CURRICULUM STRUCTURE

THIRD YEAR UG: B.TECH

COMPUTER SCIENCE AND ENGINEERING

REVISION: FRCRCE-3-26

Effective from Academic Year 2026-27
Board of Studies Approval: 4/3/26
Academic Council Approval: 27/3/26

Dr. DEEPAK BHOIR	**Dr. JAGRUTI SAVE**	**DR. SAPNA PRABHU**
Dean Academics	**HOD (CSE)**	**Principal**

Institute Vision

To be a leading institution in education, transforming students into globally competent professionals with strong ethical values, innovation capabilities, and a commitment to sustainable societal development.

Institute Mission

- Develop industry-ready engineers with strong academic foundations, practical skills, and an entrepreneurial mindset capable of addressing industry and societal needs.
- Cultivate a culture of innovation and research to address real-world challenges through interdisciplinary approaches and sustainable practices.
- Foster a vibrant industry–academia ecosystem by actively engaging faculty and students in knowledge exchange, collaborative learning and professional skill development.

Department Vision

To achieve excellence in Computer Science education by nurturing technically competent and socially responsible professionals who solve real-world problems.

Department Mission

1. Facilitate an excellent scholastic environment for students and faculty to promote research, innovation and interdisciplinary learning.
2. Foster a culture of experiential learning and sustainable practices through projects, internships and industry engagement.
3. Promote participation in co-curricular and extra-curricular activities to build integrity, communication, teamwork and leadership.

Program Educational Objectives (PEO)

Graduates will be able to

1. Design, develop and implement innovative computing solutions that promote sustainable development and socio-economic growth.
2. Pursue lifelong learning and advanced education by adapting to evolving technologies and diverse career opportunities.
3. Exhibit ethical conduct, leadership, and effective interpersonal skills in their professional domain.

Program Specific Outcomes (PSO)

Students will be able to

1. Apply computational thinking, algorithmic design and software development principles to build efficient computing solutions.

2. Develop scalable and intelligent computing solutions using emerging technologies such as Artificial Intelligence, Data Science and Distributed Systems to solve complex problems.

Preamble:

Fr Conceicao Rodrigues College of Engineering (Fr. CRCE) is an Autonomous Institute from the year 2024-25. The University Grant Commission vide letter No. F. 2-10/2023(AC-Policy) dated 23rd Nov 2023 conferred the autonomous status to Fr. Conceicao Rodrigues College of Engineering, Fr. Agnel Ashram, Bandstand, Bandra (West), Mumbai 400050 affiliated to the University of Mumbai for a period of 10 years from the Academic year 2024-2025 to 2033-2034 as per clause 7.5 of the UGC (Conferment of Autonomous Status Upon Colleges and Measures for Maintenance of Standards in Autonomous Colleges) Regulations, 2023. We look towards Autonomy as a great opportunity to design and implement a curriculum sensitive to the needs of Learners, Indian Society, and Industries. We are committed to the effective implementation of the UGC Regulations and NEP 2020 in its spirit. The Government of Maharashtra has directed Autonomous Colleges to revise their curriculum in line with the National Education Policy (NEP) 2020 through a Government Resolution dated 4th July 2023. Accordingly, degree options are given to the students admitted from the academic year 2024-25 based on the UGC circulars and the DTE guidelines ref no. 17/DTE/NEP-2020/2024/111 dated 4th June 2024 related to the implementation of NEP.

Based on the recent recommendations of the GR, we are pleased to offer our holistic curriculum, a "H-Tree Model" of Engineering Education. A unique "H-Tree Model" of Engineering Education Curriculum is carefully designed to systematically develop IQ (Intelligence Quotient), PQ (Physical Quotient), EQ (Emotional Quotient) and SQ (Spiritual Quotient) of a learner. This curriculum aims at the development of a well-rounded personality through a holistic approach to education in which the learner receives 25% teacher-led learning, 25% peer learning, 25% self-learning and 25% experiential learning. The curriculum model is outcome-based that focuses on learning by doing. The curriculum is designed to provide multiple learning opportunities for students to acquire and demonstrate competencies for rewarding careers. It offers curated, interest-driven pathways that empower learners to acquire skills through structured, strategic planning. It has 7 verticals aligned to the GR recommendations with a strong science and mathematics foundation and Program core, Sequence of electives, Multidisciplinary Minor courses, Humanities & Management courses along with sufficient experiential learning through projects and a semester-long industry / research internship along with employable skill-based courses. Learners get an opportunity to acquire skills through NSDC-aligned courses during the summer vacations. Additionally, learners can choose from multiple degree pathways, including a built-in Multidisciplinary Minor, a Double Minor in emerging fields, or Honors with Research.

The curriculum integrates emerging industry trends with skill-based learning to foster innovation and analytical problem-solving. It offers flexible, multidisciplinary course choices with a strong emphasis on experiential and project-based learning. The Program Core Courses comprehensively cover the fundamental and advanced areas of Computer Science and Engineering, including Data Structures and Algorithms, Artificial Intelligence, Machine Learning, Data Science, Cybersecurity, Software Engineering, Computer Networks, Operating Systems, and other department-specific specializations.

Various steps are taken to transform the teaching-learning process to make learning a joyful experience for students. We believe that this curriculum will raise the bar of academic standards with the active involvement and cooperation from students, academic and administrative units.

Curriculum Structure for UG Programs at Fr CRCE w.e.f. A.Y. 2026-27

Nomenclature of the courses in the curriculum	
Abbreviation	**Title**
BSESC	Basic Science & Engineering Science Courses
PCPEC	Program Core and Program Elective Courses
MDC	Multidisciplinary Courses
SC	Skill Courses
HSSM	Humanities, Social Sciences and Management
EL	Experiential Learning
LLC	Liberal Learning Courses
BSC	Basic Science Courses
ESC	Engineering Science Courses
PCC	Program Core Courses
PEC	Program Elective Courses
MDM	Multidisciplinary Minor
OE	Open Elective
VSEC	Vocational and Skill Enhancement Course
VSC	Vocational Skill Courses
SEC	Skill Enhancement Courses
AEC	Ability Enhancement Course
EEMC	Entrepreneurship, Economics and Management Course
IKS	Indian Knowledge System
VEC	Value Education
RM	Research Methodologies
CEFP	Community Engagement or Field Project
ELC	Experiential Learning Courses
PRJ	Project
INT	Internship
CC	Cocurricular Courses
HMM	Honors and Multidisciplinary Minor
DM	Double Minor
HR	Honors with Research

Credit Specification:

- Theory: 1 credit=13 to 15 hrs of teaching
- Lab: 1 Credit=26 to 30 hrs of lab work
- Studio Activities: 1 Credit= 26 to 30 hrs of creative activities
- Workshop Based Activities: 1 Credit=26 to 30 hrs of hands-on activities related to vocation/professional practice/skill based
- Seminar/Group Discussion: 1 Credit=13 to 15 hrs of participation
- Internship: 1 Credit=Per 2 weeks OR 36 to 40 hrs of engagement
- Field Based Learning/Practices: 1 Credit=26 to 30 hrs of learning activities
- Community Engagement Projects: 1 Credit=26 to 30 hrs of contact time along with 13 to 15 hrs of activities preparation, report writing, independent reading etc.

Credit requirements for four different options of the Degrees:

Degree/SEM	I	II	III	IV	V	VI	VII	VIII	Total
B.Tech with Multidisciplinary Minor	20	20	22	22	22	22	20	20	168
B.Tech with Double Minor (Multidisciplinary & Specialisation Minor)	20	20 +2*	22 +4*	22 +4*	22 +4*	22 +4*	20 +2$	20 +2$	188
B.Tech with Honors with Research (Multidisciplinary Minor)	20	20 +2*	22 +4*	22 +4*	22 +4*	22 +4*	20 +2$	20 +2$	188

*Optional Credits
$ optional 2 credits can be earned either in VII or VIII Semester

1. Learners who earn a minimum of total 168 credits will be awarded "B.Tech in Engg. /Tech. with Multidisciplinary Minor (MDM)" degree.
2. Learners will have the following options to earn B. Tech. in Engg. /Tech. degree in
    a. Major Engg./Tech Discipline with Double Minor (Multidisciplinary and Specialization Minor)
    b. Major Engg./Tech Discipline with Honors with Research and Multidisciplinary Minor
3. Major Engg./Tech Discipline with Double Minor (Multidisciplinary and Specialization Minor) (additional 20 credits): 168 +18+2 =188 Min Credits. There will be four courses (4 credits each), one in each semester starting from the III semester which will be from emerging areas of specialisation. In VII or VIII semester students will complete 2 credits seminar/project. Admission eligibility min CGPA=7.5 after First year
4. B.Tech in Engg./ Tech.- Honors with Research and Multidisciplinary Minor (additional 20 credits by research): 168 +18+2=188 Min Credits. (Admission eligibility: min CGPA=7.5 after First and should maintain CGPA=7.5 after Third year)
5. Learner can earn the certificate/Diploma/Degree based on his/her exit from the program as follows. College shall explore feasibility to offer NSDC aligned skill-based courses to the learners:
    a. UG Certificate: After a one-year (40 credits to be earned) and 8-credits summer workshop/vocational courses/internship
    b. UG Diploma: After two-years (80 credits to be earned) and 8-credits summer workshop/vocational courses/internship/Project
    c. B.Voc.: After three-years (120 credits to be earned) and 8-credits summer workshop/vocational courses/internship/Project

Salient Features of Curriculum:

- ✓ Framed as per Government Resolution dated 4th July 2023 in line with National Education Policy (NEP) 2020.
- ✓ Minimum 168 choice-based credit structure with options of Degrees earning additional credits
- ✓ Unique 'H-Tree' Model of Curriculum: Hybrid model for holistic development with happy learning environment having bridge connecting verticals providing unique path for each learner for 3-dimensional growth, Life Long Learning, multiple entry-exit, inclusive model indicating equal distribution of central resources
- ✓ More emphasis on laboratory based and experiential learning
- ✓ More weightage to continuous assessment to reduce examination stress
- ✓ Mandatory Semester-long internship, courses with emotional & spiritual learning and skill-based learning aligned with NSDC framework
- ✓ Well balanced curriculum to attain Program Outcomes and skills of 21st century learner
- ✓ Curriculum is designed to create excitement among learners for education through stories, activities, collaboration, hackathon, contest, case studies, creative art etc.
- ✓ Curriculum is designed to make graduates responsible citizens of country with future ready skills to handle challenges of 21st Century

Society of St. Francis Xavier, Pilar's
Fr. Conceicao Rodrigues College of Engineering
Fr. Agnel Ashram, Bandstand, Bandra (W), Mumbai – 400 050
(Autonomous College affiliated to University of Mumbai)

Curriculum Structure for UG Programs at Fr CRCE
Third Year Computer Science and Engineering Program

SEM-V

Course Code	Course Vertical	Sub-Vertical	Course Name		Contact Hours	Examination Marks (1 Credit=50 Marks)					Credits	
						ISE	MSE	ESE				
								Min	Max	Total	Points	Total
25PCC13CS21	PCPEC	PCC	Operating System	TH	2	20	30	20	50	100	2	3
				PR	2	50	-	-	-	50	1	
				SL	2							
25PCC13CS22	PCPEC	PCC	Software Engineering	TH	2	20	30	20	50	100	2	3
				PR	2	50	-	-	-	50	1	
				SL	2							
25PCC13CS13	PCPEC	PCC	Artificial Intelligence	TH	2	20	30	20	50	100	2	3
				PR	2	50	-	-	-	50	1	
				SL	2							
25PCC13CS14	PCPEC	PCC	Machine Learning	TH	2	20	30	20	50	100	2	3
				PR	2	50	-	-	-	50	1	
				SL	2							
25PCC13CS15	PCPEC	PCC	Theoretical Computer Science	TH	2	20	30	20	50	100	2	3
				TU	1	50	-	-	-	50	1	
				SL	3							
25PEC13CS1X	PCPEC	PEC	Program Elective Course	TH	2	20	30	20	50	100	2	3
				PR	2	50	-	-	-	50	1	
				SL	2							
25OE13CS3X	MDC	OE	1. Health, Wellness and Psychology 2. Emotional and Spiritual Intelligence	TH	2	100	-	-	-	100	2	2
				SL	2							
25MDMXX4	MDC	MDM	MDM Course-4	TH	2	20	30	20	50	100	2	2
				SL	2							
25DMX3	DM	DM	Double Minor Course	TH	2	20	30	20	50	100	2	4*
				TU	2	50	-	-	-	50	2	
				SL	4							
25HR04	HR	HR	Honors with Research	-	-	-	-	-	-	-	4	4*
				Total	**TH:TU:PR 16:1:10=27**			**-**	**-**	**1100**	**-**	**22**

SEM-VI

Course Code	Course Vertical	Sub-Vertical	Course Name		Contact Hours	Examination Marks (1 Credit=50 Marks)					Credits	
						ISE	MSE	ESE				
								Min	Max	Total	Points	Total
25PCC13CS16	PCPEC	PCC	Cryptography and Computer Security	TH	2	20	30	20	50	100	2	3
				PR	2	50	-	-	-	50	1	
				SL	2							
25PCC13CS17	PCPEC	PCC	Data Warehousing and Mining	TH	2	20	30	20	50	100	2	3
				PR	2	50	-	-	-	50	1	
				SL	2							
25PCC13CS18	PCPEC	PCC	Cloud Computing	PR	2	50	-	-	-	50	1	1
25PCC13CS19	PCPEC	PCC	Deep Learning	TH	2	20	30	20	50	100	2	3
				PR	2	50	-	-	-	50	1	
				SL	2							
25PCC13CS20	PCPEC	PCC	Software Testing Lab	PR	2	20	-	30	-	50	1	1
25PEC13CS2X	PCPEC	PEC	Program Elective Course	TH	2	20	30	20	50	100	2	3
				TU	1	50	-	-	-	50	1	
				SL	3							
25PEC13CS3X	PCPEC	PEC	Program Elective Course	TH	2	20	30	20	50	100	2	3
				TU	1	50	-	-	-	50	1	
				SL	3							
25PEL13CS1X	PCPEC	PEC	Program Elective Lab	PR	2	50	-	-	-	50	1	1
25OE13CS4	MDC	OE	Public Relations and Corporate Communication	TH	2	100	-	-	-	100	2	2
				SL	2							
25MDMXX5	MDC	MDM	MDM Course-5	TH	2	50	-	50	-	100	2	2
				SL	2							
25DMX4	DM	DM	Double Minor Course	TH	2	20	30	20	50	100	2	4*
				TU	2	50	-	-	-	50	2	
				SL	4							
25HR05	HR	HR	Honors with Research	-	-	-	-	-	-	-	4	4*
				Total	**TH:TU:PR 14:2:12=28**			**-**	**-**	**1100**	**-**	**22**

List of Program Elective Courses:

Track-1:

SEM-V: **25PEC13CS11:** Big Data Analytics

SEM-VI: **25PEC13CS21:** Social Media Analytics

25PEC13CS31: Graph Data Science

SEM-VI: Lab:

25PEL13CS11: knowledge Representation and Ontology Lab

Track-2:

SEM-V: **25PEC13CS12:** Blockchain

SEM-VI: **25PEC13CS22:** AI in Cyber Security

25PEC13CS32: FinTech

SEM-VI: Lab:

25PEL13CS12: Generative AI Lab

Track-3:

SEM-V: **25PEC13CS13:** Image Processing

SEM-VI: **25PEC13CS23:** UI/UX Design

25PEC13CS33: Computer Vision

SEM-VI: Lab:

25PEL13CS13: Soft Computing Lab

List of Multidisciplinary Minor Courses (MDM):

		Course Code	**Course Name**
Minor in Business Management	MDM Course-4	25MDMBM4	Human Resource Management
	MDM Course-5	25MDMBM5	Digital Marketing
Minor in Healthcare Management	MDM Course-4	25MDMHM4	Digital Transformation in HealthCare
	MDM Course-5	25MDMHM5	Bioinformatics and Computational Biology
Minor in Design	MDM Course-4	25MDMDE4	Interaction Design
	MDM Course-5	25MDMDE5	Mobility and Vehicle Design
Minor in Communication Engineering	MDM Course-4	25MDMCM4	Communication and Computer Networks
	MDM Course-5	25MDMCM5	Mobile Communication and Computing
Minor in Mechanical Engineering	MDM Course-4	25MDMME4	Industrial Engineering
	MDM Course-5	25MDMME5	Supply Chain Management

Course Code	Course Name	Teaching Scheme (Hrs/week)				Credits Assigned			
		L	T	P	SL	L	T	P	Total
		2	--	2	2	2	--	1	3
25PCC13CS21	Operating System	Examination Scheme							
			ISE	MSE		ESE		Total	
						Min	Max		
		Theory	20	30		20	50	100	
		Lab	50	--		--	--	50	

Pre-requisite Courses	Discrete Maths, Programming Fundamentals, Data structure	
After the successful completion students should be able to:		
Course Outcomes	CO1	Comprehend the primitive concepts of Operating System functionality and services.
	CO2	Apply concurrency and synchronization techniques in software development.
	CO3	Implement algorithms for memory management.
	CO4	Evaluate various algorithms of File Storage & I/O management.
	CO5	Analyze advanced operating system architectures and functionalities through case studies of modern systems.

Module No.	Topics	Ref	Hrs
1	**Overview of Operating Systems-** Role, functions, and evolution of operating systems, Types of OS, OS Architecture, Components of an OS-Kernel, Shell, and File System, Processes- Definition, lifecycle, and Process Control Block (PCB), Threads vs. Processes.	1,2, 3, 4	2
2	**Process Management** CPU Scheduling- Goals, criteria, and types of scheduling, Scheduling algorithms, Process Synchronization- Critical sections and race conditions, Semaphores, Monitors, and Mutex, Deadlock Handling, IPC Mechanism.	1,2,3,4	8
3	**Memory Management** Contiguous and Non-Contiguous Allocation, Paging and Segmentation, Page replacement algorithms, Virtual Memory, Memory Allocation Techniques-Fixed, Variable Partitioning, and Buddy System.	1,2,3,4	7
4	**File Systems and I/O Management** File Systems - File attributes, directory structures, and access methods, File allocation techniques: Contiguous, Linked, and Indexed, Disk Scheduling- FCFS, SSTF, SCAN, C-SCAN, I/O Management-Device drivers, interrupts, and buffering.	1,2,3,4	5

5	**Advances in Modern Operating Systems** Case Studies- Cloud and Mobile OS, Real-Time and Edge OS, AI and OS, Modern Linux-Based Systems, Experimental OS, Communication in Distributed Systems, Synchronization in Distributed Systems, Security in distributed systems	5,6,7,8,9, 10	4
		TOTAL	**26**

Self-Learning:

1. Self-learning hours include MOOCs, spoken tutorials, online resources, and extended study hours to enhance independent learning and better understanding of each module of the course content.
2. Evaluation of the self-learning components is carried out in all the evaluation components.

Course Assessment: -

Theory:

ISE:

ISE activities carry 20 marks. These activities will be conducted throughout the semester.

MSE:

The written summative examination of 30 marks based on 50% syllabus for 90 minutes.

ESE:

The written summative examination will be conducted for 50 marks based on the complete syllabus (20% questions on syllabus covered before MSE and 80% questions on the remaining syllabus) for 120 minutes.

Module No.	Sr. no	Suggested List of experiments (Any 8 experiments can be conducted)	Ref	Hrs
2	1	Implement basic scheduling algorithms	1,2,3,4	2
2	2	Simulate producer-consumer synchronization.	1,2,3,4	2
2	3	Simulate a system with processes and resources to detect and resolve deadlocks using a resource allocation graph.	1,2,3,4	2
3	4	Write a program to simulate page replacement algorithms	1,2,3,4	2
3	5	Write a program to simulate memory allocation techniques.	1,2,3,4	2
4	6	Simulate file allocation techniques	1,2,3,4	2
4	7	Implement disk scheduling algorithms.	1,2,3,4	2
4	8	Implement buffering techniques for a simulated I/O device to manage data streams efficiently.	1,2,3,4	2
5	9	Analyze Linux kernel logs for specific events (scheduling, I/O operations) using tools like dmesg or syslog.	9	2
5	10	Explore OS vulnerabilities using a controlled virtual environment. Analyze patching or mitigation strategies.	5,6,7,8, 9	2
5	11	Simulator based experiments (EduMIPS64, GAIL (General Algorithm Interactive Learning) ,NS-3	10	2
3	12	Implement loadable kernel modules to extend the xv6 kernel to replace or extend subsystems of the xv6 kernel. For example, make the file system a kernel module so that you can add a kernel module to read DOS file systems, or replace the xv6 file system.	9	3

Course Assessment: -
Lab:

ISE: Laboratory ISE is divided into two components: 25 marks for submission of experiments and 25 marks for oral/practical evaluation.

Recommended Books:

1. Silberschatz A., Galvin P., Gagne G. "Operating Systems Principles", Willey Eight edition
2. Achyut S. Godbole , Atul Kahate "Operating Systems" McGraw Hill Third Edition
3. "Operating System-Internal & Design Principles", William Stallings, Pearson
4. Andrew S. Tanenbaum, "Modern Operating System", Prentice Hall.
5. "Cloud Computing: Concepts, Technology & Architecture" by Thomas Erl, Ricardo Puttini, and Zaigham Mahmood
6. "Mobile Operating Systems: Concepts and Practices" by Dr. R. Latha and S. Pavithra
7. "Embedded and Real-Time Operating Systems" by K.C. Wang
8. "Quantum Computing: A Gentle Introduction" by Eleanor Rieffel and Wolfgang Polak
9. "Linux Kernel Development" by Robert Love
10. Official Website of GAIL on GitHub,NS-3 Official Website,EduMIPS64 Official Website

Online Resources:

1. http://www.nptelvideos.in/2012/11/compiler-design.html
2. https://www.coursera.org/lecture/nand2tetris2/unit-4-1-syntax-analysis-5pC2Z
3. https://onlinecourses.nptel.ac.in/noc21_cs72/preview
4. https://www.scaler.com/topics/course/free-operating-system-course/

Suggested CO - PO articulation Matrix

Course Outcomes	**Programme Outcomes (POs)**										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	2										
CO2	2	2	2	2	3						
CO3	2	2	2	2	3						
CO4	2	2	2	2	3						
CO5		2		2				3	3		3

Legends :- High: 03, Medium: 02,Low: 01, No Mapping: -

Blooms level

Remember	Understand	Apply ✓	Analyze ✓	Evaluate	Create

Course Code	Course Name	Teaching Scheme (Hrs/week)				Credits Assigned			
		L	T	P	SL	L	T	P	Total
25PCC13CS22	Software Engineering	2	--	2	2	2	--	1	3

Examination Scheme					
	ISE	MSE	ESE		Total
			Min	Max	
Theory	20	30	20	50	100
Lab	50	--	--	--	50

Pre-requisite Courses		Object Oriented Programming with Java
After the successful completion students should be able to:		
Course Outcomes	CO1	Apply appropriate software engineering process models for effective software project development.
	CO2	Analyze and specify software requirement of a software system in an SRS document.
	CO3	Apply strategies to effectively plan, schedule, and monitor project progress.
	CO4	Apply software design concepts to create architectural, user interface, and component-level designs using appropriate patterns and styles.
	CO5	Identify risks and manage the change to assure quality in software projects.
	CO6	Apply testing strategies and techniques to a software system.

Module No.	Unit No.	Topics	Ref.	Hrs.
1		**Introduction To Software Engineering and Process Models:**	1,2,5, 6	6
	1.1	Software engineering- a layered technology, Process Framework and Umbrella Activities, Capability Maturity Model (CMM)		
	1.2	Prescriptive Process Models: Waterfall model, Incremental Process Models, Evolutionary Process Models: RAD, Spiral, Prototyping		
	1.3	Agile process model: Agile Principles, Scrum, Kanban, Extreme Programming (XP)		
	1.4	AI-assisted Software Development- Introduction to AI in software engineering, AI-assisted coding tools (Github Copilot, CodeWhisperer), Role of AI in agile development, Benefits and limitations (productivity vs. over-reliance), Ethical concerns (Code ownership, plagiarism)		
2		**Software Requirements Analysis and Modelling:**	1,2,4	4
	2.1	Types of requirements- Functional and Nonfunctional requirements, system and user requirements, requirement elicitation methods, Requirement Engineering process.		
	2.2	Software Requirement Specification document format (IEEE), Using AI tools to generate SRS drafts, refine requirements and		

		detect ambiguity, Prompt engineering for requirement generation		
	2.3	Requirement Modelling: Scenario based Models, Behavioral models, Data flow diagram.		
3		**Project estimation, Tracking and Scheduling:**	1,6	5
	3.1	Project Estimation techniques-LOC, FP, COCOMO II		
	3.2	Project Scheduling and Tracking: Defining a Task Set for the Software Project, Gantt Chart, Program Evaluation Review Techniques (PERT), Tracking the Schedule		
	3.3	AI for project estimation and planning- AI assisted estimation tools, Using AI for effort prediction, risk prediction and schedule generation		
4		**Design Engineering**	1,3,6	6
	4.1	Software design concepts, design model		
	4.2	Architectural Design, User interface design, Component level design, Architectural styles and Patterns.		
	4.3	AI-assisted design and code generation- AI for UML generation from text, AI for code generation from design, AI for UI prototyping (Figma AI)		
5		**Software Risk & Quality Management**	1,2,5, 6	5
	5.1	Software Risk, Types of Risk, Risk Identification, Risk Assessment, Risk Projection, RMMM Plan, AI for bug prediction, risk identification		
	5.2	Software Quality Assurance Task, SCM process, change and version control, Formal Technical Review (FTR), Walkthrough, AI for code quality analysis. AI in DevOps pipelines		
6		**Software Testing**	1,2.6	4
	6.1	Unit testing, Integration testing, Validation testing, System testing		
	6.2	Testing Techniques: white-box testing- Basis path, Control structure testing Black-box testing: Graph based, Equivalence partitioning, Boundary value analysis		
	6.3	Types of Software Maintenance, Re-Engineering, Reverse Engineering		
	6.4	AI-based testing techniques- AI for test generation, AI for automated testing and defect prediction, AI in CI/CD pipelines		
			Total	**30**

Self-Learning:

1. Self-learning hours include MOOCs, spoken tutorials, online resources, and extended study hours to enhance independent learning and better understanding of each module of the course content.
2. Evaluation of the self-learning components is carried out in all the evaluation components.

Course Assessment: -

Theory:

ISE:

ISE activities carry 20 marks. These activities will be conducted throughout the semester.

MSE:

The written summative examination of 30 marks based on 50% syllabus for 90 minutes.

ESE:

The written summative examination will be conducted for 50 marks based on the complete syllabus (20% questions on syllabus covered before MSE and 80% questions on the remaining syllabus) for 120 minutes.

Module No.	Exp. No.	Suggested List of experiments	Ref.	Hrs.
1	**1**	Apply different process models (Waterfall, Iterative, Incremental and Spiral) to a selected case study and compare their effectiveness in terms of time, flexibility, and quality.	1,2	2
2	**2**	Gather requirements for a sample software system using requirement elicitation methods and document them in IEEE format. (SRS document) (Students generate SRS using AI, manually refine it and validate it).	1,2,4	3
	3	Develop UML diagrams (use case, activity, sequence, DFD etc.) for a selected case study. Design UML diagrams using AI tools and validate correctness.	1,2,4	3
3	**4**	Estimate size, cost and effort for a selected case study using function point method.	1.6	2
	5	Estimate size, cost and effort for a selected case study using COCOMO Model.	1.6	2
	6	Create a project plan using Gantt and PERT for a software development project and track progress.	1.6	2
	7	Use any project management tool like JIRA, Trello, Asana etc. to learn project management principles and manage the lifecycle of a selected case study. Use AI plugins or automation suggestions in JIRA. Compare manual vs. AI-generated task breakdown.	1.6	3
4	**8**	Design the architecture of a selected case study using architectural styles and patterns.	1,4	2
	9	Design a user interface for a selected case study focusing on usability and accessibility	1.4	3
	10	Create a component level design of a chosen software system.	1,4	2
5	**11**	Conduct a risk assessment for a software project and create a Risk Mitigation and Management Plan (RMMM). Use AI tools to identify risks in project case study and compare manual vs. AI risk analysis.	1	2
	12	Experiments based on manual and automated software testing. Generate test cases using AI. Compare coverage with manual testing.		

Course Assessment: -
Lab:

ISE: Laboratory ISE is divided into two components: 25 marks for submission of experiments and 25 marks for oral/practical evaluation

Recommended Books:

1. Roger Pressman, "Software Engineering: A Practitioner's Approach", 9th edition , McGraw-Hill Publications, 2019
2. Ian Sommerville, "Software Engineering", 9th edition, Pearson Education, 2011
3. Ali Behfrooz and Fredeick J. Hudson, "Software Engineering Fundamentals", Oxford University Press, 1997
4. Grady Booch, James Rambaugh, Ivar Jacobson, "The unified modeling language user guide", 2nd edition, Pearson Education, 2005
5. Pankaj Jalote, "An integrated approach to Software Engineering", 3rd edition, Springer, 2005
6. Rajib Mall, "Fundamentals of Software Engineering", 5th edition, Prentice Hall India, 2014
7. Jibitesh Mishra and Ashok Mohanty, "Software Engineering", Pearson, 2011
8. Ugrasen Suman, "Software Engineering – Concepts and Practices", Cengage Learning, 2013
9. Waman S Jawadekar, "Software Engineering principles and practice", McGraw Hill Education, 2004

Online Resources:

1. Software Engineering specialization by the Hong Kong university of Science and Technology https://www.coursera.org/specializations/software-engineering
2. https://nptel.ac.in/courses/106/105/106105182/
3. https://onlinecourses.nptel.ac.in/noc19_cs69/preview
5. https://www.mooc-list.com/course/software-engineering-introduction-edx
6. https://www.geeksforgeeks.org/software-engineering/
7. https://www.atlassian.com/software/jira
8. Terragni, V., et al. (2025). The Future of AI-Driven Software Engineering. Published in ACM Transactions on Software Engineering and Methodology (TOSEM)
9. https://github.com/DrAlzahraniProjects/csusb_fall2024_cse6550_team4/wiki/Github-Copilot
10. Almutaz (2025): Artificial Intelligence (AI) in Scaled Agile Framework (SAFe). Published in the International Journal of Applied Information Systems.
11. Umar, M. A., Lano, K., & Abubakar, A. K. (2025). Automated requirements engineering framework for agile model-driven development. Frontiers in Computer Science
12. MDPI. (2025). Artificial Intelligence in Project Success: A Systematic Literature Review. Information, 16(8), 682
13. Alenezi, M., & Akour, M. (2025). AI-Driven Innovations in Software Engineering: A Review of Current Practices and Future Directions. *Applied Sciences*, *15*(3), 1344

Suggested CO - PO articulation Matrix

Course Outcomes	Programme Outcomes (POs)										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	3									
CO2	3	3						3	3	3	3
CO3	3	3			3						
CO4	3	3	3	3	3			3	3	3	3
CO5	3	3						3	3	3	3
CO6	3	3									

Legends:- High: 03, Medium: 02, Low: 01, No Mapping: -

Blooms level

Remember	Understand ✓	Apply ✓	Analyze ✓	Evaluate	Create

Course Code	Course Name	Teaching Scheme (Hrs/week)				Credits Assigned			
		L	T	P	SL	L	T	P	Total
25PCC13CS13	Artificial Intelligence	2	--	2	2	2	--	1	3
		Examination Scheme							
						ESE			
			ISE	MSE		Min	Max	Total	
		Theory	20	30		20	50	100	
		Lab	50	--		--	--	50	

Pre-requisite Courses		Programming Fundamentals, Data Structures
After the successful completion students should be able to:		
Course Outcomes	CO1	Identify suitable agent architectures and search techniques to solve basic AI problems.
	CO2	Solve constraint satisfaction problem using appropriate AI technique
	CO3	Apply appropriate knowledge representation and inference methods to given AI problems.
	CO4	Use suitable AI methods to find solution of given planning and learning problems.
	CO5	Apply communication and perception methods to given AI problems.
	CO6	Explain Responsible AI concepts and basic Explainable AI (XAI) techniques.

Module No.	Unit No.	Topics	Ref.	Hrs.
		Introduction to Artificial Intelligence and Intelligent Agents	1	3
1	**1.1**	Definition of AI, Philosophy of AI- thinking and acting humanly, thinking and acting rationally.		
	1.2	The nature of environments- fully and partially observable environment, single and multi-agent, deterministic and stochastic, episodic and sequential, static and dynamic, discrete and continuous		
	1.3	Structure of agents- agent programs and types of agent programs- simple reflex agent, model-based agent, goal based agent, utility-based agent		
2		**Problem Solving by Searching**	1,2	8
	2.1	Problem solving agents, problem formulation and example problems		
	2.2	Uninformed search strategies- BFS, Uniform Cost Search, DFS, Depth Limited, Iterative Deepening DFS, Bidirectional Search		
	2.3	Informed search strategies- Heuristic function, Greedy Best First Search, A* Search		
	2.4	Local search strategies- Hill Climbing Search, Simulated		

		Annealing, Local Beam Search, Genetic Algorithm		
	2.5	Adversarial Search- Minimax algorithm, alpha-beta pruning		
3		**Constraint Satisfaction Problems (CSP)**	1,2	3
	3.1	Defining CSP, Inference in CSPs		
	3.2	Backtracking search in CSPs		
	3.3	Local search for CSPs		
4		**Knowledge Representation and Reasoning**	1,2	7
	4.1	Knowledge representation systems, syntax and semantics using FOPL		
	4.2	Inference using forward chaining, backward chaining and resolution		
	4.3	Reasoning under uncertainty- Basics of probability and Bayes Theorem		
	4.4	Inference using Bayesian Networks		
5		**Planning and Learning**	1,2	3
	5.1	Planning process, components of planning system, total and partial order planning, hierarchical planning		
	5.2	What is learning? types of learning- supervised, unsupervised, semi-supervised, ensemble and reinforcement learning		
6		**Communication and Perception**	2	2
	6.1	Introduction to Natural Language Processing- Steps in the process including morphological analysis, syntactic analysis, semantic analysis, discourse integration, pragmatic analysis		
	6.2	Perception- vision, speech recognition		
			Total	**26**

Exp. No.	**Suggested List of experiments**
1	Design of an AI agent specifying PEAS description, type of environment in which the agent performs the task and block diagram of the agent for given problem statement.
2	To solve trivial AI problems using Prolog.
3	To solve given AI problem using informed and uninformed search. Compare the performance of both the techniques.
4	To solve given AI problem using adversarial search technique.
5	To solve given Constraint Satisfaction Problem using appropriate AI technique.
6	Apply SAT solvers like DPLL, WalkSAT algorithms to solve given problem using appropriate knowledge representation scheme.
7	Use Bayesian network to infer from the given knowledge base in uncertain environment.
8	To solve a given planning problem using appropriate technique.
9	Mini Project covering areas of AI like communication, perception, learning etc. It is recommended to make group of 2-3 students and make them solve real world problem.

Self-Learning:

1. Self-learning hours include MOOCs, spoken tutorials, online resources, and extended study hours to enhance independent learning and better understanding of each module of the course content.
2. Evaluation of the self-learning components is carried out in all the evaluation components.

Course Assessment: -
Theory:

ISE:
ISE activities carry 20 marks. These activities will be conducted throughout the semester.
MSE:
The written summative examination of 30 marks based on 50% syllabus for 90 minutes.
ESE:
The written summative examination will be conducted for 50 marks based on the complete syllabus (20% questions on syllabus covered before MSE and 80% questions on the remaining syllabus) for 120 minutes.

Course Assessment: -
Lab:

ISE: Laboratory ISE is divided into two components: 25 marks for submission of experiments and 25 marks for oral/practical evaluation.

Recommended Books:

1. Stuart Russell and Peter Norvig, "Artificial Intelligence: A Modern Approach", Prentice Hall, 3rd Edition.
2. Elain Rich, Kevin Knight, and Shivashankar Nair, "Artificial Intelligence", McGraw Hill Education, 3rd Edition.
3. Responsible AI: Implementing Ethical and Unbiased Algorithms – Sray Agarwal & Shashin Mishra
4. Explainable Artificial Intelligence: An Introduction to Interpretable Machine Learning – Uday Kamath & John Liu

Online Resources:

1. https://onlinecourses.nptel.ac.in/noc22_cs56/
2. https://ocw.mit.edu/courses/6-034-artificial-intelligence-fall-2010/
3. https://onlinecourses.nptel.ac.in/noc25_cs118/

Suggested CO - PO articulation Matrix

Course Outcomes	**Programme Outcomes (POs)**										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	2									
CO2	3	3	2	2							
CO3	3	2	3								
CO4	2	3		2							
CO5	3	2									
CO6	3					2		3			2

Legends :- High: 03, Medium: 02, Low: 01, No Mapping: -

Blooms level

Remember	Understand ✓	Apply ✓	Analyze ✓	Evaluate	Create

Course Code	Course Name	Teaching Scheme (Hrs/week)				Credits Assigned			
		L	T	P	SL	L	T	P	Total
25PCC13CS14	Machine Learning	2	--	2	2	2	--	1	3
		Examination Scheme							
			ISE	MSE	ESE		Total		
					Min	Max			
		Theory	20	30	20	50	100		
		Lab	50	--	--	--	50		

Pre-requisite Courses		Mathematics
After the successful completion students should be able to:		
Course Outcomes	CO1	Apply supervised learning algorithms to solve classification and regression problems and evaluate model performance using appropriate metrics.
	CO2	Apply Probabilistic Learning to perform probabilistic inference and reasoning under uncertainty.
	CO3	Design and implement feed forward neural networks using gradient-based learning and back propagation algorithms for classification and regression task.
	CO4	Apply clustering and dimensionality reduction techniques for unsupervised data analysis.

Module No.	Unit No.	Topics	Ref.	Hrs.
1		**Introduction to Machine Learning**	2,5	2
	1.1	Fundamental of ML: Introduction to machine learning concepts, Applications of machine learning, Machine Learning Pipeline, Types of data and problem formulation.		
	1.2	Types of Learning: Supervised/Unsupervised/Reinforcement Learning, Classification/ Regression.		
2		**Supervised Learning**	1, 2,5	8
	2.1	K-Nearest Neighbour (KNN): Working Principle, Distance Metrics, Curse of Dimensionality. Decision Trees: Learning by Induction, Tree Representation, DT Learning algorithm, Quantifying uncertainty: Entropy, Information Gain, Gini Index, ID3 Vs CART, Overfitting and pruning.		
	2.2	Support Vector Machines: Maximum Margin Classifier, Hard Margin Vs. Soft Margin, Kernel Trick, Linear vs. Non-linear SVM.		
	2.3	Ensemble Methods: Bagging, Random Forest, Boosting (Adaboost), Gradient Boosting		
	2.4	Evaluating a model: Bias, Variance, Cross-validation, Confusion Matrix, ROC Curve, Out-of-Bag metric		
3		**Probabilistic Learning**	1,3	6

	3.1	Bayesian Learning: Bayes Theorem, Generative vs. Discriminative Models, Naïve Bayes learning (Types and Assumptions)		
	3.2	Bayesian Networks: Building Bayesian Networks, Inference in BNs, D-Separation, Serial, Diverging and Converging Connections.		
4	**Neural Networks**		1, 6	6
	4.1	Perceptron, Activation functions, Perceptron learning algorithm, Linearly Vs. Non-linearly separable patterns, From perceptron to Sigmoid Neuron: Need for differentiable activation, Sigmoid neuron. Delta Learning rule (Gradient based learning)		
	4.2	Multi-layer Perceptron (MLP), Training Feedforward Neural network, Computing Gradients, Error Back-propagation.		
	4.3	Optimization Techniques: Gradient Descent, Mini batch Gradient Decent, Stochastic Gradient Descent, Convergence Analysis.		
	4.4	Regularization: Training Error, Generalization Error, Overfitting in Linear Models, Regularization Techniques: Ridge Regression, Lasso Regression.		
5	**Un-supervised Learning**		1,2	6
	5.1	Clustering: K-means Clustering, Convergence and Applications, Hard Vs. Soft Clustering, Gaussian Mixture Models (GMM), Expectation Maximization (EM) Algorithm Cluster Evaluation Metrics: Need for evaluation, cohesion and separation, Silhouette Score, Elbow Method (WCSS), Davies–Bouldin Index.		
	5.2	Dimensionality Reduction: Curse of Dimensionality, Principal Component Analysis (PCA), Applications of PCA.		
			Total	**28**

Self-Learning:

1. Self-learning hours include MOOCs, spoken tutorials, online resources, and extended study hours to enhance independent learning and better understanding of each module of the course content.
2. Evaluation of the self-learning components is carried out in all the evaluation components.

Course Assessment: -

Theory:

ISE:

ISE activities carry 20 marks. These activities will be conducted throughout the semester.

MSE:

The written summative examination of 30 marks based on 50% syllabus for 90 minutes.

ESE:

The written summative examination will be conducted for 50 marks based on the complete syllabus (20% questions on syllabus covered before MSE and 80% questions on the remaining syllabus) for 120 minutes.

To be Taught in laboratory			
	Topics	**Ref.**	**Hrs.**
1	**K-Nearest Neighbour** Write a program to implement k-Nearest Neighbour algorithm to classify the iris data set. Print both correct and wrong predictions.	4,7	2
2	**Decision Tree:** Write a program to demonstrate the working of the decision tree based ID3/CART algorithm. Use an appropriate data set for building the decision tree and apply this knowledge to classify a new sample.	4,8	2
3	**Naïve Bayes Classifier:** Implement the **Naïve Bayes classifier** to classify English text documents into predefined categories (e.g., spam/ham or positive/negative sentiment). Preprocess the text data by performing tokenization, stop-word removal, and converting text into numerical features using Bag of Words or TF-IDF representation. Train the Naïve Bayes model, evaluate its performance using accuracy and confusion matrix, and classify a new unseen text sample.	4,7	2
4	**Ensemble Learning:** To implement and compare different **ensemble learning techniques** for improving the performance of machine learning models on a classification problem.	4,7	2
5	**Perceptron:** To implement the Perceptron algorithm for binary classification and analyze its performance on linearly separable data.	4,7	2
6	Training Feed forward Neural Network using Back propagation: To design and implement a feedforward neural network using gradient-based learning and the Backpropagation algorithm for solving classification and regression problems.	4,7	2
7	**SVM:** To implement Support Vector Machine (SVM) using different kernel functions and analyze non-linear decision boundaries.	4,7	2
8	**K-Means Clustering:** Implement the K-Means clustering algorithm to group unlabeled data points into K distinct clusters based on similarity. Apply the algorithm on an appropriate dataset (e.g., Iris dataset or a customer segmentation dataset) and visualize the formed clusters. Analyze the effect of different values of K on cluster formation and evaluate clustering performance using appropriate metrics (e.g., inertia or silhouette score).	4,7	2
9	**PCA:** Implement Principal Component Analysis (PCA) to reduce the dimensionality of a given high-dimensional dataset while preserving maximum variance. Apply PCA on an appropriate dataset and transform the data into lower dimensions (2D or 3D) for visualization. Analyze the explained variance ratio and compare model performance before and after dimensionality reduction.	4,7	2
10	**Mini Project - Real-World Applications of Machine Learning:**	4,7	8

	To design and implement a complete machine learning solution for a real-world problem by integrating data preprocessing, model building, evaluation, result analysis and interpretation.		
		Total	**26**

Note: Please note that the datasets and models referenced in the experiments may be subject to change. These are only suggested datasets and models. Students are encouraged to explore alternative datasets and models, in consultation with the subject teacher.

Course Assessment: -
Lab:

ISE: Laboratory ISE is divided into two components: 25 marks for submission of experiments and 25 marks for oral/practical evaluation.

Recommended Books:

1. Christopher Bishop: Pattern Recognition and Machine Learning, Springer-Verlag New York Inc., 2006.
2. Tom M. Mitchell: Machine Learning, The McGraw-Hill, Indian Edition, 2017.
3. Kevin Murphy: Machine Learning: A Probabilistic Perspective, MIT Press, 2012
4. Shai Shalev-Shwartz and Shai Ben-David: Understanding Machine Learning: From Theory to Algorithms, Cambridge University Press, 2014
5. Ethem Alpaydin: Introduction to Machine Learning, 3rd Edition, MIT Press, 2014
6. Satish Kumar "Neural Networks A Classroom Approach", Tata McGraw-Hill.
7. Aurelien Geron: Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow, O'Reilly, 2019.
8. Ian Goodfellow, Yoshua Bengio, Aaron Courville. - Deep Learning, MIT Press Ltd, 2016
9. Mukherjee, Animesh; Kulshrestha, Juhi; Chakraborty, Abhijnan; Kumar, Srijan — *Ethics in Artificial Intelligence: Bias, Fairness and Beyond*, Springer Singapore, 2023

Online Resources:

1. Google Machine Learning crash course: https://developers.google.com/machine-learning/crash-course
2. NPTEL course on Introduction to Machine Learning: https://onlinecourses.nptel.ac.in/noc23_cs18/preview
3. IBM course on Machine Learning with Python: https://www.coursera.org/learn/machine-learning-with-python
4. https://www.youtube.com/watch?v=lfiw2Rh2v8k
5. https://www.youtube.com/watch?v=1Ic7GRtDrPM
6. https://www.youtube.com/watch?v=Bt5g7c2s38M
7. https://www.datacamp.com/tutorial/explainable-ai-understanding-and-trusting-machine-learning-models

Suggested CO - PO articulation Matrix

Course Outcomes	**Programme Outcomes (POs)**										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	3	3	3	3						3
CO2	3	3	3	3	3						3
CO3	3	3	3	3	3						3
CO4	3	3	3	3	3						3
CO5	3	3	3	3	3						3

Legends :- High: 03, Medium: 02,Low: 01, No Mapping: -

Blooms level

Remember	Understand✓	Apply✓	Analyze✓	Evaluate✓	Create

Course Code	Course Name	Teaching Scheme (Hrs/week)				Credits Assigned			
		L	T	P	SL	L	T	P	Total
25PCC13CS15	Theoretical Computer Science	2	1	--	3	2	1	--	3
		Examination Scheme							
						ESE			
			ISE	MSE		Min	Max	Total	
		Theory	20	30		20	50	100	
		Tutorial	50	--		--	--	50	

Pre-requisite Courses		Discrete Maths
After the successful completion students should be able to:		
Course Outcomes	CO1	Design DFA, NFA, Moore, and Mealy machines, demonstrating their equivalence and computational efficiency.
	CO2	Derive the equivalence of languages described by finite automata and regular expressions.
	CO3	Apply grammar principles to address ambiguity, and perform conversions and simplifications of CFGs into Normal Forms.
	CO4	Analyze Pushdown Automata and their equivalence to context free grammars and languages.
	CO5	Analyze Turing Machines, their variants, and advanced concepts to evaluate computational problems and un-decidability.
	CO6	Apply regular expressions, parsing techniques, and recursive functions to model and solve computational problems in NLP.

Module No.	Unit No.	Topics	Ref.	Hrs.
1	1.1	**Introduction to Theory of Computation and Finite Automata** Core concepts including automata, alphabets, symbols, strings, and formal languages; Deterministic Finite Automata (DFA), Non-Deterministic Finite Automata (NFA), and their equivalence; NFA with ε-transitions, its conversion to standard NFA, and Automata minimization (Myhill–Nerode theorem).	1,2,3	5
	1.2	**Finite Automata with Output** Finite Automata with Output: Moore Machine, Mealy Machine, Equivalence of Moore and Mealy Machines	1,2,3	2
2	2.1	**Regular Languages** Regular Expressions, Conversion between RE and FA, Introduction to Algebraic Laws of Regular Languages, Pumping Lemma for proving non-regularity, Closure Properties of Regular Languages including Union, Concatenation, Complement, Intersection, and Kleene Star.	1,2	3
3	3.1	**Grammar** Chomsky Hierarchy, Context free Grammar, Ambiguity in CFG and Methods to Remove Ambiguity, Derivation Trees and	1,3	3

		Ambiguity, Regular Grammars - Right Linear And Left Linear Grammars, Conversion of FA to Regular Grammar and Regular Grammar to FA		
	3.2	**Normal Forms** Simplification of CFG, Normal Forms - Chomsky Normal Form (CNF), Greibach Normal Form (GNF)	1,3	3
4	**4.1**	**Push Down Automata (PDA)** Mathematical Framework of PDA, Transition Diagrams, Functions and Tables, Deterministic Push- Down Automata (DPDA) - Definition, Nondeterministic Pushdown Automata (NPDA), Equivalence Of Context Free Grammars And PDA, Properties Of Context Free Languages.	1,2	5
5	**5.1**	**Turing Machines** Mathematical Framework of TM, Language Acceptability of Turing Machines, Turing Machine Construction	1,2	3
	5.2	**Variants and Advanced Concepts** Composite, iterative, multi-tape, multi-stack, and multitrack Turing machines. Universal Turing machines, Church's thesis, Post Correspondence Problem (PCP), Halting problem. Introduction to Mapping Reductions for undecidability proofs, Rice's Theorem (Statement and Applications).	1,2	3
6	**6.1**	**Applications** Regular expressions for lexical analysis, text editing, and pattern searching; parsing techniques using leftmost and rightmost derivations; recursive functions and recursive and recursively enumerable languages to model computational aspects of natural language. Use of JFLAP tool for visualization and simulation of FA, PDA, and Turing Machines.	1,2,4	3
			Total	**30**

Self-Learning:

1. Self-learning hours include MOOCs, spoken tutorials, online resources, and extended study hours to enhance independent learning and better understanding of each module of the course content.
2. Evaluation of the self-learning components is carried out in all the evaluation components.

Course Assessment: -

Theory:

ISE:

ISE activities carry 20 marks. These activities will be conducted throughout the semester.

MSE:

The written summative examination of 30 marks based on 50% syllabus for 90 minutes.

ESE:

The written summative examination will be conducted for 50 marks based on the complete syllabus (20% questions on syllabus covered before MSE and 80% questions on the remaining syllabus) for 120 minutes.

Suggested list of Tutorials			
	Topics	**Ref.**	**Hrs.**
1	Design DFA accepting the given language.	1,2,3	2
2	DFA Minimization using Myhill–Nerode theorem and State Equivalence Method.	1,2,3	1
3	Design Finite Automata with output (Moore and Mealy Machine), NFA to DFA conversion practice problems.	1,2,3	2
4	Understand how Finite Automata (FA) and Regular Expressions (RE) are connected, with a focus on designing FA from RE, creating RE for specific patterns, and converting between FA and RE.	1,2	2
5	Simplification of CFG, Normal Forms	1,3	2
6	Design/ Construct PDA accepting given language/grammar	1,2	2
7	Design a Turing machine to accept the given language.	1,2	2
8	Use of JFLAP tool for visualization and simulation of DFA, NFA, PDA, and Turing Machines; Implementation and testing of designed automata using JFLAP.	1,2,4	2
	Total		15

Course Assessment:
Tutorial:
ISE: Assessment shall be based on tutorials, evaluated through continuous assessment of analytical thinking, problem-solving skills, and accuracy of solutions**.**

Recommended Books:

1. K.L.P. Mishra and N. Chandrasekaran, "Theory of Computer Science: Automata, Languages, and Computation", 2nd Edition, PHI Learning.
2. John E. Hopcroft, Rajeev Motwani, and Jeffrey D. Ullman, "Automata Theory, Languages, and Computation", 3rd Edition, Pearson.
3. Vivek Kulkarni, "Theory of Computation", Oxford Higher Education.
4. Michael Sipser, "Introduction to the Theory of Computation", Cengage Learning.
5. Peter Linz, "An Introduction to Formal Languages and Automata", Jones & Bartlett Learning.
6. Daniel A. Cohen, "Introduction to Computer Theory", Wiley Publication.
7. John C. Martin, "Introduction to Languages and the Theory of Computation", McGraw Hill.
8. E.V. Krishnamurthy, "Theory of Computer Science", EWP Publication.

Suggested CO - PO articulation Matrix

Course Outcomes	**Programme Outcomes (POs)**										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	3	2	-	-	-	-	-	-	-	-
CO2	3	2	-	-	-	-	-	-	-	-	-
CO3	2	3	-	-	-	-	-	-	-	-	-
CO4	2	2	2	-	-	-	-	-	-	-	-
CO5	2	2	2	-	-	-	-	-	-	-	-
CO6	3	3	-	-	2	-	-	-	-	-	2

Legends:- High: 03, Medium: 02, Low: 01, No Mapping: -

Blooms level

Remember	Understand ✓	Apply✓	Analyze✓	Evaluate	Create

Course Code	Course Name	Teaching Scheme (Hrs/week)				Credits Assigned			
		L	T	P	SL	L	T	P	Total
25PEC13CS11	Big Data Analytics	2	--	2	2	2	--	1	3
		Examination Scheme							
						ESE			
			ISE	MSE		Min	Max	Total	
		Theory	20	30		20	50	100	
		Lab	50	--		--	--	50	

Pre-requisite Courses		Data base management systems
After the successful completion students should be able to:		
Course Outcomes	CO1	Explain building blocks of Big Data Analytics.
	CO2	Apply fundamental enabling techniques like Hadoop and MapReduce in solving real world problems.
	CO3	Analyze different NoSQL database systems and evaluate how they manage and process big data efficiently.
	CO4	Apply advanced techniques for emerging applications like stream analytics.
	CO5	Apply big data analytics to recommendation and graph-based systems.
	CO6	Apply feature engineering techniques and distributed processing frameworks to prepare large-scale datasets for machine learning.

Module No.	Unit No.	Topics	Ref.	Hrs.
1	1.1	**Introduction to Big Data and Hadoop** Introduction to Big Data - Big Data characteristics and Types of Big Data -5 V's, Semi-structured and Structured, Sources of data, working with unstructured data, Big Data vs Data Science vs Data Engineering	1,4	2
	1.2	Big Data Analysis Life Cycle, Case Study of Big Data Solutions, Concept of Hadoop, Core Hadoop Components; Hadoop Ecosystem	1,4	2
2	2.1	**Hadoop HDFS and MapReduce** Distributed File Systems: Physical Organization of Compute Nodes, Large Scale File-System Organization	1,4	3
	2.2	MapReduce: The Map Tasks, Grouping by Key, The Reduce Tasks, Combiners, Details of MapReduce Execution, Coping With Node Failures,	1,4	2
	2.3	Algorithms Using MapReduce: Matrix-Vector Multiplication by MapReduce, Relational-Algebra Operations, Computing Selections by MapReduce, Computing Projections by MapReduce, Union, Intersection, and Difference by MapReduce Hadoop Technology Stack: Hive, Pig, Zookeeper, Swoop, oozie, flume, etc Hadoop Limitations	1,4	2
3	3.1	**NoSQL Databases for Big Data** Introduction to NoSQL, NoSQL Business Drivers, NoSQL Data Architecture Patterns: Key-value stores, Graph stores, Column family (Bigtable)stores, Document stores,Variations of NoSQL architectural patterns, CAP Theorem, ACID vs BASE	1,4	3

	3.2	NoSQL solution for big data, Understanding the types of big data problems; Analyzing big data with a shared-nothing architecture; Choosing distribution models: master-slave versus peer-to-peer; NoSQL systems to handle big data problems. Big data for E-Commerce Big data for blogs, Case Studies MongoDB and Cassandra, Graph Databases-Neo4j, Google BigTable	1,4	1
4	**4.1**	**Mining Data Streams** The Stream Data Model: A Data-Stream-Management System, Examples of Stream Sources, Stream Queries, Issues in Stream Processing, Sampling Data techniques in a Stream, Filtering Streams: Bloom Filter with Analysis.	1,2,4	3
	4.2	Counting Distinct Elements in a Stream, Count Distinct Problem, Flajolet-Martin Algorithm, Combining Estimates, Space Requirements, Real time Analytics Platform (RTAP) applications Counting Ones in a Window: The Cost of Exact Counts, The Datar-Gionis-Indyk-Motwani Algorithm, Query Answering in the DGIM Algorithm, Decaying Windows, Case Studies as Spark, Spark DataFrame operations, Kafka stream processing	1,2,4	2
5	**5.1**	**Real-Time Big Data Models** A Model for Recommendation Systems, Content-Based Recommendations, Collaborative Filtering Social Networks as Graphs, Clustering of Social-Network Graphs, Direct Discovery of Communities in a social graph	1,4	2
6	**6.1**	**Big Data and AI Integration** Big Data for Machine Learning, Distributed Machine Learning Frameworks, Feature Engineering at Scale, Feature Stores, Big Data in Deep Learning, Distributed Deep Learning, LLM Data Pipelines, Vector Databases and Embeddings, Retrieval Augmented Generation (RAG) Architecture, Ethical Issues in Big Data and AI	2,3	4
			Total	**26**

Self-Learning:

1. Self-learning hours include MOOCs, spoken tutorials, online resources, and extended study hours to enhance independent learning and better understanding of each module of the course content.
2. Evaluation of the self-learning components is carried out in all the evaluation components.

Course Assessment: -

Theory:

ISE:

ISE activities carry 20 marks. These activities will be conducted throughout the semester.

MSE:

The written summative examination of 30 marks based on 50% syllabus for 90 minutes.

ESE:

The written summative examination will be conducted for 50 marks based on the complete syllabus (20% questions on syllabus covered before MSE and 80% questions on the remaining syllabus) for 120 minutes.

To be Taught in laboratory			
	Topics	**Ref.**	**Hrs.**
1	• **Hadoop HDFS Practical:** • HDFS Basics, Hadoop Ecosystem Tools Overview. • Installing Hadoop. • Copying File to Hadoop. • Copy from Hadoop File system and deleting file. • Moving and displaying files in HDFS. • Programming exercises on Hadoop	3	2
2	Experiment on Hadoop Map-Reduce: Write a program to implement a word count program usingMapReduce.	3	2
3	Experiment on Hadoop Map-Reduce: -Implementing simple algorithms in MapReduce: Matrix multiplication, Aggregates, Joins, Sorting, Searching, etc	3	4
4	To install and configure MongoDB/ Cassandra/ HBase/ Hyper table to execute NoSQL commands	2	2
5	**Data Stream Algorithms (any one):** • Implementing DGIM algorithm using any Programming Language • Implement Bloom Filter using any programming language • Implement Flajolet Martin algorithm using any programming language	3	4
6	Exploratory Data Analysis using Spark/ Pyspark	1,4	2
7	Data Visualization using Hive/PIG/R/Tableau	1	4
8	Mini Project: One real life large data application to be implemented (Use standard Datasets available on the web). • Streaming data analysis – use flume for data capture, HIVE/PYS park for analysis of twitter data, chat data, weblog analysis etc. • Recommendation System (for example: Health Care System, Stock Market Prediction, Movie Recommendation, etc.) Spatio Temporal Data Analytics	3	6
	Total		26

Course Assessment: -
Lab:

ISE: Laboratory ISE is divided into two components: 25 marks for submission of experiments and 25 marks for oral/practical evaluation.

Recommended Books:

1. Cre Anand Rajaraman and Jeff Ullman ―Mining of Massive Datasets, Cambridge University Press
2. Alex Holmes ―Hadoop in Practice‖, Manning Press, Dreamtech Press.
3. Dan Mcary and Ann Kelly ―Making Sense of NoSQL‖ – A guide for managers and the rest of us, Manning Press.

4. DT Editorial Services, ―Big Data Black Book‖, Dreamtech Press
5. EMC Education Services, Data Science and Big Data Analytics, Wiley
6. Bill Franks , ―Taming The Big Data Tidal Wave: Finding Opportunities In Huge Data Streams with Advanced Analytics, Wiley
7. Chuck Lam, ―Hadoop in Action, Dreamtech Press
8. Jared Dean, ―Big Data, Data Mining, and Machine Learning: Value Creation for Business Leaders and Practitioners, Wiley India Private Limited, 2014.
9. Jiawei Han and Micheline Kamber, ―Data Mining: Concepts and Techniques‖, Morgan Kaufmann Publishers, 3rd ed, 2010.

Suggested CO - PO articulation Matrix

Course Outcomes	**Programme Outcomes (POs)**										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	2	1		1						
CO2	3	3	2	2	3						
CO3	3	3	2	3	3						
CO4	3	3	3	3	3						
CO5	3	3	3	3	3						
CO6	3	3	3	3	3						

Legends :- High: 03, Medium: 02,Low: 01, No Mapping: -

Blooms level

Remember	Understand √	Apply √	Analyze	Evaluate	Create

Course Code	Course Name		Teaching Scheme (Hrs/week)			Credits Assigned			
25PEC13CS12	Blockchain	L	T	P	SL	L	T	P	Total
		2	--	2	2	2	--	1	3
			Examination Scheme						
						ESE			
			ISE	MSE		Min	Max	Total	
		Theory	20	30		20	50	100	
		Lab	50	--		--	--	50	

Pre-requisite Courses	Data structures, Creative Coding in Python	
After the successful completion students should be able to:		
Course Outcomes	CO1	Explain the Fundamental Concepts of Blockchain
	CO2	Examine Consensus Algorithms and Blockchain Security Challenges
	CO3	Analyze Cryptocurrencies and Bitcoin Mechanisms
	CO4	Evaluate Public and Private Blockchain Platforms
	CO5	Develop Smart Contracts using Solidity

Module No.	Unit No.	Topics	Ref.	Hrs.
1	**Introduction to Blockchain**		1,2	5
	1.1	What is a blockchain, Centralization vs. Decentralization, Blockchain defined- peer to peer, Distributed Ledger, Cryptographically Secure, Append-only, Updatable via consensus, The Structure of a Block, Block header, Genesis block, Mining, Rewards, Foundation of blockchain: Merkle trees		
	1.2	Components of blockchain, Types: Public, Private, and Consortium, Consensus Protocol- Proof-of-Work (PoW), Proof-of-Burn (PoB), Proof-of-Stake (PoS), and Proof-of-Elapsed Time (PoET), Limitations and Challenges of blockchain		
2	**Bitcoin and Cryptocurrency**		1,2	5
	2.1	Cryptocurrency: Bitcoin, Altcoin, and Tokens (Utility and Security), Cryptocurrency wallets: Hot and cold wallets, Cryptocurrency usage, Transactions in Blockchain, UTXO and double spending problem in Bitcoin		
	2.2	Mining difficulty, Mining pool, Private keys in Bitcoin, Public Keys in Bitcoin, Addresses in Bitcoin		
3	**Public Blockchain**		4	4
	3.1	Introduction to Public Blockchain, Ethereum and its Components, Mining in Ethereum, Ethereum Virtual Machine (EVM), Transaction, Accounts, Architecture		

		and Workflow, Comparison between Bitcoin and Ethereum		
	3.2	Types of test-networks used in Ethereum, Transferring Ethers using MetaMask, Ethereum frameworks, Case study of Ganache for Ethereum blockchain.		
4	**Programming for Blockchain**		2	8
	4.1	Introduction to Smart Contracts, Types of Smart Contracts, Structure of a Smart Contract, Smart Contract Approaches, Limitations of Smart Contracts		
	4.2	Introduction to Programming: Solidity Programming – Basics, functions, Visibility and Activity Qualifiers, Address and Address Payable, Bytes and Enums, Arrays-Fixed and Dynamic Arrays, Special Arrays-Bytes and strings, Struct, Mapping, Inheritance, Error handling		
5	**Private Blockchain**		1,3,5	4
	5.1	Introduction, Key characteristics, Need of Private Blockchain, Smart Contract in a Private Environment, State Machine Replication, Consensus Algorithms for Private Blockchain - PAXOS and RAFT, Byzantine Faults: Byzantine Fault Tolerant (BFT) and Practical BFT		
	5.2	Introduction to Hyperledger, Tools and Frameworks, Hyperledger Fabric, Comparison between Hyperledger Fabric & OtherTechnologies		
	5.3	Hyperledger Fabric Architecture, Components of Hyperledger Fabric: MSP, Chain Codes, Transaction Flow, Working of Hyperledger Fabric, Creating Hyperledger Network, Case Study of Supply Chain Management using Hyperledger		
			Total	**26**

Self-Learning:

1. Self-learning hours include MOOCs, spoken tutorials, online resources, and extended study hours to enhance independent learning and better understanding of each module of the course content.
2. Evaluation of the self-learning components is carried out in all the evaluation components.

Course Assessment: -

Theory:

ISE:

ISE activities carry 20 marks. These activities will be conducted throughout the semester.

MSE:

The written summative examination of 30 marks based on 50% syllabus for 90 minutes.

ESE:

The written summative examination will be conducted for 50 marks based on the complete syllabus (20% questions on syllabus covered before MSE and 80% questions on the remaining syllabus) for 120 minutes.

Sr. No.	Suggested List of Experiments
1	**1. SHA-256 Hashing and Immutability** **Objective:** To understand cryptographic hashing, block integrity, and immutability in blockchain. **Experiments:** • Implement a SHA-256 hashing program from scratch. • Demonstrate how small changes in input affect the hash output. • Show immutability in a simple blockchain model when a block is altered.
2	**Basic Blockchain Implementation** **Objective:** To build and understand core components of a blockchain. **Experiments:** • Create a basic blockchain using Python/Java with genesis block, proof of work, and data blocks. • Construct a Merkle tree and compute root hashes for transactions. • Validate blockchain integrity after adding multiple blocks.
3	**Installation and Setup of Blockchain Tools** **Objective:** To explore essential blockchain development tools. **Experiments:** • Install MetaMask wallet and connect to test networks. • Setup Ganache local blockchain. • Explore Remix IDE for smart contract development.
4	**Wallet Creation and Testnet Transactions** **Objective:** To perform real cryptocurrency transactions in a controlled environment. **Experiments:** • Create a blockchain wallet (e.g., MetaMask). • Connect MetaMask to an Ethereum test network. • Execute multiple testnet transactions and verify them using explorers.
5	**Smart Contract Development using Remix** **Objective:** To learn fundamentals of smart contract programming and deployment. **Experiments:** • Write a smart contract in Solidity using Remix IDE. • Compile and deploy the contract to a test network.

	• Interact with the deployed contract functions.
6	**Smart Contract Integration with Web Interface** **Objective:** To connect smart contracts with user interfaces through Web3 libraries. **Experiments:** • Use Web3.js / Ethers.js to connect a frontend application to a smart contract. • Implement contract function calls from a simple web page. • Perform transactions from the UI and observe state changes.
7	**Development of a Simple DApp** **Objective:** To design and develop a decentralized application combining frontend and blockchain backend. **Experiments:** • Build a basic DApp with frontend UI and smart contract backend. • Implement user interactions (e.g., token transfer, data storage). • Test the DApp on a local or test blockchain network.
8	**Decentralized Storage Using IPFS** **Objective:** To understand decentralized file storage and retrieval. **Experiments:** • Upload files to IPFS (InterPlanetary File System). • Retrieve files using IPFS content identifiers (CIDs). • Integrate IPFS storage with smart contracts and DApp front end.
9	**Hyperledger Fabric Setup and Chaincode Deployment** **Objective:** To explore permissioned blockchain frameworks used in enterprise systems. **Experiments:** • Install and configure Hyperledger Fabric network. • Write and deploy simple chaincode (smart contract) in Fabric. • Invoke transactions and query ledger state.
10	**Mini Project: Blockchain Application Development** **Objective:** To design, implement, and present a complete blockchain application for real-world use cases. **Experiments:** • Select a use case (e.g., Supply Chain, Medical Records, Land Records). • Design system architecture combining blockchain, smart contracts, and UI. • Implement solution, test end-to-end, and prepare a project report. • Present demo and submit documentation as per guidelines.

Course Assessment: -
Lab:

ISE: Laboratory ISE is divided into two components: 25 marks for submission of experiments and 25 marks for oral/practical evaluation.

Recommended Books:

1. Blockchain Technology, Chandramouli Subramanian, Asha A. George, Abhillash K. A and Meena Karthikeyen, Universities Press.
2. Mastering Ethereum, Building Smart Contract and Dapps, Andreas M. Antonopoulos Dr. Gavin Wood, O'reilly.
3. Imran Bashir, Mastering Blockchain: A deep dive into distributed ledgers, consensus protocols, smart contracts, DApps, cryptocurrencies, Ethereum, and more, 3rd Edition, Packt Publishing.
4. Blockchain with Hyperledger Fabric, Luc Desrosiers, Nitin Gaur, Salman A. Baset, Venkatraman Ramakrishna, Packt Publishing
5. Blockchin enabled Applications, Vikram Dhillon, DevidMetcalf, Max Hooper, Apress

Online Resources:

1. https://ethereum.org/en/
2. https://hyperledger-fabric.readthedocs.io/en/release-2.2/whatis.html
3. https://www.blockchain.com/
4. https://docs.soliditylang.org/en/v0.7.4

Suggested CO - PO articulation Matrix

Course Outcomes	Programme Outcomes (POs)										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	3				2					3
CO2		3		3		3			3		
CO3						2					3
CO4			2						3		
CO5		3		3		3					3

Legends :- High: 03, Medium: 02,Low: 01, No Mapping: -

Blooms level

Remember	Understand ✓	Apply ✓	Analyze ✓	Evaluate ✓	Create

Course Code	Course Name	Teaching Scheme (Hrs/week)				Credits Assigned			
		L	T	P	SL	L	T	P	Total
25PEC13CS13	Image Processing	2	--	2	2	2	--	1	3
		Examination Scheme							
						ESE			
			ISE	MSE		Min	Max	Total	
		Theory	20	30		20	50	100	
		Lab	50	--		--	--	50	

Pre-requisite Courses		Linear algebra, Matrices
After the successful completion students should be able to:		
Course Outcomes	CO1	Explain the fundamental concepts of digital image processing
	CO2	Apply image enhancement techniques to improve image quality.
	CO3	Analyse and implement image segmentation techniques for extracting meaningful regions in an image.
	CO4	Evaluate and implement lossless and lossy image compression techniques
	CO5	Apply transform domain techniques for image processing tasks.
	CO6	Extract and analyse image features such as shape and texture descriptors, and apply basic classification techniques for image analysis.

Module No.	Unit No.	Topics	Ref.	Hrs.
1	**Digital Image Fundamentals and Enhancement**		1,4	7
	1.1	Digital Image Fundamentals: What is Digital Image Processing? Origins of Digital Image Processing, Examples of fields that use DIP, Fundamental Steps in Digital Image Processing, Components of an Image Processing System, Image Sampling and Quantization, Some Basic Relationships Between Pixels, Image File Formats: BMP, TIFF and JPEG.		
	1.2	Introduction to Image Enhancement: Gray Level Transformations, Zero Memory Point Operations, Histogram Processing, Neighborhood Processing, Spatial Filtering, Smoothing and Sharpening Filters		
2	**Image Segmentation**		1,2	5
	2.1	Segmentation based on Discontinuities (point, Line, Edge), Image Edge detection using Robert, Sobel, Previtt masks, Image Edge detection using Laplacian Mask, Edge linking		
	2.2	Region-Oriented Segmentation: Region growing by pixel Aggregation, Split and Merge		
3	**Image Compression**		1,2	6
	3.1	Introduction, Redundancy, Fidelity Criteria, Lossless Compression Techniques: Run length Coding, Arithmetic Coding, Huffman Coding		
	3.2	Lossy Compression Techniques: Improved Gray Scale		

		Quantization, Vector Quantization		
4	**Image Transform**		3	5
	4.1	Introduction to Unitary Transforms, orthogonal transform, Discrete Fourier Transform (DFT), Inverse DFT, Properties of DFT, Fast Fourier Transform (FFT)		
	4.2	Discrete Hadamard Transform (DHT), Inverse DHT, Fast Hadamard Transform (FHT), Discrete Cosine transform (DCT), Inverse DCT, Walsh Transform, Haar transform, Basis images		
5	**Feature Extraction and Image Analysis**		1,2	3
	5.1	Image Representation and Feature Extraction: Boundary representation, Region descriptors, Shape features (area, perimeter, compactness), Texture features, Gray Level Co-occurrence Matrix (GLCM)		
	5.2	Image Classification Basics: Feature vectors, Similarity measures, Introduction to image classification, Performance evaluation metrics		
			Total	**26**

Self-Learning:

1. Self-learning hours include MOOCs, spoken tutorials, online resources, and extended study hours to enhance independent learning and better understanding of each module of the course content.
2. Evaluation of the self-learning components is carried out in all the evaluation components.

Suggested List of Experiments	
Sr. No.	Implementation of following methods in any programming language
1	Any two point processing image enhancement techniques
2	Histogram Equalization, Histogram matching
3	Spatial low pass and high pass filter
4	Edge detection using derivative filter
5	Region based edge detection
6	Lossless compression method, Lossy compression method
7	Generate Walsh transform of an image
8	Extract texture features using GLCM, Compute shape descriptors
9	Simple image classification using feature vectors
10	Mini project

Course Assessment: -

Theory:

ISE:

ISE activities carry 20 marks. These activities will be conducted throughout the semester.

MSE:

The written summative examination of 30 marks based on 50% syllabus for 90 minutes.

ESE:
The written summative examination will be conducted for 50 marks based on the complete syllabus (20% questions on syllabus covered before MSE and 80% questions on the remaining syllabus) for 120 minutes.

Course Assessment: -
Lab:

ISE: Laboratory ISE is divided into two components: 25 marks for submission of experiments and 25 marks for oral/practical evaluation

Recommended Books:

1. R.C.Gonzalez & R.E.Woods, "Digital Image Processing", Pearson Education, 3rd edition.
2. William K. Pratt, "Digital Image Processing", John Wiley, NJ, 4th Edition,200
3. Anil K.Jain, Fundamentals of Digital Image Processing, Prentice Hall of India,2nd Edition,2004.
4. Sid Ahmed M.A., "Image Processing Theory, Algorithm and Architectures", McGraw-Hill, 1995.
5. S. Jayaraman Digital Image Processing TMH (McGraw Hill) publication

Suggested CO - PO articulation Matrix

Course Outcomes	**Programme Outcomes (POs)**										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	2	-	-	-	-	-	-	-	-	-
CO2	3	3	-	2	3	-	-	-	-	-	-
CO3	3	3	2	2	3	-	-	-	-	-	-
CO4	3	3	2	2	3	-	-	-	-	-	-
CO5	3	3	-	2	3	-	-	-	-	-	-
CO6	3	3	3	2	3	-	-	-	-	-	-

Legends :- High: 03, Medium: 02,Low: 01, No Mapping: -

Blooms level

Remember	Understand✓	Apply✓	Analyze✓	Evaluate✓	Create

Course Code	Course Name	Teaching Scheme (Hrs/week)				Credits Assigned			
		L	T	P	SL	L	T	P	Total
25OE13CS31	Health, Wellness and Psychology	2	--	--	2	2	--	--	2
		Examination Scheme							
			ISE	MSE		ESE		Total	
		Theory	100	--		--		100	

Pre-requisite Course Codes		--
After the successful completion students should be able to:		
Course Outcomes	CO1	Introduce the concept of health, wellness and psychology, and understand its Effectiveness in handling stress.
	CO2	Develop human strength and life-enhancement skills through recovery and goal setting.
	CO3	Apply the holistic well-being quotient for personal and professional benefits.

Module No.	Exp. No.	Topics	Ref.	Hrs.
1		**Introduction to Psychology, Health and Wellness**		
	1.1	Understanding holistic health- Meaning, components of holistic health- components of wellbeing, Psychology of overall health-enhancing behaviors component, Types of health-compromising behaviors, Illness Management and wellness enhancement.	1	4
	1.2	Nature and source of stress, personal and professional triggers of stress, Effects of stress, coping with stress (minimalistic yet effective exercise habits)	2	4
2	2	**Promoting Personal and Professional Wellness:Human Strengths & Life-Enhancement**		
	2.1	Strength: Definition, meaning; Realizing strength; Maximizing Unrealized strength Weakness: Definition, meaning; Identifying and overcoming weakness; Developing hope and optimistic approach.	2, 3	4
	2.2	Recovery and Goal Setting: analyzing trends in personality, Approaching Individual differences; Meaning of Goal setting, Types and effectiveness of Goal Setting. Motivation: Meaning, Theory of Needs, 4A's of coping with stress during or after goal setting.	2	4
	2.3	Eudaimonic Wellness: Meaning and characteristics; concept of defensive coping.	14	2
3		**Positive Approach and The Psychology of Living in The Present**		
	3.1	The Psychology of Living in the Present: meaning, self-registering to the flow of positive thoughts and actions;	1,2, 4	4

		addressing positive and negative emotions; Eliminating daily hassles, creating happiness. Responding to overthinking: Socio cultural factor sand self-realization.		
	3.2	Resilience: Meaning and Nature; How to build resilience; Self-Communication and self-care, reframe thoughts; channelize gratitude; practice, Resilience building: physical and mental exercises.	3 4	4
			Total	**26**

Self-Learning:

1. Self-learning hours include MOOCs, spoken tutorials, online resources, and extended study hours to enhance independent learning and better understanding of each module of the course content.
2. Evaluation of the self-learning components is carried out in all the evaluation components.

Course Assessment:

ISE: a) Two activities as formative assessment for 50 marks based on 50% syllabus

b) Two activities as formative/Summative assessment for 50 marks based on remaining syllabus

Recommended Books:

1. Emmons, R.A., & McCullough, M.E. (2003). Counting blessings versus burdens: An experimental investigation of gratitude and subjective well-being in daily life. Journal of Personality & Social Psychology, 88, 377-389
2. Carpenter, S.(2012).Awakeningtosleep.MonitoronPsychology,44(1),40.
3. Emmons, R.A., & Mishra, A.(2012).Why gratitude enhances well-being: What we know, What We Need to Know.
4. Carr,A.(2004).Positive Psychology: The science of happiness and human strength UK Routledge.

Online Resources:

1. The Science of Well-Being by Coursera
2. Managing Emotions in Times of Uncertainty & Stress by Yale University (offered via Coursera)
3. Positive Psychology by The University of North Carolina at Chapel Hill (offered via Coursera)
4. Health and Well-Being by SWAYAM
5. Introduction to Psychology by MIT OpenCourseWare

Suggested CO - PO articulation Matrix

Course Outcomes	**Programme Outcomes (POs)**										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1						3	2	2	2	2	2
CO2						2	2	2	3	2	3
CO3						3	2	3	2	2	3

Legends:- High: 03, Medium: 02, Low: 01, No Mapping: -

Blooms level

Remember	Understand	Apply ✓	Analyze	Evaluate	Create

Course Code	Course Name	Teaching Scheme (Hrs/week)				Credits Assigned			
		L	T	P	SL	L	T	P	Total
25OE13CS32	Emotional and Spiritual Intelligence	2	-	-	2	2	-	-	2
			Examination Scheme						
			ISE	MSE		ESE		Total	
		Theory	100	---		--		100	

Pre-requisite Course Codes		--
After the successful completion students should be able to:		
Course Outcomes	CO1	Introduce the concept of emotional intelligence, its models, components and measures of emotional intelligence
	CO2	Understand the significance of emotional intelligence in self-growth and building effective relationships, Understand the professional impact of emotional intelligence
	CO3	Develop a wide range of work and life skills.
	CO4	Display spiritual intelligence in different roles.
	CO5	Apply the spiritual quotient for corporate benefits.

Module No.	Unit No.	Topics	Ref.	Hrs.
1		**Fundamentals of Emotional Intelligence**		
	1.1	Emotion- Meaning, characteristics of emotion, components of emotion- cognitive component, physiological component, Behavioural component. Types of emotions, exposing the myths about emotion, physiological or bodily changes accompanying emotions, how emotions affect our thinking and actions	1	3
	1.2	Nature and Significance of EI, Models of emotional intelligence: Ability, Trait and Mixed, Building blocks of emotional intelligence: self-awareness, self- management, social awareness, and relationship management	2	3
2		**Personal and Social Competence**		
	2.1	Self-Awareness: Observing and recognizing one's own feelings, Knowing one's strengths and areas of development Self-Management: Managing emotions, anxiety, fear, and anger	2	3
	2.2	Social Awareness: Others' Perspectives, Empathy and Compassion Relationship Management: Effective communication,Collaboration, Teamwork, and Conflict management (professional impact)	2	3
	2.3	Strategies to develop and enhance emotional intelligence and using them effectively in professional life	1	2
3		**Background and Approach: Spiritual Intelligence and Karma Yoga**		

	3.1	• Spiritual Intelligence- Definition, need, state of presence, psychological element, Intuitive intelligence. • Foundation of Spiritual Intelligence • Types of spiritual actions • Models- SQ and SI-Growth model • Yoga of Action and Spirituality: Professionalism	6,8	3
	3.2	• Types of spiritual actions • Models- SQ and SI-Growth model • Readiness for spiritual intelligence: self-leadership, synthesize high performance, spiritual awareness, neuropsychology, and state of conscious identity.	5, 7	3
4		**Opposite Polarity in SI and Overall Impact on Personality**		4
	4.1	• Twin poles of attention- subject and the object pole • Benefits of Spiritual Intelligence- personal, social and corporate • Dimensions of Spiritual Intelligence- SI and Self Esteem, SI and Restoration of confidence SI and clarity of thoughts and speech, Personality moulding and SI.	8	
5	**5.1**	**Spiritual Ecology and Environmental Grassroots Activism**		
		• Spiritual Stewardship and Ecology: Case studies based on making a difference in ecology through environmental grassroots activism	4	2
			Total	26

Self-Learning:

Self-Learning:

1. Self-learning hours include MOOCs, spoken tutorials, online resources, and extended study hours to enhance independent learning and better understanding of each module of the course content.
2. Evaluation of the self-learning components is carried out in all the evaluation components.

Course Assessment:

ISE: a) Two activities as formative assessment for 50 marks based on 50% syllabus

b) Two activities as formative/Summative assessment for 50 marks based on remaining syllabus

Recommended Books:

1. Bar-On, R., & Parker, J.D.A.(Eds.) (2000). The handbook of emotional intelligence, San Francisco, California: Jossey Bros.
2. Goleman, D. (2005). Emotional Intelligence. New York: Bantam Book.
3. Sternberg, R. J. (Ed.). (2000). Handbook of intelligence. Cambridge University Press.
4. Thich Nhat Hanh, V. S. (2016). Spiritual Ecology: The Cry of the Earth. Golden Sufi Center.

Suggested CO - PO articulation Matrix

Course Outcomes	**Programme Outcomes (POs)**										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1						2		2	2	3	2
CO2						2	2	2	2	3	3
CO3							1	2	2	2	2
CO4								2	2		2
CO5						2	3	3	2		3

Legends:- High: 03, Medium: 02, Low: 01, No Mapping: -

Blooms level

Remember	Understand ✓	Apply ✓	Analyze	Evaluate	Create

Course Code	Course Name	Teaching Scheme (Hrs/week)				Credits Assigned			
		L	T	P	SL	L	T	P	Total
25PCC13CS16	Cryptography and Computer Security	2	--	2	2	2	--	1	3
		Examination Scheme							
			ISE	MSE	ESE		Total		
					Min	Max			
		Theory	20	30	20	30	100		
		Lab	50	--	--	--	50		

Pre-requisite Courses		Discrete Maths and Statistics
After the successful completion students should be able to:		
Course Outcomes	CO1	Apply concepts of modular arithmetic and number theory to classical encryption techniques to achieve system security goals.
	CO2	Apply modern cryptographic techniques to a given problem
	CO3	Analyze various hash functions and digital signature algorithms to authenticate and verify integrity
	CO4	Analyze various attacks on network security, and different security protocols.

Module No.	Unit No.	Topics	Ref.	Hrs.
		Introduction - Number Theory and Basic Cryptography	1,2,3	8
1	**1.1**	Security Goals, Security Attacks, Security Services and Security Mechanisms		
	1.2	Modular Arithmetic: Prime No, Euclidean Algorithm, Extended Euclidean Algorithm		
	1.3	Classical Encryption techniques, mono-alphabetic and polyalphabetic ciphers		
	1.4	Substitution techniques: Vigenère cipher, Playfair cipher, Hill cipher, transposition techniques: keyed and keyless transposition ciphers		
2		**Symmetric and Asymmetric key Cryptography and key Management**	1,2,5,6	8
	2.1	Block cipher principles, DES, Double DES, Triple DES		
	2.2	Stream Ciphers: RC4 algorithm		
	2.3	Public key cryptography: Principles of public key cryptosystems- The RSA Cryptosystem		
	2.4	Symmetric key agreement: Diffie Hellman Key Exchange		
	2.5	Public key Distribution: Digital Certificate: X.509, PKI		
	2.6	Introduction to Post-Quantum Cryptography: Need for quantum-safe cryptography, Impact of quantum computing on RSA and Diffie-Hellman, Overview of quantum-resistant algorithms: Lattice-based, Hash-based, Code-based cryptography		
		Cryptographic Hash Functions	1,2,4	3

3	**3.1**	Cryptographic hash functions, Properties of secure hash function, MD5, SHA-1		
	3.2	MAC, HMAC		
4		**Authentication Protocols and Digital Signature Schemes**	1,2	4
	4.1	Symmetric Key Distribution: Needham-Schroeder protocol(symmetric), Kerberos Authentication protocol		
	4.2	RSA as a Digital Signature		
5		**Network and System Security**	1,2	5
	5.1	Network security basics: TCP/IP vulnerabilities (Layer wise), Network Attacks: Packet Sniffing, ARP spoofing, port scanning, IP spoofing, Denial of Service: DOS attacks, ICMP flood, SYN flood, UDP flood, Distributed Denial of Service		
	5.2	Firewall Characteristics Types of Firewalls, Intrusion Detection Systems: Host based and Network Based IDS, SSL and IPSEC :AH , ESP		
	5.3	System Security: Buffer Overflow, malicious Programs: Worms and Viruses, SQL injection, Trojan Horse		
			Total	**28**

Exp. No.	**Suggested List of experiments (Any 10)**
1	Design and Implementation of a product cipher using Substitution and Transposition ciphers
2	Implementation and analysis of public key cryptography
3	Implementation of Diffie-Hellman Key exchange algorithm.
4	For varying message sizes, test integrity of message using MD-5, SHA-1 and analyse the performance of the two protocols.
5	Implementation and analysis of Digital signature scheme.
6	Implementation of Salt and Pepper password protection technique
7	Implement Needham Schroeder authentication protocol
8	Explore the GPG tool of Linux to implement email security
9	Study and Implement SQL Injection
10	Study and Implement DOS Attacks
11	Using NMAP for ports monitoring.
12	Using open SSL for web server - browser communication.
13	Explorer Kali Linux operating system and explain any one tool of kali Linux
14	EXPLORING N-STALKER : To download the N-Stalker Vulnerability Assessment Tool and exploring the features.

Self-Learning:

1. Self-learning hours include MOOCs, spoken tutorials, online resources, and extended study hours to enhance independent learning and better understanding of each module of the course content.
2. Evaluation of the self-learning components is carried out in all the evaluation components.

Course Assessment: -
Theory:

ISE:
ISE activities carry 20 marks. These activities will be conducted throughout the semester.
MSE:
The written summative examination of 30 marks based on 50% syllabus for 90 minutes.
ESE:
The written summative examination will be conducted for 50 marks based on the complete syllabus (20% questions on syllabus covered before MSE and 80% questions on the remaining syllabus) for 120 minutes.

Course Assessment: -
Lab:

ISE: Laboratory ISE is divided into two components: 25 marks for submission of experiments and 25 marks for oral/practical evaluation.

Recommended Books:

1. Atul Kahate, "Cryptography and Network Security", Tata McGraw-Hill Education, 2003.
2. William Stallings, "Cryptography and Network Security, Principles and Practice", 6th Edition, Pearson Education, March 2013
3. Behrouz A. Ferouzan, "Cryptography & Network Security", Tata McGraw Hill
4. Behrouz A. Forouzan & Debdeep Mukhopadhyay, "Cryptography and Network Security" 3rd Edition, McGraw Hill
5. Post-Quantum Cryptography, Springer, 2009.
6. NIST, Reports and Publications, National Institute of Standards and Technology.

Online Resources:

1. http://nptel.ac.in/courses/106105031/ lecture by Dr. Debdeep MukhopadhyayIIT Kharagpur
2. https://archive.nptel.ac.in/courses/106/105/106105162/
3. https://www.geeksforgeeks.org/cryptography-and-network-security-principles/

Suggested CO - PO articulation Matrix

Course Outcomes	**Programme Outcomes (POs)**										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	2		–	–	–	–	–	–	–	–
CO2	3	3	2		2	–	–	2	_	–	–
CO3	3	3	-	2	2	–	–	2	–	–	–
CO4	-	3	-	2	2	2	–	2		–	–

Legends :- High: 03, Medium: 02,Low: 01, No Mapping: -

Blooms level

Remember	Understand ✓	Apply ✓	Analyze ✓	Evaluate	Create

Course Code	Course Name	Teaching Scheme (Hrs/week)				Credits Assigned			
		L	T	P	SL	L	T	P	Total
25PCC13CS17	Data Warehousing and Mining	2	--	2	2	2	--	1	3
		Examination Scheme							
			ISE	MSE	ESE		Total		
					Min	Max			
		Theory	20	30	20	50	100		
		Lab	50	--	--	--	50		

Pre-requisite Courses		DBMS, Data analytics and visualization, Machine Learning
After the successful completion students should be able to:		
Course Outcomes	CO1	Design and develop data warehouse schemas and ETL processes for real-world applications.
	CO2	Apply OLAP operations and data warehouse techniques to analyze multidimensional data for decision-making.
	CO3	Apply data mining techniques including rule-based classification, clustering, and pattern mining to extract knowledge from data.
	CO4	Analyze web data using web mining techniques to discover usage patterns and support recommendation systems.

Module No.	Unit No.	Topics	Ref.	Hrs.
		Data Warehousing Fundamentals	3	6
1	1.1	Introduction to Data Warehouse, Architecture, Data Warehouse vs Data Mart, OLTP vs OLAP, Data Warehouse Lifecycle, E-R Modeling versus Dimensional Modeling, Information Package Diagram		
	1.2	Dimensional Modeling: Fact tables, Dimension tables, Star Schema, Snowflake Schema, Factless Fact Table, Fact Constellation. ETL Process: Data Extraction, Transformation, Loading, Data Cleaning, Data Integration		
	1.3	OLAP Operations: Slice, Dice, Roll-up, Drill-down, Pivot, Cloud-Based Data Warehousing: What is cloud data warehousing, Difference between on-premises and cloud DW, Basic architecture		
2		**Data Warehouse Design & Implementation**	1,2	4
	2.1	Data Warehouse Design Methodology: Top-down vs Bottom-up approach, Kimball vs Inmon approach. Data Warehouse Implementation: ETL vs ELT, Metadata, Data Quality, Data Governance		
	2.2	Data Warehouse Performance: Indexing techniques (Bitmap index), Partitioning, Query optimization. Modern Data Warehousing: Data Lake vs Data Warehouse		
3		**Data Mining Process & Applications**	1,2	6
	3.1	Data Mining Task Primitives, Architecture, KDD process, Issues in Data Mining, Applications of Data Mining, Basic Concepts,		

		IF–THEN Rule Classifiers, Sequential Covering Algorithm, Rule Pruning, fuzzy classification		
	3.2	Clustering, Proximity Measures: Euclidean distance, Manhattan distance, Minkowski distance, Cosine similarity, Partitioning Methods: k-Means Algorithm. k-Means++ Initialization. k-Medoids (PAM Algorithm), Hierarchical Methods: Agglomerative Clustering, Divisive Clustering, Linkage Criteria: Single Linkage, Complete Linkage, Average Linkage, Dendrogram. Density-Based Clustering: DBSCAN Algorithm, Core, Border, and Noise Points, Advantages over k-Means		
4		**Mining frequent patterns and associations**	1,2	5
	4.1	Market Basket Analysis, Frequent Item sets, Closed Item sets, and Association Rule, Frequent Pattern Mining, Apriori Algorithm		
	4.2	Association Rule Generation, Improving the Efficiency of Apriori, Mining Frequent Item sets without candidate generation, Introduction to Mining Multilevel Association Rules and Mining Multidimensional Association Rules.		
5		**Web mining**	1,2,3	5
	5.1	Introduction, Web Content Mining: Crawlers, Harvest System, Virtual Web View, Personalization, Web Structure Mining: Hyperlink Analysis, PageRank Algorithm (Mathematical intuition), HITS Algorithm (Authorities & Hubs), Comparison of PageRank and HITS		
	5.2	Web Usage Mining: Web Server Logs, Clickstream Data, Preprocessing of Web Logs, User Session Identification, Pattern Discovery in Web Usage, Applications in Recommendation Systems		
			Total	**26**

Exp. No.	Suggested List of experiments
1	Data Warehouse Construction a) Real life Problem to be defined for Warehouse Design b) Construction of star schema and snowflake schema c) Extract data from CSV / database and Perform: Data cleaning, Transformation, Loading into warehouse tables (ETL Operations)
2	Create OLAP cube and Perform: Slice, Dice, Roll-up, Drill-down. Write OLAP queries
3	Compare query performance: With and without indexing, Partitioned vs non-partitioned tables, Analyze execution time
4	Implement IF–THEN rule-based classifier OR use tool (e.g., Orange / Weka) Analyze rules generated
5	Using open-source tools Implement Clustering Algorithms
6	Implementation of anyone clustering algorithm using languages like JAVA/ python
7	Implementation of any one association mining algorithm using languages like JAVA/ python
8	Implementation of page rank algorithm.

9	Implementation of HITS algorithm.
10	Pre-process web log data, Identify: User sessions, Navigation patterns

Self-Learning:

a. Self-learning hours include MOOCs, spoken tutorials, online resources, and extended study hours to enhance independent learning and better understanding of each module of the course content.
b. Evaluation of the self-learning components is carried out in all the evaluation components.

Course Assessment: -
Theory:

ISE:
ISE activities carry 20 marks. These activities will be conducted throughout the semester.
MSE:
The written summative examination of 30 marks based on 50% syllabus for 90 minutes.
ESE:
The written summative examination will be conducted for 50 marks based on the complete syllabus (20% questions on syllabus covered before MSE and 80% questions on the remaining syllabus) for 120 minutes.

Course Assessment: -
Lab:

ISE: Laboratory ISE is divided into two components: 25 marks for submission of experiments and 25 marks for oral/practical evaluation.

Recommended Books:

1. J Han and H. Kamber, "Data Mining: Concepts and Techniques", 3rd edition, Morgan Kaufmann.
2. P. N. Tan, M. Steinbach, Vipin Kumar, "Introduction to Data Mining", 2nd edition, Pearson Education.
3. Paulraj Ponniah, "Data Warehousing: Fundamentals for IT Professionals", 2nd edition, Wiley India

Online Resources:

1. www.leetcode.com
2. www.hackerrank.com
3. www.cs.usfca.edu/~galles/visualization/Algorithms.html
4. www.codechef.com
5. https://www.geeksforgeeks.org/web-mining

Suggested CO - PO articulation Matrix

Course Outcomes	**Programme Outcomes (POs)**										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	2	3	–	2	–	–	–	–	–	–
CO2	2	3	2	2	2	–	–	-	-	–	-
CO3	3	3	3	2	2	-	–	-	–	–	–
CO4	2	2	-	-	2	-	-	-	-	–	–

Legends:- High: 03, Medium: 02, Low: 01, No Mapping: -

Blooms level

Remember	Understand	Apply ✓	Analyze ✓	Evaluate	Create

Course Code	Course Name	Teaching Scheme (Hrs/week)			Credits Assigned			
		L	T	P	L	T	P	Total
		--	--	2	--	--	1	1
		Examination Scheme						
					ESE			
25PCC13CS18	Cloud Computing		ISE	MSE	Min	Max	Total	
		Theory	--	--	--	--	--	
		Lab	50	--	--	--	50	

Pre-requisite Courses	Operating System, Computer Network	
After the successful completion students should be able to:		
Course Outcomes	CO1	Analyze cloud computing service models and develop real-world web applications for deployment on commercial cloud platforms.
	CO2	Apply various virtualization techniques in practical scenarios.
	CO3	Create and deploy real-world web applications on commercial cloud platforms.
	CO4	Implement serverless solutions with messaging services like AWS Lambda, SQS, and SNS for event-driven architectures.
	CO5	Explore key security mechanisms in the cloud and propose solutions to mitigate associated challenges.
	CO6	Apply the principles of containerization to practical implementations.

Exp. No.	Suggested list of experiments	Ref	COs	Hrs
	Cloud basics			
1	**Title: Introduction and overview of cloud computing.** Objective: To understand the origin of cloud computing, cloud cube model, NIST model, characteristics of cloud, different deployment models, service models, advantages and disadvantages	3	1	2
	Virtualization			
2	**Title: To study and implement Hosted Virtualization using VirtualBox& KVM.** Objective: To know the concept of Virtualization along with their types, structures and mechanisms. This experiment should have demonstration of creating and running Virtual machines inside hosted hypervisors like VirtualBox and KVM with their comparison based on various virtualization parameters.	3	2	2
3	**Title: To study and Implement Bare-metal Virtualization using Xen, HyperV or VMware Esxi.** Objective: To understand the functionality of Bare-metal	3	2	

	hypervisors and their relevance in cloud computing platforms. This experiment should have demonstration of install, configure and manage Bare Metal hypervisor along with instructions to create and run virtual machines inside it. It should also emphasize on accessing VMs in different environments along with additional services provided by them like Load balancing, Auto-Scaling, Security etc.			
	Services			
4	**Title: To study and Implement Infrastructure as a Service using AWS/Microsoft Azure.** Objective: To demonstrate the steps to create and run virtual machines inside Public cloud platform. This experiment should emphasize on creating and running Linux/Windows Virtual machine inside Amazon EC2 or Microsoft Azure Compute and accessing them using RDP or VNC tools.	1,2,4	1,3	2
5	**Title: To study and Implement Platform as a Service using AWS Elastic Beanstalk/ Microsoft Azure App Service.** Objective: To demonstrate the steps to deploy Web applications or Web services written in different languages on AWS Elastic Beanstalk/ Microsoft Azure App Service.	1,2,4	1,3	2
6	**Title: To study and Implement Storage as a Service using Own Cloud/ AWS S3, Glaciers/ Azure Storage.** Objective: To understand the concept of Cloud storage and to demonstrate the different types of storages like object storage, block level storages etc. supported by Cloud Platforms like Own Cloud/ AWS S3, Glaciers/ Azure Storage.	1,2,4	1,3	2
7	**Title: To study and Implement Database as a Service on SQL/NOSQL databases like AWS RDS, AZURE SQL/ MongoDB Lab/ Firebase.** Objective: To know the concept of Database as a Service running on cloud and to demonstrate the CRUD operations on different SQL and NOSQL databases running on cloud like AWS RDS, AZURE SQL/ Mongo Lab/ Firebase.	1,2,4	1,3	2
	Post MSE			
	SECURITY			
8	**Title: To study and Implement Security as a Service on AWS/Azure** Objective: To understand the Security practices available in public cloud platforms and to demonstrate various Threat detection, Data protection and Infrastructure protection services in AWS and Azure.	1,4	1,5	2
9	**Title: To study and implement Identity and Access Management (IAM) practices on AWS/Azure cloud.** Objective: To understand the working of Identity and Access Management IAM in cloud computing and to demonstrate the	1,2,4	1,3,5	2

	case study based on Identity and Access Management (IAM) on AWS/Azure cloud platform.			
	Serverless and messaging			
10	**Lab Title: Exploring AWS Lambda: Building Serverless Event-Driven Applications** Objective: To understand and implement serverless event handling using AWS Lambda, focusing on creating, deploying, and testing event-driven functions integrated with AWS services.	1,4	1,4	2
11	**Lab Title: Implementing Scalable Messaging Systems with AWS SQS and SNS.** Lab Objective: To explore and implement cloud messaging services using Amazon SQS and SNS, focusing on building scalable, reliable, and event-driven communication systems.	1,4	1,4	
	Containerization			
12	**Title: To study and Implement Containerization using Docker** Objective: To know the basic differences between Virtual machine and Container. It involves demonstration of creating, finding, building, installing, and running Linux/Windows application containers inside local machine or cloud platform.	6,7	6	2
13	**Title: To study and implement container orchestration using Kubernetes** Objective: To understand the steps to deploy Kubernetes Cluster on local systems, deploy applications on Kubernetes, creating a Service in Kubernetes, develop Kubernetes configuration files in YAML and creating a deployment in Kubernetes using YAML	6,7	6	2
	Mini Project			
14	**Design a Web Application hosted on public cloud platform (Suggested list of Mini Project Topics)** 1. Deployment of a scalable web application on AWS using EC2, S3, and RDS. 2. Implementing a CI/CD pipeline for a machine learning project using Jenkins and Docker. 3. Building a serverless application using AWS Lambda for real-time data processing. 4. Developing a microservices architecture application deployed on Kubernetes. 5. Setting up and managing a multi-tier application on a private cloud using OpenStack.	Online resources	1,2,3, 4,5,6	

Course Assessment:

Lab:

ISE: Laboratory ISE is divided into two components: 25 marks for submission of experiments and 25 marks for oral/practical evaluation.

Recommended Books:

1. Bernard Golden, "Amazon Web Services for Dummies", John Wiley & Sons, Inc.
2. Michael Collier, Robin Shahan, "Fundamentals of Azure, Microsoft Azure Essentials", Microsoft Press.
3. RajkumarBuyya, Christian Vecchiola, S ThamaraiSelvi, "Mastering Cloud Computing", Tata McGraw-Hill Education.
4. Barrie Sosinsky, "Cloud Computing Bible", Wiley publishing ,John Paul Mueller, "AWS for Admins for Developers", John Wiley & Sons, Inc.
5. Ken Cochrane, Jeeva S. Chelladhurai, NeependraKhare , "Docker Cookbook - Second . Edition", Packt publication
6. Jonathan Baier, "Getting Started with Kubernetes-Second Edition", Packt Publication

Online Resources:

1. Website link :
   Docker Containers and Kubernetes Fundamentals – Full Hands-On Course
   https://www.youtube.com/watch?v=kTp5xUtcalw
   Docker and Kubernetes Tutorials Playlist
   https://www.youtube.com/playlist?list=PLuZ-P8G2omalspeot9_F_qnJjeLNVADbw
   Docker and Kubernetes Tutorial for Beginners
   https://www.youtube.com/playlist?list=PLy7NrYWoggjwPggqtFsI_zMAwvG0SqYCb
   Complete Kubernetes Tutorial for Beginners
   https://www.youtube.com/playlist?list=PLy7NrYWoggjziYQIDorlXjTvvwweTYoNC
2. NPtel link :
   https://onlinecourses-archive.nptel.ac.in/noc18_cs16/preview
   https://onlinecourses.nptel.ac.in/noc23_cs90/preview
   https://www.youtube.com/playlist?list=PLfiOAkfpIBRxwkGNQ25v_EY2HbU27luaN
3. Certification link :
   - AWS Cloud Solutions Architect Professional Certificate
   - Cloud Engineering with Google Cloud Professional Certificate
   - Preparing for Google Cloud Certification: Cloud Architect Professional Certificate

Suggested CO - PO articulation Matrix

Course Outcomes	**Programme Outcomes (POs)**										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3				2						3
CO2		3		2							
CO3	3		2		3				3		3
CO4			3						3		3
CO5					3		2		3		2
CO6			3		2				2		3

Legends :- High: 03, Medium: 02, Low: 01, No Mapping: -

Blooms level

Remember	Understand ✓	Apply ✓	Analyze ✓	Evaluate ✓	Create ✓

Course Code	Course Name	Teaching Scheme (Hrs/week)				Credits Assigned			
		L	T	P	SL	L	T	P	Total
25PCC13CS19	Deep Learning	2	--	2	2	2	--	1	3
		Examination Scheme							
			ISE	MSE	ESE		Total		
					Min	Max			
		Theory	20	30	20	50	100		
		Lab	50	--	--	--	50		

Pre-requisite Courses		Machine Learning
After the successful completion students should be able to:		
Course Outcomes	CO1	Design and Train deep learning models for supervised learning task.
	CO2	Design and Train deep learning models for unsupervised learning task
	CO3	Design, train, and optimize deep learning models by tuning hyperparameters to improve model performance.
	CO4	Select and implement appropriate deep learning model to solve real world problem.

Module No.	Unit No.	Topics	Ref.	Hrs.
1	**Fundamentals of Neural network**		3	2
	1.1	History of Deep Learning, Deep Learning Success Stories, Brief recap of Neural Network basics (Perceptron, MLP)		
	1.2	Deep Networks: Three Classes of Deep Learning. Basic Terminologies of Deep Learning.		
2	**Training, Optimization and Regularization of Deep Neural Network**		3	6
	2.1	Training Multi Layered Feed Forward Neural Network, Learning Factors, Activation functions: Tanh, Logistic, Linear, Softmax, ReLU, Leaky ReLU, Loss functions: Squared Error loss, Cross Entropy, Choosing output function and loss function		
	2.2	Advanced Optimization: Momentum Based GD, Nesterov Accelerated GD, AdaGrad, Adam, RMSProp		
	2.3	Regularization: Overview of Overfitting, Types of biases, Bias Variance Tradeoff Regularization Methods: L1, L2 regularization, Parameter sharing, Dropout, Weight Decay, Batch normalization, Early stopping, Data Augmentation, Adding noise to input and output		
3	**Convolutional Neural Networks (CNN): Supervised Learning**		1,2	7
	4.1	Convolution operation, Padding, Stride, Relation between input, output and filter size, CNN architecture: Convolution layer, Pooling Layer, Weight Sharing in CNN, Fully Connected NN vs CNN, Variants of basic Convolution function, Multichannel convolution operation,2D convolution.		
	4.2	Modern Deep Learning Architectures: LeNET: Architecture, AlexNET: Architecture, ResNet : Architecture		

		Transfer Learning: Pre-trained models, fine-tuning		
4	**Recurrent Neural Networks (RNN): Supervised Learning**		1,2,3	6
	5.1	Sequence Learning Problem, Unfolding Computational graphs, Recurrent Neural Network, Bidirectional RNN, Backpropagation Through Time (BTT), Limitation of "vanilla RNN" Vanishing and Exploding Gradients, Truncated BTT		
	5.2	Long Short Term Memory(LSTM): Selective Read, Selective write, Selective Forget, Gated Recurrent Unit (GRU)		
	5.3	Transformer Architecture • Scaled Dot-Product Attention and Self-Attention Mechanism • Multi-Head Attention and Positional Encoding • Encoder–Decoder Architecture of Transformer	8	
5	**Autoencoders: Unsupervised Learning**		1,2	5
	3.1	Introduction, Linear Auto encoder, Under complete Auto encoder, over complete Auto encoders, Regularization in Auto encoders.		
	3.2	Denoising Auto encoders, Sparse Auto encoders, Contractive Auto encoders		
	3.3	Generative Adversarial Network: Architecture, working and applications.		
			Total	**26**

Self-Learning:

1. Self-learning hours include MOOCs, spoken tutorials, online resources, and extended study hours to enhance independent learning and better understanding of each module of the course content.
2. Evaluation of the self-learning components is carried out in all the evaluation components.

Course Assessment: -

Theory:

ISE:

ISE activities carry 20 marks. These activities will be conducted throughout the semester.

MSE:

The written summative examination of 30 marks based on 50% syllabus for 90 minutes.

ESE:

The written summative examination will be conducted for 50 marks based on the complete syllabus (20% questions on syllabus covered before MSE and 80% questions on the remaining syllabus) for 120 minutes.

To be Taught in laboratory			
	Topics	**Ref.**	**Hrs.**
1	**Introduction to Python Libraries for Deep Learning:** **Objectives:** To introduce various python libraries used for DL models. **Task:** Explore python libraries for deep learning e.g. Theano, TensorFlow, pytorch etc.	5	2
2	**Optimization algorithms:**	4,9	2

	a. Stochastic Gradient Descent b. Mini Batch Gradient Descent c. Momentum GD d. Nestorev GD e. Adagrad GD f. Adam Learning GD		
3	**Fully Connected Neural Network:** Design and implement a fully connected deep neural network for classification, object recognition. Use appropriate Learning Algorithm, output function and loss function.	4,6	2
4	**Convolutional Neural Networks (CNNs) for Image Classification:** Design and implement CNNs for image classification tasks. . Build a CNN model with Convolutional layers (with filters, e.g., 32 filters of size 3x3), Pooling layers (max pooling), Fully connected layers at the end, Softmax output layer	4,6	2
5	**Transfer Learning with Pre-Trained Models:** Use a smaller dataset (e.g., Flowers dataset with 5 classes or any other suitable dataset). Classify flower species using a pre-trained models such as **VGG16, ResNet50**, or **InceptionV3** from Keras. Remove the final fully connected layers. Add a custom fully connected layer suited for your task.	4,6	2
6	**Time-Series Forecasting with Recurrent Neural Networks (RNNs):** Use Stock price data, temperature data, or any time-series dataset. Build and train an RNN/LSTM/GRU to predict future values based on historical data.	4,6	2
7	**Auto encoders for Dimensionality Reduction and Anomaly Detection:** Detect anomalous data points (e.g., outlier detection). Build an autoencoder with: • Encoder: A few convolutional or dense layers to reduce dimensionality. • Decoder: Reconstructs the input data.	4,6	2
8	**Generative Adversarial Networks (GANs) for Image Generation:** Use the **CelebA** dataset (celebrity faces). Generate realistic-looking faces from random noise. Implement a GAN with: • A **generator** network to generate fake images from random noise. • A **discriminator** network to distinguish real vs. fake images.	4,6	2
9	**Hyperparameter Tuning and Model Optimization:** To perform hyperparameter tuning for better model performance. Use CIFAR-10 dataset. Build a neural network (MLP or CNN). 1. **Hyperparameters to tune**: Learning rate, batch size, number of epochs. Number of layers, units per layer, activation functions. 2. **Method**: Use **Grid Search** or **Random Search** for hyperparameter optimization.	4,6	2

10	**Mini Project:** **Task:** Defining the problem statement and objectives. Select, implement and train a suitable deep learning model to solve the real world problem. Evaluate the model based on suitable evaluation metrics. Interpret the results to understand how well the model addresses the problem. Implement the idea of Mini Project based on the content of the syllabus (Group of 2-3 students)		6
	Total		24

Note: Please note that the datasets and models referenced in the experiments may be subject to change. These are only Suggested datasets and models. Students are encouraged to explore alternative datasets and models, in consultation with the subject teacher.

Course Assessment: -
Lab:

ISE: Laboratory ISE is divided into two components: 25 marks for submission of experiments and 25 marks for oral/practical evaluation

Recommended Books:

1. Ian Goodfellow, Yoshua Bengio, Aaron Courville. - Deep Learning, MIT Press Ltd, 2016
2. Li Deng and Dong Yu, ―Deep Learning Methods and Applications, Publishers Inc.
3. Satish Kumar "Neural Networks A Classroom Approach", Tata McGraw-Hill.
4. Deep Learning from Scratch: Building with Python from First Principles- Seth Weidman by O`Reilley
5. François Chollet. ―Deep learning with Python (Vol. 361). 2018 New York: Manning.
6. Douwe Osinga. ―Deep Learning Cookbook, O'REILLY, SPD Publishers, Delhi.
7. JM Zurada ―Introduction to Artificial Neural Systems, Jaico Publishing House
8. Simon J.D. Prince - Understanding Deep Learning, MIT press 2023
9. M. J. Kochenderfer, Tim A. Wheeler. ―Algorithms for Optimization, MIT Press.
10. Rajalingappaa Shanmugamani - Deep Learning for Computer Vision, Packt Publishing

Online Resources:

1. DeepLearning.AI Coursera: https://www.coursera.org/specializations/deep-learning
2. NPTEL course on Deep Learning: https://onlinecourses.nptel.ac.in/noc20_cs62/preview
3. https://stanford.edu/~shervine/teaching/cs-230/cheatsheet-recurrent-neuralnetworks
4. https://keras.io/examples/vision/autoencoder/
5. https://stanford.edu/~shervine/teaching/cs-230/cheatsheet-convolutional-neuralnetworks
6. https://www.superannotate.com/blog/object-detection-with-deep-learning
7. https://keylabs.ai/blog/hyperparameter-tuning-grid-search-random-search-and-bayesian-optimization/

Suggested CO - PO articulation Matrix

Course Outcomes	**Programme Outcomes (POs)**										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	3	3	2	3						2
CO2	3	3	3	2	3						2
CO3	3	3	3	3	3						2
CO4	3	3	3	3	3				3		3

Legends :- High: 03, Medium: 02,Low: 01, No Mapping: -

Blooms level

Remember	Understand✓	Apply✓	Analyze✓	Evaluate✓	Create

Course Code	Course Name	Teaching Scheme (Hrs/week)				Credits Assigned			
		L	T	P	SL	L	T	P	Total
25PCC13CS20	Software Testing Lab	--	--	2	--		--	1	1
		Examination Scheme							
			ISE	MSE	ESE			Total	
					Min		Max		
		Theory	--	--	--		--	--	
		Lab	50	--	--		--	50	

Pre-requisite Courses		Programming Fundamentals, Object Oriented Programming with JAVA, Web Programming
After the successful completion students should be able to:		
Course Outcomes	CO1	Identify and classify failures in the given system based on observed behavior and outputs.
	CO2	Design test cases for the given application using appropriate software testing techniques.
	CO3	Design a structured test plan for a given application based on specified requirements.
	CO4	Execute test cases using appropriate automated testing tools
	CO5	Use appropriate tools to manage and track the software testing process.

Sr. No.	Suggested list of Experiments	Ref.
1	Given an application code, perform code inspection to identify failures and classify them.	1, 2
2	Design test cases for the application using boundary value analysis and equivalence class partitioning as black box testing techniques.	1, 2
3	For the given requirements, write an application logic and design test cases using basis path testing as a white box testing technique.	1, 2
4	Design a test plan document for the given application.	1, 2
5	Perform unit testing using Junit as unit testing framework.	3
6	i. Design and execute automated test cases for a web application using Selenium. ii. Testing web application using Playwright. iii. API testing using Postman.	4, Online Resource 4, 5
7	Planning and managing software testing activities using Qase.	Online Resource 1
8	Defect management for a software application using JIRA.	Online Resource 2
9	Mini project in a group of 2-3 students to develop an application for the given problem statement and then design test cases for the same. Perform actual testing based on designed test cases using automated testing tools. Use of Jenkins for test automation pipelines is recommended.	-

Course Assessment: -
Lab:

ISE: Laboratory ISE is divided into two components: 25 marks for submission of experiments and 25 marks for oral/practical evaluation.

Recommended Books:

1. Naresh Chauhan, "Software Testing Principles and Practices", Oxford Higher Education, 2nd Edition
2. Kshirasagar Naik and Priyadarshi Tripathy, "Software Testing and quality assurance theory and practice", Wiley Publication, 1st Edition
3. Catalin Tudose, "Junit in Action", Manning Publishing, 3rd Edition
4. Mark Collin, "Mastering Selenium WebDriver", Packt Publishing, 1st Edition

Online Resources:

1. https://docs.qase.io/ - Test case management with Qase, official documentation
2. https://confluence.atlassian.com/jira - Official JIRA documentation
3. https://www.jenkins.io/doc/ - Jenkins official documentation
4. https://playwright.dev/docs/intro - Playwright official documentation
5. https://learning.postman.com/docs/introduction/overview - Postman official documentation

Suggested CO - PO articulation Matrix

Course Outcomes	**Programme Outcomes (POs)**										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	2	3	1	2	1	-	-	-	-	-	-
CO2	2	3	3	2	1	-	-	-	-	-	-
CO3	2	2	3	2	1	-	-	-	1	-	-
CO4	1	2	2	1	3	-	-	-	-	-	-
CO5	1	1	1	-	3	-	-	-	-	-	-

Legends :- High: 03, Medium: 02, Low: 01, No Mapping: -

Blooms level

Remember	Understand ✓	Apply ✓	Analyze ✓	Evaluate	Create

Course Code	Course Name	Teaching Scheme (Hrs/week)				Credits Assigned			
		L	T	P	SL	L	T	P	Total
25PEC13CS21	Social Media Analytics	2	1	-	3	2	1	-	3
		Examination Scheme							
			ISE	MSE	ESE		Total		
					Min	Max			
		Theory	20	30	20	50	100		
		Tut	50	--	--	--	50		

Pre-requisite Course Codes		--
After the successful completion students should be able to:		
Course Outcomes	CO1	Explain Concept of Social Media Analysis
	CO2	Compute network measures of a social media networks
	CO3	Analyze and review different social media data
	CO4	To use different social media analytics tools effectively and efficiently.
	CO5	Apply information filtering for recommendation system.
	CO6	Explain social media applications, privacy policies, and associated risks.

Module No.	Unit No.	Topics	Ref.	Hrs.
1		**Social Media Analytics: An Overview**	1,2	2
	1.1	Core Characteristics of Social Media, Types of Social Media, Social media landscape, Need for Social Media Analytics (SMA), SMA in small & large organizations. Purpose of Social Media Analytics, Social Media vs. Traditional Business Analytics,		
	1.2	Seven Layers of Social Media Analytics, Types of Social Media Analytics, Social Media Analytics Cycle, Challenges to Social Media Analytics, Social Media Analytics Tools		
2		**Social Network Structure, Measures & Visualization**	1,2	6
	2.1	Basics of Social Network Structure - Nodes, Edges & Tie Describing the Networks Measures - Degree Distribution, Density, Connectivity, Centralization, Tie Strength & Trust Network Visualization - Graph Layout, Visualizing Network features, Scale Issues.		
	2.2	Social Media Network Analytics - Common Network Terms, Common Social Media Network Types, Types of Networks, Common Network Terminologies, Network Analytics Tools.		
3		**Social Media Text, Action & Hyperlink Analytics**	1,2	4
	3.1	Social Media Text Analytics - Types of Social Media Text, Purpose of Text Analytics, Steps in Text Analytics, Social Media Text 8 Analysis Tools		
	3.2	Social Media Action Analytics - What Is Actions Analytics? Common Social Media Actions, Actions Analytics Tools		

	3.3	Social Media Hyperlink Analytics - Types of Hyperlinks, Types of Hyperlink Analytics, Hyperlink Analytics Tools		
4		**Social Media Location & Search Engine Analytics**	1,2	6
	4.1	Location Analytics - Sources of Location Data, Categories of Location Analytics, Location Analytics and Privacy Concerns, Location Analytics Tools		
	4.2	Search Engine Analytics - Types of Search Engines, Search Engine Analytics, Search Engine Analytics Tools		
5		**Social Information Filtering**	1,2	6
	5.1	Social Information Filtering - Social Sharing and filtering , Automated Recommendation systems, Traditional Vs social Recommendation Systems		
	5.2	Understanding Social Media and Business Alignment, Social Media KPI, Formulating a Social Media Strategy, Managing Social Media Risks		
6		**Social Media Analytics Applications and Privacy**	1	6
	6.1	Social media in public sector - Analyzing public sector social media, analyzing individual users, case study. Business use of Social Media - Measuring success, Interaction and monitoring, case study.		
	6.2	Privacy - Privacy policies, data ownership and maintaining privacy online.		
			Total	**30**

Self-Learning:

1. Self-learning hours include MOOCs, spoken tutorials, online resources, and extended study hours to enhance independent learning and better understanding of each module of the course content.
2. Evaluation of the self-learning components is carried out in all the evaluation components.

Course Assessment: -

Theory:

ISE:

ISE activities carry 20 marks. These activities will be conducted throughout the semester.

MSE:

The written summative examination of 30 marks based on 50% syllabus for 90 minutes.

ESE:

The written summative examination will be conducted for 50 marks based on the complete syllabus (20% questions on syllabus covered before MSE and 80% questions on the remaining syllabus) for 120 minutes.

Tut. No.	Suggested List of Tutorials	CO	Hrs.
1	Case Study–Based Tutorial on Social Media Analytics: Concepts, Landscape & Business Applications	1	2
2,3	Problem Solving based on the Social Media Network Measures	2	4
4	Study various – i) Social Media platforms (Facebook, twitter, YouTube etc) ii) Social Media analytics tools (Facebook insights, google analytics netlytics etc) iii) Social Media Analytics techniques and engagement metrics (page level, post level, member level) Applications of Social media analytics for business. e.g. Google Analytics https://marketingplatform.google.com/about/analytics/ https://netlytic.org/	4	2
5	Data Collection-Select the social media platforms of your choice (Twitter, Facebook, LinkedIn, YouTube, Web blogs etc), connect to and capture social media data for business (scraping, crawling, parsing).	3	2
6	Analyze the Social Media data. (e.g. Content Based Analysis :Topic , Issue ,Trend, sentiment/opinion analysis, audio, video, image analytics)	3	2
7	Problems solving on social media based recommendation system.	5	2
8	Implement content based and collaborative based filtering.	5	2
9	Design the creative content for promotion of your business on social media platform. Develop Risk management plan for promotion of product.	6	2
10	Analyze how Individual / Organization use social media and social media privacy policies.	6	2

Course Assessment

Tutorial:

ISE: Assessment shall be based on tutorials, evaluated through continuous assessment of analytical thinking, problem-solving skills, and accuracy of solutions.

Textbooks:

1. Seven Layers of Social Media Analytics_ Mining Business Insights from Social Media Text, Actions, Networks, Hyperlinks, Apps, Search Engine, and Location Data, Gohar F. Khan,(ISBN-10: 1507823207).
2. Analyzing the Social Web 1st Edition by Jennifer Golbeck
3. Mining the Social Web_ Analyzing Data from Facebook, Twitter, LinkedIn, and Other Social Media Sites, Matthew A Russell, O'Reilly
4. Charu Aggarwal (ed.), Social Network Data Analytics, Springer, 2011

Reference Books:

1. Social Media Analytics [2015], Techniques and Insights for Extracting Business Value Out of Social Media, Matthew Ganis, AvinashKohirkar, IBM Press
2. Social Media Analytics Strategy_ Using Data to Optimize Business Performance, Alex Gonçalves, APress Business Team
3. Social Media Data Mining and Analytics, Szabo, G., G. Polatkan, O. Boykin & A. Chalkiopoulus (2019), Wiley, ISBN 978-1-118-82485-6

Online Resources:

1. https://cse.iitkgp.ac.in/~pawang/courses/SC16.html
2. https://onlinecourses.nptel.ac.in/noc20_cs78/preview
3. https://nptel.ac.in/courses/106106146
4. https://7layersanalytics.com/

Suggested CO - PO articulation Matrix

Course Outcomes	**Programme Outcomes (POs)**										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3										
CO2	3	3									
CO3	3	3									
CO4	3				3						
CO5	3	3									
CO6	3	3					3				

Legends:- High: 03, Medium: 02, Low: 01, No Mapping: -

Blooms level

Remember	Understand ✓	Apply ✓	Analyze ✓	Evaluate	Create

Course Code	Course Name	Teaching Scheme (Hrs/week)				Credits Assigned			
		L	T	P	SL	L	T	P	Total
25PEC13CS31	Graph Data Science	2	1	--	3	2	1	-	3
		Examination Scheme							
						ESE			
			ISE	MSE		Min	Max	Total	
		Theory	20	30		20	50	100	
		Tutorial	50	--		--	--	50	

Pre-requisite Courses		Data Structures, Design and Analysis of Algorithms, Database Management Systems
After the successful completion students should be able to:		
Course Outcomes	CO1	Explain the fundamental concepts of graph theory and graph data science.
	CO2	Model real-world problems using graph databases and query them using Cypher.
	CO3	Apply graph algorithms for pathfinding and centrality measurements.
	CO4	Analyze networks using community detection and similarity algorithms.
	CO5	Apply graph machine learning techniques including graph embeddings.
	CO6	Design and develop scalable graph-based solutions for predictive modeling.

Module No.	Unit No.	Topics	Ref.	Hrs.
1		**Introduction to Graphs and Graph Theory**	1	3
	1.1	• Basics of Graph Theory • Graph Representations: Adjacency Matrix, Adjacency List • Types of Graphs: Directed, Undirected, Weighted, Bipartite, Planar, Trees, Cyclic, Acyclic		
	1.2	• Graph Traversal Algorithms: Breadth-First Search (BFS), Depth-First Search (DFS) • Applications: Networks, Social Graphs, Biological Graphs, Web Graphs		
2		**Fundamental Graph Algorithms**	2	4
	2.1	• Shortest Path Algorithms: Dijkstra, Bellman-Ford, Floyd-Warshall • Minimum Spanning Trees: Kruskal's and Prim's Algorithm • Eulerian and Hamiltonian Graphs		
	2.2	• Topological Sorting (Kahn's Algorithm, DFS-based approach) • Strongly Connected Components (Kosaraju's and Tarjan's Algorithm)		
3		**Graph Databases and Query Languages**	3,6	5
		• Introduction to Graph Databases		

		• Graph Data Models: Property Graphs, RDF Graphs • Query Languages: Cypher (Neo4j), Gremlin, SPARQL • Comparative Study of Graph Databases (Neo4j, ArangoDB, JanusGraph) • Case Study: Social Network Analysis, Recommendation Systems		
4		**Graph Machine Learning & Network Science**	4	5
	4.1	• Graph Neural Networks (GNNs) and Their Applications • Node Classification, Link Prediction, Community Detection		
	4.2	• Graph Embeddings: Node2Vec, DeepWalk, GraphSAGE • Random Walks and PageRank Algorithm • Applications: Fraud Detection, Drug Discovery, Knowledge Graphs		
5		**Large-scale Graph Processing & Distributed Computing**	7	6
	5.1	• Large-scale Graph Processing Frameworks: Google Pregel, Apache Giraph • Distributed Graph Computing: Apache Spark GraphX, Deep Graph Library (DGL), PyTorch Geometric (PyG)		
	5.2	• Parallel Graph Processing Techniques • Graph Theory in Artificial Intelligence and Natural Language Processing (NLP) • Industry Case Study: Graphs in Cybersecurity, Healthcare Analytics		
6		**Applications of Graph Data Science**	7	3
	6.1	Recommendation Engines, Fraud Detection, Supply Chain routing, Knowledge Graphs		
			Total	**26**

Self-Learning:

1. Self-learning hours include MOOCs, spoken tutorials, online resources, and extended study hours to enhance independent learning and better understanding of each module of the course content.
2. Evaluation of the self-learning components is carried out in all the evaluation components.

Suggested List of Tutorials

1. Practice problems on BFS, DFS, and graph representations.
2. Problem-solving on shortest paths and MST algorithms.
3. Solve problems on Dijkstra and Bellman-Ford.
4. Writing queries in Cypher (Neo4j).
5. Understanding and implementing PageRank.

6. Apply Louvain method for detecting graph communities.
7. Hands-on example with GraphX or PyG.
8. Use Node2Vec on a real-world dataset.
9. Explore graphs in fraud detection or bioinformatics.
10. Research paper discussion on the latest GNN models and Graph Data Science Advancements

Course Assessment:

Theory:

ISE:
ISE activities carry 20 marks. These activities will be conducted throughout the semester.

MSE:
The written summative examination of 30 marks based on 50% syllabus for 90 minutes.

ESE:
The written summative examination will be conducted for 50 marks based on the complete syllabus (20% questions on syllabus covered before MSE and 80% questions on the remaining syllabus) for 120 minutes.

Tutorial:

ISE: Assessment shall be based on tutorials, evaluated through continuous assessment of analytical thinking, problem-solving skills, and accuracy of solutions.

Recommended Books:

1. Introduction to Graph Theory ,Douglas B. West , Pearson Education
2. Algorithms ,Sanjoy Dasgupta, Christos H. Papadimitriou, Umesh Vazirani, McGraw-Hill Education
3. Graph Algorithms: Practical Examples in Apache Spark & Neo4j – Mark Needham, Amy E. Hodler , O'Reilly Media
4. Graph-Powered Machine Learning, Alessandro Negro, Manning Publications
5. Network Science, Albert-László Barabási , Cambridge University Press
6. Graph Data Science: An Introductory Guide, Neo4j Team, Neo4j
7. Deep Learning on Graphs, Yao Ma, Jiliang Tang, Cambridge University Press

Online Resources:

1. https://www.coursera.org/learn/graph-analytics
2. https://neo4j.com/graphacademy/
3. http://web.stanford.edu/class/cs224w/

Suggested CO - PO articulation Matrix

Course Outcomes	**Programme Outcomes (POs)**										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	2	-	-	-	-	-	-	-	-	-
CO2	2	3	3	2	3	-	-	-	-	-	-
CO3	3	3	2	2	3	-	-	-	-	-	-
CO4	2	3	3	3	3	-	-	-	-	-	-
CO5	2	2	3	3	3	-	-	-	-	-	-
CO6	2	2	3	3	3	-	-	-	-	-	-

Legends :- High: 03, Medium: 02,Low: 01, No Mapping: -

Blooms level

Remember	Understand ✓	Apply ✓	Analyze ✓	Evaluate	Create

Course Code	Course Name	Teaching Scheme (Hrs/week)				Credits Assigned			
		L	T	P	SL	L	T	P	Total
25PEC13CS22	AI in Cyber Security	2	1	--	3	2	1	--	3
		Examination Scheme							
			ISE	MSE	ESE		Total		
					Min	Max			
		Theory	20	30	20	50	100		
		Tut	50	--	--	--	50		

Pre-requisite Courses		Artificial Intelligence
After the successful completion students should be able to:		
Course Outcomes	CO1	Understand key security risks in AI models and applications
	CO2	Evaluate security defenses for AI models
	CO3	Analyze the impact of adversarial AI in cybersecurity
	CO4	Assess AI security governance, regulatory frameworks, and secure AI development methodologies
	CO5	Explore real-world case studies of AI security failures and defenses

Module No.	Unit No.	Topics	Ref.	Hrs.
1	**Foundations of AI Security**			
	1.1	Introduction to AI Security: • Overview of AI, Machine Learning, and Deep Learning Security. • Security challenges in AI-driven applications. • Threat Models in Adversarial Machine Learning.	1,2	3
	1.2	Understanding AI Vulnerabilities and Attack Surfaces: • Model tampering and AI security risks • Threat landscapes for AI in cybersecurity.	1,2	2
2	**AI Security Defenses and Robustness Strategies**			
	2.1	Adversarial AI Attacks and Threat Models: • Categories of adversarial attacks • Model tampering with Trojan horses and adversarial manipulations • Real-world AI attack case studies	1,2	3
	2.2	Defensive Strategies for AI Systems: • Supply chain vulnerabilities in AI models • Secure AI model deployment techniques • Protection against data poisoning and model theft	1,2	3
3	**AI in Cybersecurity – Adversarial AI and Cyber Threats**			
	3.1	Using AI for Cyber Threat Intelligence • AI-powered cyber-attacks and adversarial prompt injections	1,2	3

		• Generative AI threats: Deepfakes and adversarial content generation • AI-powered cybersecurity: AI in malware detection		
	3.2	Mitigation Strategies for AI-Driven Cyber Threats: • AI in phishing detection and fraud prevention • Security against adversarial prompts and AI prompt injection attacks	1,2	2
4	**AI Governance, Ethics, and Security Compliance**			
	4.1	AI Privacy Risks and Governance Challenges: • Model extraction attacks and adversarial privacy threats • Privacy risks in AI-powered applications • Model inversion and membership inference attacks	1,2	3
	4.2	Privacy-Preserving AI and Secure AI Development: • Differential privacy, homomorphic encryption • Secure AI frameworks and compliance	1,2	2
5	**Future of AI Security and Real-World Case Studies**			
	5.1	Security by Design in AI • Building trustworthy AI models • Secure model development and AI robustness testing • AI security compliance and best practices	1,2	3
	5.2	AI Security Operations and Continuous Monitoring: • AI security governance frameworks • AI risk management in industry applications • AI in cybersecurity operations (MLSecOps) • Case studies in AI security breaches	1,2	2
			Total Hours	**26**

Self-Learning:

1. Self-learning hours include MOOCs, spoken tutorials, online resources, and extended study hours to enhance independent learning and better understanding of each module of the course content.
2. Evaluation of the self-learning components is carried out in all the evaluation components.

Course Assessment: -

Theory:

ISE:

ISE activities carry 20 marks. These activities will be conducted throughout the semester.

MSE:

The written summative examination of 30 marks based on 50% syllabus for 90 minutes.

ESE:

The written summative examination will be conducted for 50 marks based on the complete syllabus (20% questions on syllabus covered before MSE and 80% questions on the remaining syllabus) for 120 minutes.

Sr. No.	Suggested Tutorials	Ref.	Hrs.
1	Case Study Analysis: AI Security Breaches Activity Type: Case Study Review Objective: Analyze real-world AI security breaches to understand vulnerabilities and mitigation strategies. Instructions: • Choose a real-world AI security incident (e.g., Google AI bias, Tesla autopilot failures, ChatGPT jailbreaks, etc.). • Discuss attack vectors, AI vulnerabilities, and failure points. • Suggest improvements and mitigation techniques. • Submission: 3–4-page report.	1,2	1
2	Group Discussion: The Future of AI Security Threats Activity Type: Group Discussion Objective: Encourage students to explore emerging threats in AI security. Instructions: • Divide students into groups. Each group presents a potential future AI security threat (e.g., AI-powered malware, adversarial deepfakes, autonomous system hacking). • Discuss ethical, technical, and regulatory challenges. • Propose risk mitigation strategies.	1,2	1
3	Poster Making: AI Attack & Defense Strategies Activity Type: Creative Poster Making Objective: Visually represent AI adversarial attacks and corresponding defenses. Instructions: • Create a poster explaining adversarial attacks and defense mechanisms (e.g., FGSM attack vs. Adversarial Training, Model Stealing vs. Encryption). • Include real-world examples. • Present and explain to the class.	1,2	1
4	Research Assignment: AI Governance & Legal Frameworks Activity Type: Research-Based Assignment Objective: Explore global and Indian regulatory frameworks for AI security. Instructions: • Compare AI security laws like GDPR, NIST AI RMF, EU AI Act, Indian AI regulations. • Discuss compliance requirements for AI models, data privacy, and ethical AI development. • Propose regulatory improvements for securing AI systems.	1,2	1

5	Simple Implementation: Adversarial Thinking Exercise Activity Type: Threat Modeling Exercise (No Coding) Objective: Develop adversarial thinking for AI security. Instructions: • Provide an AI-based system scenario (e.g., autonomous vehicle AI, AI in healthcare, chatbot AI). • Students must identify potential threats and vulnerabilities. • Suggest security defenses.	1,2	1
6	Demo/Presentation: Deepfake & Generative AI Security Activity Type: Student Presentations (Conceptual) Objective: Explore security concerns in deepfake and generative AI models. Instructions: • Each student picks a subtopic (e.g., how deepfakes are made, detecting deepfakes, deepfake regulations, GAN security). • Present findings using real-world case studies. • Discuss threats and countermeasures.	1,2	1
7	Case Study: AI in Cybersecurity - Strengths & Risks Activity Type: Case Study Analysis Objective: Understand the role of AI in cybersecurity and its risks. Instructions: • Choose an AI-powered security system (e.g., AI in malware detection, phishing detection, threat intelligence). • Explain how AI enhances cybersecurity and discuss limitations and risks (e.g., bias, adversarial attacks).	1,2	1
8	Discussion Panel: Ethical Dilemmas in AI Security Activity Type: Panel Discussion Objective: Debate ethical issues in AI security. Instructions: • Assign students roles (e.g., AI ethics expert, security analyst, government regulator, hacker, business leader). • Discuss topics like privacy vs. security, AI bias, accountability in AI security breaches.	1,2	1
9	Mock AI Security Policy Design Activity Type: Policy Drafting Exercise Objective: Draft an AI security policy for a hypothetical AI company. Instructions: • Define AI security guidelines for model development, adversarial attack protection, data privacy. • Align with global AI security regulations.	1,2	1
10	Research Report: AI & Supply Chain Security Risks Activity Type: Research Paper Objective: Study AI supply chain threats and propose security measures. Instructions: • Identify risks in the AI supply chain (e.g., model theft, poisoned datasets, software dependencies). • Research real-world cases of AI supply chain attacks (or industry specific case).	1,2	1

11	Debate: "Can AI Fully Secure Itself?" Activity Type: Debate Objective: Evaluate if AI can be fully self-secured. Instructions: • Split students into "YES" and "NO" teams. • Debate autonomous AI security, AI-driven SOC (Security Operations Center), self-healing AI models.	2	2
12	Write an article: "The Future of AI Security" Activity Type: Writing Assignment Objective: Encourage students to speculate on the future of AI security. Instructions: • Write a 3-page article on where AI security is headed. • Topics: Autonomous AI security, AI vs. AI cyber warfare, future regulatory landscapes.	1,2	2
Total Hours			14

Tutorial Assessment:
ISE: Assessment shall be based on tutorials, evaluated through continuous assessment of analytical thinking, problem-solving skills, and accuracy of solutions.

Recommended Books:

1. Hu, Fei, and Xiali Hei, eds. AI, Machine Learning and Deep Learning: A Security Perspective. CRC Press, 2023.
2. John Sotiropoulos, Adversarial AI: Attacks, Mitigations, and Defense Strategies (2024), Packt Publishing
3. Parisi, Alessandro. Hands-On Artificial Intelligence for Cybersecurity: Implement smart AI systems for preventing cyber-attacks and detecting threats and network anomalies. Packt Publishing, 2019.

Online Resources:

1. https://www.europarl.europa.eu/topics/en/article/20230601STO93804/eu-ai-act-first-regulation-on-artificial-intelligence, https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX%3A32024R1689
2. https://www.niti.gov.in/sites/default/files/2023-03/National-Strategy-for-Artificial-Intelligence.pdf
3. https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX%3A32024R1689

Suggested CO - PO articulation Matrix

Course Outcomes	**Programme Outcomes (POs)**										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	2	-	2	-	-	-	-	-	-	-
CO2	3	3	2	2	-	-	-	-	-	-	-
CO3	3	3	-	3	-	-	-	-	-	-	-
CO4	2	2	-	2	-	2	3	2	2	2	-

CO5	2	3	-	2	-	-	2	3	3	2	-

Legends: - High: 03, Medium: 02, Low: 01, No Mapping: -

Blooms level

Remember	Understand	Apply ✓	Analyse ✓	Evaluate ✓	Create

Course Code	Course Name	Teaching Scheme (Hrs/week)				Credits Assigned			
		L	T	P	SL	L	T	P	Total
25PEC13CS32	FinTech	2	1	--	3	2	1	--	3
		Examination Scheme							
			ISE	MSE		ESE		Total	
						Min	Max		
		Theory	20	30		20	50	100	
		Tutorial	50	--		--	--	50	

Pre-requisite Courses	Blockchain	
After the successful completion students should be able to:		
Course Outcomes	CO1	Analyze the evolution and growth of FinTech globally and recognize its impact on financial systems.
	CO2	Analyze the FinTech Ecosystem and Stakeholders
	CO3	Evaluate Digital Payment Systems and Mobile Wallets
	CO4	Apply blockchain, AI in Fintech application

Module No.	Unit No.	Topics	Ref.	Hrs.
1		**Introduction to FinTech and Digital Finance**	1	**5**
	1.1	**FinTech Overview** – Definition and evolution of FinTech Key segments in FinTech: Digital payments, lending, wealth management, blockchain, InsurTech FinTech ecosystems: Startups, incumbents, and regulators Global FinTech landscape and trends Key drivers of FinTech growth		
	1.2	**Financial Services vs. FinTech: Traditional vs. Disruptive Models** Traditional financial services: Banks, insurance, stock exchanges How FinTech disrupts traditional finance Benefits and challenges of FinTech innovation Key players in the FinTech ecosystem (e.g., PayPal, Square, Revolut, Stripe)		
2		**FinTech Ecosystem in India**	2	**5**
	2.1	**Understanding the FinTech Ecosystem** Overview of the FinTech ecosystem: Key players and stakeholders **Startups**: FinTech firms, accelerators, incubators		

		Incumbents: Banks, traditional financial institutions, and their role in the ecosystem **Regulators**: Role of financial regulatory bodies (e.g., SEC, FCA, CFTC) in shaping the ecosystem **Investors**: Venture capital, private equity, crowdfunding, and angel investors in FinTech **Technology Providers**: Cloud services, APIs, data platforms, blockchain developers Collaboration between startups and incumbents Challenges of operating within a fragmented ecosystem Key market trends and the rise of partnerships and ecosystems within FinTech		
	2.2	**Key Players in the FinTech Ecosystem** • **Banks and Financial Institutions**: Their adaptation to FinTech through partnerships and internal innovation (e.g., open banking, API integrations) • **Startups**: The role of challenger banks and disruptive FinTech firms • **Tech Giants**: The involvement of tech companies like Google, Apple, and Amazon in digital payments and financial services • **Regulatory Bodies**: How regulators (e.g., PSD2, GDPR) influence FinTech development • **Investors and VCs**: The impact of funding and venture capital on FinTech innovation (e.g., Sequoia Capital, Andreessen Horowitz)		
3		**Digital Payments and Mobile Wallets**	3	**6**
	3.1	**Types of digital payments**: Online payments, contactless payments, mobile payments, Payment gateways, processing networks, and acquiring banks, Key players in the payment ecosystem: Visa, MasterCard, PayPal, Stripe, and fintech startups, Advantages of digital payments: Speed, accessibility, and security		
	3.2	**Mobile Wallets and Peer-to-Peer Payments** : introduction to mobile wallets (Apple Pay, Google Pay, Samsung Pay, Venmo) Key technologies: NFC, QR codes, tokenization, and biometrics Peer-to-peer (P2P) payment systems: Venmo, PayPal, Zelle Security and fraud prevention in digital payment systems		
4		**Role of Blockchain, AI in Finance**	4	10
	4.1	**Introduction to blockchain**: Structure, decentralization, consensus algorithms, Types of blockchains: Public, private, and consortium, Blockchain use cases beyond cryptocurrencies: Smart contracts, decentralized applications (DApps), Blockchain in financial services: Payments, remittances,		

		and clearing		
	4.2	**Cryptocurrency and Decentralized Finance (DeFi)** Overview of cryptocurrencies: Bitcoin, Ethereum, and altcoins Blockchain's role in cryptocurrency and DeFi (Decentralized Finance), Decentralized exchanges (DEXs), liquidity pools, and smart contracts, Risks and challenges in cryptocurrency markets, Regulation and compliance in crypto markets		
	4.3	**AI applications in finance**: Algorithmic trading, robo-advisors, fraud detection Predictive analytics in risk management and credit scoring		
			Total	**26**

Self-Learning:

1. Self-learning hours include MOOCs, spoken tutorials, online resources, and extended study hours to enhance independent learning and better understanding of each module of the course content.
2. Evaluation of the self-learning components is carried out in all the evaluation components.

Course Assessment: -

Theory:

ISE:

ISE activities carry 20 marks. These activities will be conducted throughout the semester.

MSE:

The written summative examination of 30 marks based on 50% syllabus for 90 minutes.

ESE:

The written summative examination will be conducted for 50 marks based on the complete syllabus (20% questions on syllabus covered before MSE and 80% questions on the remaining syllabus) for 120 minutes.

Exp. No.	Suggested list of tutorials
1	Interactive discussion on the shift from traditional banking to digital banking.
2	Group analysis of real-world FinTech companies and their innovation
3	Hands-on activity to understand how blockchain transactions work.
4	Simulation of a simple smart contract
5	Demonstration of a mobile wallet application.
6	Group discussion on the pros and cons of digital payments.
7	Case study analysis of AI applications in major FinTech firms.
8	Scenario-based exercise on handling a data breach in a FinTech company.
9	**FinTech Company Analysis: Business Models and Innovations** : Choose a FinTech company (e.g., Stripe, Revolut, Robinhood, Square, or any other notable FinTech firm). Provide a detailed overview of the company's business model

	(e.g., how it makes money, what services it offers).Analyze the innovation introduced by the company in the financial space. Discuss the company's market positioning, competition, and customer base. Evaluate the impact of the company on traditional finance and its contribution to financial inclusion or disruption. Conclude by assessing the company's prospects in the FinTech sector. A report (1500-2000 words) summarizing the findings. And PowerPoint presentation (5-10 slides) to highlight key insights.
10	**Designing a Digital Payment System** Identify a specific financial problem or pain point related to payments (e.g., international remittance, peer-to-peer payments, or mobile payments in underserved areas). Design a digital payment system that addresses this problem. Outline the key features of your system (e.g., payment channels, security features, user experience). Explain the technology behind the system (e.g., QR codes, NFC, blockchain). Discuss the potential regulatory challenges and compliance requirements for the system. Consider the scalability, security, and accessibility of your solution. Deliverables: A detailed project report (1000-1500 words) explaining your payment solution. A prototype/mockup of the digital payment system (could be through wireframes or an app design tool).

Tutorial:

ISE: Assessment shall be based on tutorials, evaluated through continuous assessment of analytical thinking, problem-solving skills, and accuracy of solutions.

Recommended Books:

1. "FinTech: The New DNA of Financial Services", by Pranay Gupta, T. M. Vinod Kumar, 1st Edition, Publisher: De Gruyter
2. "The FinTech Book", by Susanne Chishti & Janos Barberis, first edition, John Wiley & Sons Publication
3. "Digital Payments in India: Background, Trends and Opportunities" Jaspal Singh, published by New Century Publications.
4. "Mastering Bitcoin: Unlocking Digital Cryptocurrencies", by Andreas M. Antonopoulos, O'Reilly Media publication

Online Resources:

1. https://www.udemy.com/course/fintech-fundamentals/learn/lecture/33707706#overview

Suggested CO - PO articulation Matrix

Course Outcomes	**Programme Outcomes (POs)**										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	2							3		3
CO2	3		2					2		2	3
CO3	3					2			3		3
CO4	3		2				2		3		3

Legends :- High: 03, Medium: 02,Low: 01, No Mapping: -

Blooms level

Remember	Understand ✓	Apply ✓	Analyze ✓	Evaluate ✓	Create

Course Code	Course Name	Teaching Scheme (Hrs/week)				Credits Assigned			
		L	T	P	SL	L	T	P	Total
25PEC13CS23	UI/UX Design	2	1	-	3	2	1	--	3
		Examination Scheme							
			ISE1	MSE	ISE2	ESE	Total		
		Theory	20	30	20	30	100		
		Tut	20	--	30	--	50		

Pre-requisite Courses		--
After the successful completion students should be able to:		
Course Outcomes	CO1	Understand the Foundations of UI/UX Design.
	CO2	apply Design Thinking and Process Models.
	CO3	Conduct Effective UX Research and Create User-Centric Artifacts.
	CO4	Design Interactive Prototypes and Evaluate Usability.
	CO5	Leverage Generative AI for Enhancing UI/UX Workflows.

Module No.	Unit No.	Topics	Ref.	Hrs.
1	1.1	**Introduction to UI/UX:** UI vs UX, A Day in the life of a UX Designer, Different design roles in the industry, Agile and Waterfall Process, Design System Fundamentals	1,2,4	3
2	2.1	**Design Process and Thinking Method:** Solving Problems with UI/UX, UX problems with Examples, Introduction to Design Thinking Process, Double Diamond Model, Case study Examples, Measuring Design Impact, Introduction to Research in UX, Choosing the Right Research Method	1,2	6
3	3.1	**Research to Discover:** User interview, UX Surveys, Field Study, Stakeholder Interview, Well Conducted Primary Research Examples, Writing aUX Research Report	1,2,3	5
	3.2	**Research to Explore:** Competitive analysis, Persona creation, Empathy Map, User flows, Customer Journey Map, User stories with poor and better examples, Card sorting with examples		
4	4.1	**Introduction to UI:** Atomic Design Principle, UI Elements, UI Design Principles, 5 Levels of UI Design Skills, Low Fidelity Wireframes, Mid Fidelity Wireframes, High Fidelity Wireframes	5	4
	4.2	**Introduction to UX Design:** UX design patterns, Laws of UX with practical examples, Principles of making Good UX Design, Interaction Design Process		
5	5.1	**Usability Evaluation**: Intro to usability evaluation, Types of usability evaluation, Quantitative and qualitative evaluation, User loads (Cognitive, motor and visual), Conducting a usability study	5	8

	5.2	**Application of Generative AI in the UI/UX:** Introduction to Generative AI in UI/UX, Generative AI applications in assisting UX/UI design, AI for Visual Design, AI for Rapid Prototyping, Generative AI for User Research, Predictive Analytics for UI/UX Optimization, Ethical Considerations of AI in Design, Future Trends in AI for UI/UX, How designers and AI can work together, Creating Effective Text Prompt for UX Design		
			Total	**26**

Self-Learning:

1. Self-learning hours include MOOCs, spoken tutorials, online resources, and extended study hours to enhance independent learning and better understanding of each module of the course content.
2. Evaluation of the self-learning components is carried out in all the evaluation components.

Course Assessment: -

Theory:

ISE:

ISE activities carry 20 marks. These activities will be conducted throughout the semester.

MSE:

The written summative examination of 30 marks based on 50% syllabus for 90 minutes.

ESE:

The written summative examination will be conducted for 50 marks based on the complete syllabus (20% questions on syllabus covered before MSE and 80% questions on the remaining syllabus) for 120 minutes.

To be Taught in laboratory			
	Topics	**Ref.**	**Hrs.**
1	**Installation of Figma. Demonstration of the working** • Figma Interface • Frames • Images: Raster and Vector • Vector Networks &Booleans • Masks • Gradients • Plugins	1,2, 3	1
2	**Demonstration of the working Graphic Design** • Balance • Contrast • Emphasis • Unity • Alignment • Hierarchy • Proportion • White Space	1,3	1

	• Typography • Color in UI Design		
3	**Know your client [Design UI Prototype only]** - UI design for AI chat for mental health. - UI design for chatbot for bank KYC. - UI design for teaching Math subject to school children (age 7 to 14). - UI design for teaching Physics subject to school children (age 7 to 14). - UI design for teaching Chemistry subject to school children (age 7 to 14). - UI design for teaching Biology subject to school children (age 7 to 14). - UI design for teaching Geometry subject to school children (age 7 to 14). - UI design for Math using Abacus. - UI for No-dues process at SPIT. - UI for online defaulter tracking for subject/ lab at SPIT. - UI for Fire Safety at School/College/Restaurant/Society/Railway stations etc - Online Bhelwala - Student's IV planner - Online Defaulter system for school/colleges - Women's entrepreneurship - Deaf and Dumb …Traffic direction	1,2, 3	2
4	**Goal oriented design** - Create a UI design for [an experience] for a counselor / psychiatrist to analyze mental fitness of students. - Create a UI design for a building contractor to monitor daily given work on the site. - Create a UI design for a TeaShop owner to visualize the quality, feedback and suggestions about the tea from the various customers.	1,2	2
5	**Design Principles** - UI design for boost cleaning and hygiene at SPIT. - UI design for real time sport system for maharashtra system. - UI design for fake social media profile detection and reporting. - UI design for vehicle tracking system for samruddhi marg. - UI design for Nasha Mukti. - UI for Lab Network health monitoring at SPIT. - Crop Health monitoring. - Mental health monitoring of autistic childrens.	1,2	2
6	**Menus & Navigation** - UI for women representation in gram panchayat. - UI for linkedin for needy, searching for a job.	1,4	1

	- UI for college students' accommodation, online paying guests.
	- UI for No-dues use application at SPIT.
	- UI for population control and traffic management at metro.
	- UI for child care.[baby sitting]
	- UI for pet care.
	- UI design for mumbai dabbawala (tiffin wala).
	- UI design for operation theater.
	- UI design for plant disease detection.
	- UI design for Ludo game.
	- UI design for onion health monitoring for farmers.
	- UI design for online C.A for startup.
	- UI design for online college recommendation for HSC students.
	- UI design for women safety (by keeping blind in mind).
	- UI design for physiotherapy.
	- UI design for navigation control design for visual impact.
	- UI design for the stock market.
	- Patients visiting hospitals
	- Students visiting University of Mumbai
	- Visitors visit to Museum
	- Food quality monitoring for Canteen
	- Ayurvedika: Leaf identification and its usage as medicine, product recommendation
	- Citizen safety app against cybercrime
	- Plate number recognition [PNR] system for cities/taluka
	- SPAM detection
	- Fake news detection
	- **Icon Design:**
	  - R.T.O
	  - College hygiene
	  - Agriculture
	  - Sanitization, Cleaning, nutrition
	  - Fire safety at Shree Ganesh Festival
	  - Fire safety at Society
	  - FIre Safety at SPIT
	  - Fire safety at Schools
	- **Colors:**
	  - Save Water
	  - Save Electricity
	  - Save Earth
	  - Reduce CO2
	  - Food processing manufacturing plant.
	  - Automotive Industry		

	- Children's Hosptial The following experiments from 7 and 8 will be based on Any one of the problem statements given from Experiment no. 6 and will be implemented as per the themes mentioned for each.		
7	Graphic Designing with Figma Windows & Screen controls, Colors Selection, Map Based UI	1	1
8	**UI/UX Design Capstone Project** A full-fledged capstone project with 2 to 3 projects as options, **1. Personalized Health Coach App** Users often struggle to achieve and maintain their fitness goals due to generic workout and diet plans that don't account for individual needs. The problem statement here is to create an app that helps users create a personalised fitness plan with the help of top-rated personal fitness trainers or health coaches. **2. Smart Travel Assistant App** Travellers frequently face difficulties in planning trips due to the abundance of information and lack of personalised recommendations that cater to their preferences. The problem statement here is to create a smart travel assistant app that helps personalize travel plans in minutes. **3. AI-Enhanced Customer Support Platform** Businesses often encounter challenges in managing a high volume of customer queries with limited human resources, leading to slow response times and customer dissatisfaction. The problem statement here is to create a web application that helps reduce the turnaround time and increases customer satisfaction. **4. Intelligent Content Recommendation System** Users on media and e-commerce websites often struggle to find relevant content due to overwhelming options and generic recommendations. The problem statement here is to build an intelligent system that recommends the best content as per user preference.	1	3
	Total		13

Course Assessment:

Tutorial:

ISE: Assessment shall be based on tutorials, evaluated through continuous assessment of analytical thinking, problem-solving skills, and accuracy of solutions.

Recommended Books:

1. Krug, Steve. Don't Make Me Think, Revisited: A Common-Sense Approach to Web Usability. New Riders.
2. Norman, Don. The Design of Everyday Things. Basic Books, A Member of the Perseus Books Group, New York.
3. Allen, Jesmond, and James Chudley. Designing User Experiences. New Riders

4. Unger, Russ, and Carolyn Chandler. A Project Guide to UX Design: For User Experience Designers in the Field or in the Making. New Riders
5. Eyal, Nir. Hooked: How to Build Habit-Forming Products. Portfolio

Suggested CO - PO articulation Matrix

Course Outcomes	**Programme Outcomes (POs)**										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	2		2								
CO2	2		2								
CO3	2	3			2						
CO4	2		2								
CO5	2	3									

Legends :- High: 03, Medium: 02,Low: 01, No Mapping: -

Blooms level

Remember	Understand	Apply √	Analyze	Evaluate	Create

Course Code	Course Name	Teaching Scheme (Hrs/week)				Credits Assigned			
		L	T	P	SL	L	T	P	Total
25PEC13CS33	Computer Vision	2	1	--	3	2	1	--	3
		Examination Scheme							
			ISE	MSE	ESE		Total		
					Min	Max			
		Theory	20	30	20	50	100		
		Tut	50	--	--	--	50		

Pre-requisite Courses		Linear Algebra, Image Processing
After the successful completion students should be able to:		
Course Outcomes	CO1	Apply geometric transformations and camera calibration techniques.
	CO2	Analyze image features and perform feature matching.
	CO3	Develop stereo vision and depth estimation solutions.
	CO4	Implement Structure from Motion (SfM) for 3D reconstruction.
	CO5	Apply motion tracking and object recognition in real-world applications.

Module No.	Unit No.	Topics	Ref.	Hrs.
1	1.1	**Introduction to Computer Vision** • Definition, applications, and significance of Computer Vision • Image Processing vs Computer Vision vs Computer Graphics • Fundamental concepts: cameras, image formation, and perspective projection • Overview of modern Computer Vision applications	1,2	3
2	2.1	**Image Representation and Feature detection** • Image formation, pinhole camera model, and coordinate systems • Homogeneous coordinates and 2D projective transformations • Affine, perspective, and homography transformations • Camera calibration techniques (intrinsic and extrinsic parameters)	1,2,4	9
	2.2	• Edge and corner detection (Harris, Shi-Tomasi) • Scale-Invariant Feature Transform (SIFT) and Speeded-Up Robust Features (SURF) • Oriented FAST and Rotated BRIEF (ORB) features • Feature matching using RANSAC for robust homography estimation		
3	3.1	**Multi view Geometry** • Epipolar geometry and fundamental matrix • Essential matrix and relative pose estimation • Stereo vision and disparity maps	1,3,5	8

		• Triangulation and depth estimation • Multi-view geometry and bundle adjustment • Structure from Motion (SfM) pipeline • 3D point cloud generation and reconstruction • Applications in robotics and augmented reality		
4	**4.1**	**Object Detection and Deep Learning Integration** Classical Object Detection • Contour detection and shape analysis • Template matching • Hough Transform for shape detection	2	5
	4.2	**Deep Learning Integration in Vision** • CNNs as feature extractors (conceptual linkage to ML/DL course) • Transfer learning for vision tasks • YOLO pipeline overview for object detection		
5	**5.1**	**Motion Analysis and Tracking** • Optical flow: Lucas–Kanade method • Kalman filtering for motion prediction • Object tracking using Mean-Shift and CAMShift algorithms • Applications in robotics and video surveillance	1,6	3
			Total	**28**

Self-Learning:

1. Self-learning hours include MOOCs, spoken tutorials, online resources, and extended study hours to enhance independent learning and better understanding of each module of the course content.
2. Evaluation of the self-learning components is carried out in all the evaluation components.

Course Assessment: -

Theory:

ISE:

ISE activities carry 20 marks. These activities will be conducted throughout the semester.

MSE:

The written summative examination of 30 marks based on 50% syllabus for 90 minutes.

ESE:

The written summative examination will be conducted for 50 marks based on the complete syllabus (20% questions on syllabus covered before MSE and 80% questions on the remaining syllabus) for 120 minutes.

Suggested list of Tutorial			
	Topics	**Ref.**	**Hrs.**
1	Understand image formation, homogeneous coordinates, and transformations. • Load and display an image using OpenCV. • Convert images between color spaces (RGB ↔ Grayscale).	1,4,7	2

	• Apply affine transformations (scaling, rotation, translation) using • OpenCV.		
2	Calibrate a camera and compute intrinsic and extrinsic parameters. • Capture images of a chessboard for calibration. • Use OpenCV's cv2.calibrateCamera() function to estimate parameters. • Compute the perspective projection matrix and transform 3D points to 2D.	1,3,6, 7	2
3	Implement keypoint detection and matching techniques. • Apply Harris corner detection and Shi-Tomasi corner detection. • Detect features using SIFT, SURF, and ORB. • Match keypoints between two images using FLANN or Brute-Force Matcher.	1,4,7	2
4	Estimate the homography matrix and warp images to create a panorama. • Detect and match keypoints between two overlapping images. • Compute the homography matrix using RANSAC. • Warp and blend images to create a panoramic image.	1,3,7	2
5	Understand epipolar constraints in stereo vision. • Compute Fundamental Matrix (F) from corresponding points. • Plot epipolar lines on stereo image pairs. • Validate epipolar constraints by verifying that x'Fx = 0.	3,5,7	2
6	Implement stereo disparity estimation to compute depth. • Capture stereo image pairs or use an online dataset (e.g., KITTI). • Compute disparity maps using OpenCV's StereoBM() or StereoSGBM(). • Convert disparity maps to depth maps using camera parameters.	1,3,7	2
7	Recover a 3D scene structure from multiple 2D images. • Detect keypoints and track features across multiple images. • Compute the Essential Matrix (E) and recover camera motion. • Generate a 3D point cloud using triangulation.	3,5,7	2
8	Detect objects using contours and shape analysis. • Convert an image to binary and detect contours. • Fit bounding boxes, circles, and ellipses to objects. • Classify objects based on Hu Moments or HOG descriptors.	4,7	2
9	Track motion in videos using optical flow. • Compute Lucas-Kanade Optical Flow on a moving object. • Apply Dense Optical Flow (Farneback Method) to track object motion. • Use Kalman Filtering to predict the object's next position.	1,6,7	2
10	Implement real-time object tracking. • Track a moving object in a video using Mean-Shift Algorithm. • Improve tracking using CAMShift (adaptive Mean-Shift). • Compare results with Optical Flow-based tracking.	1,7	2
	Total		20

Course Assessment:

Tutorial:

ISE: Assessment shall be based on tutorials, evaluated through continuous assessment of analytical thinking, problem-solving skills, and accuracy of solutions.

Recommended Books:

1. "Computer Vision: Algorithms and Applications" – Richard Szeliski, 2nd Edition, Springer Publication
2. "Computer Vision: A Modern Approach" – David Forsyth, Jean Ponce, 2nd Edition, Pearson Publication
3. Multiple View Geometry in Computer Vision" – Richard Hartley, Andrew Zisserman 2nd Edition, Cambridge University Press
4. Feature Extraction and Image Processing for Computer Vision" – Mark S. Nixon, Alberto S. Aguado, 4th Edition, Academic Press (Elsevier)
5. An Invitation to 3D Vision" – Yi Ma, Stefano Soatto, Jana Košecká, and S. Shankar Sastry, 1st Edition, Springer
6. "Robotics, Vision and Control" – Peter Corke, 2nd Edition, Springer Publication
7. OpenCV Online Documentation (API & Tutorials) :https://docs.opencv.org/4.x/

Suggested CO - PO articulation Matrix

Course Outcomes	**Programme Outcomes (POs)**										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	2	-	-	3	-	-	-	-	-	-
CO2	3	3	-	-	3	-	-	-	-	-	-
CO3	3	3	2	-	3	-	-	-	-	-	-
CO4	3	3	-	-	3	-	-	-	-	-	-
CO5	2	3	-	-	3	2	-	-	-	-	2

Legends:- High: 03, Medium: 02, Low: 01, No Mapping: -

Blooms level

Remember	Understand ✓	Apply ✓	Analyze ✓	Evaluate	Create

Course Code	Course Name	Teaching Scheme (Hrs/week)				Credits Assigned			
		L	T	P	SL	L	T	P	Total
25PEL13CS11	Knowledge Representation and Ontology Lab	--	--	2	--		--	1	1
		Examination Scheme							
			ISE	MSE	ESE		Total		
					Min	Max			
		Theory	--	--	--	--	--		
		Lab	50	--	--	--	50		

Pre-requisite Courses		Database Management Systems, Web Programming
After the successful completion students should be able to:		
Course Outcomes	CO1	To Design and Implement ontologies for specific domains, including class definition, properties, instances, and relationships using tools like Protégé and OWL.
	CO2	To Query ontologies using SPARQL and Perform automated reasoning using tools like HermiT and Pellet to infer new knowledge.
	CO3	To Demonstrate the ability to model, query, and optimize semantic datasets by integrating relational and linked data using advanced SPARQL techniques and tools, enabling efficient data retrieval and interoperability across heterogeneous data sources.
	CO4	To Integrate knowledge representation methods with technologies such as Natural Language Processing (NLP) and/or Machine Learning (ML) to enhance automated reasoning, knowledge extraction, and decision-making in com lex s stems.
	CO5	To Apply ontologies in real-world scenarios, such as multi-agent s stems, expert systems, and the Semantic Web.

To be Taught in laboratory			
	Topics wise List of Experiments with relevant topic	**Ref.**	**COs**
1	Study Semantic Web open-source tools- Apache TinkerPro, RDFLib, Apache Jena, Protégé, Sesame.	2	1
2	Construct a Simple Ontology- Design and Create an ontology (RDF/OWL) to represent a domain. Define classes, object properties, data properties, and 、 instances. Tools: Protégé	2	1
3	Semantic Querying Using SPARQL Use SPARQL to use RDF dataset for data retrieval. Tool: Protégé	1	2
4	Reasoning with Description Logics: Load the ontology into Protégé. Use a reasoner to check for consistency, infer new facts, and classify concepts. Demonstrate how inferred knowledge can be extracted based on logical rules. Tools: HermiT or Pellet Reasoner (via Protégé)	2	2

Post MSE			
5	Semantic Web and Linked Data: Create linked datasets and use SPARQL to perform federated queries across multi le datasets. Tools: Protégé, SPARQL end point	4	3
6	Advanced SPARQL Queries and Optimization: Explore advanced SPARQL techniques such as filtering, aggregation, and optimization. query large datasets efficiently. .Tools: SPARQL end points, Apache Jena	1	3
7	Integrating Ontologies with Machine Learning: Use SPARQL queries to fetch relevant data from an ontology. Integrate the data into a machine learning model for classification or prediction. Tools: Python, scikit-learn, RDFLib, SPARQL	1, 4	4
8	NLP for Ontology Population : Extract entities and relationships from unstructured text. Populate an existing ontology using the extracted data using NLP. Tools: Stanford NLP, SpaCy , Protégé	3	4
9	Mini Project: (Group of 2-3 students) The Mini project will demonstrate the integration of concepts learned throughout the course. It will be evaluated based on the complexity of the problem, the design of the ontology or knowledge representation system, and the quality of the implementation showcased with presentation. Solve real world problem. (e.g. Ontology-based decision support system, Semantic data integration, etc.).	1,2, 3	5

Course Assessment: -
Lab:

ISE: Laboratory ISE is divided into two components: 25 marks for submission of experiments and 25 marks for oral/practical evaluation.

Recommended Books:

1. "Learning SPARQL: Querying and Updating with SPARQL l. I " by Bob DuCharme
2. "Knowledge Representation and Reasoning" by Ronald J. Brachman and Hector J. Levesque
3. "Semantic Web Technologies Trends and Research in Ontology-based Systems", John Davies, Rudi Studer and Paul Warren, Wiley, 2006 Edition
4. "Linked Data: Evolving the Web into a Global Data Space" by Tom Heath and Christian Bizer

Reference Books:

1. "Foundations of Artificial Intelligence: A Knowledge Representation Approach" by David W. Aha.
2. "Semantic Web for the Working Ontologist: Effective Modeling in RDFS and OWL" by Dean Allemang and James Hendler.
3. "The Description Logic Handbook: Theory, Implementation, and Applications" by Franz Baader et al.
4. "Foundations of Semantic Web Technologies" by Pascal Hitzler, Markus Krötzsch, and Sebastian Rudolph

Online Resources:

1. Protégé: https://protege.stanford.edu/
2. SPARQL: https://www.w3.org/TR/rdf-sparql-query/
3. OWL: https://www.w3.org/TR/owl-guide/
4. LINKED Data: https://www.coursera.org/learn/web-data#modules
5. Jena :https://jena.apache.org/documentation/inference/
6. Pellet Reasoner : https://github.com/stardog-union/pellet
7. Jess Expert System Shell: https://www.jessrules.com/
8. Stanford NLP: https://stanfordnlp.github.io/CoreNLP/

Suggested CO - PO articulation Matrix

Course Outcomes	**Programme Outcomes (POs)**										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	2	2	3	2	3	-	-	-	-	-	-
CO2	2	3	2	3	3	-	-	-	-	-	-
CO3	2	3	3	2	3	-	-	-	-	-	-
CO4	2	2	3	3	3	-	-	-	1	-	2
CO5	2	2	3	2	3	-	-	1	1	-	2

Legends:- High: 03, Medium: 02, Low: 01, No Mapping: -

Blooms level

Remember	Understand	Apply ✓	Analyze ✓	Evaluate	Create

Course Code	Course Name	Teaching Scheme (Hrs/week)				Credits Assigned			
		L	T	P	SL	L	T	P	Total
25PEL13CS12	Generative AI Lab	--	--	2	--		--	1	1
		Examination Scheme							
			ISE	MSE	ESE		Total		
					Min	Max			
		Theory	--	--	--	--	--		
		Lab	50	--	--	--	50		

Pre-requisite Courses		Artificial Intelligence
After the successful completion students should be able to:		
Course Outcomes	CO1	Apply deep neural network fundamentals for solving a given problem.
	CO2	Implement a variational autoencoder for a given generation task.
	CO3	Implement a Generative Adversarial Network for given generation task.
	CO4	Develop a transformer model for a given problem.
	CO5	Implement diffusion and stable diffusion model for the given problem.
	CO6	Develop a generative AI application for the given requirement.

To be Taught in laboratory			
	Topics wise List of Experiments with relevant topic	**Ref.**	**COs**
1	**Fundamentals of deep neural networks:** Convolutional Neural Networks, Recurrent Neural Networks, Setting up of TensorFlow lab Tasks: a. Develop an application for image classification using CNNs b. Develop an application of sentence completion using RNNs.	4,5	1
2	**Variational Autoencoders:** Autoencoder basics, Variational autoencoder building blocks Task: Develop an application to generate new images using variational autoencoder.	1,2	2
3	**Generative Adversarial Networks:** Basics of Generative Adversarial Network, Vanilla GAN architecture Tasks: a. Develop an application for generating an image using Vanilla GANs b. Style transfer using CycleGANs or PixtoPix GANs	1,2	3
4	**Transformer models:** Basics of transformer models, Fundamentals of language models- text tokenization, predicting probabilities, generating text, zero-shot and fewshot generalization Task: Implement a transformer model for text generation task.	3,6	4
5	**Diffusion models:** Basics of diffusion models, training a diffusion model, noise schedules,	3,6	5

	UNets fundamentals Task: Implement a diffusion model for image generation task.		
6	**Stable diffusion models:** Conditional diffusion models, latent diffusion, stable diffusion fundamental building blocks Task: Implement a stable diffusion model for image generation task.	3,6	5

Course Assessment: -
Lab:

ISE: Laboratory ISE is divided into two components: 25 marks for submission of experiments and 25 marks for oral/practical evaluation.

Recommended Books:

1. David Foster, "Generative Deep Learning- teaching machines to paint, write, compose and play", O'Reilly Media, 1st Edition
2. Joseph Babcock and Raghav Bali, "Generative AI with Python and TensorFlow2", Packt Publishing, 1st Edition
3. Omar Sanseviero, Pedro Cuenca, Apolinario Passos and Jonathan Whitaker, "Handson Generative AI with Transformers and Diffusion Models", 1st Edition

Online Resources :

1. https://www.coursera.org/learn/introduction-generative-ai
2. https://archive.nptel.ac.in/noc/courses/noc19/SEM1/noc19-cs18/
3. https://youtu.be/XfpMkf4rD6E?si=efzEk2GaaCqMfWqF – Stanford CS25: V2 |Introduction to transformers w/Andrej karpathy
4. https://youtube.com/playlist?list=PLoROMvodv4rPOWAomMM6STXaWW4FvJT8&si=hh44mZSLMBD7-2L8 - Stanford CS236: Deep Generative Models

Suggested CO - PO articulation Matrix

Course Outcomes	**Programme Outcomes (POs)**										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	2			3						
CO2	3	2			3						
CO3	3	3			3						
CO4	3	3			3						
CO5	3	3			3						
CO6	3	3	3		3	2	2	3	3		

Legends :- High: 03, Medium: 02, Low: 01, No Mapping: -

Blooms level

Remember	Understand	Apply ✓	Analyze	Evaluate	Create

Course Code	Course Name	Teaching Scheme (Hrs/week)				Credits Assigned			
		L	T	P	SL	L	T	P	Total
25PEL13CS13	Soft Computing Lab	--	--	2	--		--	1	1
		Examination Scheme							
			ISE	MSE	ESE		Total		
					Min	Max			
		Theory	--	--	--	--	--		
		Lab	50	--	--	--	50		

Pre-requisite Courses		Machine Learning, Python
After the successful completion students should be able to:		
Course Outcomes	CO1	Implement fuzzy logic control system for various real-world applications.
	CO2	Implement and train neural network models for a variety of tasks using different neural network architectures
	CO3	Apply genetic algorithms to solve optimization problems in various domains, such as function optimization, scheduling and engineering design problems.
	CO4	Solve real-world engineering and computational problems using hybrid soft computing techniques

To be Taught in laboratory			
Sr. No.	**Topics wise List of Experiments with relevant topic**	**Ref.**	**COs**
1	Introduction to Soft Computing • Objective: Understanding soft computing concepts and basic algorithms. • Theory: Introduction of soft computing, difference between soft computing and hard computing, Overview of Fuzzy Logic, Genetic Algorithms, Neural Networks. Practical: Setup and basic tools for Soft Computing, Introduction to MATLAB/Python.	1	1
2 3 4	Fuzzy Logic • Objective: Implement a fuzzy inference system. • Theory: Concepts of fuzzy sets, fuzzy rules, and fuzzy inference systems. • Practical list: ➢ Implement Fuzzy set operations (Union, Intersection, Difference and Complement) ➢ To Perform Union, Intersection, max-min and max-product composition of two fuzzy relations ➢ To implement controller using MAMDANI fuzzy model	2	1

5 6 7 8	Artificial Neural Networks (ANN) • Objective: Implement and train a neural network. • Theory: Understanding perceptrons, multi-layer perceptrons, and backpropagation. • Practical list: ➢ Implement an ANN for pattern recognition or regression using MATLAB or Python (with TensorFlow/Keras). ➢ Implement Error back propagation training algorithm (EBPTA) and solve XOR problem ➢ Write a program to implement Hopfield auto-associative memory ➢ Simulate Boltzmann Machine	3	2
9 10 11	Genetic Algorithm • Objective: Implement a basic genetic algorithm for optimization. • Theory: Basic concepts of Genetic Algorithm (Selection, Crossover, Mutation). • Practical list: ➢ Code a genetic algorithm to solve optimization problems such as the Traveling Salesman Problem. ➢ Genetic algorithm for job scheduling ➢ Genetic algorithm for knapsack problem	1,4	3
12 13	Hybrid Soft Computing Techniques • Objective: Combine fuzzy logic with neural networks or genetic algorithms. • Theory: Overview of hybrid systems. Practical list: ➢ Design a hybrid system for real-time applications. ➢ Case study of adaptive neuro fuzzy inference system (ANFIS)	1,3	4

Course Assessment: -
Lab:

ISE: Laboratory ISE is divided into two components: 25 marks for submission of experiments and 25 marks for oral/practical evaluation.

Recommended Books:

1. "Soft Computing: Techniques and Applications" by S. N. Sivanandam and S. N. Deepa
2. "Fuzzy Logic with Engineering Applications" by Timothy J. Ross
3. "Neural Networks: A Comprehensive Foundation" by Simon Haykin
4. "Introduction to Genetic Algorithms" by S. N. Sivanandam and S. Sumathi

Suggested CO - PO articulation Matrix

Course Outcomes	Programme Outcomes (POs)										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	3	2	-	3	-	2	-	-	-	-
CO2	3	3	-	3	3	-	-	-	2	2	-
CO3	3	3	2	2	3	-	-	-	-	-	-
CO4	3	3	3	3	3	-	2	2	-	-	2

Legends: - High: 03, Medium: 02, Low: 01, No Mapping: -

Blooms level

Remember	Understand	Apply ✓	Analyze ✓	Evaluate	Create

Course Code	Course Name	Teaching Scheme (Hrs/week)				Credits Assigned			
		L	T	P	SL	L	T	P	Total
25OE13CS4	Public Relations and Corporate Communication	2	--	--	2	2	--	--	2
			Examination Scheme						
			ISE	MSE		ESE		Total	
		Theory	100	--		--		100	
		Lab	--	--		--		--	

Pre-requisite Course Codes		SLRW Skills
After the successful completion students should be able to:		
Course Outcomes	CO1	Develop employable communication skills through practical usage
	CO2	Draft professional documents with precision
	CO3	Develop effective communication strategies for diverse, cultural and global business environment

Module No.	Unit No.	Topics	Ref.	Hrs.
1		**Career Advancement Skills**	R-1,3,	10
	1.1	Resume Writing & Cover Letter for Employment		
	1.2	Group Discussion		
	1.3	Impressive Grooming		
	1.4	Communication Simulation		
	1.5	Interview Techniques		
2		**Synergy Communication**	R-4	8
	2.1	Presentation Skills		
	2.2	Report Writing- Importance, Objective, types…		
	2.3	Meetings and Documentation: Notice, Agenda, Minutes		
3		**Cross-Cultural Communication**	R-2,4	4
	3.1	Cultural awareness		
	3.2	Language barriers		
	3.3	Global communication strategies		
4		**Corporate Identity and Branding**	R-5	4
	4.1	Corporate image and reputation		
	4.2	Branding strategies		
	4.3	Visual identity		
	4.4	Messaging and tone		
			Total	**26**

Sr. No.	List of Written and Oral Assignments	Marks
1	Resume/ Cover Letter	20
2	GD Practices	20
3	Mock Interview HR Question	20
4	Team Building Activity	20
5	Notice & Agenda and Minutes of the Meeting	20
	Total	100

Self-Learning:

1. Self-learning hours include MOOCs, spoken tutorials, online resources, and extended study hours to enhance independent learning and better understanding of each module of the course content.
2. Evaluation of the self-learning components is carried out in all the evaluation components.

Course Assessment:

ISE: a) Two activities as formative assessment for 50 marks based on 50% syllabus

b) Two activities as formative/Summative assessment for 50 marks based on remaining syllabus

Recommended Textbooks:

1. Dr. K.Alex, Soft Skills- Know Yourself & know the World, S.Chand
2. John Hayes, Interpersonal Skills at Work, McGraw Hill Education
3. Ankur Malhotra,Campus Placement: A Comprehensive Guide, McGraw Hill Education
4. Meenakshi Raman,Sangeeta Sharma, Communication Skills, Oxford, India
5. Courtland L. Bovee, Business Communication Today, Pearson

Suggested CO - PO articulation Matrix

Course Outcomes	Programme Outcomes (POs)										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	-	-	-	-	-	2	2	3	3	-	3
CO2	-	-	-	-	-	-	2	2	3	-	2
CO3	-	-	-	-	-	3	3	2	3	-	3

Legends :- High: 03, Medium: 02, Low: 01, No Mapping: -

Blooms level

Remember	Understand✓	Apply✓	Analyze	Evaluate	Create