



Maratha Vidya Prasarak Samaj's
Karmaveer Adv. Baburao Ganpatrao Thakare College of Engineering
An Autonomous Institute affiliated to Savitribai Phule Pune University, Pune
Udoji Maratha Boarding Campus, Gangapur Road, Nashik - 422 013,
Maharashtra, India

Syllabus
Third Year B.Tech. Computer Engineering (2024 Pattern)
(As per NEP 2020)
Academic Year 2026-27
(Copy for Student Circulation Only)

Program Specific Outcomes (PSOs).

PSO1: To apply mathematical and Computer Engineering fundamentals.

PSO2: To apply standard practices and strategies for software development and project management.

PSO3: To adapt programming languages, modern computer tools and technologies, and soft skills for career enrichment.

Program Educational Outcomes (PEOs).

PEO1: To inculcate computational and programming skills in the field of Computer Engineering.

PEO2: To prepare the graduates to fulfill professional requirements in industry.

PEO3: To develop the graduates to solve problems related to the society.

Program Outcomes (POs)

PO1: Engineering Knowledge: Apply knowledge of mathematics, natural science, computing, engineering fundamentals and an engineering specialization as specified in WK1 to WK4 to develop to the solution of complex engineering problems.

PO2: Problem Analysis: Identify, formulate, review research literature and analyze complex engineering problems reaching substantiated conclusions with consideration for sustainable development (WK1 to WK4).

PO3: Design/Development of Solutions: Design creative solutions for complex engineering problems and design/develop systems/components/processes to meet identified needs with consideration for the public health and safety, whole-life cost, net zero carbon, culture, society and environment as required (WK5).

PO4: Conduct Investigations of Complex Problems: Conduct investigations of complex engineering problems using research-based knowledge including design of experiments, modelling, analysis & interpretation of data to provide valid conclusions (WK8).

PO5: Engineering Tool Usage: Create, select and apply appropriate techniques, resources and modern engineering & IT tools, including prediction and modelling recognizing their limitations to solve complex engineering problems. (WK2 and WK6).

PO6: The Engineer and The World: Analyze and evaluate societal and environmental aspects while solving complex engineering problems for its impact on sustainability with reference to economy, health, safety, legal framework, culture and environment. (WK1, WK5 and WK7).

PO7: Ethics: Apply ethical principles and commit to professional ethics, human values, diversity and inclusion; adhere to national & international laws. (WK9).

PO8: Individual and Collaborative Team work: Function effectively as an individual, and as a member or leader in diverse/multi-disciplinary teams.

PO9: Communication: Communicate effectively and inclusively within the engineering community and society at large, such as being able to comprehend and write effective

reports and design documentation, make effective presentations considering cultural, language, and learning differences.

PO10: Project Management and Finance: Apply knowledge and understanding of engineering management principles and economic decision-making and apply these to one's own work, as a member and leader in a team, and to manage projects and in multidisciplinary environments.

PO11: Life-Long Learning: Recognize the need for, and have the preparation and ability for i) Independent and life-long learning, ii) Adaptability to new and emerging technologies and iii) Critical thinking in the broadest context of technological change. (WK8).

Knowledge and Attitude Profile (WK)

WK1: A systematic, theory-based understanding of the natural sciences applicable to the discipline and awareness of relevant social sciences.

WK2: Conceptually-based mathematics, numerical analysis, data analysis, statistics and formal aspects of computer and information science to support detailed analysis and modelling applicable to the discipline.

WK3: A systematic, theory-based formulation of engineering fundamentals required in the engineering discipline.

WK4: Engineering specialist knowledge that provides theoretical frameworks and bodies of knowledge for the accepted practice areas is the engineering discipline; much is at the forefront of the discipline.

WK5: Knowledge, including efficient resource use, environmental impacts, whole-life cost, re-use of resources, net zero carbon, and similar concepts, that supports engineering design and operations in a practice area.

WK6: Knowledge of engineering practice (technology) in the practice areas in the engineering discipline.

WK7: Knowledge of the role of engineering in society and identified issues in engineering practice in the discipline, such as the professional responsibility of an engineer to public safety and sustainable development.

WK8: Engagement with selected knowledge in the current research literature of the discipline, awareness of the power of critical thinking and creative approaches to evaluate emerging issues.

WK9: Ethics, inclusive behavior and conduct. Knowledge of professional ethics, responsibilities, and norms of engineering practice. Awareness of the need for diversity by reason of ethnicity, gender, age, physical ability etc. with mutual understanding and respect, and of inclusive attitudes.

Third Year B.Tech. Computer Engineering Curriculum Structure (2024 Pattern) Semester – V

Course Code	Course Type	Course Name	Teaching Scheme (Hrs/Week)			Evaluation Scheme and Marks					Credits			
			TH	PR	TU	CCE	ESE	TW	PR + OR	TOT	TH	PR	TU	TOT
102501	PCC-8	Computer Networks	3	2	-	40	60	25	25	150	3	1	-	4
102502	PCC-9	Machine Learning	2	2	-	40	60	25	25	150	2	1	-	3
102503	PCC-10	Automata Theory	3	-	-	40	60	-	-	100	3	-	-	3
102504	PEC-1	Program Elective Course-1	3	2	-	40	60	25	-	125	3	1	-	4
170505	MDM-3	Multi-Disciplinary Minor-3	3	2	-	40	60	25	-	125	3	1	-	4
102506	VSEC-4	Web Technologies	-	4	-	-	-	50	50	100	-	2	-	2
Total			14	12	0	200	300	150	100	750	14	6	0	20

Course Code	PEC Course – 1
102504A	Systems Programming and Compilers
102504B	Distributed Systems
102504C	Cryptography and Network Security

Third Year B.Tech. Computer Engineering Curriculum Structure (2024 Pattern) Semester – VI

Course Code	Course Type	Course Name	Teaching Scheme (Hrs/Week)			Evaluation Scheme and Marks					Credits			
			TH	PR	TU	CCE	ESE	TW	PR + OR	TOT	TH	PR	TU	TOT
102601	PCC-11	Software Engineering	3	2	-	40	60	25	25	150	3	1	-	4
102602	PCC-12	Design and Analysis of Algorithms	3	2	-	40	60	25	25	150	3	1	-	4
102603	PEC-2	Program Elective Course-2	3	2	-	40	60	25	25	150	3	1	-	4
102604	PEC-3	Program Elective Course-3	3	2	-	40	60	25	25	150	3	1	-	4
170605	MDM-4	Multi-Disciplinary Minor-4	3	-	1	40	60	50	-	150	3	-	1	4
Total			15	8	1	200	300	150	100	750	15	4	1	20

Course Code	PEC Course – 2	Course Code	PEC Course – 3
102603A	Big Data Analytics	102604A	High Performance Computing
102603B	Cloud Computing	102604B	UI/UX Design
102603C	Ethical Hacking	102604C	Quantum Computing

Third Year B.Tech. Multi-Disciplinary Course – 3 and 4
List of Courses (2024 Pattern) Semester – V and VI

S #	Offered by Department	Streams (Choose One Stream Only)	Semester-V		Semester-VI		Semester-VIII		Eligible Student
			MDM – 3 Code	MDM – 3 Courses	MDM – 4 Code	MDM – 4 Courses	MDM – 5 Code	MDM – 5 Courses	
1.	Mechanical Engineering	Automobile Engineering	170505A	Systems in Mechanical Engineering	170605A	Automotive Technology	170805A	Automotive Control System	All, except Mech. Engg. Dept
2.	Computer Engineering	Software Development	170505B	Data Structures and Algorithms	170605B	Database Systems	170805B	Software System Design	All, except Computer Engg., Information Tech. and AI&DS Dept.
3.	Instru. & Control Engg.	Measurement and Control	170505C	Engineering Measurements	170605C	Industrial Control Components	170805C	Industrial Control Applications	All, except Instru. Control Engg. Dept.
4.	Electronics & Tele. Engg.	Internet of Things	170505D	Sensors and Actuators	170605D	Embedded System in IoT	170805D	Applications of IoT	All, except E&TC Department
5.	Civil Engineering	Infrastructure Planning	170505E	Infrastructure and Planning	170605E	Governing Policies	170805E	Transportation Systems	All, except Civil Engg. Department
6.	Information Technology	Web Technology	170505F	Web Development	170605F	Application Development	170805F	Digital Governance	All, except Info. Tech., Computer and AI&DS Dept.
7.	AI&DS	Data Analytics	170505G	Applied Data Science	170605G	Data Visualization	170805G	Business Analytics	All, except AI&DS, Computer and Information Tech. Dept.
8.	Electrical Engineering	Electrical Machines & Drives	170505H	Electrical Machines	170605H	Drives and Controls	170805H	Industrial Applications	All, except Electrical Engg. Dept

- **Summary of Credits and Total Marks:**

Semester	Credits	Marks
V	20	750
VI	20	750
Total	40	1500

- **Definition of Credit :**

The Under Graduate (U.G.) programmes will have credit system. The details of credit will be as follow.

1 Credit = 1 hour/week for lecture
 = 2 hours/week for practical
 = 1 hour/week for tutorial

- **Description of Various Courses:**

Type of Course	Description
PCC	Programme Core Course
PEC	Programme Elective Course
MDM	Multidisciplinary Minor Course
VSEC	Vocational and Skill Enhancement Course (Skill Courses)

Semester - V

Course Code: 102501	Course Name: Computer Networks	
Teaching Scheme	Credit	Evaluation Scheme
Theory : 3 Hours/Week Practical : 2 Hours/Week	3 1	CCE : 40 Marks ESE : 60 Marks TW : 25 Marks PR + OR : 25 Marks

Expected Prerequisite Courses:

- Discrete Mathematics, Operating Systems.

Course Objectives:

- To learn fundamental concepts of computer networks, layered architectures, protocols, and network devices.
- To learn data link layer services such as framing, error control, flow control, and multiple access protocols.
- To understand network layer addressing schemes, routing algorithms, routing protocols, and congestion control.
- To study transport layer protocols including TCP and UDP with emphasis on reliability, flow and congestion control.
- To study application layer protocols and examine advanced networking technologies such as High-Performance Networks, SDN, ICN, and Network Virtualization.

Course Outcomes:

After successful completion of the course, learner will be able to:

CO1: Explain the fundamentals of computer networks, network models, and transmission media.

CO2: Apply error control, flow control, and MAC techniques in data link communication.

CO3: Analyze IP addressing, subnetting, and routing algorithms.

CO4: Evaluate transport layer functions, TCP/UDP services, and congestion control.

CO5: Demonstrate application layer protocols and understand emerging technologies.

Course Contents:

Unit-I: Foundation of Computer Networks

08 Hours

Introduction to Computer Networks, Goals and applications of networks, Types of Networks, Network Topologies, OSI Model – Layers and functions, TCP/IP Protocol Suite.

Physical Layer: Network Hardware and Software, Transmission Media: wired and wireless,

Network Devices.

Case Study: Study of College Campus Network.

Unit-II: Data Link Layer

08 Hours

Data Link Layer design issues, Framing techniques, Error detection and correction, Flow control and Error control mechanisms.

Data Link Protocols: Stop-and-Wait, Go-Back-N, Selective Repeat.

MAC Sub-layer Protocols: HDLC, PPP, CSMA: CSMA/CA, CSMA/CD.

Switching Techniques: Circuit, Packet, Message Switching.

Case Study: Demonstration of DLL Protocol.

Unit-III: Network Layer

08 Hours

Network Layer functions, IPv4 and IPv6 addressing, Subnetting and Supernetting.

Routing Algorithms: Distance Vector, Link State, Dijkstra's Algorithm, Bellman-Ford.

Routing Protocols: RIP, OSPF, BGP.

IP Protocols: ARP, RARP, ICMP, NAT, Mobile IP. Congestion Control approaches.

Case Study: Demonstration of Network Simulator: OMNeT++ /CISCO packet tracer/ NS2.

Unit-IV: Transport Layer

08 Hours

Transport Layer Services, Multiplexing and Demultiplexing, Socket Address, Socket Primitives, Transport Layer Protocol: User Datagram Protocol (UDP), Transmission Control Protocol (TCP): TCP connection management (3-way handshake), Flow control and Congestion control, Reliable data transfer, TCP segment format, Quality of Service (QoS) concepts, SCTP.

Case Study: Network analyzer tool: Wireshark.

Unit-V: Application Layer

08 Hours

Introduction to application layer services, Application Layer Protocols: DNS, Email protocols: SMTP, POP3, IMAP, HTTP, HTTPS, SNMP, DHCP, FTP, TELNET.

Introduction to Advanced Computer Networks.

Case Study: High-Performance Networks/ Information-Centric Networking/ Software-Defined Networking.

Learning Resources:

Text Books:

1. Fourauzan B., "Data Communications and Networking", 4th edition, Tata McGraw Hill, Publications, 2006, ISBN 0 - 07 – 0634145.

2. Kurose, Ross “Computer Networking a Top-Down Approach Featuring the Internet”, Pearson; 6th edition (March 5, 2012), ISBN-10: 0132856204, ISBN-13: 978-0132856201.

Reference Books:

1. Andrew Tenenbaum, “Computer Networks”, 4th Edition, PHI, ISBN81-203-2175-8.
2. Computer Networks: A Top-Down Approach, Behrouz A. Forouzan, Firouz Mosharraf, McGraw-Hill Education.
3. Larry L. Peterson and Bruce S. Davie, Computer Networks - A Systems Approach 5th Edition, Morgan Kaufmann, Elsevier, 2011.
4. An Engineering Approach to Computer Networks - S. Keshav, 2nd Edition, Pearson Education.

Web link for MOOC / NPTEL Links:

1. Computer Networks and Internet Protocol by Prof. Soumya Kanti Ghosh, Prof. Sandip Chakraborty, IIT Kharagpur.
https://onlinecourses.nptel.ac.in/noc26_cs35/preview
2. Advanced Computer Networks by Prof. Neminath Hubballi, IIT Indore, Prof. Sameer G Kulkarni, IIT Gandhi Nagar.
https://onlinecourses.nptel.ac.in/noc26_cs60/preview
3. Networking Basics by Cisco Networking Academy.
<https://www.cisco.com/site/us/en/learn/topics/small-business/networking-basics.html>

Suggested Activity-based Class Learning:

1. Flipped Classroom.
2. OSI Layer Role Play.
3. Simulation of routing protocol.
4. Poster Presentation.
5. Quizzes/Assignment.

Guidelines for Practicals:

1. The course instructor should design assignments by considering the required background knowledge, relevant technologies, practical applications and current trends related to the course.
2. Preferably, there should be multiple sets of assignments and distributed among batches of students.
3. Giving assignments based on real-life problems can help the students with future projects.
4. Students should be encouraged to use free and open-source software.

5. Instructors can also include an additional assignment that goes beyond the syllabus, but aligns with the course.
6. Instructors should include a course project to be implemented by a group of 2 or 3 students. It should be based on different topics studied in the course. Instructors should frame proper guidelines for the effective implementation of the course project.

Course Code: 102502	Course Name: Machine Learning	
Teaching Scheme	Credit	Evaluation Scheme
Theory : 2 Hours/Week Practical : 2 Hours/Week	2 1	CCE : 40 Marks ESE : 60 Marks TW : 25 Marks PR + OR : 25 Marks

Expected Prerequisite Courses:

- Basics of Statistics and Probability.

Course Objectives:

- To understand the basic concepts of machine learning and apply them for the various problems.
- To learn various machine learning types and use it for the various machine learning tasks.
- To optimize the machine learning model and generalize it.
- To understand the need for multi-class classifiers.
- To learn the working of clustering algorithms.
- To learn fundamental neural network algorithms.

Course Outcomes:

After successful completion of the course, learner will be able to:

CO1: Identify the needs and challenges of machine learning for real time applications.

CO2: Apply regression and classification techniques to solve machine learning problems.

CO3: Use appropriate supervised machine learning algorithms for different applications.

CO4: Compare and contrast different clustering algorithms.

CO5: Analyze the performance and architecture of neural networks for solving engineering problems.

Course Contents:**Unit-I: Foundation of Machine Learning****06 Hours**

Introduction to Machine Learning: Concepts, real-world applications, Architecture of ML. Data Types: Numerical, Categorical, Feature Selection, Data Pre-processing Techniques, and Dataset Preparation, Types of Learning: Supervised, Unsupervised and Reinforcement learning.

Case Study: Personalised Retail Strategy for E-Commerce.

Unit-II: Supervised Learning – Classification and Regression**06 Hours**

Regression: Regression and its type, Cost Functions: MSE, MAE, R^2 , over-fitting and under-fitting, Bias-Variance Trade-off, Gradient Descent Algorithm.

Classification: Binary and Multiclass Classification, Logistic Regression and Support Vector Machines. Evaluation Measures.

Case Study: Predictive Modeling for House Prices and Disease Diagnosis.

Unit-III: Non-Linear Machine Learning Classifiers**06 Hours**

Non-Linear Models: Decision Trees, Probabilistic Models: Bayes Theorem, Naïve Bayes Classifier, K-Nearest Neighbor, Ensemble Learning.

Case Study: Spam Filtering and Document Classification.

Unit-IV: Unsupervised Learning**06 Hours**

Types of Clustering: Hierarchical, Agglomerative clustering and Divisive clustering, Partition Clustering- K-means clustering, Elbow Method, Outliers, Associative rule Learning.

Case Study: Customer Segmentation and Market Basket Analysis.

Unit-V: Introduction to Neural Networks**06 Hours**

Neural Network Fundamentals: Biological vs Artificial Neuron, McCulloch–Pitts Model, Perceptron Learning Rule, Multi-Layer Perceptron, Activation Functions, Loss functions and back propagation.

Case Study: Image Feature Extraction and Classification using Multi-Layer Perceptron and Back propagation.

Learning Resources:**Text Books:**

1. Saikut Dutt, Subramanian Chandramouli and Amit Kumar das, Machine Learning (Pearson).
2. Nikhil Buduma, Fundamentals of Deep Learning: A Classroom Approach (OReilly).

Reference Books:

1. Ian H. Witten , Data Mining: Practical Machine Learning Tools and Techniques (Morgan Kaufmann Series in Data Management Systems).

Web link for MOOC / NPTEL Links:

1. Introduction to Machine Learning – Prof. P. K. Bora, IIT Guwahati.
<https://nptel.ac.in/courses/106/106/106106139>
2. Machine Learning – Prof. Balaraman Ravindran, IIT Madras.
<https://nptel.ac.in/courses/106/106/106106202>

3. Applied Machine Learning – IIT Madras.
<https://nptel.ac.in/courses/106106213>
4. Neural Networks and Deep Learning – DeepLearning.AI.
<https://www.coursera.org/specializations/deep-learning>

Suggested Activity-based Class Learning:

1. Flipped Classroom.
2. OSI Layer Role Play.
3. Simulation of routing protocol.
4. Poster Presentation.
5. Quizzes/Assignment.

Guidelines for Practicals:

1. The course instructor should design assignments by considering the required background knowledge, relevant technologies, practical applications and current trends related to the course.
2. Preferably, there should be multiple sets of assignments and distributed among batches of students.
3. Giving assignments based on real-life problems can help the students with future projects.
4. Students should be encouraged to use free and open-source software.
5. Instructors can also include an additional assignment that goes beyond the syllabus, but aligns with the course.
6. Instructors should include a course project to be implemented by a group of 2 or 3 students. It should be based on different topics studied in the course. Instructors should frame proper guidelines for the effective implementation of the course project.

Course Code: 102503	Course Name: Automata Theory	
Teaching Scheme	Credit	Evaluation Scheme
Theory : 3 Hours/Week	3	CCE : 40 Marks ESE : 60 Marks

Expected Prerequisite Courses:

- Discrete Mathematics.

Course Objectives:

- Introduce the mathematical foundations of automata theory.
- Study different computational models like finite automata, regular expressions, context-free grammar, push-down automata, and Turing machines.
- Identify and design appropriate automata according to the type of language.
- Correlate different computational models to real-world applications.

Course Outcomes:

After successful completion of the course, learner will be able to:

CO1: Construct finite automata and its variants to represent regular languages.

CO2: Apply regular expressions to represent regular languages.

CO3: Simplify context free grammars for regular and context free languages.

CO4: Examine the functionality of pushdown automata for recognizing context free languages.

CO5: Analyze the working of Turing Machines to solve computational problems for different requirements.

Course Contents:**Unit-I: Finite Automata****08 Hours**

Understanding formal language, the essentials of translation, and language representation.

Finite Automata (FA): Introduction, Acceptance of a string and a language, Deterministic Finite Automata (DFA), Nondeterministic Finite Automata (NFA), Equivalence of NFA and DFA, NFA with epsilon transition and its significance, Minimization of DFA.

FA with Output: Moore Machine, Mealy Machine, Equivalence of Moore and Mealy Machines, and Interconversion.

Case Study: Traffic Light Control System, Vending Machine Controller.

Unit-II: Regular Expressions**08 Hours**

Introduction, Operators of Regular Expressions (RE), Precedence of operators, Algebraic laws

for RE, Language to Regular Expressions, Equivalence of two REs.

Equivalence of RE and FA: RE to NFA with epsilon moves conversion, DFA to RE Conversion using Arden's Theorem, Pumping Lemma for Regular Languages and its applications, Applications of RE and FA.

Case Study: Validation of user email formats using regular expressions.

Unit-III: Context Free Grammars

08 Hours

Context Free Grammars (CFG): Introduction, Derivation Process and Tree, Writing grammar for languages, Context Free Languages (CFL), Ambiguous CFG, Removal of ambiguity, Simplification of CFG, Normal Forms: Chomsky Normal Form and Greibach Normal Form; Chomsky Hierarchy.

Case Study: Modeling of if-else, while, and block structures in any programming language; XML and Document Type Definitions.

Unit-IV: Pushdown Automata

08 Hours

Introduction, FA vs Pushdown Automata (PDA), PDA accepting CFLs, Instantaneous Description, Acceptance of CFL by Empty Stack and Final State, Deterministic PDA vs Nondeterministic PDA, Equivalence of CFG and PDA, Closure Properties of CFLs, Application of PDA for parsing, Comparison between FA and PDA.

Case study: Pattern recognition, Post machine.

Unit-V: Turing Machines

08 Hours

Introduction, Instantaneous Description, Techniques for Turing Machines (TM) Construction, Design of TM, Complexity of TM, Universal TM, TM Variants; Concepts of Halting Problem, Church-Turing Hypothesis, Recursively Enumerable Languages, Linear Bounded Automata.

Case Study: Compiler design tools: Lex and Yacc.

Learning Resources:

Text Books:

1. Vivek Kulkarni "Theory of Computation", Oxford University Press, 2013, ISBN-13 978-0198084587.
2. John Martin, "Introduction to Languages and The Theory of Computation", 2nd Edition, McGraw-Hill Education, 4th edition, 2010, ISBN-13: 978-0073191461.

Reference Books:

1. John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman, "Introduction to Automata Theory Languages and Computation", Pearson, 3rd Edition, 2008, ISBN-13: 978-8131720479.

Web link for MOOC / NPTEL Links:

1. Theory of Computation, By Prof. Raghunath Tewari, IIT Kanpur.
https://onlinecourses.nptel.ac.in/noc21_cs83/preview

Suggested Activity-based Class Learning:

1. Flipped Classroom.
2. Gamification.
3. Simulation Tools.
4. Collaborative and Individual Problem-based learning.
5. Quizzes/Assignments.
6. Vlab.

Course Code: 102504A	Course Name: Systems Programming and Compilers	
Teaching Scheme	Credit	Evaluation Scheme
Theory : 3 Hours/Week Practical : 2 Hours/Week	3 1	CCE : 40 Marks ESE : 60 Marks TW : 25 Marks

Expected Prerequisite Courses:

- Data Structures.

Course Objectives:

- To develop an understanding of system software components, machine architecture, language processors, and assembly language translation mechanisms.
- To introduce the concepts and design of macro processors, loaders, and linkers used in system software.
- To understand lexical and syntax analysis and explain the concepts of token recognition, parsing, syntax-directed translation, and error handling.
- To analyze different intermediate code representations, basic code generation techniques, and optimization methods for improving program efficiency.

Course Outcomes:

After successful completion of the course, learner will be able to:

CO1: Explain the concepts, functions, and design principles of system software components and assemblers.

CO2: Apply conditional macro expansion, two-pass macro processor design, and loader-linker concepts in system software.

CO3: Use parsing techniques to analyze grammars.

CO4: Apply syntax-directed translation, syntax error handling, and recovery mechanisms in parser design.

CO5: Analyze basic blocks, flow graphs, and code optimization techniques for efficient code generation.

Course Contents:**Unit-I: Introduction to Systems Programming****08 Hours**

Introduction to System Software, Overview of Language Processors: Assembler, Compiler, Interpreter, Loader, Linker, Macro Processor, Debugger, Machine Language and Assembly Language, Fundamentals of Language Processing, Symbol Table, Literal Table, Pass structures

of Assemblers, Design of Two Pass Assembler.

Case Study: GCC, Debuggers (GDB, IDE debuggers), MASM, TASM.

Unit-II: Macro Processor, Linkers and Loaders

08 Hours

Introduction: Macro Definition and Call, Macro Expansion, Nested Macro Calls, Conditional Macro Expansion, Design of Two pass Macro Processor.

Loader Schemes: Compile and Go, General Loader Scheme, Absolute Loader, Subroutine Linkages, Relocating Loaders, Direct Linking Loader, Relocation and Linking Concept, Self-Relocating Programs.

Case Study: A Payroll Processing Program Using Macro Processor.

Unit-III: Lexical Analysis and Top-Down Parsing

08 Hours

Phases of a Compiler, Lexical analysis: role of lexical analyzer, tokens, patterns and lexemes, lexical errors, Specification of Tokens: Regular Expressions, Regular Definitions, Recognition of Tokens using Transition Diagrams, Syntax Analysis: Role of Parser, Syntax Error Handling, Top-Down Parsing: Recursive-Descend Parsing, LL(1) Parser.

Case Study: Lexical-Analyzer Generator Lex/Flex.

Unit-IV: Bottom-Up Parsing and Syntax-Directed Translation

08 Hours

Introduction, Shift-Reduce Parsing, Conflicts During Shift-Reduce Parsing, LR Parsing: Simple LR, Constructing SLR-Parsing Tables, LR(0), Canonical LR(1), Basic error handling and recovery mechanisms. **Syntax-Directed Translation:** Syntax-Directed Definitions, Syntax-Directed Translation

Case Study: YACC/BISON.

Unit-V: Intermediate-Code Generation and Optimization

08 Hours

Intermediate Code Generation: Variants of Syntax Trees, Three-Address Code, Code Generation: Issues in the Design of a Code Generator, Basic Blocks and Flow Graphs, Optimization of Basic Blocks, a Simple Code Generator. **Code Optimization:** Peephole Optimization, the Principal Sources of Optimization

Case Study: Optimization of a Mathematical Expression.

Learning Resources:

Text Books:

1. D. M. Dhamdhere, Systems Programming and Operating Systems, McGraw Hill Education.

2. Alfred V. Aho, Monica S. Lam, Ravi Sethi & Jeffrey D. Ullman, “Compilers: Principles, Techniques, and Tools (2nd Edition)”, Pearson Education, ISBN-13: 978-0321486813.

Reference Books:

1. Leland L. Beck, System Software: An Introduction to Systems Programming, 3rd Edition, Pearson Education.
2. Andrew W. Appel, “Modern Compiler Implementation in C”, Cambridge University Press, ISBN13: 978-0521607650.
3. John J. Donovan, Systems Programming, Tata McGraw-Hill, ISBN-13: 978-0-07-460482-3.

Web link for MOOC / NPTEL Links:

1. Compiler Design By Prof. Santanu Chattopadhyay
https://onlinecourses.nptel.ac.in/e-learning/preview/noc20_cs13
2. Coursera – Compilers (Stanford University)
<https://www.coursera.org/learn/compilers>
3. Udemy – Compiler Design
<https://www.udemy.com/course/compilerdesign/>

Guidelines for Practicals:

1. The course instructor should design assignments by considering the required background knowledge, relevant technologies, practical applications and current trends related to the course.
2. Preferably, there should be multiple sets of assignments and distributed among batches of students.
3. Giving assignments based on real-life problems can help the students with future projects.
4. Students should be encouraged to use free and open-source software.
5. Instructors can also include an additional assignment that goes beyond the syllabus, but aligns with the course.
6. Instructors should include a course project to be implemented by a group of 2 or 3 students. It should be based on different topics studied in the course. Instructors should frame proper guidelines for the effective implementation of the course project.

Course Code: 102504B	Course Name: Distributed Systems	
Teaching Scheme	Credit	Evaluation Scheme
Theory : 3 Hours/Week Practical : 2 Hours/Week	3 1	CCE : 40 Marks ESE : 60 Marks TW : 25 Marks

Expected Prerequisite Courses:

- Data Structures, Operating Systems.

Course Objectives:

- Apply the fundamentals and working principles of distributed systems to solve basic computing problems.
- Apply communication mechanisms and coordination techniques used in distributed operating systems.
- Analyze replication and fault-tolerance techniques used in distributed systems.
- Analyze distributed file systems and distributed applications used in distributed systems.
- Evaluate security mechanisms and emerging trends in distributed systems.

Course Outcomes:

After successful completion of the course, learner will be able to:

- CO1:** Use the basic concepts of distributed systems to understand their models, challenges, trends, and real-world applications.
- CO2:** Analyze replication and fault tolerance techniques used in distributed systems for reliability and recovery.
- CO3:** Analyze replication and fault tolerance techniques used in distributed systems for reliability and recovery.
- CO4:** Analyze distributed file systems, name services, and distributed applications used in modern distributed systems.
- CO5:** Assess security mechanisms, access control techniques, and emerging trends in distributed systems.

Course Contents:**Unit-I: Characterizations of Distributed Systems****08 Hours**

Introduction: Definition of Distributed System, Examples of Distributed Systems, Goals of Distributed Systems, Types of Distributed Systems, Challenges, Trends in Distributed

Systems. **System Models:** Architectural Model, Fundamental Model, CAP Theorem.

Case Study: Video Streaming Platform as a Distributed System.

Unit-II: Communication and Coordination

08 Hours

Communication: Introduction, Layered protocols and Protocol Stacks, Types of communication, Inter-process Communication, Remote Procedure Call (RPC), Message oriented communication, Multicast Communication, Virtualization and It's role in distributed Systems, **Coordination:** Clock Synchronization issues, Physical Clocks, Logical Clocks, Mutual Exclusion algorithms, Election algorithms, Distributed event matching, Gossip Based coordination, Preference for an odd number of nodes, Majority consensus and split-brain avoidance, **Observability in Distributed Systems:** Tracing, Metrics, and Logging.

Case Study: ZooKeeper Coordination Service / Raft Consensus in Modern Systems.

Unit-III: Replication and Fault Tolerance

08 Hours

Replication: Need and advantages of replication, Replica placement, Replica management, Failure masking through replication, Data consistency models, Data management and consistency mechanisms, Consistency protocols, Caching and Replication in Web-based Systems, **Fault Tolerance:** Introduction, Failure models, Reliable client server communication, Reliable group communication, Distributed commit protocols, Recovery techniques.

Case Study: Web Caching and CDN Replication.

Unit-IV: Distributed File Systems and Applications

08 Hours

Distributed File Systems: Introduction, Design Goals and Challenges of Distributed Storage, File service architecture, Sun Network File System, Hadoop Distributed File System (HDFS). **Name Services:** Introduction, Name services and the Domain Name System, Directory services, **Distributed Applications:** Overview of distributed applications, peer to peer applications, Content distribution systems.

Case study: Domain Name System (DNS), BitTorrent Peer-to-Peer Distribution.

Unit-V: Distributed Security And Emerging Trends

08 Hours

Introduction: Security Threats, Policies and Mechanisms, Design Issues, Cryptography. **Secure Channels:** Authentication, Message Integrity and Confidentiality, Secure Group Communication. **Access Control:** General Issues in Access Control, Firewalls, Secure Mobile Code, Denial of Service (DOS). **Security Management:** Key Management, Secure Group Management, Authorization Management, Emerging Trends in Distributed Systems

Case Study: Cloud Service-Oriented Architecture, Edge and Fog Computing in IoT.

Learning Resources:**Text Books:**

1. Andrew S. Tanenbaum, Maarten Van Steen, Distributed Systems: Principles and Paradigms, Pearson Education.
2. George Coulouris, Jean Dollimore, Tim Kindberg, Distributed Systems: Concepts and Design, Pearson Education.
3. M. Kleppmann, Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems. Sebastopol, CA, USA: O'Reilly Media, 2017.

Reference Books:

1. Pradeep K. Sinha, Distributed Operating Systems, PHI Learning.
2. Singhal and Shivaratri, Advanced Concepts in Operating Systems, McGraw-Hill.

Web link for MOOC / NPTEL Links:

1. Distributed Systems By Prof. Rajiv Misra, IIT Patna.
<https://nptel.ac.in/courses/106106168>
https://onlinecourses.nptel.ac.in/noc22_cs66/preview
2. Distributed Systems and Cloud Computing with Java.
<https://www.udemy.com/course/distributed-systems-cloud-computing-with-java/>

Guidelines for Practicals:

1. The course instructor should design assignments by considering the required background knowledge, relevant technologies, practical applications and current trends related to the course.
2. Preferably, there should be multiple sets of assignments and distributed among batches of students.
3. Giving assignments based on real-life problems can help the students with future projects.
4. Students should be encouraged to use free and open-source software.
5. Instructors can also include an additional assignment that goes beyond the syllabus, but aligns with the course.
6. Instructors should include a course project to be implemented by a group of 2 or 3 students. It should be based on different topics studied in the course. Instructors should frame proper guidelines for the effective implementation of the course project.

Course Code: 102504C	Course Name: Cryptography and Network Security	
Teaching Scheme	Credit	Evaluation Scheme
Theory : 3 Hours/Week Practical : 2 Hours/Week	3 1	CCE : 40 Marks ESE : 60 Marks TW : 25 Marks

Expected Prerequisite Courses:

- Discrete Mathematics, Introduction to Cyber Security, Computer Networks, Operating Systems.

Course Objectives:

- Use fundamental concepts and mechanisms of cryptography and network security.
- Demonstrate encryption and decryption algorithms for data confidentiality.
- Apply key management, authentication, and digital signature techniques.
- Apply network security protocols like SSL, IPsec, and VPNs.
- Analyze cryptographic techniques to design secure communication systems.

Course Outcomes:

After successful completion of the course, learner will be able to:

CO1: Demonstrate security concepts and cryptographic techniques in secure communication systems.

CO2: Apply encryption algorithms such as DES, Triple DES, AES, RC5, and IDEA.

CO3: Apply the RSA algorithm for key generation, encryption, and decryption in secure communication.

CO4: Implement digital signatures for data authenticity and non-repudiation.

CO5: Analyze modern network security architectures and their applications.

Course Contents:

Unit-I: Introduction to Cryptography

08 Hours

Security goals: Confidentiality, Integrity, Authentication, Authorization, Non-repudiation,

Types of attacks: Active and Passive attacks, Security services and mechanisms,

Cryptography: Terminologies and objectives, **Classical encryption techniques:** Substitution and transposition cipher principles, Stream cipher vs block cipher concept, **Classical**

Cryptanalysis: Frequency analysis and brute-force search, **Modern Cryptanalysis:** Differential and linear cryptanalysis, Meet-in-the-middle attacks, Birthday attacks on hash functions, Side-channel cryptanalysis, Statistical and pattern-based attacks.

Case Study: Security Breach in an Online Banking System.

Unit-II: Symmetric Key Cryptography

08 Hours

Block cipher design principles, Feistel structure, **DES (Data Encryption Standard)**: structure, key generation, rounds Triple DES, Limitations of DES, **AES (Advanced Encryption Standard)**: rounds, key expansion, structure RC5, IDEA overview, **Modes of operation**: ECB, CBC, CFB, OFB, CTR, Key distribution problem and random number generation.

Case Study: Secure E-Commerce Payment System.

Unit-III: Asymmetric Key Cryptography

08 Hours

Principles of public-key cryptosystems, RSA algorithm: key generation, encryption/decryption, security analysis, Mathematical foundations: Modular arithmetic, GCD, Euler's theorem, Totient function, Diffie-Hellman key exchange algorithm, Comparison of symmetric and asymmetric encryption, Introduction to Elliptic Curve Cryptography. Case Study: Secure Communication System for Online Government Services.

Unit-IV: Message Authentication and Hash Functions

08 Hours

Message authentication requirements and functions, **Hash functions**: MD5, SHA-1, SHA-256, SHA-512. Message Authentication Code (MAC), Digital signatures and digital certificates, Public Key Infrastructure (PKI), X.509 certificates and certification authorities.

Case Study: Secure Digital Document Verification System (E-Governance Portal).

Unit-V: Network Security Practices

08 Hours

Authentication protocols: Kerberos, Email security: PGP (Pretty Good Privacy), S/MIME IP security: IPsec architecture, AH and ESP protocols, Web security: SSL/TLS protocols Firewalls: types (packet-filtering, stateful, proxy), design goals, Intrusion Detection Systems (IDS), Virtual Private Networks (VPNs)

Case Study: Case Study: Secure Enterprise Network System.

Learning Resources:

Text Books:

1. W. Stallings, Cryptography and Network Security: Principles and Practice, 6th ed. Noida, India: Pearson Education.
2. A. Kahate, Cryptography and Network Security, 3rd ed. New Delhi, India: McGraw Hill Education.

Reference Books:

1. B. A. Forouzan and D. Mukhopadhyay, Cryptography and Network Security, 3rd ed. New Delhi, India: McGraw Hill Education.

Web link for MOOC / NPTEL Links:

1. Cryptography and Network Security By Prof. Sourav Mukhopadhyay, IIT Kharagpur.
<https://nptel.ac.in/courses/106105162>
2. Virtual lab.
<https://ns-amrt.vlabs.ac.in/List%20of%20experiments.html>

Guidelines for Practicals:

1. The course instructor should design assignments by considering the required background knowledge, relevant technologies, practical applications and current trends related to the course.
2. Preferably, there should be multiple sets of assignments and distributed among batches of students.
3. Giving assignments based on real-life problems can help the students with future projects.
4. Students should be encouraged to use free and open-source software.
5. Instructors can also include an additional assignment that goes beyond the syllabus, but aligns with the course.
6. Instructors should include a course project to be implemented by a group of 2 or 3 students. It should be based on different topics studied in the course. Instructors should frame proper guidelines for the effective implementation of the course project.

Course Code: 170505A	Course Name: Systems in Mechanical Engineering	
Teaching Scheme	Credit	Evaluation Scheme
Theory : 3 Hours/Week Practical : 2 Hours/Week	3 1	CCE : 40 Marks ESE : 60 Marks TW : 25 Marks

Expected Prerequisite Courses:

- Engineering Physics, Engineering Chemistry.

Course Objectives:

- To identify the sources of energy and their conversions.
- To explain the basic concept of engineering thermodynamics and its application.
- To introduce proper manufacturing process applicable to produce components.
- To understand the basic concepts of mechanical engineering.

Course Outcomes:

After successful completion of the course, learner will be able to:

CO1: Apply knowledge of thermodynamics concepts.

CO2: Explain basic laws of thermodynamics, heat transfer and their applications.

CO3: Select different materials for different applications.

CO4: Explain the basics of engineering components.

CO5: Describe and get exposure to the latest trends in manufacturing processes.

Course Contents:**Unit-I: Thermodynamic Concepts****08 Hours**

Introduction to thermodynamics, introduction to various thermodynamic properties, heat and work, thermodynamic laws. Power plant engineering and sources of energy : conventional thermal, nuclear, hydraulic power plant. Nonconventional- wind, solar, tidal, geothermal.

Unit-II: Heat Transfer**08 Hours**

Modes of heat transfer: conduction, convection and radiation. Fourier's law, Newton's law of cooling and Stefan-Boltzmann's law. Domestic appliances, viz., refrigerator, air-conditioner.

Unit-III: Material Science**08 Hours**

Classification of materials, their properties and applications, phase change material (PCM), composite material.

Unit-IV: Introduction to Transmission Devices**08 Hours**

Study of shafts, belt drive, chain drive, gear and gear trains and clutches.

Unit-V: Manufacturing Processes**08 Hours**

Metal casting, forging, sheet metal working, machining and machine tools, and metal joining processes.

Learning Resources:**Text Books:**

1. Nag, P. K., "Engineering Thermodynamics" Tata McGraw-Hill Publisher Co. Ltd.
2. Chaudhari and Hajra, "Elements of Workshop Technology", Volume I and II, Media Promoters and Publishers, Mumbai.
3. Rajput, R.K., "Basic Mechanical Engineering", Laxmi Publications Pvt. Ltd.
4. Raghavan V., "Materials Science And Engineering: A First Course", Hall of India.

Reference Books:

1. Khan, B. H., "Non-Conventional Energy Sources, Tata McGraw-Hill Publisher Co Ltd.
2. Khurmi, R.S, and Gupta, J. K., "A Textbook of Thermal Engineering", S. Chand & Sons.
3. Incropera, F. P. and Dewitt, D.P., "Fundamentals of Heat and Mass Transfer, John Wiley and Sons, USA.
4. Juvinal, R. C., Fundamentals of Machine Component Design, John Wiley & Sons, USA.
5. Kripal Sing, "Automobile Engineering", Vol.1 and Vol.2.

Web link for MOOC / NPTEL Links:

1. Energy resources. <https://www.youtube.com/watch?v=Zgp86PVXXuQ>
2. Thermodynamic Concepts. <https://nptel.ac.in/courses/112103275>
3. <https://www.youtube.com/playlist?list=PLbRMhDVUMngeVrxtbBzn8HvP8KAWBpI5>

List of Practicals: (Any 5)

1. Study of non-conventional or renewable sources of energy.
2. Study of laws of thermodynamics.
3. Study of laws of heat transfer.
4. Study of domestic appliances like the refrigerator and the air conditioner.
5. Study of materials used for different applications.
6. Study of power transmission devices.
7. Study of manufacturing processes.

Course Code: 170505B	Course Name: Data Structures and Algorithms	
Teaching Scheme	Credit	Evaluation Scheme
Theory : 3 Hours/Week Practical : 2 Hours/Week	3 1	CCE : 40 Marks ESE : 60 Marks TW : 25 Marks

Expected Prerequisite Courses:

- Fundamentals of Programming, Programming and Problem Solving.

Course Objectives:

- To learn the fundamentals of data structures.
- To understand how data can be logically organized using data structures.
- To illustrate the types of data structures and their use in engineering.
- To choose suitable data structures for real engineering problems.
- To learn system efficiency, such as time and memory usage, without mathematics or programming.

Course Outcomes:

After successful completion of the course, learner will be able to:

CO1: Understand basic concepts of data structures.

CO2: Apply the concepts of stacks and queues.

CO3: Implement appropriate non-linear data structures.

CO4: Demonstrate searching and sorting techniques in engineering data handling.

CO5: Illustrate real-world engineering case studies using data structures.

Course Contents:

Unit-I: Introduction to Data Structures

08 Hours

Need for data organization, Overview of Data structures: Types of data structures, **Abstract Data Types (ADT):** Concept and significance, **Arrays:** Characteristics and representation, types of arrays: 1D and 2D, Use in storing engineering measurements and parameters.

Case Study: Engineering Data Management in a Weather Monitoring System.

Unit-II: Linear Data Structures

08 Hours

Introduction to linear data structures, **Stack:** Concept, operations, and behavior (LIFO), Applications: undo operations and automation systems. **Queue:** Concept, operations, and behavior (FIFO), Applications: scheduling, buffering, and resource management, Conceptual comparison between stack and queue, **Linked Lists:** Introduction, need, types and applications.

Case Study: Smart Office Management System.**Unit-III: Non-Linear Data Structures****08 Hours**

Need for non-linear data representation, Trees: Terminology: root, node, parent, child, and level, Types of trees: Binary tree, Binary search tree, Tree traversal, Applications of Non-linear Data Structures, Graphs: Basic concepts, Concept of directed and undirected graphs, graph representation. Case Study: College Campus Management System.

Unit-IV: Searching and Sorting**08 Hours**

Importance of searching and sorting, **Searching techniques:** Concept of linear and binary search, Use cases in engineering, **Sorting techniques:** Concept of ordering and prioritization, Examples such as sorting by size, cost and priority using bubble sort, selection sort, merge sort, Time and Space Complexity Concepts. **Case Study:** Engineering Inventory and Component Management System.

Unit-V: Engineering Case Studies**08 Hours**

Data structures used in real time applications, Identification of case studies from Mechanical / Civil / Electrical systems etc., Selection of appropriate data structures for given engineering problems, and discussion on the benefits of proper data structuring.

Learning Resources:**Text Books:**

1. Y. Kanetkar and A. Kanetkar, Let Us C++, New Delhi, India: BPB Publications.
2. E. Horowitz and S. Sahni, Fundamentals of Data Structures in C++, New York, NY, USA: W. H. Freeman, ISBN: 978-0716782926.

Reference Books:

1. M. A. Weiss, Data Structures and Algorithm Analysis in C++, 2nd ed., New Delhi, India: Pearson Education, ISBN: 81-7808-670-0.

Web link for MOOC / NPTEL Links:

1. Data Structures And Algorithms By Prof. Naveen Garg, IIT Delhi.
<https://nptel.ac.in/courses/106102064>

List of Practicals:

1. The course instructor should design assignments by considering the required background knowledge, relevant technologies, practical applications, and current trends related to the subject.
2. Preferably, there should be multiple sets of assignments and distributed among batches of students.
3. Giving assignments based on real-life problems can help the students with future projects.
4. Students should be encouraged to use free and open-source software.
5. Instructors can also include an additional assignment or mini-project that goes beyond the syllabus but aligns with the course.

List of Suggested Assignments:

1. Implementation of an array.
2. Implementation of stack.
3. Implementation of queue.
4. Implementation of trees.
5. Implementation of graphs.
6. Implementation of searching techniques.
7. Implementation of sorting techniques.
8. Mini project.

List of Suggested Activity-based Learning:

1. Flipped Classroom.
2. Gamification.
3. Online Interactive Tool.
4. Collaborative and Individual Problem-based learning.
5. Quizzes/Assignments.
6. VLab.

Course Code: 170505C	Course Name: Engineering Measurements	
Teaching Scheme	Credit	Evaluation Scheme
Theory : 3 Hours/Week Practical : 2 Hours/Week	3 1	CCE : 40 Marks ESE : 60 Marks TW : 25 Marks

Expected Prerequisite Courses:

- Engineering Physics, Engineering Chemistry, Basic Electronics and Electrical Engineering.

Course Objectives:

- To explain the static/dynamic characteristics of measurement systems and the working principles of displacement, speed, vibration, and force/torque transducers.
- To impart knowledge on industrial pressure and vacuum measurement techniques along with standard calibration methods.
- To explain the construction, signal compensation, and operational characteristics of contact and non-contact temperature sensors.
- To introduce the mechanics, and selection criteria of various fluid level and flow monitoring instruments.
- To familiarize students with the principles of viscosity, density, humidity, and chemical sensors, alongside an introduction to intelligent SMART sensing systems.

Course Outcomes:

After successful completion of the course, learner will be able to:

- CO1:** Evaluate the performance of transducers used for mechanical parameters like displacement, speed, vibration, force, and torque.
- CO2:** Calibrate pressure and vacuum measuring devices using standard dead-weight or vacuum testing rigs.
- CO3:** Select appropriate temperature sensors for precise thermal monitoring.
- CO4:** Evaluate the operational dynamics of industrial fluid flow and level sensors based on principles.
- CO5:** Demonstrate miscellaneous sensors according to their working principles.

Course Contents:**Unit-I: Fundamentals of Engineering Measurements****08 Hours**

Measurement system elements, static and dynamic characteristics.

Need for sensors and transducers, definition and classification of transducers.

Displacement Measurement: Resistive-potentiometers, inductive-LVDT, capacitive, piezoelectric, ultrasonic, and proximity sensors.

Speed measurement, vibration measurement, force and torque measurement.

Unit-II: Pressure Measurement**08 Hours**

Units and their relations, manometers, elastic sensors, piezoelectric secondary transducers, differential pressure sensors, capacitive (delta cell), vacuum gauges, dead weight tester.

Unit-III: Temperature Measurement**08 Hours**

Temperature scales, units and their relations, classification of temperature sensors, bimetallic thermometer, Resistance temperature detectors (RTD), types of RTD, lead wire compensation, thermistors, Thermocouples, thermopiles, thermos-well, cold junction compensation techniques, pyrometers, temperature IC sensor LM35.

Unit-IV: Flow Measurement**08 Hours**

Bernoulli's equation for incompressible flow, head type flow meters (orifice, venture meter and pitot tube), variable area type (Rotameter), turbine, electromagnetic, ultrasonic, mass flow meter: Coriolis flow meter.

Unit-V: Level and Miscellaneous Measurement**08 Hours**

Level Measurement: Float, DP cell, ultrasonic, capacitive.

Viscosity: Saybolt, Searle's rotating cylinder, rotameter. **Density:** Hydrometers. **Humidity:** Resistive and capacitive type sensors. PH sensors, conductivity sensors.

Learning Resources:**Text Books:**

1. Instrumentation Measurement and Analysis – B.C. Nakra, K.K. Chaudhry, 4th Edition, McGraw Hill, 2010, ISBN: 978-0070709868
2. Principles of Industrial Instrumentation" — D. Patranabis, ISBN (latest editions): 978 0070605502 (approximate for recent editions)
3. Transducers and Instrumentation – D.V.S. Murty, 3rd Edition, PHI Learning, 2011, ISBN: 978-8120346876.

Reference Books:

1. Industrial Instrumentation – A.K. Sawhney, 4th Edition, Dhanpat Rai & Co., 2010, ISBN: 978-8177000311
2. Measurement Systems: Application and Design – Ernst O. Doebelin, 5th Edition, McGraw Hill, 2003, ISBN: 978-0071223122.

Web link for MOOC / NPTEL Links:

1. https://onlinecourses.nptel.ac.in/e-learning/preview/noc23_ee105
2. https://onlinecourses.nptel.ac.in/e-learning/preview/noc21_ee32

List of Practicals (Any 8):

1. Measure linear displacement using a Linear Variable Differential Transformer (LVDT).
2. Determine the characteristics of different proximity sensors.
3. Calibrate a load cell and determine its sensitivity for weight measurement.
4. Measure angular displacement using a shaft angle encoder.
5. Calibrate a Bourdon tube pressure gauge using a dead-weight tester.
6. Calibrate a vacuum gauge using a vacuum gauge tester.
7. Calibrate a Resistance Temperature Detector (RTD) and determine the temperature of an unknown liquid.
8. Determine temperature using LM35.
9. Flow measurement using DP transmitter.
10. Measure pH sensor output and calibrate it using standard buffer solutions.
11. Flow measurement using Orifice / Venturi flowmeter.

Course Code: 170505D	Course Name: Sensors and Actuators	
Teaching Scheme	Credit	Evaluation Scheme
Theory : 3 Hours/Week Practical : 2 Hours/Week	3 1	CCE : 40 Marks ESE : 60 Marks TW : 25 Marks

Expected Prerequisite Courses:

- Fundamentals of Electronic Engineering, Programming Language C.

Course Objectives:

- To enable students to apply IoT concepts, architectures (3-layer and 5-layer), and system components (sensors, actuators, devices) in designing basic IoT-based solutions for real-world applications.
- To analyse different types of sensors (such as temperature, pressure, proximity, and optical sensors) and their applications in real-world systems.
- To study various actuators (electrical, mechanical, hydraulic, and pneumatic) and understand their role in control systems.
- To develop the ability to interface sensors and actuators with microcontrollers and embedded systems for data acquisition and control.
- To apply knowledge of sensors and actuators in designing and implementing practical systems used in automation, robotics, and industrial applications.

Course Outcomes:

After successful completion of the course, learner will be able to:

- CO1:** Apply knowledge of IoT building blocks, architectures, and components to develop and implement simple IoT applications for real-world applications.
- CO2:** Analyze sensor/transducer performance, signal conditioning, and data acquisition to assess measurement accuracy and errors.
- CO3:** Apply principles of temperature, pressure, force, flow, and level sensors to select and use appropriate sensors for measurement applications.
- CO4:** Examine the operating principles and performance characteristics of position, proximity, and motion sensors to compare and select appropriate sensors for specific measurement and control applications.
- CO5:** Evaluate performance, working principles and control mechanisms of electrical, fluid power, and specialized actuators for various automation and control system applications.

Course Contents:**Unit-I: Introduction to IoT and Architecture****08 Hours**

Definition, Need & Scope of IoT, Evolution of IoT, Industry 4.0, Basic Components of IoT: Sensors, Actuators, Devices, IoT System Building Blocks, IoT Architecture: 3-Layer, 5-Layer models (simple explanation), IoT Applications across various sectors, (healthcare, manufacturing, smart homes, agriculture, transport).

Unit-II: Introduction to Signal Conditioning**08 Hours**

Concepts: Sensors vs. Transducers, Sensor Classification (Active/Passive, Digital/Analog), Performance Characteristics (Static and Dynamic: Sensitivity, Linearity, Hysteresis, Response Time). **Signal Conditioning:** Bridge circuits, Amplifiers (Op-Amp), Filtering (Low-pass, High-pass), Data Acquisition (ADC/DAC). **Error Analysis:** Calibration, Sensitivity Analysis, and Uncertainty.

Unit-III: Thermal and Mechanical Sensors**08 Hours**

Temperature Sensors: Thermocouples (Seebeck effect), Resistance Temperature Detectors (RTDs), Thermistors (NTC/PTC), Bimetallic strips, Pyrometers.

Pressure Sensors: Manometer, Bourdon Tube, Diaphragm, Bellows, Piezoelectric and Strain Gauge.

Force & Weight Sensors: Strain Gauges (Bridge circuits, gauge factor), Load Cells.

Flow & Level Sensors: Ultrasonic, Capacitive, Float Sensors, Turbine Flowmeters, Venturi Tubes.

Unit-IV: Position, Proximity and Motion Sensors**08 Hours**

Position and Displacement Sensors: Potentiometers (Linear/Rotary), Linear Variable Differential Transformers (LVDT), Eddy current probes.

Proximity Sensors: Inductive (metallic), Capacitive (non-metallic), Optical/Photoelectric, Magnetic (Reed switches), Light Dependent Resistors (LDR), Fiber-optic sensors.

Motion & Acceleration Sensors: Accelerometers (1-3 axis), Gyroscopes, MEMS inertial sensors, Velocity sensors.

Unit-V: Actuation Systems**08 Hours**

Electric Actuators: DC Motors, AC Motors, Stepper Motors, Servo Motors, Solenoids, Relays.

Fluid Power Actuators: Hydraulic and Pneumatic Cylinders, Valves, Pumps.

Specialized Actuators: Piezoelectric actuators, Shape Memory Alloys.

Control Methods: PWM (Pulse Width Modulation) speed control.

Learning Resources:**Text Books:**

1. Milan Milenkovic, “Internet of Things: Concepts and System Design”, Springer, 2020.
2. D. V. S. Murthy, “Transducers and Instrumentation” Prentice Hall of India, 2nd Edition, 2012.
3. Adrian McEwen, Hakim Cassimally, “Designing the Internet of Things”, Wiley, 2014
4. Rolando Herrero, “Fundamentals of IoT Communication Technologies”, Springer, 2022.

Reference Books:

1. Ian Sinclair & John Dunton, “Sensors and Transducers”, Newnes (Elsevier) 3rd Edition, 2016.
2. Samayveer Singh, Manju, Aruna Malik, Pradeep Kumar Singh, “IoT-enabled Sensor Networks: Architecture, Methodologies, Security, and Futuristic Applications”, Bentham Science Publishers, 2024.
3. C.S. Rangarajan, “Instrumentation Devices and Systems”, Tata McGraw-Hill, 1st Edition, 2010.
4. Mohan Lal Kolhe, Kailash J. Karande, Sampat G. Deshmukh, “Artificial Intelligence, Internet of Things (IoT) and Smart Materials for Energy Applications”, CRC Press (Taylor & Francis), 2023.

Web link for MOOC / NPTEL Links:

1. https://onlinecourses.nptel.ac.in/noc25_ee76/preview
2. https://onlinecourses.nptel.ac.in/noc25_ee82/preview
3. https://onlinecourses.nptel.ac.in/noc26_ee41/preview

List of Practicals: (Any 8)

1. Interfacing of an LED with Arduino board.
2. Display real-time temperature & humidity values on serial monitor using DHT11.
3. IoT-Based Motion Detection / Security System (Interface PIR Motion Sensor. Send alert notification (app pop-up, buzzer or LED indicator).
4. IoT-Based Gas/Smoke Detection System. (Interface MQ-2 Gas sensor (or simulator). Trigger buzzer/alert message when gas level exceeds a threshold. Send alert notification to mobile using cloud app (Blynk/MQTT).
5. IoT-Controlled Servo Motor / Door Lock System. (Interface a servo motor. Control angle/position remotely using a cloud dashboard or mobile app.)
6. Heartbeat / Pulse Monitoring System (IoT-Based). (Interface pulse sensor or simulated heartbeat data. Send health parameters to cloud for real-time monitoring.)

7. Water Level Monitoring & Alert System. (Use ultrasonic sensor to measure water tank level. Send water-level data to cloud and generate alerts when level is low or high.)
 8. Fire Detection and Alert System. (Flame sensor or temperature sensor combination. Trigger instant alerts and send real-time fire detection signals to mobile/cloud.)
 9. Voice-Controlled IoT Device (Google Assistant/Alexa Integration). Control LED/Fan simulation using voice commands through Google Assistant + IFTTT or Alexa IoT skills.
 10. IoT-Based Smart Street Lighting System. Automatically control a street light (LED) using an LDR sensor and send status (ON/OFF) to an IoT dashboard.
 11. Smart Energy Monitoring System. Measuring and visualizing electrical parameters (current/voltage simulation allowed) and uploading data to cloud.
 12. IoT-Based Weather Station (Multi-Sensor System). Interfacing two or more sensors (temperature, humidity, light, pressure) and publishing combined data to cloud.
- .

Course Code: 170505E	Course Name: Infrastructure and Planning	
Teaching Scheme	Credit	Evaluation Scheme
Theory : 3 Hours/Week Practical : 2 Hours/Week	3 1	CCE : 40 Marks ESE : 60 Marks TW : 25 Marks

Expected Prerequisite Courses:

- Fundamentals of Building Technology and Architectural Planning.

Course Objectives:

- To understand and analyze urban infrastructure systems, including their types, planning data, standards, and the roles of stakeholders in urban development.
- To apply concepts of water supply, sanitation, stormwater, solid waste, and transport planning to propose sustainable and efficient urban infrastructure solutions.
- To develop knowledge of rural planning principles and government schemes to improve village infrastructure, housing, livelihoods, and social services.
- To use modern planning tools and technologies (GIS, remote sensing, drones) for the implementation, monitoring, management of urban and rural infrastructure projects.

Course Outcomes:

After successful completion of the course, learner will be able to:

- CO1:** Explain the role and types of urban infrastructure, key planning data and standards, and the planning process with stakeholder roles.
- CO2:** Apply concepts of water supply, sanitation, stormwater, and solid waste management to plan and propose basic urban infrastructure improvements.
- CO3:** Apply transport and land-use planning concepts to improve urban mobility, accessibility, and sustainability
- CO4:** Develop rural planning strategies using government schemes to improve village infrastructure, housing, and livelihoods.
- CO5:** Implement rural infrastructure and connectivity solutions using GIS and other planning tools.

Course Contents:

Unit-I: Introduction to Urban Planning and Infrastructure 08 Hours

Role of infrastructure in urban development, Types of infrastructure: physical (water, power, transport), social (education, health), utilities Planning data, resource analysis, norms and standards Infrastructure planning process and stakeholder roles layouts.

Unit-II: Water, Sanitation and Stormwater Systems**08 Hours**

Water supply sources, treatment, storage, distribution, Water demand forecasting, leakage, reuse, rainwater harvesting, Sanitation systems: sewage generation, collection, treatment, greywater management, Stormwater drainage, natural drainage, stormwater harvesting Institutional and management issues, Municipal solid waste: generation, collection, treatment, recycling.

Unit-III: Transport and Mobility Planning**08 Hours**

Importance of transport infrastructure in urban form transport modes (public, private, non-motorized), and mobility demand land use–transport integration, accessibility, TOD (Transit-Oriented Development) transport modeling basics, pricing, congestion and sustainable mobility.

Unit-IV: Principles and Strategies of Rural Planning**08 Hours**

Introduction to rural planning: need, scope, and objectives, rural settlement patterns and typologies, planning strategies for rural development central and State government schemes: MGNREGA, PMAY-G, Swachh Bharat Mission, PMGSY, Jal Jeevan Mission, role of planning in livelihood, housing, and social infrastructure.

Unit-V: Rural Infrastructure Management**08 Hours**

Physical infrastructure: water supply, sanitation, drainage, roads, energy, solid waste management, social infrastructure: schools, healthcare, community centers, Transport and connectivity: rural road hierarchy, public transport, traffic management, Tools and technology: GIS, remote sensing, drones for rural planning, Implementation, monitoring, and governance of rural projects.

Learning Resources:**Text Books:**

1. Town Planning, G. K. Hiraskar, Dhanpat Rai Publications.
2. Town Planning, S. C. Rangwala, Charotar Publishing House Pvt. Ltd.
3. Rural Development: Planning Strategy & Policy Imperatives - Vivek Saurath & Manoj Agarwal).

Reference Books:

1. Urban Planning by Chris Couch.
2. Urban Infrastructure and Governance by G. Ramesh, Vishnu Prasad Nagadevara, Gopal Naik, Anil Suraj.
3. Urban and Regional Planning Education: Learning for India (edited by Ashok Kumar, Diwakar S. Meshram, Krishne Gowda.

4. MRTP Act 1966 : The director, government printing, stationary and publications, Maharashtra state, Mumbai.
5. URDPFI & AMRUT Guidelines: Ministry of housing and urban affairs, Government of India.
6. Role of Edge Analytics on Sustainable Smart City Development: Challenges and Solutions by G. R. Kanagachidambaresan.
7. Sustainable Smart Cities in India: Challenges and Future Perspectives (The Urban Book Series) by Poonam Sharma and Swati Rajput.
8. Rural Development: Principles, Policies and Development by Katar Singh and Anil Shishodia. Atlantic Publications.

Web link for MOOC / NPTEL Links:

1. https://onlinecourses.swayam2.ac.in/cec20_ar01/preview
2. <https://www.edx.org/certificates/professional-certificate/delftx-inclusive-and-sustainable-cities?index=product&queryId=2d5f1360201b4959787e19d485c4178e&position=3>
3. <https://www.coursera.org/learn/smart-cities>
4. https://onlinecourses.nptel.ac.in/noc25_ar27/preview
5. https://onlinecourses.nptel.ac.in/noc26_ar08/preview
6. https://onlinecourses.nptel.ac.in/noc25_ar25/preview
7. https://onlinecourses.nptel.ac.in/noc26_ce48/preview

List of Practicals:

1. Prepare a case study for any infrastructure project of India and give presentation. (Group Activity).
2. Design a practical rainwater harvesting solution for any building. (Individual work).
3. Prepare a land-use map for a neighborhood with classification (residential, commercial, industrial, etc.) by calculation. (Group Activity)
4. Identify the major causes of congestion on any busy road in the city and suggest practical improvements to reduce traffic delays. Present your findings with brief observations and supporting photos or sketches. (Group work)
5. Draw a simple O–D diagram of your route from home to college, marking key points like home, major junctions, bus stops, and the college gate. Include the approximate distance and travel time with neat sketch map with brief travel details. (Individual work)
6. Select one rural development scheme (MGNREGA, PMAY-G, SBM, PMGSY, or JJM) and prepare a brief report covering its objectives, beneficiaries, key components, impact on rural livelihood or infrastructure, and a simple implementation flowchart (Individual work)

7. Prepare a village infrastructure improvement proposal for a hypothetical village of 1,000–1,500 people, listing existing issues, suggesting improvements in water supply, sanitation, housing, livelihood, and roads, and identifying relevant government schemes for implementation. (Group work)
8. Site visit to any urban infrastructure or rural infrastructure management project.

Course Code: 170505F	Course Name: Web Development	
Teaching Scheme	Credit	Evaluation Scheme
Theory : 3 Hours/Week Practical : 2 Hours/Week	3 1	CCE : 40 Marks ESE : 60 Marks TW : 25 Marks

Expected Prerequisite Courses:

- Computer Networks.

Course Objectives:

- To familiarize students with Web Programming basic concepts.
- To learn and understand Web scripting languages.
- To introduce students to responsive web design frameworks and modern UI components.
- To provide knowledge of web page styling and layout design for effective user interfaces.
- To familiarize students with website development using no-code platforms.

Course Outcomes:

After successful completion of the course, learner will be able to:

CO1: Understand a basic of web application.

CO2: Develop structured web pages using HTML.

CO3: Apply styling and layout design principles to create visually appealing web interfaces.

CO4: Develop responsive and user-friendly web pages using modern design frameworks and layout approaches.

CO5: Create and publish functional websites using modern website development tools.

Course Contents:

Unit-I: Introduction to Web Applications

08 Hours

Internet, Basic internet protocols, World Wide Web, HTTP Request message, HTTP response message, Client–Server Architecture, Frontend vs Backend, World Wide Web, Role of browsers and web servers, Introduction to W3C and web standards.

Unit-II: Web Essentials and Mark-up Language- HTML

08 Hours

HTML: Introduction, history and versions. Tags, elements, and attributes, HTML elements: headings, paragraphs, line break, links, frames, lists, tables, images and forms, Input elements and basic validation, difference between HTML and HTML5.

Unit-III: CSS Styling and Layout Design**08 Hours**

CSS: Introduction to Style Sheet, CSS features, Types of CSS, CSS core: Properties, Classes, Child-Class (Nested CSS), Colors, Text, Background, Border, Margin, Padding, Positioning (flex, grid, inline, block), Animation, Transition.

Unit-IV: Bootstrap for Responsive Web**08 Hours**

Bootstrap vs plain CSS, Installing and using Bootstrap, Bootstrap grid system, Responsive design using Bootstrap, **Common Bootstrap Classes:** Buttons, Tables, Forms, Cards, Navigation bar, Alerts and Models.

Unit-V: Website Design Using No-Code Platforms**08 Hours**

Introduction to no-code and low-code web development and Advantages, Website design using **Wix Studio:** Page structure and layout, Responsive design tools, Customization of UI components, **Website design using WordPress:** Themes and templates, Page builders, Content management basics, Publishing and maintaining websites.

Learning Resources:**Text Books:**

1. Jeffrey C. Jackson, "Web Technologies: A Computer Science Perspective", Second Edition, Pearson Education, 2007, ISBN 978-0131856035.
2. Robert W. Sebesta, "Programming the World Wide Web", 4th Edition, Pearson Education.
3. Jon Duckett, HTML and CSS: Design and Build Websites, Wiley Publications.

Reference Books:

1. Marty Hall, Larry Brown, "Core Web Programming", Second Edition, Pearson Education, 2001, ISBN 978-0130897930.
2. HTML5 & CSS3 Complete Reference", McGraw-Hill.

Web link for MOOC / NPTEL Links:

1. https://onlinecourses.swayam2.ac.in/e-learning/preview/ntr26_ed79

List of Practicals:

1. Create a simple HTML webpage that displays basic information about yourself including name, photo, educational details, hobbies, and contact information using headings, paragraphs, images, and hyperlinks, ordered and unordered list.
2. Develop a webpage that displays a student marksheet using HTML tables including subjects, marks, and total percentage.

3. Create an HTML form for student registration including fields such as name, email, gender, date of birth, course selection, and submit/reset buttons. Apply basic input validation.
4. Design a webpage using external CSS to demonstrate styling of text, background colors, borders, and fonts also make use CSS Flexbox to arrange elements such as header, navigation bar, content section, and footer.
5. Design a webpage demonstrating CSS animation and transition effects on buttons, images, or text elements.
6. Create a responsive webpage using Bootstrap that includes navigation bar, image banner, cards, alerts, buttons, and tables and content sections.
7. Design a responsive webpage using Bootstrap Grid System that adjusts layout for mobile, tablet, and desktop screens.
8. Develop and publish a basic website using a No-Code platform such as WordPress or Wix Studio that includes at least three pages (Home, About, Contact).
9. Mini Project: Design and develop a complete responsive website for a small business, institute, or portfolio using HTML, CSS, and Bootstrap or a No-Code platform. The website should include navigation, forms, images and a responsive layout.

Course Code: 170505G	Course Name: Applied Data Science	
Teaching Scheme	Credit	Evaluation Scheme
Theory : 3 Hours/Week Practical : 2 Hours/Week	3 1	CCE : 40 Marks ESE : 60 Marks TW : 25 Marks

Expected Prerequisite Courses:

- Programming and Problem Solving.

Course Objectives:

- To introduce the fundamentals of Data Science.
- To apply Data Science in different domain.
- To do exploratory analysis on a given data.
- To understand the concept of regression models.
- To facilitate learning of data pre-processing using R Programming.

Course Outcomes:

After successful completion of the course, learner will be able to:

CO1: Apply statistics and probability methods to solve real-life problems.

CO2: Demonstrate the Data Science Process and how its components interact.

CO3: Apply basic tools (plots, graphs, summary statistics) to carry out EDA.

CO4: Solve real life problems using regression techniques.

CO5: Apply machine learning algorithms for real world applications.

Course Contents:

Unit-I: Probability and Statistics

08 Hours

Basics: sample space, outcomes, probability. Events: mutually exclusive, Conditional probability, Bayes' rule. Describing a Single Set of Data, Correlation and Causation Probability: Dependence and Independence, Conditional Probability, Random Variables, Continuous Distributions, The Normal Distribution. **Case Study:** Probability Analysis of Credit Card Fraud Detection.

Unit-II: Introduction to Data Science and Data Preprocessing

08 Hours

Defining data science and big data, Recognizing the different types of data, Machine Learning Definition and relation with Data Science, Data Science Process: Data collection; Data preparation; Training a model on the data, evaluation of the model performance, Data visualization techniques and inferences. **Case Study:** Data Preprocessing for Online Shopping

Recommendation System.

Unit-III: Exploratory Data Analysis (EDA)**08 Hours**

Statistical measures, Basic tools (plots, graphs and summary statistics) of EDA, Data Analytics Lifecycle, Discovery Developing Initial Hypotheses, Identifying Potential Data Sources, EDA case study, testing hypotheses on means, proportions and variances. **Case Study:** Analysis of Food Delivery Trends.

Unit-IV: Regression models**08 Hours**

Simple linear regression, least-squares principle, MLR, logistic regression, Multiple correlation, Partial correlation, Cross Validation, Overfitting, Under Fitting and Model Selection, Prediction by using Ridge Regression. **Case Study:** House Price Prediction Using Real Estate Data.

Unit-V: Introduction to Machine Learning & R Programming**08 Hours**

Supervised learning, Unsupervised learning, Association Rule mining, Naive Bayes Classifier, k-means clustering, Introduction to R, R packages, Tidy data, Tabular data and data import, Strings and regular expressions. **Case Study:** Customer Segmentation in Retail Business.

Learning Resources:**Text Books:**

1. Data Science from Scratch: Joel Grus, O'Reilly Media Inc. 2015, ISBN: 9781491901427.
2. R for data science: import, tidy, transform, visualize, and model data", Wickham, Hadley, and Garrett Grolemund. O'Reilly Media, Inc.", 2016.
3. Cathy O'Neil and Rachel Schutt, "Doing Data Science", O'Reilly, 2015, ISBN: 9781449358655.

Reference Books:

1. Introduction to Probability and Statistics for and Engineers and Scientists, S. Ross, 3rd Edition, Elsevier, 2004.
2. Software for Data Analysis: Programming with R (Statistics and Computing), John M. Chambers, Springer.
3. Hadley Wickham, Garrett Grolemund, "R for Data Science", O'Reilly Media, 12 December 2016, ISBN:9781491910344, 1491910348.

Web link for MOOC / NPTEL Links:

1. Python for Data Science
https://onlinecourses.nptel.ac.in/noc20_cs36/preview

List of Practicals:

1. Download any standard dataset from Kaggle. Using a programming language such as Python /R, perform statistical analysis to understand the structure, distribution, and relationships within the data.
2. Write python code that loads any standard dataset downloaded from Kaggle and plot the graph.
3. Perform some basic data cleaning operations on standard dataset using Python / R (Dataset should contain minimum 1000 records)
4. Perform data preprocessing, exploratory analysis on any standard dataset (dataset should contain minimum 1000 records) Using Python /R.
5. Perform correlation and regression analysis for any standard dataset downloaded from Kaggle, Government databases etc using Python / R.
6. Perform Simple Linear regression on for any standard data set downloaded from Kaggle, UCI, Government databases Using Python / R
7. Perform data analysis using Python / R / Weka for readily available data set using any one machine learning algorithm.

Course Code: 170505H	Course Name: Electrical Machines	
Teaching Scheme	Credit	Evaluation Scheme
Theory : 3 Hours/Week Practical : 2 Hours/Week	3 1	CCE : 40 Marks ESE : 60 Marks TW : 25 Marks

Expected Prerequisite Courses:

- Basic Electrical Engineering.

Course Objectives:

- To understand selection of machines for specific applications.
- To understand the construction, principle of operation of various motors.
- To test & analyze the performance of single phase and three phase motors.

Course Outcomes:

After successful completion of the course, learner will be able to:

CO1: Understand the principles of electromechanical energy conversion, torque equation, and constructional features of electric motors.

CO2: Apply characteristics and speed control methods of DC motors to select suitable drives for applications.

CO3: Examine performance characteristics of three-phase induction motors for selection and control in mechanical loads.

CO4: Compare single-phase motors to select appropriate types for domestic and light-industrial use.

CO5: Apply operating principles and control methods of synchronous, BLDC, stepper, and servo motors in engineering applications.

Course Contents:**Unit-I: Fundamentals of Electric Motors****08 Hours**

Basic concepts of electromechanical energy conversion, Fleming's rules, torque and EMF equations. Constructional features: Stator, rotor, windings, bearings, cooling systems, Comparison of AC vs DC motors and their industrial relevance, Safety considerations & motor nameplate data interpretation.

Unit-II: DC Motors**08 Hours**

Construction: Lap and wave windings, commutator, field excitation, Working principle and back EMF, Types: shunt, series, compound motors, Torque-speed characteristics and performance curves, Speed control techniques (armature and field control), Applications in traction, lifts, rolling mills, robotics. Brushless DC motors: construction, electronic commutation, uses in EV and drones.

Unit-III: Three-Phase Induction Motors**08 Hours**

Construction and working: squirrel cage vs slip-ring, Rotating magnetic field concept, slip, torque production, Torque-speed characteristics: starting torque, breakdown torque, Starting methods: DOL, star-delta, auto-transformer, Speed control: V/f control, rotor resistance, slip power recovery, Selection criteria for mechanical drives (load type, duty cycle), Use in pumps, compressors, crushers, conveyors, HVAC systems.

Unit-IV: Single-Phase Induction Motors & Auxiliary Motors**08 Hours**

Limitations of single-phase supply, double-revolving field theory, Types: split-phase, capacitor-start, capacitor-run, shaded pole, Universal motor: working and uses, Applications in domestic & light-industrial mechanical equipment (fans, washing machines, small tools).

Unit-V: Synchronous & Special Motors**08 Hours**

Synchronous Motors: excitation, torque-angle characteristics, Starting methods, V-curves, power factor improvement. **PMSM:** Construction, electronic controller, uses in EV. **Stepper Motors:** Variable reluctance & PM types, step angle, control, Servo motors for automatic control systems & CNC automation, Application focus in robotics, precision positioning and aerospace actuators.

Learning Resources:**Text Books:**

1. Edward Hughes “Electrical Technology”, ELBS, Pearson Education.
2. Ashfaq Husain, “Electrical Machines”, Dhanpat Rai & Sons.
3. S. K. Bhattacharya, “Electrical Machine”, Tata McGraw Hill publishing Co. Ltd, 2nd Edition.
4. Nagrath & Kothari, “Electrical Machines”, Tata McGraw Hill.
5. Bhag S Guru, Husein R. Hiziroglu, “Electrical Machines”, Oxford University Press.
6. K Krishna Reddy, “Electrical Machines- I and II”, SCITECH Publications (India) Pvt. Ltd. Chennai.

Reference Books:

1. I. J. Nagrath and D. P. Kothari, Electric Machines, Tata McGraw Hill, New Delhi, 2005.
2. M. G. Say, The performance and design of alternating current machines, CBS Publishers and distributors, Delhi, 1983.
3. Fitzgerald, Kingsley and Umans, Electric Machinery, Tata McGraw Hill, New Delhi, 2003
4. S. K. Sen, Electrical Machinery, Khanna Pub., Delhi, 2012.
5. Mukherjee and Chakravorty, Electrical Machines, Dhanpat Rai Pub., New Delhi, 2005.

Web link for MOOC / NPTEL Links:

1. https://onlinecourses.nptel.ac.in/noc19_ee69/preview
2. <https://archive.nptel.ac.in/courses/108/102/108102146/>

List of Practicals (Any 8):

1. Load test on DC shunt motor to obtain torque-speed, efficiency and performance characteristics.
2. Speed control of DC motor using armature voltage and field control methods.
3. No-load and blocked-rotor tests on 3-phase induction motor to determine equivalent circuit parameters.
4. Load test on 3-phase induction motor to obtain performance characteristics.
5. Study and comparison of DOL, star-delta & auto-transformer starters for induction motor.
6. V/f (open-loop) speed control of induction motor using inverter (simulation or hardware).
7. Load test on synchronous motor to plot V-curves and study power-factor control.
8. Performance characterization of BLDC motor (simulation or hardware kit).
9. Stepper motor stepping and micro-stepping experiment to determine step angle and positioning accuracy.
10. Servo motor position control using PID controller and performance analysis.
11. Thermal/run-down and insulation resistance test (theoretical analysis for motor selection).
12. Motor selection and design mini-project based on given mechanical load (simulation/theoretical).

Course Code: 102506	Course Name: Web Technologies	
Teaching Scheme	Credit	Evaluation Scheme
Practical : 4 Hours/Week	2	TW : 50 Marks PR + OR : 50 Marks

Expected Prerequisite Courses:

- Fundamentals of Programming, Database Management System.

Course Objectives:

- To understand fundamentals of modern web technologies and web application development.
- To design responsive and interactive web pages using HTML, CSS, and JavaScript.
- To develop Single Page Applications using ReactJS.
- To apply server-side scripting and database connectivity using PHP and Node.js.
- To design and develop full-stack web applications for real-world problems.

Course Outcomes:

After successful completion of the course, learner will be able to:

CO1: Explain the fundamentals of HTML, CSS, responsive design, and accessibility concepts.

CO2: Develop interactive web applications using JavaScript, DOM manipulation, and API integration.

CO3: Analyze Single Page Applications using ReactJS concepts and hooks.

CO4: Analyze server-side web applications using PHP, MySQL, and CRUD operations.

CO5: Design and develop dynamic full-stack web applications using Node.js, Express, databases, and APIs.

Guidelines for Practicals:

- The course instructor should design assignments by considering the required background knowledge, relevant technologies, practical applications and current trends related to the course.
- Preferably, there should be multiple sets of assignments and distributed among batches of students.
- Giving assignments based on real-life problems can help the students with future projects.
- Students should be encouraged to use free and open-source software.
- Instructors can also include an additional assignment that goes beyond the syllabus, but aligns with the course.

6. Instructors should include a course project to be implemented by a group of 2 or 3 students. It should be based on different topics studied in the course. Instructors should frame proper guidelines for the effective implementation of the course project.

Learning Resources:

Text Books:

1. Robert W. Sebesta, Programming the World Wide Web, 8th Edition, Pearson Education, 2021.
2. F. Max Coller, Web Development: Full Stack, 1st Edition, Cengage Learning, 2024.

Reference Books:

1. Kumar, A., Web Technologies, CRC Press, 2019.
2. Artemij Fedosejev and Adam Boduch, React 16 Essentials, 2nd Edition, Packt Publishing, 2018.
3. Ethan Brown, Web Development with Node and Express, O'Reilly Media, 2019.
4. Robin Nixon, Learning PHP, MySQL & JavaScript, O'Reilly Media, 2021.

Web link for MOOC / NPTEL Links:

1. Web Technology by Dr. Ashutosh Kumar Bhatt.
https://onlinecourses.swayam2.ac.in/nou24_cs09/preview
2. Web Technologies by Dr. S. Sasirekha.
https://onlinecourses.swayam2.ac.in/ntr25_ed123/preview

List of Suggested Practicals:

1. Design a basic webpage using HTML to demonstrate headings, lists, and images.
2. Develop a webpage using semantic HTML tags.
3. Apply CSS (inline, internal, external) to style a webpage.
4. Design a responsive webpage using Flexbox, CSS Grid, and media queries.
5. Apply CSS Positions & other properties.
6. Link JS file and apply basic events, array functions.
7. Apply JS frontend functionalities and form validations.
8. Create a basic Single Page Application using React with functional components and JSX.
9. Develop a ReactJS application demonstrating props, state, hooks, event handling, and conditional rendering.
10. Develop a webpage to fetch and display data from an API using Fetch and JSON.
11. Create a webpage to demonstrate DOM manipulation, event handling.
12. Develop a PHP application demonstrating form handling, validation, string manipulation, and session management.
13. Develop a PHP-MySQL application to perform database connectivity and CRUD operations.

14. Develop a Node.js and Express application to create a web server, implement routing and serve static resources
15. Develop a REST API using Node.js, Express, and database connectivity.
16. Course Project.

Semester - VI

Course Code: 102601	Course Name: Software Engineering	
Teaching Scheme	Credit	Evaluation Scheme
Theory : 3 Hours/Week Practical : 2 Hours/Week	3 1	CCE : 40 Marks ESE : 60 Marks TW : 25 Marks PR + OR : 25 Marks

Expected Prerequisite Courses:

- Fundamentals of Programming, Operating Systems, Data Structures, Database Management System.

Course Objectives:

- To introduce systematic software development approaches
- To develop an understanding of project management principles and systematic requirements analysis.
- To introduce architectural principles and styles for designing robust software systems.
- To enable the modeling of software systems using standard UML diagrams.
- To introduce software testing techniques and quality assurance practices.
- To develop practical skills through real-world case studies.

Course Outcomes:

After successful completion of the course, learner will be able to:

CO1: Explain software engineering principles and process models.

CO2: Apply requirement engineering techniques to manage functional and non-functional requirements.

CO3: Use suitable software architecture for software development.

CO4: Develop UML models to represent system structure and behavior.

CO5: Analyze test cases using appropriate testing techniques.

Course Contents:

Unit-I: Introduction to Software Engineering

08 Hours

Software Engineering: Definition, Characteristics, Software Crisis and Evolution, Software Engineering Ethics, Software Process Framework, SDLC, Process Models, Comparison and Selection of Process Models. Agile Manifesto and Principles, Overview of Scrum and Kanban (Jira Tool), Agile vs. Traditional Models, Introduction to DevOps culture, Emerging Trends in Software Engineering.

Case Study: Case studies of software failures and successes.

Unit-II: Requirements Engineering

08 Hours

Requirement Engineering: Types of Requirements (Functional, Non-Functional), Process, Requirement Validation and Management. User Stories and Acceptance Criteria, Requirement Traceability Matrix. Software Requirement Specification, Model-based requirement engineering, Software configuration management.

Software project management fundamentals, Project feasibility analysis, Project planning and scheduling, Software cost estimation techniques: LOC and Function Point (FP), COCOMO, Risk management: identification, analysis, mitigation.

Case Study: Denver Airport Baggage System and Hospital Management System.

Unit-III: Software Architecture

08 Hours

Introduction to Software Architecture, Architectural design principles and patterns, Design patterns vs Architecture patterns, **Modern software Engineering techniques and architectures:** Layered, Client-Server, Micro services, MVC, Cloud-native architecture, Server less architecture, API-First Design, Test Driven Development (TDD), Behaviour Driven development (BDD), Domain Driven Development, Infrastructure as a Code (IaC), CI/CD Development, Introduction to Figma Tool.

Case study: Netflix Micro services and Online Food Delivery System.

Unit-IV: Software Modelling and Modern Software Development

08 Hours

UML: Introduction to UML and its importance, Structural diagrams and Behavioural diagrams UML tools and notations, Role of UML in Agile projects, Mapping UML models to code.

Modern Software Development Techniques: AI Integration, Cloud Platforms, Frontend Technologies, Backend Technologies, Databases, Observability and Monitoring, Cybersecurity Technologies.

Case study: Online Shopping System and Online Banking System.

Unit-V: Software Testing

08 Hours

Fundamentals of software testing, Verification and Validation, Levels of testing: Unit, Integration, System, Acceptance, Testing techniques: Black box and White box testing, Test case design and test data preparation, Introduction to automated testing tools.

Modern Testing architectures: Test Automation Pyramid (Modernized), API-First Testing Architecture, Cloud-Native Test Infrastructure, Micro services-Aware Test Architecture, Event-Driven and Asynchronous Systems Testing, Test Data Management Architecture, Continuous Testing in CI/CD Architecture, Software Quality Concepts, Software Quality

Assurance (SQA).

Case study: Ariane 5 Rocket Failure.

Learning Resources:

Text Books:

1. R. S. Pressman and B. R. Maxim, Software Engineering: A Practitioner's Approach, 9th ed. New York, NY, USA: McGraw-Hill Education, 2023.
2. P. Jalote, A Concise Introduction to Software Engineering: With Open Source and GenAI, 2nd ed. Cham, Switzerland: Springer Nature, 2025.
3. I. Sommerville, Software Engineering, 10th ed. Harlow, UK: Pearson Education, 2017.
Pankaj Jalote, An Integrated Approach to Software Engineering, Springer.

Reference Books:

1. H. Washizaki, ed., Guide to the Software Engineering Body of Knowledge (SWEBOK Guide), Version 4.0. IEEE Computer Society, 2024.
2. C. Ghezzi, M. Jazayeri, and D. Mandrioli, Fundamentals of Software Engineering, 2nd ed. Upper Saddle River, NJ, USA: Prentice Hall, 2003. (Classic reference)
3. A. S. Tamhane and S. A. Nirmal, Software Testing: Principles and Practices, 2nd ed. Oxford, UK: Oxford University Press, 2021.
4. F. Brooks, The Mythical Man-Month: Essays on Software Engineering, Anniversary ed. Boston, MA, USA: Addison-Wesley, 1995.

Web link for MOOC / NPTEL Links:

1. Software Engineering By Prof. Rajib Mall, IIT Kharagpur
https://onlinecourses.nptel.ac.in/noc25_cs108/preview
2. Software Engineering, Interactive Learning Portal
<https://www.nptelprep.in/courses/106105182>

Suggested Activity-based Class Learning:

1. Online Examination System.
2. Library Management System.
3. Hostel Management System.
4. E-Commerce Website.
5. Smart Attendance System.

Guidelines for Practicals:

1. The course instructor should design assignments by considering the required background knowledge, relevant technologies, practical applications and current trends related to the course.
2. Preferably, there should be multiple sets of assignments and distributed among batches of students.

3. Giving assignments based on real-life problems can help the students with future projects.
4. Students should be encouraged to use free and open-source software.
5. Instructors can also include an additional assignment that goes beyond the syllabus, but aligns with the course.
6. Instructors should include a course project to be implemented by a group of 2 or 3 students. It should be based on different topics studied in the course. Instructors should frame proper guidelines for the effective implementation of the course project.

Course Code: 102602	Course Name: Design and Analysis of Algorithms	
Teaching Scheme	Credit	Evaluation Scheme
Theory : 3 Hours/Week Practical : 2 Hours/Week	3 1	CCE : 40 Marks ESE : 60 Marks TW : 25 Marks PR + OR : 25 Marks

Expected Prerequisite Courses:

- Discrete Mathematics, Data Structures, Automata Theory.

Course Objectives:

- To enhance problem-solving skills through the application of mathematical theories.
- To evaluate and analyze the efficiency and performance of algorithms.
- To understand and study different algorithm design strategies.

Course Outcomes:

After successful completion of the course, learner will be able to:

CO1: Analyse time and space complexity of various algorithms.

CO2: Analyze asymptotic performance of algorithms.

CO3: Decide between Divide and Conquer and Greedy strategies.

CO4: Apply Dynamic programming to solve problems.

CO5: Choose suitable algorithmic strategy for problem solving.

Course Contents:**Unit-I: Algorithmic Analysis****08 Hours**

Algorithm: The Role of Algorithms in Computing, Asymptotic notations (Big Oh, small oh, Big Omega, Theta notations). Best case, average case, and worst-case time and space complexity of algorithms, classification of time complexities (linear, logarithmic etc)-Growth of function, Overview of searching, sorting algorithms, Recurrences: Formulation and solving recurrence equations using Master Theorem

Case Study: Finding products using an E-commerce website.

Unit-II: Complexity Theory**08 Hours**

Introduction: P, NP, NP Hard, NP Complete, polynomial and non-polynomial problems, deterministic and non-deterministic algorithms, P-class problems, NP-class of problems, Polynomial problem reduction, NP complete problems- vertex cover and 3-SAT and NP hard problem - Hamiltonian cycle

Case Study: A Sudoku puzzle solving , Camera placement for security purposes.

Unit-III: Divide and Conquer and Greedy Strategies**08 Hours**

Introduction to algorithmic strategies. **Divide and Conquer strategy:** Principle, control abstraction, time analysis of control abstraction, Binary search, Quick sort, Integer Arithmetic-Multiplication.

Greedy strategy: Principle, control abstraction, time analysis of control abstraction, knapsack problem, scheduling algorithms-Job scheduling and activity selection problem.

Case Study: Sorting employee salary records using Dynamic Programming, Scheduling of maximum events by event manager in a hall using greedy Strategy.

Unit-IV: Dynamic Programming**08 Hours**

Dynamic Programming: Principle, control abstraction, time analysis of control abstraction, computing Fibonacci series, binomial coefficients, OBST, 0/1 knapsack, All pair shortest path algorithm, sum of subset.

Case Study: Population growth generation by applying fibonacci series.

Unit-V: Algorithmic Strategy and Concepts**08 Hours**

Backtracking Strategy: Principle, control abstraction, 8-queen problem, graph coloring problem, knapsack problem.

Branch and Bound Strategy: Principle, control abstraction, strategies – FIFO, LIFO and LC approaches, knapsack problem (FIFO, LIFO, LC), TSP using Branch and Bound.

Concept and Use Case: Approximation algorithms, Randomization algorithms, Parallel algorithms, Distributed algorithms.

Case Study: Maximizing profit within a limited budget for a business using branch and bound, Weather forecast prediction using Parallel algorithms.

Learning Resources:**Text Books:**

1. Horowitz and Sahani, "Fundamentals of Computer Algorithms", University Press, ISBN: 978817371-6126, 81-7371-61262.
2. Parag Himanshu Dave, Himanshu Bhalchandra Dave, Design and Analysis of Algorithms, Pearson Education, ISBN 81-7758-595-9.

Reference Books:

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein, "Introduction to Algorithms", MIT Press; ISBN 978-0-262-03384-8.

2. Gilles Brassard, Paul Bratley, "Fundamentals of Algorithmics", PHI, ISBN 978-81-203-1131-2.
3. Sipser, Michael. "Introduction to the Theory of Computation." *ACM Sigact News* 27.1 (1996): 27-29.

Web link for MOOC / NPTEL Links:

1. Design & Analysis of Algorithms by Madhavan Mukund, Chennai Mathematical Institute
https://onlinecourses-archive.nptel.ac.in/noc18_cs20/preview

Guidelines for Practicals:

1. The course instructor should design assignments by considering the required background knowledge, relevant technologies, practical applications and current trends related to the course.
2. Preferably, there should be multiple sets of assignments and distributed among batches of students.
3. Giving assignments based on real-life problems can help the students with future projects.
4. Students should be encouraged to use free and open-source software.
5. Instructors can also include an additional assignment that goes beyond the syllabus, but aligns with the course.
6. Instructors should include a course project to be implemented by a group of 2 or 3 students. It should be based on different topics studied in the course. Instructors should frame proper guidelines for the effective implementation of the course project.

Course Code: 102603A	Course Name: Big Data Analytics	
Teaching Scheme	Credit	Evaluation Scheme
Theory : 3 Hours/Week Practical : 2 Hours/Week	3 1	CCE : 40 Marks ESE : 60 Marks TW : 25 Marks PR + OR : 25 Marks

Expected Prerequisite Courses:

- Database Management Systems, Data structures, Operating systems.

Course Objectives:

- To understand the fundamentals and challenges of Big Data.
- To learn distributed storage and processing frameworks.
- To implement data processing using Hadoop and MapReduce.
- To understand Big Data tools such as Hive, Pig, and Spark.
- To apply Big Data analytics techniques to large datasets.

Course Outcomes:

After successful completion of the course, learner will be able to:

CO1: Explain the concepts, characteristics, and applications of Big Data and Big Data Analytics.

CO2: Describe the Hadoop ecosystem and HDFS architecture used for the distributed storage of large datasets.

CO3: Apply the MapReduce programming model for processing and analyzing big data.

CO4: Use Big Data tools such as Hive, Pig, and NoSQL databases for managing and querying large datasets.

CO5: Implement Big Data analytics using Apache Spark for large-scale data processing.

Course Contents:

Unit-I: Introduction to Big Data 08 Hours

Evolution of Big Data, Definition and Characteristics of Big Data (5V's: Volume, Velocity, Variety, Veracity, Value), Traditional Data Processing vs Big Data Processing, Big Data Applications (Healthcare, Finance, Social Media, E-commerce), Big Data Analytics Lifecycle, Challenges in Big Data Management.

Case Study: Social Media Sentiment Analysis for Product Reviews.

Unit-II: Hadoop Ecosystem 08 Hours

Introduction to Hadoop, Hadoop Architecture, Hadoop Distributed File System (HDFS),

HDFS Components: NameNode, DataNode, HDFS Block Structure and Replication, Hadoop Cluster Architecture, Advantages and Limitations of Hadoop.

Case Study: Large-Scale Log Analysis Using Hadoop HDFS.

Unit-III: MapReduce Framework

08 Hours

Introduction to MapReduce, MapReduce Architecture, Working of Map and Reduce Functions, Data Flow in MapReduce, Input Splits and Record Readers, Shuffle and Sort Phase, Reduce Programs, Combiner and Partitioner, Applications of MapReduce.

Case Study: Word Count Analysis of Large Document Collections.

Unit-IV: Big Data Tools and Technologies

08 Hours

Introduction to Apache Hive, Hive Architecture and HiveQL, Introduction to Apache Pig, Pig Latin Programming, NoSQL Databases (Introduction), Types of NoSQL Databases: Key-Value, Column, Document, Graph, Introduction to HBase.

Case Study: E-Commerce Recommendation System Using NoSQL.

Unit-V: Emerging Technologies in Analytics

08 Hours

Introduction to Apache Spark, Spark Architecture, Resilient Distributed Datasets (RDD), Spark Components: Spark SQL, Spark Streaming, MLlib, Data Processing using Spark, Real-time Big Data Analytics, Case Studies: Big Data Applications.

Case Study: Real-Time Traffic Monitoring Using Apache Spark.

Learning Resources:

Text Books:

1. Seema Acharya, Subhasini Chellappan, "Big Data Analytics," Wiley, 2015.
2. Hadoop: The Definitive Guide – Tom White.

Reference Books:

1. Arvind Sathi, "Big Data Analytics: Disruptive Technologies for Changing the Game", MC Press, 2012.
2. Jay Liebowitz, "Big Data and Business Analytics," Auerbach Publications, CRC Press (2013).
3. Michael Mineli, Michele Chambers, Ambiga Dhiraj, "Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's Businesses", Wiley Publications, 2013.
4. Adam Shook and Donald Mine, "MapReduce Design Patterns: Building Effective Algorithms and Analytics for Hadoop and Other Systems" - O'Reilly 2012.
5. Thomas Erl, Wajid Khattak, and Paul Buhler, Big Data Fundamentals: Concepts, Drivers and Techniques, Pearson India Education Service Pvt. Ltd., 1st Edition, 2016.

Web link for MOOC / NPTEL Links:

1. Data Analytics with Python by Prof. A Ramesh, IIT Roorkee.
<https://nptel.ac.in/courses/106107220>
2. Data Analysis and Decision Making - I, Prof. Raghu Nandan Sengupta, IIT Kanpur.
<https://nptel.ac.in/courses/110104094>

Suggested Activity-Based Class Learning:

1. Flipped Classroom.
2. Gamification.
3. Online Interactive Tool.
4. Collaborative and Individual Problem-based learning.
5. Quizzes/Assignment.

Guidelines for Practicals:

1. The course instructor should design assignments by considering the required background knowledge, relevant technologies, practical applications and current trends related to the course.
2. Preferably, there should be multiple sets of assignments and distributed among batches of students.
3. Giving assignments based on real-life problems can help the students with future projects.
4. Students should be encouraged to use free and open-source software.
5. Instructors can also include an additional assignment that goes beyond the syllabus, but aligns with the course.
6. Instructors should include a course project to be implemented by a group of 2 or 3 students. It should be based on different topics studied in the course. Instructors should frame proper guidelines for the effective implementation of the course project.

Course Code: 102603B	Course Name: Cloud Computing	
Teaching Scheme	Credit	Evaluation Scheme
Theory : 3 Hours/Week Practical : 2 Hours/Week	3 1	CCE : 40 Marks ESE : 60 Marks TW : 25 Marks PR + OR : 25 Marks

Expected Prerequisite Courses:

- Operating Systems, Computer Networks.

Course Objectives:

- To provide students with the fundamentals and essentials of cloud computing
- To learn the basics of virtualization and its importance.
- To provide students a sound foundation of the cloud computing so that they are able to start using and adopting cloud computing services and tools in their real life scenarios
- To acquire knowledge on data storage and various Cloud Application Platform.
- To equip learners with the knowledge and skills to design, implement, and manage secure cloud environments.

Course Outcomes:

After successful completion of the course, learner will be able to:

CO1: Articulate the main concepts, key technologies and fundamentals of cloud computing.

CO2: Understand cloud enabling technologies and virtualization.

CO3: Analyse various cloud programming models and apply them to solve problems on the cloud.

CO4: Develop and deploy applications on Cloud.

CO5: Apply security-providing techniques for cloud applications.

Course Contents:**Unit-I: Introduction to Cloud Computing****08 Hours**

Origins and Influences, Goals and Benefits, Risks and Challenges, Roles and Boundaries, Cloud Characteristics, Migrating into the Cloud, Seven-step model of migration into a Cloud, Cloud Service/Delivery Models: IAAS,PAAS,SAAS, Cloud Deployment Models: Private ,Public, Hybrid, Community, Cloud System Architecture, Federated Cloud/ Intercloud,

Case Studies: Spotify Cloud Transformation, Netflix Cloud Migration.

Unit-II: Cloud Enabling Technology and Virtualization**08 Hours**

Cloud-Enabling Technology: Broadband Networks and Internet Architecture, Data Center

Technology Virtualization Technology, Web Technology, Multitenant Technology, Service Technology.

Virtualization: Definition of Virtualization, Implementation Levels of Virtualization, Types of Hypervisors, Virtualization of CPU, Memory, and I/O Devices, Virtual Clusters and Resource Management, Virtualization for Data-Center Automation

Case Studies: Cloud Infrastructure for an E-Commerce Platform.

Unit-III: Common Standards and Cloud Platforms

08 Hours

Common Standards: The Open Cloud Consortium, Open Virtualization Format, Standards for Application Developers: Browsers (Ajax), Data (XML, JSON), Solution Stacks (LAMP and LAPP), Syndication (Atom, Atom Publishing Protocol, and RSS), and Standards for Security.

Amazon web services: Compute services Storage Services Communication Services Additional services.

Google App Engine: Architecture 47 and core concepts, Application life cycle, Cost model

Microsoft Azure: Azure core concepts, SQL Azure, Windows Azure platform appliance.

Comparison of AWS, Azure and GCP.

Unit-IV: Data Storage and Cloud Computing Applications

08 Hours

Data Storage: Introduction to Enterprise Data Storage, Direct Attached Storage, Storage Area Network, Network Attached Storage,

Cloud file systems: GFS and HDFS, BigTable, HBase and Dynamo Cloud data stores: Datastore and Simple DB Gautam Shrauf, Cloud Storage-Overview, Cloud Storage Providers

Cloud Computing Applications: ECG Analysis in the Cloud, Protein Structure Prediction, Satellite Image Processing, CRM and ERP, Social Networking. Overview of OpenStack architecture.

Case Studies: Cloud-Based Healthcare and Scientific Data Processing System.

Unit-V: Security in Cloud

08 Hours

Risks in Cloud Computing: Types of Risks in Cloud Computing, Risk Management, Enterprise-Wide Risk Management,

Data Security in Cloud: Security Issues, Challenges, advantages, disadvantages, Cloud Digital persona and Data security, Content Level Security.

Cloud Security Services: Confidentiality, Integrity and Availability, Security Authorization Challenges in the Cloud, Secure Cloud Software Requirements, Secure Cloud Software Testing, Cloud Security Audit, Introducing Business Continuity and Disaster Recovery. Disaster Recovery- Understanding the Threats.

Case Studies: Capital One AWS breach analysis / GitHub Disaster Recovery Case Study.

Learning Resources:**Text Books:**

1. A. Srinivasan, J. Suresh, “Cloud Computing: A Practical Approach for Learning and Implementation”, Pearson, ISBN: 978-81-317-7651-3
2. Gautam Shrof “ENTERPRISE CLOUD COMPUTING Technology Architecture, Applications”, Cambridge University Press, ISBN: 9780511778476
3. Thomas Erl, Zaigham Mahmood and Ricardo Puttini, Cloud Computing: Concepts, Technology and Architecture, Pearson, ISBN :978 9332535923, 9332535922, 1 st Edition
4. Anthony T. Velte Toby J. Velte, Robert Elsenpeter, “Cloud Computing: A Practical Approach”, 2010, The McGraw-Hill.

Reference Books:

1. Rajkumar Buyya, Christian Vecchiola, S. ThamaraiSelvi, Mastering Cloud Computing: Foundations and Applications Programming, McGraw Hill, ISBN: 978 1259029950, 1259029956.
2. Jack J. Dongarra, Kai Hwang, Geoffrey C. Fox, Distributed and Cloud Computing: From Parallel Processing to the Internet of Things, Elsevier, ISBN :9789381269237, 9381269238, 1st Edition.
3. Brian J.S. Chee and Curtis Franklin, Jr., Cloud Computing: Technologies and Strategies of the Ubiquitous Data Center, CRC Press, ISBN :9781439806128.
4. Dr. Kumar Saurabh, “Cloud Computing, 4ed: Architecting Next-Gen Transformation Paradigms”, Wiley publication, ISBN: 9788126570966
5. Rishabh Sharma, “Cloud Computing: Fundamentals, Industry Approach and Trends”, Wiley publication.

Web link for MOOC / NPTEL Links:

1. Cloud Computing by Prof. Soumya Kanti Ghosh,, IIT Kharagpur
<https://nptel.ac.in/courses/106105167>
2. Cloud Infrastructure and Services by Prof. Dr. Nilesh K. Modi.
https://onlinecourses.swayam2.ac.in/e-learning/preview/nou25_cs33

Suggested Activity based Class Learning:

1. Cloud Cost Optimization Game.
 2. Deploy a Database on Cloud.
 3. Flipped Classroom.
 4. Poster presentation.
- Quizzes.

Guidelines for Practicals:

1. The course instructor should design assignments by considering the required background knowledge, relevant technologies, practical applications and current trends related to the course.
2. Preferably, there should be multiple sets of assignments and distributed among batches of students.
3. Giving assignments based on real-life problems can help the students with future projects.
4. Students should be encouraged to use free and open-source software.
5. Instructors can also include an additional assignment that goes beyond the syllabus, but aligns with the course.
6. Instructors should include a course project to be implemented by a group of 2 or 3 students. It should be based on different topics studied in the course. Instructors should frame proper guidelines for the effective implementation of the course project.

Course Code: 102603C	Course Name: Ethical Hacking	
Teaching Scheme	Credit	Evaluation Scheme
Theory : 3 Hours/Week Practical : 2 Hours/Week	3 1	CCE : 40 Marks ESE : 60 Marks TW : 25 Marks PR + OR : 25 Marks

Expected Prerequisite Courses:

- Introduction to Cyber Security, Computer Networks, Cryptography and Network Security.

Course Objectives:

- To introduce ethical hacking concepts and professional security practices.
- To develop practical skills in information gathering and security assessment techniques.
- To provide knowledge of web application, network, and wireless security threats and protections.
- To familiarize learners with modern security tools and ethical hacking methodologies.
- To introduce AI-based approaches and responsible practices in ethical hacking and threat detection.

Course Outcomes:

After successful completion of the course, learner will be able to:

CO1: Explain ethical hacking concepts, phases, and penetration testing methodologies.

CO2: Apply footprinting, scanning, and enumeration techniques using ethical hacking tools.

CO3: Analyze web application vulnerabilities and security practices.

CO4: Examine network and wireless security threats and protection techniques.

CO5: Evaluate AI-based techniques in ethical hacking and threat detection.

Course Contents:**Unit-I: Introduction to Ethical Hacking****08 Hours**

Ethical Hacking: Introduction, history, importance, Types of hackers: White hat, Black hat, Grey hat, Phases of ethical hacking, **Important Terminologies-** Asset, Vulnerability, Threat, Exploit, Risk, Vulnerability Assessments, **Penetration Test** - Introduction, Categories, Types, Report Writing, Introduction to Kali Linux.

Case Study: Physical Penetration Test.

Unit-II: Footprinting and Scanning**08 Hours**

Introduction to Information Gathering in Ethical Hacking, **Footprinting** – Introduction

Reconnaissance- Introduction, Types Information Gathering Methodology, Tools Used in the Reconnaissance Phase, **Port Scanning** - Introduction, Port Scanning using tools such as Nmap and Port Scanner, Ping Sweeps and Basic Scanning Scripts, Enumeration-Introduction.

Case Study: Enumerating Windows OS and Linux OS.

Unit-III: Web Application Security and Attacks

08 Hours

Introduction to Web Application Security, Common Web Application Vulnerabilities- Cross-Site Scripting (XSS), Cross-Site Request Forgery (CSRF), XML External Entity (XXE), Injections: SQL Injection and Code Injection, Denial of Service (DoS), Web Application Security Practices.

Case Study: WASP Top 10 Web Application Security Risks and tools.

Unit-IV: Network and Wireless Security

08 Hours

Network Security: Introduction, Threats and Attacks. **Wireless Network Security:** Introduction, Security Analysis of WPA/WPA2 Protocols, Threats to Wi-Fi Security, MAC Filtering and MAC Spoofing Concepts, Wireless Network Assessment Tools and Techniques, Evil Twin Attack.

Case Study: Network Sniffing and Social Engineering.

Unit-V: AI in Ethical Hacking

08 Hours

Introduction to AI in ethical hacking, AI for threat and anomaly detection, AI in penetration testing, AI for malware and phishing detection, comparison of traditional and AI-based threat detection techniques, evaluation of AI-driven security tools and techniques, challenges, and ethical considerations.

Learning Resources:

Text Books:

1. Ethical Hacking and Penetration Testing Guide by Rafay Baloch.
2. The Basics of Hacking and Penetration Testing by Patrick Engebretson.
3. Web Application Security-Exploitation and Countermeasures for Modern Web Applications, Andrew Hoffman, O'Reilly publication.
4. Cryptography and Network Security: Principles and Practice” — William Stallings.

Reference Books:

1. Network Security Bible (2nd Edition) Eric Cole, Ronald Krutz, James W. Conley — Wiley India Pvt. Ltd.
2. Penetration Testing: A Hands-on Introduction to Hacking by Georgia Weidman 2 Toolkit, Object- Oriented Systems Analysis and Design Using UML, 2nd Edition, Tata McGraw-Hill, 2002
3. Cryptography and Network security, Atul Kahate, The McGraw Hill, second Edition.

Web link for MOOC / NPTEL Links:

1. Ethical Hacking by Prof. Indranil Sengupta, IIT Kharagpur
https://onlinecourses.nptel.ac.in/e-learning/preview/noc23_cs75

Guidelines for Practicals:

1. The course instructor should design assignments by considering the required background knowledge, relevant technologies, practical applications and current trends related to the course.
2. Preferably, there should be multiple sets of assignments and distributed among batches of students.
3. Giving assignments based on real-life problems can help the students with future projects.
4. Students should be encouraged to use free and open-source software.
5. Instructors can also include an additional assignment that goes beyond the syllabus, but aligns with the course.
6. Instructors should include a course project to be implemented by a group of 2 or 3 students. It should be based on different topics studied in the course. Instructors should frame proper guidelines for the effective implementation of the course project.

Course Code: 102604A	Course Name: High Performance Computing	
Teaching Scheme	Credit	Evaluation Scheme
Theory : 3 Hours/Week Practical : 2 Hours/Week	3 1	CCE : 40 Marks ESE : 60 Marks TW : 25 Marks PR + OR : 25 Marks

Expected Prerequisite Courses:

- Digital Logic and Computer Organization, Computer Networks.

Course Objectives:

- To understand different parallel programming models
- To analyze the performance and modeling of parallel programs
- To illustrate the various techniques to parallelize the algorithm
- To implement parallel communication operations.
- To understand CUDA architecture.

Course Outcomes:

After successful completion of the course, learner will be able to:

CO1: Analyze various parallel paradigm.

CO2: Evaluate efficient parallel algorithm to solve given problem.

CO3: Illustrate the processes involved in data communication within various parallel computing architectures.

CO4: Apply CUDA programming architecture for parallel programs.

CO5: Evaluate various High Performance Computing applications.

Course Contents:**Unit I: Parallel Computing Introduction****08 Hours**

Introduction to Parallel Computing: Motivating Parallelism, Modern Processor: Stored program computer architecture, General-purpose Cache-based Microprocessor architecture. Parallel Programming Platforms: Implicit Parallelism, Dichotomy of Parallel Computing Platforms, Physical Organization of Parallel Platforms, Communication Costs in Parallel Machines. Levels of parallelism, Models: SIMD, MIMD, SIMT, SPMD, Data Flow Models, Architectures: N-wide superscalar architectures, multi-core, multi-threaded.

Case Study: Weather Forecasting using Parallel Computing.

Unit-II: Design of Parallel Algorithms**08 Hours**

Principles of Parallel Algorithm Design: Preliminaries, Decomposition Techniques,

Characteristics of Tasks and Interactions, Mapping Techniques for Load Balancing, Parallel Algorithm Models: Data, Task, Work Pool and Master Slave Model, Complexities: Sequential and Parallel Computational Complexity.

Case Study: Parallel Web Search Engine Indexing.

Unit-III: Communication in Parallel Environment

08 Hours

Basic Communication: One-to-All Broadcast, All-to-One Reduction, All-to-All Broadcast and Reduction, All-Reduce and Prefix-Sum Operations, Collective Communication using MPI: Scatter, Gather, Broadcast, Blocking and non-blocking MPI, All-to-All Personalized Communication, Circular Shift, OPENMP

Case Study: Parallel Weather Data Processing using MPI.

Unit-IV: CUDA Architecture

08 Hours

Sources of Overhead in Parallel Programs, Performance Measures and Analysis: Amdahl's and Gustafson's Laws, Speedup Factor and Efficiency, Cost and Utilization, Execution Rate and Redundancy, Introduction to GPU: Introduction to GPU Architecture overview, Introduction to CUDA C-CUDA programming model, write and launch a CUDA kernel, Handling Errors, CUDA memory model, Manage communication and synchronization, Parallel programming in CUDA-C.

Case Study: GPU-based Image Processing System.

Unit-V: Applications of High Performance Computing

08 Hours

Scope of Parallel Computing, Parallel Search Algorithms: Depth First Search(DFS), Breadth First Search(BFS), Parallel Sorting: Bubble and Merge, Distributed Computing: Document classification, Frameworks – Kuberbets, GPU Applications, Parallel Computing for AI/ ML Cluster Computing: Clustering Models, Clustering Architectures, Clustering Architectures key factors, types of clusters.

Case Study: Parallel AI-Based Document Classification System.

Learning Resources:

Text Books:

1. Ananth Grama, Anshul Gupta, George Karypis, and Vipin Kumar, "Introduction to Parallel Computing", 2nd edition, Addison-Wesley, 2003, ISBN: 0-201-64865-2
2. John Cheng, Max Grossman, and Ty McKercher, "Professional CUDA C Programming", John Wiley and Sons, Inc., ISBN: 978-1-118-73932-7.

Reference Books:

1. Georg Hager and Gerhard Wellein, Introduction to High Performance Computing for Scientists and Engineers, CRC Press

2. Kai Hwang, "Scalable Parallel Computing", McGraw-Hill, 1998.

Web link for MOOC / NPTEL Links:

1. <https://nptel.ac.in/courses/106108055>
2. <https://www.digimat.in/nptel/courses/video/106104120/L01.html>

Guidelines for Practicals:

1. The course instructor should design assignments by considering the required background knowledge, relevant technologies, practical applications and current trends related to the course.
2. Preferably, there should be multiple sets of assignments and distributed among batches of students.
3. Giving assignments based on real-life problems can help the students with future projects.
4. Students should be encouraged to use free and open-source software.
5. Instructors can also include an additional assignment that goes beyond the syllabus, but aligns with the course.
6. Instructors should include a course project to be implemented by a group of 2 or 3 students. It should be based on different topics studied in the course. Instructors should frame proper guidelines for the effective implementation of the course project.

Course Code: 102604B	Course Name: UI/UX Design	
Teaching Scheme	Credit	Evaluation Scheme
Theory : 3 Hours/Week Practical : 2 Hours/Week	3 1	CCE : 40 Marks ESE : 60 Marks TW : 25 Marks PR + OR : 25 Marks

Expected Prerequisite Courses:

- Problem Solving, Logical Thinking and Object-Oriented Technologies.

Course Objectives:

- To introduce fundamentals of User Interface (UI) and User Experience (UX) design.
- To teach students industry-standard design principles, workflows, and tools.
- To enable students to design interactive prototypes using modern tools.
- To develop skills in user research, usability testing, wireframing and visual design.
- To prepare students to apply UI/UX in real engineering and industrial applications.

Course Outcomes:

After successful completion of the course, learner will be able to:

CO1: Explain key UI/UX concepts, human-centered design principles, and design thinking.

CO2: Conduct user research, create personas, and identify user needs.

CO3: Create wireframes, user flows, and information architecture for applications.

CO4: Create visual design and apply heuristic evaluation and conduct usability testing.

CO5: Design industry-ready interfaces for web, mobile, and IoT systems.

Course Contents:

Unit I: Introduction to UI/UX

08 Hours

Disciplines involved in UI and UX, Why UI and UX study is important, Role of microcopy Importance of Human–Computer Interaction, Principles of UI/UX, User-centered Design, Measurable Human factors, Difference between UI and UX, Design Thinking process and Industrial relevance of UI/UX, Good vs. bad UX writing examples.

Case Study: ATM interface design.

Unit-II: User Research, Understanding the Human and Human Interaction **08 Hours**

User research methods: Interviews, surveys, observation, contextual inquiry, Empathy maps, Persona creation, and user journey maps. Task analysis and scenario mapping, Input-output channels, Qualitative and Quantitative Research Methods, Human memory, Human emotions, Individual differences, Psychology. Ergonomics, Human errors, Models of interaction,

Paradigms of Interactions, Interaction styles.

Case Study: Social Media Application Interface Design.

Unit-III: Interaction Design, Wireframing and Prototyping

08 Hours

Sitemap Creation, Hierarchy, navigation models, Interaction design principles, User flows and task flows, Information Architecture, Cognitive load, mental models, Accessibility standards (WCAG, ADA), Low-fidelity vs high-fidelity wireframes, Sketching techniques, Use of Figma / Adobe XD / Sketch/Miro/Wireframe/Canva, Designing interactive prototypes and Components, grids, constraints, auto layout, Handoff and Collaboration.

Case Study: Online Learning Management System (LMS).

Unit-IV: Visual Design, Usability and Testing

08 Hours

Typography, Color theory, Iconography, Layout systems, UI patterns and components, Design systems and style guides, Material Design and Human Interface Guidelines, Nielsen's Heuristics, Usability metrics (SUS Score, task completion, error rates), A/B testing, Heatmaps and Eye-tracking basics, Portfolio Building.

Case Study: Travel Booking Website.

Unit-V: Industry Applications of UI/UX

08 Hours

Automotive infotainment, Mobile apps, Web applications, IoT dashboards, AR/VR interfaces and Healthcare and fintech applications.

Case studies: Google, Apple, Amazon and Tesla.

Learning Resources:

Text Books:

1. About Face: The Essentials of Interaction Design About Face: The Essentials of Interaction Design, Alan Cooper; Robert Reimann; David Cronin; Christopher Noessel, ISBN-13: 978-1-118-76658-3, Publisher: John Wiley and Sons.
2. The Psychology of Everyday Things, Don Norman (Donald A. Norman), ISBN-13: 978-0-465-05065-9, Publisher: Basic Books.
3. Designing with the Mind in Mind: Simple Guide to Understanding User Interface Design Guidelines, Jeff Johnson, ISBN: 978-0-12-818202-4, Publisher: Morgan Kaufmann.

Reference Books:

1. The UX Book: Process and Guidelines for Ensuring a Quality User Experience, Rex Hartson and Pardha S. Pyla, ISBN: 978-0-12-385241-0, Publisher : Morgan Kaufmann.
2. Don't Make Me Think: A Common Sense Approach to Web Usability, Steve Krug, ISBN: 0-7897-2310-7, Publisher: New Riders Publishing.

3. Visualizing Data: Exploring and Explaining Data with the Processing Environment, Ben Fry, ISBN-13: 978-0-596-51455-6, Publisher: O'Reilly Media.
4. Sketching User Experiences: Getting the Design Right and the Right Design, Bill Buxton, ISBN-13: 978-0-12-374037-3, Publisher: Morgan Kaufmann.

Web link for MOOC / NPTEL Links:

1. Design and Implementation of Human-Computer Interfaces by Prof. Samit Bhattacharya, IIT Guwahati.
https://onlinecourses.nptel.ac.in/noc25_cs135/preview
2. User Interface Design by Prof. Saptarshi Kolay | IIT Roorkee.
https://onlinecourses.nptel.ac.in/noc21_ar05/preview
3. UI/UX Design with Figma : SWAYAM Plus
4. Introduction to Interaction Design by Prof. Sonal Atreya | IIT Roorkee.
https://onlinecourses.nptel.ac.in/noc24_ar23/preview

e-books:

1. UXPin Free E-Books.
2. UI, UX and Interaction Design Library (GitHub).
3. Free UX Tools and Strategy eBooks (Designflowww).
4. Free UX E-Books Worth Reading by WebFX.
5. Free UX Books for Designers (Lapa.Ninja).

Suggested Activity-Based Class Learning:

1. Paper Prototyping.
2. Card Sorting Activity.
3. Online Interactive Tool.
4. Quizzes/Assignment.

Guidelines for Practicals:

1. The course instructor should design assignments by considering the required background knowledge, relevant technologies, practical applications and current trends related to the course.
2. Preferably, there should be multiple sets of assignments and distributed among batches of students.
3. Giving assignments based on real-life problems can help the students with future projects.
4. Students should be encouraged to use free and open-source software.
5. Instructors can also include an additional assignment that goes beyond the syllabus, but aligns with the course.
6. Instructors should include a course project to be implemented by a group of 2 or 3 students. It should be based on different topics studied in the course. Instructors should frame proper guidelines for the effective implementation of the course project.

Course Code: 102604C	Course Name: Quantum Computing	
Teaching Scheme	Credit	Evaluation Scheme
Theory : 3 Hours/Week Practical : 2 Hours/Week	3 1	CCE : 40 Marks ESE : 60 Marks TW : 25 Marks PR + OR : 25 Marks

Expected Prerequisite Courses:

- Fundamentals of Computer & Programming, Operating Systems, Basic concepts of Networks.

Course Objectives:

- To introduce the fundamental concepts and motivation behind quantum computing.
- To develop the ability to use vector spaces, Hilbert spaces, operators, and tensor products in quantum mechanics.
- To enable students to evaluate the advantages and limitations of quantum algorithms in solving computational problems.
- To provide insight into quantum supervised and unsupervised learning techniques.
- To familiarize with Quantum AI approaches for decision-making and problem-solving in diverse domains.

Course Outcomes:

After successful completion of the course, learner will be able to:

CO1: Apply principles of superposition, entanglement, and quantum interference to represent and manipulate qubits.

CO2: Utilize the postulates of quantum mechanics to describe quantum measurements, expectation values, and time evolution of quantum systems.

CO3: Make use of fundamental quantum algorithms to address computational problems efficiently.

CO4: Analyze quantum learning techniques and quantum circuits for solving complex problems.

CO5: Examine Quantum AI techniques for decision-making and problem-solving in diverse domains.

Course Contents:

Unit I: Fundamentals of Quantum

08 Hours

Overview of Quantum Computing: Motivation and evolution of quantum computing.

Limitations of Classical Computing: Computational complexity, parallelism limits, and the need for quantum advantage. Quantum Mechanics for Computing: Superposition, entanglement, and quantum interference. Qubits vs Classical Bits: Representation of qubits and Bloch sphere visualization. Quantum Gates: Pauli gates (X, Y, Z), Hadamard gate, ControlledNOT gate (CNOT). Quantum Circuits and Measurement: Quantum circuit model and measurement in quantum systems.

Unit-II: Mathematical Foundations of Quantum Mechanics

08 Hours

Linear Algebra Framework for Quantum Mechanics: Vector spaces, complex Hilbert spaces, inner products, orthogonality, linear operators, eigenvalues and eigenvectors, Hermitian and unitary operators, tensor products. Postulates of Quantum Mechanics: Quantum states and state space, observables as operators, measurement postulate and Born rule, expectation values, time evolution of quantum states, quantum states in Hilbert space, qubits and Bloch sphere representation, quantum measurement theory, density operator formalism, composite quantum systems. Fundamentals of Quantum Principle: No-cloning theorem and implications for quantum information and quantum AI.

Unit-III: Quantum Algorithms

08 Hours

Overview of quantum algorithms and quantum speedup, Phil's algorithm, David Deutsch's algorithm, and David Deutsch–Jozsa algorithm, Daniel Simon's algorithm, Lov Grover's Search algorithm, Quantum Fourier Transform (QFT), Peter Shor's algorithm for integer factorization, introduction to quantum walk and search application.

Unit-IV: Quantum Machine Learning

08 Hours

Introduction to quantum-classical hybrid learning, data encoding and quantum feature maps, vibrational quantum circuits (VQC) and parametrized quantum circuits, quantum optimization and the parameter shift rule.

Quantum Supervised Learning: Quantum support vector machine (QSVM) and quantum neural networks.

Quantum Unsupervised Learning: Quantum K-Means and Quantum Principal Component Analysis (QPCA), quantum generative models, and the barren plateau problem.

Unit-V: Applications of Quantum Artificial Intelligence

08 Hours

Quantum AI for Optimization Problems: Quantum-enhanced combinatorial optimization, logistics and supply chain optimization, resource allocation and scheduling, smart grid, and traffic optimization.

Quantum AI in Finance and Business Analytics: Portfolio optimization, risk analysis and modeling, fraud detection, and financial forecasting.

Quantum AI in Healthcare and Life Sciences: Drug discovery optimization, molecular modeling, genomic data analysis.

Quantum AI in Cybersecurity: Quantum-enhanced anomaly detection, intrusion detection systems, AI-assisted post-quantum cryptanalysis.

Quantum AI in Intelligent Systems: Quantum machine learning applications, intelligent decision systems, and advanced AI architectures.

Learning Resources:

Text Books:

1. Wichert, Andreas, “Quantum Artificial Intelligence with Qiskit,” 1st Edition, Chapman & Hall / CRC Press, 2025, ISBN 978-1032448978.
2. Bernhardt, Chris, “Quantum Computing for Everyone,” 1st Edition, The MIT Press, 2019, ISBN 978-0262039253.

Reference Books:

1. Nielsen, M. A., Chuang, I. L., “Quantum Computation and Quantum Information”, 10th Edition, Cambridge University Press, 2010, ISBN 978-1107002173.
2. Wittek, Peter, “Quantum Machine Learning: What Quantum Computing Means to Data Mining”, 1st Edition, Elsevier Science, 2014, ISBN 978-0128010990.
3. Yanofsky, Noson S., Mannucci, Mirco A., “Quantum Computing for Computer Scientists”, 1st Edition, Cambridge University Press, ISBN 978-0-521-87996-5.

Web link for MOOC / NPTEL Links:

1. Quantum Computing by Prof. Debabrata Goswami, IIT Kanpur.
<https://nptel.ac.in/courses/104104082>
2. Introduction to Quantum Computing: Quantum Algorithms and Qiskit by Prof. Prabha Mandayam, Prof. Anupama Ray, Prof. Sheshashayee Raghunathan, IIT Madras & IBM Research. https://onlinecourses.nptel.ac.in/noc26_cs89/preview

Suggested Activity-based Class Learning:

1. Flipped Classrooms.
2. Virtual Lab.
3. Quizzes /Assignments.

Guidelines for Practicals:

1. The course instructor should design assignments by considering the required background knowledge, relevant technologies, practical applications and current trends related to the course.
2. Preferably, there should be multiple sets of assignments and distributed among batches of students.

3. Giving assignments based on real-life problems can help the students with future projects.
4. Students should be encouraged to use free and open-source software.
5. Instructors can also include an additional assignment that goes beyond the syllabus, but aligns with the course.
6. Instructors should include a course project to be implemented by a group of 2 or 3 students. It should be based on different topics studied in the course. Instructors should frame proper guidelines for the effective implementation of the course project.

Course Code: 170605A	Course Name: Automotive Technology	
Teaching Scheme	Credit	Evaluation Scheme
Theory : 3 Hours/Week Tutorial : 1 Hour/Week	3 1	CCE : 40 Marks ESE : 60 Marks TW : 50 Marks

Expected Prerequisite Courses:

- Engineering Physics, Engineering Mechanics.

Course Objectives:

- To provide fundamental understanding of construction and working of automobile subsystems.
- To equip students with tools and methods to measure and analyze performance of automobiles.
- To acquaint students with contemporary automotive technologies.

Course Outcomes:

After successful completion of the course, learner will be able to:

CO1: Compare the various chassis and drive-trains of an automobile.

CO2: Analyze axles, wheels, tyres and the steering system of an automobile.

CO3: Select suspension and braking systems of an automobile.

CO4: Measure and analyze the performance of an automobile and describe the safety aspects.

CO5: Discuss electrical systems and assess modern propulsion technologies of an automobile.

Course Contents:**Unit-I: Introduction to Chassis and Drive Train****08 Hours**

Introduction: Current scenario in Indian auto/ancillary industries, vehicle specifications and classification.

Chassis and Frames: Types of chassis layout with reference to power plant locations and drive, various types of frames and constructional details.

Drive Trains: Necessity and selection of clutch, necessity of gearbox and different types, overdrive, torque convertor, continuous variable transmission, dual clutch transmission, automated manual transmission, propeller shaft, final drive, and differential.

Unit-II: Axles, Wheels, Tyres and Steering System**08 Hours**

Axles: Purpose, requirement and types of front and rear axle, loads acting on rear axles.

Wheels and Tyres: Wheel construction, alloy wheel, wheel balancing and rotation, type of

tyres, tyre construction, tyre materials, factors affecting tyre life.

Steering System: Steering mechanism, steering geometry, cornering force, slip angle, scrub radius, steering characteristics, steering linkages and gearbox, power steering, collapsible steering, reversibility of steering, four-wheel steering, wheel alignment.

Unit-III: Suspension and Brake Systems

08 Hours

Suspension: Types of suspension linkages, types of suspension springs- leaf, coil, air springs, hydrogas, interconnected suspension, self-leveling suspension (active suspension), shock absorbers (hydraulic and air). **Brake Systems:** Drum, disc, mechanical, hydraulic, air brakes, power-assisted brakes, hand brake, ABS, EBD, BA.

Unit-IV: Vehicle Performance and Safety

08 Hours

Vehicle performance: parameters; vehicle resistances; traction and tractive effort; power requirements for propulsion; road performance curves (numerical examples); vehicle stability; vehicle testing on a chassis dynamometer; NVH in automobiles. **Vehicle safety:** Types of active and passive safety, vehicle interior and ergonomics.

Unit-V: Electrical Systems and Modern Propulsion Technologies

08 Hours

Batteries: Principles and construction of lead-acid battery, characteristics of battery, rating capacity and efficiency of batteries, various tests on battery condition, charging methods, introduction to lithium batteries.

Electrical System and Accessories: Basic electric layout of the vehicle, automotive sensors and actuators, electronic control unit/module.

Modern Propulsion Technologies: Concept and environmental importance of EVs and HEVs, Layout, construction and working, Challenges and future scope of EVs and HEV.

Learning Resources:

Text Books:

1. William H. Crouse., “Automotive Mechanics”, Tata McGraw Hill Publishing House.
2. Narang G. B. S, “Automobile Engineering”, S. Chand and Company Ltd.

Reference Books:

1. Kirpal Singh, “Automobile Engineering”, Volume 1, Standard Publishers distributors.
2. R. K. Rajput, “Automobile Engineering”, Laxmi Publications.

Web link for MOOC / NPTEL Links:

1. Fundamentals of Automotive Systems:
https://onlinecourses.nptel.ac.in/noc20_de06/preview
2. Autotronics: https://onlinecourses.nptel.ac.in/noc26_me101/preview

3. Vocational: Automotive Services Technician (Four-Wheeler):
https://onlinecourses.swayam2.ac.in/nos24_ge28/preview
4. Automotive Industrial Engineering:
<https://www.coursera.org/learn/automotive-industrial-engineering>
5. Automotive Engineering: Automobile Fundamentals:
<https://www.udemy.com/course/automotive-engineering-automobile-fundamentals-and-advanced/>

List of Practicals: (Any 6)

1. Identify four vehicles having different chassis layout and compare their performance.
2. Identify four vehicles equipped with different types of automatic transmission and compare their performance.
3. Visit an automobile service centre for demonstration of wheel alignment and write a detailed report on axle, wheels, tyres and wheel alignment of one car.
4. Experimental study of shock absorber.
5. Measurement of noise and vibration in an automobile.
6. Study of requirements of BNCAP test.
7. Demonstration on basic maintenance and checks of an automobile such as doors etting, consumable checking and replacing, brake pad checking & replacing, checking tyre wear etc.

Course Code: 170605B	Course Name: Database Systems	
Teaching Scheme	Credit	Evaluation Scheme
Theory : 3 Hours/Week Tutorial : 1 Hour/Week	3 1	CCE : 40 Marks ESE : 60 Marks TW : 50 Marks

Expected Prerequisite Courses:

- Fundamentals of Programming.

Course Objectives:

- To understand the fundamental concepts of database management.
- To identify and classify entities, attributes, and relationships in real-world problem domains.
- To understand the principles and structure of the relational data model.
- To demonstrate the use of SQL for secure and efficient database querying and management.
- To understand the basic concepts of transaction processing and concurrency control.

Course Outcomes:

After successful completion of the course, learner will be able to:

CO1: Explain the basic concepts, architecture, and components of database systems.

CO2: Convert ER diagrams into equivalent relational database schemas.

CO3: Normalize relational schemas using functional dependencies.

CO4: Explain the purpose and components of SQL in relational database systems.

CO5: Identify various transaction states and common transaction operations.

Course Contents:

Unit-I: Introduction to Database Systems

08 Hours

Need of database system, Types of Data, Database System Applications, Database System VS File System, View of Data, Data Abstraction, data independence, Data Models, Structure of Database Management Systems. **Case Study:** Study of Architecture of any Database Management System (DBMS) like Oracle or MySQL.

Unit-II: Entity Relationship Modeling

08 Hours

Conceptual data modeling: entities, entity types, attributes and its types, relationships, cardinalities on relationship, Entity Relationship diagram extended Features: Specialisation and Generalization on the ER Model. **Case Study:** ER Diagram conversion into Tables for

Unit-III: Relational Data Model**08 Hours**

Concept of relations, schema-instance distinction, keys and their types, Codd's Rules, Concept of Anomalies, Normalization, functional dependencies and Normal forms, 1NF, 2NF, 3NF and BCNF decompositions and desirable properties of them. **Case Study:** Library management system to identify different types of keys.

Unit-IV: Structured Query Language (SQL)**08 Hours**

Querying in SQL, DDL to create database and tables, table constraints, update database-update behaviors, DML, aggregation functions group by and having clauses, retrieve data from the database, generate and query views. Different types of Joins. **Case Study:** Design and implementation of Student Course Management System using SQL for effective management of students, courses, and faculty members.

Unit-V: Overview of Transaction Management**08 Hours**

The ACID Properties, Transactions and Schedules, Concurrency Control–The Need for Concurrency Control, Serializability, Locking Methods, Deadlock, Time Stamping Methods, Concurrency Control Protocols. Database Recovery: Need for Recovery, Recovery Techniques. **Case Study:** Design Online Shopping Cart Transaction Management In an e-commerce platform, multiple users simultaneously add, update, and purchase products.

Learning Resources:**Text Books:**

1. Silberschatz A., Korth H., Sudarshan S., "Database System Concepts", McGraw Hill Publishers, ISBN 0-07-120413-X, 6th edition
2. S.K.Singh, "Database Systems: Concepts, Design and Application", Pearson Education, ISBN 978-81-317-6092-5
3. Ramez Elmasri, Shamkant B. Navathe, "Fundamentals of Database Systems", Sixth Edition, ISBN-13: 978-0-136-08620-8.

Reference Books:

1. Shio Kumar Singh, Database Systems Concepts Design and Applications, Pearson
2. Mario Piattini, Oscar Diaz - Advanced Database Technology and Design- online book.

Web link for MOOC / NPTEL Links:

1. Introduction to Database Systems By Prof. Sreenivasa Kumar, IIT Madras.

https://onlinecourses.nptel.ac.in/noc20_cs03/preview

List of Practicals:

1. Study of installation of various database software like MySQL, MariaDB & Oracle, etc.
2. Draw an ER diagram for any database system with various entities and their attributes on ERWin/ERDPLUS Software.
3. Implement SQL queries to provide students with hands-on experience for studying various commands of DDL.
4. Implement SQL queries to provide students with hands-on experience for studying various commands of DML.
5. Implement SQL queries to provide students with hands-on experience for studying various Logical Operators of SQL.
6. Implement SQL queries to provide students with hands-on experience in implementing various data constraints using SQL commands in MySQL/MariaDB.
7. Write at least 10 SQL queries for suitable database applications using SQL DML statements by implementing all types of joins.
8. Write at least 10 SQL queries for suitable database applications using SQL DML statements by implementing sub-queries and views.
9. Write at least 10 SQL queries for suitable database applications using SQL DML statements for studying various aggregation functions.
10. ER Modeling and Normalization: a. Consider case studies and formulate.
 - a. Problem statement for any application to be developed.
 - b. Propose a conceptual design using ER.
 - c. Convert ER diagram into relational tables and normalize relational data model.
11. Mini- Project.

Course Code: 170605C	Course Name: Industrial Control Components	
Teaching Scheme	Credit	Evaluation Scheme
Theory : 3 Hours/Week Tutorial : 1 Hour/Week	3 1	CCE : 40 Marks ESE : 60 Marks TW : 50 Marks

Expected Prerequisite Courses:

- Fundamentals Electronics Engineering.

Course Objectives:

- To understand the basics of process control, including variables, system types, and real-world applications of control loops.
- to learn about transmitters, signal standardization, communication protocols, and converters used for accurate measurement in process control.
- To explore control actions and tuning methods to optimize controller performance and prevent issues like reset windup.
- To understand control valve design, terminology, types, and sizing using Cv calculations in line with industry standards.
- To learn about valve accessories and actuators, focusing on their design and function for optimal control system performance.

Course Outcomes:

After successful completion of the course, learner will be able to:

- CO1:** Analyze process variables, control system types, and process characteristics, and evaluate examples of process control loops in industry.
- CO2:** Apply knowledge of transmitters, signal standards, protocols, and converters to evaluate their role in process control systems.
- CO3:** Design and apply control actions and tuning methods to optimize controller performance and system stability.
- CO4:** Evaluate control valve selection, sizing, and design based on key characteristics and industry standards.
- CO5:** Analyze and evaluate valve accessories and actuators for optimal performance in process control systems.

Course Contents:**Unit-I: Introduction to Process Control****08 Hours**

Key Concepts in Process Control: Process Variables: Definition and differentiation of manipulated, disturbance, and output variables.

Control System Types: Understanding Open-Loop vs. Closed-Loop systems.

Block Diagram Representation: Structure and components of control loops. Examples of Process Control Loops: Application examples of control loops in industries—temperature, flow, level, and pressure.

Unit-II: Transmitters and Converters**08 Hours**

Transmitters and Signal Standardization: Importance of transmitters in process control systems. Standardization of signals: current (4-20 mA), voltage (0-10 V), and pressure signal standards. **Types of Transmitters:** Live zero vs. dead zero, and concept of two-wire. SMART Transmitters. **Differential Pressure Transmitters (DPT):** Types, installation, and calibration of DPTs. Application in level and flow measurement. Zero elevation, suppression.

Converters: Understanding the difference between converters and transmitters. Types: Current-to-pressure and pressure-to-current converters.

Unit-III: Control Actions and Tuning Techniques**08 Hours**

Discontinuous Control: On-Off, multi-position, floating control modes. **Continuous Control:** Proportional (P), Integral (I), Derivative (D), PI, PD, and PID control modes.

Reset Windup: Techniques to prevent reset windup, rate before reset, and bumpless transfer.

Controller Tuning Methods: Introduction to process reaction curve. Tuning techniques: Ziegler-Nichols.

Unit-IV: Control Valve: Design, Sizing, and Selection**08 Hours**

Control Valve Fundamentals: Key terminology: Rangeability, turndown ratio, cavitation, flashing, noise, viscosity index, valve capacity, AO, AC, and fail-safe actions. **Valve Characteristics:** Linear, equal percentage, and quick-opening characteristics.

Valve Types and Applications: Overview of different valve types: Globe, Ball, Butterfly, Diaphragm, and specialty valves for high-temperature and high-pressure applications.

Valve Sizing and Selection: Cv Calculation.

Unit-V: Control Valve Accessories and Actuator Design**08 Hours**

Valve Accessories: Detailed discussion on accessories such as positioners (pneumatic, electro-pneumatic, and digital), I/P converters, volume boosters, air lock, limit switches, solenoid valves, and hand wheels.

Actuator Types and Design: Understanding different actuator types: Spring and diaphragm, piston cylinder, and smart actuator.

Learning Resources:

Text Books:

1. C. D. Johnson, Process Control and Instrument Technology, Tata McGraw Hill, Publications, 8th Edition.
2. N. A. Anderson, Instrumentation for Process Measurement and Control, CRC Press.

Reference Books:

1. G. Liptak, “Process Control”, Instrument Engineering Handbook CRC Press,

Web link for MOOC / NPTEL Links:

1. Industrial Automation and Control, By Prof. Siddhartha Mukhopadhyay, IIT Kharagpur.
<https://nptel.ac.in/courses/108105062>
2. Process Control and Instrumentation, by Dr. P.K. Saha, IIT Guwahati.
<https://nptel.ac.in/courses/103103037>

List of Practicals: (Any 8)

1. Identification of industrial control components.
2. Block diagram of an industrial automation system.
3. Comparison of open-loop and closed-loop systems.
4. Direction control valve applications.
5. Control valve sizing basics.
6. Actuator selection.
7. Valve characteristic comparison.
8. PID control exercises.
9. Tuning of PID controller for level/flow control loop.
10. Controller tuning problems.
11. Process reaction curve analysis.
12. Study of the control valve & plot the installed characteristics of the control valve.

Course Code: 170605D	Course Name: Embedded System in IoT	
Teaching Scheme	Credit	Evaluation Scheme
Theory : 3 Hours/Week Tutorial : 1 Hour/Week	3 1	CCE : 40 Marks ESE : 60 Marks TW : 50 Marks

Expected Prerequisite Courses:

- Basic Electronics Fundamentals, Sensors and Actuators, Basic C Programming.

Course Objectives:

- To understand the fundamentals of embedded systems, IoT architecture, and the role of hardware–software co-design in connected devices.
- To develop the ability to interface sensors, actuators, and communication modules with microcontrollers for IoT applications.
- To analyze and implement embedded programming concepts using platforms such as Arduino, Raspberry Pi, or ARM-based controllers.
- To design and develop IoT-based systems using networking protocols (e.g., MQTT, HTTP) and cloud integration techniques.
- To evaluate security, power management, and real-time constraints in embedded IoT systems for efficient and reliable performance.

Course Outcomes:

After successful completion of the course, learner will be able to:

- CO1:** Explain fundamentals of embedded systems, including components, classifications, design metrics, and the role of microcontrollers.
- CO2:** Apply IoT hardware design concepts using microcontroller platforms, interfaces, and communication modules with effective power management.
- CO3:** Apply IoT communication models, protocols, data transmission, and basic cloud integration concepts.
- CO4:** Analyze Embedded C programming, real-time processing, RTOS concepts, and low-power design techniques in IoT-based embedded systems.
- CO5:** Evaluate end-to-end IoT system design, security mechanisms, and real-world IoT applications across domains.

Course Contents:**Unit-I: Introduction to Embedded Systems****08 Hours**

Definition, characteristics, and requirements of embedded systems, Embedded system components: hardware, software, processors, peripherals, Classification of embedded systems, Embedded system design metrics: cost, power, performance, memory, reliability, Applications of embedded systems in various domains, Overview of microprocessors vs microcontrollers, Role of 8-bit microcontrollers in embedded development.

Unit-II: IoT Hardware Platforms and Interfacing**08 Hours**

Design Considerations for IoT Hardware, Microcontroller Platforms: Arduino, ESP32, NodeMCU, Nucleo (STM32 Overview), ADC, DAC, GPIO Programming, Communication Interfaces: UART, SPI, I2C, Wireless Communication Modules (Wi-Fi, BLE, GSM, LoRa – Basics), Power Management in IoT Devices.

Unit-III: IoT Communication Protocols and Networking**08 Hours**

IoT Communication Models, IP Addressing Basics, Application Layer Protocols: MQTT, HTTP, CoAP (Overview), Client-Server vs Publish-Subscribe Model, Data Acquisition and Transmission, Introduction to Cloud Platforms (ThingsBoard / AWS IoT – Conceptual), REST API Basics.

Unit-IV: Embedded Software and Real-Time Concepts for IoT**08 Hours**

Embedded C for IoT Applications, Interrupt Handling and Timer-Based Programming, Real-Time Data Monitoring, Introduction to RTOS for IoT Devices, Task Scheduling Basics, Data Logging and Local Storage, Edge Computing Concept, Low Power IoT Design Techniques.

Unit-V: IoT System Design, Security and Applications**08 Hours**

End-to-End IoT System Design Methodology, IoT Security: Authentication, Encryption Basics, Secure Communication (TLS Concept), Data Analytics Basics for IoT, Industrial IoT (IIoT) Overview, Smart City Applications, Smart Energy Monitoring System, IoT-Based Environmental Monitoring, Smart Irrigation System, Remote Health Monitoring System.

Learning Resources:**Text Books:**

1. Muhammad Ali Mazidi, Janice Gillispie Mazidi & Rolin D. McKinlay, “The 8051 Microcontroller and Embedded Systems: Using Assembly and C”, Pearson Education.
2. Zdravko Karakehayov, “Embedded Systems Design with 8051 Microcontrollers: Hardware and Software”, CRC Press / Taylor & Francis.
3. Raj Kamal, “Embedded Systems: Architecture, Programming & Design”, Tata McGraw-Hill.

4. David E. Simon, “An Embedded Software Primer”, Pearson Education.

Reference Books:

1. Daniel D. Gajski, Samar Abdi, “Embedded System Design: Modeling, Synthesis and Verification”, Springer, 2021.
2. Peter Marwedel, “Embedded System Design: Foundations of Cyber-Physical Systems and the Internet of Things”, Springer
3. Qing Li & Caroline Yao, “Real-Time Concepts for Embedded Systems”, CRC Press (Taylor & Francis)
4. Marilyn Wolf, “Computers as Components: Principles of Embedded Computing System Design”, Elsevier
5. Catherine H. Gebotys, “Security in Embedded Devices”, Springer.

Web link for MOOC / NPTEL Links:

1. https://onlinecourses.nptel.ac.in/noc24_ee68/preview
2. https://onlinecourses-archive.nptel.ac.in/noc19_cs32/course
3. <https://www.coursera.org/learn/iot>

List of Practicals: (Any 8)

1. Draw and explain 3-layer and 5-layer IoT architecture. Map real-world application to architecture layers.
2. Comparison of IoT Development Boards. Compare Arduino, ESP32, Raspberry Pi based on: Processing speed, Memory, Connectivity, Power consumption, Application suitability.
3. Sensor & Actuator Selection. Select appropriate sensors for:
 - a. Smart irrigation system
 - b. Health monitoring system
 - c. Smart energy meterJustify selection with technical reasoning.
4. Protocol Comparison. Compare MQTT, HTTP, CoAP. Identify best protocol for given IoT scenario.
5. IoT Communication Model Analysis. Device-to-Device. Device-to-Cloud. Device-to-Gateway. Draw block diagrams and explain.
6. Embedded C Programming Problems. Write logic for:
 - a. Sensor threshold detection
 - b. PWM-based motor control
 - c. Interrupt-based event detection.
7. IoT Security Case Study. Identify security threats in smart home system. Suggest countermeasures.

8. Design complete IoT solution for Smart parking.
9. Design complete IoT solution for Weather monitoring.
10. Industrial IIoT Monitoring Plan. Design a predictive maintenance node with: Vibration analysis, Data transmission, Dashboard planning.
11. Smart Campus IoT Design. Complete architecture including: Sensors, Networking, Cloud, Security, Cost estimation.
12. Sensor Signal Conditioning Design. Design amplifier gain for low-voltage sensor. Select filter type for noise removal.

Course Code: 170605E	Course Name: Governing Policies	
Teaching Scheme	Credit	Evaluation Scheme
Theory : 3 Hours/Week Tutorial : 1 Hour/Week	3 1	CCE : 40 Marks ESE : 60 Marks TW : 50 Marks

Expected Prerequisite Courses:

- Business Economics.

Course Objectives:

- To explain the principles of infrastructure systems, governance frameworks, and policy implementation in urban and rural contexts.
- To examine economic evaluation tools, including demand forecasting, cost–benefit analysis, financing options, and PPP models.
- To interpret national infrastructure programs, policies, and regulations for effective development planning.
- To apply sustainable, inclusive, and technology-driven approaches to plan and manage urban and rural infrastructure.

Course Outcomes:

After successful completion of the course, learner will be able to:

- CO1:** Understand infrastructure, governance, and policy implementation in urban and rural development.
- CO2:** Analyze economic aspects of infrastructure, apply demand forecasting, cost–benefit analysis, financing mechanisms, and PPP models.
- CO3:** Apply national infrastructure programs, policies, and regulations to plan and assess urban and rural development projects.
- CO4:** Implement planning and management tools for urban and rural infrastructure projects.
- CO5:** Apply concepts of sustainable and inclusive infrastructure by integrating climate resilience, risk reduction, resource efficiency, digital governance, and global case insights.

Course Contents:

Unit-I: Infrastructure Governance

08 Hours

Concept, scope, and components of urban & rural infrastructure, Infrastructure lifecycle and asset management, Principles of good governance, Role of central, state, and local governments, Urban Local Bodies (ULBs), Panchayati Raj Institutions (PRIs), National

policies on housing, transport, water, sanitation, energy, rural development, Policy formulation, implementation, monitoring.

Unit-II: Economics of Urban and Rural Infrastructure**08 Hours**

Economic characteristics of infrastructure (public goods, externalities, natural monopoly), Demand forecasting & cost–benefit analysis (CBA), Budgetary allocations, grants, and intergovernmental transfers, Municipal finance and rural development financing, PPP models, Case studies, Life-cycle costing, Economic impact assessment, social cost–benefit analysis.

Unit-III: Policies, Acts and Regulatory Frameworks**08 Hours**

AMRUT, Smart Cities Mission, Swachh Bharat Mission, National Urban Housing Policy, PMAY, National Transport Policies, PMGSY, NRLM, MGNREGA, Rural drinking water, sanitation, electrification programs, Environmental regulations (EIA, Water/Air Acts, Waste Management Rules), Land acquisition and R&R policies, Infrastructure concession frameworks.

Unit-IV: Planning, Implementation and Monitoring of Infra. Projects**08 Hours**

Master planning, regional planning, Integrated urban/rural planning approaches, GIS-based infrastructure planning, DPR preparation, feasibility studies, Procurement methods, contract management, Risk management, KPIs for urban and rural services, Service level benchmarks (SLBs), Digital governance & monitoring systems.

Unit-V: Sustainable, Inclusive and Resilient Infrastructure Governance**08 Hours**

Green and climate-resilient infrastructure, Disaster risk reduction, Energy efficiency, circular economy, Infrastructure for marginalized groups, Rural–urban linkages for balanced development, E-governance tools IoT, AI, and data analytics, UN SDGs and infrastructure governance, international case studies.

Learning Resources:**Text Books:**

1. Town Planning, G. K. Hiraskar, Dhanpat Rai Publications
2. Town Planning, S. C. Rangwala, Charotar Publishing House Pvt. Ltd.
3. Rural Development: Planning Strategy & Policy Imperatives — Vivek Saurath & Manoj Agarwal.

Reference Books:

1. Urban Planning by Chris Couch.
2. Urban Infrastructure and Governance by G. Ramesh, Vishnu Prasad Nagadevara, Gopal Naik, Anil Suraj.

3. Urban and Regional Planning Education: Learning for India (edited by Ashok Kumar, Diwakar S. Meshram, Krishne Gowda).
4. Swachh Bharat Mission Guidelines.
5. MRTP Act 1966: The director, government printing, stationary and publications, Maharashtra state, Mumbai
6. URDPFI & AMRUT Guidelines: Ministry of housing and urban affairs, Government of India.
7. Natural Infrastructure and Resilient Cities: Role of Green Infrastructure in Urban Planning for Adaptation to Climate Change in India (2025, by National Institute of Disaster Management, Delhi).
8. Role of Edge Analytics on Sustainable Smart City Development: Challenges and Solutions by G. R. Kanagachidambaresan.
9. Sustainable Smart Cities in India: Challenges and Future Perspectives (The Urban Book Series) by Poonam Sharma and Swati Rajput.
10. Rural Development: Principles, Policies and Development by Katar Singh and Anil Shishodia. Atlantic Publications.

Web link for MOOC / NPTEL Links:

1. https://onlinecourses.swayam2.ac.in/imb21_mg02/preview
2. https://onlinecourses.swayam2.ac.in/ugc23_ge04/preview
3. https://onlinecourses.nptel.ac.in/noc23_ce52/preview
4. https://onlinecourses.swayam2.ac.in/cec20_ar01/preview
5. <https://www.coursera.org/learn/smart-cities>
6. https://onlinecourses.nptel.ac.in/noc25_ar27/preview
7. https://onlinecourses.nptel.ac.in/noc26_ar08/preview
8. https://onlinecourses.nptel.ac.in/noc25_ar25/preview
9. https://onlinecourses.nptel.ac.in/noc26_ce48/preview

List of Practicals (Any 8):

1. Presentation on any national policy for housing and transport. (Group Activity)
2. Analyze the latest central/state government budget allocations for infrastructure sectors. (Individual Activity)
3. Comparison of Public-Private Partnership (PPP) models in infrastructure project (Individual Activity)
4. Conduct a sanitation or waste management study under Swachh Bharat guidelines with case study. (Individual Activity)
5. Prepare a compliance checklist for industries under Water Act, Air Act, and Waste Management Rules. (Group Activity)

6. Prepare a land use map and zoning proposal for a neighborhood. (Group work)
7. Conduct a basic energy audit of a building (college, hostel, home) and propose energy-saving measures. (Group Activity)
8. Suggest a small-scale circular economy project for rural or urban ward. (Group Activity)
9. Map how infrastructure projects (roads, water supply, housing, digital systems) contribute to SDGs 6, 7, 9, 11, and 13. (Individual Activity)
10. Identify and explain components of a tender document by referring to any online tender document related to infrastructure project.

Course Code: 170605F	Course Name: Application Development	
Teaching Scheme	Credit	Evaluation Scheme
Theory : 3 Hours/Week Tutorial : 1 Hour/Week	3 1	CCE : 40 Marks ESE : 60 Marks TW : 50 Marks

Expected Prerequisite Courses:

- Web Development, Database Management Systems.

Course Objectives:

- To familiarize students with client-side scripting concepts.
- To provide knowledge of server-side development and backend processing in web applications.
- To enable students to understand basic database concepts.
- To understand and learn Web application deployment.

Course Outcomes:

After successful completion of the course, learner will be able to:

CO1: Apply the client-side technologies for web development.

CO2: Develop user interfaces using component-based design approaches.

CO3: Apply the server-side technologies for web development.

CO4: Create effective web applications for business functionalities using the latest web development platforms.

CO5: Deploy web applications in collaborative environments.

Course Contents:

Unit-I: Basics of Web Application Interactivity	08 Hours
--	-----------------

Introduction to JavaScript, JS Variables and Constants, JS Variable Scopes, JS Data Types, JS Functions, JS Array, JS Object, JS Events.

Unit-II: ReactJS	08 Hours
-------------------------	-----------------

Introduction to ReactJS, ReactJS Components Styling, Routing, Redux- Architecture, Hooks- Basic hooks, useState() hook, useEffect() hook, useContext() hook.

Unit-III: Backend Development Using NodeJS	08 Hours
---	-----------------

Role of backend in web applications, Introduction to NodeJS, Environment Setup, NodeJS Events, NodeJS Functions, NodeJS Built-in Modules, File System, Handling Data I/O in NodeJS, Create HTTP Server, Create Socket Server.

Unit-IV: Database Integration Using MongoDB**08 Hours**

Introduction to databases, Need for databases in web applications, Types of databases: SQL vs NoSQL, MongoDB fundamentals, Data models and schemas, Basic CRUD operations.

Unit-V: Managing and Deploying Web Applications: GitHub**08 Hours**

Deployment strategies, Overview of Git, Git Repositories: Public vs private repositories, Basic Git Workflow & commands, collaboration using GitHub.

Learning Resources:**Text Books:**

1. Jeffrey C.Jackson, "Web Technologies: A Computer Science Perspective", Second Edition, Pearson Education, 2007, ISBN 978-0131856035.
2. Kogent Learning Solutions Inc, Web Technologies: HTML, JAVASCRIPT, PHP, JAVA, JSP, XML and AJAX, Blackbook, Dreamtech Press, ISBN: 9788177228496.

Reference Books:

1. Shama Hoque, Full-Stack React Projects: Learn MERN Stack Development by Building Modern Web Apps, Packt Publishing.
2. Dr. Hiren Joshi, Web Technology and Application Development, DreamTech, 1st Ed.

Web link for MOOC / NPTEL Links:

1. <https://www.coursera.org/learn/programming-with-javascript>
2. <https://www.coursera.org/specializations/build-dynamic-website>

List of Practicals:

1. Design responsive user registration and login which will do the following actions: Register user, login user functionality.
2. Create a To-Do List application where users can add tasks, mark tasks as completed, and delete tasks.
3. Develop a frontend using React JS which includes forms and navigation.
4. Create a Node.JS Application which serves a static website.
5. Develop a Node.js application with MongoDB to perform CRUD Operations.
6. Create version control account on GitHub and implement basic git commands and repository handling.
7. Mini Project: Develop a web application using front end and backend technologies in any of the domains: Social Media, E-commerce, Finance, Education, Any other.

Course Code: 170605G	Course Name: Data Visualization	
Teaching Scheme	Credit	Evaluation Scheme
Theory : 3 Hours/Week Tutorial : 1 Hour/Week	3 1	CCE : 40 Marks ESE : 60 Marks TW : 50 Marks

Expected Prerequisite Courses:

- Programming and Problem Solving.

Course Objectives:

- To introduce the fundamental concepts and importance of data visualization.
- To develop the ability to create basic graphical representations such as bar charts, line charts, etc.
- To enable students to explore and summarize datasets using filtering, aggregation, and basic statistical measures.
- To apply visualization techniques to analyze time-series and geospatial data commonly used in engineering applications.
- To develop skills for creating dashboards and communicating insights through visual storytelling and business analytics tools.

Course Outcomes:

After successful completion of the course, learner will be able to:

- CO1:** Explain the importance, principles, and techniques of data visualization and perform basic data exploration using filtering and aggregation methods.
- CO2:** Apply appropriate visualization techniques to create basic and advanced charts.
- CO3:** Analyze time-series datasets to identify trends, seasonality, and noise using suitable visualization techniques.
- CO4:** Analyze geospatial data using mapping techniques to derive meaningful insights.
- CO5:** Develop interactive dashboards and visual reports using visualization tools to communicate data insights effectively.

Course Contents:

Unit-I: Overview of Data Visualization

08 Hours

Importance and benefits of good data visualization– Design principles. Statistical Measures, Introduction to python libraries for visualization, Exploring Data – Reduce Items and Attributes: Filter and Aggregate.

Unit-II: Creation of Basic Visualization**08 Hours**

Histogram, Bar (Vertical and Horizontal) and Line Chart, Box plot, Scatter plot (Examples and Exercises to be given for practice). Color palettes – Creation of 3D Charts. Creation of Advanced Visualization: Heat Map– Facet Grid – Interaction Techniques: Manipulate View – Creation of interactive Network topologies and Trees.

Unit-III: Visualization of Time Series Data**08 Hours**

Summary statistics and plotting aggregated views - Visualization of seasonality, trends and noise– working with multiple time series data such as Power load variation, Machine temperature monitoring, Rainfall and water level data, Signal variation in electronics.

Unit-IV: Visualization of Geospatial Data**08 Hours**

Spatial join - overlapping geospatial data to maps and adding special cues – Case Study- Visualization of multimodal data and analysis-case study sensor data and health care, genome and biomedical data.

Unit-V: Business Analytics and Visualization Tools**08 Hours**

Tableau, PowerBI, Creating Interactive Dashboards and charts to organize data using visualization principles and Data Storytelling.

Learning Resources:**Text Books:**

1. Python Data Visualization: An Easy Introduction to Data Visualization in Python, ISBN: 9780070087705, Tata McGraw-Hill Education, 2009.
2. Tamara Munzner, Visualization Analysis and Design, A K Peters Visualization Series, CRC Press, 2014.

Reference Books:

1. Hands-On Data Visualization: Interactive Storytelling from Spreadsheets to Code Jack Dougherty and Ilya Ilyankou 1st Edition, O'Reilly Media, 2021.
2. Alberto Cairo, The Functional Art: An Introduction to Information Graphics and Visualization, New Riders, 2012.

Web link for MOOC / NPTEL Links:

1. <https://www.coursera.org/learn/advanced-data-visualization-with-tableau-public>

List of Practicals:

1. Load a dataset and perform basic data exploration by identifying attributes, data types, and displaying summary statistics.

2. Using a dataset, compute statistical measures (mean, median, standard deviation) and perform filtering and aggregation
3. Create Histogram, Bar Chart (vertical and horizontal), and Line Chart using a suitable dataset.
4. Using a dataset, create Box Plot and Scatter Plot to analyze data distribution and relationships between variables.
5. Using a dataset, generate advanced visualizations such as Heat Map and Facet Grid with appropriate color palettes.
6. Using a time-series dataset, create visualizations to identify trends, seasonality, and patterns.
7. Using a geospatial dataset, plot location-based data on a map and analyze spatial distribution.
8. Create an interactive dashboard using Tableau or Power BI and present insights using data storytelling.

Course Code: 170605H	Course Name: Drives and Controls	
Teaching Scheme	Credit	Evaluation Scheme
Theory : 3 Hours/Week Tutorial : 1 Hour/Week	3 1	CCE : 40 Marks ESE : 60 Marks TW : 50 Marks

Expected Prerequisite Courses:

- Power Electronics, Electric Machines.

Course Objectives:

- To understand the fundamentals, components, and control principles of electric drive systems.
- To analyze thermal models and select appropriate motor power ratings for different duty cycles.
- To study performance, speed control, and braking methods of DC motor drives.
- To understand operation, control techniques, and speed control methods of induction motor drives
- To understand the types, operation, and control of synchronous and special motor drives for fixed and variable speed applications.

Course Outcomes:

After successful completion of the course, learner will be able to:

- CO1:** Explain the fundamentals, components, selection criteria, and control techniques of electric drives, including closed-loop and PLL control.
- CO2:** Analyze the thermal characteristics of motors to determine suitable motor power ratings under different duty cycles and load conditions.
- CO3:** Analyze performance, speed control, multi-quadrant operation, and closed-loop control of DC motor drives.
- CO4:** Explain starting, speed control methods, slip power recovery techniques, and closed-loop control of induction motor drives.
- CO5:** Analyze the operation and control of synchronous, Permanent Magnet Synchronous Motor, BLDC, stepper, and switched reluctance motor drives.

Course Contents:**Unit-I: Basics of Electric Drives and Control****08 Hours**

Definition, Advantages of electrical drives, Components of Electric drive system, Selection Factors, status of Electrical Drives (DC & AC), speed control and drive classifications, close

loop control of drives, phase locked loop (PLL) control.

Unit-II: Selection of Motor Power Rating**08 Hours**

Thermal model of motor for heating and cooling-Relationship between temperature rise and time, Heating time constant, Cooling time constant. classes of motor duty- continuous duty, short time duty, intermittent periodic duty, intermittent periodic duty with starting, intermittent periodic duty with starting & braking, continuous duty with intermittent periodic loading, continuous duty with starting & braking, Continuous duty with periodic speed changes. Determination of motor ratings, Selection of power ratings of motor -continuous duty and constant load, continuous duty and variable load, Short time rating of motor.

Unit-III: DC Motor Drives**08 Hours**

DC motors and their performance starting, transient analysis, speed control, Controlled rectifier fed drives, full controlled 3 phase rectifier control of dc separately excited motor, Multi quadrant operation, Chopper controlled drives closed loop speed control of DC motor.

Unit-IV: Induction Motor Drives**08 Hours**

Induction motor analysis, starting and speed control methods- voltage and frequency control, current control, closed loop control of induction motor drives, rotor resistance control, Slip power recovery: Static Kramer and Scherbius Drive, Single phase induction motor starting, braking and speed control.

Unit-V: Special Purpose Motor Drives**08 Hours**

Synchronous motor types, operation with fixed frequency, variable speed drives, Permanent Magnet Synchronous Motor and BLDC motor drives, Stepper motor drives, switch reluctance motor drives.

Learning Resources:**Text Books:**

1. G. K. Dubey, Fundamentals of Electrical Drives, 2nd Edition (sixth reprint), Narosa Publishing House, 2001.
2. M. L. Soni, P.V. Gupta, U. S. Bhatnagar, “A course in Electrical Power”, 1999, Dhanpat Rai & Sons.

Reference Books:

1. M. H. Rashid, Power Electronics -Circuits, devices and Applications, 3rd Edition, PHI Pub. 2004.
2. B. K. Bose, Modern Power Electronics and AC Drives, Pearson Education, Asia, 2003.

Web link for MOOC / NPTEL Links:

1. <http://nptel.ac.in/courses/108104140>

List of Practicals:

1. Based on Basics of Electric Drives and Control.
2. Based on Selection of Motor Power Rating.
3. Based on DC Motor Drives.
4. Based on Induction Motor Drives.
5. Based on Special Purpose Motor Drives.
6. Mini project/case study on Electrical drives.
7. Seminar on Electrical drives.
8. Industrial Visit to electrical drive and control industry and prepare report.

Curriculum Design and Development	
Chairman - Board of Studies and Advisors	
Dr. Balasaheb S. Tarle	Dr. Umakant Kulkarni
Dr. Subhash Shinde	Dr. Pramod D. Patil
Mr. Abhay Malpure	Mr. Pravin Thakur
Mr. Sandip Shinde	Dr. Rajeev Papneja

Course Design Team Members	
Dr. S. R. Devane	Dr. S. A. Talekar
Dr. V. S. pawar	Mrs. V. V. Ahire
Mr. S. T. Datir	Dr. A. K. Patil
Ms. S. A. Dhumne	Mr. P. D. Aher
Ms. M. C. Deore	Ms. P. P. Boraste
Mr. Y. S. Bhavsar	Mr. R. S. Thakare
Ms. M. V. Malode	Dr. S. H. Jadhav
Ms. A. P. Shinde	Mr. R. P. Aher
Ms. A. V. Tidke	Dr. V. S. Tidake
Ms. R. S. More	Ms. R. R. Shevale
Dr. V. R. Sonawane	Ms. P. V. Shinde
Ms. S. V. Pawar	Ms. D. L. Tidke
Mr. B. S. Gholap	Ms. M. B. Thombare
Ms. P. S. Kubal	Ms. P. P. Pingale
Ms. A. B. Uphade	