ANNA UNIVERSITY, CHENNAI

NON- AUTONOMOUS COLLEGES

AFFILIATED TO ANNA UNIVERSITY

M. E. COMPUTER SCIENCE AND ENGINEERING

(WITH SPECIALIZATION IN NETWORKS)

REGULATIONS 2025

PROGRAMME OUTCOMES (POs):

PO	Programme Outcomes
PO1	An ability to independently carry out research /investigation and development work to solve practical problems
PO2	An ability to write and present a substantial technical report/document.
PO3	Students should be able to demonstrate a degree of mastery over the area as per the specialization of the program. The mastery should be at a level higher than the requirements in the appropriate bachelor program

PROGRAMME SPECIFIC OUTCOMES:
PSO1: Advanced Network Architecture and Protocol Design: Design and implement scalable, efficient, and secure network architectures, leveraging advanced protocols, software-defined networking (SDN), and next-generation communication technologies.
PSO2: Research, Innovation and Best Security Practices: Conduct cutting-edge research in Network Security, Cyber Defense and Network Performance Optimization, develop innovative solutions and mitigate network-based threats and vulnerabilities using cryptographic techniques, intrusion detection systems, and zero-trust security models.

ANNA UNIVERSITY, CHENNAI

POSTGRADUATE CURRICULUM (NON-AUTONOMOUS AFFILIATED INSTITUTIONS)

Programme: M. E. Computer Science and Engineering (with Specialization in Networks)
Regulations: 2025

Abbreviations:

BS–Basic Science (Mathematics)

ES – Engineering Science (Programme Core (**PC**), Programme Elective (**PE**))

SL – Self Learning

L –Laboratory Course

T – Theory

LIT –Laboratory Integrated Theory

TCP –Total Contact Period(s)

Semester - I									
S. No.	**Course Code**	**Course Title**	**Type**	**Periods per week**			**TCP**	**Credits**	**Category**
				L	**T**	**P**			
1.	MA25C07	Advanced Mathematical Methods (CSIE)	T	3	1	0	4	4	BS
2.	CP25C01	Advanced Data Structures and Algorithms	LIT	3	0	4	7	5	ES (PC)
3.	NE25C01	Advanced Internet Technologies	T	3	0	0	3	3	ES (PC)
4.	CP25C03	Advanced Operating Systems	T	3	0	0	3	3	ES (PC)
5.	CP25C04	Advanced Compiler Design	T	3	0	0	3	3	ES (PC)
6.	NE25101	Technical Seminar	-	0	0	2	2	1	SD
Total Credits							**22**	**19**	

Semester - II									
S. No.	**Course Code**	**Course Title**	**Type**	**Periods per week**			**TCP**	**Credits**	**Category**
				L	**T**	**P**			
1.	NE25201	Advanced Network Security	LIT	3	0	2	5	4	ES (PC)
2.	CP25C05	Artificial Intelligence and Machine Learning	T	3	0	0	3	3	ES (PC)
3.	CP25C06	Cloud and Big Data Analytics	LIT	3	0	0	3	3	ES (PC)
4.	CP25C07	Quantum Computing	T	2	0	0	2	2	ES (PC)
5.		Programme Elective I	T	3	0	0	3	3	ES (PE)
6.		Industry Oriented Course I	-	1	0	0	1	1	SD
7.	NE25202	Industrial Training	-	-	-	-	-	2	SD
8.		Self-Learning Course	-	-	-	-	-	1	-
Total Credits							**17**	**19**	

Semester – III									
S. No.	**Course Code**	**Course Title**	**Type**	**Periods per week**			**TCP**	**Credits**	**Category**
				L	**T**	**P**			
1.		Programme Elective II	T	3	0	0	3	3	ES (PE)
2.		Programme Elective III	T	3	0	0	3	3	ES (PE)
3.		Programme Elective IV	T	3	0	0	3	3	ES (PE)
4.		Programme Elective V	T	3	0	0	3	3	ES (PE)
5.		Industry Oriented Course II	T	1	0	0	1	1	SD
6.	NE25301	Project Work I	-	0	0	12	12	6	SD
Total Credits							**25**	**19**	

Semester - IV									
S. No.	**Course Code**	**Course Title**	**Type**	**Periods per week**			**TCP**	**Credits**	**Category**
				L	**T**	**P**			
1.	NE25401	Project Work II	-	0	0	24	24	12	SD
Total Credits							**24**	**12**	

PROGRAMME ELECTIVE COURSES (PE)

S. No.	Course Code	Course Title	Periods			Total Contact Periods	Credits
			L	T	P		
1.	CP25C08	Advanced Software Testing and Quality Assurance	3	0	0	3	3
2.	CP25C09	Agile Methodologies	3	0	0	3	3
3.	CP25C10	Web of Things	3	0	0	3	3
4.	NE25001	Network Automation	3	0	0	3	3
5.	NE25002	Unified Communications	3	0	0	3	3
6.	CP25C12	Quantum Cryptography	3	0	0	3	3
7.	CP25C13	Quantum Machine Learning	3	0	0	3	3
8.	CP25C14	AI in IoT	3	0	0	3	3
9.	CP25C15	Web 3.0	3	0	0	3	3
10.	NE25003	Smart Grid Networks	3	0	0	3	3
11.	CP25C17	Edge and Fog Computing	3	0	0	3	3
12.	NE25004	Futured Network Systems	3	0	0	3	3
13.	CP25C18	Green Computing and Sustainability	3	0	0	3	3
14.	NE25005	Vehicular Networks	3	0	0	3	3
15.	CP25C19	Cognitive Computing	3	0	0	3	3
16.	NE25006	Intelligent Network and Communication	3	0	0	3	3
17.	NE25007	Wireless Engineering	3	0	0	3	3
18.	CP25C22	Blockchains Architecture and Design	3	0	0	3	3
19.	CP25C24	Vibe Coding	3	0	0	3	3
20.	CP25C25	Federated Learning	3	0	0	3	3

MA25C07	Advanced Mathematical Methods (CSIE)	L	T	P	C
		3	1	0	4

Course Objectives:

- Develop an in-depth understanding of advanced concepts in linear algebra, multivariate analysis, and number theory for computer science applications.
- Apply mathematical tools such as eigenvalue decomposition, SVD, and multivariate statistical methods to real-world computing and data-driven problems.
- Analyze and implement number-theoretic techniques for cryptography, security, and algorithmic problem-solving in computer science.

Linear Algebra: Vector spaces, norms, Inner Products, Eigenvalues using QR transformations, QR factorization, generalized eigenvectors, Canonical forms, singular value decomposition and applications, pseudo inverse, least square approximations.

Multivariate Analysis: Random vectors and matrices, Mean vectors and covariance matrices, Multivariate normal density and its properties, Principal components, Population principal components, Principal components from standardized variables.

Elementary Number Theory: The division algorithm, Divisibility and the Euclidean algorithm, The fundamental theorem of arithmetic, Modular arithmetic and basic properties of congruences; Principles of mathematical induction and well ordering principle. Primality Testing algorithms, Chinese Remainder Theorem, Quadratic Congruence.

Advanced Number Theory: Advanced Number Theory, Primality Testing algorithms, Chinese Remainder Theorem, Quadratic Congruence, Discrete Logarithm, Factorization Methods, Side Channel Attacks, Shannon Theory, Perfect Secrecy, Semantic Security.

Weightage: Continuous Assessment: 40%, End Semester Examinations: 60%.

Assessment Methodology: Assignments (15), Quiz (10), Virtual Demo (20), Flipped Class Room (10), Review of Gate and IES Questions (25), Project (20).

References:

1. Gilbert Strang, Linear Algebra and Its Applications, Cengage Learning.
2. Richard A. Johnson & Dean W. Wichern, Applied Multivariate Statistical Analysis, Pearson.
3. Neal Koblitz, A Course in Number Theory and Cryptography, Springer.
4. Victor Shoup, A Computational Introduction to Number Theory and Algebra, Cambridge University Press.

E-resources:

1. https://ocw.mit.edu/courses/18-06-linear-algebra
2. https://nptel.ac.in/courses/111105041
3. https://crypto.stanford.edu/pbc/notes/numbertheory

CP25C01	Advanced Data Structures and Algorithms	L	T	P	C
		3	0	4	5

Course Objectives:

- To explore advanced linear, tree, and graph data structures and their applications.
- To design efficient algorithms using appropriate algorithmic paradigms.
- To evaluate computational complexity and identify tractable vs. intractable problems.

Linear Data Structures and Memory Optimization: Advanced arrays: Sparse arrays, dynamic arrays, cache-aware structures, Linked lists: Skip lists, unrolled linked lists, XOR linked lists, Stacks and Queues: Priority queues, double-ended queues, circular buffers, Hashing: Perfect hashing, cuckoo hashing, extendible hashing.

Practical:

- Implement skip lists and measure performance compared with balanced BST.
- Experiment with cache-aware data structures and analyze memory utilization.

Advanced Tree Data Structures: Balanced Trees: AVL, Red-Black Trees, Splay Trees, Treaps, Multi-way Trees: B-Trees, B+ Trees, R-Trees, Segment Trees, Fenwick Trees, Suffix Trees and Tries for string processing, Applications in indexing, text retrieval, computational geometry.

Practical:

- Implement B+ tree for database indexing use-case.
- Design a suffix tree-based algorithm for DNA sequence matching.

Graph Data Structures and Algorithms: Representation: Adjacency list/matrix, incidence matrix, compressed storage, Traversals: DFS, BFS with applications, Shortest Path Algorithms: Dijkstra, Bellman-Ford, Floyd-Warshall, Johnson's algorithm, Minimum Spanning Trees: Prim's, Kruskal's, Borůvka's algorithm, Network Flow Algorithms: Ford-Fulkerson, Edmonds-Karp, Push-Relabel.

Practical:

- Implement Johnson's algorithm for sparse graph shortest paths.
- Demonstration of Maximum flow in traffic or network routing simulation.

Algorithm Design and Paradigms: Divide and Conquer: Karatsuba's multiplication, Strassen's algorithm, Greedy Methods: Huffman coding, interval scheduling, set cover approximation, Dynamic Programming: Matrix chain multiplication, Floyd-Warshall, knapsack variants, Backtracking and Branch-and-Bound, Randomized Algorithms and Probabilistic Analysis.

Practical:

- Implement Strassen's algorithm and compare with naive matrix multiplication.
- Develop a randomized algorithm for primality testing (Miller–Rabin).

Computational Complexity and Approximation Algorithms: Complexity Classes: P, NP, NP-Complete, NP-Hard, Reductions: Polynomial-time reductions, Cook-Levin theorem (overview), Approximation Algorithms: Vertex cover, set cover, TSP, k-center problem, Heuristic Algorithms: Local search, simulated annealing, genetic algorithms.

Practical:

- Implement approximation algorithm for vertex cover.
- Complexity analysis of a chosen NP-hard problem and implement a heuristic.

Advanced Topics and Emerging Trends: Randomized Algorithms – Monte Carlo Algorithms, Parallel and Distributed Algorithms – PRAM Model, Divide and Conquer in Parallel, Load Balancing, Streaming Algorithms – Data Stream Models, Sketching and Sampling, Frequency Moments, Advanced String Matching – Suffix Trees, Suffix Arrays, Pattern Matching in Linear Time.

Practical:

- Implement randomized and streaming algorithms on real-world datasets.
- Design of parallel and distributed algorithms.

Weightage: Continuous Assessment: 50%, End Semester Examinations: 50%

Assessment Methodology: Assignments (15), Quiz (10), Virtual Demo (20), Flipped Class Room (10), Review of Gate and IES Questions (25), Project (20)

References:

1. Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2009). Introduction to algorithms. MIT Press.
2. La Rocca, M. (2021). Advanced algorithms and data structures. Manning Publications.
3. Goodrich, M. T., Tamassia, R., & Mount, D. M. (2011). Data structures and algorithms in C++. John Wiley & Sons, Inc.
4. Weiss, M. A. (2014). Data structures and algorithm analysis in C++. Pearson Education.
5. Drozdek, A. (2013). Data structures and algorithms in C++. Cengage Publications.

E-resources:

1. https://www.theiotacademy.co/blog/data-structures-and-algorithms-in-c/
2. https://github.com/afrid18/Data_structures_and_algorithms_in_cpp
3. https://www.udemy.com/course/introduction-to-algorithms-and-data-structures-in-c/?srsltid=AfmBOorEZlkgV7QzaEh6IqzAaKLjC-IpFU1NGgWFoHMLhOos-uDVKjCK

	Description of CO	PO	PSO
CO1	Describe data structures and implement algorithmic solutions for complex computational problems.	--	--

	Description of CO	PO	PSO
CO2	Analyze the time complexity and efficiency of algorithms for various computing problems.	PO1(3)	PSO1(3)
CO3	Evaluate algorithmic techniques and data structures to determine their suitability for different applications.	PO3(2)	PSO2(2)
CO4	Design optimized solutions for real-world problems using appropriate algorithms and data structures.	PO2(1)	PSO1(3)

NE25C01	Advanced Internet Technologies	L	T	P	C
		3	0	0	3

Course Objective:

- To provide knowledge about modern Internet technologies and applications.
- To enable students to design, implement, and evaluate web-based and distributed applications.
- To develop skills in applying security mechanisms and scalable architectures for real-world Internet-based solutions.

Internet and Web Design: Understanding HTML and XHTML Connections, Understanding Cascading Stylesheets, Understanding JavaScript, Working with Fonts, Texts and Lists, Tables, Internal and External Links, Working with colours and Multimedia.

Activity: Design a responsive webpage using HTML5 and CSS3.

CSS and Dynamic websites: Understanding CSS Box Model and positioning, Navigations, creating fixed and liquid layouts, understanding Dynamic websites.

Activity: Demonstrate RESTful APIs using HTTP methods

Java Script Programming: Working with DOM, Java Script variables, strings and arrays, Functions and objects, Controls and loops, Responding to events, windows and frames using Un obtrusive Java script, using third party libararies, AJAX.

Activity: Build a dynamic SPA using JavaScript.

Emerging Internet Technologies – I: Cookies and sessions, Middleware, Sending Email, production Concerns, Persistence, Routing, Rets APIs and JSON.

Activity: Demonstration using REST APIs and JSON.

Emerging Internet Technologies-II: Static Content, Implementing MVC in express, Security Integrating third party APIs, Debugging, Going live, Maintenance.

Activity: Design of simple login system with JWT authentication

Flask web development in Internet Technologies: Basic Application structure, Templates, Databases, Email, Large Application structure, Application Programming Interfaces, Testing, Deployment.

Activity: Build a simple web application using Flask APIs.

Weightage: Continuous Assessment: 40%, End Semester Examinations: 60%

Assessment Methodology: Assignments (15), Quiz (10), Virtual Demo (20), Flipped Class Room (10), Review of Gate and IES Questions (25), Project (20).

References:

1. Brown, E. (2023). Web development with Node and Express: Leveraging the JavaScript stack (2nd ed.). O'Reilly Media.
2. Wargo, J. M. (2021). Learning progressive web apps: Building modern web apps using service workers and web APIs. Addison-Wesley Professional.
3. Meloni, J. C. (2023). HTML, CSS, and JavaScript all in one (4th ed.). Sams Publishing.
4. Grinberg, M. (2018). Flask web development: Developing web applications with Python (2nd ed.). O'Reilly Media.
5. Vemula, R. (2023). Real-time web application development. Packt Publishing.

	Description of CO	**PO**	**PSO**
CO1	Explain web technologies to create responsive and interactive web pages.	--	--
CO2	Develop dynamic web applications and third-party libraries for enhanced user experience.	PO1(3)	PSO1(3)
CO3	Integrate middleware and IoT components to build scalable and secure internet-based applications.	PO3(2)	PSO2(2)
CO4	Design and deploy full-stack applications using Flask.	PO2(1)	PSO1(3)

CP25C03	Advanced Operating Systems	L	T	P	C
		3	0	0	3

Course Objectives:

- To analyze the architectures and design issues of advanced operating systems.
- To develop the model for process synchronization and recovery in complex environments.
- To evaluate algorithms for distributed coordination, resource management, fault tolerance, and security.

Advanced Process and Thread Management: Multithreading models, thread pools, context switching, Synchronization issues and solutions: semaphores, monitors, lock-free data structures, CPU scheduling in multi-core systems

Activity: CPU scheduler simulation for multicore systems.

Memory and Resource Management in Modern OS: Virtual memory, demand paging, page replacement policies-Huge pages, NUMA-aware memory management-Resource allocation in cloud-native environments

Activity: Simulate demand paging and page replacement algorithms.

Virtualization and Containerization: Hypervisors (Type I & II), KVM, QEMU, Xen-Containers: Docker, LXC, systemd-nspawn-OS-level virtualization and namespaces

Activity: Deploy and configure Docker containers with various images.

Distributed Operating Systems and File Systems: Distributed scheduling, communication, and synchronization-Distributed file systems: NFS, GFS, HDFS-Transparency issues and fault tolerance

Activity: Simulate distributed process synchronization.

Security and Trust in Operating Systems: Access control models: DAC, MAC, RBAC-OS hardening techniques, sandboxing, SELinux, AppArmor-Secure boot, rootkit detection, trusted execution environments

Activity: Implement Role-Based Access Control (RBAC) using Linux user and group permissions.

Real-Time and Embedded Operating Systems: Real-time scheduling algorithms (EDF, RM)-POSIX RT extensions, RTOS architecture-TinyOS, FreeRTOS case studies

Activity: Analyze FreeRTOS task scheduling behavior.

Edge and Cloud OS: Future Paradigms: Serverless OS, unikernels, lightweight OS for edge computing-Mobile OS internals (Android, iOS)-OS for quantum and neuromorphic computing (intro)

Activity: Analyze Android's system architecture using emulator tools.

Weightage: Continuous Assessment: 40%, End Semester Examinations: 60%

Assessment Methodology: Assignments (15), Quiz (10), Virtual Demo (20), Flipped Class Room (10), Review of Gate and IES Questions (25), Project (20).

References:

1. Tanenbaum, A. S., & Bos, H. (2023). Modern operating systems. Pearson.
2. Buyya, R., et al. (2022). Content delivery networks and emerging operating systems. Springer.
3. Silberschatz, A., Galvin, P. B., & Gagne, G. (2022). Operating system concepts. Wiley.
4. Anderson, T., & Dahlin, M. (2021). Operating systems: Principles and practice. Recursive Books.
5. Arpaci-Dusseau, R. H., & Arpaci-Dusseau, A. C. (2020). Operating systems: Three easy pieces.

E-Resources:

1. Prof. Smruti Ranjan Sarangi, “Advanced Distributed Systems”, IIT Delhi, NPTEL, https://onlinecourses.nptel.ac.in/noc22_cs80/preview
2. Prof. Rajiv Misra, “Cloud Computing and Distributed Systems”, IIT Patna, NPTEL**,** https://nptel.ac.in/courses/106104182

	Description of CO	**PO**	**PSO**
CO1	Describe operating system concepts for memory and resource management.	--	--
CO2	Analyse virtualization and distributed OS mechanisms for scalability and performance.	PO1(3)	PSO1(3)
CO3	Evaluate OS security and resource handling strategies in diverse environments.	PO3(2)	PSO2(2)
CO4	Design innovative OS solutions using modern tools and techniques.	PO2(1)	PSO1(3)

CP25C04	Advanced Compiler Design	L	T	P	C
		3	0	0	3

Course Objectives:

- To analyze the theory and principles of modern compiler design and advanced optimization techniques.
- To design and implement efficient front-end and back-end compiler components for programming languages.
- To evaluate code optimization strategies and runtime environment management in contemporary architectures.

Intermediate Representations and Control Flow Analysis: Static single assignment (SSA) form- Context-Free Grammer (CFG) construction-dominance relations-Intermediate Representation (IR) design for functional and imperative languages-Static single assignment and def-use chains

Activities:

1. Convert source code to SSA form using LLVM IR.
2. Visualize control flow graphs from SSA using LLVM tools.

Program Analysis and Transformations: Data flow analysis, live variable analysis, reaching definitions, Alias analysis and dependence analysis-Loop optimizations and transformations

Activities:

1. Perform loop unrolling and strength reduction.
2. Conduct live variable analysis and visualize data flow graphs.

Advanced Optimizations and Polyhedral Compilation: Polyhedral model for loop nests-Tiling, skewing, fusion, and vectorization-Profile-guided and feedback-directed optimizations

Activities:

1. Implement loop tiling and loop skewing on a matrix multiplication program.
2. Analyze the effect on loop-intensive code with LLVM optimization flags.

Just-in-Time (JIT) and Runtime Compilation: JIT compilation models: tracing, method-based-GraalVM architecture, Java HotSpot internals-LLVM JIT and dynamic language support

Activities:

1. Develop a basic JIT-enabled interpreter with LLVM or GraalVM.
2. Implement dynamic dispatch using LLVM JIT API.

Machine Learning in Compiler Design: ML for phase ordering, auto-tuning, and IR prediction-Reinforcement learning for optimization passes-Dataset creation and benchmarking for compiler ML

Activities:

1. Train an ML model to predict optimization passes.
2. Use reinforcement learning for pass selection in toy compiler.

Domain-Specific Languages (DSLs) and Compiler Extensions: Designing DSLs for AI/ML, DSP, graphics-Code generation for custom accelerators-Integration with TensorFlow XLA and Halide

Activities:

1. Design and test a simple DSL grammar using ANTLR.
2. Integrate a DSL with TensorFlow XLA or Halide.

Security, Verification, and Future Trends: Secure compilation and type-safe intermediate representations-Compiler fuzzing and formal verification (e.g., CompCert)-Quantum compilers, multi-target compilers, and neuromorphic systems

Activities:

1. Use CompCert to verify compilation of simple programs.
2. Apply compiler fuzzing using tools like libFuzzer.

Weightage: Continuous Assessment: 40%, End Semester Examinations: 60%

Assessment Methodology: Assignments (15), Quiz (10), Virtual Demo (20), Flipped Class Room (10), Review of Gate and IES Questions (25), Project (20).

References:

1. Cooper, K. D., & Torczon, L. (2023). Engineering a compiler. Morgan Kaufmann.
2. Grune, D., Bal, H. E., Jacobs, C. J. H., & Langendoen, K. G. (2012). Modern compiler design (2nd ed.). Springer.
3. Aho, A. V., Lam, M. S., Sethi, R., & Ullman, J. D. (2006). Compilers: Principles, techniques, and tools (2nd ed.). Pearson.
4. Völter, M. (2013). DSL engineering: Designing, implementing and using domain-specific languages. dslbook.org.
5. Sarda, S., & Pandey, M. (2015). LLVM essentials. Packt Publishing.

E-Resources:

1. Prof. AmeyKarkare, IIT Kanpur, “Advanced Compiler Optimizations” Link: https://www.cse.iitk.ac.in/users/karkare/Courses/cs738/
2. Prof. Santanu Chattopadhyay, “Compiler Design”, IIT Kharagpur Link:” https://onlinecourses.nptel.ac.in/noc21_cs07/preview”

	Description of CO	PO	PSO
CO1	Explain intermediate control flow techniques in compiler design.	--	--
CO2	Apply program analysis techniques and advanced optimizations for design of compilers.	PO1(3)	PSO1(3)
CO3	Develop compiler features and machine learning techniques for optimization.	PO3(2)	PSO2(2)
CO4	Evaluate secure compilation strategies for quantum and multi-target compilation.	PO2(1)	PSO1(3)

NE25201	Advanced Network Security	L	T	P	C
		3	0	2	4

Course Objective:

This course enables students to explore advanced concepts in network security, including threat modelling, secure protocol design, access control, intrusion detection, and incident response. It equips learners with the tools and knowledge necessary to defend complex systems and networks against evolving cyber threats while engaging with cutting-edge research and practices.

Network Threats and Vulnerabilities

Threat landscape and attack surface – Advanced Persistent Threats (APTs), zero-day vulnerabilities – Scanning, reconnaissance, and enumeration tools – Case study: SolarWinds, Colonial Pipeline breach

Practical

1. Perform network reconnaissance with Nmap and Wireshark
2. Analyze real-world packet captures and identify attack signatures

Secure Communication Protocols – I

IPsec, SSL/TLS internals and configuration – VPN technologies: IKEv2, WireGuard

Practical

1. Configure IPsec tunnels and analyze encrypted traffic
2. Simulate man-in-the-middle attacks and demonstrate TLS protection

Secure Communication Protocols – II

Secure routing: OSPF, BGP security – WPA3 and wireless EAP protocols

Practical

1. Configure Snort or Suricata and analyze detection results
2. Simulate brute-force and DoS attacks in a controlled environment

Intrusion Detection, Prevention & Response

IDS/IPS architectures (Snort, Suricata) – Anomaly detection, log correlation, SIEM basics – Incident response lifecycle and forensics – Real-time threat monitoring with open-source tools

Practical

1. Set up VLAN segmentation and firewall policies
2. Configure NAC with authentication and access rules

Access Control and Network Segmentation

AAA protocols: RADIUS, TACACS+ – Identity-based access and 802.1X – VLAN, NAC, firewall configuration – Software Defined Networking and Zero Trust Architecture

Practical

1. Use SIEM (Wazuh/Graylog) for threat visualization
2. Perform network forensics with packet analysis tools

Research Trends in Network Security – I and II

Security in SDN, NFV, cloud, IoT, and 5G – Machine learning for threat detection - Quantum-safe networking and blockchain-based models – Emerging research challenges and ethical implications

Practical

1. Develop a Project to simulate a threat model

Weightage: Continuous Assessment: 40%, End Semester Examinations: 60%

Assessment Methodology: Assignments (15), Quiz (10), Virtual Demo (20), Flipped Class Room (10), Review of Gate and IES Questions (25), Project (20).

References

1. William Stallings, Network Security Essentials: Applications and Standards, 7th Ed., Pearson, 2023.
2. Charlie Kaufman, Radia Perlman, Mike Speciner, Network Security: Private Communication in a Public World, 3rd Ed., Pearson, 2022.
3. Chris Sanders, Applied Network Security Monitoring, Syngress/Elsevier, 2017.
4. Seymour Bosworth, Computer Security Handbook, 6th Ed., Wiley, 2020.
5. Richard Bejtlich, The Practice of Network Security Monitoring, No Starch Press, 2014.

E-Resources

- NPTEL Course: Network Security and Cryptography
- **Udemy** Courses on:
  - Advanced Network Security
  - Firewalls, VPNs, IDS/IPS configuration
- Zero Trust Architecture - NIST 800-207
- IEEE Xplore Digital Library – For access to top research articles and papers

CO	Description of CO	PO	PSO
CO1	Describe the fundamental concepts, principles, and protocols of advanced network security for protecting information and communication systems.	--	--
CO2	**Analyze** network security architectures, cryptographic techniques, and threat models to understand vulnerabilities and protective measures.	PO1 (3)	PSO1 (3)
CO3	**Evaluate** security policies, intrusion detection systems, and mitigation strategies to assess their effectiveness in defending against cyber threats.	PO3 (2)	PSO2 (2)
CO4	**Design** secure network solutions by selecting appropriate cryptographic methods, security protocols, and monitoring strategies for real-world applications.	PO2 (1)	PSO1 (3)

CP25C05	Artificial Intelligence and Machine Learning	L	T	P	C
		3	0	0	3

Course Objectives:

- To understand the fundamentals of Artificial Intelligence and its core problem-solving techniques using intelligent agents, search strategies, and logic-based reasoning.
- To comprehend the theoretical foundations and algorithmic frameworks of machine learning, with an emphasis on supervised and unsupervised learning methodologies.

Problem Solving and Knowledge Representation: Solving problems by searching – adversarial search – constraint satisfaction problems – Logical agents – Propositional logic – First order logic – Forward chaining – Backward chaining – Ontological Representations and Reasoning Systems.

Activities: Develop a simple Intelligent Agent that navigates a grid world using search algorithms.

Uncertain Knowledge and Reasoning: Overview of uncertainty and basic probability – Baye's rule – Bayesian networks – Hidden Markov models – Kalman filters – Utility Functions – Decision Networks – Sequential decision problems – Game theory

Activities: Develop a simple Intelligent Agent that makes decisions in adversarial conditions

Machine Learning Introduction: Types of learning – Hypothesis space – Inductive bias – Evaluation – Cross validation – Bias variance – Bias Variance Trade off.

Activities: Develop a simple Intelligent Agent that uses logical reasoning to infer hidden facts or constraints

Supervised Learning: Linear Regression – Logistic Regression - Decision trees: Classification and regression trees – Neural networks, multilayer perceptron – Support vector machines: linear and non-linear kernel functions – K nearest neighbours – Ensemble learning: bagging and boosting.

Activities: Behaviour Modelling, Disease Diagnosis and any other similar ones suitable for supervised learning

Unsupervised and Probabilistic Learning: Clustering: partition, hierarchical and density based clustering – Self organizing maps – Expectation maximization – Gaussian mixture models – Principal component analysis – Bayesian learning: Bayes optimal classifier, Naïve Bayes classifier, Bayesian belief networks.

Activities: Customer Segmentation, Image Processing and any other similar ones suitable for unsupervised and probabilistic learning

Weightage: Continuous Assessment: 40%, End Semester Theory Examination: 60%

Assessment Methodology: Assignments (30), Quiz (10), Virtual demonstration (25), Flipped Classroom (10), Review of GATE & IES questions (25).

References:

1. Stuart Russell and Peter Norvig Artificial Intelligence - A Modern Approach, Prentice Hall, 3rd edition, 2011.
2. Elaine Rich, Kevin Knight and Shiv Shankar B. Nair, Artificial Intelligence, 3rd edition, Tata McGraw Hill, 2009.
3. Tom Mitchell, “Machine Learning”, McGraw Hill, 3rd Edition, 1997.
4. Ethem Alpaydin,"Introduction to Machine Learning”, MIT Press, Prentice Hall of India, Third Edition 2014.
5. Mehryar Mohri, Afshin Rostamizadeh, Ameet Talwalkar "Foundations of Machine Learning”, MIT Press, 2012.

E-resources:

1. Nptel Course, An Introduction to Artificial Intelligence by Prof Mausam
2. Nptel Course, Machine Learning,By Prof. Carl Gustaf Jansson

CO	Description of CO	PO	PSO
CO1	Describe the fundamental concepts, scope, and applications of artificial intelligence and machine learning in solving real-world problems.	--	--
CO2	Analyze various artificial intelligence and machine learning models to understand their behavior, performance, and suitability for different problem domains.	PO1 (3)	PSO1 (3)
CO3	Evaluate machine learning algorithms and artificial intelligence techniques based on accuracy, efficiency, and applicability to specific use cases.	PO3 (2)	PSO2 (2)
CO4	Design intelligent systems by selecting and integrating appropriate machine learning models and AI techniques to address complex real-world applications.	PO2 (1)	PSO1 (3)

CP25C06	**Cloud and Big Data Analytics**	L	T	P	C
		3	0	0	3

Course Objectives:

- To understand cloud computing paradigms and service models.
- To study big data characteristics, technologies and storage.
- To explore Hadoop ecosystem and processing tools.
- To analyze, store, and visualize large-scale data.
- To develop and deploy cloud-based big data applications.

Cloud Computing: Introduction to Cloud Computing - Service Models: IaaS, PaaS, SaaS - Deployment Models: Public, Private, Hybrid, Community - Virtualization: Hypervisors, VM provisioning - Cloud providers: AWS, Azure, Google Cloud overview

Activities: Create a free-tier AWS account and explore EC2, S3.

Big Data Ecosystem & Hadoop: Big Data characteristics (Volume, Velocity, Variety, Veracity, Value) - Challenges in Big Data - Hadoop Architecture and HDFS - MapReduce Programming Model –YARN

Activities: Watch Hadoop tutorial and submit a summary report.

Data Storage & Processing Techniques: NoSQL Databases: MongoDB, Cassandra, HBase - Data Ingestion: Sqoop, Flume - Hive, Pig – Architecture and Querying - Data Cleaning and Preparation.

Activities: HiveQL assignments on datasets from Kaggle.

Advanced Big Data Analytics: Introduction to Apache Spark - Spark vs. MapReduce - Spark SQL, Data Frames, RDDs- Machine Learning using Spark MLlib - Real-time processing: Kafka and Spark Streaming

Activities: Mini-project using PySpark on COVID or Traffic dataset.

Cloud Integration and Security: Data on Cloud – Storage and Compute Services - Data Security and Privacy in Cloud - Cloud-based Data Analytics Use Cases - Cloud-Native vs Traditional Applications - Case Study: Smart Cities / Healthcare Analytics

Activities: Create a cloud deployment report using Azure/AWS.

Weightage: Continuous Assessment:40%, End Semester Theory Examination: 60%

Assessment Methodology: Assignments (30), Quiz (10), Virtual demonstration (25), Flipped Classroom (10), Review of GATE & IES questions (25).

References:

1. Rajkumar Buyya, James Broberg, Andrzej Goscinski, Cloud Computing: Principles and Paradigms, Wiley.
2. Tom White, Hadoop: The Definitive Guide, O'Reilly Media.
3. Vignesh Prajapati, Big Data Analytics with R and Hadoop, Packt Publishing.
4. Kleppmann, Martin, Designing Data-Intensive Applications, O'Reilly.
5. Alan Gates, Programming Pig, O'Reilly.
6. Jure Leskovec et al., Mining of Massive Datasets, Cambridge University Press.

E-resources:

https://www.nasa.gov/smallsat-institute/sst-soa/thermal-control/

CO	Description of CO	PO	PSO
CO1	Describe the fundamental concepts, architectures, and service models of cloud computing and big data analytics used in modern data-driven environments.	--	--
CO2	Analyze cloud-based big data processing frameworks and analytics techniques to understand their performance, scalability, and resource utilization.	PO1 (3)	PSO1 (3)
CO3	Evaluate big data analytics tools, cloud platforms, and data processing models for their effectiveness in handling large-scale and complex datasets.	PO3 (2)	PSO2 (2)
CO4	Design scalable cloud-based big data analytics solutions by selecting appropriate architectures, platforms, and analytical methods for real-world applications.	PO2 (1)	PSO1 (3)

CP25C07	**Quantum Computing**	L	T	P	C
		2	0	0	2

Course Objective:

1. To provide a mathematical foundation for Quantum Computing and provide the basics of working
2. To interpret the various aspects and applications of quantum computing.
3. To examine the factors that affect Quantum computation

Physical properties and mathematical foundations

Double Slit Experiment; Light: Particle Vs Wave; Heisenberg Uncertainty Principle.Vector spaces – basis; Inner product; Outer product; Tensor product; Linear operators

Activities:

- Simulate the experiment using an interactive virtual lab
- Construct simple operators and visualize action on vectors

Quantum computing postulates and gates

Review of postulates, Bloch sphere, Single qubit states and gates, superposition; Two Qubit States and Gates - Bell States, Entanglement, CNOT gate, Phase oracles, Pauli Gates.

Activities:

- Group quiz to match postulates to physical implications
- Visualization with Bloch sphere simulators

Quantum computing circuits

Dirac’s notation for quantum computing, Computational Basis, Orthonormality, Hadamard and Phase Gates- building quantum circuits

Activities:

- Use IBM Q Composer to build and simulate custom circuits
- Hands-on: Apply X, H, Z gates and observe results on simulators

Fundamental Quantum Algorithms

Deutsch–Josza Algorithm, Grover search algorithm: Problem definition, Amplitude amplification, Grover oracle, diffuser, multiple solutions in the search space

Activities:

- Construct DJ circuit for a 3-bit input function
- Simulation of Grover's algorithm with multiple marked elements

Programming on a real quantum computer

Coding a real time quantum computer via IBMQ to carry out basic quantum measurement and state analysis.

Activities:

- Connect Qiskit with IBMQ using personal API token
- Hands-on: Create 1- and 2-qubit circuits using Hadamard, X, Z, and measurement gates
- Compare real and simulated results and Observe impact of quantum noise

Text Books:

1. Chuck Easttom, "Quantum Computing Fundamentals", 1st edition, Published by Addison-Wesley Professional (June 1st 2021)
2. Qiskit TextBook - https://qiskit.org/textbook/preface.html (2022)

References:

1. Kasirajan, Venkateswaran. Fundamentals of quantum computing. Springer International Publishing, 2021.
2. Chris Bernhardt, Quantum Computing for Everyone, The MIT Press, Cambridge, 2020
3. Nielsen, Michael A., and Isaac L. Chuang, "Quantum Computation and Quantum Information" Cambridge University Press (5 April 2013)

Weightage: Continuous Assessment:40%, End Semester Theory Examination: 60%

Assessment Methodology: Assignments (30), Quiz (10), Virtual demonstration (25), Flipped Classroom (10), Review of GATE & IES questions (25).

CO	Description of CO	PO	PSO
CO1	Describe the fundamental principles, postulates, and computational models of quantum computing and their significance in next-generation computing systems.	--	--
CO2	Analyze quantum algorithms and quantum circuit models to understand their computational advantages, limitations, and performance characteristics.	PO1 (3)	PSO1 (3)

CO3	Evaluate quantum computing paradigms, error correction techniques, and hardware technologies for their suitability in solving complex computational problems.	PO3 (2)	PSO2 (2)
CO4	Design quantum circuits and algorithmic solutions by selecting appropriate quantum gates, qubit architectures, and computational models for real-world problem scenarios.	PO2 (1)	PSO1 (3)

CP25C08	Advanced Software Testing and Quality Assurance	L	T	P	C
		3	0	0	3

Course Objectives:

1. Provide an in-depth understanding of advanced software testing techniques and their role in quality assurance.
2. Explore automated testing frameworks, tools, and methods for test design and test process improvement.
3. Enable students to design test plans, develop test cases, and assess test results critically.

Foundations and Test Lifecycle: Evolution of software testing in agile and DevOps ecosystems -Software Test Life Cycle (STLC) phases: Requirements analysis to test Closure - Quality attributes: reliability, maintainability, testability - Introduction to testing types and classifications.

Activity: Mapping STLC to Agile/DevOps pipelines.

Types of Testing and Risk-Based Approaches: Types of testing: Unit, Integration, System, Acceptance, Regression, Smoke, and Sanity - Risk-based testing and test Prioritization - Aligning testing strategies with product risk and delivery goals

Activity: Test strategy design for risk-based scenarios

Advanced Test Design Techniques: Black-box and white-box test strategies: BVA, ECP, decision tables, control/data flow - State transition and use-case testing - Model-based testing, combinatorial testing, and cause-effect graphing - Mutation testing, fault injection, fuzz testing - Test case minimization and prioritization techniques

Activity: Create test cases for a real-world problem using model-based design

Test Automation and Continuous Testing: Design of reusable and maintainable automation frameworks - Open-source tools: Selenium, TestNG, JUnit, Robot Framework, Cypress - Performance testing: JMeter and basic LoadRunner Concepts - Integration with Jenkins and GitHub for continuous testing - BDD and TDD using Cucumber

Activity: Implement a hybrid automation framework for a web-based application

Software Quality Assurance Frameworks and Metrics: Quality planning, control, assurance, and improvement - SQA tools and activities: Reviews, audits, inspections - Quality models: McCall, Boehm, ISO 9126 / ISO 25010 -Metrics for testing: test coverage, defect density, MTBF, test effort -Cost of quality and defect leakage

Activity: Analyze QA strategy in enterprise software (e.g., ERP or banking)

Test Process Improvement and Research Trends: Test maturity models: TMMi, CMMI-Dev (QA perspective) - Test process audits, benchmarking, and reviews - Empirical methods in testing and defect prediction models -AI/ML-based testing approaches: self-healing tests, anomaly detection -Testing trends: shift-left/right testing, visual and mobile testing.

Activity: Design a test improvement roadmap based on TMMi for a sample company

Weightage: Continuous Assessment: 40%, End Semester Theory Examinations: 60%

Assessment Methodology: Assignments (30), Quiz (10), Virtual Demo (25), Flipped Class Room (10), Review of Gate and IES Questions (25)

References:

1. Glenford J. Myers, The Art of Software Testing, 3rd edition, Wiley
2. Paul C. Jorgensen, Software Testing: A Craftsman's Approach, 4th edition, CRC Press
3. Ron Patton, Software Testing, Second Edition, Sams Publishing, Pearson Education, 2007
4. Aditya Mathur, Foundations of Software Testing, 2nd edition, Pearson, 2013
5. Pressman & Maxim, Software Engineering: A Practitioner’s Approach, McGraw-Hill.

E-Resources:

1. MIT OpenCourseWare (Software Testing)
2. NPTEL (Sftware Testing by Prof. Nandini Prasad, IIT Kharagpur)

CO	Description of CO	PO	PSO
CO1	Describe the fundamental concepts, principles, and processes involved in advanced software testing and quality assurance for reliable software development.	--	--
CO2	**Analyze** software systems and testing strategies to identify defects, assess risks, and ensure compliance with quality standards.	PO1 (3)	PSO1 (3)
CO3	**Evaluate** software testing techniques, quality models, and assurance frameworks to determine their effectiveness in diverse development environments.	PO3 (2)	PSO2 (2)
CO4	**Design** comprehensive testing and quality assurance strategies by selecting appropriate testing tools, automation frameworks, and quality metrics for real-world software projects.	PO2 (1)	PSO1 (3)

CP25C09	Agile Methodologies	L	T	P	C
		3	0	0	3

Course Objectives:

- Introduce students to the principles, values, and practices of Agile software development.
- Provide hands-on experience with Agile tools and techniques used in industry.
- Develop the ability to plan, manage, and deliver software in Agile environments.

Evolution of Software Development and Agile Foundations: Evolution of Software Development Methodologies - Challenges in traditional models that led to Agile - The Agile Manifesto - Agile Values and Principles

Activity: Agile vs Waterfall in large-scale government or startup projects (e.g., FBI Sentinel, Spotify)

Agile Mindset and Cultural Transformation: Agile Mindset in Individuals and Teams - Characteristics of Agile teams and leaders - **Lean Thinking in Software Development -** Cultural Shift from Traditional to Agile Thinking - Applying Lean in Agile - Agile vs Traditional Methodologies

Activity: Reflective write-up on Agile mind set adoption challenges in traditional teams

Agile Process Models and Frameworks: Scrum: Roles, artifacts, events - Extreme Programming (XP): Pair programming, test-first, refactoring - Kanban: Flow, work-in-progress (WIP), pull systems - SAFe (Scaled Agile Framework) and Disciplined Agile Delivery (DAD) - Feature-Driven Development (FDD), Crystal, LeSS, Nexus

Activity: Team simulation using Scrum ceremonies (sprint planning, stand-ups, retrospectives)

Agile Project Management and Planning: User stories and story mapping - Estimation techniques: Planning poker, T-shirt sizing - Product backlog grooming and sprint planning - Agile metrics: Velocity, burndown chart, cumulative flow diagram (CFD) - Risk management in Agile

Activity: Tool Demo**:** Jira, Trello, Azure DevOps

Agile Engineering Practices: Test-Driven Development (TDD) and Behaviour Driven Development (BDD) - Continuous Integration and Continuous Delivery (CI/CD) - DevOps principles and its relationship with Agile - Refactoring, code smells, clean code - Design patterns in Agile context

Activity: Lab Component: Automate testing and CI/CD for a sample app

Agile at Scale and Industry Case Studies: Agile scaling frameworks and challenges - Distributed Agile teams and remote collaboration - Agile in non-software contexts (e.g., education, HR, manufacturing) - Agile contracting and governance - Agile case studies from Google, Spotify, IBM, and startups

Activity: Project based experiential learning: End-to-end Agile project using Scrum/Kanban and CI/CD pipeline

References:

1. Ken Schwaber & Jeff Sutherland, Software in 30 Days: How Agile Managers Beat the Odds, Wiley
2. Mike Cohn, User Stories Applied: For Agile Software Development, Addison-Wesley
3. Craig Larman & Bas Vodde, Scaling Lean & Agile Development, Addison-Wesley
4. Robert C. Martin, Clean Code: A Handbook of Agile Software Craftsmanship, Pearson

E-Resources:

1. NPTEL*:* Agile Software Development – Prof. Rajib Mall (IIT Kharagpur)
2. MIT Open Course Ware Agile *content*

Weightage: Continuous Assessment: 40%, End Semester Theory Examinations: 60%

Assessment Methodology: Assignments (30), Quiz (10), Virtual Demo (25), Flipped Class Room (10), Review of Gate and IES Questions (25)

CO	Description of CO	PO	PSO
CO1	Describe the fundamental principles, values, and frameworks of agile methodologies and their role in modern software development practices.	--	--
CO2	**Analyze** agile processes and practices to understand their effectiveness in managing requirements, collaboration, and iterative software development.	PO1 (3)	PSO1 (3)
CO3	**Evaluate** agile models, project management techniques, and team practices to assess their suitability for different organizational and project contexts.	PO3 (2)	PSO2 (2)
CO4	**Design** agile-based project workflows by selecting appropriate methodologies, roles, and practices to deliver high-quality software solutions.	PO2 (1)	PSO1 (3)

CP25C10	Web of Things	L	T	P	C
		3	0	0	3

Course Objective:

To equip students with the knowledge and skills to integrate Internet of Things (IoT) devices with web technologies using the Web of Things (WoT) framework. This course enables learners to design interoperable, secure, and scalable smart applications by leveraging standardized WoT architectures, Thing Descriptions, and scripting interfaces.

Introduction to Web of Things: Evolution: From IoT to WoT, Architecture of WoT: Thing Description, Binding, Servient, Protocol Bindings, Web protocols overview (HTTP, Web Sockets, MQTT, CoAP), WoT Scripting and Interaction Models (Properties, Actions, Events)

Activities:

- Mind map comparison of IoT vs WoT
- Analysis of WoT reference architecture

WoT Standards and Thing Description (TD): W3C WoT Architecture and Building Blocks, Thing Description (TD): Syntax, Vocabulary, JSON-LD format, TD Schema validation and interpretation, Semantic Annotation in TD

Activities:

- Write TD for a smart device in JSON-LD
- Validate TD using WoT tools

Implementation of WoT Servients: WoT Servient structure: Exposed vs Consumed Things, Node-WoT framework, WoT Scripting API, Protocol bindings: HTTP, MQTT, CoAP

Activities:

- Build a WoT Servient with Node-WoT
- Demo using WoT Scripting API

Interoperability, Security, and Privacy: Interoperability challenges, Security mechanisms: HTTPS, OAuth2.0, Access Control, Privacy and anonymization, Trust and identity in WoT ecosystems

Activities:

- Present security issues in WoT
- Design secure architecture for smart home

Applications and Future Trends in WoT: Smart Cities, Smart Homes, e-Health, IIoT, Cloud and Edge integration, Data analytics and AI in WoT, Future directions: Digital twins, Metaverse

Activities:

- Present case studies of real-world WoT
- Brainstorm WoT smart city features

Integration to WoT: Integrating an Arduino to WoT – Integrating BeagleBone to WoT – Integrating an Intel Edition to WoT – Integrating other Embedded systems to WoT

Activities: Create a simple project to integrate an embedded device with WoT

Weightage: Continuous Assessment: 40%, End Semester Theory Examinations: 60%

Assessment Methodology: Assignments (30), Quiz (10), Virtual Demo (25), Flipped Class Room (10), Review of Gate and IES Questions (25)

References

1. Michael Kuniavsky, *Smart Things: Ubiquitous Computing User Experience Design*, Morgan Kaufmann, 2010.
2. Dom Guinard & Vlad Trifa, *Building the Web of Things*, Manning Publications, 2016.
3. akima Chaouchi, *The Internet of Things: Connecting Objects to the Web*, Wiley, 2010.
4. Peter Waher, *Mastering Internet of Things*, Packt Publishing, 2018.
5. Alasdair Gilchrist, *Industry 4.0: The Industrial Internet of Things*, Apress, 2016.

Web resource

W3C WoT Working Group Documents: https://www.w3.org/WoT/H

CO	Description of CO	PO	PSO
CO1	Describe the fundamental concepts, architectures, and enabling technologies of the Web of Things for connecting smart devices to web-based platforms.	--	--
CO2	Analyze Web of Things frameworks and communication protocols to understand their performance, interoperability, and scalability in distributed environments.	PO1 (3)	PSO1 (3)
CO3	Evaluate Web of Things platforms, data integration techniques, and security mechanisms for their suitability in various application domains.	PO3 (2)	PSO2 (2)
CO4	Design Web of Things–based solutions by selecting appropriate architectures, protocols, and services to develop scalable and secure smart applications.	PO2 (1)	PSO1 (3)

NE25001	Network Automation	L	T	P	C
		3	0	0	3

Course Objective:

- This course aims to provide in-depth knowledge of automating network infrastructure using software-defined networking (SDN), network function virtualisation (NFV), and scripting tools.
- It equips students with the skills to design, deploy, and manage programmable networks using automation frameworks and open-source tools.
- To Emphasis is placed on scalability, efficiency, and security in modern data centres and cloud environments.

INTRODUCTION TO NETWORK AUTOMATION

Overview of traditional vs. automated networks – Need for automation – Basic concepts of SDN and NFV – Network programmability – Use cases in data centres and service providers

Activities

- Seminar on key differences between traditional and software-defined networks.
- Prepare a peer-taught session on SDN and NFV concepts with real-time examples.
- Prepare a poster presentation for programmable network architecture.

NETWORK AUTOMATION TOOLS AND LANGUAGES

Introduction to Python for network automation – APIs and RESTful interfaces – Using tools: Ansible, Netmiko, NAPALM – Configuration management basics

Suggested Activities

- Write a Python script to fetch configuration from a router using Netmiko.
- Automate switch configuration using Ansible playbooks.
- Quiz on API and YAML syntax concepts

SOFTWARE-DEFINED NETWORKING (SDN)

SDN architecture – OpenFlow protocol – Northbound and Southbound APIs – SDN Controllers: Open Day light, ONOS – Virtualization in networking

Activities

- Simulate a mini SDN environment using Mininet and OpenDaylight.
- Make a model of the layered architecture of SDN with controllers.
- Prepare a poster presentation on SDN controllers based on scalability and features.

NETWORK FUNCTION VIRTUALIZATION (NFV)

NFV architecture and components – NFV MANO – Integration with SDN – Service function chaining – VNF deployment and orchestration

Activities

- Design a project of a simple VNF deployment architecture using open-source tools.
- Using a flipped Classroom, explain NFV MANO and its layers.
- Prepare a case study on real-time NFV use in a telecom environment.

AUTOMATION IN ENTERPRISE AND CLOUD NETWORKS

Cloud-native network automation – CI/CD for network configuration – Monitoring and telemetry – Network security automation – Compliance and audit frameworks

Suggested Activities

- Design a mini-Project to automate ACL and firewall rule deployment.
- Prepare a poster presentation on CI/CD integration for networking.

Demonstrate the usage of telemetry tools for automated network health reporting

MONITORING AND TELEMETRY IN AUTOMATED NETWORKS

Real-time network monitoring principles – Telemetry data collection and analysis – SNMP, NetFlow, sFlow, gNMI protocols – Alerting and automated response systems – Integration of telemetry with automation frameworks

Activities

- Configure a network device to export telemetry data using SNMP or NetFlow and analyze the output
- Create a poster showing the telemetry data flow from the device to the monitoring dashboard
- Conduct a mini-project integrating telemetry data into an automated alerting system

SECURITY AUTOMATION AND POLICY MANAGEMENT

Principles of security automation – Automated firewall and ACL management – Role-based access control automation – Policy as code – Integrating security policies into CI/CD pipelines – Threat detection and automated mitigation

Activities

- Simulate automated deployment of security policies using Ansible or a similar tool
- Case study on a real-world automated security incident response system
- Group discussion on the benefits and challenges of policy as code in network security

Weightage: Continuous Assessment: 40%, End Semester Theory Examinations: 60%

Assessment Methodology: Assignments (30), Quiz (10), Virtual Demo (25), Flipped Class Room (10), Review of Gate and IES Questions (25)

References

1. Jason Edelman, Scott Lowe, and Matt Oswalt, Network Programmability and Automation, O'Reilly Media, 2018.
2. Russ White, Network Automation Made Easy, Packt Publishing, 2021.
3. Antonio Sanchez Monge and Krzysztof Grzegorz Szarkowicz, MPLS in the SDN Era, O'Reilly Media, 2015.
4. Nilesh Shah, Mastering Python Networking, Packt Publishing, 2021.
5. Thomas D. Nadeau and Ken Gray, SDN: Software Defined Networks, O'Reilly Media, 2013.

E-Resources

- https://nptel.ac.in/courses/108105159
- http://digimat.in/nptel/courses/video/108105159/L01.html

CO	Description of CO	PO	PSO
CO1	Describe the fundamental concepts, tools, and techniques of network automation for modern communication systems.	--	--
CO2	**Analyze** network automation workflows, protocols, and management tools to understand efficiency, scalability, and reliability in network operations.	PO1 (3)	PSO1 (3)
CO3	**Evaluate** automation frameworks, scripting approaches, and monitoring strategies to assess their effectiveness in optimizing network performance.	PO3 (2)	PSO2 (2)
CO4	**Design** automated network solutions by selecting appropriate tools, protocols, and strategies for real-world deployment scenarios.	PO2 (1)	PSO1 (3)

NE25002	Unified Communications	L	T	P	C
		3	0	0	3

Course Objective:

- This course introduces the principles and architecture of Unified Communications (UC), focusing on signalling protocols, real-time media transmission, and integration techniques across voice, video, and data systems.
- It aims to equip students with the technical knowledge and practical skills to analyse, design.
- To secure UC solutions for modern enterprise and cloud environments.

Introduction to Unified Communications

Evolution from PSTN to IP-based communication-Definitions: VoIP, UC, multimedia conferencing, instant messaging-UC system components: endpoints, proxies, servers, gateways- Network architectures for UC: client-server, peer-to-peer, cloud-native

Activities

- Present a comparison between PSTN and VoIP systems focusing on components and cost-effectiveness.
- Create a poster showing the architecture of a modern UC system with roles of endpoints, proxies, and gateways.
- Analyse a real-world deployment of a cloud-native UC solution and discuss its benefits over traditional systems

Signaling and Control Protocols

SIP: architecture, message formats, session control-H.323: comparison with SIP-RTP/RTCP for media transport and QoS Monitoring-Codec standards: G.711, G.729, H.264

Activities

- Simulate a SIP session using Wireshark and analyze the SIP message flow.
- Build a layered diagram model comparing SIP and H.323 protocols.
- Conduct a timed quiz to assess understanding of RTP, RTCP, and codec standards like G.711 vs. G.729.

Media Transport and Control

Real-time Transport Protocol (RTP) and RTX-SIP RTP Control Protocol (RTCP)-Jitter, latency, packet loss, synchronization-Media encoding and codecs (G.711, G.729, H.264)

Activities

- Demonstrate media stream jitter and packet loss using tools like NetEm or Dummynet.
- Encode and decode a small media file using G.711 and H.264, then compare quality and size.
- Analyse synchronization issues in video/audio transmission over IP and suggest corrective techniques.

Quality of Service and Media Management

Network impairments: jitter, delay, packet loss-QoS strategies: DiffServ, IntServ, RSVP-NAT traversal: STUN, TURN, ICE-Media gateways, SBCs - Configure QoS parameters in a simulated network using DiffServ or IntServ models.

Activities

- Configure QoS parameters in a simulated network using DiffServ or IntServ models.
- Create a NAT traversal setup using STUN/TURN/ICE protocols and test its effectiveness.
- Prepare a peer-taught session on real-time communication QoS techniques and their trade-offs.

Security and Advanced UC Concepts

SIP over TLS, SRTP, authentication mechanisms-Mobility and presence in UC-UCaaS and WebRTC fundamentals-AI integration in communication systems

Activities

- Configure SRTP for a SIP session and analyze the encrypted payload in Wireshar.
- Build a WebRTC-based video calling application with basic AI feature (e.g., background blur).
- Debate on the ethical and security implications of cloud-hosted UCaaS platforms.

Collaboration Platforms and Integration

Integration of UC with collaboration platforms: Teams, Slack, Zoom, Presence, chat, file sharing, whiteboarding, APIs and connectors for third-party applications, Integration challenges: security, compliance, data privacy

Activities

- Develop a small workflow integrating UC with a collaboration tool, e.g., Slack and VoIP system.
- Create a poster on API-based integration between UC and enterprise applications.
- Conduct a case study on collaboration platform adoption in a large organization.

Emerging Trends and Future Directions In UC

5G and UC: low-latency applications, Edge computing for UC, AI and ML in UC: speech analytics, transcription, sentiment analysis, Future challenges: interoperability, standardization, user privacy

Activities

- Seminar, discuss how 5G will transform real-time communication in UC systems
- Mini-project, prototype a simple speech-to-text feature using an AI API integrated into a UC app

- Group discussion, debate on balancing innovation and privacy in next-gen UC solutions

Weightage: Continuous Assessment: 40%, End Semester Theory Examinations: 60%

Assessment Methodology: Assignments (30), Quiz (10), Virtual Demo (25), Flipped Class Room (10), Review of Gate and IES Questions (25)

References

1. Alan B. Johnston, SIP: Understanding the Session Initiation Protocol, Artech House, 3rd Edition, 2021.
2. Stephen A. Rackley, Unified Communications, Elsevier, 2016.
3. R. Droms et al., IP Telephony with H.323, McGraw-Hill, 2015.
4. Charles Perkins, IP Multimedia Communications, Pearson Education, 2017.
5. Kumar Reddy and Satish Babu, Multimedia Communications, Tata McGraw-Hill, 2018.

E-Resources

- https://nptel.ac.in/courses/117105144
- https://nptel.ac.in/courses/117101051

CO	Description of CO	PO	PSO
CO1	Describe the fundamental concepts, components, and technologies of unified communications for integrated voice, video, and data services.	--	--
CO2	**Analyze** unified communication systems, protocols, and deployment architectures to understand performance, interoperability, and quality of service considerations.	PO1 (3)	PSO1 (3)
CO3	**Evaluate** unified communication solutions, platforms, and security mechanisms to assess their effectiveness in enterprise and distributed environments.	PO3 (2)	PSO2 (2)
CO4	**Design** unified communication systems by selecting appropriate technologies, protocols, and integration strategies for real-world applications.	PO2 (1)	PSO1 (3)

CP25C12	Quantum Cryptography	L	T	P	C
		3	0	0	3

Course Objective:
The objective of this course is to introduce the principles of quantum information and computation, along with key quantum algorithms and their practical applications. It also aims to provide a clear understanding of post-quantum and advanced quantum cryptographic techniques essential for secure communication in the quantum era.

Quantum Information: Qubit - Single and Multiple qubits – Mathematical Model for Quantum mechanics – Quantum measurements – Quantum Computation and No-cloning: Phase shift – Bit flips – Hadamard transform – Arbitrary Transforms – Entanglement - Quantum gates and circuits: CNOT gate – CCNOT gate – Reversible gates - Universal quantum gates - quantum circuit – quantum parallelism

Activities

- Build and simulate quantum circuits using Qiskit (or another quantum simulator)
- Create a Bell state using a Hadamard gate followed by a CNOT gate.

Quantum Algorithms: Deutsch's algorithm – Phase Kickback – Generalizing to n bits: Deutsch-Jozsa algorithm – Simon's algorithm: Analysis - Quantum Fourier Transform, Grover's Algorithm, Shor's Algorithm.

Activities

- Simulate both Deutsch's algorithm (1-bit) and the Deutsch-Jozsa algorithm (n-bit) using Qiskit.

Quantum Cryptanalysis: Quantum order finding – Factoring – Discrete logarithms, Hidden subgroup problems (HSP) – Key Exchange: Diffie-Hellman (DH) problems - Computational DH – Decisional DH – Indistinguishable Chosen Plaintext Attack (IND-CPA): Applications in RSA Encryption and Elgamal Encryption

Activities

- Simulate Shor's Algorithm using Qiskit or any quantum simulator to factor small numbers

Post-Quantum Cryptography: Post Quantum Crypto: Introduction to lattices – Codes – Isogenies - Lattice Problems. Learning with Errors (LWE) and Short Integer Solution (SIS) problem. Connection to dihedral hidden subgroup problem - Public Key Encryption (PKE) from LWE - Fully Homomorphic Encryption (FHE).

Activities

- Implement a simplified version of a public-key encryption scheme based on the LWE problem using Python or SageMath

Advanced Quantum Cryptography - I : Quantum Key Distribution and bit commitment – Random Oracles - Quantum One Time Pad and Encryption - Quantum PKE – Quantum FHE - Quantum Indifferentiability - Quantum Money

Activities

• Simulate the BB84 protocol for quantum key distribution between two parties in Python using Qiskit
Advanced Qunatum Cryptography – II : Quantum key distribution with imperfect devices – beyond point -to -point quantum key distribution – device independent Quantum cryptography **Activities:** • Simulate Quantum key distribution between two IoT based device in python using Qiskit
Weightage: Continuous Assessment: 40%, End Semester Theory Examinations: 60%
Assessment Methodology: Assignments (30), Quiz (10), Virtual Demo (25), Flipped Class Room (10), Review of Gate and IES Questions (25)
References: 1. Quantum Computation and Quantum Information, M. A. Nielsen and I. Chuang, Cambridge University Press, 2012. 2. Quantum Computing from the Ground Up, Riley Tipton Perry, World Scientific Publishing Ltd., 2012. 3. Quantum algorithms via linear Algebra Primer, Richard J. Lipton Kenneth W. Regan, The MIT Press Cambridge, 2014. 4. Quantum Computing: An Applied Approach, Jack D. Hidary, 1st Edition, Springer, 2019.
E-resources: 1. Quantum Algorithms and Cryptography (Video) – NPTEL 2. Introduction to Quantum Computing: Quantum Algorithms and Qiskit (Video) – NPTEL 3. Practical Quantum Computing with IBM Qiskit for Beginners (Video) – Coursera

CO-PO-PSO Mapping

CO	Description of CO	PO	PSO
CO1	Describe the fundamental principles, protocols, and security concepts of quantum cryptography and their significance in secure communication systems.	--	--
CO2	Analyze quantum cryptographic protocols and security mechanisms to understand their robustness, vulnerabilities, and performance characteristics.	PO1 (3)	PSO1 (3)
CO3	Evaluate quantum cryptographic techniques and implementations to assess their effectiveness in ensuring confidentiality and secure key distribution.	PO3 (2)	PSO2 (2)
CO4	Design secure communication systems by applying appropriate quantum cryptographic protocols, architectures, and security strategies for real-world applications.	PO2 (1)	PSO1 (3)

CP25C13	Quantum Machine Learning	L	T	P	C
		3	0	0	3

Course Objective:

The objective of the Quantum Machine Learning (QML) course is to provide students with a comprehensive understanding of the intersection between quantum computing and machine learning**.** This course aims to equip students with the theoretical foundation and practical skills necessary to design, develop, and implement quantum algorithms for solving complex machine learning problems.

Introduction to Quantum Computing and Quantum Machine Learning: Basics of Quantum Computing: Qubits, Superposition, and Entanglement - Quantum Gates, Circuits, and Measurement - Quantum Algorithms: Grover's and Shor's Algorithm - Introduction to Quantum Machine Learning (QML) - Classical vs Quantum ML - Applications of Quantum ML

Activities:

- Implement a quantum circuit using Qiskit to demonstrate superposition and entanglement (Create a simple Bell state).
- Use Qiskit to implement and visualize Grover's search algorithm for an unsorted database problem.

Quantum Information Theory and Quantum Algorithms: Quantum Entropy and Quantum Information - Quantum Teleportation and Superdense Coding - No-Cloning Theorem and Quantum Cryptography - Quantum Fourier Transform (QFT) - Quantum Phase Estimation (QPE)

Activities:

- Implement a Quantum Fourier Transform (QFT) using Qiskit and visualize the output on a simple example (e.g., the 3-qubit QFT).
- Simulate quantum teleportation using Qiskit to transfer quantum information between two qubits.

Quantum Algorithms for Machine Learning: Quantum Support Vector Machines (QSVM) - Quantum Principal Component Analysis (QPCA) - Quantum k-Nearest Neighbors (QkNN) - Quantum Neural Networks (QNN)

Activities:

- Implement Quantum Support Vector Machine (QSVM) using Qiskit to classify a small dataset (e.g., Iris dataset).
- Use Qiskit to perform Quantum Principal Component Analysis (QPCA) on a synthetic dataset for dimensionality reduction.

Quantum Optimization and Hybrid Quantum-Classical Models: Variational Quantum Eigensolver (VQE) - Quantum Approximate Optimization Algorithm (QAOA) - Hybrid Quantum-Classical Algorithms - Applications of Quantum Optimization in Machine Learning

Activities:

- Implement the Variational Quantum Eigensolver (VQE) to find the ground state energy of a Hamiltonian using Qiskit.
- Create a Quantum Approximate Optimization Algorithm (QAOA) to solve a Max-Cut problem on a simple graph using Qiskit**.**

Applications, Tools, and Future of Quantum Machine Learning: Applications of Quantum Machine Learning in Finance, Healthcare, NLP, and Drug Discovery - Quantum Programming Tools: Qiskit, TensorFlow Quantum - The Future of Quantum ML: Challenges and Opportunities

Activities:

- Use TensorFlow Quantum to build a simple quantum classifier and run it on a small dataset.
- Explore Qiskit and create a quantum-based recommendation system using hybrid quantum-classical methods.

Advanced Quantum Machine Learning – I: Quantum Mechanics and Data-Driven Physics - Kernelizing Quantum Mechanics - Qubit Maps - One-Qubit Transverse-Field Ising Model and Variational Quantum Algorithms

Activities

- Write a case study about combinatorial optimization with the Ising model and its quantum counterpart.

Advanced Quantum Machine Learning – II: Two-Qubit Transverse-Field Ising Model and Entanglement - Variational Algorithms, Quantum Approximate Optimization Algorithm, and Neural Network Quantum States with Two Qubits - Quantum Reservoir Computing

Activities: Write a case study about Quantum Approximate Optimization Algorithm

Observations from an Industry Exposure Visit: For the Quantum Machine Learning (QML) course, a hands-on learning experience can be organized where students visit a quantum computing research lab, a quantum software company, or a machine learning-focused organization that integrates quantum computing into real-world applications. During this visit, students will have the opportunity to observe cutting-edge workflows involving quantum algorithms, quantum optimization techniques, and hybrid classical-quantum models applied to machine learning tasks. Students will gain insights into how these technologies are deployed to solve complex, high-dimensional problems in industries like finance, healthcare, and optimization. Following the visit, students will write a reflection report summarizing their observations, discussing the tools and technologies encountered, and reflecting on the practical challenges faced in quantum machine learning applications.

Weightage: Continuous Assessment: 40%, End Semester Theory Examinations: 60%

Assessment Methodology: Assignments (30), Quiz (10), Virtual Demo (25), Flipped Class Room (10), Review of Gate and IES Questions (25)

References :

1. Claudio Conti, Quantum Machine Learning Thinking and Exploration in Neural Network Models for Quantum Science and Quantum Computing, Springer Nature Link Publications 2024
2. Michael A. Nielsen, Isaac L. Chuang, Quantum Computation and Quantum Information, Cambridge University Press, 2nd Edition, 2010.
3. Peter Wittek, Quantum Machine Learning: What Quantum Computing Means to Data Mining, Springer, 2014.
4. Chris Bernhardt, Quantum Computing for Everyone, MIT Press, 1st Edition, 2019.
5. Eleanor G. Rieffel, Wolfgang H. Polak, Quantum Computing: A Gentle Introduction, MIT Press, 1st Edition, 2014.
6. Nathan Wiebe, Ashish Kapoor, Krysta Svore, Quantum Machine Learning, Springer, 1st Edition, 2021.
7. Brendon S. H. Tan, Quantum Computing and Quantum Information Science, CRC Press, 1st Edition, 2021.

E-resources/E-materials:

1. **Qiskit Documentation**
   https://qiskit.org/documentation/
   IBM Quantum Experience
   https://quantum-computing.ibm.com/

2. **Quantum Computing for the Very Curious**
   https://quantum.country/qcvc

3. **Google Quantum AI**
   https://quantumai.google/

4. **TensorFlow Quantum**
   https://www.tensorflow.org/quantum

NPTEL/MOOC/SWAYAM Courses:

1. **NPTEL – Introduction to Quantum Computing**
   https://nptel.ac.in/courses/106/105/106105226/
2. **NPTEL – Quantum Computing and Quantum Information**
   https://nptel.ac.in/courses/106/106/106106220/
3. **Swayam – Quantum Machine Learning: Algorithms and Applications**
   https://swayam.gov.in/nd1_noc21_cs56/
4. **Coursera – Quantum Computing and Machine Learning**
   https://www.coursera.org/learn/quantum-machine-learning
5. **Udemy – Quantum Machine Learning**
   https://www.udemy.com/course/quantum-machine-learning/

CO	Description of CO	PO	PSO
CO1	Describe the fundamental concepts, models, and principles of quantum machine learning and their role in next-generation intelligent systems.	--	--
CO2	Analyze quantum machine learning algorithms and hybrid quantum–classical models to understand their computational advantages and limitations.	PO1 (3)	PSO1(3)
CO3	Evaluate quantum machine learning techniques and frameworks to assess their effectiveness for complex data-driven and optimization problems.	PO3 (2)	PSO2(2)
CO4	Design quantum machine learning solutions by selecting appropriate quantum circuits, learning models, and hybrid architectures for real-world applications.	PO2 (1)	PSO1(3)

CP25C14	AI in IoT	L	T	P	C
		3	0	0	3

Course Objective:

- To gain a deep understanding of AI concepts, IoT architectures, and their convergence for intelligent systems.
- To build and train machine learning and deep learning models suited for IoT data processing using TensorFlow / Keras.
- To apply data normalization, filtering, and feature engineering to enhance AI model accuracy
- To deploy AI models on edge devices for low-latency decision-making in IoT environments.
- To implement AI-based cyber security solutions for intrusion detection and anomaly detection in IoT networks.

Introduction to AI in IoT: Fundamentals of AI and IoT - IoT Architecture and AI Integration - AI Models for IoT (Machine Learning, Deep Learning) - AI-Driven IoT Applications (Smart Cities, Healthcare, Industrial IoT) - Ethical Considerations in AI-Enabled IoT – Case Study: Smart Agriculture with AI-Enabled IoT Sensors.

Activities:

- Analytical study of AI integration within IoT ecosystems, focusing on architectural layers, data flow, and intelligence distribution across device, edge, and cloud.
- Simulation-based generation and statistical analysis of IoT sensor data using computational tools (e.g., Jupyter Notebook) to interpret trends, patterns, and variability.

Data Acquisition & Pre-processing in AI-Enabled IoT: Sensor Data Collection & Processing - AI-Based Data Normalization, Filtering & Feature Engineering - Challenges in Data Handling & Storage for AI Models - Ethical Considerations in AI-Driven Data Processing – Case Study: Predictive Maintenance in Smart Factories

Activities:

- Develop a conceptual framework for real-time sensor data preprocessing, detailing data collection, normalization, filtering, and feature engineering stages.
- Analyze AI-enhanced IoT data filtering techniques, focusing on methodologies, performance metrics, and practical challenges.

AI Models for IoT Data Processing & Decision-Making: Supervised & Unsupervised AI Models in IoT - Classification, Clustering & Anomaly Detection - Deep Learning Models for IoT: CNNs, RNNs - Performance Metrics & Model Optimization – Case Study: AI-Driven Traffic Monitoring System

Activities

- Analyze and justify the selection of appropriate AI models for real-time IoT data classification, considering problem context, data characteristics, and performance constraints.
- Conduct structured discussions or guided demonstrations on deep learning models (CNNs, RNNs) for IoT applications, focusing on architecture, working principles, and suitability for time-series data.

Edge AI & Security for IoT Systems: Edge AI Deployment Strategies - Edge AI for Real-Time Decision-Making - Deploying AI Models in Resource-Constrained IoT Devices - AI-Powered Security & Intrusion Detection in IoT Networks - Ethical Implications & Privacy Concerns – Case Study: AI- Based Intrusion Detection in Healthcare IoT Networks

Activities:

- Conceptually analyze AI-powered anomaly detection frameworks for IoT systems, focusing on model architecture, detection strategies, and performance evaluation criteria.

- Critically review and present recent research articles on security and privacy mechanisms in IoT systems, with emphasis on AI-based intrusion detection approaches.

AI in IoT – Industry Trends: Industry Applications of AI in IoT - Smart Cities, Healthcare, & Industrial IoT - Scaling AI for Large-Scale IoT Deployments - Sustainability & Scalability Challenges in AI-Enabled IoT

Activities:

- Undertake an analytical case study of AI-driven Smart Cities, examining sustainability, scalability, governance frameworks, and ethical considerations.

AI in IoT – Ethics and Future Trends: Monetization Strategies & AI Business Models in IoT **-** Green AI and IoT Regulations and Standardization - Edge AI advancements for real-time IoT decision-making - Quantum AI & IoT interoperability for future applications - Case Study: AI-Driven Smart Cities for Sustainable Urban Development

Activities:

- Prepare a conceptual design proposal for a real-time IoT-based decision-making system, detailing system architecture, AI model rationale, deployment strategies, and regulatory compliance aspects.

Weightage: Continuous Assessment: 40%, End Semester Theory Examinations: 60%

Assessment Methodology: Assignments (30), Quiz (10), Virtual Demo (25), Flipped Class Room (10), Review of Gate and IES Questions (25)

References

1. Fadi Al-Turjman, “Artificial Intelligence in IoT”, Springer, 2019
2. Ambika Nagaraj,” The Role of AI in Enhancing IoT-Cloud Applications”, Bentham Books, 2023
3. Rajkumar Buyya and Amir Vahid Dastjerdi, "Internet of Things: Principles and Paradigms", Elsevier, 2016
4. Jason Ioannou, "Data Analytics for IoT and Smart Systems", Springer, 2020
5. Ian Goodfellow, Yoshua Bengio, and Aaron Courville, "Deep Learning", MIT

Press, 2016
6. Weisong Shi, "Edge Computing: Vision and Challenges", Springer, 2020
7. Mauro Conti, "Security and Privacy in the Age of IoT", Springer, 2020
8. François Chollet, “Deep Learning with Python” (2nd Edition), Manning Publications, 2021
9. Stuart Russell & Peter Norvig, “Artificial Intelligence: A Modern Approach (4th Edition), Pearson, 2020
10. Arshdeep Bahga & Vijay Madisetti, “Internet of Things: A Hands-On Approach” (2nd Edition), Universities Press, 2016

Web References

https://www.springer.com/gp/campaign/artificial-intelligence/artificial-intelligence-in-iot
https://iotworldmagazine.com/2025/01/16/2675/list-of-top-50-best-ai-automation-tools- in-2025
https://www.opensourceforu.com/2025/02/open-source-ai-frameworks-integrating-ai-with-iot/

CO	Description of CO	PO	PSO
CO1	Describe the fundamental concepts, architectures, and applications of integrating artificial intelligence techniques with Internet of Things (IoT) systems.	--	--
CO2	Analyze AI-enabled IoT systems to understand data processing, learning models, and decision-making mechanisms in distributed environments.	PO1 (3)	PSO1 (3)
CO3	Evaluate AI techniques and IoT platforms to assess their effectiveness, scalability, and reliability for intelligent connected applications.	PO3 (2)	PSO2 (2)
CO4	Design intelligent IoT solutions by selecting appropriate AI models, sensing strategies, and system architectures for real-world applications.	PO2 (1)	PSO1 (3)

CP25C15	Web 3.0	L	T	P	C
		3	0	0	3

Course Objective:

- To understand the progression from Web 1.0 and Web 2.0 to Web 3.0, with an emphasis on decentralization, semantic interoperability, and enhanced privacy.
- To gain hands-on expertise in blockchain technology, consensus algorithms, smart contract development, and dApp design by employing open standards.
- To deploy NFT tokenization models and digital identity systems in accordance with W3C standards.
- To develop the ability to integrate emerging technologies (IoT, AI, Semantic Web) while ensuring accessibility, security, and global interoperability.
- To critically assess decentralized governance models and regulatory challenges, proposing innovative solutions based on industry best practices.

Introduction: Evolution of the Internet – Transition from Web 1.0 (static/read-only) to Web 2.0 (interactive/centralized) and on to Web 3.0 (decentralized) – Core Concepts – Principles – Decentralisation – Semantic Interoperability - Exploration of Web 3.0's layered structure: Blockchain, Decentralized storage, P2P networks, and Semantic Web protocols.

Activities:

- Presentations on key historical transitions and share related research papers before in-class discussions.

Blockchain Technology & Distributed Ledger Systems: Blockchain Architecture – Consensus Mechanism – Proof-of-Work (PoW), Proof-of-Stake (PoS), and hybrid models – Pros and Cons - Directed Acyclic Graphs (DAGs) and emphasis on RESTful API design following open standards.

Activities:

- Pre-record lectures on consensus mechanisms and present comparative analyses.
- Reproduce comparative studies on consensus mechanisms using available simulation tools.

Smart Contracts, dApps, Block chain Integration

Fundamentals of design, deployment, and management of smart contracts on Blockchain platforms **-** Decentralized Applications (dApps) - Integration of Blockchain back - ends with front-end systems - HTML5, semantic HTML, and ARIA/WCAG

Activities

- Team-based development of the dApp, with roles distributed for front-end, smart contract, and oracle integration.
- Mini-tutorials on dApps and present case studies of dApps projects in class.

Digital Asset Tokenization (NFTs) and Decentralized Storage: Digital Asset Tokenization & NFTs - ERC-721/1155 standards - Metadata structuring with JSON-LD for interoperability - Oracle Services – Dynamic and real-time data - Decentralized Storage Systems – Storage Protocols – IPFS – Filecoin

Activities

- Team-based development of the NFT smart contract
- Mini-tutorials on Decentralized storage .

Security, Digital Identity, and Peer-to-Peer (P2P) Networking: Security – Scalability – Interoperability with traditional systems – Digital Identity – Decentralized Identity (DIDs) self-sovereign identity principles, and W3C DID and verifiable credentials standards.

Activities:

- Group projects to develop and test decentralized storage solutions and identity modules.
- Pre-session assignments where students analyze recent research on decentralized identity and storage.

Decentralized Governance & Tech Integration: Decentralized Governance & DAOs – Token – based governance – Structure of DAOs – voting mechanism - Regulatory & Economic Frameworks – Technology integration - Semantic Web protocols (RDF, OWL, SPARQL), AI for data curation, and IoT for real-time data collection - open standard compliance.

Activities:

- Case studies on DAO governance and present summaries of regulatory research.
- Collaboration in groups to design and prototype DAO governance models and IoT integration solutions.

Weightage: Continuous Assessment: 40%, End Semester Theory Examinations: 60%

Assessment Methodology: Assignments (30), Quiz (10), Virtual Demo (25), Flipped Class Room (10), Review of Gate and IES Questions (25)

References

1. Prabhat Kumar Srivastav, Prateek Singhal, Basudeo Singh, Nitin Sharma, “Web 3.0 | The Next Generation’s Internet and Understanding the Concept”, CRC Press, 2024.
2. Imran Bashir, “Mastering Blockchain: Unlocking the World of Cryptocurrencies”, Latest Edition, Packt Publishing, 2023.
3. Andreas M. Antonopoulos and Gavin Wood, “Mastering Ethereum: Building Smart Contracts and DApps”, 2nd Edition, O'Reilly Media, 2018.
4. Alex Tapscott, “Web3: Charting the Internet’s Next Economic and Cultural Frontier, Portfolio”, 2023.

Web References

1. https://ipfs.tech/
2. https://docs.filecoin.io/
3. https://identity.foundation/
4. https://www.w3.org/TR/did-1.0/
5. https://www.aragon.org/

CO	Description of CO	PO	PSO
CO1	Describe the fundamental concepts, technologies, and architectural principles of Web 3.0 and their impact on next-generation internet applications.	--	--
CO2	Analyze Web 3.0 platforms and decentralized technologies to understand their functionality, performance, and security implications.	PO1 (3)	PSO1 (3)
CO3	Evaluate Web 3.0 technologies, protocols, and applications to assess their suitability for decentralized and user-centric systems.	PO3 (2)	PSO2 (2)
CO4	Design Web 3.0–based applications by selecting appropriate decentralized architectures, protocols, and technologies for real-world use cases.	PO2 (1)	PSO1 (3)

NE25003	Smart Grid Networks	L	T	P	C
		3	0	0	3

Course Objective:

- Provide comprehensive knowledge of smart grid architecture and communication technologies.
- Analyse cyber-physical aspects of smart grids, including control, computation, and communication.
- Introduce standards, protocols, and security frameworks for reliable smart grid operation.

Introduction to Smart Grids and System Architecture

Evolution from traditional grid to smart grid-Smart grid components: generation, transmission, distribution, control centers-Functional architecture of smart grids and Key features: self-healing, real-time monitoring, resiliency, integration -Overview of India Smart Grid Vision & European Smart Grid Task Force

Activity: Case study analysis of India Smart Grid Vision and European Smart Grid Task Force reports.

Smart Metering and IoT Integration

Advanced Metering Infrastructure (AMI) - IoT stack, IoT enabling technologies - IoT levels: sensing and actuation in smart grids -Home Area Network (HAN), Neighborhood Area Network (NAN), Wide Area Network (WAN) - Sensor types and actuator types

Activity: Hands-on: Identify IoT layers in a smart metering system and evaluate sensing/actuation types

Communication Technologies for Smart Grids

Role of communication in smart grid automation-Wired technologies: PLC, Ethernet, Fiber optics-Wireless technologies: ZigBee, Wi-Fi, LoRa, 5G, LTE, Cellular IoT-Smart grid Protocols: DNP3, IEC 61850, Modbus, MQTT, DLMS/COSEM-Interoperability, latency, QoS and bandwidth considerations.

Activity: Simulate a smart metering communication setup using NS-3 or OMNeT++.

Distributed Energy Resources and Grid Integration

Integration of renewables: solar, wind – microgrids and energy storage systems-Power electronics interface and control-Grid code compliance and synchronization issues-Virtual power plants (VPP), prosumers, energy trading markets-Microgrid management and islanding detection.

Activity: Design a case study to simulate DER and battery control in a grid-connected mode using MATLAB/Simulink.

Control Systems and Demand Side Management

Load forecasting and response Strategies- Pricing models: Real-time pricing, Time of Use (ToU), peak shaving-Smart appliances and automation-Distributed control, decentralized vs centralized control strategies-Role of AI/ML in load prediction and energy optimization.

Activity: Mini project: Develop a demand response strategy for a smart home using Python or MATLAB.

Security, Standards, and Future Trends

Cybersecurity threats in smart grid: replay attacks, data manipulation and intrusion-Cryptographic techniques: Authentication, encryption, data privacy mechanisms-IEEE, NIST, and IEC standards for smart grids

Activity: Critical review of a recent cyberattack on a smart grid system and discussion on mitigation strategies.

Future Trends and Emerging Technologies

Blockchain and AI in smart grid security and optimization - Transactive energy models and peer-to-peer trading - Vehicle-to-Grid (V2G) systems and integration with EVs - Digital twins for grid simulation and planning - Future outlook: 6G-enabled smart grids and edge intelligence

Activity: Group Discussion or Presentation: Envisioning the future of smart grid with emerging technologies

Reference Books:

1. James Momoh, Smart Grid: Fundamentals of Design and Analysis, Wiley

2. Gilbert N. Sorebo, Michael C. Echols, Smart Grid Security: An End-to-End View of Security in the New Electrical Grid, CRC Press
3. Janaka Ekanayake et al., Smart Grid: Technology and Applications, Wiley

E-Resources:

1. NPTEL Course: *Smart Grid – Prof. Ashok Jhunjhunwala (IIT Madras)*
2. IEEE Standards: *IEEE 2030, IEEE 1547, IEEE C37.118*, IEC 61850
3. MIT OpenCourseWare: *Cyber-Physical Systems in Smart Grids*

CO	Description of CO	PO	PSO
CO1	Describe the fundamental concepts, components, and communication technologies of smart grid networks used in modern power systems.	--	--
CO2	**Analyze** smart grid network architectures, communication protocols, and data management mechanisms to understand reliability, efficiency, and scalability.	PO1 (3)	PSO1 (3)
CO3	**Evaluate** smart grid networking technologies, security mechanisms, and control strategies to assess their effectiveness in intelligent power distribution systems.	PO3 (2)	PSO2 (2)
CO4	**Design** smart grid network solutions by selecting appropriate communication technologies, architectures, and control mechanisms for real-world energy applications.	PO2 (1)	PSO1 (3)

CP25C17	Edge and Fog Computing	L	T	P	C
		3	0	0	3

Course Objective:

This course aims to provide students with a comprehensive understanding of designing computing systems that process data closer to where it's created, rather than sending everything to distant cloud servers. Students will learn to build smart applications that work across different computing layers - from powerful cloud servers to small edge devices like sensors and smartphones. By the end, students can create practical solutions for smart cities, healthcare, and industry that respond quickly and work efficiently.

Cloud to Edge Computing Paradigms: Evolution from Cloud to Fog and Edge Computing - IoT Architecture Requirements and Latency Limitations - Fog vs Edge vs Cloud: Differences in Computation and Storage - Multi-access Edge Computing (MEC) Concepts - Multi-tier Architectural Models for Distributed Systems

Suggested Activities:

- Case Study: IoT solution using fog or edge
- Group Discussion: Business opportunities in edge markets

Edge Offloading and System Optimization: Edge Offloading Strategies and Placement Issues - Challenges in Latency, Energy Efficiency, QoS, and Privacy - Queuing Theory and System Performance Metrics - Optimization Techniques using Machine Learning for Slice Orchestration

Suggested Activities:

- Performance Comparison: Cloud vs Edge
- Mathematical Modeling: Latency or offloading simulation

Resource Federation and Network Slicing: Cloud-to-Fog-to-Things (C2F2T) Integration - Network Slicing for 5G/6G: Slice Types and Lifecycle - Resource Virtualization: SDN, NFV, and Container-Based Federation - Modeling and Simulation for Resource Management

Suggested Activities:

- Implementation Case Study: 5G network slicing for verticals
- Quiz: Virtualization and resource slicing concepts

Middleware for Edge Systems: Middleware Design Goals: Heterogeneity, Discovery, Scalability – Service Orchestration at the Edge - Lightweight Containers: Docker and Kubernetes on Constrained Devices - Edge Cluster Formation and

Node Coordination - Communication Protocols: MQTT, CoAP, and Microservices - Distributed Storage, Synchronization, and Fault Recovery

Suggested Activities:

- Prototype: Middleware architecture on edge devices
- Containerization Project: Docker/Kubernetes deployment
- Security Analysis: Vulnerabilities in edge middleware

Intelligence and Analytics at the Edge: Data Management at the Fog Layer: Caching, Placement, Lifecycle - Federated Learning and Distributed ML for Edge Analytics - Stream Processing and Real-Time Analytics Frameworks - Use Cases: Smart Cities, Healthcare, Smart Homes - Security Challenges: ML-Based Threat Detection, Behavioral Analysis - Privacy Techniques: Differential Privacy, Federated Anonymization

Suggested Activities:

- Implement Predictive Analytics: Apache Kafka + Flink
- Develop ML Model: IoT threat detection
- Case Study: Privacy-preserving fog computing

Advanced Technologies for Secure and Efficient Edge Computing: Blockchain for Fog Resource Security and Transparency - Software-defined Edge Networks and Intent-Based Networking - Green and Sustainable Edge Computing-Security and Privacy Challenges in Edge Environments

Suggested Activities:

- Case Study: Edge security or sustainability
- Technical Report: Designing a secure, green edge architecture

Future Directions and Emerging Trends: Edge AI and TinyML: Neural Compression, Quantization, Model Deployment - 6G Vision: Ultra-Low Latency, Terahertz Communications - Advanced Paradigms: Digital Twins, Serverless Edge, Cognitive Networks

Suggested Activities:

- Research Project: Emerging tech integration with edge computing
- Presentation: Future research directions and innovation trends
- Technical presentation on future research directions in edge computing

Weightage: Continuous Assessment: 40%, End Semester Theory Examinations: 60%

Assessment Methodology: Assignments (30), Quiz (10), Virtual Demo (25), Flipped Class Room (10), Review of Gate and IES Questions (25)

References:

1. Rajiv Pandey, Sunil Kumar Khatri, Neeraj Kumar Singh, Parul Verma, Artificial Intelligence and Machine Learning for EDGE Computing, 1st Edition,2022, Academic Press
2. S. Goundar, Ed., 'Edge Computing - Technology, Management and Integration'. IntechOpen,2023.
3. Ahmed Banafa, Secure And Smart Internet of Things (IoT) using Blockchain and Artificial Intelligence (AI).2018, River publications
4. Javid Taheri, Schahram Dustdar, Albert Zomaya, Shuiguang Deng, Edge Intelligence From Theory to Practice, 2023, Springer
5. Fog, Edge, and Pervasive Computing in Intelligent IoT Driven Applications Deepak Gupta, Aditya Khamparia ,2021, Wiley-IEEE Press

E-Resources:

1. Kubernetes on Edge: K3s Documentation and Tutorials - https://k3s.io/
2. Foundation of Cloud IoT Edge ML, NPTEL, Prof. Rajiv Misra, IIT Patna - https://archive.nptel.ac.in/courses/106/104/106104242/
3. Open5GS Documentation - https://open5gs.org/
4. Eclipse IoT Working Group - https://iot.eclipse.org/
5. TensorFlow Lite for Microcontrollers - https://www.tensorflow.org/lite/microcontrollers
6. Apache Flink Training - https://flink.apache.org/
7. Hyperledger Fabric Tutorials - https://hyperledger-fabric.readthedocs.io/
8. Subedi, P., Alsadoon, A., Prasad, P.W.C. et al. Network slicing: a next generation 5G perspective. J Wireless Com Network 2021, 102 (2021). https://doi.org/10.1186/s13638-021-01983-7

CO	CO Description	PO	PSO
CO1	Explain the evolution of cloud-to-edge paradigms and describe the role of edge and fog computing in real-time IoT applications.	PO2, PO3	PSO1
CO2	Design and optimize network slicing strategies for integrated cloud–fog–edge systems using mathematical modeling and optimization techniques.	PO1, PO3	PSO1, PSO2
CO3	Develop middleware architectures and container-based solutions for distributed edge computing environments.	PO1, PO3	PSO1
CO4	Implement intelligent data management systems and machine learning solutions at the fog layer for real-time analytics.	PO1, PO3	PSO1, PSO2

CO5	Evaluate emerging technologies and research directions in edge and fog computing for future system design.	PO1, PO2, PO3	PSO2

NE25004	**Futured Network Systems**	L	T	P	C
		3	0	0	3

Course Objective:

- This course covers the evolution and limitations of network architectures, introducing SDN and NFV for enhanced flexibility.
- It examines the integration of edge, fog, and cloud computing with 5G/6G technologies.
- Students will learn about IoE and Cyber-Physical Systems, focusing on real-time communication and security.

Network Evolution and Architectural Limitations

Evolution of Internet Architecture – Traditional vs Software-Defined Networks – Limitations of the current model – Drivers for change: IoT, latency, mobility, scale – Network programmability – Virtualization in Networks -Information Security and Data Flow in Future Networks.

Activity: Panel Discussion/Quiz

Students engage in a structured panel discussion, comparing traditional and software-defined networking. Each group defends viewpoints on architecture evolution, latency, and scalability. Followed by a quiz to reinforce key technical takeaways.

Software-Defined Networking and NFV

SDN architecture – Separation of Control and Data Plane – OpenFlow Protocol – SDN Controllers (POX, Ryu, ONOS) –Introduction to NFV – Use Cases of SDN/NFV in data centers and WANs – Orchestration in NFV -Relationship between SDN and NFV.

Activity: Seminar/Role Play

Learners present seminars on SDN architecture, controllers, and NFV use cases. Role play simulates interaction between network layers or stakeholders (e.g., controller vs data plane). This helps contextualize abstract protocol concepts in real environments.

EDGE, FOG, AI, Cloud Continuum and 5G/6G Networking

Fog Computing vs Edge vs Cloud – MEC (Multi-access Edge Computing) – Resource Management – Use Cases: Smart City, Vehicular Networks – 5G

Architecture – Network Slicing – URLLC, eMBB, mMTC – Vision of 6G – AI-native networks, Terahertz spectrum.

Activity: Case Study Presentation
Teams explore real-world implementations of fog, edge, or 5G technologies in domains like smart cities or autonomous vehicles. Each group presents technical design, challenges, and outcomes. Encourages application-oriented learning and critical thinking.

Internet of Everything and Cyber-Physical Systems
IoT, IIoT architecture – Cyber-Physical Systems – Protocols: MQTT, CoAP, LoRaWAN – Real-time communication – Sensor networks – Privacy and security challenges – Integration of edge intelligence in CPS.

Activity: Concept Mapping Activity/Quiz
Students build visual concept maps linking components of IoT and CPS systems, such as sensors, protocols, and edge AI. This helps clarify system integration and security challenges. A short quiz tests understanding of protocol-function mapping.

Paradigm Shifts in Networks
Information-Centric Networking (ICN) – Delay-Tolerant Networking (DTN) – Zero Trust Architecture – Quantum Networking -Programmable Optical Networks – Satellite and Space-based Networks-

Activity: Critical Review Assignment/Paper Reproduction
Students select a research paper on paradigm shifts like ICN or Quantum Networking and critically analyze methodology and results. Alternatively, they reproduce simplified experiments or simulations from the paper. Enhances research literacy and innovation awareness.

Emerging Trends in Futured Network
Emerging Standards and Research Problems: AI for Network Management- Big Data in Networks- Blockchain in Networks.

Activity:
Write an Assignment about AI in Blockchain

Total: 45 Periods

Weightage: Continuous Assessment: 40%, End Semester Theory Examinations: 60%

Assessment Methodology: Assignments (30), Quiz (10), Virtual Demo (25), Flipped Class Room (10), Review of Gate and IES Questions (25)

References :

1. Vikas Kumar Jha, Bishwajeet Kumar Pandey, Ciro Rodriguez, "Network Evolution and Applications", 1st Edition, CRC Press, 2024.
2. Mahmoud Elkhodr, Qusay F. Hassan, Seyed Shahrestani, "Networks of the Future: Architectures, Technologies, and Implementations", 1st Edition, Chapman & Hall/CRC, 2020.
3. Brij B. Gupta, Amrita Dahiya, Elhadj Benkhelifa, "SDN and NFV: A New Dimension to Virtualization", 1st Edition, World Scientific Publishing, 2022.
4. Patrick Hung, Hongwei Kan, Greg Knopf, "Edge Computing Acceleration: From 5G to 6G and Beyond", 1st Edition, Wiley-IEEE Press, 2024.
5. Keshav Kaushik, Susheela Dahiya, Akashdeep Bhardwaj, Yassine Maleh, "Internet of Things and Cyber Physical Systems: Security and Forensics", 1st Edition, Routledge, 2023.

E-Resources:

1. https://onlinecourses.nptel.ac.in/noc25_cs15/preview
2. https://www.coursera.org/learn/computer-networking
3. https://www.coursera.org/learn/5g-mobile-networks-course-3
4. https://www.udemy.com/topic/network-engineering/

https://onlinecourses.nptel.ac.in/noc20_cs69/preview

CO	Description of CO	PO	PSO
CO1	Describe the fundamental concepts, architectures, and evolution of future network systems and their role in next-generation communication technologies.	--	--
CO2	**Analyze** future network architectures, protocols, and enabling technologies to understand performance, scalability, and reliability challenges.	PO1 (3)	PSO1 (3)
CO3	**Evaluate** emerging network technologies, standards, and architectures to assess their suitability for advanced communication and networking applications.	PO3 (2)	PSO2 (2)
CO4	**Design** future network solutions by selecting appropriate architectures, protocols, and technologies to address real-world networking requirements.	PO2 (1)	PSO1 (3)

CP25C18	Green Computing and Sustainability	L	T	P	C
		3	0	0	3

Course Objective:

- To understand the principles of green computing.
- To assess environmental impacts of computing technologies and need for green computing.
- To explore sustainable IT practices and energy-efficient hardware system for sustainability growth.

Introduction to Green Computing: Scope - Environmental Impact of Computing - Need for Sustainable Computing – Power-Aware Resource Management - Green IT - OCED Green IT Framework – Greening IT - Benefits and Challenges - Quantifying IT Energy Efficiency.

Activities: Review: E-waste management.

Green Hardware: Life cycle of hardware - Reuse, Recycle and Dispose Green Design and Manufacturing of Computers- Green Software - Energy-Saving Software Techniques

Activities: Demo: DAQ tools

Sustainable Software Development: attributes – metrics – Methodology - Energy-aware Software Engineering - Green Algorithms

Activities: Case study: Project's Carbon Footprint

Green Data Center: Infrastructure for Energy Efficiency – server power management

Activities: Poster presentation: Energy issues in Data Centers

Green Data Storage: Practices for Improving the Energy Efficiency of Data Storage - Optical Interconnects for Green Computers and Data Centers

Activities: Case study: Energy-efficient cloud computing

Managing Green IT: Strategizing Green Initiatives – Implementation - Green IT Laws, Standards and Protocols- Regulatory Environment and IT Manufacturers

Activities: Demo: Green IT Laws in India.
Debate: Role of IT professional in Green Computing.

Nonregulatory Government Initiatives: Green Building Standards - Sustainability Reporting and Green Audits

Activities: Perform Energy audit in computer lab using software tools.

Weightage: Continuous Assessment: 40%, End Semester Theory Examinations: 60%

Assessment Methodology: Assignments (30), Quiz (10), Virtual Demo (25), Flipped Class Room (10), Review of Gate and IES Questions (25)

References

1. San Murugesan, Harnessing Green IT: Principles and Practices, Wiley, 2012.
2. Suyel Namasudra(Editor), Green and Sustainable Computing, 1st Edition, Volume 87, Elsevier, 2012.
3. Toby Velte, Anthony Velte, Robert Elsenpeter, "Green IT: Reduce Your Information System's Environmental Impact", McGraw Hill, 2008.
4. Bhuvan Unhelkar, “Green IT Strategies and Applications: Using Environmental Intelligence”, CRC Press, 2011.
5. Balamurugan Balusamy, Naveen Chilamkurti, Seifedine Kadry, Green Computing in Smart Cities: Simulation and Techniques, Springer Singapore, 2020.
6. Mike Halsey, *The Green IT Guide* - How to Make Your IT Systems and Business Sustainable and Carbon Neutral, Apress, 2025

E-Resources

1. Prof. Deepu Philip, Prof. Amandeep Singh | IIT Kanpur
   Link http://onlinecourses.nptel.ac.in/noc21_mg85/preview
2. Prof. Sayak Banerjee | IIT Hyderabad
   Link “https://onlinecourses.nptel.ac.in/noc23_me138/preview”

CO	Description of CO	PO	PSO
CO1	Describe the fundamental concepts, principles, and practices of green computing and sustainability in modern information technology systems.	--	--
CO2	Analyze computing systems and technologies to assess their energy efficiency, environmental impact, and sustainability metrics.	PO1 (3)	PSO1 (3)
CO3	Evaluate green computing strategies, sustainable IT practices, and regulatory frameworks for their effectiveness in reducing environmental impact.	PO3 (2)	PSO2 (2)
CO4	Design sustainable computing solutions by selecting energy-efficient architectures, technologies, and practices for environmentally responsible systems.	PO2 (1)	PSO1 (3)

NE25005	Vehicular Networks	L	T	P	C
		3	0	0	3

Course Objectives:

- Provide an in-depth understanding of vehicular communication architectures and protocols.
- Explore Vehicle-to-Everything (V2X) technologies including V2V, V2I, and V2P.
- Examine the role of vehicular networks in Intelligent Transportation Systems (ITS).

Introduction to Vehicular Networks and ITS

Overview of vehicular networks and real-time applications – Role of Vehicular Ad-hoc Networks (VANETs) in Intelligent Transport Systems (ITS)- Overview of ITS reference architecture (IEEE, ISO, EU ITS-G5, DSRC)

Activity: Use real-world traffic datasets to analyse vehicular communication flow.

Communication Types and System Architecture

Types of communications: V2V, V2I, V2X, V2P - Basic VANET architecture: OBUs, RSUs, communication links - Role of vehicular networks in cooperative and autonomous driving

Activity: Diagram-based mapping of vehicular communication modes in smart transportation use cases.

Communication Technologies and Standards

Dedicated Short Range Communication (DSRC), IEEE 802.11p-Cellular V2X (C-V2X): LTE-V2X, 5G NR-V2X- Comparison of DSRC vs C-V2X- Network architecture: RSUs, OBUs, backhaul integration- Edge computing in vehicular networks.

Activity: Hands-on simulation using NS-3 or OMNeT++ with Veins for DSRC scenarios.

Routing Protocols and Mobility Models in VANETs

Characteristics and challenges in vehicular routing- Position-based, geocast, broadcast, delay-tolerant routing-Routing protocols: GPSR, GSR, AODV, DSR for VANETs- Realistic vehicular mobility models: Manhattan, Freeway, SUMO.

Activity: Implement GPSR over SUMO-generated traces.

Security and Privacy in Vehicular Networks

Threat models: eavesdropping, Sybil attacks, DoS, spoofing- Authentication, message integrity, and confidentiality- Vehicular Public Key Infrastructure (PKI) and Certificate Revocation- Pseudonymity, privacy-preserving protocols-Blockchain in secure V2X communication.

Activity: Case study: Analysis of a real vehicular network attack and countermeasure.

Emerging Architectures: Vehicular Cloud, Edge, and Fog

Vehicular cloud and fog computing paradigms - Edge computing and low-latency offloading - Integration of cloud and edge in connected vehicles

Activity: Design task: Create a lightweight fog-based architecture for vehicle data aggregation.

Applications and Future Trends

Applications: Traffic safety, collision avoidance, infotainment, platooning-Vehicular cloud, fog, and edge computing-Integration with autonomous vehicles and smart cities- AI/ML for context-aware routing and mobility prediction -Research challenges and standardization efforts (ETSI, 3GPP, IEEE).

Activity: Mini-project: Design a context-aware V2X-based traffic management solution.

Weightage: Continuous Assessment: 40%, End Semester Theory Examinations: 60%

Assessment Methodology: Assignments (30), Quiz (10), Virtual Demo (25), Flipped Class Room (10), Review of Gate and IES Questions (25)

References:

1. **H. Hartenstein and K. Laberteaux**, *VANET: Vehicular Applications and Inter-Networking Technologies*, Wiley
2. **C. Campolo, A. Molinaro**, *Vehicular Ad Hoc Networks: Standards, Solutions, and Research*, Springer
3. **D. Jiang**, *IEEE 802.11p and DSRC: Spectrum and Standards for Vehicular Communications*

E-Resources:

1. **NPTEL Course**: *Intelligent Transport Systems* – Prof. Bhaskar Ramamurthi (IIT Madras
2. **Veins framework documentation**, OMNeT++ tutorials

CO	Description of CO	PO	PSO
CO1	Describe the fundamental concepts, architectures, and communication technologies used in vehicular networks and intelligent transportation systems.	--	--
CO2	**Analyze** vehicular network protocols, communication models, and mobility management techniques to understand performance, reliability, and safety requirements.	PO1 (3)	PSO1 (3)
CO3	**Evaluate** vehicular networking technologies, security mechanisms, and routing strategies to assess their effectiveness in dynamic and safety-critical environments.	PO3 (2)	PSO2 (2)
CO4	**Design** vehicular network solutions by selecting appropriate communication protocols, architectures, and security strategies for real-world transportation applications.	PO2 (1)	PSO1 (3)

CP25C19	Cognitive Computing	L	T	P	C
		3	0	0	3

Course Objectives:

- To understand the core principles of cognitive computing and how they emulate human thought processes.
- To apply cognitive computing to solve complex, unstructured, and real-world problems.
- Develop cognitive computing applications.

Human cognition - Introduction to Cognitive Computing – Elements of a Cognitive System - Design Principles for Cognitive Systems - Artificial Intelligence, Natural Language Processing and Machine Learning in Cognitive Computing.

Activities:
1. Concept Quizzes: Human cognition and Cognitive Computing
2. Research paper review on cognitive systems

Cognitive architectures: ACT-R, Soar, OpenCog - Knowledge Representation and Reasoning - Machine Learning models in cognitive systems - Human-like Learning: Supervised, Unsupervised, Reinforcement

Activities: Demo on IBM Watson APIs (Language Translator)

Role of NLP in a Cognitive System - NLP pipeline: POS tagging, NER, parsing - Sentiment Analysis and Question Answering

Debate: Sentiment analysis and its challenges in Regional Languages.

Semantic Web - Semantic Understanding: Ontologies and Knowledge Graphs

Activities:
Ideathon: Innovative solution to health care problems.

Cognitive Agents - Rule-based Agents and Expert Systems - Decision Support Systems -Emotion-aware computing and human-computer interaction- Cognitive bias and ethics in decision making

Activities: Mini project: Develop health care chatbot. Prototype Demonstrations.

Process of Building a Cognitive Application - IBM’s Watson as a Cognitive System- Business Implications of Cognitive Computing

Activities: Research Paper Review: Cognitive Computing

Cognitive Computing Applications: Healthcare, Finance, Education, Legal, Robotics- Building a Cognitive Healthcare Application - Emerging Cognitive Computing Areas - Cognitive cloud- Cognitive IoT - Cognitive Chatbots - Neuromorphic computing and brain-inspired models.

Activities:
1. Prototype Demonstrations.
2. Project demo and peer review.

Weightage: Continuous Assessment: 40%, End Semester Theory Examinations: 60%

Assessment Methodology: Assignments (30), Quiz (10), Virtual Demo (25), Flipped Class Room (10), Review of Gate and IES Questions (25)

References

1. Velankar, Mahalle & Shinde – Cognitive Computing for Machine Thinking, CRC Press, 2024.
2. Judith S. Hurwitz, Marcia Kaufman, Adrian Bowles – Cognitive Computing and Big Data Analytics, Wiley, 2015.
3. Peter Fingar , Cognitive Computing: A Brief Guide for Game Changers, Meghan Kiffer, 2015.
4. Building Cognitive Applications with IBM Watson Services, Redbooks, 2017
5. Steven Bird, Ewan Klein, Edward Loper, Natural Language Processing with Python, O'Reilly, 2009.
6. Fuchun Sun, Qinghu Meng, Zhumu Fu, Bin Fang (Editors), Communications in Computer and Information Science (CCIS, volume 1918) – Springer Series, 2024.

E-Resources

- Prof. Sharba Bandyopadhyay, Prof. Rajlakshmi Guha | IIT Kharagpur, IIT Kharagpur
  Link https://onlinecourses.nptel.ac.in/noc22_ee122/preview
- Cognition and its Computation, IIT Kharagpur
  Prof. Sharba Bandyopadhyay, Prof. Rajlakshmi Guha
  Link "https://nptel.ac.in/courses/108105185"

CO	Description of CO	PO	PSO
CO1	Describe the fundamental concepts, architectures, and paradigms of cognitive computing and their role in intelligent decision-making systems.	--	--
CO2	Analyze cognitive computing models and learning mechanisms to understand perception, reasoning, and knowledge representation in intelligent systems.	PO1 (3)	PSO1 (3)
CO3	Evaluate cognitive computing techniques, frameworks, and applications to assess their effectiveness in solving complex real-world problems.	PO3 (2)	PSO2 (2)
CO4	Design cognitive computing solutions by selecting appropriate models, algorithms, and system architectures for human-centric intelligent applications.	PO2 (1)	PSO1 (3)

NE25006	Intelligent Network and Communication	L	T	P	C
		3	0	0	3

Course Objectives:

- To provide fundamental knowledge of intelligent systems applied in communication networks.
- To explore the integration of Artificial Intelligence (AI) and Machine Learning (ML) in network analysis, optimization, and automation.
- To study intelligent communication paradigms including SDN and NFV.

Introduction to Intelligent Networks

Definition and motivation – Traditional vs. intelligent networks – AI/ML in networking – Intelligent agents – Self-healing networks – Case studies in AI-driven network monitoring

Activity: Develop a Python-based ML model for network traffic classification.

Machine Learning for Networking

Supervised learning – Unsupervised learning – Reinforcement learning – Feature engineering for network data

Activity: Simulate SDN routing using Mininet with AI-enhanced decision logic.

Advanced Machine Learning Applications in Networking

Network traffic prediction – Anomaly detection – Intelligent routing and congestion control – Federated learning for edge networks

Activity: Visualize real-world network anomaly datasets using clustering.

Fundamentals of Software-Defined Networking

Software Defined Networking (SDN) architecture – Southbound and Northbound APIs – OpenFlow and REST APIs.

Activity: Analyze 5G network slicing using open-source tools

Intelligent Control and Management in SDN

Communication between controllers and network devices/applications – Intent-Based Networking concepts – Role of Artificial Intelligence in SDN control plane for intelligent network management and automation.

Activity: Study and present industry use cases (e.g., Google B4, NetNORAD)

Intelligent Communication Systems

5G/6G architecture – Network slicing – Edge AI and fog computing – Intelligent transport networks – IoT network intelligence – Resource allocation using AI

Activity: Submit an Assignment related to Edge AI

NFV and Case Studies

NFV architecture and management – Virtual network functions – AI for orchestration and automation – Service assurance using AI – Real-time use cases from Google B4 and Meta NetNORAD – Research challenges and future directions.

Activity: Write a Case study about Google B4

Weightage: Continuous Assessment: 40%, End Semester Theory Examinations: 60%

Assessment Methodology: Assignments (30), Quiz (10), Virtual Demo (25), Flipped Class Room (10), Review of Gate and IES Questions (25)

References:

1. William Stallings, "Foundations of Modern Networking, Pearson Education, 2016." (Provides a solid foundation for SDN, NFV, IoT, and general network concepts).
2. Aurelien Geron, "Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow" (O'Reilly Media): For in-depth ML algorithms and practical application.
3. Lim, Ng, Xiong, Niyato, Miao, "Federated Learning Over Wireless Edge Networks" (Springer, 2022): For dedicated coverage of Federated Learning.
4. Abdulrahman Yarali, "From 5G to 6G: Technologies, Architecture, AI, and Security" (Wiley-IEEE Press, 2023): For latest cellular network architectures, edge computing, and AI in transport networks.
5. Omar Santos, Samer Salam, and Hazim Dahir, "The AI Revolution in Networking, Cybersecurity, and Emerging Technologies" (Addison-Wesley Professional, February 2024): For practical AI applications in network automation, orchestration, and real-world case studies.

E-Resources

- https://onlinecourses.nptel.ac.in/noc22_ee61/preview
- https://onlinecourses.nptel.ac.in/noc24_hs58/preview

CO	Description of CO	PO	PSO
CO1	Describe the fundamental concepts, architectures, and technologies of intelligent networks and communication systems used in modern digital infrastructures.	--	--
CO2	**Analyze** intelligent networking techniques, communication protocols, and AI-driven network management methods to understand performance and adaptability.	PO1 (3)	PSO1 (3)
CO3	**Evaluate** intelligent communication systems, network optimization techniques, and security mechanisms to assess their effectiveness in dynamic environments.	PO3 (2)	PSO2 (2)
CO4	**Design** intelligent network and communication solutions by selecting appropriate architectures, protocols, and AI-based techniques for real-world applications.	PO2 (1)	PSO1 (3)

NE25007	Wireless Engineering	L	T	P	C
		3	0	0	3

Course Objective:

- This wireless engineering course provides a comprehensive understanding of wireless communication systems. You'll delve into fundamental principles, radio propagation, and the evolution of cellular networks.
- The curriculum emphasizes mastering digital modulation and crucial wireless networking protocols for reliable data transmission.
- Students explore cutting-edge technologies such as 5G, Wi-Fi 6, and LPWAN, preparing you for future advancements in this dynamic field.

Fundamentals of Wireless Engineering

Wireless communication vs. wired communication-Frequency spectrum and regulations-Electromagnetic wave propagation, fading, path loss, and shadowing-Multipath and Doppler effects-Link budget analysis.

Activity: Visualizing Electromagnetic Wave Propagation (Interactive Tool Exploration)-“Seminar”.

In this seminar activity, students interact with online tools to visualize electric and magnetic field oscillations in space and time. They observe wave behavior across different media and frequencies, enhancing conceptual clarity. Learners present their observations and interpretations, fostering peer learning and scientific communication.

Cellular Systems and Multiple Access Techniques

Cellular concepts: Frequency reuse, cell splitting, handoff strategies-Channel assignment and interference management-FDMA, TDMA, CDMA, OFDMA-Capacity and coverage analysis.

Activity: Multiple Access Technique- "Debate".

Students are divided into groups, each defending a specific multiple access technique (FDMA, TDMA, CDMA, OFDMA). They debate based on efficiency, complexity, and real-world applications. This encourages critical thinking and technical articulation.

Modulation and Coding Techniques

Digital modulation: BPSK, QPSK, QAM, FSK, OFDM-Error control coding: Block codes, convolutional codes-Spread spectrum techniques: DSSS, FHSS-Spectral efficiency and performance analysis.

Activity: "Error Correction Story" - Why Do We Need It?-“Seminar”.

Learners create a short narrative or case study illustrating a real-life communication failure without error correction. The seminar highlights the importance of block and convolutional codes. It reinforces concepts through storytelling and technical explanation.

Wireless Networking and Protocols

MAC protocols for wireless networks (CSMA/CA, TDMA, polling)-WLAN, Bluetooth, Zigbee, LTE protocol stack-Routing in wireless ad hoc and sensor networks-QoS and energy efficiency considerations.

Activity: Wireless Networking and Protocols-"Quiz".
An interactive quiz tests students' understanding of MAC protocols, routing, and wireless standards like WLAN and LTE. Includes application-based and scenario questions. Promotes active recall and quick thinking.

Emerging Wireless Technologies
5G and beyond: Architecture, use cases, key technologies (MIMO, mmWave, network slicing)-Wi-Fi 6 (802.11ax): Features and improvements over previous standards-LPWAN: LoRa, NB-IoT, Sigfox-Wireless system simulation and design tools.

Activity: "Wireless Innovation Pitch" :"Presentation".
Students present innovative ideas leveraging emerging technologies like 5G, Wi-Fi 6, or LoRa. Each pitch includes technical feasibility, use case, and future impact. Fosters creativity, technical depth, and communication skills.

Real Time Case studies of Wireless Engineering
A Real-Time Price Recognition System using Lightweight Deep Neural Networks on Mobile Devices – TinyML Network Applications for Smart Cities - Emerging Application Use Cases and Future Directions

Weightage: Continuous Assessment: 40%, End Semester Theory Examinations: 60%

Assessment Methodology: Assignments (30), Quiz (10), Virtual Demo (25), Flipped Class Room (10), Review of Gate and IES Questions (25)

References:

1. Chaudhari, B. S., Ghorpade, S. N., Zennaro, M., & Paškauskas, R., TinyML for Edge Intelligence in IoT and LPWAN Networks, 1st Ed., Elsevier, 2024.
2. Bhowmick, A., Kumar Choukiker, Y., Singh, I., & Nallanathan, A. (Eds.), 5G and Beyond Wireless Communications: Fundamentals, Applications, and Challenges, 1st Ed., CRC Press, 2024.
3. Singh, I., Tayal, S., Singh, N. P., Tripathi, V. S., & Singh, G. (Eds.), 5G and Beyond Wireless Networks: Technology, Network Deployments, and Materials for Antenna Design, 1st Ed., CRC Press, 2024.
4. Yang, L., Shi, J., Feng, K., Shen, L., Wu, S., & Lee, T., Resource Optimization in Wireless Communications: Fundamentals, Algorithms, and Applications, 1st Ed., Academic Press, 2025.

E-Resources:

1. https://www.semtech.com/lora/resources/academy-for-lorawan
2. https://academy.qualcomm.com/course-catalog/Wi-Fi-6-Technical-Training

3. https://www.coursera.org/learn/wireless-communications
4. https://www.khanacademy.org/science/ap-biology/cell-communication-and-cell-cycle/cell-communication/v/cellular-communication
5. https://archive.nptel.ac.in/courses/108/106/106106167/

CO	Description of CO	PO	PSO
CO1	Describe the fundamental concepts, principles, and components of wireless engineering and modern wireless communication systems.	--	--
CO2	**Analyze** wireless communication techniques, propagation models, and system architectures to understand performance, coverage, and reliability.	PO1 (3)	PSO1 (3)
CO3	**Evaluate** wireless technologies, standards, and system designs to assess their effectiveness in various communication environments.	PO3 (2)	PSO2 (2)
CO4	**Design** wireless communication systems by selecting appropriate technologies, architectures, and design parameters for real-world applications.	PO2 (1)	PSO1 (3)

CP25C22	Blockchains Architecture and Design	L	T	P	C
		3	0	0	3

Course Objective:
The objective of this course is to study the Blockchain Technology and its uses across a range of fields, with an emphasis on Blockchain fundamentals, Smart contracts, Consensus processes and Security issues.

Introduction to Blockchain: Introduction to Blockchain – Layers of Blockchain – Types of Blockchain – Generic Elements of Blockchain – Basic Cryptographic primitives used in Blockchain – Hash Functions – Merkle Tree – Structure of Block – Linking Blocks in the Blockchain – Mining the Block – Validating a New Block – Assembling and Selecting Chains of Blocks

Activity: Quiz on core Blockchain concepts like hashing, signing data and building Merkle tree

Bitcoin and Consensus Mechanisms: Bitcoin Overview – Bitcoin Scripts – Bitcoin Transactions – Bitcoin Network and Payments – Consensus mechanism – Types of Consensus mechanism

Activity: Debates and Discussion on the pros and cons of various consensus protocols

Permissioned & Permissionless Protocols: Proof of Work (PoW) – Consensus Protocols for Permissionless Blockchain Environment – Consensus Protocols for Permissioned Blockchain Environment.

Activity:Identify and set up Ethereum testnet – Transfer ethers between accounts

Smart Contracts and Ethereum: Smart Contract – Definition – Ricardian Contracts – Introduction to Ethereum – Ethereum Network – Components of the Ethereum – Ethereum Virtual Machine (EVM) – Ethereum Development Environment – Solidity Language

Activity: Presentation prepared on Smart Contract and Ethereum concepts from research papers

Web3 And Hyperledger Fabric: Exploring Web3 with Geth – Contract Deployment – Interacting with contracts via frontends – Truffle – Introduction to Hyperledger – Reference Architecture – Hyperledger Fabric – Hyperledger Sawtooth – Corda

Activity: Explore and Discuss research papers related to Hyperledger and Blockchain technology

Blockchain Applications: Blockchain for Government: Border control – Voting – Citizen Identification – Blockchain for Health Industry – Blockchain for Finance: Insurance – Post Trade Settlement – Financial Crime Prevention

Activity: Research and present use cases like supply chain, digital identity, etc.

Challenges And Future Directions: Scalability – Privacy – Emerging Trends – Limitations of Current Blockchain Platforms – Open Research Challenges

Activity:
Present mini-projects on performance optimization or privacy enhancement in Blockchain systems

Weightage: Continuous Assessment: 40%, End Semester Theory Examinations: 60%

Assessment Methodology: Assignments (30), Quiz (10), Virtual Demo (25), Flipped Class Room (10), Review of Gate and IES Questions (25)

References:

1. Imran Bashir, “Mastering Blockchain: A deep dive into distributed ledgers, consensus protocols, smart contracts, Dapps, cryptocurrencies, Ethereum,and more”, Third Edition, Packt Publishing, 2020.
2. Andreas M. Antonopoulos, David A. Harding, “Mastering Bitcoin: Programming the open Blockchain”, Third Edition, O’Reilly Publishing, 2023.
3. Ahmed Banafa, “Blockchain Technology and Applications“, First Edition, River Publishers, 2024.
4. Melanie Swan, “Blockchain: Blueprint for New Economy”, First Edition, O’Reilly Publishing, 2015.
5. Bikramaditya Singhal, Gautam Dhameja, Priyansu Sekhar Panda, “Beginning
6. Blockchain : A Beginner’s Guide to Building Blockchain Solutions“ , 1st edition, Apress, NewYork, 2018.

E-Resources:

1. Zero to Blockchain - An IBM Redbooks course, by Bob Dill, David Smits https://www.redbooks.ibm.com/Redbooks.nsf/RedbookAbstracts/crse0401.html
2. Hyperledger Fabric - https://www.hyperledger.org/projects/fabric
3. https://www.tutorialspoint.com/blockchain/index.htm
4. https://www.nptel.ac.in/courses/106105184 by Prof.S.Chakraborty and P.Jayachandran,IIT Kharagpur

CO–PO–PSO Mapping:

CO	Description of CO	PO	PSO
CO1	Describe the fundamental concepts, components, and operational principles of blockchain architectures and distributed ledger technologies.	--	--
CO2	Analyze blockchain consensus mechanisms, data structures, and network architectures to understand security, performance, and scalability issues.	PO1 (3)	PSO1 (3)
CO3	Evaluate blockchain platforms, smart contract frameworks, and architectural designs to assess their suitability for various decentralized applications.	PO3 (2)	PSO2 (2)
CO4	Design blockchain-based systems by selecting appropriate architectures, consensus protocols, and smart contract models for real-world use cases.	PO2 (1)	PSO1 (3)

CP25C24	Vibe Coding	L	T	P	C
		3	0	0	3

Course Objective:

- This course introduces the basics of AI-assisted coding and Vibe Coding
- Participants will learn to use AI tools, build simple applications, and understand the roles and ethics involved.
- Students able to create their own AI-powered projects.

Introduction To Vibe Coding and Human-Ai Collaboration

Introduction to Vibe Coding-Basic Prompting Techniques for Coding Tasks-Roles and -Responsibilities in Human-AI Collaboration-AI Assistant-Prompt Engineer-Developer-Designing Simple AI-Augmented Workflows-Interaction Between AI-Generated Code and Developer Feedback

Activities

- **Role Play:** Students take on roles such as AI, prompt engineer, and developer to simulate collaborative coding.
- **Flipped Classroom & Quiz:** Students learn prompting basics at home and participate in a class quiz and discussion.

Tools And Platforms for Ai-Assisted Coding

Overview of AI-Assisted Coding Tools-Cursor, v0, Bolt, etc.-Features and Use Cases of Each Tool-Setting Up and Navigating AI-Coding Platforms-Building Simple Web Applications with AI -Integration-Workflow Prototyping Using AI Tools

Activities

- **Flipped Classroom & Hands-on Tutorial:** Students explore AI tools (e.g., Cursor or v0) at home and build a basic app in class.
- **Team Activity:** Groups design a simple development workflow using AI tools and explain tool selection and purpose.

AI Prompt Design, Oversight, and Ethical Coding Practices

Vibe Coder's Toolkit: Prompts, Context, and Flow-Prompt Optimization Techniques-Human Oversight in AI Coding Systems-Ethical and Responsible AI Usage-Challenges and Best Practices in AI-Augmented Development

Activities

- **Team Presentation:** Each group presents a complete AI workflow, including prompt design, context handling, and oversight strategy.
- **Ethical Case Discussion:** Class discussion on accountability, code quality, and risks of over-reliance on AI in software development.

The Unbundled Programmer and the Future of Coding Roles

Evolving Developer Roles in the AI Era-The Shift from Coding to Orchestration and Strategy-AI Integration Across the Software Development Lifecycle-Human-AI Collaboration in Debug, Design, and Deployment-Strategic Thinking and Adaptability in AI-Augmented Teams

Activities

- **Team Presentation:** Showcase how AI transforms developer roles across the SDLC.
- **Role Play & Quiz:** Simulate a modern dev team using AI tools and complete a quiz based on workflow observations.

Code Quality, Reliability, and Ethical Development

Long-Term Maintainability of AI-Generated Code-Refactoring AI-Assisted Code for Clarity and Standards-Documentation Strategies for Mixed Human-AI Codebases-Common Security and Reliability Issues in Vibe Code-Ethical Coding Practices in AI-Augmented Environments-Navigating Cultural Shifts in 'Vibe Coding' and Accountability

Activities

- **Tutorial & MCQ:** Refactor example AI-generated code and answer related questions on quality and security.
- **Flipped Classroom:** Study ethical case studies at home and present findings in class.

Applied Projects and Real-World Ai Applications

AI-Powered Task Manager and To-Do Lists-Resume Builder with Smart AI Suggestions-Note-Taking App with Auto-Organization-Recipe Generator with Ingredient Substitution Logic-Visual Memory Game Using AI Tools-Non-diagnostic Health Symptom Checker

Activities

- **Team Project:** Build a working prototype (e.g., to-do list, resume builder) using AI-assisted tools.
- **Presentation & Quiz:** Present the final project and complete a quiz on the design and development process.

Advanced Ai-Driven Web Application Deployment

Planning and Scoping AI-Assisted Applications-Workflow Mapping from Idea to Deployment-Integrating Front-End, Back-End, and AI Logic-Testing and Deploying AI-Augmented Applications-Human Oversight and Continuous Improvement Loops-Project Ownership and Accountability in AI Codebases

Activities

- **Capstone Project:** Design and deploy a full AI-assisted web application as a team.
- **Demo Day & Reflection:** Teams demo their solutions and reflect on human-AI collaboration challenges.

Weightage: Continuous Assessment: 40%, End Semester Theory Examinations: 60%

Assessment Methodology: Assignments (30), Quiz (10), Virtual Demo (25), Flipped Class Room (10), Review of Gate and IES Questions (25)

References:

1. Ted Winston, Mastering Vibe Coding: Build, Debug, and Ship Software with AI Assistants like Cursor, Replit, and GPT, Kindle Edition, April 25, 2025.
2. Ethan Voss, Vibe Coding: Build & Sell Apps Without Coding Experience: How to Use AI to Create Profitable Apps Even If You've Never Written a Line of Code, Kindle Edition, 26 March 2025.
3. Amit Iyer, Beginner's Guide to Vibe Coding: Exploring Features, Use Cases and Understanding Vibe Coding, Paperback, 19 March 2025.
4. Gene Kim, Steve Yegge, Vibe Coding: Building Production-grade Software With GenAI, Chat, Agents, and Beyond, Paperback, 21 October 2025.
5. Addy Osmani, Beyond Vibe Coding, O'Reilly Media, Inc., August 2025. ISBN: 9798341634756

E-Resources

- https://elearn.nptel.ac.in/shop/iit-workshops/ongoing/computer-science/applied-vibe-coding-workshop/?v=13b5bfe96f3e
- https://www.deeplearning.ai/short-courses/vibe-coding-101-with-replit/

CO	Description of CO	PO	PSO
CO1	Describe the fundamental concepts, philosophy, and practices of vibe coding and its role in modern, collaborative, and creative software development.	--	--
CO2	**Analyze** vibe coding workflows, tools, and collaborative practices to understand productivity, creativity, and developer experience.	PO1 (3)	PSO1 (3)
CO3	**Evaluate** vibe coding approaches, platforms, and community-driven practices to assess their effectiveness in real-world software development scenarios.	PO3 (2)	PSO2 (2)

CO4	**Design** software development solutions using vibe coding principles by selecting appropriate tools, collaboration strategies, and creative workflows.	PO2 (1)	PSO1 (3)

CP25C25	**Federated Learning**	L	T	P	C
		3	0	0	3

Course Objective:

1. Understand the fundamentals, architecture, and real-world relevance of Federated Learning.
2. Learn core FL algorithms, privacy techniques, and system challenges in decentralized settings.
3. Apply FL frameworks and tools to build practical applications through hands-on project work.

Introduction to Federated Learning

Definition and Scope – Motivation: Data Privacy, Decentralization, Compliance (GDPR, HIPAA) – Centralized vs. Federated vs. Distributed Learning – Types of FL: Horizontal, Vertical, and Transfer FL – Real-World Applications: Mobile Keyboard (GBoard), Healthcare, IoT – FL System Architecture – Federated Learning Workflow and Lifecycle – Communication Protocols.

Activity

Create a Quiz for understanding Federated Learning Basics

Federated Learning Algorithms

Federated Averaging (FedAvg) – Federated SGD (FedSGD) – FedProx for handling system heterogeneity – Personalized Federated Learning (pFedMe, FedPer, FedBN) – FL with Non-IID Data – Aggregation and Weight Update Strategies – Convergence Issues – Performance Metrics.

Activity

Create a flipped class room for Understanding federated learning algorithms

Privacy and Security in FL

Security Threats in FL – Differential Privacy (DP): Concept, Application in FL – Secure Multiparty Computation (SMC) – Homomorphic Encryption (Overview) – Secure Aggregation Protocols – Byzantine Robustness – Model Inversion Attacks – Data Poisoning – Mitigation Strategies.

Activity

Ask students to submit an assignment on Privacy and security in FL

System & Communication Challenges

Device Heterogeneity – Intermittent Availability – Resource Constraints (Memory, Battery, Network) – Client Selection Strategies – Communication Efficiency: Quantization, Compression, Sparsification – Dropout and Partial Client Participation – FL over Edge Devices – Energy-aware FL.

Activity

Create a flipped class room communication challenges

Applications and Frameworks

FL Applications in Healthcare, Finance, Smart City, Recommender Systems – Case Studies – Tools and Frameworks: TensorFlow Federated, PySyft, Flower – FL Simulators

Activity

Ask students to submit an Assignment on applications of FL

Project Work and Future Directions

Project Work: Building a Simple Federated Learning Model – Model Deployment and Evaluation – Future Directions: Split Learning, Hybrid FL, Federated Analytics, FL + AI/ML

Activity

Ask students to submit simple project on FL

References:

1. Q. Yang, Y. Liu, T. Chen, and Y. Tong, *Federated Learning*, PHI Learning Pvt. Ltd., 2021.
2. M. Mohri, A. Rostamizadeh, and A. Talwalkar, *Foundations of Machine Learning*, 2nd ed., Pearson Education, 2020.
3. K. P. Murphy, *Machine Learning: A Probabilistic Perspective*, Pearson Education, 2017.
4. C. M. Bishop, *Pattern Recognition and Machine Learning*, Oxford University Press, 2016.
5. T. Mitchell, *Machine Learning*, Tata McGraw-Hill Education, 2017.

Weightage:	**Continuous Assessment: 40%** **(i) Activity: 10%** **(ii) Internal Theory Examination: 30%**	**End Semester Theory Examinations: 60%**
Mandated Activities with Marks: Assignments (30), Quiz (10), Virtual Demo (25), Flipped Class Room (10), Review of Gate and IES Questions (25)		
Internal Examinations: Two Tests		

E-resources:

1. https://youtu.be/E3Kn5v-vdUc?si=UMnGqSNOB9NjheTS- NPTEL YouTube– Lecture 42: Introduction to Federated Learning
2. https://github.com/adap/flower- Flower Framework GitHub
3. https://github.com/OpenMined/PySyft- PySyft (OpenMined)

CO	Description of CO	PO	PSO
CO1	Describe the fundamental concepts, architectures, and operational principles of federated learning and privacy-preserving machine learning systems.	--	--
CO2	**Analyze** federated learning algorithms and communication strategies to understand model convergence, privacy, and system performance.	PO1 (3)	PSO1 (3)
CO3	**Evaluate** federated learning frameworks, security mechanisms, and deployment approaches to assess their effectiveness in distributed and sensitive data environments.	PO3 (2)	PSO2 (2)
CO4	**Design** federated learning solutions by selecting appropriate aggregation methods, communication protocols, and privacy techniques for real-world applications.	PO2 (1)	PSO1 (3)