MAR BASELIOS COLLEGE OF ENGINEERING AND TECHNOLOGY

(Autonomous)

(Autonomous Institution under APJ Abdul Kalam Technological University)

Curriculum structure

for

Master of Technology (M.Tech) in Computer Science and Engineering

(Year of Introduction: 2026)

Mar IvaniosVidyanagar
Nalanchira, Thiruvananthapuram,
Kerala, India. Pin: 695015

www.mbcet.ac.in

PROGRAM OUTCOMES - PO

Program outcomes are the attributes that are expected to be demonstrated by a graduate after completing the course.

PO1: An ability to independently carry out research/ investigation and development work in engineering and allied streams

PO2: An ability to communicate effectively, write and present technical reports on complex engineering activities by interacting with the engineering fraternity and with society at large.

PO3: An ability to demonstrate a degree of mastery over the area as per the specialization of the program. The mastery should be at a level higher than the requirements in the appropriate bachelor program

PO4: An ability to apply stream knowledge to design or develop solutions for real world problems by following the standards

PO5: An ability to identify, select and apply appropriate techniques, resources and state-of-the-art tool to model, analyze and solve practical engineering problems.

PO6: An ability to engage in life-long learning for the design and development related to the stream related problems taking into consideration sustainability, societal, ethical and environmental aspects

PO7: An ability to develop cognitive load management skills related to project management and finance which focus on Entrepreneurship and Industry relevance.

The departments conducting the M. Tech course can define their own PSOs if required, and assessment shall also be done for the same.

CONTENTS

		Page No.
	Program Outcomes – PO	2
1.	Semester-wise Distribution of the Courses	4
	a) Semester I	4
	b) Semester II	5
	c) Semester III	5
	d) Semester IV	6
2.	Course types in this Curriculum structure	7
3.	Credit requirements for registering to higher semesters	8
4.	Course Coding scheme	8
5.	Assessment Pattern	9
	a) Program Core Course/ Program Core Course with Practical component	9
	b) Program Elective Course/ Program Elective Course with Practical component	10
	c) Audit Course (Research Methodology and IPR)	10
	d) Internship	11
	e) Laboratory Courses	17
	f) Industry Elective Course	17
	g) Skill/ Ability Enhancement Course	18
	h) Mini Project	18
	i) Project	20
6.	Teaching Assistanceship (TA)	24
7.	Points to Remember	25

1. Semester-wise Distribution of the Courses

a) Semester I (M1)

Slot	Course Type	Course Code	Course	Marks		Hours L - T - P	Credits
				CIA	ESE		
A	PCC	26MA061A	Mathematical Foundations of Computing Systems	50	50	3 – 1 – 0	4
B	PCCP	26CS161A	Advanced Computer Networks	50	50	3 – 0 – 2	4
C	PCC	26CS161B	Advanced Data Structures and Algorithms	50	50	3 – 1 – 0	4
D	PEC/ PECP	26CS1XXX	Program Elective Course1/ Program Elective Course with practical component 1	50	50	3 – 0 – 0/ 2 – 1 – 0/ 2 – 0 - 2	3
E	PEC/ PECP	26CS1XXX	Program Elective Course1/ Program Elective Course with practical component 1	50	50	3 – 0 – 0/ 2 – 1 – 0/ 2 – 0 - 2	3
S	AC	26AC061A	Research Methodology and IPR	50	50	2 – 0 - 0	0
T	LBC	26CS169A	Algorithm Design Laboratory	100	-	0 – 0 - 3	2
			Total	**400**	**300**	**23 - 28**	**20**

Teaching Assistance: Upto 6 hours

Program Elective Courses and Program Elective Course with practical component in M1 and M2 must be chosen from a single basket of all PECs and PECPs.

b) Semester II (M2)

Slot	Course Type	Course Code	Course	Marks		Hours L - T - P	Credits
				CIA	ESE		
A	PCC	26CS161C	Advanced Database Management Systems	50	50	3 – 1 – 0	4
B	PEC/ PECP	26CS1XXX	Program Elective 3	50	50	3 – 0 – 0/ 2 – 1 – 0/ 2 – 0 - 2	3
C	PEC/ PECP	26CS1XXX	Program Elective 4	50	50	3 – 0 – 0/ 2 – 1 – 0/ 2 – 0 - 2	3
D	IEC/ SAEC*	26CS1XXX	Industry Elective / (Skill/Ability Enhancement Course)	50	50	3 – 0 – 0/ 2 – 1 – 0/ 2 – 0 - 2	3
S	PR	26CS167A	Mini project	100	-	0 – 0 - 6	3
T	LBC	26CS169B	Full Stack Development Laboratory	100	-	0 – 0 - 3	2
			Total	**400**	**200**	**22- 26**	**18**

Teaching Assistance: Upto 6 hours

***Marks / GPA earned in this SAEC will be used for awarding GPA for this course.

c) Semester III (M3)

Slot	Course Type	Course Code	Course	Marks		Hours L - T - P	Credits
				CIA	ESE		
A	SAEC**	26CS1XXX	Skill/ Ability Enhancement Course	To be successfully completed		-	3
D	PR	26CS178A	Project (Phase I)/ Project/ Internship	100 100 100	- 100 -	0 – 0 – 24 0 – 0 – 24 Industry norms	16
			Total	**100**	**-/ 100/ -**	**24**	**19**

Teaching Assistance for students doing Project (Phase I)/ Project in the college: 5 hours

** This SAEC can be carried out at any time from M1 to M3, and credited in M3.

d) Semester IV (M4)

Slot	Course Type	Course Code	Course	Marks		Hours L - T - P	Credits
				CIA	ESE		
D	PR	26CS178B	Project (Phase II)/ Project/ Internship	100 100 100	100 100 -	0 – 0 – 24 0 – 0 – 24 Industry norms	16
			Total	**100**	**100/ 100/ -**	**24/ Industry norms**	**16**

Teaching Assistance for students doing Project (Phase II)/ Project in the college: 5 hours

Program Elective courses

#	Course code	Course Name
1.	26CS162A	Data Mining &Warehousing
2.	26CS162B	Soft Computing Techniques
3.	26CS262K	Natural Language Processing (Lab Integrated)
4.	26CS162D	Deep Learning Techniques
5.	26CS162E	Foundations of Data Science
6.	26CS162F	Data Visualization using Python
7.	26CS262A	Data Analytics
8.	26CS162G	Fuzzy set Theory and Applications
9.	26CS162H	Applied Machine Learning
10.	26CS162I	Computer Vision
11.	26CS162J	Computational Linguistics
12.	26CS162K	High Performance computing
13.	26CS162L	Advanced Topics in Distributed Systems
14.	26CS162M	Advanced Operating Systems
15.	26CS162N	Advanced Graph Theory

16.	26CS162O	Essentials of Cyber Security
17.	26CS162P	Block Chain Technologies and its applications
18.	26CS162Q	Software Project Management
19.	26CS162R	Advanced Cloud computing

Industry Elective courses

#	Course code	Course Name	Offering Department
1.	26CS166A	Cyber Forensics Basics	CSE
2.	26CS166B	Stream Processing and Analytics	CSE

2. Course types in this Curriculum structure

AC: Audit Course

IEC: Industry Elective Course - (offered by an industry)

LBC: Laboratory Course

PCC: Program Core Course

PCCP: Program Core Course with practical component

PEC: Program Elective Course

PECP: Program Elective Course with practical component

PR: Project Course - (Internship Course also belongs to this Course type)

SAEC: Skill/Ability Enhancement Course. This is a MOOC offered by AICTE/ NPTEL/ SWAYAM/ NITTTR)

3. Credit requirements for registering to higher semesters

Semester	Allotted credits	Cumulative credits	Minimum credits required
M1	20	20	Not Applicable
M2	18	38	Not Insisted
M3	19	57	12 credits from M1
M4	16	73	Not Insisted

1 hour of Lecture per week: 1 credit
2 hours of Mini project per week: 1 credit
1½ hours of Laboratory/ Project per week: 1 credit

4. Course Coding Scheme

Structure of a Course Number: ABDCPYTN(R)

#	Character in the Course code	Description	Remarks
1.	**AB**	Year of curriculum introduction	The last two digits of the year in which the curriculum is introduced
2.	**DC**	Department offering the course/ Category of the course (The order for inclusion in the course code : MA, AC Dept code)	CE**:** Department of CE CS: Department of CSE EE: Department of EEE EC: Department of ECE ME: Department of ME MA: Mathematics course AC: Audit Course offered by the department/ from outside the department
3.	**P**	**P**rogram number (In the chronological order of commencement of the M.Tech program in the department)	1: Course offered for the 1st M.Tech program in the department 2: Course offered for the 2nd M.Tech program in the department 0: Course offered for more than one program/ courses offered from outside the department.
4.	**Y**	**Y**ear/ Level	6: First/ Second semester M.Tech 7: Third/ Fourth semester M.Tech

5.	**T**	**T**ypes of Course	1: Program Core course/ AC 2,3: Program Elective course 4,5: Skill/Ability Enhancement Course 6: Industry Elective course 7: Mini Project course 8: Project course/ Internship course 9: Laboratory course
6.	**N**	**N**umber for the course	A – Z shall be used for unique identification of the Course.
7.	**(R)**	**R**evision number	(1): 1st revision (2): 2nd revision etc. :**Blank** at the introduction of the course

Sample Course code	Description
26CE161B	A program core course in the 2026 curriculum offered in the first/ second semester of M.Tech in SE, introduced for the first time
26CS162M(2)	A program elective course in the 2026 curriculum offered in the first/ second semester of M.Tech in CSE, in its second revision

5. Assessment Pattern

a) Program Core Course/ Program Core Course with practical component

A Program Core Course can be conducted as a theory course or a theory course along with its related laboratory experiments. A Program Core Course conducted as a theory course along with its related laboratory experiments comes under the course type PCCP. A PCC/ PCCP is evaluated out of 100 marks; 50 marks for Continuous internal assessment (CIA) and 50 marks for End semester evaluation (ESE).

Evaluation shall include application, analysis, and design based questions for both CIA and ESE.

Continuous Internal Assessment (CIA): 50 marks

Micro project/Laboratory/ Course based project: 30 marks

Course based task/ Seminar/Quiz: 10 marks

Continuous Assessment Test (CAT), 1 No: 10 marks
(CAT shall include minimum 60% of the syllabus)

Micro project/ Course based project shall be done individually. Group projects are not permitted.

End Semester Examination (ESE): 50 marks

ESE will be conducted by the Controller of Examinations (CoE). Duration of the examination shall be 180 minutes.

The question paper will contain 7 questions with minimum one question from each module, having 10 marks for each question. A question can have sub parts. Students shall answer any five questions.

The questions shall be useful in the testing of knowledge, skills, comprehension, application, analysis, synthesis, overall achievement and maturity of the students in a course, through questions relating to theoretical/ practical knowledge, derivations, problem solving and quantitative evaluation.

b) Program Elective Course/ Program Elective Course with practical component

A Program Elective Course can be conducted as a theory course or a theory course along with its related laboratory experiments. A Program Elective Course conducted as a theory course along with its related laboratory experiments comes under the course type PECP. A PEC/ PECP is evaluated out of 100 marks; 50 marks for CIA and 50 marks for ESE.

Evaluation shall include application, analysis, and design based questions for both CIA and ESE.

Continuous Internal Assessment: 50 marks

Preparing a review article based on peer reviewed original publications (Minimum 10 publications shall be referred)/ Micro project/Laboratory :	30 marks
Course based task/ Seminar/ Data collection and interpretation:	10 marks
Continuous Assessment Test (CAT), 1 No: (CAT shall include minimum 60% of the syllabus)	10 marks

End Semester Examination: 50 marks

The ESE will be conducted by the CoE. Duration of the examination shall be 180 minutes.

The question paper will contain 7 questions with minimum one question from each module, having 10 marks for each question. A question can have two or more sub parts. Students shall answer any five questions.

The questions shall be useful in the testing of knowledge, skills, comprehension, application, analysis, synthesis, testing of overall achievement and maturity of the students in a course, through questions relating to theoretical/ practical knowledge, derivations, problem solving and quantitative evaluation.

c) Audit Course (Research Methodology and IPR)

An audit course is evaluated out of 100 marks; 50 marks for CIA and 50 marks for ESE.

Continuous Internal Evaluation: 50 marks

Course based task:	20 marks
Seminar/Quiz:	20 marks

Continuous assessment Test (CAT), 1 No: 10 marks
(CAT shall include minimum 60% of the syllabus)

End Semester Examination: 50 marks

The ESE will be conducted by the CoE. Duration of the examination shall be 180 minutes.

The question paper will contain 7 questions with minimum one question from each module, having 10 marks for each question. Students shall answer any five questions.

d) Internship

Internships are educational and career development opportunities, providing practical experience in a field or discipline. They are structured, short-term, supervised placements often focused around particular tasks or projects with defined timescales. A student has the opportunity to do internship for one semester either in M3 or in M4. Such students will carry out a Project work in the other semester.

An internship may be compensated or non-compensated by the organization providing the internship. Internship has to be meaningful and mutually beneficial to the intern and the organization. It is important that the objectives of the internship program are clearly defined and understood. Internship offers the students an opportunity to

(i) Gain hands-on industrial or organizational exposure
(ii) Integrate the knowledge and skills acquired through his/her coursework
(iii) Interact with professionals and other interns, and
(iv) Improve the presentation, writing, and communication skills of interns.

Internship often acts as a gateway for final placement for many students.

A student shall carry out the Internship at an Industry/ Research Organization or at another institute of higher learning and repute (Academia). The students must select the organization for doing Internship on their own, with prior approval from the respective PG Programme coordinator. Every student shall be assigned a Faculty supervisor at the beginning of his/her Internship. The training shall be related to their specialization. The internship must be carried out for duration of four to five months, during the third semester or fourth semester. On completion of the Internship course, the student is expected to be able to develop skills in facing and solving the problems experienced in the related field.

Objectives

- Exposure to the industrial environment, which cannot be simulated in the class room and hence creating competent professionals for the industry.
- Provide possible opportunities to learn understand and sharpen the real time technical/ managerial skills required at the job.
- Exposure to the current technological developments relevant to the subject area of training.
- Create conducive conditions with quest for knowledge and its applicability on the job.
- Understand the social, environmental, economic and administrative considerations that influence the working environment.

- Exposure to the engineer's responsibilities and ethics.

Benefits of Internship to Students

- An opportunity to get hired by the Industry/ organization.
- Practical experience in an organizational setting and Industry environment.
- An opportunity to see how the theoretical aspects learned in classes are integrated into the practical world. On-floor experience provides much more professional experience which is often worth more than classroom teaching.
- Helps the intern to decide if the industry and the profession is the best career option to pursue.
- Opportunity to learn new skills and supplement knowledge.
- Opportunity to practice communication and teamwork skills.
- Opportunity to learn strategies like time management, multi-tasking etc in an industrial setup.
- Makes a valuable addition to their resume
- Enhances their candidacy for higher education/placement.
- Creating network and social circle and developing relationships with industry people.
- Provides opportunity to evaluate the organization before committing to a fulltime position.

Benefits of Internship to the Institute

- Build industry academia relations.
- Makes the placement process easier.
- Improve institutional credibility and branding.
- Curriculum revision can be made based on feedback from Industry/ students.
- Improvement in teaching learning process.

Benefits of Internship to the Industry

- Availability of ready to contribute candidates for employment.
- Students bring new perspectives to problem solving.
- Visibility of the organization is increased on campus.
- Quality candidate's availability for temporary or seasonal positions and projects.
- Freedom for industrial staff to pursue more creative projects.
- Availability of flexible, cost-effective workforce not requiring a long-term employer commitment.
- Cost-effective way to recruit and evaluate potential employees.
- Enhancement of employer's image in the community by contributing to the educational enterprise.

Types of Internships

- Industry Internship with/ without Stipend
- Government / PSU Internship (BARC/ Railway/ ISRO etc.)
- Internship with prominent education/ Research Institutes

- Internship with Incubation centers/ Start-ups

Guidelines

- All the students need to go for internship for minimum duration of four moths and a maximum duration of five months.
- Students can take mini projects, assignments, case studies by discussing it with concerned authority from industry and can work on it during internship.
- All students should compulsorily follow the rules and regulations of the industry.
- Every student should take prior permissions from concerned industrial authority if they want to use any drawings, photographs or any other document from industry.
- Student should follow all ethical practices and Standard Operating Procedure (SOP) of the industry.
- Students must take necessary health and safety precautions as laid by the industry.
- Student should contact his /her Guide/Supervisor from the College on a weekly basis to communicate the progress.
- Each student has to maintain a diary/log book
- After completion of internship, students are required to submit
  - ✓ Report of work done
  - ✓ Copy of Internship certificate
  - ✓ Feedback from internship mentor in the place of internship
  - ✓ Proof of stipend (in case of paid internship).

Evaluation of Internship

Internship will be evaluated out of 100 marks for CIA.

Student's diary/ Daily Log:	25 Marks
Evaluation done by the Industry:	25 Marks
Internship Report:	25 Marks
Comprehensive Viva Voce:	25 Marks

Student's Diary/ Daily Log:

The main purpose of writing daily diary is to cultivate the habit of documenting and to encourage the students to search for details. It develops the students' thought process and reasoning abilities. The students should record in the daily training diary the day to day account of the observations, impressions, information gathered and suggestions given, if any. It should contain the sketches and drawings related to the observations made by the students. Student's diary must be signed each day by the supervisor/ in charge of the section where the student has been working.

Format of Student's Diary

Name of the Organization/Section:

Name and Address of the Section Head:

Name and Address of the Supervisor:

Name and address of the student:

Internship Duration: From ..... To .....

Brief description about the nature of internship:

Day	Brief write up about the Activities carried out: Such as design, sketches, result observed, issues identified, data recorded, etc.
1	
2	
3	

Signature of Industry supervisor Signature of Head/ HR Manager

Office Seal

Format of Attendance Sheet

Name of the Organization/ Section:

Name and Address of the Section Head:

Name and Address of the Supervisor:

Name and address of the student:

Internship Duration: From ..... To .....

Month & Year	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	…
Month & Year																			
Month & Year																			

Signature of Industry supervisor Signature of Head/ HR Manager

Office Seal

Note:

- Student's Diary shall be submitted by the students along with attendance record and an evaluation sheet duly signed and stamped by the industry to the Institute immediately after the completion of the training.
- Attendance Sheet should remain affixed in daily training diary. Do not remove or tear it off.
- Student shall sign in the attendance column. Do not mark 'P'.
- Holidays should be marked in red ink in the attendance column. Absence should be marked as 'A' in red ink.

Student's diary will be evaluated on the basis of the following criteria:

- Regularity in maintenance of the diary
- Adequacy and quality of information recorded
- Drawings, design, sketches and data recorded
- Thought process and recording techniques used
- Organization of the information.

Format for Evaluation of Intern by industry

Student Name : Date:

Supervisor Name : Designation:

Company/ Organization :

Internship Address:

Dates of Internship: From_____ To_____

Please evaluate intern by indicating the frequency with which you observed the following parameters:

Parameters/ Marks	Needs improvement (0 – 0.25 marks)	Satisfactory (0.25 – 0.5 marks)	Good (0.75 marks)	Excellent (1 mark)
Behavior				
Performs in a dependable manner				

Cooperates with coworkers and supervisor				
Shows interest in work				
Learns quickly				
Shows initiative				
Produces high quality work				
Accepts responsibility				
Accepts criticism				
Demonstrates organizational skills				
Uses technical knowledge and expertise				
Shows good judgment				
Demonstrates creativity/ originality				
Analyzes problems effectively				
Is self reliant				
Communicates well				
Writes effectively				
Has a professional attitude				
Gives professional appearance				
Is punctual				
Uses time effectively				

Overall performance of student Intern (Please Tick one):

Needs improvement (0.50 marks)
Satisfactory (1.0 mark)
Good (1.5 marks)
Excellent (2.0 marks)

Additional comments, if any (2 marks):

Signature of Industry Supervisor *Signature of Section Head/HR Manager*

Office Seal

Internship Report:

After completion of the internship, the student should prepare a comprehensive report to indicate what he has observed and learnt in the training period and should be submitted to the faculty Supervisor. The student should prepare the final report on the assigned topics. Diary/ daily log will also help to a great extent in writing the report since much of the information has already been incorporated by the student into the diary. The training report should be signed by the Internship supervisor, PG Programme Coordinator and Faculty mentor.

The Internship report will be evaluated on the basis of following criteria:

- Originality
- Adequacy and purposeful write-up
- Organization, format, drawings, sketches, style, language etc.
- Variety and relevance of learning experience
- Practical applications, relationships with basic theory and concepts taught in the course

Comprehensive Viva Voce:

Viva Voce will be done by a committee comprising Faculty Supervisor, PG Programme Coordinator, and one faculty member from a sister department. This committee shall evaluate the internship report also.

e) Laboratory Courses

Laboratory courses will have only continuous internal assessment and carries 100 marks. Final assessment shall be done by two examiners; one examiner will be a senior faculty from the same department.

Continuous internal assessment: 100 marks

Performance in regular laboratory experiments:	70 marks
Final assessment/ laboratory test:	30 marks

f) Industry Elective Course

Engineering students frequently aspire to work in areas and domains that are key topics in the industry. There are concerns by recruiters that skill sets of engineering students do not match with the Industry requirements, especially in the field of latest topics.

Industry knowledge aids in the bridge building process between academic institutions and industry. It also aids students in expanding their knowledge and innovating by allowing them to create something new. Core engineering courses provide students with a strong foundation. Evolving technology necessitates new methods and approaches to progress, prosperity, and the inculcation of problem-solving techniques. Industry knowledge will enable the students to deal with any scenario more effectively, thus fulfilling the current industry demands.

Rapid technological advancements have resulted in a massive revival in the way engineering works in the industry. Projects necessitate the integration of knowledge and abilities from a diverse variety of engineering specialties, with the barriers between them becoming increasingly blurred.

Students can choose courses offered by Industries that cover a wide range of highly relevant topics such as artificial intelligence, internet of things, big data, automation, and other relatable courses.

IEC will be evaluated out of 100 marks; 50 marks for CIA and 50 marks for ESE.

Continuous internal assessment: 50 marks

The continuous internal evaluation will be done by the expert in the Industry handling the course, and the coordinator from the college.

Micro project/ Course based project:	30 marks
Course based task/Seminar/Quiz:	10 marks
Continuous assessment Test (CAT), 1 No:	10 marks

(CAT shall include minimum 60% of the syllabus)

End Semester Examination: 50 marks

ESE will be conducted by the CoE using the question paper provided by the industry. Duration of the examination shall be 180 minutes.

The question paper will contain 7 questions with minimum one question from each module, having 10 marks for each question. Students shall answer any five questions. Evaluation of the answer scripts will be done by the expert in the Industry handling the course or the coordinator from the college under the expert's guidance.

g) Skill/ Ability Enhancement Course

SAEC are online MOOC of 12 weeks duration and shall be considered only if it is conducted by the agencies namely AICTE/ NPTEL/ SWAYAM/ NITTTR. The course should have a proctored/ offline end semester examination. The students can do the SAEC credited in M3 according to their convenience from their first semester, but shall complete it by third semester. The list of courses is to be approved by the concerned Board of studies. A course may be approved only if at least 70% of the course content matches with the area/ stream of study. The course shall not be considered if its content has more than 50% of overlap with a core/ elective course in the concerned discipline or with an open elective.

A credit of 3 and a grade point of 10 will be awarded to all students whoever successfully completes the SAEC credited in M3. Marks/ GPA awarded to the other SAEC shall be used for SGPA/CGPA computation.

h) Mini Project

Mini project helps to strengthen the understanding of the fundamentals through application of theoretical concepts, and to boost their skills and widen the horizon of thinking. The ultimate aim of an

engineering student is to resolve a problem by applying theoretical knowledge. Doing more projects enhances problem solving skills. The Mini project ensures preparedness of students to undertake their project work in M3 and M4. Students should identify a topic of interest in consultation with his/her PG Programme Coordinator. They should demonstrate the novelty of the project through the results and outputs. This mini project work is assessed in three evaluations, two interim evaluations and a final evaluation. The evaluations will be done by a committee comprising of Project Coordinator, Two senior faculty members in the department, and the student's Project Supervisor

Final evaluation will be conducted only if the Interim project report approved by the student's supervisor is submitted. The Plagiarism level in the report should be ≤ 25%, assessed based on the overall similarity index given by Turnitin licensed to the College.

Mini Project will be evaluated out of 100 marks under CIA, and has no ESE.

a) First evaluation:

Evaluation committee: 20 marks

Literature Survey:	7 marks
Objectives and Methodology:	7 marks
Clarity of presentation:	6 marks

b) Second evaluation:

Evaluation committee: 20 marks

Design:	7 marks
Implementation plan:	5 marks
Expected results:	8 marks

c) Final evaluation: 60 marks

a) Supervisor/ Guide: 10 marks

Log book and Regularity:	5 marks
Overall evaluation of the project work:	5 marks

b) Evaluation committee: 50 marks

Demonstration of functionality/ specifications:	20 marks
Level of completion:	5 marks
Clarity of presentation:	5 marks
Knowledge on the project work:	5 marks
Interim project report:	
Technical content:	5 marks

Adequacy of references:	5 marks
Templates followed:	5 marks

i) Project

The students must carry out the project work either in the college or in any CSIR/ industrial R&D organization/ any other reputed Institute which have facilities to carry out project work in the proposed area.

Project work outside the College:

For doing project work outside the college, the following conditions are to be met:

- They have successfully completed the course work prescribed in the approved curriculum up to the second semester.
- The student has to get prior approval from the DLAC.
- Students availing this facility should continue as regular students of the College.
- Facilities required for doing the project work shall be available in the Organization/ Industry. A certificate stating the time period for which the facilities shall be made available to the student, issued by a competent authority from the Organization/ Industry shall be submitted by the student along with the application.
- The student should have an external as well as an internal supervisor. The internal supervisor should belong to the college and the external supervisor shall be a Scientist or Engineer from the Institution/ Industry/ R&D organization with which the student proposes to do his project work. The external supervisor shall be with a minimum Post graduate degree in the related area.
- The MOOC must be completed as per the curriculum requirements:
- The student has to furnish his/her monthly progress as well as attendance report signed by the external supervisor and submit the same to the concerned internal supervisor.
- The external supervisor is to be preferably present during all the stages of evaluation of the project.

Internship leading to Project:

The students who, after completion of 6 to 8 weeks internship at some reputed organization, are allowed to continue their work as project for the third and fourth semester can do so after getting approval from the DLAC. Such students shall make a brief presentation regarding the work they propose to carry out before the DLAC for a detailed scrutiny and to resolve its suitability for accepting it as an M.Tech project. Once accepted, they will be permitted to complete their project in that organization (where they have successfully completed their internship) during their third semester and fourth semester.

Project as part of Employment:

Students may be permitted to discontinue the programme and take up a job, provided they have completed all the courses till the second semester (FE status students are not permitted) prescribed in the approved curriculum. The project work can be done during a later period either in the organization

where they work if it has R & D facility, or in the College. Such students shall submit an application with details (copy of employment offer, and the plan of completion of their project) to the Dean (PGSR) through the HoD. When the student plans to do the project work in the organization with R & D facility where they are employed, they shall submit a separate application with the following details:

- Name of R & D Organization/Industry
- Name and designation of an external supervisor from the proposed organization/ industry (a scientist or engineer with a minimum post graduate degree in the related area), along with his profile, and consent letter.
- Name and designation of a faculty member of the College as internal supervisor, and his/her consent letter.
- Letter from the competent authority from the Organization/ Industry granting permission to do the project work.
- Details of the proposed project work along with the work plan for completion of the project.

DLAC will scrutinize the proposal and forward to CLAC for approval.

When a student does his project work along with the job in the organization (with R & D facility) where they are employed, the project work shall be completed in four semesters (two semesters of dissertation work along with the job may be considered as equivalent to one semester of dissertation work at the college). He should complete the M. Tech programme within four years from the date of admission as per the regulation. Extensions may be granted based on requests from the student and recommendation of the supervisors. Method of assessment of the project will be the same as in the case of regular students.

Evaluation of Project (Phase I) in M3

Project (Phase I) will be evaluated out of 100 marks under CIA, and has no ESE. There will be two evaluations (first evaluation and final evaluation). The assessment shall be done by the student's Project Supervisor, and a committee comprising of Project Coordinator, two senior faculty members in the department, and the student's Project Supervisor. Project Coordinator shall enter the marks in the CoE portal.

Final evaluation will be conducted only if the student has submitted the Interim project report approved by the Supervisor, and Plagiarism level in the Interim project report is ≤25%.

1) **First evaluation: 30 marks**

Project Supervisor: 10 marks

i. Progress of work: 5 marks
(Literature Survey, Objectives, Methodology)

ii. Log book and Regularity: 5 marks

Evaluation committee: 20 marks

i. Topic, Objectives: 5 marks

ii. Methodology and Implementation plan for the work in M3: 10 marks

iii. Clarity in presentation: 5 marks

2) Final evaluation: 70 marks

Project Supervisor: 25 marks

i. Progress of work: 15 marks

ii. Log book and Regularity: 5 marks

iii. Interim project report: 5 marks

Evaluation committee: 45 marks

i. Demonstration of work completed: 15 marks

ii. Presentation and Viva voce: 10 marks

iii. Implementation plan of work in M4: 5 marks

iv. Interim project report: 15 marks

Technical content: 10 marks

Adequacy of references and Templates followed: 5 marks

Evaluation of Project (Phase II) in M4/ Project in M3 or M4

The evaluation of Project (Phase II) has CIA for 100 marks, and ESE for 100 marks. The continuous internal assessment is done under two evaluations (first evaluation and final evaluation), by the student's Project Supervisor, and a committee comprising of Project Coordinator, two senior faculty members in the department, and the student's Project Supervisor. Project Coordinator shall enter the marks in the CoE portal.

Final evaluation will be conducted only if the student has submitted the project report approved by the Supervisor, and Plagiarism level in the project report is ≤25%.

Continuous internal assessment: 100 marks

1) First evaluation: 40 marks

Project Supervisor: 15 marks

i. Progress of work: 10 marks
(Experimentation and results)

ii. Log book and Regularity: 5 marks

Evaluation committee: 25 marks

i. Demonstration of work completed: 15 marks

ii. Presentation and Viva: 10 marks

2) Final evaluation: 60 marks

Project Supervisor: 15 marks

i. Progress of work: 10 marks
(Quality and quantum of work)

ii. Project report: 5 marks

Evaluation committee: 45 marks

i. Demonstration of work completed: 10 marks

ii. Presentation and viva: 10 marks

iii. Project report: 10 marks
Technical content: 5 marks
Adequacy of references: 5 marks

iv. Paper publication: 15 marks
(Published/accepted for publication in a journal/conference)

End semester examination (Viva-voce examination): 100 marks

The ESE will be done by a committee that comprises of the Project Coordinator, an external expert (from industry or research/academic institute), and the student's Project Supervisor

Each department must submit a panel of external experts to Dean (PGSR), as per the academic calendar. The minimum qualification requirement for an external examiner is M.Tech. The number of experts to be submitted is one more than number of students divided by 6 (rounded to the next integer). Honorarium for the external expert will be as fixed by the College.

The Project coordinator will enter the ESE marks in the CoE portal.

Marks Distribution for Viva-voce examination

i. Innovation & originality: 15 marks
(Introduction, Recent and related literature, Scope of the work, Objectives)

ii. Implementation and execution: 20 marks
(Methodology and work plan, Results and discussions, Quality of work done)

iii. Project Documentation: 20 marks
(Introduction, Problem Statement , Literature review, Methodology, Results and discussions, Conclusions, Future work, References)

iv. Presentation and Defense: 40 marks
(Clarity and effectiveness of presentation, Ability to explain the project objectives, Methodology and Findings, Handling questions and providing satisfactory answers)

v. Publication: 5 marks
(Published/accepted for publication in a journal/conference)

6. Teaching Assistanceship (TA)

All M.Tech students irrespective of their category of admission, shall undertake TA duties for a minimum duration as per the curriculum. Being a TA, the student will get an excellent opportunity to improve their expertise in the technical content of the course, enhance communication skills, obtain hands-on experience in handling the experiments in the laboratory and improve peer interactions.

Typical responsibilities of a TA include the following:

a) Facilitate a discussion session or tutorial for a theory course
b) Facilitate to assist the students for a laboratory course
c) Serve as a mentor for students, and act as the course web-master

TAs may be required to attend the instructor's lecture regularly. A TA shall not be employed as a substitute instructor, where the effect is to relieve the instructor of his/her teaching responsibilities. Students who are doing their project work outside the college are not required to do TA work during their third semester and fourth semester.

Handling a tutorial session:

(i) The TA must meet the instructor and understand his/her responsibilities well in advance, attend the lectures of the course for which the TA is a tutor, work out the solutions for all the tutorial problems self, approach the teacher if there is any discrepancy or need help in solving the tutorial problems, use reference text books, be innovative and express everything in English only.

(ii) The TA must try to lead the students to the correct solutions by providing appropriate hints rather than solving the entire problem by self, encourage questions from the students, lead the group to a discussion based on their questions, plan to ask them some questions, be friendly and open with the students, simultaneously being firm with them.

(iii) The TA must keep track of the progress of each student in his/her group, give a periodic feedback to the student about their progress, issue warnings if the student is consistently under-performing, report to the instructor if a particular student is consistently underperforming, pay special attention to slow-learners and be open to the feedback and comments from the students and faculty.

(iv) After the tutorial session the TA may be required to grade the tutorials/assignments/tests. Make sure to work out the solutions to the questions self, work out possible alternate solutions to the same question, and discuss the marking scheme with the instructor.

(v) Consult the instructor and ensure impartial approach to the students in their grading. Follow basic ethics.

Handling a laboratory Session:

(i) Meet the faculty – in- charge a few days in advance of the actual lab class and get the details of the experiment, get clarifications regarding all aspects of the experiment and the expectations, prepare by reading about the theoretical background of the experiment, know the physical concepts involved in the experiment, go to the laboratory and check out the condition of the equipment/instrumentation, perform the laboratory experiment at least once before the actual laboratory class, familiarize with safety/ security aspects of the experiment/equipment/laboratory, prepare an instruction sheet for the experiment in consultation with the faculty, and keep sufficient copies ready for distribution to students for their reference.

(ii) Verify condition of the equipment/set up about 30 minutes before the students arrive in the class and be ready with the hand outs, make brief introductory remarks about the experiment, its importance, its relevance to the theory they have studied in the class, ask the students suitable questions to know their level of preparation for the experiment, discuss how to interpret results, ask them comment on the results.

(iii) Correct/evaluate/grade the submitted reports after receiving suitable instructions from the faculty in charge, continue to interact with students if they have any clarifications regarding any aspect of the laboratory session, including course grading, Carefully observe instrument and human safety in laboratory class. Prepare simple questions for short oral quizzing during explanation of experiments enables active participation of students, facilitate attention, and provide feedback and formative assessment.

7. Points to Remember

1) Each department must conduct an awareness programme to all M.Tech students on day 1 regarding the curriculum and the regulation. There will be a common induction programme on 'Universal Human Values' for students of all M.Tech streams during the first three days of Semester 1.
2) The departments must publish the list of MOOCs suitable to their programmes.
3) While choosing the Industry and the Industry electives, it should be ensured that the programme is relevant and updated in that discipline. The Industry expert handling the elective shall be a postgraduate degree holder. The evaluation procedure shall also be clearly explained to them.
4) The departments may invite the Industries/research organizations during first semester and inform them about the internship that the students can undergo in M3/ M4. The possibility of doing their project work at the Industry shall also be explored. They should also be made aware about the evaluation procedure of the internship. They may also be informed that it is possible to continue internship provided it leads to their project work. Proposals may be collected from them for allotting to students according to their fields of interest.
5) Make sure that all internal assessments and the end semester examinations to be conducted by the college are carried out as per the assessment procedure listed in the curriculum. Any dilution from the prescribed procedure shall be viewed seriously.

6) Teaching assistance shall be assigned to all students as per the curriculum. However, a TA shall not be employed as a substitute instructor, where the effect is to relieve the instructor of his or her teaching responsibilities (strictly prohibited by University Policy).
7) The possible TA responsibilities include the following:
    a) Facilitate a discussion session or tutorial for a theory course
    b) Facilitate to assist the students for a laboratory course
    c) Serve as a mentor for students, and act as the course web-master

PROGRAM CORE COURSES (PCC)

Course Code	Course Name	Category Code	L	T	P	Credit	Year of Introduction
26MA061A	MATHEMATICAL FOUNDATIONS OF COMPUTING SYSTEMS	PCC	3	1	0	4	2026

i) COURSE OBJECTIVES

This course is to build a strong mathematical foundation and enable the application of linear algebra, probability, graph theory, and group theory to solve problems in computing systems and networks.

ii) COURSE OUTCOMES

After the completion of the course, the student will be able to:

CO1	Apply standard proof techniques to reason rigorously about vector spaces, linear independence, basis, and dimension in linear algebra.	Apply
CO2	Apply eigenvalue methods and matrix decompositions, including SVD, to analyze linear transformations and data-driven applications.	Apply
CO3	Apply random variables and probability distributions to practical situations.	Apply
CO4	Apply graph theory concepts and algorithms to solve network and computing problems	Apply
CO5	Apply group theory concepts to solve problems in graphs and computing systems.	Apply

iii) SYLLABUS

This syllabus builds a strong mathematical foundation for computing by covering proof techniques, linear algebra, probability, graph theory, and group theory, with emphasis on rigorous reasoning, structural analysis, and applications to data science, networks, and computing systems.

iv) REFERENCES

1) J. P. Tremblay, R. Manohar, “Discrete Mathematical Structures with Application to Computer Science”, Tata McGraw Hill, 2000.

2) Kenneth H. Rosen, “Discrete Mathematics and its Applications”, 7/e, McGraw Hill Inc, 2011.

3) Robert V. Hogg, Elliot A. Tanis, Meda J. M. Rao, “Probability and Statistical Inference”, 7/e,, Pearson Education India, 2006.

4)J. Truss, “Discrete Mathematics for Computer Scientists”, 2/e, Addison Wesley, 1999. Bernard Kolman, Robert C Busby, SharonKutler Ross, “Discrete Mathematical Structures”, 2/e, Prentice-Hall India Private Limited, 1996.

5) Ralph P. Grimaldi , B.V Ramana, Discrete and Combinatorial Mathematics, Fifth Edition, Pearson 2016

6) Doulgas B West, Introduction to Graph Theory, Prentice Hall India Ltd.

v) COURSE PLAN

Module	Contents	No. of hours
I	Proof Techniques and Linear Algebra Foundations :Techniques of mathematical proof: Direct proof, proof by contraposition, proof by contradiction, proof by cases, Mathematical reasoning in linear algebra, Vector spaces and subspaces, Linear combinations, span, basis, and dimension, Linear independence and dependence.	10
II	Linear Transformations, Eigenvalues, and Matrix Decompositions: Linear transformations and matrix representation; rank–nullity theorem; eigenvalues and eigenvectors; symmetric matrices and orthogonality; inner product spaces and matrix norms; Singular Value Decomposition (SVD); applications in data compression, PCA, signal processing, and machine learning.	14
III	Probability and Random Variables : Probability axioms; conditional probability, independence, Bayes' theorem; discrete and continuous random variables; expectation and variance; Binomial, Poisson, Uniform, Exponential, and Normal distributions.	14
IV	Graph Theory and Algorithms : Graphs and terminology, Degree sequence, paths, cycles, and connectivity, Euler and Hamiltonian graphs, Planar graphs and Euler's formula (proof), Graph coloring and chromatic number, Warshall's algorithm, Applications to networks and computing systems.	11
V	Group Theory and Symmetry in Computing : Groups and subgroups, Cyclic and permutation groups, Cosets and Lagrange's theorem, Group homomorphisms and isomorphism theorems, Applications of group theory to symmetry in graphs and computing. .	11
	Total hours	**60**

Course Code	Course Name	Category	L	T	P	Credit	Year of Introduction
26CS161A	**ADVANCED COMPUTER NETWORKS**	**PCCP**	**3**	**0**	**2**	**4**	**2026**

i)COURSE OBJECTIVES

This course imparts a deeper understanding of protocols, Switching, VPN, quality of service and congestion management. It also helps to analyze the issues of transmitting real time data. Also helps to identify the technologies that can transmit data efficiently.

ii) COURSE OUTCOMES

After the completion of the course, the student will be able to:

CO1	Analyze existing techniques for developing new technologies for transmitting data in real time without congestion.	Analyze
CO2	Apply the knowledge to design computer networks with optimal routing.	Apply
CO3	Interpret and exemplify current QoS architectures and mechanisms, and the QoS support challenges in future networks	Analyze
CO4	Compare and choose appropriate and advanced techniques to build the computer network	Analyze

iii)SYLLABUS

Network Architecture - Internet Protocol - Packet switching- Internetworking devices-– Switching basics- Routers-Path Vectors and policies-Congestion Management - -Quality of Service - Peer to Peer Networks - Virtual Private Networks and tunnels –Network management-installation and maintenance

iv)REFERENCES

1. Larry L. Peterson, Bruce S. Davie, “Computer Networks – A Systems Approach”, Elsevier Fourth Edition, 2008.
2. William Stallings, “SNMP, SNMPv2, SNMPv3, and RMON 1 and 2,” AddisonWesley, 2005.

3. Douglas Comer, “Automated network management systems current and future capabilities,” Pearson Prentice-Hall, 2007
4. Jim Guichard, Ivan Pepelnjak, " MPLS and VPN Architectures: A Practical Guide to Understanding, Designing and Deploying MPLS and MPLS-Enabled VPNs," Cisco Press, October 2000
5. Packet Guide to Routing and Switching by Bruce Hartpence Released August 2011 Publisher(s): O'Reilly Media, Inc.

v) COURSE PLAN

Module	Contents	No. of hours
I	Network Management: The SNMP communication Model – The SNMP Architecture, Administrative Model, SNMP Specifications, SNMP Operations, SNMP MIB Group, Functional Model. Network Installation and Maintenance; Network and System Management, Network Management System platform, Future of Network Management. Emerging network types: Data Center, Online social networks (OSN), Software Defined Networking, Network virtualization, Delay Tolerant Network (DTN), Vehicular Ad-hoc Network, VoIP, Body area networks (BDN) and case studies.	9
	• Build an ISP. It should include a DNS, DHCP, Leased Line, PPP, Webserver, Internet Backbone with appropriate routing protocols. Experiments may be implemented using real simulators. • Software Defined Network	6
II	Quality of Service: Requirements and parameters of Quality of Service, Integrated Services, Resource Reservation Protocol (RSVP), Differentiated Services. Peer to Peer Networks: Gnutella, BitTorrent. Node Lookup in Peer to Peer Networks. Introduction to Virtual Private Networks- Overlay and Peer-to-Peer VPN ,Major VPN Topologies , MPLS VPN Architecture , MPLS VPN Routing Model ,MPLS VPN Packet Forwarding ,MPLS VPN Spanning more than One AS.	8
	• Familiarization of Wireshark • Detailed Study of protocols in TCP/IP model usingWireshark	6

	• Familiarization of TFsense. •	
III	Path Vectors and Policies - Computing Paths– Routing Information Protocol – OSPF - OSPF – Intermediate System to Intermediate System - Border Gateway Protocol - Multicast Routing - Inter-Gateway Routing Protocol - Inter-Domain Routing Protocol. Congestion Management: Congestion control in Data Networks and Internets, Random Early Detection (RED). TCP congestion control: Additive increase/Multiplicative decrease, Slow start, Fast retransmit and Fast recovery. Congestion-Avoidance Mechanisms – DECbit - Random Early Detection - Source- Based Congestion Avoidance – Tahoe- Reno- and Vegas.	10
	• Develop LAN with WAN. It should employ a firewall to redirect all external traffic. Use CIDR for forming departments. Experiment may be implemented using real Simulators • Build a firewall using IPTables.	6
IV	Switching Basics. Managing the LAN switch as a networking device, basic switch configuration. Spanning Tree protocol (STP). Virtual LANs and frame-tagging. Routing between VLANs. Securing network devices using packet filters and firewall by applying access control lists (ACL). Routers: Router functions, Classification of routers, Features of IP Routers, Filtering, Network Address Translation (NAT). *Experiment: Configuring Routers and Switches*	9
	• InterVLAN Routing • Access Control List	8
V	Network Architecture: Reference models of OSI, TCP/IP, ATM. Protocol implementation issues. Physical address, Logical address. Internet Protocol: Packet Format (IPV4 and IPV6), Features of IPv6, CIDR notation, DHCP. Packet switching: Datagrams, Virtual circuit switching, Fragmentation of IP packets. Cell switching in ATM, Gigabit Networks. Internetworking devices: Repeaters, Hubs, Bridges, LAN switches, Routers and Gateway.	9

	Experiment: Familiarization of Networking Devices	
	• Familiarization of Networking devices and router commands.	4
	Total Hours	**75**

Course Code	Course Name	Category	L	T	P	Credit	Year of Introduction
26CS161B	ADVANCED DATA STRUCTURES AND ALGORITHMS	PCC	3	1	0	4	2026

i) COURSE OVERVIEW

Goal of this course is to introduce different advanced data structures and to analyze and establish correctness of algorithms. Also, this course aims to introduce various classes of algorithms.

ii) COURSE OUTCOMES

After the completion of the course, the student will be able to:

CO1	Apply various advanced data structures efficiently to solve problems.	Apply
CO2	Analyze number theoretic algorithms and string-matching algorithms.	Analyze
CO3	Apply flow networks to solve real world problems.	Apply
CO4	Analyze various classes of algorithms.	Analyze
CO5	Apply geometric algorithms to solve real world problems.	Apply
CO6	Design, prove the correctness and analyze new algorithms.	Analyze

iii) SYLLABUS

Amortized Analysis – aggregate, accounting and potential methods. Advanced data structures: binomial heap, Fibonacci heap, disjoint sets - applications. Number-Theoretic algorithms: maxflow-min-cut theorem, String matching: Overview of Complexity Classes Probabilistic algorithms: Numerical algorithms Las Vegas algorithms, Complexity classes in randomized algorithms – RP, PP, ZPP, BPP. Geometric Algorithms.

iv) REFERENCES:

1. T. H. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein, “Introduction to algorithms”, Prentice-hall of India Private Limited, New Delhi, 2010.
2. Gilles Brassard and Paul Bratley, “Fundamentals of algorithms”, Prentice-hall of India Private Limited, New Delhi, 2001.

3. Rajeev Motwani, Prabhakar Raghavan, “Randomized Algorithms”, Cambridge University Press, 200.

v) COURSE PLAN

Module	Contents	No. of hours
I	Amortized Analysis – aggregate, accounting and potential methods. Advanced data structures: binomial heap, Fibonacci heap, Disjoint sets -Applications.	13
II	Number-Theoretic algorithms: GCD algorithm, Extended Euclid’s Algorithm. Primality testing -Miller-Rabin test. Integer factorization - Pollard Rho heuristic. String matching: Rabin-Karp, Knuth-Morris-Pratt algorithms.	10
III	Network flow algorithms: flow properties, augmenting path, Ford-Fulkerson method, Edmonds-Karp heuristics, Maxflow-mincut theorem. Push-relabel, relabel-to-front algorithms	12
IV	Probabilistic algorithms: Numerical algorithms: Integration, Counting. Monte-Carlo algorithms - verifying matrix multiplication, min-cut in a network. Las Vegas algorithms, selection sort, quick sort, Dixon's factorization. Complexity classes in randomized algorithms – RP, PP, ZPP, BPP	12
V	Geometric Algorithms: Plane sweep technique, role of sweep- line status and event-point schedule, line segment intersection problem. Convex Hull: Graham’s scan algorithm, Jarvis March algorithm. Finding the closest pair of points, proof of correctness. Overview of Complexity classes – P, NP, Co-NP, NP-hard, NP-complete. Space complexity.	13
	Total hours	**60**

Course Code	Course Name	Category	L	T	P	Credit	Year of Introduction
26CS161C	ADVANCED DATABASE MANAGEMENT SYSTEMS	PCC	3	1	0	4	2026

i) COURSE OBJECTIVES

Students will be able to understand the transaction processing and query optimization. The course focuses on the principles of special, temporal and distributed databases.

ii) COURSE OUTCOMES

After the completion of the course the student will be able to:

CO1	Apply relational data model concepts, database constraints, and update operations to design correct and consistent relational database schemas.	Apply
CO2	Design and implement object-oriented and object-relational database applications using object models, ODL, OQL, and extended SQL features.	Apply
CO3	Apply parallel and distributed database architectures, query processing, transaction management, and recovery techniques for large-scale database systems.	Apply
CO4	Utilize data warehousing, OLAP, and data mining techniques to support decision support systems and analytical processing.	Apply
CO5	Apply advanced and emerging database models—including active, temporal, spatial, mobile, multimedia, GIS, genome, and cloud databases—to address modern data management challenges.	Apply

iii) SYLLABUS

Review of relational data model (constraints, schemas, update operations, anomalies, and constraint violations), fundamental object-oriented concepts (encapsulation, inheritance, and polymorphism), object-oriented and object-relational databases (complex objects, ODMG standards, ODL, OQL, SQL object-relational features, nested relational models, and language bindings), parallel and distributed databases (architectures, parallel query processing, distributed transactions, concurrency control, and recovery), data warehousing, OLAP, and data mining (multidimensional models, materialized views, rule mining,

clustering, stream mining, and decision support), and enhanced and emerging data models (temporal, spatial, active, mobile, multimedia, GIS, and genome databases).

iv) REFERENCES

1. R. Elmasri, S.B. Navathe, "Fundamentals of Database Systems", 5/e, Pearson Education/Addison Wesley, 2011.
2. Patrick O'Neil , Elizabeth O'Neil , "Database: Principles, Programming and Performance", 2/e, Morgan Kaufmann, 2011.
3. Thomas Cannolly and Carolyn Begg, "Database Systems, A Practical Approach to Design, Implementation and Management", 3/e, Pearson Education, 2010.
4. Henry F Korth, Abraham Silberschatz, S. Sudharshan, "Database System Concepts", 5/e, Tata McGraw Hill, 2006.
5. C.J. Date, A.Kannan and S. Swamynathan,"An Introduction to Database Systems", 8/e, Pearson Education India, 2006.
6. Joe Fawcett, Danny Ayers, Liam R. E. Quin, Beginning XML, 5/e, John Wiley & Sons, 2012.
7. Grigoris Antoniou. Frank van Harmelen, "A Semantic Web Primer", The MIT Press, Cambridge, Massachusetts, 2003.

v) COURSE PLAN

Module	Contents	No. of hours
I	Review of Relational Data Model and Relational Database Constraints: Relational model concepts; Relational model constraints and relational database schemas; Update operations, anomalies, dealing with constraint violations, Types and violations. Overview of Object-Oriented Concepts – Objects, Basic properties. Advantages, examples, Abstract data types, Encapsulation, class hierarchies, polymorphism, examples.	12
II	Object and Object-Relational Databases: Overview of OOP; Complex objects; Identity, structure etc. Object model of ODMG, Object definition Language ODL; Object Query Language OQL; Conceptual design of Object database. Overview of object relational features of SQL; Object-relational features of Oracle; Implementation and related issues for extended type systems; syntax and demo examples, The nested relational model. Overview of C++ language binding;	12

III	Parallel and Distributed Databases: Architectures for parallel databases; Parallel query evaluation; Parallelizing individual operations; Parallel query optimizations; Introduction to distributed databases; Distributed DBMS architectures; Storing data in a Distributed DBMS; Distributed catalog management; Distributed Query processing; Updating distributed data; Distributed transactions; Distributed Concurrency control and Recovery.	13
IV	Data Warehousing, Decision Support and Data Mining: Introduction to decision support; OLAP, multidimensional model; Window queries in SQL; Finding answers quickly; Implementation techniques for OLAP; Data Warehousing; Views and Decision support, View materialization, Maintaining materialized views. Introduction to Data Mining; Counting co-occurrences; Mining for rules; Tree-structured rules; ROC and CMC Curves; Clustering; Similarity search over sequences; Incremental mining and data streams; Additional data mining tasks.	12
V	Enhanced Data Models for Some Advanced Applications: Active database concepts and triggers; Temporal, Spatial, and Deductive Databases – basic concepts. More recent applications: Mobile databases; Multimedia databases; Geographical Information Systems (GIS); Genome data management; Cloud databases – concepts, architecture, characteristics, scalability, and challenges.	11
	Total hours	**60**

PROGRAM ELECTIVE COURSES (PEC)

Course Code	Course Name	Category	L	T	P	Credit	Year of Introduction
26CS162A	**DATA MINING & WAREHOUSING**	**PEC**	**3**	**0**	**0**	**3**	**2026**

i) COURSE OBJECTIVES

To understand and practice the fundamental and advanced concepts of Data Warehousing and Data Mining for effective data analysis and decision support.

ii) COURSE OUTCOMES

After the completion of the course, the student will be able to:

CO1	Explain the concepts and architecture of Data Warehousing systems	Understand
CO2	Apply data preprocessing techniques used in Data Mining	Apply
CO3	Apply association rule mining techniques on large datasets	Apply
CO4	Apply classification, prediction, and clustering techniques for data analysis	Apply
CO5	Apply data mining techniques on complex data types such as text, web, and multimedia data	Apply

iii) SYLLABUS

Data warehousing concepts and architecture, data warehouse components, data extraction, transformation and loading, metadata and OLAP, data mining functionalities, data preprocessing techniques, association rule mining, classification and prediction methods, clustering techniques, outlier analysis, and mining of spatial, multimedia, text, and web data.

iv) REFERENCES

1. Jiawei Han, Micheline Kamber, and Jian Pei, *Data Mining: Concepts and Techniques*, Third Edition, Elsevier, 2011.
2. Alex Berson and Stephen J. Smith, *Data Warehousing, Data Mining & OLAP*, Tata McGraw-Hill, 2007.
3. K. P. Soman, Shyam Diwakar, and V. Ajay, *Insight into Data Mining: Theory and Practice*, Prentice Hall of India, 2006.
4. G. K. Gupta, *Introduction to Data Mining with Case Studies*, Prentice Hall of India, 2006.

5. Pang-Ning Tan, Michael Steinbach, and Vipin Kumar, *Introduction to Data Mining*, Pearson Education, 2007.

v) COURSE PLAN

Module	Contents	No. of hours
I	Data Warehousing and Business Analysis Data warehousing components. Building a data warehouse. Data warehouse architecture. Database schemas for decision support. Data extraction, cleanup, and transformation tools. Metadata management. Reporting and query tools. Applications of data warehousing. Online Analytical Processing (OLAP). OLAP operations and multidimensional data analysis.	9
II	Data Mining and Association Rule Mining Data mining concepts and functionalities. Data preprocessing techniques – data cleaning, data integration and transformation, data reduction, data discretization, and concept hierarchy generation. Architecture of a typical data mining system. Classification of data mining systems. Association rule mining – efficient and scalable frequent itemset mining methods, mining different kinds of association rules, correlation analysis, and constraint-based association mining.	9
III	Classification and Prediction Issues related to classification and prediction. Decision tree classification. Bayesian classification. Rule-based classification. Classification using back propagation. Support Vector Machines. Associative classification. Lazy learners and other classification methods. Prediction techniques. Accuracy and error measures. Evaluating the performance of classifiers and predictors. Ensemble methods and model selection.	9
IV	Cluster Analysis and Outlier Detection Types of data in cluster analysis. Categorization of clustering methods. Partitioning methods. Hierarchical clustering methods. Density-based clustering methods. Grid-based clustering methods. Model-based	9

	clustering methods. Clustering high-dimensional data. Constraint-based cluster analysis. Outlier analysis.	
V	Mining Complex Data Types Multidimensional analysis and descriptive mining of complex data objects. Spatial data mining. Multimedia data mining. Text mining. Mining the World Wide Web.	9
	Total hours	**45**

Course Code	Course Name	Category	L	T	P	Credit	Year of Introduction
26CS162B	SOFT COMPUTING TECHNIQUES	PEC	3	0	0	3	2026

i) COURSE OBJECTIVES

This course introduces learners to the fundamental concepts of Soft Computing techniques and their practical applications. It covers key topics such as Artificial Neural Networks, Fuzzy Logic, Genetic Algorithms, and Multi-Objective Optimization methods. Through this course, students gain the skills to design algorithms and develop solutions for a variety of real-world problems.

ii) COURSE OUTCOMES

After the completion of the course, the student will be able to:

CO1	Explain soft computing techniques and the basic models of Artificial Neural Network	Understand
CO2	Utilize neural network techniques to analyze and solve computational problems.	Apply
CO3	Apply the concepts of fuzzy set theory, membership functions, fuzzy relations, fuzzy propositions, and fuzzy inference systems to analyze, and solve computational problems.	Apply
CO4	Apply the concepts of Genetic Algorithm.	Apply
CO5	Explain the concepts of multi-objective optimization models and the need for using hybrid soft computing approaches	Understand

iii) SYLLABUS

Introduction to Soft Computing, Artificial Neural Networks and Neuron Models, Supervised Learning Networks, Unsupervised Learning Networks, Fuzzy Sets and Fuzzy Logic Concepts, Fuzzy Systems, Genetic Algorithms, Multi-Objective Optimization and Pareto Optimality, Hybrid Systems – Neural, Fuzzy and Genetic.

iv) REFERENCES

1. S.N.Sivanandam and S.N. Deepa, Principles of Soft Computing , 2 ndEdition, John Wiley & Sons.
2. Kalyanmoy Deb, Multi-objective Optimization using Evolutionary Algorithms, 1 st Edition, John Wiley & Sons.
2. Timothy J Ross, Fuzzy Logic with Engineering Applications, John Wiley & Sons, 2016.
3. T.S.Rajasekaran, G.A.Vijaylakshmi Pai "Neural Networks, Fuzzy Logic & Genetic Algorithms Synthesis and Applications", Prentice-Hall India.
4. Simon Haykin, "Neural Networks- A Comprehensive Foundation", 2/e, Pearson Education.
5. Zimmermann H. J, "Fuzzy Set Theory & Its Applications", Allied Publishers Ltd.

v) COURSE PLAN

Module	Contents	No. of hours
I	Introduction to Soft Computing & Artificial Neural Network Introduction to Soft Computing, Difference between Hard Computing & Soft Computing & Applications of Soft Computing, Artificial Neurons Vs Biological Neurons, Basic models of artificial neural networks, Activation Functions McCulloch and Pitts Neuron, Hebb network.	10
II	Unsupervised Learning Network – K-Means, Fuzzy C-Means, Hierarchical Clustering, Mountain Clustering. Introduction to Non-traditional Metaheuristic Optimization Techniques - Random Optimization, Ant Colony Optimization, Particle Swarm Optimization, Evolutionary Algorithms - Firefly Algorithm, Bee Algorithm, Bat algorithm. MATLAB Environment for Soft Computing Techniques.	9
III	Fuzzy Set Theory - Introduction to Fuzzy Set, Properties & operations on fuzzy sets. Fuzzy relations, Operations on Fuzzy Relation. Membership functions - Introduction, Features, & Fuzzification, Methods of Membership Value Assignment; Defuzzification., Fuzzy Propositions & Fuzzy Implications Lamda cuts for fuzzy sets, Defuzzification methods. Fuzzy Systems - Crisp Logic, Fuzzy Logic, Fuzzy Rule Base and Approximate Reasoning, Fuzzy Quantifiers, Fuzzy Inference Systems – Mamdani, Sugeno, Fuzzy Decision Making, Fuzzy Logic Control System.	9

IV	Introduction to Genetic Algorithms (GA) and their Terminology - Traditional Optimization and Search Techniques vs. Genetic Algorithm, Operators in Genetic Algorithms, Problem Solving using Genetic Algorithm, Classification of Genetic Algorithms, Genetic Programming, Advantages and Limitations of Genetic Algorithm, Applications of Genetic Algorithm, Applications of GA in Machine Learning. MATLAB Environment for Genetic Algorithm.	5
V	Multi-Objective Optimization & Hybrid System MOOP-Linear & Nonlinear, Convex & Non Convex, Principles of MOO-Illustrating Pareto Optimal Solutions, Objectives in MOO, Dominance & Pareto-Optimality-Concept of Domination, Properties of Dominance Relation, Pareto Optimality, Procedure for finding a non-dominated set, Optimality Conditions. Hybrid System - Neuro Fuzzy system, Neuro Genetic system, Fuzzy Genetic system.	12
	Total hours	**45**

Course Code	Course Name	Category	L	T	P	Credit	Year of Introduction
26CS262K	**NATURAL LANGUAGE PROCESSING**	PECP	2	0	2	**3**	2026

i)COURSE OBJECTIVES:

This course provides an in-depth and research-oriented study of Natural Language Processing, covering linguistic foundations, probabilistic and neural modeling techniques, and large language models. Students will critically analyze, design, and evaluate NLP systems while addressing ethical, multilingual, and real-world deployment challenges.

ii)COURSE OUTCOMES

After the completion of the course, the student will be able to:

CO1	Apply linguistic representations and text preprocessing techniques to improve the performance of NLP models	Apply
CO2	Develop statistical language models and classical machine learning methods for text classification and language modelling tasks.	Apply
CO3	Apply probabilistic sequence models and information extraction techniques such as Named Entity Recognition and spelling correction	Apply
C04	Apply syntactic and semantic structures using parsing, word sense disambiguation, and semantic role labeling to address language ambiguity.	Apply
CO5	Build neural network–based NLP models using word embeddings, recurrent architectures, and transformer-based pretrained models for real-world applications.	Apply

iii)SYLLABUS

Fundamentals of NLP and Linguistic Concepts: NLP definition and applications – challenges in NLP – linguistic levels: morphology – syntax – semantics – pragmatics – discourse – linguistic ambiguity – lexical semantics – lexical resources: WordNet – VerbNet.

Text Processing and Representation: Tokenization – stemming – lemmatization – stop-word removal – normalization – Part-of-Speech tagging – Bag-of-Words – TF-IDF.

Statistical and Traditional Machine Learning Approaches: Language modeling – n-gram models – smoothing techniques – perplexity – text classification – Naïve Bayes – Logistic Regression – Support Vector Machines – sequence labeling – Hidden Markov Models – Conditional Random Fields.

Parsing and Semantic Processing: Constituency parsing – dependency parsing – context-free

grammars – probabilistic parsing – CYK algorithm – word sense disambiguation – semantic role labeling – coreference resolution – discourse analysis – syntax–semantics interface.

Deep Learning for NLP: Word embeddings: Word2Vec – GloVe – FastText – neural networks for NLP – RNNs – LSTMs – GRUs – sequence-to-sequence models – attention mechanism – transformer architecture – contextual embeddings – pretrained language models: BERT – GPT – T5 – fine-tuning – transfer learning.

LLMs,Applications, and Ethics: Large language models – prompt engineering – multilingual NLP – bias – fairness – explainability – privacy – responsible and ethical deployment of NLP systems.

iv) REFERENCES

- Speech and language processing (3rd ed., draft). Jurafsky, D., & Martin, J. H. (2023). Pearson Education.
- Natural language processing with Python: Analyzing text with the natural language toolkit. Bird, S., Klein, E., & Loper, E. (2009). O'Reilly Media.
- Deep learning for natural language processing. Goyal, P., Pandey, S., & Jain, K. (2018). Apress.
- Natural Language Processing with Transformers. Tunstall, L., von Werra, L., & Wolf, T. (2022). O'Reilly Media.
- Fairness and Machine Learning: Limitations and Opportunities. Barocas, S., Hardt, M., & Narayanan, A. (2023). MIT Press.
- Conversational AI: Dialogue Systems, Conversational Agents, and Chatbots. McTear, M. (2020). Morgan & Claypool Publishers.

v) COURSE PLAN

Module	Contents	No. of hours
I	Natural Language Processing: scope, importance, and research challenges. Applications in healthcare, social media, finance, and legal systems. Linguistic foundations: morphology, syntax, semantics, pragmatics, and discourse. Morphological analysis: stemming, lemmatization, and morphological parsing. Text preprocessing: tokenization, sentence segmentation, normalization, stop-word removal, handling noisy and code-mixed data. Part-of-Speech tagging. Corpus design and development: multilingual and domain-specific corpora. Annotation standards, annotation schemes, inter-annotator agreement, annotation bias, and linguistic ambiguity.	6
II	Statistical language modeling and N-gram models: unigram, bigram, trigram. Data sparsity problem and smoothing techniques: Laplace and Good–Turing estimation. Language model evaluation: perplexity. Text representation: Bag-of-Words, TF-IDF, and feature engineering.	6

	Classical text classification models: Naïve Bayes, Logistic Regression, Support Vector Machines. Evaluation metrics: precision, recall, F1-score. Sequence labeling tasks: Named Entity Recognition and POS tagging. Probabilistic sequence models: Hidden Markov Models and Conditional Random Fields.	
III	Syntactic parsing: dependency parsing, constituency parsing, context-free grammars, probabilistic parsing, CYK algorithm, parse trees. Semantic analysis: word sense disambiguation, semantic similarity, semantic role labeling. Lexical and semantic resources such as WordNet, VerbNet, and FrameNet. Discourse processing: coherence, structure, and coreference resolution. Knowledge representation: ontologies, first-order predicate logic, and knowledge graphs. Semantic search and question answering.	6
IV	Representation learning and distributed word representations. Word embeddings: Word2Vec, GloVe, FastText, contextual embeddings. Recurrent Neural Networks, Long Short-Term Memory, and Gated Recurrent Units. Sequence-to-sequence models, encoder–decoder architecture, and attention mechanisms. Applications in machine translation, summarization, and text generation.	6
V	Transformer architecture and self-attention. Pretrained language models such as BERT and RoBERTa. Transfer learning and fine-tuning. Multilingual and cross-lingual NLP, low-resource languages, and code-mixed data. Cross-lingual transfer models such as mBERT and XLM-R. Ethical and responsible AI: bias, fairness, explainability, and privacy. Emerging applications: conversational agents, social media analysis, sentiment and hate speech detection.	6
	Total hours	**30**

LAB / PRACTICAL

	Experiment	**No. of hours**
Module-1		
1	1) Perform text preprocessing on a given dataset. Implement tokenization, normalization, stop-word removal, stemming, and lemmatization. 2) Create a small text corpus and perform annotation for tasks such as Part-of-Speech tagging or sentiment labeling. Evaluate inter-annotator agreement.	6
Module2		
2	1) Implement N-gram language models and apply smoothing techniques. Evaluate the model using perplexity. 2) Develop a text classification system using classical machine learning algorithms such as Naïve Bayes or Logistic Regression. Evaluate the model using precision, recall, and F1-score.	6
Module3		
3	1) Perform syntactic parsing and generate dependency or constituency parse trees. 2) Implement Word Sense Disambiguation or semantic similarity using lexical resources such as WordNet.	6
Module-4		
4	1) Train word embeddings such as Word2Vec or FastText and evaluate their performance. 2) Develop a deep learning model such as RNN or LSTM for	6

	text classification or sentiment analysis.	
Module-5-Project		
5	1) Develop an NLP application, such as machine translation, chatbot, or text summarization, etc., by fine-tuning pretrained models such as BER.	6
	TOTAL HOURS	30

Course Code	Course Name	Category	L	T	P	Credit	Year of Introduction
26CS162D	DEEP LEARNING TECHNIQUES	PECP	2	0	2	3	2026

i) COURSE OBJECTIVES

This course introduces the core concepts of deep learning and also provides an insight into recent developments in the field. The concepts covered in the course include Neural Network Optimization techniques, Regularization, Convolutional Neural networks, Recurrent Neural Networks, Transformers and Object Detection. This course helps the students to develop solutions to real world applications using deep learning techniques.

ii) COURSE OUTCOMES

After the completion of the course, students will be able to:

CO1	Apply the standard regularization and optimization techniques for the effective training of deep neural networks.	Apply
CO2	Apply probabilistic and generative deep learning concepts to build and utilize Variational Autoencoders and Generative Adversarial Networks for real-world data modeling and generation tasks.	Apply
CO3	Make use of recurrent neural networks and its variants in relevant application areas.	Apply
CO4	Apply the transformer architecture and compare it with earlier architectures.	Apply
CO5	Identify solutions to real world problems by applying deep learning techniques.	Apply

iii) SYLLABUS

Neural network training methods, optimization techniques, and regularization. Convolutional Neural Networks (CNNs) and popular architectures for image classification. Recurrent Neural Networks (RNNs), including LSTM and GRU, for sequential data modeling. Transformer-based models, including attention mechanisms and BERT.

iv) TEXTBOOKS

1. Ian Goodfellow, Yoshua Bengio, Aaron Courville, Deep Learning, MIT Press, 2016.
2. M.Gopal, Deep Learning, Pearson , 2022.
3. Aston Zhang, Zachary C.Lipton, Mu Li ,and Alexander J.Smola, Dive into Deep Learning, available online at **https://d2l.ai**.

REFERENCES

1. Bishop, C., M., Pattern Recognition and Machine Learning, Springer, 2006.
2. Russell Reed and Robert J. Marks II, Neural Smithing: Supervised Learning in Feedforward Artificial Neural Networks, Bradford Book, 2014.
3. Michael Nielsen, Neural Networks and Deep Learning, 2015.

v) COURSE PLAN

Module	Contents	No. of hours
I	Deep Learning vs traditional machine learning, Gradient Descent, Adam Optimization, Weight initialization strategies, Batch Normalization. Convolutional Neural Networks: – convolution operation, pooling,Relation between input size, output size and filter size, 3D Convolution, Backpropagation in Convolutional Layers.	7
II	Convolutional Architectures: GoogleNet, VGG, ResNet, EfficientNet, Transfer Learning. Deep Autoencoders- sparse autoencoders, Denoising autoencoders. Introduction to GAN , Vanilla GAN, Diffusion Models.	6
III	Recurrent Neural Networks, Back propagation through time , Vanishing and exploding gradients with RNNs, Different types of RNNs-overview, Long Short Term Memory (LSTM), Gated Recurrent Unit (GRU). Applications of RNN: word prediction, Chatbots, Image captioning .	6
IV	Attention Model - Multi-head attention, Self-attention and positional encoding, Transformer architecture, Transformers for vision, Large-scale pretraining with transformers	6
V	Instance Segmentation v/s Semantic Segmentation, Object Detection : RCNN , Fast RCNN, Faster RCNN , YOLO, Mask RCNN	5

	Total hours	**30**

Sl No.	Contents	No. of hours
1.	Analyze the impact of optimization, weight initialization techniques, dropout and regularization techniques, and visualize the change in performance.	3
2.	Implement a CNN with at least 2 convolutional layers to classify images into binary classes (e.g., even vs odd digits for MNIST, or T-shirt vs trouser for Fashion MNIST). Include padding and Batch Normalization, and record accuracy vs epochs.	3
3.	Design and implement a CNN model (with 4+ layers of convolutions) to classify multi category image datasets. Use the concept of regularization and dropout while designing the CNN model. Use the Fashion MNIST datasets. Record the Training accuracy and Test accuracy corresponding to the following architectures: a. Base Model b. Model with L1 Regularization c. Model with L2 Regularization d. Model with Dropout e. Model with both L2 (or L1) and Dropout	3
4.	Digit classification using pre-trained networks like VGGnet-19 or ResNet for MNIST dataset and analyze and visualize performance improvement.	3
5.	Implement a shallow autoencoder for image reconstruction using MNIST dataset.	3
6.	Implement image generation using GAN.	3
7.	Implement a simple RNN. Analyze and visualize the performance change while using LSTM and GRU instead of simple RNN.	3
8.	Implement a Transformer-based classifier in PyTorch for the MNIST (or Fashion-MNIST) dataset and train it to classify images into 10 classes.	3
9.	Object detection using YOLO and Faster RCNN.	3
10.	Implement image segmentation using Mask R-CNN	3
	Total hours	**30**

Course Code	Course Name	Category	L	T	P	Credit	Year of Introduction
26CS162E	FOUNDATIONS OF DATASCIENCE	PEC	3	0	0	3	2026

i) COURSE OBJECTIVES

To impart fundamental knowledge of Data Science and to understand the role of statistics and optimization in performing mathematical operations for data-driven analysis. To enable learners to document, interpret, and effectively communicate analytical results using appropriate data visualization techniques. To provide foundational knowledge of various open-source Data Science tools and to understand their application workflows for solving real-world industrial problems. To develop core competencies and conceptual understanding required to pursue a career as a Data Science professional.

ii) COURSE OUTCOMES

After the completion of the course, students will be able to:

CO1	Demonstrate fundamental concepts and principles of Data Science.	Understand
CO2	Demonstrate the basic concepts and methods of statistical data analysis.	Understand
CO3	Apply basic tools and techniques of Exploratory Data Analysis (EDA) to explore, summarize and visualize data.	Apply
CO4	Apply programming techniques to handle different types of data and visualize them for knowledge representation.	Apply
CO5	Make use of numerous open-source data science tools to solve real-world problems through industrial case studies.	Apply

iii)SYLLABUS

This syllabus introduces the end-to-end data science workflow, covering problem formulation, data types, and structured analytical thinking. It builds strong statistical and optimization foundations for data analysis and modeling. Learners gain hands-on programming skills in Python along with exploratory data analysis techniques. The course emphasizes data

handling, preprocessing, and effective visualization methods. It concludes with practical exposure to popular data science tools and real-world case studies.

iv)TEXTBOOKS

1 Avrim Blum, John Hopcroft, Ravindran Kannan, “Foundations of Data Science”,Cambridge University Press, 2020.

2 R. V. Hogg, J. W. McKean and A. Craig, Introduction to Mathematical Statistics, 8th Ed., Pearson Education India, 2019.

3 Hossein Pishro-Nik, “Introduction to Probability, Statistics, and Random Processes”, Kappa Research, LLC, 2014.

v) REFERENCES

1. Ani Adhikari and John DeNero, “Computational and Inferential Thinking: The Foundations of Data Science”, GitBook, 2019.
2. Cathy O“Neil and Rachel Schutt, “Doing Data Science: Straight Talk from the Frontline”, O“Reilly Media, 2013.

vi) COURSE PLAN

Module	Contents	No. of hours
I	Basics of Data Science -Introduction; Typology of problems; Importance of linear algebra, statistics and optimization from a data science perspective; Structured thinking for solving data science problems, Structured and unstructured data.	5
II	Statistical Foundations-Descriptive statistics, Statistical Features, summarizing the data, outlier analysis, Understanding distributions and plots, Univariate statistical plots and usage, Bivariate and multivariate statistics, Dimensionality Reduction, Over and Under Sampling, Bayesian Statistics, Statistical Modeling for data analysis Optimization- Unconstrained optimization; Necessary and sufficiency conditions for optima; Gradient descent methods; Constrained optimization, KKT conditions; Introduction to non-gradient techniques; Introduction to least squares optimization	11

III	Programming Foundation and Exploratory Data Analysis -Introduction to Python Programming, Types, Expressions and Variables, String Operations, selection, iteration, Data Structures- Strings, Regular Expression, List and Tuples, Dictionaries, Sets. Exploratory Data Analysis (EDA) - Definition, Motivation, Steps in data exploration, The basic datatypes, Data type Portability, Basic Tools of EDA, Data Analytics Life cycle, Discovery .	9
IV	Data Handling and Visualization -Data Acquisition, Data Pre-processing and Preparation, Data Quality and Transformation, Handling Text Data; Introduction to data visualization-Bar Chart, Line chart, Table, Heat Map, Tree Map, Packed Bubble, Tooltip. Visualization Workflow: Describing data visualization workflow, Visualization Periodic Table; Data Abstraction -Analysis: Four Levels of Validation- Task Abstraction. Data Representation- Chart types: categorical, hierarchical, relational, temporal & spatial	10
V	Data Science Tools and Techniques-Overview and Demonstration of Open source tools such as R, Octave, Scilab. Python libraries: SciPy and sci-kitLearn, PyBrain, Pylearn2; Weka. Case study implementation using Data Science tools.	10
	Total Hours	**45**

Course Code	Course Name	Category	L	T	P	Credit	Year of Introduction
26CS162F	DATA VISUALIZATION USING PYTHON	PECP	2	0	2	3	2026

i) COURSE OBJECTIVES

This course provides a rigorous and practice-oriented understanding of data visualization principles, techniques, and tools using Python. Emphasis is placed on visual analytics for complex, high-dimensional, and real-world datasets arising in engineering, science, healthcare, and industry. The course integrates data acquisition, preprocessing, dimensionality reduction, and advanced visualization techniques with interactive and exploratory visual analysis.

ii) COURSE OUTCOMES

After the completion of the course, the student will be able to:

CO1	Apply knowledge of data characteristics to select and use appropriate visualization techniques for effective data representation.	Apply
CO2	Apply Python-based visualization libraries to generate effective plots and charts.	Apply
CO3	Apply appropriate visualization techniques to represent multivariate, temporal, spatial, and high-dimensional datasets effectively.	Apply
CO4	Apply advanced visualization methods to design and implement interactive visualizations for complex datasets.	Apply
CO5	Apply data preprocessing and dimensionality reduction techniques to integrate and enhance visualizations for effective data analysis.	Apply

iii) SYLLABUS

Introduction to data visualization: definition, scope, importance, benefits, and applications in data science and analytics. Data acquisition concepts including structured, semi-structured, and unstructured data; data extraction from files, databases, APIs, and sensors. Data preprocessing for visualization: data cleaning, annotation, integration, transformation, and reduction. Visualization terminology, design principles, perceptual issues, and color theory. Basic visualization techniques using Python for one-dimensional and two-dimensional data. Chart-based visualizations including area, box-and-whisker, heat maps, and stacked charts. Multivariate data visualization and advanced visualization techniques. Pixel-oriented, geometric projection, icon-based, and hierarchical visualization techniques. Interactive visualization concepts and manipulation of views. Visualization of networks, trees, and complex data relationships. High-dimensional data visualization and dimensionality reduction techniques. Visualization-driven exploratory data analysis and case studies using real-world datasets.

iv) REFERENCES

1. Andy Kirk, Data Visualization: A Handbook for Data-Driven Design, SAGE, 2016.
2. Jake VanderPlas, Python Data Science Handbook, O'Reilly, 2016.
3. Dr Ossama Embarak, 'Data Analysis and Visualisation Using python- Analyze Data to Create Visualizations for BI Systems -Apress
4. Fabio Nelli, "Python Data Analytics with pandas, NumPy, Matplotlib", Second Edition, Apress
5. Matt Harrison and Michael Prentiss, "Learning the Pandas Library- Python tools for Data Munging, Data Analysis, and Visualisation"
6. Tamara Munzner, Visualization Analysis and Design, CRC Press, 2014.
7. Nathan Yau, Visualize This, Wiley, 2011.
8. Alberto Cairo, The Functional Art, New Riders, 2012.
9. Scott Murray, Interactive Data Visualization for the Web, O'Reilly, 2013.
10. Official documentation of Seaborn, Plotly, and Pygal.

v) COURSE PLAN

Module	Contents	No. of hours
I	**Introduction to Data Visualization and Data Preparation:** Data visualization: definition, scope, and role in data science and analytics, Importance and benefits of effective visualization in decision making and research, Applications of data visualization, Data acquisition concepts: structured, semi-structured, and unstructured data sources, Data extraction techniques from files, databases, APIs, and sensors, Data cleaning: handling missing values, outliers, noise, and inconsistencies, Data annotation and labeling concepts, Data integration: combining heterogeneous data sources, Data transformation: normalization, scaling, encoding, aggregation, Data reduction: sampling, feature selection, and summarization, End-to-end visualization workflow and pipeline	6
II	**Visualization Foundations and Basic Charts:** Visualization terminology: data types, attributes, marks, channels, encodings, Human perception and cognition in visualization, Design principles: clarity, accuracy, efficiency, and ethics, Color theory: perceptual color models, sequential, diverging, and categorical palettes, One-dimensional visualizations: Pictograms and symbol-based charts, Bar charts and variants, Pie charts and limitations, Two-dimensional visualizations: Scatter plots and scatter matrices, Histograms and density plots, Ogive curves and cumulative plots, Line plots for temporal data, Chart-based visualizations: Area charts	6

	and stacked area charts, Stacked and grouped bar charts, Box-and-whisker plots, Waterfall charts, Gantt charts for project visualization.	
III	**Multivariate and Advanced Visualization Techniques:** Multivariate data visualization concepts and challenges, Scatter plot matrices and parallel coordinate plots, Heat maps: construction, interpretation, and use cases, Correlation visualization and matrix-based views, Facet grids and small multiples for comparative analysis, Pixel-oriented visualization techniques for large datasets, Geometric projection visualization techniques, Icon-based visualization and glyph design, Hierarchical visualization techniques: Trees and dendrograms, Treemaps and sunburst charts	6
IV	**Interactive and Complex Data Visualization:** Principles of interaction in visualization, Interaction techniques: zooming, panning, filtering, brushing, and linking, Manipulating views for exploratory data analysis, Visualization of relational and graph-based data, Network visualization concepts and layouts, Creation of interactive network topologies, Visualization of hierarchical data using interactive treemaps, Visualizing complex data relationships and structures, Introduction to three-dimensional visualization, Creation and interpretation of 3D charts.	6
V	**Dimensionality Reduction and Visual Analytics:** Challenges of high-dimensional data visualization, Curse of dimensionality: visual perspective, Dimensionality reduction for visualization: Principal Component Analysis (PCA), t-SNE and UMAP (concepts and visualization use), Visual analytics and exploratory data analysis (EDA), Visualization-driven hypothesis generation, Visualizing multimodal datasets Case studies: Sensor and IoT data visualization, Healthcare and clinical data visualization, Genomics and biomedical data visualization.	6
	Total hours	**30**

Exp No.	Contents	No. of hours
1	Design a robust data preprocessing pipeline for a mixed dataset (numerical + categorical + temporal) and justify preprocessing choices for visualization.	2
2	Compare the impact of different outlier-handling strategies on downstream visual interpretation.	2
3	Build a reusable visualization-ready data pipeline using functional programming style in Python.	2
4	Demonstrate how poor color palette selection can mislead interpretation and correct it using perceptual color theory.	2
5	Redesign a visualization to reduce cognitive load while preserving information density.	2
6	Evaluate ethical issues in visualization using scale manipulation and truncation.	2
7	Analyze high-dimensional correlations using combined heatmap + dendrogram visualization.	2
8	Design a glyph-based visualization to encode three numerical variables.	2
9	Compare parallel coordinates vs scatter matrix for multivariate pattern discovery.	2
10	Design an interactive exploratory system using linked views (scatter + histogram)	2
11	Analyze network centrality visually and validate findings algorithmically.	2
12	Evaluate limitations of 3D visualizations through comparative analysis.	2
13	Compare PCA, t-SNE, and UMAP visually and analytically for the same dataset.	2
14	Evaluate stability of dimensionality reduction under random initialization.	2

15	Perform visual hypothesis generation from reduced-dimensional views.	2
	Total Hours	**30**

Course Code	Course Name	Category	L	T	P	Credit	Year of Introduction
26CS262A	DATA ANALYTICS	PEC	2	0	2	3	2026

i) COURSE OBJECTIVES

This course is offered to introduce fundamental algorithmic ideas in processing data. The preliminary concepts of Hadoop and Map Reduce are included as part of this course.

ii) COURSE OUTCOMES

After the completion of the course, students will be able to:

CO1	Apply data science concepts to identify real-world problems	Apply
CO2	Apply big data concepts by demonstrating Hadoop basics, HDFS operations, and MapReduce processing with YARN.	Apply
CO3	Apply Hadoop MapReduce concepts to analyze data .	Apply
CO4	Apply Power BI tools to import, clean, model, and analyze data using Power Query and DAX	Apply
CO5	Apply MLOps practices to build and deploy reproducible machine learning workflows.	Apply

iii)SYLLABUS

Big data fundamentals, five V's, analytics practices, real-world use cases, and overview of Apache Hadoop and its ecosystem.

HDFS architecture, design principles, daemons, metadata management, and data read/write operations.

MapReduce framework including stages, job anatomy, scheduling, shuffle and sort, task execution, and YARN.

Big data management tools: Pig (Pig Latin, execution modes), Hive (architecture, HiveQL), and introductory NoSQL concepts.

iv)TEXTBOOKS

1. Davy Cielen, Arno D. B. Meysman, and Mohamed Ali ,“Introducing Data Science - Big data, machine learning, and more, using Python tools” , Dreamtech Press 2016.
2. Michael Minelli, Michelle Chambers, and AmbigaDhiraj, "Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's Businesses",Wiley,2013
3. EMC Education Services, “Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data”, Wiley ,January 2015
4. Tom White,"Hadoop: The Definitive Guide", Third Edition, O'Reilley,2012.
5. Eric Sammer,"Hadoop Operations",O'Reilly Media, Inc ,2012
6. E. Capriolo, D. Wampler, and J. Rutherglen, "Programming Hive", O'Reilley, 2012. 7. "Programming Pig", Alan Gates, O’Reilley,2011.

REFERENCES

1. Sourabh Mukherjee, Amit Kumar Das and Sayan Goswami, “ Big Data Simplified”, Pearson, 1st edition, 2019.
2. Murtaza Haider, “Getting Started with Data Science”, First Edition, Kindle Edition, IBM Press, 2015.
3. Thomas Erl, Wajid Khattak and Paul Buhler “ Big Data Fundamentals:Concepts, Drivers and Techniques”, Prentice Hall, Pearson Service, 2016.

v) COURSE PLAN

Module	Contents	No. of hours
I	Data Analytics Types of Analytics Data Science vs Data Analytics vs Big Data, Data Understanding, Structured / Semi-structured / Unstructured data Parquet Data Quality & Data Governance Basics, Sampling and Data Collection Strategies, Intro to Data Storage Concepts, Basic Statistical Concepts for Analytics, Introduction to EDA Tools, Feature Engineering Basics, Model Evaluation Concepts	6

II	Big Data Overview–the five V's of big data-State of the Practice in Analytics, Examples of Big Data Analytics-Apache Hadoop and the Hadoop Ecosystem-HDFS-Design of HDFS, HDFS Concepts-Daemons-Reading and Writing Data-Managing File system Metadata- Map Reduce-The Stages of Map Reduce -Introducing Hadoop Map Reduce ,Daemons-YARN	6
III	Analyzing the Data with Hadoop using Map and Reduce-Developing a Map Reduce Application-Anatomy of a Map Reduce Job- Scheduling-Shuffle and Sort - Task execution. Big data Management Tools: PIG- : Introduction to PIG, Execution Modes of Pig, Pig Latin, HIVE: Hive Architecture, HIVEQL, Introduction to NoSQL.	6
IV	Power BI overview and workflow (Desktop, Service, reports, dashboards),Data import from Excel/CSV/SQL and refresh basics ,Data cleaning using Power Query (filter, split, merge, append, missing values), Data modeling (relationships, star schema, columns vs measures), DAX basics for calculations and KPIs (CALCULATE, filters, time intelligence), Creating visuals and interactive reports (charts, slicers, drill-down, formatting) Publishing and sharing in Power BI Service (dashboards, workspaces)	6
V	Introduction to MLOps (lifecycle, MLOps vs DevOps/DataOps) ML workflow recap (training, evaluation, reproducibility). Version control for code, data, and models (Git, DVC). Experiment tracking & model registry (MLflow)Automated ML pipelines and CI/CD (GitHub Actions/Jenkins) . Model packaging & environment management (Docker) Deployment & model serving (API, batch, real-time). Scaling basics (Kubernetes intro) Governance, security, compliance, Responsible AI Cloud deployment basics (AWS/Azure/GCP)	6
	Total Hours	**30**

LAB / PRACTICAL

Module	Experiment	No. of hours
I	Introduction to Big Data & Ecosystem (Setup & Acquisition) a) Environment Setup & HDFS Management: Install and configure Apache Hadoop in Pseudo-Distributed mode. Practice Hadoop Distributed File System (HDFS) commands (copyFromLocal, copyToLocal, mkdir, ls, rm) to manage data. b) Big Data Ingestion (Informatica/Sqoop): Utilize Apache Sqoop to import data from a relational database (MySQL/Oracle) into HDFS, or use a data integration tool (like Pentaho Data Integration - Kettle) to ingest raw data. c)Security & Auditing: Configure HDFS permissions and run auditing logs to track file access.	6
II	Data Analysis & Processing (MapReduce & Hive) a)Word Count MapReduce: Implement a Java-based MapReduce program to perform a word count on a large text file to understand the "4 Vs" (Volume, Variety). b) Advanced MapReduce: Write a MapReduce program to analyze weather data (find max/min temperature) or perform Matrix Multiplication to process structured/semi-structured data. c) Data Analysis with Hive: Create Hive tables (Internal/External), perform data loading, and execute HiveQL queries to analyze dataset patterns.	6
III	Data Analysis Tools & Visualization (Business Analytics) a)Pentaho Data Integration (PDI): Use Pentaho (Spoon) to design transformations and jobs for cleaning, filtering, and aggregating data from disparate sources. b) Business Reporting (Cognos/Microstrategy): Connect to a data source (Hive or MySQL) and create interactive dashboards using Cognos Analytics or MicroStrategy to visualize data trends.	6

IV	Intelligent Predictive Analytics & Machine Learning a) Unsupervised Learning (K-Means Clustering): Implement K-Means clustering in Hadoop/Spark to group similar data points (e.g., customer segmentation). b) Association Rule Mining (Apriori Algorithm): Implement the Apriori algorithm on a market basket dataset to find frequently occurring itemsets. · c)Supervised Learning (Neural Networks): Use a Python library (like Scikit-learn or TensorFlow) to implement a basic neural network or Kohonen model on a large dataset.	6
V	Module 5: Stream Computing & Real-time Analytics a) Data Streaming Basics: Setup a stream processing environment and implement data sampling techniques to analyze streaming data in real-time. b) Real-time Analytics (IBM InfoSphere Streams): Develop a simple application using IBM InfoSphere Streams (or Spark Streaming) to process streaming data and estimate moments in a stream.	6
	Total Hours	**30**

Course Code	Course Name	Category	L	T	P	Credit	Year of Introduction
26CS162G	FUZZY SET THEORY AND APPLICATIONS	PEC	3	0	0	3	2026

i) COURSE OBJECTIVES:

The course helps students to understand Fuzzy Set Theory and the basis of fuzzy logic and its applications such as fuzzy control and fuzzy decision making.

ii) COURSE OUTCOMES:

After the completion of the course the student will be able to:

CO1	Solve problems related to fuzzy sets and fuzzy relations.	Apply
CO2	Make use of membership functions, fuzzification and defuzzification process.	Apply
CO3	Apply defuzzification methods to implement and obtain crisp outputs from fuzzy inference systems.	Apply
CO4	Apply fuzzy logic control to real time systems.	Apply
CO5	Apply fuzzy inference to solve problems that include uncertainty.	Apply

iii) SYLLABUS

Classical & Fuzzy Sets, Relations – Properties and Operations, Fuzzy Membership Functions, Defuzzification to Scalars, Classical Logic, Fuzzy Logic, Applications of Fuzzy Systems.

iv) REFERENCES:

1. Timothy J Ross, *Fuzzy Logic with Engineering Applications*, 3rd edition, Wiley Publications, 2010.
2. George J Klir and Bo Yuan, *Fuzzy Sets and Fuzzy Logic: Theory and Applications*, 1st edition, PHI, 1995.
3. H J Zimmerman, *Fuzzy Set Theory and its Applications*, 4th edition, Kluwer Academic Publishers, 2001.
4. John Yen and Reza Langari, *Fuzzy Logic: Intelligence, Control and Information*, 1st edition, Pearson Education, 2002.

v) COURSE PLAN

Module	Contents	No. of hours
I	The case for imprecision, Utility and Limitations of Fuzzy Systems, Fuzzy Sets and Membership, Classical Sets – Properties, Operations, Fuzzy Sets – Properties and Operations, Classical Relations – Cartesian Product, Operations and Properties of Crisp Relations, Composition, Fuzzy Relations – Cardinality, Operations, Properties, Fuzzy Cartesian Product and Composition, Tolerance and Equivalence Relations – Crisp and Fuzzy, Similarity Methods – Cosine, Min-max.	10
II	Fuzzy Membership Functions – Features, Fuzzification, Defuzzification to Crisp Sets, λ-CUTS FOR FUZZY RELATIONS, Development of Membership Functions –Intuition, Inference, Rank ordering, Inductive reasoning.	9
III	Defuzzification to Scalars - Max membership principle, Centroid method, Weighted average method, Mean max membership, Center of sums, Center of largest area, First (or last) of maxima, Implementation of defuzzification methods Linguistic Hedges, Fuzzy Arithmetic: Fuzzy Numbers, Linguistic Variables, Introduction to Matlab/Python, Study of fuzzy logic toolbox, Implementation of defuzzification methods.	10
IV	Classical Logic, Fuzzy Logic, Multivalued Logics, Fuzzy Propositions, Fuzzy Qualifiers, Linguistic Hedges, Approximate Reasoning, Fuzzy (Rule-Based) Systems - Multiple conjunctive antecedents, Multiple disjunctive antecedents Aggregation of fuzzy rules, Graphical Techniques of Inference, Simple implementation of fuzzy logic.	8
V	Applications of Fuzzy Systems - Fuzzy Classification, Fuzzy Pattern Recognition, Fuzzy Control Systems, Fuzzy Systems and Neural Networks, Fuzzy Clustering, Fuzzy Databases and Information retrieval systems, Case Study – ANFIS.	8
	Total hours	**45**

Course Code	Course Name	Category	L	T	P	Credit	Year of Introduction
26CS162H	**APPLIED MACHINE LEARNING**	**PECP**	**2**	**0**	**2**	**3**	**2026**

i) COURSE OBJECTIVES:

The course aims to build computer systems that learn from experience. The course enables students to develop new algorithms, and to understand which algorithms should be applied in which circumstances

ii) COURSE OUTCOMES:

After the completion of the course, the student will be able to:

CO1	Make use of fundamental concepts of machine learning and data preprocessing techniques	Apply
CO2	Utilize classification algorithms and use performance metrics to assess model effectiveness.	Apply
CO3	Apply association rule mining and ensemble learning techniques to build and utilize machine learning models for solving real-world data analysis problems.	Apply
CO4	Apply artificial neural network and deep learning models for pattern recognition and predictive tasks.	Apply
CO5	Apply clustering techniques and basic reinforcement learning methods for knowledge discovery and decision-making	Apply

iii) SYLLABUS

Classification, Regression, Support Vector Machines, Artificial Neural Networks, Hidden Markov models, Decision Tree, Ensemble classification, Clustering ,Case studies

iv) REFERENCES:

1 Shai Shalev-Shwartz and Shai Ben-David, Understanding Machine Learning (3rd Edition), Cambridge University Press, 2015. ISBN 978-1107512825.

2 Trevor Hastie, Robert Tibshirani and Jerome Friedman, The Elements of Statistical Learning (ESL) (2nd Edition), Springer, 2016. ISBN 978-0387848570.

3 Ian Goodfellow, Yoshua Bengio, Aaron Courville, and Francis Bach, Deep Learning Adaptive Computation and Machine Learning series Hardcover (1st Edition), MIT Press, 2017. ISBN 978-0262035613.

4 Yuxi (Hayden) Liu, Python Machine Learning by Example Paperback Import (1st Edition), Packt Publishing, 2017. ASIN: B01MT7ATL5.

5 Trevor Hastie, Robert Tibshirani, Jerome Friedman, The Elements of Statistical Learning (ESL) (1st Edition), Springer, 2009. ISBN 978-0387848587.
6 Kevin Murphy, Machine Learning: A Probabilistic Perspective (MLAPP) (1st Edition), MIT Press, 2012. ISBN 978-0262018029.
7 Machine Learning with MATLAB/Octave: A MATLAB/Octave based collection of many ML algorithms (a supplement to the book "Machine Learning: A Probabilistic Perspective").

v) COURSE PLAN

Module	Contents	No. of hours
I	Introduction to learning, types of learning, role of learning, Machine learning, supervised learning, unsupervised learning, semi-supervised learning, Applications of machine learning. Types of data, attributes, types- nominal, ordinal, interval, ratio Similarity measures: Euclidian, Manhattan distance, Cosine similarity. Dimensionality reduction techniques- Principal Component Analysis, Parametric data reduction, Histograms	6
II	Classification- Concepts, Classifier performance- Accuracy, Error rate, Precision, Recall Decision trees, Information Gain, Gain Ratio, Gini Index, ID3 Algorithm, Bayes Theorem, Naive Bayesian Classification	6
III	Support Vector Machines- Maximum margin hyperplanes, Linear SVM, Non-linear SVM, Kernel Trick Association Learning, Basics of Association, Apriori Algorithm, Eclat Algorithm, FP Growth Algorithm. Ensemble Methods –Bagging, Boosting, AdaBoost, Random Forests Real world Application	6
IV	Artificial Neural Networks- basics, learning perception model, Multi layer feed forward network, back propagation. Deep Neural Networks. Case study	6
V	Unsupervised learning- Clustering – Partitioning Method-K-Means, KMedoids, Hierarchical Methods- Agglomerative versus Divisive clustering, Single link algorithm, Complete link algorithm, Distance measures in algorithmic methods, BIRCH- Multiphase Hierarchical clustering using clustering feature trees. Introduction Reinforcement learning. Case Study	6
	Total hours	**30**

LIST OF EXPERIMENTS

SL No.	Topics	No. of hours
1.	Build a Python-based pipeline that cleans, transforms, and visualizes a high-dimensional dataset to identify hidden patterns and correlations.	2
2.	Given a dataset (csv file) containing multiple independent variables and a target variable, Perform the following operations. (a) Implement a Python program to fit a linear trend and a non-linear (polynomial) trend to the data. (b) Evaluate which model provides a better fit using appropriate statistical metrics. (c) Demonstrate the use of Logistic Regression to classify the data into discrete classes and visualize the decision boundary.	4
3.	Implement the K-Nearest Neighbor (KNN) algorithm using Python to classify a given multivariate dataset.	2
4.	Develop a Python-based classification framework to implement the Naive Bayes algorithm on a multivariate dataset.	2
5.	Implement the ID3 (Iterative Dichotomiser 3) algorithm to construct a Decision Tree for a discrete-valued dataset.	3
6.	Implement a robust classification framework using Support Vector Machines (SVM) to identify the optimal separating hyperplane for a given multivariate dataset	3
7.	Develop a Python-based pipeline to implement Principal Component Analysis (PCA) for dimensionality reduction.	3
8.	Implement a robust K-Means Clustering pipeline to partition a multivariate dataset into distinct, non-overlapping subgroups.	3
9.	Implement the Agglomerative Hierarchical Clustering algorithm to discover nested structural patterns within a multidimensional dataset. Your implementation must demonstrate the transition from individual data points to a unified cluster hierarchy by Linkage Strategy Analysis.	4
10.	Implement a multi-layer Artificial Neural Network (ANN) to solve a non-linear classification task.	3
	Total Hours	**30**

Course Code	Course Name	Category	L	T	P	Credit	Year of Introduction
26CS162I	COMPUTER VISION	PEC	3	0	0	3	2026

i) COURSE OBJECTIVES:

This course provides the basic concepts and advanced techniques in computer vision. The areas comprising the syllabus include modern CNN architectures, modern object detection architectures, and most recent architectures using transformers. On completion of this course, the student would have an insight into the latest deep learning techniques used in computer vision.

ii) COURSE OUTCOMES:

After the completion of the course, the student will be able to:

CO1	Apply the computer vision pipeline techniques and image processing techniques.	Apply
CO2	Apply modern convolutional neural network architectures to solve real world problems.	Apply
CO3	Apply modern object detection architectures.	Apply
CO4	Apply Generative Adversarial Networks for computer vision.	Apply
CO5	Apply recent architectures for computer vision.	Apply

iii)SYLLABUS

Introduction to Computer Vision, Applications of Computer Vision, Computer Vision Pipeline, Image Classification and Transfer Learning, Open-Source Vision Datasets, Convolutional Neural Networks and Architectures (AlexNet, VGG, Inception, ResNet, MobileNet), Object Detection Frameworks and Evaluation Metrics, Region-Based Detectors (R-CNN, Fast R-CNN, Faster R-CNN), Single-Shot Detectors (SSD, YOLO, YOLOv3, YOLOv7), Generative Adversarial Networks (DCGAN, Pix2Pix, SRGAN), Image-to-Image Translation and Super-Resolution, Neural Style Transfer, Visual Embeddings, Vision Transformers, TransGAN, GPV-1, Mobile-Former, Case Studies.

iv)REFERENCES:

1. Richard Szeliski, *Computer Vision: Algorithms and Applications*, Springer, 2nd Edition, 2022.
2. Ian Goodfellow, Yoshua Bengio, Aaron Courville, *Deep Learning*, MIT Press, 2016.
3. Mohamed Elgendy, *Deep Learning for Vision Systems*, Manning Publications, 2020.
4. Adrian Rosebrock, *Deep Learning for Computer Vision with Python*, PyImageSearch, 2019.
5. Jason Brownlee, *Generative Adversarial Networks Cookbook*, Packt Publishing, 2020.
6. Francois Chollet, *Deep Learning with Python*, Manning Publications, 2nd Edition, 2021.
7. Luc Van Gool, Cordelia Schmid (Editors), *Computer Vision – ECCV & CVPR Tutorials*, Springer, 2021.

v) COURSE PLAN

Module	Contents	No. of hours
I	Introduction to computer vision, Computer vision, Applications of computer vision, Computer vision pipeline, Digital Image Representation, Color Models-RGB model, HSB model, YCbCr model, Image File Formats -JPEG, PNG, BMP, Basic Image Processing Operations - Image Enhancement -contrast stretching, histogram equalization, Image Filtering-smoothing, sharpening basics, Noise in Images - Gaussian, Salt & Pepper.	10
II	Convolutional Neural Networks (CNNs): Architecture and components of CNN, convolution operation, activation functions, pooling techniques, fully connected layers. Image classification using CNNs. Deep CNN architectures: AlexNet – architecture and features, VGGNet (VGG16, VGG19), Inception architecture and GoogLeNet, Residual Networks (ResNet), MobileNet – architecture and applications.	9
III	General object detection framework, object detection evaluation metrics, region-based convolutional neural networks (R-CNN), Fast R-CNN, Faster R-CNN, single-shot detector (SSD), YOLO (You Only Look Once), YOLOv3.	10

IV	Deep Convolutional Generative Adversarial Networks (DCGAN), image-to-image translation using GANs (Pix2Pix GAN), image super-resolution using GANs (SRGAN), neural style transfer, visual embeddings.	9
V	Advanced Vision and Transformer-based Models: YOLOv7, Vision Transformer (ViT), TransGAN, GPV-1, Mobile-Former.	7
	Total hours	**45**

Course Code	Course Name	Category	L	T	P	Credit	Year of Introduction
26CS162J	COMPUTATIONAL LINGUISTICS	PEC	3	0	0	3	2026

i) COURSE OBJECTIVES:

This course introduces the fundamentals of Language processing from a computational viewpoint. This course covers Language models, Computational Phonology and Morphology Unification, Semantics and knowledge representation and Pragmatics. It helps the student to apply NLP tasks such as POST, WSD, and modeling of languages.

ii) COURSE OUTCOMES:

After the completion of the course, the student will be able to:

CO1	Apply Probabilistic Models of Pronunciation and Spelling	Apply
CO2	Apply the different methods for Parsing with Context-Free Grammars for English	Apply
CO3	Apply basic concepts for Probabilistic Context-Free Grammars	Apply
CO4	Apply the key concepts Word Sense Disambiguation and Information Retrieval	Apply
CO5	Develop an application that uses Natural Language Generation concepts	Apply

iii) SYLLABUS

Introduction to Natural Language Processing, Words and Regular Expressions, Automata and Morphology, Finite-State Transducers and Computational Phonology, Pronunciation and Spelling Models, Syntax and N-gram Models, Part-of-Speech Tagging and Word Classes, Context-Free Grammars and Parsing Techniques, Probabilistic Context-Free Grammars and CYK Parsing, Lexicalized and Dependency Grammars, Feature Structures and Unification, Agreement and Subcategorization, Semantic Analysis and Lexical Semantics, Word Sense Disambiguation, Pragmatics, Information Retrieval, Natural Language Generation, Machine Translation.

iv) REFERENCES:

1. Jurafsky, D., & Martin, J. H. Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition. Prentice Hall, 2000.
2. Charniak, E. Statistical Language Learning. The MIT Press.1993
3. Allen, J. (1995). *Natural Language Understanding* (2nd ed.). Benjamin/Cummings Publishing.

v) COURSE PLAN

Module	Contents	No. of hours
I	Introduction- Words-Regular Expressions- Automata- Morphology- Finite-State Transducers- Computational Phonology- Pronunciation Modeling- Probabilistic Models of Pronunciation- Probabilistic Models of Spelling	7
II	Syntax- N-gram models, N-gram models of Syntax-Word Classes- Part- of-Speech Tagging- Context-Free Grammars for English- Parsing, Parsing with Context-Free Grammars	9
III	Probabilistic Context-Free Grammars- Probabilistic CYK, Parsing of PCFGs, Learning PCFG Probabilities, Problems with PCFGs, Probabilistic Lexicalized CFGs- Dependency Grammars- Human Parsing	10
IV	Unification of Feature Structures- Feature Structures in the Grammar, Agreement-Head Features- Subcategorization- Long Distance Dependencies- Implementing Unification- Unification Data Structures	9
V	Semantics and Pragmatics- Representing Meaning- Semantic Analysis, Lexical Semantics, Word Sense Disambiguation- Information Retrieval, Natural Language Generation, Machine Translation	10
	Total hours	**45**

Course Code	Course Name	Category	L	T	P	Credit	Year of Introduction
26CS162K	HIGH PERFORMANCE COMPUTING	PEC	3	0	0	3	2026

i) COURSE OBJECTIVES:

This course helps the learners to understand the different architectural features of high-end processors. This course discusses the Basics of high-end processors Architecture, Instruction-Level Parallelism, Data-Level Parallelism, Thread Level Parallelism, and GPU Architectures. This course enables the students to provide solutions to real-world problems Making use of the capabilities of HPC systems.

ii) COURSE OUTCOMES:

After the completion of the course, the student will be able to:

CO1	Choose different types of modern processing environments and parallel computing hardware.	Apply
CO2	Make use of the concepts of Instruction Level Parallelism and thread level parallelism in branch prediction and multithreading.	Apply
CO3	Apply the idea of Data Level Parallelism.	Apply
CO4	Apply the concept of shared memory architectures.	Apply
CO5	Identify the advanced features of GPU architecture.	Apply

iii) SYLLABUS

Classes of Computers, Parallelism and Parallel Architectures, Computer Architecture Fundamentals, Dependability and Quantitative Design Principles, Memory Hierarchies and Virtual Memory, Pipelining, Instruction-Level Parallelism and Compiler Techniques, Branch Prediction and Hardware Speculation, Multithreading and Thread-Level Parallelism, Vector Architectures and SIMD Extensions, Graphics Processing Units and Loop-Level Parallelism, Multiprocessor Architectures, Shared and Distributed Memory Systems, Cache Coherence and Synchronization, Memory Consistency Models, CPU–GPU Systems and Accelerated

Computing Platforms, GPU Architecture and Memory Spaces, CPU–GPU Data Transfer and PCI Bus, Multi-GPU Platforms and Performance Benefits.

iv) REFERENCES:

1. John L. Hennessy, David A. Patterson Computer Architecture, A Quantitative Approach, Morgan Kaufman, Seventh Edition,2025.distributed.
2. Robert Robey, Yuliana Zamora, Parallel and High-Performance Computing, Manning Publications, First Edition, 2021.
3. Thomas Sterling, Matthew Anderson, and Maciej Brodowicz, High-Performance Computing– Modern Systems and Practices, First Edition,2017.
4. Charles Severance, Kevin Dowd, High-Performance Computing, O'Reilly Media, Second Edition,1998.
5. Kai Hwang, Faye Alaye Briggs, Computer Architecture and Parallel Processing, McGraw-Hill,1984.

v) COURSE PLAN

Module	Contents	No. of hours
I	Classes of Computers- Classes of Parallelism and Parallel Architectures–Defining Computer Architecture– Dependability– Quantitative Principles of Computer Design– Basics of Memory Hierarchies–Virtual Memory and Virtual Machines–Pipelining	7
II	Instruction-Level Parallelism: Concepts and Challenges – Basic Compiler Techniques for Exposing ILP – Reducing Branch Costs With Advanced Branch Prediction – Hardware- Based Speculation – Multithreading: Exploiting Thread-Level Parallelism to Improve Uniprocessor Throughput	9
III	Vector Architecture – SIMD Instruction Set Extensions for Multimedia – Graphics Processing Units – Detecting and Enhancing Loop-Level Parallelism	10
IV	Multiprocessor Architecture: Issues and Approach – Centralized Shared-Memory Architectures –Performance of Symmetric Shared-Memory Multiprocessors– Distributed Shared-Memory and Directory-Based Coherence – Synchronization: The Basics – Introduction to Memory Consistency	9

V	The CPU-GPU system as an accelerated computational platform – The GPU and the thread engine – Characteristics of GPU memory spaces – The PCI bus: CPU to GPU data transfer overhead – Multi-GPU platforms – Potential benefits of GPU – accelerated platforms	10
	Total hours	**45**

Course Code	Course Name	Category	L	T	P	Credit	Year of Introduction
26CS162L	ADVANCED TOPICS IN DISTRIBUTED SYSTEMS	PEC	3	0	0	3	2026

i)COURSE OBJECTIVES:

To impart deeper understanding in Architecture and issues of distributed systems, Distributed algorithms, Hadoop systems.

ii) COURSE OUTCOMES

CO1	Make use of conceptual and practical aspects of distributed systems.	Apply
CO2	Apply Hadoop concepts to analyze datasets in a distributed environment.	Apply
CO3	Utilize MapReduce features such as counters, sorting, joins, and side data distribution to solve data analysis problems.	Apply
CO4	Apply synchronization techniques to coordinate processes and election algorithms to select coordinators in distributed systems.	Apply
CO5	Make use of algorithms in general synchronous networks.	Apply

iii)SYLLABUS

Distributed System: System Architecture, Processes, Threads, Code migration, Communication, Naming, Hadoop: Map and Reduce, Hadoop Distributed File System, Map Reduce Types, Administering Hadoop, Distributed Algorithms: Causality, Modeling a Distributed Computation, Synchronization and Election, Distributed Mutual Exclusion, Algorithms in General Synchronous Networks.

iv) TEXTBOOKS

1. Andrew S. Tanenbaum, Maarten Van Steen.” Distributed Systems – Principles and Paradigms “, 2/e, PHI, 2004.
2. Tom White, “Hadoop: The Definitive Guide”, 1/e, O’reilly, 2012.

v) REFERENCES

1. George Coulouris, Jean Dollimore, Tim Kindberg, Gordon Blair, *Distributed Systems: Concepts and Design*, 5th Edition, Pearson
2. Randy Chow Theodore Johnson, “Distributed Operating Systems and Algorithm

Analysis", Pearson Education, 2009.
3. Nancy A. Lynch, Morgan, " Distributed Algorithms", Kaufmann Publishers, Inc, 1996.
4. Eric Sammer, "Hadoop Operations: A Guide for Developers and Administrators", O'reilly, 2012.
5. Boris Lublinsky, Kevin T. Smith, Alexey Yakubovich, "Professional Hadoop Solutions", John Wiley, 2013.

vi) COURSE PLAN

Module	Contents	No. of hours
I	Distributed System: Overview, System Architecture, Processes –Threads – Virtualization – Clients – Servers – Code migration,Communication – Message Oriented – Stream Oriented – Multicast Communication,Naming – Flat – Structured – Attribute Based Naming.	8
II	Hadoop: Introduction – Comparison with Other Systems, Analyzing Data with Hadoop – Map and Reduce – Scaling Out – Data Flow –Combiner Functions, Hadoop Distributed File System – Concepts and Basic Operations.	9
III	Map Reduce Types – Input and Output Formats, Map Reduce Features– Counters – Sorting – Joins – Side Data Distribution, Administering Hadoop – Monitoring – Maintenance.Distributed Algorithms: Models of Distributed Computation –Preliminaries – Causality – Distributed Snapshots	9
IV	Synchronization and Election – Distributed Mutual Exclusion –Timestamp Algorithms – Voting – Fixed Logical Structure – PathCompression, Election – The Bully Algorithm.	9
V	Algorithms in General Synchronous Networks – Breadth First Search – Minimum Spanning Tree – Shortest Path–Maximal Independent Set.	10
	Total hours	**45**

Course Code	Course Name	Category	L	T	P	J	Credit	Year of Introduction
26CS162M	**ADVANCED OPERATING SYSTEMS**	**PEC**	**3**	**0**	**0**	**0**	**3**	**2026**

i) COURSE OVERVIEW: This course examines advanced operating system concepts with emphasis on multiprocessor and distributed operating systems. It covers system architectures, synchronization, scheduling, distributed coordination algorithms, resource management, database OS concurrency control, and protection mechanisms. The course also introduces mobile operating system architectures and real-world case studies, enabling students to analyse design, performance, and security issues in modern operating systems.

ii) COURSE OUTCOMES

After the completion of the course, the student will be able to:

CO1	Analyse design issues and resource management strategies in multiprocessor operating systems with respect to synchronization, scheduling, processor allocation, and memory management.	Analyse
CO2	Analyse distributed operating system mechanisms for communication, synchronization, and coordination to compare the performance of token-based and non-token-based algorithms.	Analyse
CO3	Use distributed file system and shared memory architectures to address resource allocation, consistency, and scheduling challenges in distributed environments.	Apply
CO4	Analyse concurrency and synchronization mechanisms used in database and distributed database systems to ensure correctness and performance.	Analyse
CO5	Apply protection models and security mechanisms to control access in operating systems, and use mobile operating system architectures to analyse security and system design through real-world case studies.	Apply

iii) TEXTBOOKS

1. M Singhal and NG Shivaratri , Advanced Concepts in Operating Systems, Tata McGraw Hill Inc, 2017
2. Andrew S. Tanenbaum, Herbert Bos, Modern Operating Systems, 5th Edition, Pearson, 2022.

iv) REFERENCES

1. Andrew S Tanenbaum, Distributed Operating Systems, Pearson Education Asia, 2002
2. Max Hailperin, Operating Systems and Middleware: Supporting Controlled Interaction. Creative Media Partners LLC, 2023.
3. Peterson, J.L. & Silbersehatz, A: Operating System Concepts, Addison, Wesley 2009 .

v) COURSE PLAN

Module	Contents	Hours
I	**Multiprocessor Operating Systems:** System Architectures- Structures of OS – OS design issues – Process synchronization – Processor Scheduling and Allocation- memory management.	8
II	**Distributed Operating Systems:** System Architecture Types- Design issues – Communication primitives – clock synchronization – mutual exclusion – election algorithms- Token and Non-Token Based Algorithms, A comparative performance analysis.	11
III	**Distributed Resource Management:** Distributed File Systems-Mechanisms for building Distributed File Systems-Design issues-Log structured File systems, Distributed shared memory-Architecture, Algorithms, Coherence Protocols, Distributed Scheduling	9
IV	**Database Operating Systems:** Requirements of a Database OS – Database Systems, The problem of concurrency control, Distributed Database systems – Synchronization primitives, Concurrency control algorithms	9
V	**Protection and Security:** The Access Matrix model-Implementations of Access Matrix-Safety in the Access Matrix model-Advanced Models of Protection. **Mobile Operating Systems**: Mobile OS Architectures - Underlying OS - Kernel structure and native level programming, Case studies on Mobile operating systems, Android, iOS, Samsung.	8
	Total Hours	**45**

Course Code	Course Name	Category	L	T	P	Credit	Year of Introduction
26CS162N	ADVANCED GRAPH THEORY	PEC	3	0	0	3	2026

i) COURSE OVERVIEW: Goal of this course is to impart deeper understanding in advanced concepts in graph theory and their practical applications.

ii) COURSE OUTCOMES

After the completion of the course, the student will be able to:

CO1	Apply Graph Theory concepts in practical scenarios.	Apply
CO2	Apply the concepts of Metrics, Convexity and Distance sequences on graphs.	Apply
CO3	Apply the concepts of Eulerian digraphs and hamiltonicity.	Apply
CO4	Solve problems based on graph coloring and flow graphs.	Apply
CO5	Apply the concepts of P, NP and NP completeness in real scenarios.	Apply

iii) SYLLABUS

Graphs, Connectivity and Hamiltonicity, Connectivity, The Center and Edge connectivity- Self Central Graphs - The Median – Central Paths - Other Generalized Centers, Extremal Distance Problems, Distance sequences, Matrices, Symmetry, Digraphs, Graph Algorithms, Critical Path Method.

iv) REFERENCES

1. Franco P. Preparata, Michael Ian Shamos, “Computational Geometry- An Introduction”, Texts and Monographs in Computer Science , Springer – Verlag.
2. N. Deo, *Graph Theory with Applications to Engineering and Computer Science*, New Delhi, India: PHI Learning, ISBN: 978-8120301450.
3. Mark de Berg, Otfried Cheong, Marc van Kreveld, Mark Overmars “ Computational Geometry, Algorithms & Applications” Springer.
4. Herbert Edelsbrunner, “Algorithms in Combinatorial Geometry”, EATCS Monographs on Theoretical Computer Science, Springer – Verlag.
5. Art Gallery Theorems, Joseph O’ Rourke, Oxford Press.

6. Joseph O' Rourke, " Computational Geometry in C", Cambridge University Press.

v) COURSE PLAN

Module	Contents	No. of hours
I	Graphs: Graphs as models- Paths and cycles - Graph Classes: Multi graphs - Complement graphs – Regular graphs – Complete graphs - Line graphs – Bipartite graphs –Planar graphs and properties – Wheel graphs – Tree and its properties – Subgraphs - Isomorphic graphs - Geodetic graphs - Graph operations. Graph Connectivity – Cut nodes and Blocks – Bridges -Vertex connectivity and edge connectivity - Menger's theorem - Properties of n-connected graphs – Circulant graphs.	9
II	Metrics on Graphs – Geodesic path- Eccentricity- Radius – Diameter- Center- Self-centered Graphs- The Median – Central Paths – Path Centers - Distance Heredity Graphs –The Center and Edge connectivity. Convexity: Closure Invariants - Symmetry: Groups- Symmetric - Distance Symmetry in graphs. Distance sequences: Degree Sequence- The Eccentric Sequence - Path Sequence - Other Sequences.	9
III	Digraphs: Digraphs and connectedness - Acyclic Digraphs.Matrices: The Adjacency Matrix - The incidence Matrix - The Distance Matrix.	7
IV	Eulerian Digraphs- Long paths in Digraphs- Tournaments. Graph Hamiltonicity: Necessary or sufficient conditions- Connectivity and Hamiltonicity- Graph operations and Hamiltonicity - Generations of Hamiltonicity.	10
V	Algorithms: Polynomial Algorithms and NP completeness - Maximum Matchings –Minimum Independent Set Problem – Minimum Vertex Cover Problem – Graph coloring. Networks: The Max- Flow Min-Cut Theorem	10
	Total hours	**45**

Course Code	Course Name	Category	L	T	P	Credit	Year of Introduction
22CS162O	**ESSENTIALS OF CYBER SECURITY**	**PEC**	**3**	**0**	**0**	**3**	**2026**

i) COURSE OBJECTIVES:

This course offers an overview of key information security concepts and vulnerabilities, and equips students with fundamental knowledge and skills to identify, assess, and defend against threats to web applications and network systems.

ii) COURSE OUTCOMES

After the completion of the course, the student will be able to:

CO1	Make use of a broad set of technical, social & political aspects of Cyber Security.	Apply
CO2	Identify various attacker techniques and motivations.	Apply
CO3	Compare various malicious codes.	Analyze
CO4	Analyze the various ways for securing devices.	Analyze
CO5	Distinguish various defense and analysis techniques.	Analyze

iii) SYLLABUS

Cyber Security Fundamentals- Attacker Techniques and Motivations- Malicious Code- Securing Devices- Defense and Analysis Techniques.

iv) REFERENCES

1. James Graham, Ryan Olson, and Rick Howard, eds. *Cyber security essentials*. CRC Press, 2016.
2. Charles J. Brooks, Christopher Grow, Philip Craig, Donald Short, *Cybersecurity Essentials*, 2018
3. William Stallings, *Network Security Essentials: Applications and Standards*. 6th ed., Pearson, 2018.

v)COURSE PLAN

Module	Contents	No. of hours
I	**Cyber Security Fundamentals** - Network and security concepts - Information Assurance Fundamentals- Authentication, Authorization,	8

	Nonrepudiation, Confidentiality, Integrity, Availability - Basic Cryptography - Symmetric Encryption - Public Key Encryption - DNS – Firewalls- Types of firewalls. Case Study - Windows Firewall.	
II	**Attacker Techniques and Motivations** - How hackers cover their tracks- Proxies -Types, Detection - Tunneling Techniques- HTTP, DNS, ICMP, Intermediaries and Steganography, Detection and Prevention - Fraud Techniques- Phishing, Smishing, Vishing, and Mobile Malicious Code, Rogue Antivirus, Click Fraud - Threat Infrastructure – Botnets, Fast-Flux, Advanced Fast-Flux.	9
III	**Malicious Code** - Self-Replicating Malicious Code- Worms, Viruses – Stealing Information and Exploitation - Form Grabbing - Man in the Middle Attack- Detection, Prevention - DLL Injection- Windows Registry DLL Injection, Injecting Applications, Reflective DLL Injections - Browser Helper Objects.	9
IV	**Securing Devices** - The three layers of Security- Defence in Depth, Need for layered security approach, Outer Perimeter Security, Network / Access Security, Host / Device Security - Securing Host Devices –- Securing Outer Perimeter Portals - Zero Trust Security - Limitations of perimeter-based security, Principles of Zero Trust Security, Architecture, Identity and device-based trust, Micro-segmentation and continuous verification.	10
V	**Defense and Analysis Techniques** - Memory Forensics – Honey Pots - Malicious Code Naming - Automated Malicious Code Analysis Systems - Passive Analysis, Active Analysis, Physical or Virtual Machines – Intrusion Detection Systems.	9
	Total Hours	**45**

Course Code	Course Name	Category	L	T	P	Credit	Year of Introduction
26CS162P	BLOCK CHAIN TECHNOLOGIES AND ITS APPLICATIONS	PEC	3	0	0	3	2026

i)COURSE OVERVIEW

This course introduces the fundamentals of blockchain and distributed systems, including consensus, fault tolerance, and cryptographic primitives. It covers major platforms such as Bitcoin, Ethereum, and Hyperledger Fabric, along with smart contracts, security, privacy, and scalability. The course also provides an overview of Web3, Layer-2 solutions, interoperability, and regulatory challenges.

ii) COURSE OUTCOMES

After the completion of the course, the student will be able to:

CO1	Apply the concepts of distributed record keeping to identify and address adversaries and fault models in decentralized systems.	Apply
CO2	Illustrate various crypto primitives used for realising the blockchain	Apply
CO3	Apply blockchain technologies to implement and utilize the core mechanisms used in Bitcoin and Ethereum systems.	Apply
CO4	Illustrate the primitives and technologies used in hyperledger fabric	Apply
CO5	Apply appropriate security and privacy mechanisms to identify, analyze, and mitigate privacy and security issues in blockchain systems using suitable preventive measures.	Apply

iii) SYLLABUS

Need for Distributed Record Keeping and Fault Models, Byzantine Generals Problem and Consensus Algorithms, CAP Theorem and Blockchain Trade-offs, Permissioned vs Permissionless Blockchains, Basic Distributed Computing and Cryptographic Primitives, Bitcoin and Cryptocurrency Systems, Consensus Mechanisms and Data Models, Ethereum and Smart Contracts, Smart Contract Security and DApps, Hyperledger Fabric and Permissioned Blockchains, Privacy and Security in Blockchain, Attacks and Advanced

Consensus (Algorand, Sharding), Layer-2 Scaling, Interoperability, Regulatory Challenges, and Introduction to Web3.

iv) REFERENCES:

1. Draft version of "S. Shukla, M. Dhawan, S. Sharma, S. Venkatesan, 'Blockchain Technology: Cryptocurrency and Applications', Oxford University Press, 2019.
2. Josh Thompson, 'Blockchain: The Blockchain for Beginnings, Guild to Blockchain Technology and Blockchain Programming', Create Space Independent Publishing Platform, 2017.
3. Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller, and Steven Goldfeder. Bitcoin and cryptocurrency technologies: a comprehensive introduction. Princeton University Press, 2016. (Free download available)
4. Joseph Bonneau et al, SoK: Research perspectives and challenges for Bitcoin and cryptocurrency, IEEE Symposium on security and Privacy, 2015 (article available for free download)

v) COURSE PLAN

Module	Contents	No. of hours
I	Need for Distributed Record Keeping, Modeling faults and adversaries Byzantine Generals problem, Consensus algorithms and their scalability problems, Blockchain based cryptocurrency, Technologies Borrowed in Blockchain – hash pointers, consensus, byzantine fault-tolerant distributed computing, digital cash .CAP Theorem and Blockchain Trade-offs. Permissioned vs Permissionless Blockchains. Basic Distributed Computing, Atomic Broadcast, Consensus, Byzantine Models of fault tolerance,	9
II	Basic Crypto primitives- Hash functions, puzzle-friendly hash, collision-resistant hash, digital signatures, public-key cryptography, Elliptic Curve Cryptography, verifiable random functions, Zero-knowledge systems. Merkle trees and Merkle Proofs Bitcoin blockchain, challenges, and solutions, proof of work, Proof of stake, alternatives to Bitcoin consensus, Bitcoin scripting language and their use. UTXO model vs Account-based model	10

III	Ethereum and Smart Contracts, Smart Contract Vulnerabilities, Gas optimization and cost model. The Turing Completeness of Smart Contract Languages and verification challenges, Using smart contracts to enforce legal contracts, Decentralized Applications(Dapps) architecture, comparing Bitcoin scripting vs. Ethereum Smart contracts	10
IV	Hyperledger fabric, Hyperledger Fabric architecture, Chaincode lifecycle, the plug and play platform and mechanisms in permissioned blockchain	10
V	Privacy, Security issues in Blockchain : Pseudo-anonymity vs. anonymity, Zcash and Zk-SNARKS for anonymity preservation, attacks on Blockchains – such as Sybil attacks, selfish mining, 51% attacks - advent of algorand, and Sharding based consensus algorithms Layer 2 Scaling Solutions, Blockchain Interoperability, Regulatory and Legal Challenges in Blockchain, Introduction to Web3.	8
	Total hours	**45**

Course Code	Course Name	Category	L	T	P	Credit	Year of Introduction
26CS162Q	SOFTWARE PROJECT MANAGEMENT	PEC	3	0	0	3	2026

i) COURSE OVERVIEW

This course aims to characterize software projects and develop an understanding of key project management activities. It provides knowledge of software estimation techniques and management practices essential for effective planning and execution. The course also emphasizes monitoring and controlling software projects, managing people, and building efficient teams to ensure successful project delivery.

ii) COURSE OUTCOMES

After the completion of the course, the student will be able to:

CO1	Demonstrate software project management concepts using appropriate life cycle models.	Understand
CO2	Identify software cost estimation techniques to estimate effort, cost, and schedule for software projects.	Apply
CO3	Utilize software quality management principles, metrics, and standards to plan, assess, and assure software quality.	Apply
CO4	Make use of configuration, risk, defect, and cost management techniques along with software metrics to monitor and control software projects.	Apply
CO5	Make use of project evaluation and estimation techniques to assess feasibility, risks, and effectiveness of software projects using modern management approaches.	Apply

iii) SYLLABUS

Software project management concepts including project life cycles, planning, tracking, and control; software cost estimation and project evaluation techniques; software quality, configuration, risk, defect, and cost management with relevant metrics; and emerging trends such as process improvement models, cloud-based software, virtualization, containers, and microservices architecture.

iv) REFERENCES:

1. Ramesh Gopalaswamy, "Managing and global Software Projects", Tata McGraw Hill Tenth Reprint, 2011.
2. Roger Pressman, "Software Engineering- A Practitioner's Approach ", 7th Edition, McGraw Hill, 2018.
3. Daniel Galin, "Software Quality Assurance: from Theory to Implementation", AddisonWesley, 2008.
4. Bob hughes and mike cotterell, "Software project Management" fifth edition,2018.
5. Royce, W. "Software Project management: A Unified Framework", Addison-Wesley, 6th print ,2008.

v) COURSE PLAN

Module	Contents	No. of hours
I	Project Concepts and management Project life cycle models-ISO 9001 model-Capability Maturity Model-Project Planning-Project tracking-Project closure. Evolution of Software Economics – Software Management Process Framework: Phases, Artifacts, Workflows, Checkpoints – Software Management Disciplines: Planning / Project Organization and Responsibilities / Automation / Project Control – Modern Project Profiles	12
II	Cost Estimation Problems in Software Estimation – Algorithmic Cost Estimation Process, Function Points, SLIM (Software Life cycle Management), COCOMO II (Constructive Cost Model) Estimating Web Application Development – Concepts of Finance, Activity Based Costing and Economic Value Added (EVA) – Balanced Score Card.	12
III	Software Quality Management Software Quality Factors – Software Quality Components – Software Quality Plan – Software Quality Metrics – Software Quality Costs – Software Quality Assurance Standard – Certification – Assessment	12
IV	Software Management and Metrics	8

	Software Configuration Management – Risk Management: Risk Assessment: Identification / Analysis / Prioritization – Risk Control: Planning / Resolution / Monitoring – Failure Mode and Effects Analysis (FMEA) – Defect Management – Cost Management. Software Metrics – Classification of Software Metrics: Product Metrics: Size Metrics, Complexity Metrics, Halstead's Product Metrics, Quality Metrics, and Process metrics	
V	Project Evaluation and Emerging trends Strategic Assessment–Technical Assessment–Cost Benefit Analysis–Cash Flow Forecasting– Cost Benefit Evaluation Technique–Risk Evaluation–Software Effort Estimation. Emerging Trends: Import of the internet on project Management –people Focused Process Models.	8
	Total Hours	**45**

Course Code	Course Name	Category	L	T	P	Credit	Year of Introduction
26CS162R	ADVANCED CLOUD COMPUTING	PEC	3	0	0	3	2026

i) COURSE OVERVIEW

To equip students with in-depth knowledge of cloud computing architectures, virtualization and container technologies, cloud networking and resource management, security mechanisms, and emerging cloud paradigms for designing and deploying scalable cloud solutions.

ii) COURSE OUTCOMES

After the completion of the course, the student will be able to:

CO1	Explain cloud computing concepts, service and deployment models, characteristics, challenges, and business adoption strategies.	Understand
CO2	Explain virtualization and containerization techniques, including VM placement, migration, clustering, and Docker ecosystem used in cloud resource management.	Understand
CO3	Apply cloud networking concepts and microservices architecture in designing Virtual Private Clouds, REST-based services, and scalable cloud solutions.	Apply
CO4	Apply cloud security mechanisms and big data processing techniques using NoSQL databases, Hadoop, and MapReduce frameworks.	Apply
CO5	Explain advanced cloud computing paradigms such as edge computing, fog computing, DevOps tools, and emerging research trends.	Understand

iii) SYLLABUS

Cloud Computing Fundamentals, Cloud Service and Deployment Models, Virtualization and Containerization, VM Management Techniques, Cloud Networking and Microservices, Virtual Private Cloud Design, Cloud Storage and Databases, Resource Provisioning and Scheduling, Auto-Scaling and Load Balancing, Cloud Security and Compliance, Big Data Processing on Cloud (NoSQL, Hadoop, MapReduce), Serverless Computing, DevOps and Cloud Tools, Edge and Fog Computing, Distributed File Systems, Public Cloud Platforms, Emerging Trends.

iv) REFERENCES:

1. Thomas Erl, Zaigham Mahmood, Ricardo Puttini. Cloud Computing: Concepts, Technology & Architecture (2nd Edition)-2023
2. Rajkumar Buyya, Christian Vecchiola, S. Thamarai Selvi. Mastering Cloud Computing (2nd Edition)-2023
3. Lizhe Wang, Rajiv Ranjan, Jinjun Chen and Boualem Benatallah, Cloud Computing (1 ed.), CRC Press, 2017. ISBN 978-1351833097.
4. Judith S. Hurwitz and Daniel Kirsch, Cloud Computing For Dummies (2 ed.), Hoboken: John Wiley &Sons, 2020.

v)COURSE PLAN

Module	Contents	No. of hours
I	Cloud Computing, Adoption of cloud-based IT resources, Service Models: Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), Software-as-a-Service (SaaS), Deployment models: Public Cloud, Private Cloud, Hybrid Cloud, Community Cloud, Cloud Computing Characteristics, Challenges of cloud computing.	9
II	Virtualization concept, Types of virtualization, Demo of virtualization, Virtualization merits and demerits, Role of virtualization in cloud computing, VM placement, VM migration and demo, VM clustering, Design issues in VM clustering, Need of Dockers and Containers, Docker ecosystem, Hypervisor vs Docker.	10
III	Microservices, Service-Oriented Architecture, REST API, IP addressing, Subnetting and Supernetting, Designing Virtual Private Cloud (VPC), VPC demo, VPC peering, VPC case study, Cloud storage, Serverless computing, Cloud API Gateway, Cloud databases, Resource provisioning, Time-shared and space-shared models, Efficient VM consolidation on cloud servers, Task and DAG scheduling algorithms (Min-Min, Max-Min, MET, B-level, T-level), Task–VM mapping, Auto-scaling, Load balancing.	9
IV	Cloud market analysis, Security and compliance, Shared security model in IaaS, PaaS and SaaS, Shared technology issues, Data loss or leakage, Account or service hijacking, Implementation of cloud security, Security groups, Network Access Control Lists, Parallel query execution with NoSQL databases, Big Data, Handling Big Data on cloud platforms,	9

	MapReduce framework using Hadoop, Design of data applications based on MapReduce.	
V	Comparative study of public clouds, Edge computing, Fog computing, Data offloading, Cloud-based DevOps tools, Task partitioning, Data partitioning, Data synchronization, Distributed file systems, Data centers, Ongoing research topics in cloud computing.	8
	Total Hours	**45**

LABORATORY COURSE

Course Code	Course Name	Category	L	T	P	Credit	Year of Introduction
26CS169A	ALGORITHM DESIGN LABORATORY	LBC	0	0	3	2	2026

i) COURSE OBJECTIVES: The course aims to offer students a hands-on experience on different advanced data structures

ii) COURSE OUTCOMES

After the completion of the course the student will be able to:

CO1	Implement different advanced data structures.	Apply
CO2	Implement a primality testing algorithm.	Apply
CO3	Implement algorithm to find factors of a given number.	Apply
CO4	Implement pattern matching algorithms.	Apply
CO5	Analyze flow in a network using different algorithms.	Apply
CO6	Apply randomized algorithms to find mincut in a network and to perform selection sort.	Apply
CO7	Implement algorithms to find convex hull from a set of points.	Apply

iii) SYLLABUS

Advanced data structures: binomial heap, Fibonacci heap, disjoint sets, String matching algorithm, Randomized algorithms, Geometric Algorithms, Network flow algorithms, Integer factorization, Primality testing.

iv) REFERENCES

1. T. H. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein, “Introduction to algorithms”, Prentice-hall of India Private Limited, New Delhi, 2010.

v) COURSE PLAN

Sl No	Contents	No. of hours
1	Binomial heap	4

2	Fibonacci heap	4
3	Disjoint sets	3
4	Primality testing	2
5	Integer factorization	2
6	Rabin-Karp algorithm	2
7	Knuth-Morris-Pratt algorithm	3
8	Dinic's algorithm	3
9	Push-relabel algorithm	4
10	Relabel-to-front algorithm	4
11	Pseudo random generator	2
12	Randomized min-cut algorithm	2
13	Randomized selection algorithm	2
14	Graham's scan algorithm	4
15	Jarvis march algorithm	4
	Total hours	**45**

Course Code	Course Name	Category	L	T	P	Credit	Year of Introduction
26CS169B	**FULL STACK DEVELOPMENT LABORATORY**	**LBC**	**0**	**0**	**3**	**2**	**2026**

i) COURSE OBJECTIVES: The course aims to offer students a hands-on experience on different advanced data bases.

ii) COURSE OUTCOMES

After the completion of the course the student will be able to:

CO1	Design static and dynamic web pages using HTML and CSS.	Apply
CO2	Develop interactive and dynamic web applications using JavaScript, DOM, and jQuery.	Apply
CO3	Build front-end applications using React JS.	Apply
CO4	Implement authentication and CRUD operations in web applications.	Apply
CO5	Integrate databases and cloud services using MongoDB and Firebase for data storage, authentication, and application management.	Apply
CO6	Develop, version-control, and deploy full-stack web applications using GitHub and modern full-stack technologies.	Apply

iii)SYLLABUS

Design and development of static and dynamic web pages using HTML, CSS, and JavaScript, implementation of interactive features using jQuery, development of single-page applications using React JS, integration of databases and cloud services such as Firebase, and building complete full-stack web applications including authentication, CRUD operations, and mini projects.

iv) REFERENCES

1. Ethan Brown, *Web Development with Node and Express*, O'Reilly Media, 2018.
2. Robin Nixon, *Learning PHP, MySQL & JavaScript*, O'Reilly Media, 2018.
3. Jon Duckett, *HTML and CSS: Design and Build Websites*, Wiley, 2014.
4. Alex Banks and Eve Porcello, *Learning React*, O'Reilly Media, 2020.

v)COURSE PLAN

Sl No	Contents	No. of hours
1	Creating static web pages using HTML	2
2	Designing responsive web pages using HTML and CSS	2
3	Develop a web page that allows users to switch between light mode and dark mode.	2
4	Client-side scripting using JavaScript	2
5	Interactive web applications using JavaScript and DOM	2
6	Web application development using jQuery	3
7	Form validation and UI components using jQuery	2
8	Front-end application development using React JS	2
9	Building single-page applications using React JS	2
10	Authentication and login modules using React JS	2
11	Develop a React application that filters items dynamically based on user input.	3
12	Database integration and CRUD operations using MongoDB	2
13	Create a simple blog management system without backend	2
14	Cloud database integration using Firebase	2
15	Developing REST-based web applications	2
16	Fetch and display data from a public REST API.	2
17	Building content management systems	3
18	Developing e-commerce web applications	3
19	Mini project using full-stack technologies with MongoDB, Firebase, and GitHub (version control, collaboration, deployment workflow)	5
	Total hours	**45**

INDUSTRY ELECTIVE

Course Code	Course Name	Category	L	T	P	Credit	Year of Introduction
26CS166A	CYBER FORENSICS BASICS	IEC	3	0	0	3	2026

i) COURSE OBJECTIVES:

To develop students with the knowledge of digital forensic principles, legal requirements, and professional procedures, enabling them to effectively acquire, validate, and analyze digital evidence using forensic tools across diverse environments, including email and mobile devices, and to prepare comprehensive investigation reports.

ii) COURSE OUTCOMES:

After the completion of the course the student will be able to:

CO1	Demonstrate digital forensic principles, and professional procedures to conduct cyber forensic investigations.	Understand
CO2	Make use of appropriate acquisition methods and forensic tools to manage digital evidence.	Apply
CO3	Make use of forensic tools to analyze file systems and operating environments across multiple platforms.	Apply
CO4	Identify methods to collect, validate, and analyze forensic data, including handling hidden data, file formats, compression, live acquisitions, and network forensics.	Apply
CO5	Develop the skills in investigating email and mobile device crimes, using forensic tools and preparing professional investigation reports.	Apply

iii) SYLLABUS

Introduction to Computer Forensics, Understanding Computer Investigations , Requirements for forensic lab certification , Data Acquisition , Processing Crime and Incident Scene, Analysis and validation , Recovering Graphics Files, Network Forensics , Email Investigations , Report writing for high tech investigations.

iv)REFERENCES

1. Bill Nelson, Amelia Phillips, Frank Enfinger, Christofer Steuart , “Computer Forensics and Investigations”, Second Indian Reprint , Cengage Learning India Private Limited,2009.

2.Eoghan Casey – “Digital Evidence and Computer Crime”, 3rd Edition (2018)

v)COURSE PLAN

Module	Contents	No. of hours
I	Introduction to Digital Forensics, History of Digital Forensics, Understanding Case Law, Developing Digital Forensics Resources, Preparing for Digital Investigations, Understanding Law Enforcement Agency Investigations, Maintaining Professional Conduct, Procedures for Private-Sector High-Tech Investigations, Understanding Data Recovery Workstations and Software, Conducting an Investigation, Determining the Physical Requirements for a Digital Forensics Lab	9
II	Understanding Storage Formats for Digital Evidence, Determining the Best Acquisition Method, Contingency Planning for Image Acquisitions, Using Acquisition Tools, Validating Data Acquisitions, Performing RAID Data Acquisitions, Using Remote Network Acquisition Tools, Using Other Forensics Acquisition Tools	9
III	Working with Windows and CLI Systems, Understanding File Systems, Exploring Microsoft File Structures, Examining NTFS Disks, Understanding Whole Disk Encryption, Understanding the Windows Registry, Understanding Microsoft Startup Tasks, Understanding Virtual Machines, Digital Forensics Software and Hardware Tools, Validating and Testing Forensics Software, Examining Linux File Structures, Understanding Macintosh File Structures	9
IV	Determining What Data to Collect and Analyze, Validating Forensic Data, Addressing Data-Hiding Techniques, recognizing a Graphics File, Understanding Data Compression, Identifying Unknown File Formats, Understanding Copyright Issues with Graphics, Performing Live Acquisitions, Network Forensics	9
V	Role of E-mail in Investigations, Roles of the Client and Server in E-mail, Investigating E-mail Crimes and Violations, Understanding E-mail Servers, Using Specialized E-mail Forensics Tools, Mobile Device Forensics, Acquisition Procedures for Mobile Devices, Cloud Forensics, Report Writing for High-Tech Investigations, Guidelines for Writing Reports	9
	Total Hours	**45**

Course Code	Course Name	Category	L	T	P	Credit	Year of Introduction
26CS166B	STREAM PROCESSING AND ANALYTICS	PEC	3	0	0	3	2026

i) COURSE OBJECTIVES

This course introduces students to the architecture of streaming data processing systems. It provides an understanding of end-to-end solutions for the cost-effective analysis and visualization of streaming data using various open-source technologies available in this domain. The course also equips students with the knowledge and skills to implement and apply appropriate algorithms and data structures for streaming applications.

ii) COURSE OUTCOMES

CO1	Apply streaming data systems to process and utilize real-time data effectively.	Apply
CO2	Analyze the architecture of streaming data systems to understand data flow, components, and system design.	Analyze
CO3	Apply algorithmic techniques used in streaming data systems to process and manage continuous data streams efficiently.	Apply
CO4	Analyze tools and techniques required for streaming data analytics to select appropriate platforms and methods for real-time analysis.	Analyze

iii) SYLLABUS

Scalable Streaming Data Systems, Streaming Data Systems Architecture, Streaming Data Frameworks, Streaming Analytics, Advanced Streaming Applications.

iv) REFERENCES

1. Andrew G. Psaltis, *Streaming Data: Understanding the Real-Time Pipeline*. Manning Publications, 2017.
2. Byron Ellis. *Real-Time Analytics: Techniques to Analyze and Visualize Streaming Data*. Wiley, 2014.

v) COURSE PLAN

Module	Contents	No. of hours
I	**Thinking about Data Systems:** Reliable, Scalable and Maintainable Data Applications, Properties of Data, Scaling with the traditional databases, Big Data Systems, Desired properties of Big Data Systems, Data Model for Big Data, Generalized Big Data System Architecture, Real time systems, Difference between Batch processing and Stream Processing, Difference between real time and streaming systems.	9
II	**Generalized Streaming Data Architecture:** Lambda Architecture, Kappa Architecture, Streaming Data system Component, Features of Real time Architecture, A real time architecture checklist, Service Configuration and Coordination Systems, Maintaining the state, Apache ZooKeeper, Data Flow Manager, Managing distributed data flows with Apache Kafka, Apache Flume.	10
III	**Kafka Fundamentals:** Overview, Use-Cases and applications, Architecture, Kafka Topics, Producer and Consumer Using CLI, Programming Kafka, Simple Kafka Producer, Simple Kafka Consumer Producer, Consumer Configuration, Producer, Consumer Execution, Kafka Consumer Groups.	9
IV	**Streaming Data Processor Concepts:** Timing Concepts, Windowing, Joins, Storage for Streaming Data, NoSQL storage Systems, Choosing a Storage technology, Delivery of Streaming Metrics	8
V	**Apache Spark Streaming:** Spark Streaming fundamentals, Motivation, Difference between Spark Streaming API and Spark API, Architecture, Components of Spark Engine, Spark Application Architecture, Fault Tolerance, Comparison with Traditional Streaming Systems	9
	Total hours	**45**

Course Code	Course Name	Category	L	T	P	Credit	Year of Introduction
26AC061A	**RESEARCH METHODOLOGY AND IPR**	**RM**	**2**	**0**	**0**	**2**	**2026**

i) COURSE OBJECTIVES

This course is intended to prepare the M. Tech students to carry out their dissertation/ research project work effectively, with a research bias. The student will be able to formulate a viable research problem, do a critical analysis of publications in the area of research, and identify a research method suitable for the work. The student will achieve the capability to write a technical paper based on his/her dissertation/ research project.

ii) COURSE OUTCOMES

After the completion of the course the student will be able to:

CO1	Explain research ethics, Citation, Impact factor and Plagiarism.	Apply
CO2	Formulate a research problem, make a suitable research design, and identify the data collection methods.	Apply
CO3	Analyse the collected data.	Analyse
CO4	Explain the role of IPR and Patent law in fostering research work, leading to creation of improved products, thus supporting economic growth and social benefits.	Apply
CO5	Write a technical paper for publication.	Apply

iii) SYLLABUS

Introduction to Research Methodology- motivation for research, types of research, ethical issues. Identifying a research area and collecting related literature. Research problem-scope-objectives, literature review, identifying research gaps, and formulate the research problem. Research design and methods, data collection and analysis. Copy right – royalty - IPR and patent law. Process of patenting and development, Procedure for grant of patents. Copy left- open access, citation, plagiarism, iImpact factor. Writing a technical paper.

iv) REFERENCES

1. Stuart Melville and Wayne Goddard, *Research methodology: an introduction for science & engineering students.*
2. Ranjit Kumar, 2nd Edition, *Research Methodology: A Step by Step Guide for beginners*.
3. Ramappa T.*, Intellectual Property Rights Under WTO*, S. Chand, 2008.
4. Robert P. Merges, Peter S. Menell, Mark A. Lemley, *Intellectual Property in New Technological Age*, 2016.
5. Mayall, *Industrial Design*, McGraw Hill, 1992. Niebel, "Product Design", McGraw Hill, 1974.

v) COURSE PLAN

Module	Contents	Hours
I	Introduction to Research Methodology: Motivation towards research, Types of research. Professional ethics in research: Ethical issues, ethical committees. Identification of major conferences and important journals in a chosen area of interest. Collection of at least 10 published papers on a research problem in the chosen area.	6
II	Defining and formulating the research problem: Literature Survey, Analysing the collected papers to understand how the authors haveidentified the research gaps, arrived at their objectives, and formulated their research problem. Understanding how their research work is different from the previous works in the chosen area.	6
III	Research design and methods: Analyzing the collected papers to understand how the authors have formulated the research methods, both analytical methods and experimental methods. Data Collection and analysis: Analyzing the collected papers to understand the methods of data collection, data processing, analysis strategies, and tools used for analyzing the data.	6
IV	Copy right - royalty - Intellectual property rights and patent law – Process of Patenting and Development, Procedure for grant of patents. Reproduction of published material: Copy left- Open access, Citation and acknowledgement. Plagiarism, Impact factor.	6

V	Technical writing - Structure and components of a typical technical paper, abstract and conclusion, illustrations and tables, bibliography, referencing and footnotes. Writing a technical paper – based on the identified research problem, and using the collected papers, Literature survey, Problem formulation, and Research design, and a hypothetical result.	6
	Total hours	**30**

PROJECT

Course Code	Course Name	Category	L	T	P	Credit	Year of Introduction
26CS167A	**MINI PROJECT**	**PR**	**0**	**0**	**6**	**3**	**2026**

COURSE OBJECTIVES

To make students

1. Collect the recent publications related to the identified Mini project.
2. Do a detailed study of the Mini project based on current journals, published papers and books.
3. Present a seminar based on the Mini project.
4. Improve the writing and presentation skills.
5. Design and develop a system or application in the area of their specialization.

APPROACH

1. Students shall make a presentation for 20-25 minutes based on the detailed study on the project and submit a report of the study.
2. There will be two interim progress review of the Mini project work. The first review will focus on the topic, objectives, methodology, design and expected results.
3. The second review shall focus on the work/ Implementation and results obtained.

EXPECTED OUTCOME

Upon successful completion of the Mini project and Seminar, the student should be able to

1. Identify and solve various problems associated with designing and implementing a system or application.
2. Test the designed system or application.
3. Improve the writing and presentation skills.
4. Explore domains of interest so as to pursue the course project

Course Code	Course Name	Category	L	T	P	Credit	Year of Introduction
26CS178A	**PROJECT PHASE I**	**PR**	**0**	**0**	**24**	**16**	**2026**

To make students

1) Do an original and independent study on the area of specialization.
2) Explore in depth a subject of his/her own choice.
3) Start the preliminary background studies towards the project by conducting literature survey in the relevant field.
4) Broadly identify the area of the project work, familiarize with the tools required for the design and analysis of the project.
5) Plan the experimental platform, if any, required for project work.

APPROACH

1) There will be three interim progress review of the Project (Phase I). The first review shall focus on the topic, and objectives. This review will be conducted within one month of the commencement of third semester classes.
2) The second review shall focus on the methodology. This review will be conducted within two months of the commencement of third semester classes.
3) The third review shall focus on the design and expected results, and scope of the work which has to be accomplished in the fourth semester. This review will be conducted towards the close of the third semester.

EXPECTED OUTCOME

Upon successful completion of the Project (Phase I), the student should be able to

1) Identify the topic, objectives and methodology to carry out the project.
2) Finalize the project plan for their course project.

Course Code	Course Name	Category	L	T	P	Credit	Year of Introduction
26CS178B	PROJECT PHASE II	PR	0	0	24	16	2026

To continue and complete the project work identified in Project (Phase I).

APPROACH

1) There will be three interim progress review of the Project (Phase II). The first review shall focus on the progress of the implementation of the design made in Project (Phase I). This review will be conducted within one month of the commencement of third semester classes.
2) The second review shall focus on the quality and quantum of the work completed. This review will be conducted within two months of the commencement of third semester classes.
3) The third review shall focus on the completed implementation and the results. This review will be conducted towards the close of the third semester.
4) At least one technical paper has to be prepared and published in journals conferences based on their project work.

EXPECTED OUTCOME

Upon successful completion of the Project (Phase II), the student should be able to

1) Get a good exposure to a domain of interest.
2) Get a good domain and experience to pursue future research activities.