced	2 Hrs	FREE DLI-SP-EDATLO
Unit 5	Bootstrapping Computer Vision Models with Synthetic Data ILT: generate synthetic data using Omniverse –	Workshop (ILT)	8 Hrs	PAID DLI-ILT-

	relevant to UE5 Sequencer-based synthetic dataset generation for AI training.			BCVM SD
Units 4–5	Generative AI Explained GenAI overview – applicable to UE5's MetaHuman Creator, AI NPC behavior, and procedural content generation.	Self-Paced	1 Hr	FREE DLI+S-FX-07+V1
Unit 5	Synthetic Data Generation for Training Computer Vision Models Creating photorealistic training datasets using Omniverse – extends UE5's Sequencer and Nanite pipeline for AI training data generation.	Self-Paced	2 Hrs	FREE DLI-SP-SDGCVM
Units 3–5	Build Beautiful, Custom UI for 3D Tools on NVIDIA Omniverse Professional 3D UI design patterns – applies to UE5 UMG widget design for game HUDs, menus, and interactive visualization.	Self-Paced	2 Hrs	FREE DLI+S-OV-12+V1

AI Integration & Authentic Intelligence: Reflection (AI² Pedagogy)

- AI-Assisted Environment Generation: Use UE5's PCG framework and AI tools for environments.
- MetaHuman Creator Exploration: AI-powered digital humans and implications for identity.
- Blueprint Code Review with AI: Convert Blueprint logic to C++ using AI assistants.
- AI NPC Design: Compare behavior trees with LLM-driven NPC dialogue systems.
- Responsible Realism: Ethics of photorealistic rendering – deepfakes and digital doubles.

Applications

- AAA Video Game Development (Fortnite, Hellblade II)
- Film and Virtual Production (The Mandalorian)
- Architectural Visualization
- Automotive Design and Virtual Showrooms
- Training Simulations and Defense Applications
- Live Events and Concert Visualizations

Laboratory Component (1 Credit)

The lab emphasizes hands-on implementation, visualization, and AI-assisted debugging.

Lab Experiments

- Navigate UE5 editor; build furnished room scene using Starter Content.
- Create Blueprint Actor that rotates, changes color on overlap.
- Implement third-person character with movement/sprint/jump via Enhanced Input.
- Sculpt and paint 500m×500m landscape with foliage and water body.
- Set up outdoor environment with Lumen GI, Sky Atmosphere, volumetric clouds.
- Build pickup system with Blueprint-driven HUD counter.
- Create main menu Widget Blueprint with New Game, Continue, Quit buttons.

- Design enemy Blueprint with patrol behavior using timers and splines.
- Create Sequencer cinematic with character animation, camera movement, audio.
- Package and export playable game level as standalone Windows executable.

Mini Project Component

Industry-driven projects that simulate real software engineering problems.

Project 1 – First-Person Exploration Game

Problem: Photorealistic FP exploration with environmental storytelling, interactive objects, and atmospheric audio.

Concepts used: UE5 environment design, Blueprint interaction, Lumen/Nanite, MetaSound

Industry relevance: Walking simulation genre – popular on Steam and itch.io.

Project 2 – Third-Person Adventure Level

Problem: 3rd-person adventure level with enemies, collectibles, boss encounter, and cutscene.

Concepts used: Character Blueprint, AI behavior trees, Sequencer, UMG

Industry relevance: Action-adventure games – largest commercial game genre.

Project 3 – Architectural Visualization Experience

Problem: Photorealistic building walkthrough with real-time lighting and tour animation.

Concepts used: Quixel Megascans, Lumen, Nanite, Sequencer, Material Instances

Industry relevance: Architecture firms, real estate, interior design.

Course Outcomes

CO	Description	Bloom's
At the end of the course the student will be able to:		
CO1	Navigate UE5 editor proficiently and manage Actors, Components, and Level assets in 3D project workflows.	L2
CO2	Implement complete game logic using Blueprint Visual Scripting including character control and game state management.	L3
CO3	Design photorealistic 3D environments using Landscape, Quixel Megascans, Lumen, and Nanite technologies.	L3
CO4	Build gameplay systems including UI/HUD, animation, audio integration, and game framework setup.	L4
CO5	Create cinematic sequences with Sequencer, apply Niagara particles, optimize performance, and deploy packaged game builds.	L5

Table: Mapping Levels of COs to POs / PSOs

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PSO 1	PSO 2
CO1	2	2	2	2	3						2	2	2
CO2	3	3	3	3	3						2	3	3
CO3	3	3	3	3	3						3	3	3
CO4	3	3	3	3	3						3	3	3
CO5	3	3	3	3	3						3	3	3

Suggested Tools

- Unreal Engine 5

- Quixel Bridge – Megascans
- MetaHuman Creator
- Blender
- Visual Studio (C++ companion)
- Audacity / Reaper

TEXT BOOKS:

1. Luis Cataldi & Andi McClure, "Unreal Engine 5 Beginner's Guide", Packt, 2023.
2. Joeri Ouwerkerk, "Blueprints Visual Scripting for Unreal Engine 5", Packt, 2022.
3. Epic Games, "Unreal Engine 5 Official Documentation" (docs.unrealengine.com).

REFERENCE BOOKS:

4. Sjoerd De Jong, "Unreal Engine Lighting and Rendering Guide" (Unreal Engine Community Wiki).
5. Tom Shannon, "Unreal Development Kit Game Design", Packt, 2012.

E-RESOURCES:

- Unreal Online Learning (dev.epicgames.com/community/learning)
- Unreal Sensei YouTube Channel
- William Faucher YouTube – UE5 Environment Art
- NVIDIA DLI: learn.nvidia.com

Activity Based Learning (Suggested Activities in Class)

- Unreal Game Jam – 72-hour game creation challenge.
- Environment art showcase – students display polished UE5 scenes.
- Guest lectures from virtual production and architectural visualization professionals.

****

Quantum Cryptography

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

SEMESTER – VII

Subject Code	:		**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	45 Hrs (30 Hrs L + 15 Hrs J
L–T–P-J	:	2–0–0-2			

Prerequisites: Cryptography, Quantum Computing Systems

Course Learning Objectives:

This Course will enable students to:

1. Understand classical and quantum cryptography fundamentals
2. Learn information theory and quantum information concepts
3. Analyze quantum key distribution and secret key distillation
4. Explore quantum cryptographic protocols and their implementations
5. Evaluate security aspects of quantum communication systems

Broad Course Description

The course Quantum Cryptography provides a comprehensive understanding of secure communication systems by integrating classical cryptographic techniques with quantum information principles. It covers foundational concepts of classical cryptography, information theory, and quantum mechanics, followed by advanced topics such as Quantum Key Distribution (QKD), secret key distillation, and protocols like BB84 and GG02. The course also explores security analysis against eavesdropping and quantum attacks. Overall, it equips students with the knowledge to design secure, future-ready cryptographic systems for applications in cybersecurity, finance, and communication networks.

UNIT – I	**08 Hours**

Introduction to Classical Cryptography

Introduction, classical cryptography: Confidentiality and secret-key ciphers, Secret-key authentication, Public-key cryptography

(Text Book-1: Chapter 2)

UNIT – II	**10 Hours**

Information Theory and Quantum Information Theory

Source coding, Joint and conditional entropies, Channel coding, Renyi entropies, Continuous variables, Perfect secrecy revisited. Fundamental definitions in quantum mechanics, Qubits and qubit pairs, Density matrices and quantum systems, Entropies and coding, Particularity of quantum information, Quantum optics.

(Text Book-1: Chapter 3, Chapter 4)

UNIT – III	**09 Hours**

Cryptosystems based on Quantum Key Distribution and Secret Key Distillation:

A key distribution scheme, a secret-key encryption scheme, Combining quantum and classical cryptography, Implementation of a QKD-based cryptosystem. Secret Key Distillation: A two-step approach, Characteristics of distillation techniques, authenticated one-shot secret-key distillation, Authenticated repetitive secret-key distillation, unauthenticated secret-key distillation, secret-key distillation with continuous variables.

(Text Book 1: Chapter 5, Chapter 6)

UNIT – IV	**09 Hours**

The BB84 Protocol, and GG02 Protocol

Description, Implementation of BB84, Eavesdropping and secret key rate, from discrete to continuous variables, a protocol with squeezed states, a protocol with coherent states: the GG02 protocol,

Implementation of GG02, GG02 and secret-key distillation. (Text Book1: Chapter 10, Chapter 11)

UNIT – V	**09 Hours**
Security Analysis of Quantum Key Distribution Eavesdropping strategies and secret-key distillation, Distillation derived from entanglement purification, Application to the GG02 protocol. (Text Book1: Chapter 12)	

AI Integration & Authentic Intelligence: Reflection (AI² Pedagogy)

- **Secure Intelligent Systems Design:** Integrate classical and quantum cryptographic techniques to build secure AI systems that ensure confidentiality, integrity, and authentication of data.
- **Information-Theoretic AI Security:** Apply entropy, coding theory, and quantum information principles to enhance data privacy, secure communication, and robust AI model training.
- **Quantum-Safe Cryptographic Models:** Utilize Quantum Key Distribution (QKD) and post-quantum cryptography to protect AI systems from future quantum attacks.
- **AI-Driven Threat Detection:** Leverage AI techniques to identify eavesdropping strategies, anomalies, and potential attacks in quantum communication systems.
- **Reliable and Noise-Aware Security Systems:** Incorporate quantum noise models and error handling in cryptographic protocols to ensure robustness and reliability.
- **Responsible & Ethical AI Security:** Ensure privacy preservation, fairness, and ethical data usage in AI-driven cryptographic systems, aligning with global security and data protection standards.

Suggested Tools:

- QuNetSim
- SimulaQron
- PennyLane
- GoogleColab

Applications:

1. Secure communication systems using Quantum Key Distribution (QKD)
2. Cybersecurity and quantum-safe encryption for AI systems
3. Financial transaction security and blockchain technologies
4. Defense and secure communication networks
5. Privacy-preserving AI systems and secure data sharing
6. Cloud security using quantum-resistant cryptographic methods
7. Secure IoT and smart city infrastructures

Mini Project Component

Industry-Driven Projects (Cyber security / Quantum Security Style)

These projects simulate real-world **secure communication and quantum cryptography applications**.

Project 1 – Quantum Key Distribution Simulator (BB84)

- **Problem:** Design and simulate the BB84 protocol for secure key exchange
- **Concepts used:** QKD, qubits, measurement, eavesdropping detection
- **Industry relevance:** Secure communication systems

Project 2 – Secure Communication System using QKD

- **Problem:** Develop a system combining classical encryption with quantum key distribution
- **Concepts used:** Hybrid cryptography, key exchange, encryption
- **Industry relevance:** Banking, defense, secure networks

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Explain classical and quantum cryptography concepts	L2
2	Apply information theory concepts in quantum systems	L3
3	Analyze quantum key distribution and key distillation techniques	L4
4	Evaluate quantum cryptographic protocols and their implementations	L5
5	Assess security aspects of quantum key distribution systems	L5

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	
CO2	3	3	2	2	2							3	
CO3	2	3	2	3	2							3	2
CO4	2	3	2	3	3							3	2
CO5	3	3	3	3	3	2		2			2	3	3

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

TEXT BOOKS:

1. Gilles Van Assche "**QUANTUM CRYPTOGRAPHY AND SECRET-KEY DISTILLATION**", Cambridge University Press

REFERENCE BOOKS:

2. **Post-Quantum Cryptography** – Springer.
3. **An Introduction to Quantum Computing** – Oxford University Press.
4. **Introduction to Quantum Information Science** – Oxford University Press.

E-RESOURCES:

1. https://qiskit.org
2. https://pennylane.ai
3. https://quantum.ibm.com
4. https://nptel.ac.in

Activity Based Learning (Suggested Activities in Class)

Activity 1: Quantum Protocol Simulation
Activity 2: Cryptographic Vulnerability Analysis
Activity 3:Role-play demonstration of sender–receiver–eavesdropper scenario in secure communication
Activity 4: Group discussion on classical vs quantum cryptography and post-quantum challenges

****

Digital Business

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

SEMESTER – VII

Subject Code	:	**Credits**	: 03
Hours / Week	: 03 Hours	**Total Hours**	: 45 Hrs L
L–T–P-J	: 3–0–0-0		

Course Learning Objectives:

This Course will enable students to:

1. To understand the fundamental concepts of digital business, e-commerce, and their role in the modern digital economy.
2. To analyze various digital business models, including mobile commerce, social commerce, and platform-based ecosystems.
3. To apply digital business processes, payment systems, and logistics in real-world business scenarios.
4. To evaluate digital business applications such as B2B, B2C, FinTech, and e-governance systems.
5. To develop data-driven decision-making skills using business analytics, artificial intelligence, and modern digital tools.

Broad Course Description

This course provides a comprehensive understanding of digital business and how technology is transforming modern organizations and markets. It covers the fundamentals of e-commerce, mobile and social commerce, and the digital business ecosystem, including online transactions, logistics, and digital payment systems. The course further explores various digital business applications such as B2B, B2C, FinTech, and e-governance, highlighting their real-world impact. It also introduces business analytics, artificial intelligence, and data-driven decision-making to support strategic business operations. Overall, the course equips students with the knowledge and skills required to design, analyze, and manage digital business solutions in today's technology-driven environment.

UNIT – I	**09 Hours**

The Digital Revolution and Society, Digital Economy and Digital Enterprise, Virtual Communities and E-Commerce Definition, E-Commerce Frameworks, E-Business, E-Markets, Networks Classification of E-Commerce (by Nature of Transaction), E-Commerce Business Models (B2B, B2C, C2C, etc.), Drivers, Benefits, and Limitations of E-Commerce, Impact of E-Commerce on Business, Government, and
Society.

UNIT – II	**09 Hours**

Data Collection and Tools

M-Commerce Applications and Benefits, Mobile Marketing and Location-Based Services Social Commerce Concepts and Models, Social Business Networks and Enterprise Social

Media, Social Media Marketing and Enterprise 2.0, Social Collaboration and Digital Engagement.	
UNIT – III	**09 Hours**
E-Commerce Mechanisms and Online Purchasing Process, E-Marketplaces and Digital Intermediation Customer Shopping Mechanisms and Merchant Solutions, Auction Mechanisms and Dynamic Pricing Order Fulfilment Process and Logistics, Outsourcing, Partnerships, and Mass Customization, Digital Payments: Smart Cards, Stored Value Cards, Payment Gateways.	
UNIT – IV	**09 Hours**
B2B Applications: E-Marketplaces, E-Catalogs, Search Engines, E-Procurement and Supply Chain Integration, B2C Applications: Electronic Retailing (E-Tailing), Business Models and Revenue Models	
UNIT – V	**09 Hours**
Introduction to Business Analytics, Evolution and Need for Analytics in Digital Business, Types of Analytics: Descriptive, Predictive, Prescriptive, Tools for Analytics and Data-driven Decision	

AI Integration & Authentic Intelligence: Reflection (AI² Pedagogy)

1. **AI-enabled Digital Market Analysis:** Use AI tools to analyze market trends, customer behaviour, and digital demand patterns while ensuring ethical data usage and minimizing bias in insights.
2. **Intelligent E-Business Modelling:** Apply AI-based analytics to design and evaluate digital business models (B2B, B2C, platforms) with transparent and explainable assumptions.
3. **AI-driven Customer Engagement:** Utilize AI for personalization, recommendation systems, and social commerce strategies while maintaining privacy and responsible data handling.
4. **Smart Digital Operations:** Integrate AI in logistics, supply chain, and digital payment systems to improve efficiency, security, and scalability in business processes.
5. **Responsible AI & Digital Trust:** Evaluate ethical, legal, and societal implications of AI in digital business, ensuring fairness, accountability, transparency, and sustainability in decision-making.

Suggested Tools:

- HubSpot
- Power BI

- Uizard
- Miro
- Notion AI

Applications:

Market Intelligence & Trend Analysis:

Using tools like Google Trends and SEMrush to identify trending products, analyze competitors, and forecast market demand.

Customer Segmentation & Personalization:

Applying HubSpot and Google Analytics to segment customers based on behaviour and deliver personalized recommendations and targeted marketing.

Business Model Design & Strategy Optimization:

Using Canvanizer and Miro to design, visualize, and refine digital business models and strategies.

E-Commerce Development & Digital Transactions:

Leveraging Shopify and Razorpay to build online stores, manage transactions, and ensure secure digital payments.

Social Media Marketing & Customer Engagement:

Using Hootsuite and Mailchimp to manage campaigns, automate communication, and improve customer interaction.

Project Component

1. Identify a trending product/service and analyze its demand, target audience, and competitors.
2. Design and simulate a social media marketing campaign for a startup product.
3. Analyze customer journey and improve user experience using feedback and analytics.
4. Build a dashboard to analyze business data and generate insights using AI.

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
CO1	Explain the fundamentals of digital business, e-commerce models,	L2

	and their impact on business and society.	
CO2	Analyze mobile commerce, social commerce, and IoT applications in the digital ecosystem.	L4
CO3	Apply concepts of digital business ecosystems, online processes, logistics, and digital payment systems.	L3
CO4	Evaluate various digital business applications including B2B, B2C, FinTech, and digital government services.	L5
CO5	Apply business analytics, AI, and data-driven decision-making techniques for digital business solutions.	L3

Table: Mapping Levels of COs to POs / PSOs

COS	Program Outcomes (POs)											Program Specific Outcomes (PSOs)	
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO1	PO11	PSO 1	PSO 2
CO1	2	2	2	2	3			2	2	2	2		3
CO2	2	2	2	2	3			2	2	2	2		3
CO3	2	2	2	2	3			2	2	2	2		3
CO4	2	2	2	3	3			2	2	2	2		3
CO5	2	2	2	2	3			2	2	2	2		3

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

TEXT BOOKS:

1. Dave Chaffey, Tanya Hemphill, David Edmundson-Bird, Digital Business and E-Commerce Management, 7th Edition, Pearson Education, 2019.

REFERENCE BOOKS:

1. Efraim Turban, David King, Dennis Viehland, Jae Lee, Electronic Commerce – A Managerial Perspective, 8th Edition, Pearson Education, 2016.
2. Judy Strauss, Adel Ansary, Raymond Frost (2017), E-Marketing, Prentice Hall.
3. Dave Chaffey, Fiona Ellis-Chadwick, Kevin Johnston, Richard Mayer (2018), Internet Marketing, Pearson Education.

E-RESOURCES:

1. https://onlinecourses.nptel.ac.in/noc25_mg152/preview
2. https://www.cognizant.com/us/en/glossary/digital-business

3. https://www.coursera.org/specializations/digital-business
4. https://www.udemy.com/course/professional-diploma-in-digital-business-development/?srsltid=AfmBOopmojTc_It09-JMMrCe4ORPytSUUulRPVkQ4pd3YgzEf7Pi0FE_&couponCode=CM260417IN

Activity Based Learning (Suggested Activities in Class)

1. Design a digital business model for a startup idea using real-world case studies.
2. Analyze an existing e-commerce platform (UI/UX, payment flow, customer journey) and present findings.
3. Create and present a mock digital marketing campaign for a product using content, targeting, and engagement strategies.

****

CAPSTONE PROJECT-PHASE I

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

SEMESTER – VII

Subject Code	:	24CT4710	**Credits**	:	04
Hours / Week	:	03 Hours	**Total Hours**	:	52 Hours
L–T–P–S	:	0–0–0–8			

Course Learning Objectives:

This Course will enable students to:

1. To apply theoretical knowledge to solve practical problems.
2. Enhance technical, analytical, and problem-solving skills.
3. Foster teamwork and collaboration skills.
4. Develop creativity and innovation in addressing engineering challenges.
5. To analyze and design the solution to the selected problem statement.

DESCRIPTION:

1. Each B. Tech Project must be carried out by a group of students at the Institute. To ensure uniform participation of each student, the group size should be preferably at least 3 but not more than 4 students.
2. Each project activity must be supervised by the faculty members of the Institute. These faculty members are termed Project Guides.
3. In case the project is of multi-disciplinary nature, the Project group can be formed consisting of students from other departments. But there must be at least one student and a project Guide from the department who is offering the Project.
4. The topic proposed by both the guide and the student team should be approved by the department chairman and the department project coordinator to proceed further. A degree of industrial input and involvement will be encouraged and can be facilitated through existing academic-industrial collaborations or by addressing specific topics that are of interest to industrial partners.
5. All projects will be closely supervised by the Project Guide with ongoing feedback and guidance at all stages of the project from conception to completion.
6. The following criteria will be checked by the department chairman to approve for the project proposal:
   a. Department staff as Project guide
      i. Ability to provide direction to the student in the chosen field of interest to formulate a suitable title of the project.
      ii. Ability to design an appropriate strategy and methodology to carry out the Project by the team.
      iii. Ability to provide and evaluate the strong literature review document for the chosen topic
      iv. Ability to train students on paper / technical writing skills
   b. Student Team

i. To be dedicated and committed to work on the project by sharpening the existing and learning new technical skills.
ii. To be committed to completing the project and participate in hackathons and project exhibitions.

8. Phase-1 comprises of Literature Survey, Problem identification, Objectives and Methodology.
9. There will be CIA evaluation (Project reviews) done by a committee of senior faculty of the department based on the rubrics

Course Outcomes:

At the end of the course the student will be able to:

1. Demonstrate the ability to apply engineering principles to solve real-world problems.
2. Develop innovative thinking and thereby preparing students for Capstone project

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
CO1	3	3	3	3	3	3	3	3	3	3	3	3	3	3
CO2	3	3	3	3	3	3	3	3	3	3	3	3	3	3

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

****

CAPSTONE PROJECT-PHASE II

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

SEMESTER – VIII

Subject Code	:	24CT4801	**Credits**	:	12
Hours / Week	:	03 Hours	**Total Hours**	:	156 Hours
L–T–P–S	:	0–0–0–24			

Course Learning Objectives:

This Course will enable students to:

1. Detailed design of the solution to the problem statement and project management using software engineering skills.
2. Write efficient code and test the code to find any bugs and resolve the same leading to completion and deployment of the project using modern tools.
3. Analyze and synthesize the project results.
4. Demonstrate knowledge and understanding of writing the publication/report.
5. Able to work in teams and present the project work.

DESCRIPTION:

1. The problem statement selected in Major Project Phase-I (VII semester) will be carried in the VIII semester.
2. Phase-0II comprises of the detailed design, implementation, and testing results during the internal and external review.
3. Each Project team needs to submit the technical paper or patent or participate in hackathons and project exhibitions as well as apply for various state and national funding agencies within the stipulated time frame by the university.
4. There will be CIA evaluation (Project reviews) done by a committee of senior faculty of the Department.
5. Additionally, there will be a Semester end evaluation of the work done that would include an internal Faculty and an external academic expert.

Course Outcomes:

At the end of the course the student will be able to:

1. Conduct a survey of several available literature in the preferred field of study to find the recent advances and gaps
2. Implement the mathematics concept and engineering fundamentals, and specialization to design a solution using modern tools for the defined problem.
3. Experimenting and evaluating the results from test data to provide a conclusion to the project work.

4. Demonstrate an ability to work in teams and to prepare quality documents of project work & exhibit technical presentation skills.

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
CO1	3	3	3	3	3	3	3	3	3	3	3	3	3	3
CO2	3	3	3	3	3	3	3	3	3	3	3	3	3	3

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

****

INTERNSHIP

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

SEMESTER – VIII

Subject Code	:	24CT4802	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P–S	:	0–0–6–0			

Course Learning Objectives:

This Course will enable students to:

1. To expose students to the industrial environment.
2. To create competent professionals for the industry.
3. To provide possible opportunities to learn, understand and sharpen the real time technical /managerial skills required at the job.
4. To work on a problem assigned by a mentor at industry, prepare action plan and complete within time limit.
5. To learn, create/prepare report for Project/research as used in industry with productive and efficient way .
6. To strengthen industry-institute linkage and increase employability of the students.

Guideline for Internship:

The course includes 16 weeks of on-job training on current industry-relevant problem through supervised self-learning approach The internship is an individual activity. The student should obtain approval from the chairman/supervisor to pursue. A student shall submit a brief proposal about the work to be carried out in the internship, to a coordinator within 3 weeks, after starting the internship.

A comprehensive report is required to be prepared and submit to the department at the end of the semester. A certificate shall be attached with this report duly signed by the competent authority of the industry for the successful completion of the internship. An attendance report shall also be attached with this report. The CIA evaluation will be done by faculty mentor or Industry Supervisor. There is no SEE Exam for this course.

CO No.	Outcomes	Bloom's Taxonomy Level
CO1	Understand the modern tools used in the field of Computer science and engineering for product development.	L2
CO2	Demonstrate ethical conduct and professional accountability while working in a team for the benefit of society.	L2
CO3	Understand the resources requirement and planning to facilitate the project success.	L3
CO4	Develop and refine oral and written communication skills.	L3
CO5	Demonstrate knowledge of the industry in which the internship is done.	L3

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
CO1	3	3	3	3	3	3	3	3	3	3	3	3	3	3
CO2	3	3	3	3	3	3	3	3	3	3	3	3	3	3

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

****