Shirpur Education Society's

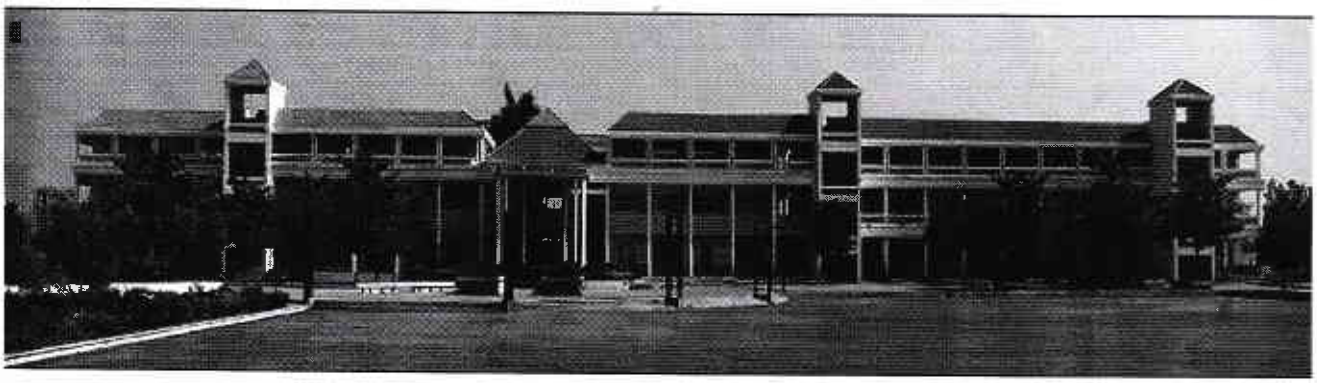
R. C. Patel Institute of Technology, Shirpur

(An Autonomous Institute)

Course Structure and Syllabus

Final Year B. Tech. (Computer Engineering)

With effect from Year 2026-27
(RCP23 NEP Scheme)



Shahada Road, Near Nimzari Naka, Shirpur, Maharashtra 425405
Ph: 02563 259 802, Web: www.rcpit.ac.in

Final Year B.Tech Computer Engineering Semester-VII/VIII (w.e.f. 2026-27)

Sr.No.	Course Category	Course Code	Course Title	Teaching Scheme			Evaluation Scheme					Total	Credits	
							Continuous Assessment (CA)							
							TA	Term Test 1 (TT1)	Term Test 2 (TT2)	Average of (TT1 & TT2)	ESE			
				L	T	P	[A]			[B]	[C]	[A+B+C]		
1	PC	RCP23CCPC701	Natural Language Processing	3			25	15	15	15	60	100	3	4
	PC	RCP23CLPC701	Natural Language Processing Laboratory			2	25				25	50	1	
2@	PE	RCP23CCPE711	Agentic AI	3			25	15	15	15	60	100	3	4
		RCP23CLPE711	Agentic AI Laboratory			2	25				25	50	1	
		RCP23CCPE712	Distributed Computing	3			25	15	15	15	60	100	3	
		RCP23CLPE712	Distributed Computing Laboratory			2	25				25	50	1	
		RCP23CCPE713	Metaverse	3			25	15	15	15	60	100	3	
		RCP23CLPE713	Metaverse Laboratory			2	25				25	50	1	
3@	PE	RCP23CCPE721	Social Network Analysis	3			25	15	15	15	60	100	3	4
		RCP23CLPE721	Social Network Analysis Laboratory			2	25				25	50	1	
		RCP23CCPE722	Virtual Infrastructure Systems	3			25	15	15	15	60	100	3	
		RCP23CLPE722	Virtual Infrastructure Systems Laboratory			2	25				25	50	1	
		RCP23CCPE723	Internet of Everything	3			25	15	15	15	60	100	3	
		RCP23CLPE723	Internet of Everything Laboratory			2	25				25	50	1	
4	MD	RCP23CCMD701	Vulnerability Assessment and Penetration Testing	2			25	15	15	15	60	100	2	
5	EL	RCP23ITELX08	Research Ecosystem		2		50					50	2	
6	EL	RCP23IPEL701	Project Stage-II			8	50				50	100	4	
7	HS	RCP23ICHSX09	Disaster Management and Preparedness	2									Audit	
8	SC	RCP23CCSC701	AI for Language Generation and Understanding	1			40					40	1	
			Total	14	2	14	315			60	365	740	21	

@ Any 1 Programme Elective Course

Prepared by:
Dr. J. S. Sonawane

Checked by:
Dr. S. M. Pardeshi

Prof. Dr. R. B. Wagh
BOS Chairman

Prof. S. P. Shukla
C.O.E.

Prof. Dr. P. J. Deore
Dean Academics & Dy. Director

Prof. Dr. J. B. Patil
Director

R. C. Patel Institute of Technology, Shirpur
SES
Autonomous

Final Year B.Tech Computer Engineering Semester-VII/VIII (w.e.f. 2026-27)														
Sr.No.	Course Category	Course Code	Course Title	Teaching Scheme			Evaluation Scheme					Total	Credits	
							Continuous Assessment (CA)							
				L	T	P	TA	Term Test 1 (TT1)	Term Test 2 (TT2)	Average of (TT1 & TT2)	ESE			
							[A]			[B]	[C]	[A+B+C]		
1@	PE	RCP23CCPE811	Quantum Computing*	3			25	15	15	15	60	100	3	3
		RCP23CCPE812	Human Computer Interaction*	3			25	15	15	15	60	100	3	
		RCP23CCPE813	Geographic Information System*	3			25	15	15	15	60	100	3	
			NPTEL/SWAYAM Course#	3			25	15	15	15	60	100	3	
2	EL	RCP23IPELX10	Internship (OJT) cum Project			36	200				200	400	18	
3	SC	RCP23CCSC801	Multi-Agent Systems	1			40					40	1	
			Total	4		36	265			15	260	540	22	

@ Any 1 Programme Elective Course

1. * Program Elective Courses offered by institute for the students doing Internship (OJT).
2. # Program Elective Courses offered by NPTEL/SWAYAM for the students doing Internship (OJT).

These courses are to be studied in self-study mode using NPTEL/SWAYAM platform.

Students doing Internship (OJT) shall submit certificates of NPTEL examination OR they have to **appear examinations conducted by institute like TT1, TT2 and ESE.**

Students undergoing Internship (OJT) have the option to appear for both the NPTEL examination and End Semester Examination (ESE) conducted by institute for respective course. In such cases, the better of the two scores (NPTEL or ESE) shall be considered for final grading.

List of NPTEL courses will be declared by concerned BOS at the beginning of semester.

Prepared by:
Dr. J. S. Sonawane

Checked by:
Dr. S. M. Pardeshi

Prof. Dr. R. B. Wagh
B.O.S Chairman

Prof. S. P. Shukla
C.O.E.

Prof. Dr. P. J. Deore
Dean Academics & Dy. Director

Prof. Dr. J. B. Patil
Director

R. C. Patel Institute of Technology, Shirpur
SFS
Autonomous

R. C. PATEL
INSTITUTE OF TECHNOLOGY
An Autonomous Institute

Department of Computer Engineering

Vision of the Institute

To become a leading Institute in Technical education fostering innovation, research, ethical values, and sustainable development for the betterment of society.

Mission of the Institute

To impart high quality Technical Education through:

M1. Innovative and Interactive learning process and high quality, globally recognized instructional programs.

M2. Fostering a collaborative scientific temper among students with ethical responsibility towards the society.

M3. Preparing students from diverse backgrounds to have aptitude for employment, entrepreneurship and research with a spirit of Professionalism.

M4. To contribute to nation's sustainable development.

Vision of the Department

To provide high quality Computer Engineering education with socio-moral values.

Mission of the Department

M1 To provide state-of-the-art ICT based teaching-learning process.

M2 To groom the students to become professionally sound computer engineers to meet emerging needs of industry and society.

M3 To make the students employable professionals by inculcating ethical values.

Program Educational Objectives (PEOs) of the Department

PEO1 Graduates will have technical proficiency, lifelong learning skills to become a professional, entrepreneur and leader.

PEO2 Graduates will function effectively in diverse cultural and professional environments, respecting societal perspectives and inclusive practices.

PEO3 To foster ethical and social values to be socially responsible human being.

R. C. PATEL
INSTITUTE OF TECHNOLOGY
An Autonomous Institute

Department of Computer Engineering

Program Specific Outcomes (PSOs)

PSO 1 Apply programming principles, algorithms, and data structures to design efficient software solutions and intelligent systems using structured, object-oriented, and emerging technologies**.**

PSO 2 Design, develop, and deploy responsive web and mobile applications integrated with databases and cloud platforms, leveraging modern frameworks and tools for digital transformation.

Semester - VII

Program: Computer Engineering	Final Year B.Tech.	Semester: VII
Natural Language Processing (RCP23CCPC701)		
Natural Language Processing Laboratory (RCP23CLPC701)		

Prerequisites:

1. Formal Languages and Automata Theory
2. Python Programming
3. Machine Learning

Course Objective(s):

1. To understand the theoretical foundations and basic principles of Natural Language Processing.
2. To apply preprocessing techniques, syntactic and semantic analysis to process and analyze text data.
3. To explore statistical and machine learning methods for solving NLP tasks such as classification, tagging, and translation.
4. To design and implement NLP applications using modern tools and frameworks with consideration of ethical issues.

Course Outcomes:

On completion of the course, the learner will be able to:

CO	Course Outcomes	Blooms Level	Blooms Description
CO1	Explain stages of NLP and apply morphological analysis techniques on any real world text.	L2	Understand
CO2	Apply appropriate techniques such as tokenization, POS tagging and syntactic parsing to real-world text.	L3	Apply
CO3	Apply feature extraction, word sense disambiguation and similarity for semantic analysis of text.	L3	Apply
CO4	Apply pragmatics and discourse segmentation to understand structure and context of text.	L3	Apply

Natural Language Processing (RCP23CCPC701) Course Contents

Unit-I 05 Hrs.

Introduction

Levels of NLP, Knowledge in Speech and Language Processing, Ambiguity in Natural Language, Challenges of NLP, Regular Expression, Finite Automata, Finite State Transducers (FST), Morphology Analysis: Survey of English Morphology, Inflectional Morphology & Derivational Morphology, Lemmatization, Morphological Parsing With FST, Lexicon Free FST Porter Stemmer, Word and Sentence Tokenization, Detection and Correction of Errors, Minimum Edit Distance with Backtracking.

Unit-II 06 Hrs.

Word Level Analysis

N – Grams, Unigrams/Bigrams Language Models, Corpora, Computing the Probability of Word Sequence, Training and Testing.

Perplexity And Entropy: Smoothing and Backup, Zipf's Law, Add One Smoothing

Unit-III 08 Hrs.

Syntax Analysis

Part-Of-Speech Tagging (POS) - Open and Closed Words. Tag Set for English (Penn Treebank), Rule Based POS Tagging, Transformation Based Tagging, Stochastic POS Tagging and Issues Multiple Tags & Words, Unknown Words. Hidden Markov Model (HMM), Maximum Entropy, And Conditional Random Field (CRF).

Context-Free Grammars, Normal Forms for grammar Parsing with CFG: Top-down parsing, Bottom-up parsing, Basic top-down parser,

Problems with the Basic Top-down Parser: Left Recursion, Ambiguity Earley parser, Probabilistic CFG, Probabilistic CYK

Unit-IV 08 Hrs.

Semantic Analysis

Lexical Semantics, Attachment for Fragment of English- Sentences, Noun Phrases, Verb Phrases, Prepositional Phrases, Relations Among Lexemes & Their Senses: Homonymy, Polysemy, Synonymy, Hyponymy, WordNet, Feature Extraction and Word Embeddings: Vector semantics,Traditional Word Embedding: TF-IDF

Static Word Embedding: Word2Vec - Continuous Bag of Words (CBOW), Skipgram, GloVe V

Sense Disambiguation: WSD using Indirect approaches: selectional restriction-based disambiguation, selectional association-based disambiguation, WSD using Direct approaches: Supervised, Unsupervised, Bootstrapping methods, Dictionary based: Lesk algorithm.

Unit-V 07 Hrs.

Pragmatics & Discourse Segmentation:

Introduction to Discourse, Cohesion and Coherence in Discourse, Discourse connectives, Reference Resolution, Reference Phenomena, Features for Pronominal Anaphora Resolution, Anaphora Resolution: Hobbs algorithm, Centering algorithm, Log-linear model, Co-reference resolution, Evaluation of Coreference resolution.

Unit-VI 08 Hrs.

NLP application for text classification:

NLP application pipeline, Evaluating Classification Model, Develop a text classification application,

Transformers:

Encoder-Decoder Framework, Attention Mechanisms, Transformer architecture, Challenges with Transformers, Transfer Learning in NLP.

Natural Language Processing Laboratory (RCP23CLPC701)

List of Laboratory Experiments

Suggested Experiments:(Any 10)

1. Study and Critical Analysis of a Research Paper on NLP Applications. To collect a text corpus from various sources and perform preprocessing tasks such as tokenization, stop-word removal, and cleaning.
2. To implement text normalization techniques including stemming, lemmatization, case folding, and lexical processing for improved text representation.
3. To construct and analyze N-Gram language models for predicting word sequences and estimating language probabilities.
4. To develop and evaluate a rule-based Part-of-Speech (POS) tagging system using linguistic rules and patterns.
5. To implement a statistical POS tagging system using Hidden Markov Models (HMM) and assess its tagging accuracy.
6. To design and apply Context-Free Grammar (CFG) rules for syntactic parsing and sent structure analysis.

7. To perform probabilistic parsing using statistical parsing techniques for determining the most likely parse tree of a sentence.

8. To conduct semantic analysis using WordNet for exploring lexical relationships such as synonymy, antonymy, and hypernymy.

9. To extract textual features and represent words and documents using vector semantics techniques.

10. To generate word embeddings and perform Word Sense Disambiguation (WSD) for understanding contextual word meanings.

11. To analyze discourse structures and implement coreference resolution for identifying entity references across text.

12. To develop and evaluate machine learning models for automated text classification tasks.

13. To implement transformer-based text classification using transfer learning techniques and evaluate model performance on benchmark datasets.

Any other practical covering the syllabus topics and subtopics can be conducted.

Oral examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.

Text Books:

1. Tanmoy Chakraborty, "Introduction to Large Language Models,Generative AI for Text", Wiley, 25th Dec 2024.

2. Denis Rothman, "Transformers for Natural Language Processing", Packt Publishing, 2nd, 2022.

3. Jurafsky and Martin, "Speech and Language Processing", 2nd Edition, Prentice Hall, January 26, 2000, ISBN: 0130950696.

Reference Books:

1. Jurafsky, D., & Martin, J. H., "Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition with Language Models (3rd ed., draft)", Self-published,2025

2. Manning and Schutze, "Statistical Natural Language Processing", MIT Press, 1st Edition, June 18, 1999, ISBN: 0262133601.

3. James Allen, "Natural Language Understanding", The Benajmins/Cummings Publishing Company Inc. 1994, ISBN 0-8053-0334-0.

4. Tom Mitchell, "Machine Learning", McGraw Hill, 1997, ISBN 0070428077.

5. Frederick Jelinek, "Statistical Methods for Speech Recognition", The MIT Press, 1998, ISBN 0-262-10066-5.

Web Resources:

1. Natural Language Processing, IIT Kharagpur
   https://nptel.ac.in/courses/106105158

2. Applied Natural Language Processing
   https://onlinecourses.nptel.ac.in/e-learning/preview/noc20$_{c}$s87

Program: Computer Engineering	Final Year B.Tech.	Semester: VII
Agentic AI (RCP23CCPE711)		
Agentic AI Laboratory (RCP23CLPE711)		

Prerequisites:

1. Artificial Intelligence
2. Machine Learning

Course Objective(s):

1. To understand the fundamental concepts, architectures, and lifecycle of Agentic AI systems.
2. To apply LLMs, agent frameworks, memory systems, and RAG techniques to develop intelligent agents.
3. To analyze and evaluate multi-agent systems, agent performance, deployment strategies, and real-world applications.

Course Outcomes:

On completion of the course, the learner will be able to:

CO	Course Outcomes	Blooms Level	Blooms Description
CO1	Explain the concepts, architectures, and operational characteristics of Agentic AI systems.	L2	Understand
CO2	Develop LLM-based agents using prompt engineering, agent frameworks, tool integration, and memory mechanisms.	L6	Create
CO3	Analyze memory retrieval, RAG techniques and multi-agent collaboration for autonomous problem solving.	L4	Analyze
CO4	Examine the performance, limitations and application scenarios of Agentic AI systems.	L3	Apply

Agentic AI (RCP23CCPE711)
Course Contents

Unit-I — 05 Hrs.

Evolution of Agentic AI: Evolution from traditional AI and Large Language Models (LLMs) to Agentic AI, Intelligent agents and the Perception–reasoning–action cycle

Agent architectures and components, Agent lifecycle, capabilities, and levels of autonomy, Types of Agents Reactive, deliberative, hybrid, task oriented and goal-oriented agents,

Agent design principles and task decomposition, ReAct Framework, fundamentals of multi-agent systems, Comparison of rule-based systems and agent-based systems Real-world applications of Agentic AI.

Unit-II — 09 Hrs.

Large Language Models and Agent Development: Introduction to Large Language Models (LLMs), role of LLMs in Agentic AI, agents versus chains, prompting techniques - zero-shot, few-shot, chain-of thought (CoT), and role prompting, structured prompting for controlling agent behavior and outputs, tool integration and function calling

Agent development frameworks: LangChain, LangGraph, CrewAI, AutoGen and Hugging Face Agents, Development of simple LLM-based agents.

Unit-III — 09 Hrs.

Memory, Retrieval and RAG Systems for AI Agents: Agent Memory Architectures: Short-term, long-term, episodic, and semantic memory, Embeddings and Vector Representations, Text chunking and similarity techniques

Vector Stores and Knowledge Retrieval: Storage and retrieval of contextual knowledge, Concepts of semantic search

Retrieval-Augmented Generation (RAG): RAG architecture and retrieval augmented memory loops, Agentic RAG, Query transformation, hybrid search, and re-ranking; Context compression and context engineering strategies, Evaluation of retrieval and RAG performance.

Unit-IV — 08 Hrs.

Multi-Agent Systems and Agent Collaboration: Fundamentals of multi-agent systems, agent collaboration and coordination, communication between agents, agent roles and responsibilities; distributed problem solving and task allocation, cooperative and competitive agent behaviors.

Multi-Agent Architectures: Planner–executor and supervisor–worker architectures, multi-agent workflows

Unit-V 06 Hrs.

Agent Evaluation and Deployment:

Agent Evaluation Metrics: Success metrics, autonomy, correctness, reliability, and robustness, self-reflection, critique, and iterative improvement, benchmarking and performance assessment of AI agents, Human-in-the-Loop (HITL) systems and approval workflows

Agent Deployment Fundamentals: Basics of agent deployment and execution environments; monitoring and maintenance of agent systems; limitations, challenges, and failure modes of autonomous agents.

Unit-VI 05 Hrs.

Applications and Future Trends of Agentic AI:

Applications of Agentic AI in healthcare, finance, education, manufacturing, and smart cities; case studies of AI assistants, research agents, coding agents, and enterprise AI agents; emerging trends in multimodal agents, autonomous workflows.

Agentic AI Laboratory (RCP23CLPE711)

List of Laboratory Experiments

Suggested Experiments: (Any 10)

1. **Setup and Configuration of Agentic AI Development Environment** Set up a Python-based development environment for building LLM agents, including API key configuration, virtual environment management, and a first call to an LLM endpoint to verify connectivity.
   Tools: Python 3.x, venv/conda, OpenAI/Anthropic API or Ollama (local), VS Code, Jupyter Notebook.

2. **Design of Reactive and Goal-Oriented Agents** Implement and compare a simple reactive agent (stimulus–response) with a goal oriented agent that selects actions to achieve a defined objective in a small environment.
   Tools: Python, OpenAI/Anthropic API, simple grid/task environment.

3. **Implementation of ReAct-based Agent** Build an agent using the ReAct (Reasoning + Acting) paradigm that interleaves chain of-thought reasoning with tool actions to solve a multi-step query.
   Tools: Python, LangChain, an LLM API, a search/calculator tool.

4. **Prompt Engineering Techniques for Agents** Apply and evaluate zero-shot, few-shot, chain-of-thought, and role-based prompting strategies, observing their effect on agent reasoning quality and output reliability.
   Tools: Python, LLM API, prompt templates.

5. **Tool Calling and Function Integration** Expose custom Python functions (e.g., calculator, weather, database lookup) as callable tools and enable the LLM to invoke them via structured function calling.

   Tools: Python, OpenAI/Anthropic function-calling API, custom tool functions.

6. **Development of a LangChain-based Agent** Construct an agent using LangChain components — chains, prompts, and a tool equipped agent executor — to autonomously answer queries requiring external tools.

   Tools: Python, LangChain, LLM API, LangChain tools (Search, Python REPL).

7. **Development of a Simple LangGraph Workflow** Design a stateful agent workflow as a graph with nodes and conditional edges using LangGraph, demonstrating branching and looping control flow.

   Tools: Python, LangGraph, LangChain, LLM API.

8. **Implementation of Agent Memory using Conversation History** Add short-term and summary-based memory to a conversational agent so it retains context across turns, and observe behaviour with and without memory.

   Tools: Python, LangChain memory modules, LLM API.

9. **Embeddings and Semantic Search using Vector Stores** Generate text embeddings, store them in a vector database, and perform semantic similarity search over a document corpus.

   Tools: Python, embedding model (OpenAI/Sentence-Transformers), FAISS/ Chroma/ Pinecone.

10. **Development of a Basic RAG System** Build a Retrieval-Augmented Generation pipeline that retrieves relevant chunks from a knowledge base and grounds the LLM's response in the retrieved context.

    Tools: Python, LangChain, vector store (Chroma/FAISS), LLM API.

11. **Query Expansion and Context Optimization in RAG** Improve a baseline RAG system using query expansion/rewriting, re-ranking, and chunking strategies, and compare answer relevance before and after optimization.

    Tools: Python, LangChain, re-ranker (e.g., Cohere/BGE), vector store, LLM API.

12. **Development of a Two-Agent Collaborative System using CrewAI** Create two role-specialized agents (e.g., researcher and writer) that collaborate on a shared task with defined roles, goals, and task delegation.

    Tools: Python, CrewAI, LLM API, web search tool.

13. **Implementation of Planner–Executor Multi-Agent Workflow** Design a multi-agent system where a planner agent decomposes a complex task into sub-tasks and an executor agent

carries them out, coordinating through a shared state.

Tools: Python, LangGraph/CrewAI/AutoGen, LLM API.

14. **Evaluation of Agent Performance** Evaluate agent and RAG outputs using metrics such as task success rate, faithfulness, relevance, latency, and token cost, using an automated evaluation framework.

    Tools: Python, RAGAS/DeepEval/LangSmith, LLM API.

Any other practical covering the syllabus topics and subtopics can be conducted.

Oral examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.

Text Books:

1. Michael Lanham, "GPT Agents in Action", Manning Publications, 1st Edition, 2024.
2. Roberto Infante, "AI Agents and Applications", Manning Publications, 2026.
3. Elliot R. Stroud, "Agentic AI with LangChain and LangGraph", Independently Published, 2025.

Reference Books:

1. Srinivasan Venkataraman and Harrison Chase, "Building Agentic AI Systems: A Developer's Guide to LLM Agents, LangChain and LangGraph", O'Reilly Media, 2024.
2. Sinan Ozdemir, "LLM Engineers' Handbook: Retrieval-Augmented Generation, Prompting, Agents, and Production AI", Packt Publishing, 2024.
3. Jerry Liu, "AI Engineering: Building Applications with Foundation Models", O'Reilly Media, 2024.
4. Tom Taulli and Gaurav Deshmukh, "Building Generative AI Agents: Using LangGraph, AutoGen, and CrewAI", Apress, 2025.
5. Olivier Blais, "Practical GenAI: Building Real-World Generative AI Products with LLMs, RAG, Agents, and Workflows", O'Reilly Media, 2024.
6. Stuart Russell and Peter Norvig, "Artificial Intelligence: A Modern Approach", Pearson Education, 4th Edition, 2021.

Web Resources:

1. Applications of GenAI and Agentic AI in Business
   https://elearn.nptel.ac.in/product-page/applications-of-genai-and-agentic-ai-in-business
2. https://www.youtube.com/watch?v=nOJrNwhokw4

Program: Computer Engineering	Final Year B.Tech.	Semester: VII
Distributed Computing (RCP23CCPE712)		
Distributed Computing Laboratory (RCP23CLPE712)		

Prerequisites:

1. Computer Networks
2. Operating Systems
3. Database Management Systems

Course Objective(s):

1. To provide students with contemporary knowledge in distributed systems.
2. To equip students with skills to analyze and design distributed applications.
3. To provide master skills to measure the performance of distributed synchronization algorithms.

Course Outcomes:

On completion of the course, the learner will be able to:

CO	Course Outcomes	Blooms Level	Blooms Description
CO1	Describe the fundamental concepts, architectures, elements, and middleware technologies used in distributed systems, including RPC, RMI, and object-based middleware.	L2	Understand
CO2	Analyze synchronization techniques in distributed systems, including clock synchronization, mutual exclusion, and related synchronization algorithms.	L4	Analyze
CO3	Explain resource management, process management, consistency models, and replication management in distributed computing environments.	L2	Understand
CO4	Apply distributed file system concepts to implement file-sharing mechanisms used in systems such as NFS and AFS for large-scale distributed applications.	L3	Apply

Distributed Computing (RCP23CCPE712)
Course Contents

Unit I — 04 Hrs.

Introduction to Distributed Systems:

Characterization of Distributed Systems: Issues, Goals, and Types of distributed systems, Distributed System Models, Hardware concepts, Software Concept. Middleware: Models of Middleware, Services offered by middleware, Client Server model.

Unit II — 08 Hrs.

Communication:

Layered Protocols, Interprocess communication (IPC): MPI, Remote Procedure Call (RPC), Remote Object Invocation, Remote Method Invocation (RMI) Message Oriented Communication, Stream Oriented Communication, Group Communication.

Unit III — 08 Hrs.

Synchronization:

Clock Synchronization, Logical Clocks, Election Algorithms, Mutual Exclusion, Distributed Mutual Exclusion-Classification of mutual Exclusion Algorithm, Requirements of Mutual Exclusion Algorithms, Performance measure.

Non Token based Algorithms: Lamport Algorithm, Ricart–Agrawala‘s Algorithm, Maekawa‘s Algorithm.

Token Based Algorithms: Suzuki-Kasami‘s Broardcast Algorithms, Singhal‘s Heurastic Algorithm, Raymond‘s Tree based Algorithm, Comparative Performance Analysis.

Unit IV — 08 Hrs.

Resource and Process Management:

Desirable Features of global Scheduling algorithm, Task assignment approach, Load balancing approach, load sharing approach.

Introduction to process management, process migration, Threads, Virtualization, Clients, Servers, Code Migration.

Unit V — 07 Hrs.

Consistency, Replication and Fault Tolerance:

Introduction to replication and consistency, Data-Centric and Client-Centric Consistency Models, Replica Management.

Fault Tolerance: Introduction, Process resilience, Reliable client-server and group communicat

Recovery.

Unit VI 07 Hrs.

Distributed File Systems and Name Services:

Introduction and features of DFS, File models, File Accessing models, File Caching Schemes, File Replication

Case Study: Distributed File Systems (DSF), Network File System (NFS), Andrew File System (AFS).

Trends in Distributed Computing: Edge Computing, Cloud Computing, Fog Computing.

Distributed Computing Laboratory(RCP23CLPE712)

List of Laboratory Experiments

Suggested Experiments:(Any 10)

1. To implement Client/server programming using RPC/RMI.
2. To implement a multi-thread application.
3. To implement Inter-process communication.
4. To implement Group Communication.
5. To implement Load Balancing Algorithm.
6. To implement Name Resolution protocol.
7. To implement the Election Algorithm.
8. To implement Clock Synchronization algorithms.
9. To implement Mutual Exclusion Algorithm.
10. To implement Deadlock management in Distributed systems.
11. To implement Distributed File System.
12. To implement a Case study on CORBA.

Any other practical covering the syllabus topics and subtopics can be conducted.

Oral examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.

Text Books:

1. Maarten van Steen and Andrew S. Tanenbaum, "Distributed Systems", 4^{th} Edition, Self-Published, 2023, ISBN: 978-90-815406-4-3.

2. George Coulouris, Jean Dollimore, Tim Kindberg, "Distributed Systems: Concepts and Design", 5^{th} Edition, Pearson Education, 2014.

Reference Books:

1. Shraddha Zanzat, Bhavana Karmore, Vishwajit Barbudhe, "Distributed Systems", Notion Press, 2020.

2. M. L. Liu, "Distributed Computing Principles and Applications", Pearson Addison Wesley, 2004.

Web Resources:

1. Distributed Systems
   https://onlinecourses.nptel.ac.in/e-learning/preview/noc21_cs87

2. Cloud Computing and Distributed Systems
   https://onlinecourses.nptel.ac.in/e-learning/preview/noc26_cs29

Program: Computer Engineering	Final Year B.Tech.	Semester: VII
Metaverse(RCP23CCPE713)		
Metaverse Laboratory(RCP23CLPE713)		

Prerequisites:

1. Computer Graphics
2. Computer Vision
3. Artificial Intelligence

Course Objective(s):

1. To introduce the Metaverse, its architecture, and enabling immersive technologies (VR/AR/MR).
2. To impart working knowledge of Virtual Reality and Blockchain/Web3 as applied to the Metaverse.
3. To develop multi-user, AI-augmented metaverse applications using open-source tools.
4. To examine security, privacy, ethical, and governance issues in metaverse deployments.
5. To build Digital Twins and Industrial Metaverse solutions and analyse their cyber-physical security requirements.

Course Outcomes:

On completion of the course, the learner will be able to:

CO	Course Outcomes	Blooms Level	Blooms Description
CO1	Explain the Metaverse architecture, ecosystem, and the role of VR/AR/MR.	L2	Understand
CO2	Apply VR, spatial computing, and Blockchain/Web3 concepts to immersive worlds.	L3	Apply
CO3	Design and develop multi-user, AI-driven metaverse applications using open-source tools.	L6	Create
CO4	Evaluate security, privacy, ethical, and governance challenges across consumer and industrial metaverse deployments	L5	Evaluate
CO5	Design Digital Twin and Industrial Metaverse solutions and assess their cyber-physical security requirements	L6	Create

Metaverse (RCP23CCPE713)
Course Contents

Unit-I — 06 Hrs.

Introduction to the Metaverse

Definition, evolution from Web 1.0 → 3.0 and the Spatial Web; characteristics — persistence, interoperability, embodiment, real-time interaction, creator economy. Seven-layer Metaverse architecture (Infrastructure, Human Interface, Decentralization, Spatial Computing, Creator Economy, Discovery, Experience). Open vs walled-garden metaverses; consumer vs Industrial Metaverse. Reference platforms: Decentraland, Roblox, Horizon Worlds, NVIDIA Omniverse. Standards: Metaverse Standards Forum, OMA3, Khronos.

Unit-II — 07 Hrs.

Virtual Reality and Spatial Computing for the Metaverse

VR concepts: immersion, presence, simulator sickness; VR hardware — HMDs, controllers, haptics, eye/hand tracking; 3DoF vs 6DoF; tracking systems (inside out, outside-in).

AR/MR and the XR continuum: OpenXR and WebXR. Stereoscopic, foveated, and asynchronous time-warp rendering; motion-to-photon latency.

Spatial computing: world anchoring, persistent maps, SLAM, plane detection, occlusion (extending Computer Vision). Spatial audio (HRTF, ambisonics).

Interaction techniques: gaze, gesture, voice.

Unit-III — 06 Hrs.

Avatars and Multi-User Virtual Worlds

Avatar types: stylized, photorealistic, codec avatars.

Vision-driven avatar capture: facial landmarks, blendshapes, pose and hand tracking, motion retargeting. Inverse kinematics, lip-sync, gaze, emotion transfer. Avatar interoperability: VRM, Ready Player Me, glTF, USD avatar schemas.

Multi-user system architecture: client–server, P2P, hybrid; authoritative servers, dead reckoning, interest management, state synchronization. Edge/5G and cloud rendering. Social presence, embodiment, accessibility, harassment-resistant design.

Unit-IV — 07 Hrs.

AI and Generative Content in the Metaverse

Role of AI in scaling content and population.

LLM-driven NPCs and intelligent agents: agentic loops, memory, goal-directed behaviour. Multimodal interaction: ASR, TTS, gesture/gaze recognition, emotion estimation.

Generative 3D: text-to-image, text-to-3D, NeRF, Gaussian splatting, procedural world generation. Animation synthesis and real-time translation for avatars. AI-based content moderation. Latency, cost, and grounding challenges of LLMs in real-time worlds.

Unit-V 08 Hrs.

Blockchain, Web3, Privacy, Ethics and Governance

Blockchain concepts: decentralization, consensus, smart contracts, EVM-compatible chains. Tokens and NFTs (ERC-20/721/1155) for avatars, land and items; crypto wallets and Web3 authentication. Decentralized storage (IPFS), Decentralized Identity (DID) and DAOs for governance.

Security and privacy threats: avatar impersonation, deepfakes, virtual harassment; privacy of immersive telemetry (gaze, gait, room geometry).

Regulations: GDPR, India's DPDP Act, child-safety norms. Governance models and wellbeing concerns. Future trends: BCI, 6G, AR cloud.

Unit-VI 08 Hrs.

Digital Twins, the Industrial Metaverse and Cyber-Physical Security

Digital Twins: digital model vs shadow vs twin; five-dimension twin model; real-time synchronization and predictive maintenance. Enabling stack — IoT sensing, MQTT/OPC-UA pipelines, simulation engines (NVIDIA Omniverse, Azure Digital Twins).

Standards: ISO 23247, DTDL, Asset Administration Shell (AAS).

Industrial Metaverse: distinction from the consumer metaverse; cyber-physical systems and the Industry 4.0/5.0 link; use cases — smart factory, smart-city, energy-grid and healthcare twins.

Cyber-Physical and OT Security: IT/OT convergence; Purdue model, IEC 62443, network segmentation and zero-trust; twin-specific threats (false-data injection, sensor spoofing, replay, twin–physical desynchronization); securing pipelines (MQTT/OPC-UA over TLS); the safety-security interplay.

Metaverse Laboratory (RCP23CLPE713)

List of Laboratory Experiments

Suggested Experiments:(Any 10)

1. Build a basic VR scene in the browser with primitive shapes, lighting, and 6DoF navigation.
2. Add spatial (3D positional) audio to a Three.js / A-Frame scene — attach a sound source to a 3D object and demonstrate distance attenuation and HRTF-based directionality as the user moves.
3. Create a markerless AR app that places a 3D model on a detected surface using the webcam.
4. Model a 3D object in Blender, export as glTF, and load it into a Three.js / A-Frame scen

5. Implement hand-tracking / gesture interaction in a WebXR scene so the user can grab, move, and release a virtual object using tracked hands or controllers.
6. Create a 3D avatar using Ready Player Me / VRoid and import it into a web scene with idle animation.
7. Drive an avatar's face / pose using webcam landmarks.
8. Build a multi-user shared room where two clients see each other's avatars move in real time.
9. Add a chatbot/LLM-driven Non-Player Characters (NPC) to a 3D scene that answers user questions via text.
10. Write, deploy, and test a simple NFT smart contract for a metaverse asset on a test network.
11. Connect a Web3 wallet (MetaMask) to a 3D web scene to authenticate a user and display owned NFTs.
12. Generate a 3D asset or PBR texture using an open-source text-to-image / text-to-3D tool, then import and place it in a web scene — illustrating AI-driven content creation.
13. Build a Digital Twin demo — simulate sensor data via MQTT and visualize it on a 3D model in the browser.
14. Secure the Digital Twin pipeline — enable TLS and token-based authentication on the MQTT/OPC-UA channel, then inject and detect a false-data-injection (sensor-spoofing) attack on the twin.

Oral examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.

Text Books:

1. Anjan Chatterjee, "Building the Metaverse: A Comprehensive Guide for Architects, Engineers, and Designers", CRC Press / Routledge, 2024.
2. Ali Darejeh, Samad Sepasgozar, "Metaverse, Generative AI, and Brain–Computer InterfacesShaping the Future of the Digital Landscape", Routledge, 2026.

Reference Books:

1. Anand Nayyar, Akshi Kumar (Eds.), "Augmented and Virtual Reality", TMH / McGraw Hill, 2023.
2. Pethuru Raj, Kavita Saini, Chellammal Surianarayanan, "Blockchain Technology and Applications", TMH / McGraw Hill, 2022.
3. Y. Wang, Z. Su et al., "Metaverse Security and Privacy", Springer, 2024.

4. Steve Aukstakalnis, “Practical Augmented Reality: A Guide to the Technologies, Applications, and Human Factors for AR and VR”, Pearson / Addison-Wesley, 2017.

5. Cathy Hackl, Dirk Lueth, Tommaso Di Bartolo, “Navigating the Metaverse: A Guide to Limitless Possibilities in a Web 3.0 World”, Wiley, 2022.

6. QuHarrison Terry, Scott “DJ Skee” Keeney, “The Metaverse Handbook: Innovating for the Internet’s Next Tomorrow”, Wiley, 2022.

7. Kumar Saurabh, Ashutosh Saxena, “Blockchain Technology: Concepts and Applications”, Wiley India, 2022.

Web Resources:

1. Foundation for Virtual and Augmented Reality Systems, IIT Guwahati
https://nptel.ac.in/courses/106103842

2. Mobile Virtual Reality and Artificial Intelligence
https://onlinecourses.nptel.ac.in/e-learning/preview/noc25_cs80

Program: Computer Engineering	Final Year B.Tech.	Semester: VII
Social Network Analysis (RCP23CCPE721)		
Social Network Analysis Laboratory (RCP23CLPE721)		

Prerequisites:

1. Discrete Graph Theory
2. Computer Networks

Course Objective(s):

1. To introduce the fundamentals of Social Network Analysis (SNA), graph representations, and network measures.
2. To study network growth models, link analysis, and community structure in real-world networks.
3. To apply link prediction, cascade behaviours, and anomaly detection techniques on social networks.
4. To explore graph representation learning and real-world applications of SNA.

Course Outcomes:

On completion of the course, the learner will be able to:

CO	Course Outcomes	Blooms Level	Blooms Description
CO1	Explain the foundations, levels, and measures of Social Network Analysis.	L2	Understand
CO2	Apply network growth models and link-analysis algorithms (PageRank, SimRank, etc.) to real world networks.	L3	Apply
CO3	Analyze community structure, link prediction, and cascade behaviours in static and dynamic networks.	L4	Analyze
CO4	Design SNA-based solutions using graph representation learning for applications such as recommender systems, anomaly detection, and OSN security.	L6	Create

Social Network Analysis (RCP23CCPE721)
Course Contents

Unit-I 07 Hrs.

Foundations of Social Network Analysis and Network Measures

Networks and Society: What is Social Network Analysis; why we study social networks; applications of SNA; preliminaries; three levels of SNA (micro, meso, macro); historical development; graph visualisation tools (Gephi, Cytoscape, NetworkX, igraph).

Network Measures: Network basics — nodes, edges, directed/undirected, weighted graphs, adjacency representations; node centrality (degree, closeness, betweenness, eigenvector, Katz, PageRank-based); assortativity; transitivity and reciprocity; similarity measures; degeneracy and k-core decomposition.

Unit-II 07 Hrs.

Network Growth Models:

Properties of real-world networks — small-world, scale-free, heavy-tailed degree distributions, clustering. Random network model (Erdős–Rényi). Ring lattice network model. Watts–Strogatz small-world model. Preferential attachment model (Barabási–Albert). Price's model for citation networks. Local-world network growth model. Network model with accelerating growth. Aging in preferential attachment.

Unit-III 07 Hrs.

Link Analysis:

Applications of link analysis. Signed networks; structural balance theory. Strong and weak ties (Granovetter). Link analysis algorithms — overview. PageRank — formulation, random surfer model, iterative computation, dangling nodes, teleportation. Personalised PageRank. DivRank — diversity-aware ranking. SimRank — structural similarity between nodes. PathSIM — meta-path-based similarity in heterogeneous networks. Comparative analysis of ranking algorithms.

Unit-IV 08 Hrs.

Community Structure and Link Prediction:

Community Structure in Networks: Applications of community detection; types of communities (disjoint, overlapping, local); community detection methods — modularity-based (Girvan–Newman, Louvain), spectral, label propagation; disjoint vs overlapping vs local community detection; community detection vs community search; evaluation metrics (modularity, NMI, F1, ARI).

Link Prediction: Applications; temporal changes in a network; problem definition and evalua
(AUC, precision-at-k); heuristic models — common neighbours, Jaccard, Adamic–Adar, Katz, pr

ential attachment; probabilistic models; supervised random walk; information-theoretic model; latest trends (GNN-based link prediction).

Unit-V 06 Hrs.

Cascade Behaviours, Network Effects and Anomaly Detection

Cascade Behaviours: Preliminaries and key terminology — diffusion, threshold, contagion. Cascade models — Linear Threshold and Independent Cascade. Probabilistic cascades. Epidemic models — SI, SIS, SIR, SEIR. Cascade prediction. Case study: the Indignados movement.

Anomaly Detection in Networks: Outliers vs network-based anomalies; challenges; anomaly detection in static networks (structural, attribute-based); anomaly detection in dynamic networks (event-, change-, community-based).

Unit-VI 07 Hrs.

Graph Representation Learning and Applications

Graph Representation Learning: Machine learning pipelines for graph data; intuition and benefits of representation learning; criteria for good graph embeddings; graph representation learning pipeline; methods — DeepWalk, Node2Vec, LINE, GraphSAGE, GCN, GAT (introductory).

Applications and Case Studies: Malicious activities on OSNs; sockpuppets in OSNs; collusion on online social networks; modelling the spread of COVID-19 using SNA; recommender systems based on SNA.

Social Network Analysis Laboratory (RCP23CLPE721)

List of Laboratory Experiments

Suggested Experiments:(Any 10)

1. **Network construction and visualization:** load a real-world network (e.g., Zachary's Karate Club / Facebook ego-net), compute basic statistics (nodes, edges, density, degree distribution), and visualize it.
2. **Node centrality analysis:** compute and compare degree, closeness, betweenness, eigenvector, and PageRank centralities; identify top-k influential nodes.
3. **Network properties:** compute clustering coefficient, transitivity, reciprocity, assortativity, and k-core decomposition for the chosen network.
4. **Network growth models:** generate Erdős–Rényi, Watts–Strogatz, and Barabási–Albert networks; compare degree distributions, clustering, and average path length with a real network.
5. **Link analysis algorithms:** implement / apply PageRank, Personalised PageRank, and [illegible] Rank on a chosen graph; rank top-k nodes and compare results.

6. **Community detection:** apply Girvan–Newman, Louvain, and Label Propagation; visualize and evaluate communities using modularity and NMI.

7. **Link prediction:** evaluate Common Neighbours, Jaccard, Adamic–Adar, and Preferential Attachment heuristics on a temporal split of the dataset; report AUC and precision-at-k.

8. **Epidemic spread modelling:** simulate SI / SIR / SEIR models on a real network and analyze how network structure affects spread (mini-version of the COVID-19 case study).

9. **Anomaly detection:** detect anomalous nodes/edges in a static network using structural features (degree, ego-net density, OddBall-style); visualize outliers.

10. **Graph representation learning:** generate node embeddings using Node2Vec / DeepWalk; perform node classification or link prediction; visualize embeddings with t-SNE.

11. **Influence maximization and cascade modelling:** implement the Linear Threshold and Independent Cascade models on a real network, then apply greedy seed selection to find the top-k most influential nodes and compare the resulting spread.

12. **Signed-network and structural-balance analysis:** load a signed network (e.g., Epinions / Slashdot), classify triads as balanced or unbalanced, measure the degree of structural balance, and relate findings to strong/weak ties.

13. **SNA-based recommender system:** build a friend- or item-recommender using network structure (common-neighbour / Adamic–Adar scoring or Node2Vec embeddings), then evaluate top-k recommendations against a held-out set.

14. **GNN-based link prediction / OSN security:** train a GCN or GraphSAGE model for link prediction (or sockpuppet/malicious-node classification) and compare its AUC against the heuristic baselines from Experiment 7.

Any other practical covering the syllabus topics and subtopics can be conducted.

Oral examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.

Text Books:

1. Tanmoy Chakraborty, "Social Network Analysis", Wiley, 2024.

2. Maksim Tsvetovat and Alexander Kouznetsov, "Social Network Analysis for Startups", O'Reilly, 2011.

Reference Books:

1. Niyati Aggrawal and Adarsh Anand, "Social Networks Modelling and Analysis", CRC Press, 2022.
2. Stanley Wasserman, Katherine Faust, "Social Network Analysis: Methods and Applications", Cambridge University Press, 1994.
3. Stephen P Borgatti, Martin G Everett and Jeffrey C Johnson, "Analyzing Social Networks", SAGE, 2013.
4. Mehmet Kaya, Reda Alhajj, "Influence and Behavior Analysis in Social Networks and Social Media", Springer, 2019.
5. Krishna Raj P.M., Ankith Mohan, K.G. Srinivasa, "Practical Social Network Analysis with Python, Springer, 2018.
6. Charu C. Aggrwal, "Social Network Data Analytics", Springer, 2011.

Web Resources:

1. Social Network Analysis
   https://onlinecourses.nptel.ac.in/e-learning/preview/noc22_cs117
2. Social Networks, IIT Ropar
   https://nptel.ac.in/courses/106106169

Program: Computer Engineering	Final Year B.Tech.	Semester: VII
Virtual Infrastructure Systems (RCP23CCPE722)		
Virtual Infrastructure Systems Laboratory (RCP23CLPE722)		

Prerequisites:

1. Operating Systems
2. Computer Networks
3. Database Management Systems
4. Basic Cloud Computing concepts

Course Objective(s):

1. To understand the architecture, components, and operation of virtualized IT infrastructure.
2. To learn virtualization methods for compute, storage, and networking.
3. To manage virtual machines and data center operations using industry tools.
4. To explore performance, security, and migration in virtual environments.
5. To understand how virtualization underpins modern cloud computing systems.

Course Outcomes:

On completion of the course, the learner will be able to:

CO	Course Outcomes	Blooms Level	Blooms Description
CO1	Explain the concepts and architecture of virtualization and cloud computing.	L2	Understand
CO2	Configure virtual machines, networks, and storage resources using virtualization tools.	L3	Apply
CO3	Analyze performance, scalability, and security aspects of virtualized systems.	L4	Analyze
CO4	Implement integrated virtualization and cloud infrastructure solutions	L3	Apply

Virtual Infrastructure Systems (RCP23CCPE722) Course Contents

Unit-I 06 Hrs.

Virtualized IT Infrastructure – Concepts and Cloud Foundations

Physical vs. Virtual IT Infrastructure, Virtualization in Data Centers – Hardware, Storage, Memory, and I/O virtualization, Types of Virtualization – Server, Desktop, Application, Network, Need for Virtualization; Benefits and Challenges, Cloud Computing Fundamentals – Definition, Characteristics, Service and Deployment Models (IaaS, PaaS, SaaS; Public, Private, Hybrid, Community Clouds), Relationship between Virtualization and Cloud Computing, Recent Trends & Emerging Technologies in Virtualized Environments.

Unit-II 07 Hrs.

Virtual Machine Management

Hypervisors – Types and Architecture, Role of VMM (Virtual Machine Monitor), VM Lifecycle – Configuration, Provisioning, Scaling, and Deletion, VM Scheduling and Load Balancing, Migration – Live, Cold, and Storage Migration; Migration Process and Tools, Software and Workload Migration (P2V, V2V), QoS Parameters – Performance, Functionality, and Reliability, VM Hosting and Comparison (Windows vs. Linux).

Unit-III 07 Hrs.

Hardware and Storage Virtualization

Abstracting Hardware – Concept, Need, and Solutions, CPU Virtualization Approaches & Algorithms, Memory Virtualization – Techniques, Ballooning, Shadow Paging, Storage Virtualization – Methods, Storage Aggregation, and Software-Defined Storage, I/O Virtualization Techniques (SR-IOV, PCI Passthrough), Role of Virtualization in Data Center Efficiency.

Unit-IV 07 Hrs.

Network Virtualization

Virtual Interfaces, Virtual Switches and Routers, VNIC Profiles, VLANs, and Virtual Network Design, Network Modes – Bridged, NAT, Host-Only, Data Flow among VMs; TUN/TAP Drivers; Host-Only vs. Bridged Networking, Data Center Network Virtualization using ESXi, Software-Defined Networking (SDN) Overview, Optimizing Resource Utilization & Network Security in Virtual Environments

Unit-V 07 Hrs.

Server Virtualization and Cloud Integration

Server Partitioning and Virtual Server Hosts, Deploying and Managing Virtual Servers (vSphere, Hyper-V, KVM, Xen), Remote Management Tools and Automation (vCenter, CLI, Web Console), Server Clustering and Network Load Balancing (NLB), Performance and Health Monitoring – vSphere Monitoring & Tools, Integration of Virtualization with Cloud Services (IaaS Implementation), Introduction to Containers and Orchestration (Docker, Kubernetes)

Unit-VI 08 Hrs.

Virtualization Performance, Security, and Emerging Trends

Performance Issues – Resource Overcommitment, VM Sprawl, and Bottlenecks, Virtualization Security Risks – Hypervisor Vulnerabilities, VM and Migration Attacks, Backup, Fault Tolerance, and Disaster Recovery in Virtualized Systems, Security Hardening and Monitoring Strategies, Security Standards and Compliance for Virtualization – CIS Benchmarks, NIST SP 800-125/125A, ISO/IEC 27017, PCI-DSS virtualization guidance; Audit Mechanisms – hypervisor/vCenter audit logging, configuration-compliance scanning, baseline validation, SIEM integration and continuous compliance monitoring; Backup, Fault Tolerance, and Disaster Recovery in Virtualized SystemsData Center Automation and Orchestration Tools (OpenStack, vCloud Suite), Emerging Trends – Edge Computing, Serverless Architecture, Green IT.

Virtual Infrastructure Systems Laboratory (RCP23CLPE722)

List of Laboratory Experiments

Suggested Experiments: (Any 10)

1. **Installation of a Hypervisor** – Install and set up Type-1 and Type-2 hypervisors (e.g., VMware ESXi, VirtualBox).
2. **Create and Manage Virtual Machines**– Configure CPU, memory, and storage parameters; start, stop, and snapshot VMs.
3. **Virtual Networking Setup** – Design NAT, Bridged, and Host-Only networks between multiple VMs.
4. **Storage Virtualization** – Configure shared storage using iSCSI or NFS and attach to virtual hosts.
5. **Virtual Machine Migration** – Perform live and cold migrations of VMs between hosts.
6. **Resource Allocation and Scheduling** – Experiment with CPU and memory resource limits, shares, and reservations.

7. **Server Virtualization and Monitoring** – Deploy and manage virtual servers using VMware vCenter or Microsoft Hyper-V Manager.

8. **Network Load Balancing**– Configure and test load balancing between multiple virtual servers.

9. **Security in Virtual Environments** – Demonstrate access controls, snapshots for recovery, and identify hypervisor vulnerabilities.

10. **Cloud Integration** – Deploy a simple IaaS environment using OpenStack or AWS EC2, linking virtualization to cloud concepts.

11. **Containerization and Orchestration** – Build and run application containers using Docker (create images, manage volumes and networks), then deploy and scale a multi-container application on a Kubernetes cluster (minikube/kind); demonstrate pods, services, deployments, and auto-scaling, and compare the resource footprint of containers vs. virtual machines.

12. **Fault Tolerance and Disaster Recovery** – Configure High Availability (HA) and Fault Tolerance for virtual machines, schedule automated backups/snapshots, and simulate a host failure to demonstrate VM failover, recovery, and rollback using tools such as VMware HA/FT or KVM with Pacemaker/Corosync.

Any other practical covering the syllabus topics and subtopics can be conducted.

Oral examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.

Text Books:

1. Matthew Portnoy – "Virtualization Essentials", Sybex/Wiley, 3rd Edition, 2023.

2. Thomas Erl, "Cloud Computing: Concepts, Technology & Architecture", 2nd Edition, Pearson, 2023.

3. John A. Davis, Steve Baca, Owen Thomas, "VCP-DCV for vSphere 8.x Cert Guide", Pearson/InformIT, 5th Edition", 2023.

Reference Books:

1. Elver Sena Sosa, "Mastering VMware NSX® for vSphere®", John Wiley & Sons, Inc, 2020.

2. Edward Haletky, "VMware vSphere and Virtual Infrastructure Security: Securing the Virtual Environment", Prentice Hall, 2009.

3. Barrie Sosinsky, "Cloud Computing Bible", Wiley, 2011.

4. Jim Smith, Ravi Nair, “Virtual Machines - Versatile Platforms for Systems and Processes”, Morgan Kaufmann, 2005.

Web Resources:

1. Cloud Infrastructure and Services
   https://onlinecourses.swayam2.ac.in/e-learning/preview/nou25_cs33
2. Cloud Computing and Distributed Systems, IIT Patna
   https://nptel.ac.in/courses/106104182

Program: Computer Engineering	Final Year B.Tech.	Semester: VII
Internet of Everything (RCP23CCPE723)		
Internet of Everything Laboratory (RCP23CLPE723)		

Prerequisites:

1. Internet of Things
2. Computer Networks

Course Objective(s):

1. To understand the fundamentals of IoT and its evolution to IoE, including architectures, components, industrial application domains, and challenges.
2. To analyse IoT/IoE devices, sensing technologies, communication networks, connectivity solutions, and system architectures.
3. To apply communication protocols, data management techniques, and analytics frameworks for intelligent IoE systems.
4. To evaluate security, governance, emerging technologies, and real-world IoE applications for designing future-ready connected systems.

Course Outcomes:

On completion of the course, the learner will be able to:

CO	Course Outcomes	Blooms Level	Blooms Description
CO1	Explain IoT fundamentals, IoT reference architectures, IoE concepts, components, benefits, and application areas.	L2	Understand
CO2	Examine IoT devices, sensors, connectivity technologies, architectures, and deployment platforms used in IoE systems.	L4	Analyze
CO3	Assess IoT communication protocols, routing mechanisms, data lifecycle management, and analytics approaches for IoE applications.	L5	Evaluate
CO4	Assess security, privacy, governance requirements, and emerging trends such as AIoT, Industry 4.0, Digital Twins, and Blockchain-enabled IoE solutions.	L5	Evaluate

Internet of Everything (RCP23CCPE723)
Course Contents

Unit-I — 06 Hrs.

IoT Foundations and Evolution to IoE

IoT Stack & Core Concepts: IoT Stack, Overview of IoT, Characteristics of IoT, IoT Reference Architecture (e.g. ITU-T Y.2060).

Evolution from IoT to IoE, IoE Components. IoE Four-Pillar Framework

IoE Application Areas & Challenges: Application areas of IoE, IoE Challenges - scalability, interoperability, power, bandwidth, security & privacy.

M2M, IOE & Disambiguation: M2M Technology, Understanding IoE, IoE Pyramid model, IoT 2.0 and IoE Benefits, IoT vs IoE vs M2M.

Unit-II — 08 Hrs.

Devices, Sensors and Connectivity

Common IoT/IoE Devices: Introduction to Sensors, Actuators, Transducers and gateways.

Sensing Technologies: RFID in IoT, NFC (Near Field Communication) Connectivity Technologies: Short-range (Wi-Fi (802.11), Bluetooth / BLE, Zigbee (802.15.4), NFC), Low-Power Wide-Area Networks: LoRa / LoRaWAN, Sigfox, NB-IoT, LTE-M), Wired and Wireless device connectivity (5G/6G)

Unit-III — 06 Hrs.

IoE Architecture and Platforms

Overview of Physical Design of IoT, Logical Design of IoT: IoT Communication Model, IoT Levels and Deployment Templates.

Cisco IoE Reference Architecture.

System Architecture: 3-tier/multi-tier models Edge Computing and Fog Computing — architecture, comparison, Use Cases

IoT Devices & Software: IoT Devices, Software, Device Management Platform such as Google Cloud IoT, Cisco IOx etc.

Unit-IV — 08 Hrs.

Communication & Data Analytics

Protocol Layer Overview: IoT Protocol Stack— mapping protocols to OSI layers; why IoT needs specialized protocols.

Application Layer Protocols: MQTT, CoAP, AMQP, DDS, REST

Network & Adaptation Layer Protocols: IPv6 for IoT, 6LoWPAN, IoT Security Protocols, OS CORE

IoT Routing Protocols: Data-centric and Flat-Architecture Routing, Flooding and Gossiping, SPIN Protocol family, LEACH Protocol.

Data Lifecycle: Collection, aggregation, transmission, and visualization.

Edge and Fog Analytics

Unit-V 08 Hrs.

Security and Governance in IoE

Security Requirements beyond IoT- Trust, authentication, and secure orchestration across IoE layers.

Data Privacy and Governance- Regulations (GDPR), policy frameworks, and ethical considerations.

Security for Edge/Fog computing environments.

Blockchain for IoE Security- Decentralized trust and traceability in data sharing.

Unit-VI 06 Hrs.

Applications and Future Trends in IoE

Domain Case Studies: Smart Home Automation, Smart Cities, Environment Monitoring, Transportation & Supply Chain / Logistics, Agriculture, Industrial IoT, Healthcare, Smart Energy / Smart Grid.

Emerging Trends: AIoT (AI + IoT) — edge inference, TinyML, on-device anomaly detection, intelligent actuators, IIoT 4.0 / Industry 4.0 — cyber physical systems, Digital Twins Concept.

Internet of Everything Laboratory (RCP23CLPE723)

List of Laboratory Experiments

Suggested Experiments:(Any 10)

1. **Traffic Light Simulation using Arduino**

   Design and implement a basic traffic light control system using Red, Yellow, and Green LEDs connected to an Arduino. Program the LEDs to follow standard traffic signal timing sequences.

   Tools: Arduino Uno, LEDs, Resistors, Breadboard, Arduino IDE.

2. **IoE Architecture Setup**

   Configure a small-scale IoE environment integrating sensors, gateways, and cloud dashboards.

   Tools: Raspberry Pi, NodeMCU, ThingSpeak/AWS IoT.

3. **Connectivity and Data Transmission**

   Interfacing a Smoke Sensor with Arduino and Implementing MQTT/CoAP-Based Data Transmission with Latency and Throughput Analysis.

   Tools: Arduino Uno, LEDs, Resistors, Breadboard, Arduino IDE, MQTT.

4. **Edge-Fog-Cloud Integration** Deploy a data processing pipeline that filters data at the edge and stores aggregate data in the cloud.
   Tools: Node-RED + AWS IoT or Azure IoT.

5. **IoE Data Analytics** Collect real-time IoE data and perform basic analytics (averages, anomalies, trends).
   Tools: Python (Pandas), Grafana, or Power BI for visualization.

6. **Machine Learning for IoE** Train a simple ML model (e.g., temperature anomaly detection) and integrate it with an IoE node for real-time decision making.
   Tools: TensorFlow Lite, Edge Impulse, or Python scikit-learn.

7. **Interoperability Experiment** Integrate heterogeneous devices (e.g., Zigbee, Wi-Fi, Bluetooth) in a single IoE network and test interaction through a middleware (e.g., Node-RED or MQTT broker).

8. **Implementation of Digital Twin** Create a digital twin of a physical device that replicates live sensor data on a virtual dashboard.

9. **IoE Security Demonstration** Implement end-to-end security using TLS/SSL for MQTT communication and observe changes in performance. Optional: Simulate token-based authentication (OAuth or JWT).

10. **IoE Smart Application Prototype** Develop a mini-project such as a Smart Parking, Smart Home Energy Monitor, or Smart Campus Dashboard integrating multiple IoE layers — sensing, transmission, analytics, and visualization.

11. **Long-Range Communication using LoRa/LoRaWAN** Establish a point-to-point or star-topology LoRa link between two end nodes and a gateway. Transmit sensor data (e.g., temperature/humidity) over a long distance and measure RSSI, SNR, packet delivery ratio, and effect of spreading factor on range vs. data rate.

12. **RFID/NFC-Based Asset Tracking and Access Control** Build a system that reads RFID/NFC tags to identify and log assets or grant access, pushing scan events with timestamps to a cloud dashboard. Demonstrate tag enrolment, authentication, and event logging.

13. **Blockchain-Enabled Data Integrity for IoE** Record IoE sensor readings on a permissioned/test blockchain to ensure tamper-evident, traceable data sharing. Compare a normal write vs. a tampered record and verify integrity through hashing.

14. **TinyML On-Device Anomaly Detection (AIoT)** Train and deploy a lightweight ML model directly on a microcontroller for on-device inference (e.g., vibration/motion or sound anomaly detection) without cloud dependence, and benchmark inference latency and mem footprint against a cloud-based approach

Any other practical covering the syllabus topics and subtopics can be conducted.

Oral examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.

Text Books:

1. Dr. Prateek Jain and Dr. Archana Sharma, "Transitioning to Internet of Everything (IoE) Key Technology Application and Recent Trends", BFC Publications, 2024.
2. T. Kavitha, V. Ajantha Devi, Internet of Everything: Smart Sensing Technologies, Nova Publishers, 2022
3. Arshdeep Bahga and Vijay Madisetti, "Internet of Things: A Hands-On Approach", 1st Edition, Universities Press, 2016.

Reference Books:

1. Hang Song, "Internet of Everything: Key Technologies, Practical Applications and Security of IoT", World Scientific, Tsinghua University Press, 2023.
2. Pethuru Raj & Anupama Raman, The Internet of Everything (IoE): IoT to Next-Gen IoE, CRC Press, 2019.
3. David Hanes et al., IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things, Cisco Press, 2017.

Web Resources:

1. Introduction to Internet of Things by Prof. Rajbabu Velmurugan
   https://www.coursera.org/learn/introduction-to-internet-of-things
2. Introduction to Internet of Things, IIT Kharagpur
   https://onlinecourses.nptel.ac.in/noc22_cs53/preview

Program: Computer Engineering	Final Year B.Tech.	Semester: VII
Vulnerability Assessment and Penetration Testing (RCP23CCMD701)		

Prerequisites: Basic understanding of Computer Networks, Fundamentals of Operating Systems, Basic Programming Knowledge, Web Technologies.

Course Objective(s):

1. To introduce students to the fundamental principles of ethical hacking, penetration testing frameworks, and reconnaissance methodologies used in cybersecurity.
2. To develop skills in identifying and assessing security vulnerabilities in networks, systems, and applications using industry-standard scanning and analysis tools.
3. To provide hands-on experience in exploiting common vulnerabilities and implementing post exploitation techniques in controlled environments.
4. To Create professional penetration testing documentation, communicating security findings, and recommending effective remediation strategies.

Course Outcomes:

On completion of the course, the learner will be able to:

CO	Course Outcomes	Blooms Level	Blooms Description
CO1	Apply penetration testing methodologies, information gathering, reconnaissance, scanning, and enumeration techniques.	L3	Apply
CO2	Analyze network and system vulnerabilities using automated and manual assessment techniques and prioritize security risks.	L4	Analyze
CO3	Exploit web application, network, and system vulnerabilities using industry-standard tools including OWASP Top 10 and Metasploit.	L3	Apply
CO4	Create comprehensive penetration testing reports documenting vulnerabilities, risk assessment, and remediation recommendations.	L6	Create

Vulnerability Assessment and Penetration Testing (RCP23CCMD701) Course Contents

Unit-I 04 Hrs.

Introduction to Penetration Testing:

Introduction to Information Security and Ethical Hacking, Difference between Vulnerability Assessment and Penetration Testing, Types of Penetration Testing: Black Box, White Box, and Grey Box, Penetration Testing Methodologies: OWASP, PTES, NIST, Legal and Ethical Considerations, Setting up Penetration Testing Lab Environment, Introduction to Kali Linux and Essential Tools.

Unit-II 05 Hrs.

Information Gathering and Reconnaissance:

Passive Reconnaissance Techniques, OSINT (Open Source Intelligence) Gathering, DNS Enumeration and Zone Transfers, WHOIS Lookups and Domain Information, Active Reconnaissance and Footprinting, Network Scanning with Nmap, Port Scanning Techniques and Service Detection, Banner Grabbing and OS Fingerprinting.

Tools: Maltego, theHarvester, Shodan, Censys, Recon-ng

Unit-III 05 Hrs.

Vulnerability Assessment and Analysis:

Understanding Vulnerabilities and Common Vulnerability Scoring System (CVSS), Vulnerability Databases: CVE, NVD, Exploit-DB, Automated Vulnerability Scanning, Network Vulnerability Scanning with Nessus and OpenVAS, Web Application Vulnerability Scanning, Manual Vulnerability Assessment Techniques, Analyzing Scan Results and False Positive Detection, Vulnerability Prioritization and Risk Assessment.

Tools: Nessus, OpenVAS, Nikto, OWASP ZAP, Burp Suite

Unit-IV 06 Hrs.

Web Application Penetration Testing:

OWASP Top 10 Vulnerabilities (Latest Version), SQL Injection: Detection and Exploitation, Cross-Site Scripting (XSS): Reflected, Stored, and DOM based, Cross-Site Request Forgery (CSRF), Authentication and Session Management Flaws, Insecure Direct Object References (IDOR), Security Misconfigurations and Sensitive Data Exposure, XML External Entity (XXE) Attacks, Server-Side Request Forgery (SSRF), File Upload Vulnerabilities and Remote Code Execution, API Security Testing.

Tools: Burp Suite Professional, OWASP ZAP, SQLMap, XSSer.

Unit-V 04 Hrs.

System and Network Exploitation:

Exploitation Frameworks: Metasploit, Empire, Cobalt Strike, Buffer Overflow Attacks and Stack-based Exploitation, Password Attacks: Brute Force, Dictionary, Rainbow Tables, Credential Harvesting and Hash Cracking, Privilege Escalation: Windows and Linux, Post-Exploitation Techniques, Lateral Movement and Pivoting, Persistence Mechanisms, Wireless Network Penetration Testing.

Tools: Metasploit, John the Ripper, Hashcat, Hydra, Mimikatz, PowerSploit.

Unit-VI 04 Hrs.

Reporting and Remediation:

Penetration Testing Report Structure and Components, Executive Summary and Technical Findings, Risk Rating and Vulnerability Classification, Evidence Collection and Documentation, Remediation Recommendations and Best Practices, Compliance and Industry Standards: PCI-DSS, ISO 27001, NIST, Effective Communication with Stakeholders, Remediation Verification and Retesting, Continuous Security Improvement.

Career Paths: Certifications (OSCP, CEH, GPEN, GWAPT, PNPT).

Text Books:

1. Matt Walker, "CEH Certified Ethical Hacker All-in-One Exam Guide" by , 6^{th} Edition, McGraw Hill Education, 2024.
2. Georgia Weidman, "Penetration Testing: A Hands-On Introduction to Hacking", No Starch Press, 2014.
3. Gordon Fyodor Lyon, "Nmap Network Scanning: The Official Nmap Project Guide to Network Discovery and Security Scanning", Insecure.Com LLC, 2009.
4. David Kennedy, Jim O'Gorman, Devon Kearns, and Mati Aharoni, "Metasploit: The Penetration Tester's Guide", No Starch Press, 2011.
5. Andrew Davies and Dan Raywood, "Reporting Security Vulnerabilities: A Practical Guide for Penetration Testers", IT Governance Publishing, 2015.

Reference Books:

1. Raphaël Hertzog, Jim O'Gorman, and Mati Aharoni, "Kali Linux Revealed: Mastering the Penetration Testing Distribution", Offsec Press, 2^{nd} Edition 2024.
2. Dafydd Stuttard and Marcus Pinto, "The Web Application Hacker's Handbook: Finding and Exploiting Security Flaws", 2^{nd} Edition, Wiley, 2011.

3. Peter Kim, Secure Planet, “The Hacker Playbook 3: Practical Guide To Penetration Testing”, LLC, 2018.

4. Ben Clark, “Reconnaissance and Information Gathering”, Packt Publishing, 2023.

5. Peter Yaworski, “Real-World Bug Hunting: A Field Guide to Web Hacking”, No Starch Press, 2019.

6. Chris Hughes and Nikki Robinson, “Effective Vulnerability Management: Managing Risk in the Vulnerable Digital Ecosystem”, Apress, 2021.

Web Resources:

1. Introduction to Information Security - I, IIT Madras
https://nptel.ac.in/courses/106106129

2. Cyber Security and Privacy
https://onlinecourses.nptel.ac.in/e-learning/preview/noc23_cs127

Program: Computer Engineering	Final Year B.Tech.	Semester: VII
Research Ecosystem (RCP23ITELX08)		

Prerequisites:

1. Basic understanding of Engineering concepts and problem-solving techniques.
2. Fundamental knowledge of project-based learning, data analysis, and domain-specific tools.
3. Basic exposure to project development, technical report writing, and emerging technologies.

Course Objective(s):

1. To understand and identify complex engineering, societal, and industrial problems suitable for research through structured analysis of literature, patents, and real-world needs.
2. To learn and apply advanced research methodologies—including experimental, simulation based, and data-driven approaches—for designing and validating engineering solutions.
3. To gain proficiency in selecting and using appropriate tools, software, and platforms (AI/ML, simulation, CAD, IoT, etc.) for experiment design, data acquisition, and performance evaluation.
4. To transform research insights into innovative prototypes, products, or startup concepts by engaging with innovation ecosystems, TRLs, and funding mechanisms.
5. To analyze socio-technical systems, ethics, and sustainability implications of engineering solutions, ensuring responsible and impactful outcomes.
6. To integrate systems thinking, multidisciplinary perspectives, policy awareness, future technology trends, and technical documentation skills to develop comprehensive research proposals and outputs (papers, patents, reports).

Course Outcomes:

On completion of the course, the learner will be able to:

CO	Course Outcomes	Blooms Level	Blooms Description
CO1	Analyze complex engineering and societal problems through literature and patent analysis to identify research gaps.	L4	Analyze
CO2	Apply advanced research methods and tools to acquire data and assess solution performance	L3	Apply
CO3	Apply innovation frameworks and Technology Readiness Levels (TRL) for prototype and startup development.	L3	Apply
CO4	Analyze ethical, socio-technical, and sustainability issues related to engineering solutions.	L4	Analyze
CO5	Analyze multidisciplinary knowledge for engineering research proposal preparation.	L4	Analyze

Research Ecosystem (RCP23ITELX08)
Course Contents

Unit-I — 05 Hrs.

Research Gap Identification & Advanced Problem Formulation:

- Identifying high-impact problems from:
  - Industry and Society
  - Emerging technologies (AI, IoT, Blockchain, EV, etc.)
- Research gap identification techniques:
  - Systematic Literature Review (SLR)
  - Patent analysis
- Defining objectives, scope, constraints, and feasibility of the identified problem
- Writing problem statement for:
  - Research paper
  - Product/Startup idea

Activities:

- Convert Industry/Societal problem into research problem.
- Analyze research articles and extract research gaps.

Unit-II — 06 Hrs.

Experimental Design, Tools & Validation Techniques :

- Advanced research methodologies:
  - Experimental
  - Simulation
  - Data-driven
  - Design Thinking
- Tool selection:
  - Computational tools and software
  - Simulation and analysis platforms
  - System prototyping and testing platforms

- Data acquisition:
  - Data collection interfaces
  - Principles of data cleaning and organization (primary/secondary data)
- Validation techniques:
  - Accuracy and performance metrics
  - Benchmarking with existing solutions
- Result interpretation and visualization

Activities:

- Selection of Research Design for final year project.
- Apply evaluation metrics (e.g., ML model accuracy, system efficiency).

Unit-III 05 Hrs.

Research to Innovation, Product & Startup Ecosystem

- Research → Prototype → Product → Startup lifecycle
- Intellectual Property Rights (IPR), patents, copyrights
- Technology Readiness Levels (TRL)
- Case studies:
  - AI startups
  - EV ecosystem
  - Smart Cities
- Industry-academia collaboration models
- Funding opportunities (grants, incubation, government schemes – Anusandhan National Research Foundation (ANRF))

Activities:

- Convert project into startup idea or product pitch.
- Analyze failure of a technology/product.

Unit-IV 06 Hrs.

Socio-Technical Systems, Ethics & Sustainability:

- Socio-technical systems: Interaction of people, technology and environment
- Ethical issues:
  - Plagiarism
  - AI bias
  - Data privacy
- Sustainability in Engineering:
  - Green practices
  - Resource optimization
  - Lifecycle considerations
- Impact assessment:
  - Social impact
  - Economic impact
  - Environmental impact
- Policies and governance (data protection laws, standards and compliance requirements)

Activities:

- Case study: Facial recognition ethics / smart surveillance.
- Impact analysis of student's own project.

Unit-V 06 Hrs.

Integrated Research, Policy and Future Engineering Challenges:

- Systems thinking and holistic analysis
- Multidisciplinary problem solving:
  - Healthcare technology
  - Climate technology
  - Smart mobility
- Writing research proposals:
  - Problem → Methodology → Expected outcomes
- Future trends:
  - AI ethics

- Quantum computing
- Industry 5.0 / 6.0

- Scaling solutions from lab to society

Activities:

- Capstone Task:
  - Identify real-world problem.
  - Apply research methodology.
  - Propose:
    - Technical solution
    - Societal impact
    - Possible commercialization

Term Work Guidelines:

The Term Work (TW) shall consist of continuous assessment based on tutorial activities and a final group presentation.

a) **Part A (25 Marks):**

Each student shall complete a minimum of five tutorials based on the syllabus topics during the semester. The tutorials may include activities such as research gap identification, literature survey, case study analysis, tool-based experimentation, impact analysis, technical report writing, and innovation-oriented problem solving.

b) **Part B (25 Marks):**

Students shall work in groups and select a project/research topic based on emerging engineering, industrial, or societal challenges. The group presentation shall include:

i. Identification of a real-world problem
ii. Application of appropriate research methodology
iii. Proposed technical solution
iv. Analysis of societal impact
v. Possible commercialization/startup opportunities

Reference Books:

1. Kothari, C. R., "Research Methodology: Methods and Techniques", New Age International Publishers, New Delhi, 2023.

2. Booth, W. C., Colomb, G. G., Williams, J. M., Bizup, J., & FitzGerald, W. T., "The Craft of Research", University of Chicago Press, Chicago, 2024.

3. Montgomery, D. C., "Design and Analysis of Experiments", Wiley, New York, 2020.

4. Singh, Pooja, Katiyar, Aman, Rawat, Shalini, Dhuri, Archana, "Research Methodology and IPR", RK Publications, India, 2025.

5. Blok, Vincent (Ed.), "Putting Responsible Research and Innovation into Practice: A Multi Stakeholder Approach", Springer Cham, Switzerland, 2022.

Web Resources:

1. Research Methodology: Quantitative Methods and Computer Application
   (https://onlinecourses.swayam2.ac.in/e-learning/preview/ini26_ma02)

2. Research Methodology and IPR
   (https://onlinecourses.swayam2.ac.in/e-learning/preview/ntr26_ed28)

3. Research and Publication Ethics
   (https://onlinecourses.swayam2.ac.in/e-learning/preview/nou26_ge46)

4. Research Ethics and Plagiarism
   (https://onlinecourses.swayam2.ac.in/e-learning/preview/nou26_ge26)

5. Quantitative and Mixed Methods Research for Management
   (https://onlinecourses.swayam2.ac.in/e-learning/preview/imb26_mg21)

6. Doing Qualitative Research
   (https://onlinecourses.swayam2.ac.in/e-learning/preview/nou26_ge61)

Program: Computer Engineering	Final Year B.Tech.	Semester: VII
Project Stage-II (RCP23IPEL701)		

Course Objective(s):

1. To understand the basic concepts and principles in the development of solution.
2. To formulate solution of the problem statement.
3. To implement the solution as per the problem statement.
4. To develop team building, writing, logical reasoning and management skills.
5. To validate the proposed solution with different test cases.
6. To become independent person with ethical values and lifelong learning skills.

Course Outcomes:

On completion of the course, the learner will be able to:

CO	Course Outcomes	Blooms Level	Blooms Description
CO1	Analyze the problem statement and produce solution of the problem considering cultural, social, environmental and economic factors using appropriate tool and method.	L4	Analyze
CO2	Interpret project based learning that allows students to transfer existing ideas into new applications.	L2	Understand
CO3	Apply the ability to work in teams and manage to conduct the project development activity.	L3	Apply
CO4	Use different perspectives from relevant disciplines which help them to get internships, jobs, and admission for higher studies.	L3	Apply
CO5	Explain the project development in the form of technical writing, and interpret what constitutes plagiarism and the use of proper referencing styles.	L2	Understand

Syllabus:

The primary objective is to meet the milestones formed in the overall project plan decided in Project Stage-I. The idea presented in Project Stage-I should be implemented in Project Stage-II with results, conclusion and future work.

Guidelines:

- Each student group is required to maintain a separate log book for documenting various activities of the project (Refer Table 1).
- Each group will be reviewed twice in a semester and marks will be allotted based on the various points mentioned in the evaluation scheme.
- In the first review of this semester, each group is expected to complete 50% of project stage-II.
- In the second review of this semester, each group is expected to complete 100% of project stage-II.

Assessment Criteria:

- At the end of the semester, after confirmation by the project guide, each project group will submit a Project Stage-II completion report in the prescribed format for assessment to the departmental committee.
- Assessment of the project stage-II (at the end of the semester) will be done by the departmental committee.
- Oral examination shall be conducted by Internal and External examiners. Students have to give presentation and demonstration based on their project stage-II.

Prescribed Project Stage-II Report Guidelines:

The size of the report shall be of minimum 45 pages (excluding the cover and front pages). The Project Stage-II report should include appropriate content for:

- **Abstract**
- **Introduction**
  - Background
  - Motivation
  - Problem Statement
  - Objectives
  - Scope

- **Literature Survey**
  - Review of Existing System(s)
  - Limitations of Existing System(s)
- **Proposed System**
  - Analysis/Framework/ Algorithm
  - Details of H/W and S/W required
  - Design details
  - Methodology (your approach to solve problem)
- **Coding / Implementation**
- **Testing**
- **Conclusion**
- **References**

Assessment criteria for Continuous Assessment:

Guide will monitor weekly progress and marks allocation will be as per Table 2.

Assessment criteria for End Semester Exam:

Departmental committee will evaluate project as per Table 3.

Table 1: Log Book Format

Sr	Week (Start Date: End Date)	Work Done	Sign of Guide	Sign of Coordinator
1				
2				

Table 2: Continuous Assessment Sheet

Sr	PRN	Name of Student	Student Attendance (10 Marks)	Log Book Maintenance (10 Marks)	Implementation (10 Marks)	Testing (10 Marks)	Report (10 Marks)	Total (50 Marks)

Table 3: Evaluation Sheet

Sr	PRN	Name of Student	Depth of understanding (10 Marks)	Implementation (10 Marks)	Testing (10 Marks)	Report (10 Marks)	Presentation (10 Marks)	Total (50 Marks)

Program: Computer Engineering	Final Year B. Tech	Semester: VII
Disaster Management and Preparedness (RCP23ICHSX09)		

Course Objective(s):

1. To provide basic understanding of hazards, disasters and various types and categories of disaster occurring around the world.
2. To identify extent and damaging capacity of a disaster.
3. To study and understand the means of losses and methods to overcome /minimize it.
4. To understand roles and responsibilities of individual and various organizations during and after disaster.
5. To appreciate the significance of GIS, GPS in the field of disaster management.
6. To understand the emergency government response structures before, during and after disaster.

Course Outcomes: On completion of the course, the learner will be able to:

CO	Course Outcomes	Blooms Level	Blooms Description
CO1	Apply disaster management principles and guidelines.	L3	Apply
CO2	Analyze risk assessments.	L4	Analyze
CO3	Demonstrate community awareness and participation.	L3	Apply
CO4	Use Science and Technology tools (GIS, GPS).	L3	Apply
CO5	Prepare disaster management plans.	L3	Apply

Disaster Management and Preparedness (RCP23ICHSX09) Course Contents

Unit-I 06 Hrs.

Understanding Disasters & Hazards:

Definition and types of disasters: Natural, Man-made and hybrid disasters.

Study of Natural disasters: Flood, drought, cloud burst, Earthquake, Landslides, Avalanches, Volcanic eruptions, Mudflow, Cyclone, Storm, Storm Surge, climate change, global warming, sea level rise, ozone depletion etc.

Study of Human/Technology Induced Disasters: Chemical, Industrial and Nuclear disasters, internally displaced persons, road and train accidents Fire Hazards, terrorism, militancy, Hazard & Vulnerability profiles of India (seismic zones, flood-prone areas). India's vulnerability to disasters, and the impact of disasters on National development.

Unit-II 06 Hrs.

Disaster Risk Reduction (DRR) & Mitigation:

Disaster Management Cycle: Prevention, Mitigation, Preparedness, Response, Recovery. Need for disaster prevention and mitigation, mitigation guiding principles, challenging areas, structural and non-structural measures for disaster risk reduction. Risk Assessment & Vulnerability Analysis.

Science & Technology: Use of information management, Geo informatics like RS, GIS, GPS and remote sensing mitigation measure.

Unit-III 04 Hrs.

Disaster Preparedness & Response:

Preparedness Planning, Early Warning Systems (EWS), & Communication. Emergency Response: Search & Rescue, Logistics, Medical Aid. Psychological Response & Management (Trauma, Stress). Role of IT, Media, Govt., NGOs, & Community.

Unit-IV 04 Hrs.

Recovery, Rehabilitation & Reconstruction:

Post-disaster damage assessment. Rehabilitation, Reconstruction, & Livelihood Restoration. Sanitation, Hygiene, & Waste Management.

Unit-V 08 Hrs.

Policy, Governance & Capacity Building:

National Disaster Management Authority (NDMA) & Legislation. Institutional Mechanisms & C

munity Mobilization. Non-Structural Mitigation: Community based disaster preparedness, capacity development and training, awareness and education, contingency plans.

Case studies on disaster (National /International):

Case study discussion of National Disasters: Tsunami (2004), Bhopal gas tragedy, Kerala and Uttarakhand flood disaster, 26th July 2005 Mumbai flood. Case study discussion of International Disasters: Hiroshima – Nagasaki (Japan), Cyclone Phailin (2013), Fukushima, Daiichi nuclear disaster (2011), Chernobyl meltdown .

Reference Books:

1. Harsh K. Gupta, "Disaster Management", Universities Press Publications, 1st Edition, Reprint 2018.
2. O. S. Dagur, "Disaster Management: An Appraisal of Institutional Mechanisms in India", published by Centre for land warfare studies, New Delhi, 2011.
3. Damon Copolla, Butterworth Heinemann, "Introduction to International Disaster Management", Elsevier Publications, 4th Edition, 2020.
4. Jack Pinkowski, CRC Press, "Disaster Management Handbook", Taylor and Francis group, 2008.
5. Rajdeep Dasgupta, "Disaster management & rehabilitation", Mittal Publications, New Delhi, 2007.
6. R B Singh, "Natural Hazards and Disaster Management, Vulnerability and Mitigation", Rawat Publications, 2006.
7. C. P. Lo Albert, K.W. Yonng, "Concepts and Techniques of Geographic Information Systems", Pearson/Prentice Hall, 2nd Edition, 2007.
8. Claudia G. Flores Gonzales, "Risk management of natural disasters", KIT Scientific Publishing, 2010.
9. W. Nick Carter, "Disaster Management – a disaster manger's handbook", Asian Development Bank, 2008.
10. R. K. Srivastava, "Disaster Management in India", Ministry of Home Affairs, GoI, New Delhi, 2011.
11. Wil Mara, "The Chernobyl Disaster: Legacy and Impact on the Future of Nuclear Energy", Marshall Cavendish Corporation, New York, 2011.

12. Ronald Eisler, "The Fukushima 2011 Disaster", Taylor & Francis, Florida, 2013.

(Learners are expected to refer reports published at national and international level and updated information available on authentic web sites.)

Program: Computer Engineering	Final Year B.Tech.	Semester: VII
AI for Language Generation and Understanding (RCP23CCSC701)		

Prerequisites: Machine Learning, Deep Learning (Neural Networks and RNNs), Engineering Mathematics (Linear Algebra, Calculus, and Statistics).

Course Objective(s):

1. To explain the attention mechanism and how it overcomes the limitations of sequential seq2seq models.
2. To develop the ability to build Transformer-based models for machine translation, summarization, and question answering.
3. To enable the use of transfer learning with state-of-the-art pre-trained models such as BERT and T5

Course Outcomes:

On completion of the course, the learner will be able to:

CO	Course Outcomes	Blooms Level	Blooms Description
CO1	Explain the attention mechanism using the queries, keys, and values formulation.	L2	Understand
CO2	Build a neural machine translation model with attention and evaluate it using BLEU and ROUGE.	L5	Evaluate
CO3	Construct Transformer architectures using self-attention and multi-head attention.	L6	Create
CO4	Develop a Transformer-based text summarization model.	L6	Create
CO5	Apply transfer learning with BERT and T5 for downstream NLP tasks.	L3	Apply
CO6	Implement a question-answering system using Hugging Face pre-trained models.	L3	Apply

AI for Language Generation and Understanding (RCP23CCSC701) Course Contents

Unit-I 04 Hrs.

Seq2Seq Models and the Attention Mechanism:

Shortcomings of traditional seq2seq models, the seq2seq model with attention, and the queries–keys–values formulation of attention; basic attention and scaled dot-product attention labs.

Unit-II 04 Hrs.

Neural Machine Translation with Attention:

Setup for machine translation, teacher forcing, building an NMT model with attention; evaluation with BLEU and ROUGE-N scores, and decoding strategies — sampling, beam search, and Minimum Bayes Risk. English-to-German translation programming assignment.

Unit-III 04 Hrs.

Transformer Architecture and Self-Attention:

Transformers vs. RNNs, transformer overview and applications, scaled dot-product attention, masked self-attention, multi-head attention, the transformer decoder, and positional encoding.

Unit-IV 04 Hrs.

Text Summarization with Transformers:

Building a transformer-based summarizer, masking, and positional encoding in practice; transformer summarizer programming assignment.

Unit-V 05 Hrs.

Transfer Learning and Pre-trained Language Models:

Transfer learning in NLP, the ELMo–GPT–BERT–T5 family, Bidirectional Encoder Representations from Transformers (BERT) and its objective, fine-tuning BERT, the T5 model, multi-task training strategy, and the GLUE benchmark.

Unit-VI 04 Hrs.

Question Answering with BERT, T5 and Hugging Face:

Building question-answering models, SentencePiece and Byte-Pair Encoding, and using the Hugging Face library with base and fine-tuned models; question-answering programming assignment.

Web Resources:

https://www.coursera.or g/learn/attention-models-in-nlp

Semester - VIII

Program: Computer Engineering	Final Year B.Tech	Semester: VIII
Quantum Computing (RCP23CCPE811)		

Prerequisites:

1. Engineering Mathematics
2. Information and Cyber Security
3. Python Programming Laboratory

Course Objective(s):

1. To introduce the mathematical and quantum-mechanical principles underlying the qubit model of computation.
2. To impart knowledge of quantum gates, circuits, entanglement and the foundational and famous quantum algorithms.
3. To develop practical skills in quantum programming on simulators and NISQ hardware using open-source frameworks.
4. To analyze the impact of quantum algorithms on classical and post-quantum cryptography, and to create awareness of emerging quantum applications and trends.

Course Outcomes:

On completion of the course, the learner will be able to:

CO	Course Outcomes	Blooms Level	Blooms Description
CO1	Apply linear-algebraic and quantum-mechanical formalism to describe qubit states and operations	L3	Apply
CO2	Design quantum circuits using single- and multi-qubit gates to demonstrate entanglement and teleportation.	L6	Create
CO3	Analyze foundational and famous quantum algorithms (Deutsch–Jozsa, Grover, Shor) and their complexity advantages.	L4	Analyze
CO4	Apply open-source frameworks to implement quantum programs on simulators and NISQ devices.	L3	Apply
CO5	Analyze quantum communication and cryptographic protocols, including QKD and post quantum cryptography.	L4	Analyze
CO6	Evaluate emerging applications of quantum computing in optimization, machine learning and chemistry.	L5	Evaluate

Quantum Computing (RCP23CCPE811)
Course Contents

Unit-I — 06 Hrs.

Introduction to Quantum Computing & Mathematical Foundations

- Motivation: limits of classical computing, Moore's law, Feynman's vision; classical vs quantum computation.
- Brief recap of classical bits, gates, reversible logic (Toffoli, Fredkin) and computational complexity classes (P, NP, BQP).
- Complex numbers and Euler's formula; vectors, inner product and norms; Dirac (bra-ket) notation.
- Matrices, Hermitian and unitary operators; eigenvalues and eigenvectors; tensor (Kronecker) products.
- Probability vs probability amplitude; the Born rule and quantum interference.
- Brief history and timeline of quantum computation milestones (Feynman, Deutsch, Shor, Grover, IBM, Google supremacy, NISQ era).

Unit-II — 07 Hrs.

Qubits, Single-Qubit Gates and Measurement

- Postulates of quantum mechanics: state space, evolution, measurement, and composite systems.
- The qubit: superposition, computational basis 0, 1; geometric representation on the Bloch sphere.
- Single-qubit gates: Pauli X, Y, Z; Hadamard (H); phase gates S and T; rotation gates R_x, R_y, R_z.
- Quantum circuit notation; identity and global vs. relative phase.
- Measurement in the computational and Hadamard bases; projective measurement and collapse.
- No-cloning theorem and its implications; reversibility of quantum gates.
- Multi-qubit systems and tensor products; product vs. entangled states; CNOT, CZ, SWAP, controlled-U, Toffoli; universal gate sets.
- Bell states (H + CNOT); quantum teleportation and superdense coding.

Unit-III 08 Hrs.

Quantum Algorithms

- Quantum parallelism, oracles and the role of interference.
- Deutsch and Deutsch–Jozsa; Bernstein–Vazirani and Simon (bridges to period-finding).
- Quantum Fourier Transform (QFT): definition, circuit, intuition; Quantum Phase Estimation (QPE).
- Grover's search: $O(\sqrt{N})$ speedup, oracle and diffusion operator; amplitude amplification.
- Shor's factoring algorithm: period/order finding via QFT (overview); complexity advantages.

Unit-IV 06 Hrs.

Quantum Programming & Simulation

- Quantum programming platforms: Qiskit, Cirq, Q#, Amazon Braket; a first quantum program.
- Quantum circuit design; statevector and other simulators; quantum assembly basics.
- Execution on cloud quantum computers (IBM Quantum, AWS Braket, Azure Quantum).
- Noise, decoherence, gate fidelity, T_1/T_2; the NISQ era.
- Error correction at an intuitive level: bit-flip/phase-flip codes; surface code (overview).
- Hands-on implementation of Deutsch and Grover algorithms; result visualization.

Unit-V 08 Hrs.

Quantum Communication & Cryptography

- Quantum channels, state transmission, noise; quantum repeaters.
- Quantum Key Distribution (QKD): BB84, B92, E91; eavesdropping detection and security intuition.
- Shor's impact on RSA, Diffie–Hellman, ECC; "harvest-now, decrypt-later" threat.
- Grover's impact on AES and hash functions—effective key-length halving.
- Post-Quantum Cryptography (PQC): NIST standards (ML-KEM/Kyber, ML-DSA/Dilithium, SLH-DSA/SPHINCS+); lattice-, code-, and hash-based families.
- QKD vs. PQC; the road to a quantum-safe Internet.

Unit-VI 07 Hrs.

Applications, Hardware & Emerging Trends

- Physical qubit implementations: superconducting, trapped ion, photonic, neutral atom; DiVincenzo criteria.
- Optimization: combinatorial problems, quantum annealing, QAOA (introductory).
- Quantum chemistry and VQE; quantum machine learning: feature spaces, variational classifiers (conceptual).
- Quantum sensing and metrology (brief).
- Emerging trends: quantum-inspired classical algorithms, India's National Quantum Mission, the quantum-safe transition roadmap.

Text Books:

1. Ahmed Banafa, "Introduction to Quantum Computing", River Publishers, 1st Edition, 2023.
2. Thomas G. Wong, "Introduction to Classical and Quantum Computing", Rooted Grove, 2022.
3. Venkateswaran Kasirajan, "Fundamentals of Quantum Computing -Theory and Practice", Springer, 1st Edition, 2021.
4. Maria Schuld, Francesco Petruccione, "Machine Learning with Quantum Computers", 2nd Edition, Springer International Publications, 2021.
5. Michael A. Nielsen and Isaac L. Chuang, "Quantum Computation and Quantum Information", Cambridge University Press, 2013.

Reference Books:

1. Vladimir Silva, "Practical Quantum Computing for Developers", Apress, 2018.
2. Eleanor Rieffel and Wolfgang Polak, "Quantum Computing: A Gentle Introduction", MIT Press, 2014.
3. Phillip Kaye, Raymond Laflamme and Michele Mosca, "An Introduction to Quantum Computing", Oxford University Press, 2007.
4. N. David Mermin, "Quantum Computer Science: An Introduction", Cambridge University Press, 2015.

Web Resources:

1. Introduction to Quantum Computing: Quantum Algorithms and Qiskit
   https://onlinecourses.nptel.ac.in/e-learning/preview/noc26_cs89
2. Quantum Computing: Algorithms and Limitations
   https://onlinecourses.nptel.ac.in/e-learning/preview/noc26_cs06

Program: Computer Engineering	Final Year B.Tech.	Semester: VIII
Human Computer Interaction (RCP23CCPE812)		

Prerequisites: Web Technologies, Experience in designing interfaces for applications and web sites, Basic knowledge of designing tools and languages like HTML, Java, etc.

Course Objective(s):

1. Learn the foundation of human machine interaction.
2. Understand the importance of human psychology in designing good interfaces.
3. Be aware of mobile interaction design and its usage in day – to – day activities.
4. Understand various design technologies to meet user requirements.
5. Encourage to indulge into research in Machine Interaction Design

Course Outcomes:

On completion of the course, the learner will be able to:

CO	Course Outcomes	Blooms Level	Blooms Description
CO1	Identify User Interface (UI) design principles.	L1	Remember
CO2	Analysis of effective user-friendly interfaces.	L4	Analyze
CO3	Apply Interactive Design process in real world applications.	L3	Apply
CO4	Evaluate UI design and justify.	L5	Evaluate
CO5	Create application for social and technical task.	L6	Create

Human Computer Interaction (RCP23CCPE812) Course Contents

Unit-I 07 Hrs.

FOUNDATIONS OF HMI

The Human: History of User Interface Designing, I/O channels, Hardware, Software and Operating environments, The Psychopathology of everyday Things, Psychology of everyday actions, Reasoning and problem solving. The computer: Devices, Memory, processing and networks. Interaction: Models, frameworks, Ergonomics, styles, elements, interactivity, Paradigms.

Unit-II 08 Hrs.

DESIGN & SOFTWARE PROCESS

Mistakes performed while designing a computer system, Human interaction with computers, importance of human characteristics, human consideration, Human interaction speeds. Interactive Design basics, process, scenarios, navigation, Iteration and prototyping. HMI in software process: software life cycle, usability engineering, Prototyping in practice, design rationale. Design rules: principles, standards, guidelines, rules. Recognize the goals, Goal directed design process. Evaluation Techniques: Universal Design.

Unit-III 07 Hrs.

GRAPHICAL USER INTERFACE

The graphical User Interface: Popularity of graphics, the concept of direct manipulation, graphical systems, Characteristics. Web user Interface: Interface popularity, characteristics. The merging of graphical Business systems and the Web. Principles of user interface design

Unit-IV 08 Hrs.

SCREEN DESIGNING:

Design goals , Screen planning and purpose, organizing screen elements, ordering of screen data and content , screen navigation and flow, Visually pleasing composition, amount of information, focus and emphasis, presentation information simply and meaningfully, information retrieval on web, statistical graphics, Technological consideration in interface design.

Unit-V 06 Hrs.

INTERFACE DESIGN FOR MOBILE DEVICES

Mobile Ecosystem: Platforms, Application frameworks: Types of Mobile Applications: Widgets, A plications, Games, Mobile Information Architecture, Mobile 2.0, Mobile Design: Elements of Mo

Design, Tools.

Unit-VI 06 Hrs.

INTERACTION STYLES AND COMMUNICATION

Windows: Characteristics, Components, Presentation styles, Types of Windows, Management, operations. Text messages: Words, Sentences, messages and text words, Text for web pages. Icons, Multimedia and colors.

Text Books:

1. Kasper Hornbæk, Per Ola Kristensson, Antti Oulasvirta, "Introduction to Human-Computer Interaction", Oxford University Press, 2026.
2. J Dhilipan, V Saravanan, R Agusthiyar, Human Machine Interaction in the Digital Era Towards Conversational Artificial Intelligence, CRC Press, 2024.
3. Gerard Jounghyun Kim, "Human–Computer Interaction", CRC Press, 2015.

Reference Books:

1. Guy A. Boy, The Handbook of Human-Machine Interaction: A Human-Centered Design Approach, CRC Press, 2011.
2. Alan Dix, Janet Finlay, Gregory Abowd, Russell Beale, —Human Computer Interaction, 3rd Edition, Pearson Education, 2004.

Web Resources:

1. Human Computer Interaction
   https://onlinecourses.nptel.ac.in/e-learning/preview/noc26_cs70
2. User-Centric Computing for Human-Computer Interaction
   https://onlinecourses.nptel.ac.in/e-learning/preview/noc20_cs45

Program: Computer Engineering	Final Year B.Tech.	Semester: VIII
Geographic Information System (RCP23CCPE813)		

Prerequisites:

1. Basic knowledge of computer systems, databases, and programming concepts.
2. Fundamental understanding of geography, maps, and spatial data representation.

Course Objective(s):

1. To introduce the fundamental concepts, components, and applications of Geographic Information Systems (GIS).
2. To understand spatial and non-spatial data models and their representation in computer systems.
3. To study raster and vector data structures and compare their advantages and limitations.
4. To design and query spatial databases using spatial data types, spatial indexing, spatial SQL and PostGIS.
5. To develop knowledge of map digitization, editing, vector topology creation and validation, and attribute data management.
6. To understand Digital Elevation Models and triangulation methods (Delaunay/TIN), and apply GIS and Remote Sensing in Computer Engineering domains such as smart cities, navigation, and AI-based image processing.

Course Outcomes:

On completion of the course, the learner will be able to:

CO	Course Outcomes	Blooms Level	Blooms Description
CO1	Explain the fundamental concepts, components, and applications of GIS and Remote Sensing systems.	L2	Understand
CO2	Differentiate raster and vector data models with respect to storage, processing, and application areas.	L4	Analyze
CO3	Implement spatial database queries using spatial data types, spatial indexing, spatial SQL and PostGIS.	L3	Apply
CO4	Construct vector topology and attribute mappings through digitization, editing, and validation techniques.	L6	Create
CO5	Design Digital Elevation Models using triangulation (Delaunay/TIN) and grid structures for surface analysis.	L6	Create
CO6	Apply GIS and Remote Sensing techniques to real-world applications such as navigation, smart-city planning, and satellite image analysis.	L3	Apply

Geographic Information System (RCP23CCPE813) Course Contents

Unit I Introduction to GIS 06 Hrs.

Definition, sources of data, types of data, concept of space and time in GIS, spatial information theory, history of GIS, elements and components of GIS, objectives of GIS, hardware and software requirements of GIS, applications of GIS.

Unit II Spatial Data Models – Raster and Vector 07 Hrs.

Layers and coverage; conceptual models of spatial information; representation of spatial data models in computer — raster and vector models; raster encoding and compression (run-length, quadtree); vector primitives (point, line, polygon); comparative overview between raster and vector models with respect to storage, processing and application.

Unit III Spatial Databases and Spatial Data Management 08 Hrs.

Review of DBMS models — hierarchical, network and relational structures, and their limitations for spatial data. Spatial database concepts: spatial data types (geometry vs geography); OGC Simple Features specification; spatial reference systems and SRID. Spatial indexing: R-tree, R+-tree, quadtree, grid index. Spatial SQL and spatial operators (intersection, containment, buffer, distance, spatial joins). PostGIS: architecture, geometry/geography types, ST_ spatial functions, spatial queries and relationships, raster support; spatial database design and query optimization.

Unit IV Vector Topology, Digitizing, Editing and Structuring of Map Data 08 Hrs.

Digitizing: manual, semi-automatic and automatic. Editing: error detection and correction, snapping and fuzzy tolerances. Vector topology: concept of topology; topological relationships — adjacency, connectivity, containment; planar enforcement. Topological data structures: arc–node structure, DIME, POLYVRT. Building topology, topological rules and validation, topology cleaning. Attribute data management, attribute map generation, and linking of spatial and attribute data.

Unit V Digital Elevation Models and Triangulation 07 Hrs.

Need and sources of DEM. Structures of DEM: regular grid, contour/line based, and triangulated irregular network (TIN). Triangulation: Delaunay triangulation and the empty-circle criterion; constrained Delaunay triangulation; Voronoi / Thiessen polygons and their duality with Delaunay. TIN generation and TIN vs grid trade-offs. Surface interpolation: IDW, spline, kriging (introducto surface analysis — slope, aspect, and contour generation.

Unit VI Applications of Remote Sensing and GIS 06 Hrs.

GIS software development; spatial database management and optimization; GPS-based navigation and route optimization systems; smart city planning and infrastructure monitoring; satellite image processing using Artificial Intelligence and Machine Learning.

Text Books:

1. Kang-Tsung Chang, Introduction to Geographic Information Systems, 10th Edition, McGraw Hill Education, 2023.
2. Paul A. Longley, Michael F. Goodchild, David J. Maguire, David W. Rhind , Geographic Information Science and Systems, 5^{th} Edition, Wiley, 2021.
3. Thomas Lillesand, Ralph W. Kiefer, Jonathan Chipman, Remote Sensing and Image Interpretation, 8^{th} Edition, Wiley, 2022.

Reference Books:

1. Surya S Durbha, Jibonananda Sanyal, Lexie Yang, Sangita S Chaudhari, Ujwala Bhangale, Ujwala Bharambe, Kuldeep Kurte, Advances in Scalable and Intelligent Geospatial Analytics Challenges and Applications, CRC Press, 2023.
2. Michael Zeiler, Modeling Our World: The ESRI Guide to Geodatabase Design–, ESRI Press, 2000.
3. David J. Maguire, Michael Batty, Michael F. Goodchild, GIS, Spatial Analysis and Modeling, ESRI Press, 2005.
4. John R. Jensen, Remote Sensing of the Environment: An Earth Resource Perspective, 3^{rd} Edition, Pearson, 2013.
5. Sathheesh Gopi, R. Sathikumar, N. Madhu, Advanced Surveying: Total Station, GIS and Remote Sensing, Pearson Education, 2006.
6. Michael N. DeMers, Fundamentals of Geographic Information Systems, 5^{th} Edition, Wiley, 2008.
7. Paul Bolstad, GIS Fundamentals: A First Text on Geographic Information Systems, 7^{th} Edition, XanEdu Publishing, 2022.
8. Peter A. Burrough, Rachael A. McDonnell, Christopher D. Lloyd, Principles of Geographic Information Systems, Oxford University Press, 2015.

Web Resources:

1. Geographic Information Systems

   https://onlinecourses.nptel.ac.in/e-learning/preview/noc22_ce26

2. Remote Sensing and GIS

   https://onlinecourses.nptel.ac.in/e-learning/preview/noc22_ce84

Program: Computer Engineering	Final Year B.Tech.	Semester: VIII
Internship (OJT) cum Project (RCP23IPELX10)		

Prerequisites:

1. Fundamental knowledge of engineering concepts and programming
2. Basic understanding of professional ethics and communication skills
3. Exposure to laboratory courses and mini-projects
4. Teamwork and problem-solving skills

Course Objective(s):

1. To provide practical exposure to industrial and professional environments.
2. To bridge the gap between academic learning and industry practices.
3. To develop technical, communication, teamwork, and managerial skills.
4. To enable application of engineering knowledge for solving real-world problems.
5. To familiarize students with modern tools, technologies, and professional ethics.
6. To prepare students for industry-oriented careers through project implementation and professional experience.

Course Outcomes:

On completion of the course, the learner will be able to:

CO	Course Outcomes	Blooms Level	Blooms Description
CO1	Apply engineering knowledge and technical skills in industrial environments.	L3	Apply
CO2	Analyze and solve practical engineering problems using appropriate tools and techniques.	L4	Analyze
CO3	Demonstrate professional ethics, teamwork, communication, and organizational skills.	L3	Apply
CO4	Utilize modern engineering tools and technologies for assigned tasks and projects.	L3	Apply
CO5	Prepare technical reports, presentations, and project documentation effectively.	L6	Create
CO6	Adapt to professional work culture and demonstrate readiness for professional careers.	L3	Apply

Internship (OJT) cum Project (RCP23IPELX10) Course Contents

Overview of the OJT Process

The On-Job Training (OJT) program begins with an introduction to workplace ethics, safety practices, organizational culture, and professional conduct to prepare students for industrial environments. Students are then exposed to industrial processes, workflow practices, tools, and modern technologies relevant to their domain of study.

During the technical training phase, students work on assigned industrial tasks and projects under the guidance of faculty mentors and industry supervisors. The program emphasizes the application of engineering concepts, teamwork, communication skills, and problem-solving abilities in real-world professional settings.

The later stages of the OJT focus on project implementation, report writing, technical documentation, and presentation preparation. Students are required to maintain logbooks and progress records throughout the training period. At the end of the semester, students present their project work through demonstrations and viva-voce examinations, followed by comprehensive evaluation and feedback.

Internship (OJT) Schedule, Duration and Credits

The Internship / On-Job Training (OJT) cum Project is an integral part of the Final Year B.Tech. program and carries a total of **18 academic credits** toward completion of the degree requirements. Each credit corresponds to **40 hours** of work, resulting in approximately **550–600 hours** of industrial training and project work over one semester.

The full-time OJT program is conducted during the designated semester as per the academic schedule. The duration of the OJT is one complete semester in the final year. Faculty mentors are assigned for mentoring, monitoring, and evaluating students throughout the OJT period.

Approximately 50% of the students undergo OJT during Semester VII while the remaining students continue academic activities in the institute. In Semester VIII, the arrangement is interchanged to ensure equal academic and industrial exposure for all students.

To avoid academic scheduling conflicts and ensure fair evaluation, oral, practical, and End Semester Examinations for both Semester VII and Semester VIII will be conducted at the end of Semester VIII. Students are issued a combined marksheet covering both semesters.

Types of External OJT Opportunities

Sr. No.	Type of OJT	Examples / Description
1	Industry OJT	Industrial internships with or without stipend.
2	Government / PSU OJT	BARC, SAMEER, Railways, ISRO, etc.
3	Research OJT	IITs, NITs, IISc, IIMs, and other research institutes.
4	Institute-Based OJT	OJT at institutes approved by RCPIT and departments.
5	Other Approved OJT	Any approved industrial/research work relevant to the domain.

OJT / Internship Sources

To facilitate OJT and internship opportunities, AICTE has signed Memorandums of Understanding (MoUs) with various industries, government organizations, and training institutions. These collaborations provide students with industrial exposure, skill development opportunities, research internships, and professional training through multiple recognized platforms and organizations.

Students may also approach organizations such as the Board of Apprenticeship Training (BOAT) and Board of Practical Training (BOPT) for additional training opportunities.

Major OJT / Internship Sources

Sr. No.	Organization / Platform	Nature of Opportunity
1	Internshala	Internship and industry training opportunities
2	NETiit (Taiwan)	International internship opportunities
3	HireMee	Skill-based internships and employability support
4	IGNCA	Research and cultural project opportunities
5	CCEI, Republic of Korea	International innovation and creative economy internships
6	IIWM, Bangalore	Waste management and sustainability projects
7	Engineering Council of India (ECI)	Engineering and professional development activities
8	Fourth Ambit	Industry training and technical internships
9	LinkedIn	Professional networking and internship opportunities
10	Telecom Sector Skill Council (TSSC)	Telecom and communication technology training
11	SCHOLARSMERIT	Internship and skill enhancement programs
12	Studenting Era	Student internship and academic support activities
13	MSME	Industry exposure and entrepreneurship opportunities
14	BOAT	Apprenticeship and industrial training programs
15	BOPT	Practical and apprenticeship-based training opportunities

Evaluation Metrics

Sr. No.	Evaluation Component	Marks
1	Internal Review 1	100
2	Final Internal Review	100
3	End Semester Evaluation (OJT Viva / Presentation)	200
Total		**400**

Evaluation Parameters

The student shall be evaluated based on the following parameters:

- Understanding of assigned work.
- Progress and task completion.
- Technical skill development.
- Learning ability and initiative.
- Professional behaviour and discipline.
- Communication and presentation skills.
- Problem solving and contribution.
- Quality of work and innovation.
- Logbook maintenance and report documentation.
- Application of engineering knowledge.

Necessary Documents

Students are required to submit and maintain the following documents during the OJT period:

1. No Objection Certificate (NOC)
2. Undertaking for opting OJT
3. OJT Joining Form
4. Parent/Guardian Undertaking Form
5. Contact Details and OJT Objectives Form
6. Student Logbook / Daily Activity Record

7. Internal Review Evaluation Sheets
8. External Supervisor Evaluation Form
9. Internal Supervisor Feedback Form
10. Student Feedback Form
11. SDG Mapping Form
12. Final OJT Report and Presentation
13. OJT Completion Certificate

Scheme of Examination

Component	**Marks**
Continuous Internal Assessment	200
End Semester OJT Evaluation (Report and Presentation)	200
Total	**400**

Program: Computer Engineering	Final Year B.Tech.	Semester: VIII
Multi-Agent Systems (RCP23CCSC801)		

Prerequisites: Python Programming and Foundations of Generative AI / Large Language Models.

Course Objective(s):

1. To introduce the architecture and design principles of single- and multi-agent AI systems.
2. To develop the ability to build collaborative agent teams using tools, memory, guardrails, and orchestration with CrewAI.
3. To enable the deployment, monitoring, and evaluation of production-ready, scalable multi-agent systems

Course Outcomes:

On completion of the course, the learner will be able to:

CO	Course Outcomes	Blooms Level	Blooms Description
CO1	Explain the foundations of AI agents and what makes multi-agent systems intelligent.	L2	Understand
CO2	Build multi-agent systems that plan, reason, and collaborate to automate workflows.	L6	Create
CO3	Apply memory, guardrails, tools, and the Model Context Protocol to control agent behavior.	L3	Apply
CO4	Orchestrate agents using collaboration patterns and CrewAI Flows.	L3	Apply
CO5	Evaluate multi-agent system reliability through monitoring, tracing, and structured evaluation.	L5	Evaluate
CO6	Analyze real-world enterprise deployments of AI agents across industries.	L4	Analyze

Multi-Agent Systems (RCP23CCSC801)
Course Contents

Unit-I 04 Hrs.

Foundations of AI Agents:

What AI agents are, real-world use cases, what makes an AI agent intelligent, and why the AI agent revolution is happening now; building your first AI agent.

Unit-II 04 Hrs.

Designing and Building Multi-Agent Systems:

Planning multi-agent systems, tuning behavior through context engineering, running systems in production, and tactics for debugging, observing, and optimizing; building multi-agent systems and an automated code reviewer.

Unit-III 04 Hrs.

Controlling Agents – Memory, Guardrails and Tools:

Understanding agent workflows, incorporating memory and knowledge, controlling agents with guardrails and execution hooks, using and building custom tools, and adopting the Model Context Protocol (MCP); building a meeting-preparation co-pilot and extending the code reviewer.

Unit-IV 04 Hrs.

Orchestrating Multi-Agent Systems:

Agent collaboration and communication, the Agent-to-Agent (A2A) Protocol, and orchestrating agents with CrewAI Flows using sequential, parallel, hierarchical, hybrid, and async processes; building coordination patterns and a deep research flow.

Unit-V 04 Hrs.

Reliability, Monitoring and Deployment:

Tactics for building reliable systems, monitoring and observability with tracing and sampling, CI/CD for agents, and training agents with human-in-the-loop feedback and structured evaluation (LLM-as-a-Judge); creating a Flow for automatic code review.

Unit-VI 02 Hrs.

Applying AI Agents in Business:

Agent applications across industries and functions, finding high-value use cases, and real-deployment case studies featuring Exa, Snyk, Weaviate, and AB InBev; the future of AI agents.

Web Resources:

https://www.coursera.org/learn/design-develop-and-deploy-multi-agent-systems-with-crewai