Padre Conceição College Of Engineering

SECOND YEAR COMPUTER SCIENCE AND ENGINEERING (ARTIFICIAL INTELLIGENECE AND MACHINE LEARNING) SCHEME AND SYLLABUS (RC2025-26)

SEMESTER – III							
Sr. No.	**Course Category**	**Course Code**	**Title of the Course**	**L**	**T**	**P**	**TCr**
1	**Major**	**AIM-200**	Data Structures using C++	3	0	0	3
		AIM-201	Data Structures using C++ Lab	0	0	1	1
		AIM-202	Foundation of Artificial Intelligence	3	0	0	3
		AIM-203	Foundation of Artificial Intelligence Lab	0	0	1	1
2	**Minor/IC/ PE**	**AIM-221**	Digital Logic Design	3	0	0	3
		AIM-222	Digital Logic Design Lab	0	0	1	1
		OR					
		AIM-223	Embedded System Design and Applications	3	0	0	3
		AIM-224	Embedded System Design and Applications Lab	0	0	1	1
3	**Multi-disciplinary**	**SHM-XXX**	Mathematical Foundations for AI and ML-I	3	0	0	3
4	**AEC**	**AEC-251**	Indian Languages				
		HIN-253	Sambhashan Kala	0	0	2	2
		OR					
		MAR-253	Sambhashan Kaushalya	0	0	2	2
		OR					
		KON-253	Sambhashan Kaushalya	0	0	2	2
5	**SEC**	**SEC-XXX**	AI Powered App Development	0	0	3	3
			TOTAL	**12**	**0**	**8**	**20**

 # Syllabus Provisional (Subject to Goa University Approval)

SEMESTER - IV							
Sr. No.	**Course Category**	**Course Code**	**Title of the Course**	**L**	**T**	**P**	**TCr**
1	**Major**	**AIM-204**	Modern Database Management Systems	2	0	0	2
		AIM-205	Modern Database Management Systems Lab	0	0	2	2
		AIM-206	Foundations of Machine Learning	3	0	0	3
		AIM-207	Foundations of Machine Learning Lab	0	0	1	1
		AIM-208	Modern Operating Systems	3	0	0	3
		AIM-209	Modern Operating Systems Lab	0	0	1	1
		ECM-XXX	Mathematical Foundations for AI and ML-II	3	1	0	4
2	**Professional Electives**	**AIM-225**	Design and Analysis of Algorithms	3	0	0	3
		AIM-226	Design and Analysis of Algorithms Lab	0	0	1	1
		OR					
		AIM-227	Fundamentals of Software Engineering	3	0	0	3
		AIM-228	Fundamentals of Software Engineering Lab	0	0	1	1
			TOTAL	**14**	**1**	**5**	**20**

SEMESTER III

Name of the Programme : B.E in Computer Science and Engineering (Artificial Intelligence and Machine Learning)
Course Code : AIM-200
Title of the Course : Data Structures using C++
Number of Credits : 3
Effective from AY : 2025-26

Pre-requisites for the Course:	Knowledge of C programming	
Course Objectives:	This course enables students to: 1. Understand fundamental data structures and basics concepts of C++. 2. Apply various basic data structures such as stacks, queues and linked lists to solve complex programming problems. 3. Analyze graph and different graph algorithms to solve real world problems like finding shortest paths on huge maps. 4. Design various sorting, searching, and hashing algorithms.	
Content:		**No of Hours**
Unit-1	**Basic concepts of C++:** Introduction and Characteristics of Object Oriented Programming **Classes and Objects:** Constructors, Destructors **Inheritance :**Single Inheritance, Multiple inheritance **Polymorphism:** Virtual Functions, Pure Virtual Functions **Introduction To Data Representation & Data Structures:** Linear and Non-Linear Data Structures, Static and Dynamic Data Structures. **Stacks:** representation of Stacks, Application of Stack, Recursion.	**12**
Unit - 2	**Queues:** Representation of Queues, circular queues, dequeues and priority queues. **Lists:** Singly linked list, doubly linked list, circular linked list, linked stacks and queues and its applications. **Trees:** Basic terminology, binary trees and their representations, traversals of trees, applications of trees – infix/postfix representation of expressions and inter-conversion, B-tree, AVL.	**11**
Unit - 3	**Sorting:** Basic concept, Exchange sort, Selection sort, Insertion sort, Quick sort, Tree sort, Merge sort, Radix sort, Heaps and Heap sort. **Searching:** Basic searching techniques, sequential and binary	**11**

	search, tree searching. **Hashing:** Hash function, collision handling mechanisms.	
Unit - 4	**Graphs:** Basic terminology, representation of graphs, directed and undirected graphs and their traversals, depth first and breadth first search, spanning trees. **Applications of Graphs:** Shortest path problem, topological sorting, matching.	**11**
Pedagogy	Interactive, reflective, and inquiry-based methods, with a strong emphasis on critical thinking and problem-solving skills.	
References/ Readings	**Text books** 1. Yashawant Kanetkar; Data Structures through C++, 5th Edition, BPB Publication, 2023, ISBN-10: 938928623X. 2. Yashawant Kanetkar; Let us C++, 3rd Edition, BPB Publications, 2019, ISBN: 9789388176644 3. S. K. Srivastava; Comprehensive Data Structures and Algorithms in C++, BPB publications, 2025, ISBN-10: 9365898579. **Reference books** 1. NB Venkateswarlu, Data Structures: Theory and Practicals, AICTE, 2022, ISBN - 978-81-959863-1-6. 2. Yedidyah Langson, Moshej Augenstein, Aaron M. Tenenbaum, Data Structures using C & C++, 2nd Edition, Prentice Hall of India, 2000, ISBN-10: 8120315737 3. Ellis Hurwitz and Sartaj Sahni; Fundamentals of Data Structures, 1st Edition, Golgotha Publications, 1984, ISBN-13. 978-0273020721. 4. Sahni; Data Structures, Algorithms and Applications in C++, 2nd Edition, MGH, 2005, ISBN 9788173715228. 5. Varsha H. Patil, Data Structures Using C++, Oxford,2012, ISBN-10: 0198066236	
Course	Upon completion of the course, students will be able to: **CO1:** Explain the basics of C++, data structures and their representations. **CO2:** Describe the different operations performed on linked list and trees.	

Outcomes:	**CO3:** Analyze the efficiency of sorting, searching and hashing algorithms. **CO4:** Apply appropriate graph data structures to solve defined computing problems.	

Name of the Programme : **B.E in Computer Science and Engineering (Artificial Intelligence and Machine Learning)**
Course Code : **AIM-201**
Title of the Course : **Data Structures using C++ Lab**
Number of Credits : **1**
Effective from AY : **2025-26**

Pre-requisites for the Course:	Knowledge of C programming	
Course Objectives:	This course will enable students to: 1. Understand the implementation of linear and non-linear data structures. 2. Apply appropriate algorithms for searching, sorting, and traversal operations. 3. Analyze the efficiency of different data structure operations. 4. Implement dynamic memory techniques using linked data structures.	
Contents:	**List of Programs**	**No of Hours**
	1. To implement classes, objects, constructors, destructors. 2. To implement polymorphism and inheritance. 3. Implementation of stack using arrays. 4. Implementation of queue using arrays. 5. Implementation of Circular queues using arrays. 6. Implementation of queue using linked lists. 7. Implementation of Singly Linked lists and Doubly Linked lists. 8. Implementation of binary search tree & its operations & traversals. 9. Implementation of sorting techniques: Insertion sort and Heap sort. 10. Implementation of sorting techniques: Merge sort and Quick sort. 11. Implementation of search techniques: Linear search and Binary search. 12. Implementation of graph traversal techniques.	**30**
Pedagogy:	Integration of instructional learning, constructive thinking, inquiry-based, collaborative, experiential, and problem-solving approaches.	

References/ Readings:	**Text books** 1. Yashawant Kanetkar; Data Structures Through C++, 5th Edition, BPB Publication, 2023, ISBN-10: 938928623X. 2. Yedidyah Langson, Moshej Augenstein, Aaron M. Tenenbaum, Data Structures using C & C++, 2nd Edition, Prentice Hall of India, 2000, ISBN- 10: 8120315737. 3. S. K. Srivastava; Comprehensive Data Structures and Algorithms in C++, BPB publications, 2025, ISBN-10: 9365898579. **Reference books** 1. NB Venkateswarlu, Data Structures: Theory and Practicals, AICTE, 2022, ISBN - 978-81-959863-1-6. 2. Ellis Hurwitz and Sartaj Sahni; Fundamentals of Data Structures, 1st Edition, Golgotha Publications, 1984, ISBN-13. 978-0273020721. 3. Sahni; Data Structures, Algorithms and Applications in C++, 2nd Edition,MGH, 2000, ISBN-10: 007049539X. 4. Varsha H. Patil, Data Structures Using C++, Oxford,2012, ISBN-10: 0198066236 5. Sachi Nandan Mohanty, Data Structure and Algorithms Using C++, Scrivener Publishing,2021, ISBN 978-1-119-75054-3

Course Outcomes:	Upon completion of the course, students will be able to: **CO1:** Explain linear and non-linear data structures. **CO2:** Analyze sorting and searching algorithm efficiency. **CO3:** Apply suitable data structures to solve problems. **CO4:** Implement programs using trees and graphs.

Name of the Programme : B.E in Computer Science and Engineering (Artificial Intelligence and Machine Learning)
Course Code : AIM-202
Title of the Course : Foundations of Artificial Intelligence
Number of Credits : 3
Effective from AY : 2025-26

Pre-requisites of the course:	Basic knowledge of Logic and Computer Programming	
Course Objectives:	**This course will enable the students to:** • Understand the fundamental concepts and applications of Artificial Intelligence and intelligent agents. • Learn different problem-solving methods and search strategies. • Apply knowledge representation techniques and reasoning mechanisms. • Gain foundational knowledge of AI paradigms and the real-world applications.	
Contents:		**No of Hours**
Unit - 1	**Introduction to Artificial Intelligence**: What is AI?, History and types of AI, Relationship between AI, ML, Language Processing and Deep Learning, Introduction to Machine Learning and Deep Learning, Applications of AI- Robotics, Drones, Future of AI, Introduction to No code and Low Code AI. **Artificial Intelligence Technologies**: Techniques in AI, Introduction to Machine Learning Model, Regression Analysis in Machine Learning, Classification Techniques, Clustering Techniques, Naive Bayes Classification, Neural Network, SVM.	13
Unit - 2	**Artificially Intelligent Machine**: Defining Intelligence, Components of Intelligence, Difference between human and Machine Intelligence, Agents and Environment, Search, Uniformed Search Algorithms, Informed Search Algorithms, Hill Climbing Algorithm in AI, Adversarial Search, Alpha-Beta Pruning, Single Agent Path Finding Problem. **Knowledge Representation**: Introduction, Knowledge Representation, Knowledge Based Agent, Types of Knowledge, Techniques of Knowledge Representation in AI, Syntax of Propositional Logic, Logical Connectives in Propositional Logic, Inference Rules, Forward Chaining and Backward Chaining in AI.	11
Unit - 3	**Reasoning and Learning in AI**: Reasoning, Probabilistic Reasoning in AI, Bayes Theorem, Learning, Clustering, Explanation Based Learning, Reinforcement Learning. **Computer Vision**: Human Vision vs. Computer Vision, Tasks in Computer Vision, Applications and Challenges in Computer Vision, Understanding Image Pixels, Convolutional Neural Network, Immersive Experience.	11

Unit - 4	**Natural Language Processing (NLP)**: Introduction, Chatbot, Components of NLP, Steps in NLP, Phases of NLP, Syntax vs. Semantic Analysis, Applications, Pros and Cons of NLP. **Current Trends in AI:** AI and Ethical Concerns, AI as a Service, Robotics, Recent Trends in AI. **Where AI is heading today?** : Characteristics and Components of Expert Systems, Advantages and Limitation of Expert Systems, Applications of Expert Systems, AIoT, Introduction to Edge Computing.	10
Pedagogy:	Interactive, reflective, and inquiry-based methods, with a strong emphasis on critical thinking and problem-solving skills.	
References	**Text Books:** 1. Reema Thareja, Artificial intelligence: beyond classical AI, Pearson education, 2023, ISBN: 978-9356069329. 2. "Artificial Intelligence", Ela Kumar, I.K. International Publishing House Pvt. Ltd. 2018, ISBN: 978-81-906566-6-5364440820. **Reference Books:** 1. Stuart Russell & Peter Norvig , "Artificial Intelligence: A Modern Approach", 4th Edition, Pearson ,2022, ISBN:978-1-292-40117-1 2. Pradeep Singh, Tapan K Gandhi, Balasubramanian Raman ,Artificial intelligence and machine learning , McGraw hill , 2025,ISBN: 978-9364440820. 3. Elian Greystone, Mastering AI with Python: From Machine Learning Basics to Building Real World AI Applications,3rd edition , 2025,ISBN 979-829214233 4. Elaine Rich, Kevin Knight, Shivashankar B Nair, Artificial intelligence, 4th Edition, MedTech Science Press, 2024, ISBN-10 819654457X. 5. Dan Fitzpatrick, Amanda Fox & Brad Mathew Weinstein, The AI Classroom: The Ultimate Guide to Artificial Intelligence in Education, teacher goals publication, 2023, ISBN 978-1-959419-11-2.	
Course Outcomes	**Upon completion of the course, students will be able to:**	

	CO1: Explain the fundamental concepts and applications of AI, intelligent agents, and agent architectures **CO2:** Apply AI problem-solving techniques such and game-playing strategies. **CO3:** Analyze knowledge representation schemes and reasoning methods for constructing intelligent systems. **CO4:** Implement simple machine learning techniques to evaluate AI applications with ethical and social factors.	

Name of the Programme : B.E in Computer Science and Engineering (Artificial Intelligence and Machine Learning)
Course Code : AIM-203
Title of the Course : Foundations of Artificial Intelligence Lab
Number of Credits 1
Effective From AY : 2025-26

Pre-requisites for the Course:	Basic knowledge of Logic and Computer programming	
Course Objectives:	**The course will enable students to:** • Provide hands-on experience in implementing intelligent agents and search algorithms. • Build competency in game-playing techniques and reasoning methods. • Develop knowledge representation models, machine learning models, and rule-based inference solutions. • Apply automated planning and related tools to solve real-world AI problems.	
Contents:	**List of Programs/Experiments**	**No of Hours**
Contents	1. Setting up python environment for building AI applications. 2. Implement Linear Regression. 3. Implement Classification Technique. 4. Implement Clustering Technique. 5. Implement the Best-First Search algorithm. 6. Implement the Depth-First Search algorithm. 7. Implement the A* search algorithm. 8. Implement Hill Climbing algorithm. 9. Implement Min-Max algorithm. 10. Implement Alpha-Beta Pruning. 11. Case study on Reasoning and Learning. 12. Case study in expert systems.	**30**
Pedagogy:	Integration of constructive thinking, Collaborative, experiential and problem-solving approaches.	

References/ Readings:	**Text books** 1. Reema Thareja, Artificial intelligence: beyond classical AI, Pearson education, 2023, ISBN: 978-9356069329 2. Gaurav Leekha, Learn AI with Python, 1st Edition, Bpb Publications, 2025, ISBN-9789391392611. **Reference Books** 1. Elaine Rich, Kevin Knight, Shivashankar B Nair, Artificial intelligence, 4th edition, MedTech Science Press, 2024, ISBN-10 819654457X. 2. Dan Fitzpatrick, Amanda Fox & Brad Mathew Weinstein, The AI Classroom: The Ultimate Guide to Artificial Intelligence in Education, teacher goals publication, 2023, ISBN 978-1-959419-11-2. 3. Ivan Bratko, Prolog Programming for Artificial Intelligence, 4th edition, Pearson, 2023, ISBN 978-0-321-41746-6. 4. Elian Greystone, Mastering AI with Python: From Machine Learning Basics to Building Real World AI Applications, 3rd Edition, 2025, ISBN 979-829214233. 5. Pradeep Singh, Tapan K Gandhi, Balasubramanian Raman, Artificial intelligence and machine learning, McGraw hill, 2025, ISBN: 978-9364440820.
Course Outcomes:	**Upon completion of the course, students will be able to:** **CO1:** Explain and construct intelligent AI problem Representations. **CO2:** Apply search and game-playing techniques to solve AI problems. **CO3:** Develop knowledge-based systems using logic programming and reasoning. **CO4:** Implement simple machine-learning algorithms and assess their performance

Name of the Programme : B.E in Computer Science and Engineering (Artificial Intelligence and Machine Learning)
Course Code : AIM-221
Title of the Course : Digital Logic Design
Number of Credits 03
Effective from AY : 2025-26

Pre-requisites for the Course:	Basics of Electrical and Electronics	
Course Objectives:	**This course will enable students to:** 1. Understand the principles of number systems, logic gates, and binary coding methodologies. 2. Explain a comprehensive understanding of Boolean algebra and advanced logic simplification techniques. 3. Apply established theoretical frameworks to analyse and resolve logic-related problems, including number-system conversions. 4. Design foundational combinational and sequential digital circuits in accordance with standard design practices.	
Contents:		**No of Hours**
UNIT- 1	**Binary Systems:** Digital Systems, Binary Numbers, Number-Base Conversions, Octal and Hexadecimal Numbers, Complements of Numbers, Signed Binary Numbers, Binary Codes, Binary Storage and Registers, Binary Logic. **Boolean Algebra and Logic Gates:** Introduction, Basic definitions, Axiomatic definition of Boolean Algebra, Basic theorems and properties of Boolean Algebra, Boolean Functions, Canonical and Standard forms, Other Logic Operations, Digital Logic Gates, Integrated Circuits.	**10**
UNIT-2	**Gate-Level Minimization:** Introduction, The Map Method, Four-Variable K-Map, Product of Sums Simplifications, Don't Care Conditions, NAND and NOR Implementation, Other Two-Level Implementations, Exclusive-OR Function. **Combinational Logic:** Introduction, Combinational Circuits, Analysis of Combinational Circuits, Design Procedure, Binary Adder-Subtractor, Decimal Adder, Binary Multiplier, Magnitude Comparators, Decoders, Encoders, Multiplexers.	**12**
UNIT -3	**Flip-flops:** Latch and Flip-Flops-SR, D, JK, and T Flip-flop. Master Slave Flip-flops, Flip-flop Excitation Tables, Applications of Flip-flops. **Shift Registers:** Buffer Register, Controlled Buffer Register, Serial In Serial Out (SISO) Shift Register, Serial in Parallel Out (SIPO) Shift Register, Parallel in Serial Out (PISO) Shift Register and Parallel in Parallel Out (PIPO) Shift Register, Applications of Shift Registers.	**11**

UNIT -4	**Counters:** Asynchronous Counters- Design of Asynchronous Counters, Effects of Propagation Delay in Ripple Counters, Synchronous Counters- Design of synchronous counters, Design of Synchronous 3-bit up counter and Synchronous 3-bit down counter. Shift Register Counters **Sequential Circuits-I:** The Finite State Model, Memory Elements, Synthesis of Synchronous Sequential Circuits. **Sequential Circuits-II:** Finite State Machine, Capabilities and Limitations of Finite State Machines.	**12**
Pedagogy:	Interactive, reflective, and inquiry-based methods, with a strong emphasis on critical thinking and problem-solving skills	
References/ Readings:	**Text Books** 1. M.Morris Mano, Michael D. Ciletti: Digital Design: With an Introduction to the Verilog HDL, VHDL, and System Verilog, 6e, Pearson Education, 2018, ISBN-13: 978-9353062019. 2. Anand Kumar: Fundamentals of Digital Circuits, Fourth Edition, PHI Learning Pvt. Ltd., 2018, ISBN-13: 978-81-203-5268-1. **Reference Books** 1. Thomas L. Floyd: Digital Fundamentals, Eleventh edition, Pearson India, 2018, ISBN-13: 978-93-325-8460-0. 2. Dr. Menka Yadav: Digital Electronics, AICTE, 2022, ISBN-13: 978-81-959863-0-9 3. B. Somanathan Nair: Digital Electronics and Logic Design, Prentice-Hall of India Pvt. Ltd., 2004, ISBN-13:81-203-1956-7. 4. Raj Kamal: Digital Systems-Principles and Design, Pearson, 2007, ISBN-13:978-81-775-8570-4. 5. Anil K. Maini: Digital Electronics – Principles and Integrated Circuits, Wiley, 2019, ISBN-13: 978-8126508631	
Course Outcomes:	Upon completion of the course, students will be able to: **CO1:** Describe the basic concepts of number systems, logic gates, and binary codes. **CO2:** Explain Boolean algebra principles and apply logic simplification techniques. **CO3:** Solve logic problems and perform number system conversions using standard methods. **CO4:** Design simple combinational and sequential digital circuits	

Name of the Programme : B.E in Computer Science and Engineering (Artificial Intelligence and Machine Learning)
Course Code : AIM-222
Title of the Course : Digital Logic Design Lab
Number of Credits :1
Effective from AY : 2025-26

Pre-requisites for the Course:	Basics of Electrical and Electronics	
Course Objectives:	**This course will enable students to:** 1. Identify basic logic gates and their truth tables. 2. Explain Boolean expressions and logic function realization. 3. Apply universal gates to implement logic functions. **4.** Design and verify combinational and sequential circuits.	
Contents:	**List of Experiments**	**No of Hours**
	1. Verification and interpretation of truth table for AND, OR, NOT, NAND, NOR, Ex-OR, Ex-NOR gates. 2. Realization of logic functions with the help of universal gates NAND and NOR Gate. 3. Realization of Boolean expressions in SOP & POS forms. 4. Design of Adders and Subtractors. 5. Design and Implementation of Multiplexers and De-multiplexers. 6. Design and Implement Comparators. 7. Verify the truth table of JK and D flip-flops. 8. Design and Implementation of Master-Slave Flip-flop. 9. Design SISO/SIPO Shift register. 10. Design of 3-bit Asynchronous Counter. 11. Design of 3-bit Synchronous Up Counter. 12. Design of Ring Counter.	**30**
Pedagogy:	Interactive, reflective, and inquiry-based methods, with a strong emphasis on critical thinking and problem-solving skills.	
References/ Readings:	**Text Books** 1. M.Morris Mano, Michael D. Ciletti: Digital Design: With an Introduction to the Verilog HDL, VHDL, and System Verilog, 6e , Pearson Education, 2018, ISBN-13: 978-9353062019. 2. Anand Kumar: Fundamentals of Digital Circuits, Fourth Edition, PHI Learning Pvt. Ltd., 2018, ISBN-13: 978-81-203-5268-1.	

	Reference Books 1. Thomas L. Floyd: Digital Fundamentals, Eleventh edition, Pearson India, 2018, ISBN-13: 978-93-325-8460-0. 2. Dr. Menka Yadav: Digital Electronics, AICTE, 2022, ISBN-13: 978-81-959863-0-9 3. B. Somanathan Nair: Digital Electronics and Logic Design, Prentice-Hall of India Pvt. Ltd., 2004, ISBN-13:81-203-1956-7. 4. Raj Kamal: Digital Systems-Principles and Design, Pearson, 2007, ISBN-13:978-81-775-8570-4. **5.** Anil K. Maini: Digital Electronics – Principles and Integrated Circuits, Wiley, 2019, ISBN-13: 978-8126508631
Course Outcomes:	Upon completion of the course, students will be able to: **CO1:** Identify truth tables of basic and flip-flop gates. **CO2:** Demonstrate realization of Boolean functions in SOP and POS forms through circuit design. **CO3:** Implement logic circuits using universal gates and multiplexers. **CO4:** Design asynchronous counters, synchronous counters and shift registers.

Name of the Programme : B.E in Computer Science and Engineering (Artificial Intelligence and Machine Learning)
Course Code : AIM-223
Title of the Course : Embedded System Design and Applications
Number of Credits : 3
Effective from AY : 2025-26

Pre-requisites for the Course:	**Basics of Electronics and Logic Design**	
Course Objectives:	**This course will enable students to:** 1. Identify the architectural features and core characteristics of embedded systems. 2. Explain the functions of microcontrollers and memory components within embedded applications. 3. Distinguish between various real-time tasks and scheduling algorithms used in RTOS environments. 4. Develop embedded system solutions utilizing platforms such as Arduino.	
Contents:		**No of Hours (45)**
Unit-1	**Introduction to Computing -** Numbering and Coding Systems, Digital Primer, Inside the Computer. **The 8051 Microcontrollers –** Microcontroller and Embedded Processors, overview of 8051 family – A brief history of the 8051, 8052 and 8031 **8051 Assembly Language Programming -** Inside the 8051, Introduction to 8051 Assembly Programming, Assembling and Running an 8051 program, the Program Counter and ROM Space in the 8051, 8051 Data Types and Directives, 8051 Flag Bits and the PSW Register. 8051 Register Banks and Stack **JUMP, LOOP, and Call Instructions –** Loop and Jump Instructions, Call Instructions, Time Delay for Various 8051 Chips	**12**
Unit - 2	**Introduction to Embedded Systems -** Application Domain of Embedded Systems, Desirable features and General Characteristics of Embedded Systems, Model of an Embedded System, Microprocessor vs Microcontroller, Example of a Simple Embedded System, Figures of Merit for an Embedded System, Classification of MCUs: 4/8/16/32-bits, History of Embedded Systems, Current Trends **Embedded Systems—The Hardware Point of View -** Microcontroller Unit (MCU), A popular 8-bit MCU, Memory for Embedded Systems, Low power Design. **Examples of Embedded Systems -** Mobile Phone, Automotive Electronics, Radio Frequency Identification (RFID), Wireless Sensor Networks (WISENET), Robotics, Biomedical Applications, Brain	**11**

	Machine Interface	
Unit - 3	**Real-time Operating Systems -** Real-time tasks, Real-time Systems, Types of Real-time Tasks, Real-time Operating Systems, Real-time Scheduling Algorithms, The Earliest Deadline First Algorithm, Qualities of a Good RTOS. **Embedded C Basics and Operators for Arduino –** Familiarizing with the Arduino IDE, Sketch Designing for Arduino, Board and Port Selection, Communication Interface Using Serial Port, Basic Understanding of the Code with Different Operations.	**11**
Unit - 4	**Embedded C Control Structure Blocks –** Program Flow and Control Structure, Looping Mechanisms, The Branching Operation based on Conditional Expressions, Difference between the C Language and Embedded C Language **Introduction to Arduino Mega -** Arduino Boards, Arduino Mega Specifications, Arduino Mega Pins and Ports **Communication with Arduino -** Communication Modules - Parallel Data Transfer, Serial Data Transfer, Synchronous Serial Communication, Asynchronous Serial Communication	**11**
Pedagogy:	Interactive, reflective, and inquiry-based methods, with a strong emphasis on critical thinking and problem-solving skills.	
References/ Readings:	**Text Books** 1. Lyla B. Das, Embedded Systems – An Integrated Approach, 1st Edition, Pearson Education, 2013, ISBN-13: 978-81-317-8766-3. 2. Muhammad Ali Mazidi, Janice Gillispie Mazidi, Rollin D. McKinlay: The 8051 Microcontroller and Embedded Systems, 2nd Edition, Pearson Education, 2011, ISBN-13: 978-81-317-5899-1. 3. Deepa M, Susithra N, Dr. G. Santhanamari: Embedded Systems, AICTE, 2024, ISBN-13: 978-93-6027-184-8. **Reference Books** 1. Raj Kamal: Embedded Systems-Architecture, Programming and Design, Tata McGraw-Hill Publishing Company Ltd, 2008, ISBN-13: 978-0-07-066764-8. 2. Shibu K V, Introduction to Embedded Systems, Second Edition 2017 McGraw Hill Education, ISBN -13: 9789339219680. 3. Simon Monk, The TAB Book of Arduino Projects, 1st Edition, Tab Books, 2014, ISBN-13: 978-0071825149. 4. Frank Vahid, Tony Givargis, Embedded System Design- A Unified Hardware/Software Introduction 2006, Wiley ISBN -13: 9788126508372	

Course Outcomes:	Upon completion of the course, students will be able to: **CO1:** List key components and application areas of embedded systems. **CO2:** Explain features of microcontrollers and memory types. **CO3:** Analyze RTOS-based scheduling and real-time system behavior. **CO4:** Develop simple embedded applications using standard platforms.

Name of the Programme : B.E in Computer Science and Engineering (Artificial Intelligence and Machine Learning)
Course Code : AIM-224
Title of the Course : Embedded System Design and Applications Lab
Number of Credits 1
Effective From AY : 2025-26

Pre-requisites for the Course:	**Basics of Microcontrollers and Digital Design**	
Course Objectives:	**This course will enable students to:** 1. Understand basic 8051 assembly programming. 2. Use Arduino boards for simple programming. 3. Apply Arduino programming concepts for basic input/output control. 4. Build embedded projects for real-world applications using Arduino.	
Contents:	**List of Programs/Experiments**	**No of Hours**
	1. Write an 8051 assembly language program to convert a decimal number to binary. 2. Write an 8051 assembly language program to add two 8-bit numbers. 3. Write an 8051 assembly language program to generate and display truth table for 3-input AND Gate. 4. Study of Arduino boards 5. Develop blinking and switch-based LED control using Arduino. 6. Develop Traffic Light System using Arduino. 7. Develop any application using stepper motor using Arduino. 8. Develop LDR-based street light control system using Arduino. 9. Develop digital speedometer using Arduino. 10. Develop an interface with Arduino using Ultrasonic Sensor. 11. Develop IOT-based humidity and temperature monitoring using Arduino. 12. Study of Serial and Parallel Communication using Arduino Mega.	**30**
Pedagogy:	Interactive, reflective, and inquiry-based methods, with a strong emphasis on critical thinking and problem-solving skills.	

References/ Readings:	**Text Books** 1. Lyla B. Das, Embedded Systems – An Integrated Approach, 1st Edition, Pearson Education, 2011, ISBN-13: 978-8131734211. 2. Muhammad Ali Mazidi, Janice Gillispie Mazidi, Rollin D. McKinlay: The 8051 Microcontroller and Embedded Systems, 2nd Edition, Pearson Education, 2011, ISBN-13: 978-81-317-5899-1. 3. Deepa M, Susithra N, Dr. G. Santhanamari: Embedded Systems, AICTE, 2024, ISBN-13: 978-93-6027-184-8. **Reference Books** 1. Raj Kamal: Embedded Systems-Architecture, Programming and Design, Tata McGraw-Hill Publishing Company Ltd, 2008, ISBN-13: 978-0-07-066764-8. 2. Shibu K V, Introduction to Embedded Systems, Second Edition, McGraw Hill Education India Private Limited, 2017, ISBN-13: 978-9339219680 3. Simon Monk, The TAB Book of Arduino Projects, 1st Edition, Tab Books, 2014, ISBN-13: 978-0071825149. 4. Frank Vahid, Tony Givargis, Embedded System Design- A Unified Hardware/Software Introduction, 2006, Wiley, ISBN-13: 978-8126508372.
Course Outcomes:	Upon completion of the course, students will be able to: **CO1:** Describe the basics of assembly language programming. **CO2:** Explain the various types of Arduino boards. **CO3:** Build simple circuits using basic components and Arduino boards. **CO4:** Develop small Arduino applications using sensors, motors, etc.

Multidisciplinary Course

Name of the Programme : B.E in Computer Science and Engineering (Artificial Intelligence and Machine Learning)
Course Code : XXXXXX
Title of the Course : Mathematical Foundation for AI and ML-I
Number of Credits : 3
Effective from AY : 2025-26

Pre-requisites for the Course:	Fundamental knowledge of Engineering Mathematics- I	
Course Objectives:	The course will enable the students to: 1. Analyze relations, functions, and integer concepts (prime factorization, GCD, modular arithmetic) 2. Apply propositional calculus, proof techniques, and counting methods. 3. Assess the properties of vector spaces, analyze linear independence and determine the rank of matrices. 4. Apply analytic geometry concepts and techniques like Gram-Schmidt and matrix decompositions.	
Contents:		**No of Hours**
Unit - 1	**Relations and Functions:** Relations and their properties, Equivalence Relations, partial orderings. One-to-One and Onto Functions, Inverse Function, Composition of functions. **Integers:** Integers and division, primes and greatest common divisors, Euclidean algorithm, Basic properties of Congruence, Modular Arithmetic, Fermat's Theorem.	**11**
Unit - 2	**Propositional Calculus:** Propositional logic, truth tables, propositional connectives (logical operators), propositional equivalences, propositional implications, consistency of statements, theory of inference in propositional logic, introduction to predicates and quantifiers. **Counting Principles**: Pigeonhole principle and Inclusion and Exclusion Principle.	**12**
Unit - 3	**Linear Algebra**: Vector Spaces, Null and Column Space of a Matrix, Linear Independence, Basis and Rank, Rank-Nullity theorem, Linear Mappings, One-to-One, Onto and Bijective Linear Maps(Isomorphisms), Matrix of a Linear Transformation. **Analytic Geometry:** Norms, Inner Products, Lengths and Distances, Angles and Orthogonality, Gram-Schmidt Orthogonalization, Orthonormal Basis, Orthogonal Complement, Inner Product of Functions.	**11**

Unit - 4	**Analytic Geometry:** Orthogonal Projections, Orthogonal Projection and the Normal Equation, Least Squares Problem, Rotations. **Decompositions**: Eigen decomposition and Diagonalization via orthogonal Transformation, Cholesky Decomposition, Singular Value Decomposition, Matrix Approximation.	11
Pedagogy:	Interactive, reflective, and inquiry-based methods, with a strong emphasis on critical thinking and problem-solving skills.	
References/ Readings:	**Text Books** 1. Swapan Kumar Sarkar; A Text Book of Discrete Mathematics; S. Chand Publication, Eighth Edition, 2019, ISBN 9352837355. 2. J. P. Tremblay and R. Manohar; Discrete Mathematical Structures with Applications to Computer Science; McGraw Hill, First Edition, 2017 ISBN**,** 9780074631133. 3. David C. Lay: Linear Algebra and Its Applications, Pearson Education India Fifth Edition, 2023, ISBN 9357059687. **Reference Books** 1. Kenneth H. Rosen; Discrete Mathematics and Its Applications; Tata McGraw Hill, Eighth Edition, 2021, ISBN 9390727359. 2. Marc Peter Deisenroth, A. Aldo Faisal, Cheng Soon Ong: Mathematics for Machine Learning, Cambridge University Press, 2020, ISBN 110845514X.	
Course Outcomes:	Upon completion of the course the student will be able to: **CO1:** Describe sets, functions, relations, and modular arithmetic for integer computations. **CO2:** Analyze combinatorial problems using logic and proof techniques. **CO3:** Solve vector spaces and apply linear transformations using matrices. **CO4:** Apply analytic geometry and matrix decomposition for multidimensional data analysis.	

Name of the Programme	**: B.E in Computer Science and Engineering (Artificial Intelligence and Machine Learning)**
Course Code	**: HIN-253**
Title of the Course	**: Sambhashan Kala**
Number of Credits	**: 2**
Effective from AY	**: 2025-26**

Name of the Programme	**: B.E in Computer Science and Engineering (Artificial Intelligence and Machine Learning)**
Course Code	**: MAR-253**
Title of the Course	**: Sambhashan Kaushalya**
Number of Credits	**: 2**
Effective from AY	**: 2025-26**

Name of the Programme	**: B.E in Computer Science and Engineering (Artificial Intelligence and Machine Learning)**
Course Code	**: KON-253**
Title of the Course	**: Sambhashan Kaushalya**
Number of Credits	**: 2**
Effective from AY	**: 2025-26**

Name of the Programme : B.E in Computer Science and Engineering (Artificial Intelligence and Machine Learning)
Course Code : SEC-XXX
Title of the Course : AI Powered App Development
Number of Credits 3
Effective from AY : 2025-26

Pre-requisites of the course:	Basics of UI/UX and Computer Programming	
Course Objectives	**This course will enable the students to:** • Understand the fundamentals of mobile and web application development, including front-end, backend, and development environments. • Learn and apply UI/UX design principles to build responsive mobile app interfaces using React Native components and layout systems. • Integrate backend services with mobile applications by using databases like Firebase for real-time and remote data operations. • Explore and utilize AI-powered low-code/no-code tools to accelerate app development, automate UI creation, and streamline deployment workflows.	
Contents:		**No of Hours**
PART A		
Unit - 1	**Unit – 1: Introduction to App Development** **Overview of mobile and web app development**: App development lifecycle, Basics of front-end and backend development. **Introduction to app frameworks**: Setting up development environments (Android Studio, React Native CLI), React Native **Programming basics for app development**: JavaScript/TypeScript	23
Unit - 2	**Building User Interfaces -** Basics of responsive design and mobile-first principles, Understanding UI/UX design principles for mobile apps, Designing layouts with Core Components, Flexbox, Grid, and constraints. **Navigation design**: Menus, Tab Bars, and Navigation Bars **Incorporating animations**: Transitions, fades, and button animations	22
Unit - 3	**Backend Integration and APIs-** Introduction to databases (SQLite, Firebase), Setting up Firebase for real-time storage, CRUD operations with SQLite and Firebase, Understanding REST APIs and JSON, Fetching, displaying, and sending data using APIs.	22
Unit - 4	**Version Control using Git and GitHub:** Create Repository, Push React Native Project to GitHub. **AI Coding Assistants (GitHub Copilot)** - UI Creation, Navigation & Multi-Screen App Structure. **Low Code Frameworks (DraftBit**): UI Creation, Code	23

	Generation, Backend Connection, Deployment. **No code Frameworks (Builder.ai)**: App Blueprinting, UI Creation, Code Generation, Backend Connection, Deployment. **Mini Project**: Develop and locally deploy a feature-rich app using learned concepts.	
Pedagogy	Integration of instructional learning, constructive thinking, inquiry-based, collaborative, experiential, and problem-solving approaches.	
References	**Text Books:** 1. Nader Dabit, React Native in Action, 1st Edition, Manning Publications, 2019, ISBN -13: 978-1617294051. 2. Dixitt, Shallabh, Ultimate AI-Assisted Development with GitHub Copilot: Unlock Faster AI-Powered Development, Testing and Automation in Java, Python, TypeScript, Go, and C++ with GitHub Copilot. New Delhi: Orange Education Pvt Ltd, 2025. ISBN 9349888203. 3. Agrawal, Ambuj. No-Code Artificial Intelligence: The New Way to Build AI Powered Applications. New Delhi: BPB Publications, 2023. ISBN 9355513496. **Reference Books:** 1. Tripling, Gwendolyn, and Michael Abel. Low-Code AI: A Practical Project-Driven Introduction to Machine Learning. 1st ed. Sebastopol, CA: O'Reilly Media, 2023. ISBN 1098146824. 2. Agnihotri, Ketan, and Pranali Dahale. React Development using TypeScript: Modern Web App Development Using Advanced React Techniques. New Delhi: BPB Publications, 2024. ISBN 978-9355517241. 3. Sakhniuk, Mikhail, and Adam Boduch. React and React Native — Fifth Edition: Build Cross-Platform JavaScript and TypeScript Apps for the Web, Desktop, and Mobile. 5th ed. Birmingham, UK: Packt Publishing, 2024. ISBN 1805127306. 4. Eisenman, Bonnie. Learning React Native: Building Native Mobile Apps with JavaScript. 2nd ed. Sebastopol, CA: O'Reilly Media, 2017. ISBN 1491989149.	
Course Outcomes	**Upon completion of the course, students will be able to:** **CO1:** Explain the fundamentals of mobile app development, including development environments, and the overall app development lifecycle. **CO2:** Design user interfaces for mobile applications using responsive design principles, UI/UX concepts, navigation structures, and basic animations. **CO3:** Integrate backend services with mobile applications by working with databases (SQLite/Firebase), performing CRUD operations. **CO4:** Apply AI-powered, low-code, and no-code tools to accelerate app development, automate UI creation, and deploy functional mobile applications.	

SEMESTER-IV

Major Courses
Name of the Programme : B.E in Computer Science and Engineering (Artificial Intelligence and Machine Learning)
Course Code : AIM-204
Title of the Course : Modern Database Management Systems
Number of Credits : 2
Effective from AY : 2025-26

Pre-requisites for the Course:	Fundamental Knowledge of Computer programming and Applications	
Course Objectives:	1. To enable students to design Entity–Relationship (ER) models and map them to relational schemas. 2. To develop skills in writing SQL queries for data definition, manipulation, and retrieval. 3. To understand normalization techniques for effective database design. 4. To introduce the fundamentals of NoSQL databases and their data models.	
Contents:		**No of Hours**
Unit - 1	**Introduction**: Introduction to database, Characteristics of Database approach, Advantages of using DBMS approach, Three-schema architecture and data independence. **Entity Relationship Model**: Using High-Level Conceptual Data Models for Database Design; Entity Types, Entity Sets, Attributes and Keys; Relationships, Relationship types, Roles and Structural Constraints; Weak Entity Types. ER-Relational Mapping Rules.	**8**
Unit - 2	**Relational Algebra** :Relational Algebra Operations ,SELECT , PROJECT , RENAME Operation, Set Theoretic Operations, JOIN Operation, DIVISION Operation, Relation Algebra Queries **SQL**: SQL Data Definition and Data Types; Specifying basic constraints in SQL; Schema change statements in SQL; Insert, Delete and Update statements in SQL, Basic queries in SQL; More complex SQL Retrieval Queries; Nested and Correlated Queries, IN, ALL, EXIST operators.	**7**
Unit - 3	**Database Design**: Informal Design Guidelines for Relation Schemas; Functional Dependencies; Normal Forms Based on Primary Keys; General Definitions of Second and Third, Fourth Normal Forms; Boyce-Codd Normal Form.	**7**
Unit - 4	**NoSQL Databases**: Limitations of Relational Databases, Motivation for Not Just/No SQL (NoSQL) Databases. **Key-Value Database**: Data Modelling terms, Basic operations (CRUD) and Applications.	**8**

	Document Databases: What Is a Document, Key-Value Pairs, Collections, Basic Operations (CRUD), and Applications. **Column Family Database**: Basic Components, Column Families, Applications. **Graph Database:** Elements of Graphs, Operations on Graphs (Union, Intersection, Traversal), Applications, Differences and Similarities between Key-Value, Document, Column Family and Graph Databases.	
Pedagogy:	Interactive, reflective, and inquiry-based methods, with a strong emphasis on critical thinking and problem-solving skills.	
References/ Readings:	**TextBook** 1. Elmasri and Navathe, Fundamentals of Database Systems, 7th Edition, Pearson Education 2013, ISBN 978-9332582705 2. Dan Sullivan, NoSQL for Mere Mortals, Addison-Wesley Professional / Pearson Education 2015, 1st Edition, ISBN 978-0134023212 **Reference Books** 1. Raghu Ramakrishnan and Johannes Gehrke, Database Management Systems, McGraw-Hill, 3rd Edition, 2013, ISBN 978-0072465631 2. Abraham Silberschatz, Henry F. Korth, S. Sudarshan, Database System Concepts, McGraw Hill, 6th Edition, 2010, ISBN 978-0073523323 3. Pramod J. Sadalagi & Basavaraj M. Patil "NoSQL: Database for Storage and Retrieval of Data in Cloud" —Wiley India Pvt. Ltd., 1st Edition, 2015, ISBN 978-1498784368	
Course Outcomes:	**Upon completion of the course, students will be able to:** **CO1:** Describe ER modeling concepts to design structured database schemas. **CO2:** Use Relation algebra and SQL queries to create, update, and retrieve data from relational databases. **CO3:** Apply normalization rules to refine and optimize relational schemas. **CO4:** Implement operations on NoSQL database models (Key-Value, Document, Column, and Graph).	

Name of the Programme : B.E in Computer Science and Engineering (Artificial Intelligence and Machine Learning)
Course Code : AIM-205
Title of the Course : Modern Database Management Systems Lab
Number of Credits : 2
Effective from AY : 2025-26

Pre-requisites for the Course:	Basic knowledge of Relational Database Management System	
Course Objectives:	The course will enable students to: 1. Develop the ability to design robust database models using Entity–Relationship diagrams. 2. Build proficiency in SQL operations 3. Apply advanced SQL and NoSQL concepts. 4. Integrate theoretical and practical skills in a mini-project.	
Content:	**List of Programs**	**No. of Hours**
	RDBMS 1. Creating Entity-Relationship Diagram using online tools. 2. Implementation of DDL commands in SQL. 3. Implementation of DML commands in SQL. 4. Basic SQL Queries. 5. Implementation of different types of constraints. 6. Implementation of different types of function with suitable examples. 7. Implementation of different types of Joins. 8. Implementation of Nested Queries and correlated Queries. 9. Implementation of Group by, Order by, having clause. 10. Implementation of Stored Procedures using RDBMS 11. Implementation of Triggers using RDBMS **NoSQL** 12. Querying Key-value Databases 13. Operations on Document Databases. 14. Operations on Graph Databases 15. Mini Project on RDBMS/NoSQL Databases	**60**
Pedagogy:	Integration of instructional learning, constructive thinking, inquiry-based, collaborative, experiential, and problem-solving approaches.	

References:	**Text Books** 1. Ramez Elmasri, Shamkant B.Navathe, Fundamental of Database systems, 7th Edition Pearson, 2018, ISBN 978-9332582705 2. Abraham Silberschatz, Henry F. Korth, S.Sudarshan, Database System Concepts 6th Edition, MC Graw HillI, 2013, ISBN 978-0073523323 3. Shannon Bradshaw, Eoin Brazil, and Kristina Chodorow.,MongoDB: The Definitive Guide - Powerful and Scalable Data Storage, 3rd Edition, ISBN 9781491954461 4. Dan Sullivan, NoSQL for Mere Mortals, Addison-Wesley Professional Pearson Education 2015, 1st Edition, ISBN 978-0134023212 **Reference Books:** 1. Raghu Ramkrishnan, Johannes Gehrke, Database Management Systems, 3rd Edition McGraw-Hill,2002, ISBN 0072465638 2. C.J. Date, A.Kannan, S. Swami Nadhan, An Introduction to Database systems, Pearson, 8th Edition, 2004, ISBN 8177585568

Course Outcomes:	Upon completion of the course, students will be able to: **CO1:** Design ER diagrams and create corresponding database schemas. **CO2:** Apply SQL commands (DDL, DML, constraints, functions, joins) to manage relational data. **CO3:** Analyze and execute advanced SQL and basic NoSQL queries. **CO4:** Develop mini-projects demonstrating complete database design and implementation.

Name of the Programme : B.E in Computer Science and Engineering (Artificial Intelligence and Machine Learning)
Course Code : AIM-206
Title of the Course : Foundations of Machine Learning
Number of Credits : 3
Effective from AY : 2025-26

Pre-requisites for the Course:	Basic knowledge of Artificial Intelligence	
Course Objectives:	**This course will enable the students to:** 1. Understand foundational concepts, terminology, and workflow of machine learning. 2. Explore and Understand datasets through preprocessing, feature extraction, and exploratory data analysis. 3. Apply core supervised learning algorithms. 4. Execute core unsupervised learning algorithms.	
Contents:		**No of Hours**
Unit - 1	**Introduction to Machine Learning:** Need for Machine learning, Machine Learning Explained, Machine Learning in Relation to Other Fields, Types of Machine Learning, Challenges of Machine Learning, Machine Learning Process, and Machine Learning Applications. **Understanding Data:** Elements of Big Data, Types of Data, Data Storage and representation, Data Collection, Data Preprocessing, Descriptive Statistics, Univariate Data Analysis and Visualization, Bivariate data and Multivariate Data.	10
Unit - 2	**Basics Of Learning Theory:** Introduction to Learning and its Types, Introduction to Computational Learning Theory, Design of a learning System, Introduction to Concept learning, Induction Biases **Similarity Based Learning:** Introduction to similarity or Instance based learning, nearest neighbor learning, Weighted K-nearest neighbor algorithm, Nearest Centroid classifier, locally weighted Regression. **Regression Analysis:** Introduction to Regression, Introduction to Linearity, Correlation and Causation. Introduction to Linear Regression, Validation of Regression Methods, Multiple Linear Regression.	10
Unit - 3	**Decision Tree Learning:** Introduction to Decision tree learning Model, Decision Tree Induction Algorithms, ID3, C4.5, CART, Validating and Pruning Decision Trees. **Bayesian Learning:** Introduction to Probability Based Learning, Fundamentals of Bayesian Theorem, Classification using Bayes Model, Naive Bayes Algorithm	12

Unit - 4	**Clustering Algorithms:** Introduction to Clustering Approaches, Proximity Measures, Hierarchical Clustering Algorithms, Partitional Clustering Algorithm, Density - Based Methods, GRID Based Approach, Probability Based Methods, Cluster Evaluation Based Methods.	13
Pedagogy:	Interactive, Reflective, and Inquiry based methods with strong emphasis on critical thinking and problem-solving skills.	

References/ Readings:	**Text Books:** 1. S Sridhar, M Vijayalakshmi , MACHINE LEARNING, Oxford University Press 1st Edition 2021, ISBN:978-0190127275 2. M N Murty, Ananthanarayana V S, Machine Learning: Theory and Practice, Universities Press 1st Edition 2023, ISBN: 978-9393330697 3. Dr Ruchi Doshi, Dr Kamal Kant Hiran, Machine Learning- Master Supervised and Unsupervised Learning Algorithms with Real Examples, BPB Publications 2021, ISBN:978-9391392352 **Reference Books:** 1. Tom Mitchell – Machine Learning, McGraw-Hill 1st edition 2017. ISBN: 978-1259096952 2. Christopher Bishop, Pattern Recognition and Machine Learning, Springer 2009, ISBN: 978-1493938438. 3. Trevor Hastie, Robert Tibshirani, Jerome Friedman, The Elements of Statistical Learning, Springer 1st edition, 2017 ISBN: 978-0387848570. 4. Ethem Alpaydin, Introduction to Machine Learning, MIT Press 4th edition 2020, ISBN: 978-0262043793. 5. Shai Shalev-Shwartz & Shai Ben-David, Understanding Machine Learning, Cambridge University Press 1st Edition 2014, ISBN: 978-1139948517
Course Outcomes:	**Upon completion of the course, students will be able to:** **CO1:** Describe fundamental machine learning concepts, processes, and use cases. **CO2:** Perform data preprocessing, feature engineering, and exploratory analysis. **CO3:** Apply basic supervised learning algorithms to practical problems. **CO4:** Execute unsupervised learning algorithms for practical problems**.**

Name of the Programme : B.E in Computer Science and Engineering (Artificial Intelligence and Machine Learning)
Course Code : AIM-207
Title of the Course : Foundations of Machine Learning Lab
Number of Credits : 1
Effective from AY : 2025-26

Pre-requisites for the Course:	Basic Knowledge of Python Programing	
Course Objectives:	**This course will enable the students to:** 1. Understand fundamental ML data preprocessing techniques. 2. Perform exploratory data analysis using Python libraries. 3. Execute supervised learning algorithms. 4. Apply unsupervised learning algorithms.	
Content:	**List of Programs**	**No of Hours**
	1. Introduction to Python ML ecosystem: NumPy, Pandas, Matplotlib. 2. Loading and exploring datasets: summary statistics, handling missing values. 3. Data preprocessing: normalization, standardization, encoding categorical features. 4. Exploratory Data Analysis: univariate, bivariate, multivariate visualizations. 5. Implementing k-Nearest Neighbors classifier. 6. Linear Regression: model fitting, residual analysis, evaluation. 7. Multiple Linear Regression and feature significance. 8. Decision Tree classifier: training, visualization, pruning. 9. Naïve Bayes classifier on text or categorical datasets. 10. Clustering with K-Means: inertia, silhouette score. 11. Hierarchical clustering and dendrogram analysis. 12. Mini project: Complete ML pipeline on a chosen dataset.	**30**
Pedagogy:	Hands-on programming, exploratory analysis, implementation-driven learning using Python.	

References/ Readings:	**Text Books:** 1. Dr. Harsh Bhasin, Machine Learning for Beginners - 2nd Edition 2020, BPB Publications, ISBN13: 978-9389845426 2. Sebastian Raschka & Vahid Mirjalili, Python Machine Learning, Packt Publishing, 3rd Edition 2019, ISBN13: 978-1789955750 3. Aurélien Géron, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, O'Reilly Media, 2nd Edition 2019, ISBN13: 978-1492032649 **Reference Books:** 1. Michael Bowles, Machine Learning in Python, Wiley, 1st Edition, 2015, ISBN13: 978-1118961742 2. Andreas C. Müller & Sarah Guido, Introduction to Machine Learning with Python, O'Reilly Media, 1st Edition 2016, ISBN13: 978-1449369415 3. Joel Grus, Data Science from Scratch, O'Reilly Media, 2nd Edition 2019, ISBN13: 978-1492041139 4. Toby Segaran, Programming Collective Intelligence, O'Reilly Media, 1st Edition 2007, ISBN13: 978-0596529321 5. M. Gopal, Applied Machine Learning, McGraw-Hill Education, 1st Edition 2019, ISBN13: 978-9385965590
Course Outcomes:	**Upon completion of the course, students will be able to:** **CO1:** Apply Python tools to analyze and manipulate datasets. **CO2:** Perform exploratory data analysis using Python libraries. **CO3:** Execute supervised learning algorithms. **CO4:** Implement unsupervised learning algorithms.

Name of the Programme : B.E in Computer Science and Engineering (Artificial Intelligence and Machine Learning)
Course Code : AIM-208
Title of the Course : Modern Operating Systems
Number of Credits : 3
Effective from AY : 2025-26

Pre-requisites of the course:	Basic knowledge of computers and programming	
Course Objectives	This course will enable the students to: 1. Understand Operating System Basics concepts, evolution, functions, and structures. 2. Gain Process and Memory Management Skills to learn process scheduling, synchronization, deadlock handling, and memory management techniques. 3. Explore File and Storage Systems to understand file system design, storage management, and UNIX fundamentals with shell scripting. 4. Analyze advanced OS concept using cloud computing, parallel and distributed systems and virtualization.	
Contents:		No of Hours
Unit - 1	**Introduction to Operating Systems**: Definitions, Operating-System Structure. **Process Management** : Processes - Concept, Process Scheduling, Threads - Multithreaded Programming - Overview, Multicore Programming, Multithreading Models. **Process Scheduling** - Basic Concepts, Scheduling Criteria, Scheduling Algorithms. **Process Synchronization**: Race condition, The Critical-Section Problem, Peterson's Solution, Synchronization Hardware, Mutex Locks, Semaphores, Monitors-Monitors Usage, Classic Problems of Synchronization - The Bounded-Buffer Problem, The Readers – Writers Problem.	11
Unit - 2	**Deadlocks**: System Model, Deadlock Characterization, Methods for Handling Deadlocks, Deadlock prevention, Deadlock avoidance, Deadlock Detection, Recovery from Deadlock. **Memory management**: Background, Contiguous memory allocation, Paging, Swapping. **Virtual memory**: Background, Demand paging, Page replacement, Thrashing.	11

Unit - 3	**Mass storage structure**: Overview, HDD Scheduling, Storage Device Management, RAID Structure- Levels. **File system**: File Concepts, Access methods, Directory Structure, File-System Structure, File-System Operations, Directory Implementation, Allocation methods, Free-Space Management. **The UNIX Architecture, Shell Commands and Basic Shell Scripts**: The UNIX architecture, cal, date, echo, printf, bc, passwd, who, uname, pwd , ls, cd, mkdir, rmdir, Absolute Pathname , Relative Pathname, cat, cp, rm, mv, file, wc, cmp , comm, diff, zip and unzip, chmod, head, tail, grep, Basic Shell Scripts. **Introduction to Android**: Characteristics, Understanding the Android Software Stack.	12
Unit - 4	**Introduction to Cloud Computing:** Definition, Deployment models: Public, Private, Hybrid, and Community Clouds, Cloud Reference models: IaaS, PaaS, and SaaS, Characteristics and benefits, Challenges, Historical developments. Computing Platforms and Technologies (AWS, GCP, Azure) **Distributed Computing**: Eras of Computing, Parallel vs Distributed, Distributed Computing – Definition, Components, System architecture styles (Client-Server, Peer-to-Peer) **Virtualization:** Introduction, Virtualization Reference Model, Features, Hardware-level Virtualization – Hypervisors, Techniques (Full, Para and Partial), OS-Level Virtualization, Pro and Cons of Virtualization.	**11**
Pedagogy	Interactive, reflective, and inquiry-based methods, with a strong emphasis on critical thinking and problem-solving skills.	
References	**Text Books:** 1. Abraham Silberschatz, Peter B Galvin, Gerg Gagne; The Operating System Concepts; Wiley Publications, Global Edition, 2025, ISBN 9789357460569. 2. Sumitabha Das, UNIX Concepts and Applications, McGraw Hill Education, Fourth Edition, 2006, ISBN-13 978-0-07-063546-3. 3. Rajkumar Buyya , Christian Vecciola, S. Thamrai Selvi, Mastering Cloud Computing , 2nd Edition, 2024, McGraw Hill, ISBN-13: 978-9355329509 **Reference Books:** 1. Andrew S. Tanenbaum, Herbert Bos, Modern Operating Systems, Fourth Edition,2016, ISBN 978-93-325-7577-6 2. B.M. Harwani, Android Programming Unleashed, Pearson, First Edition,2013, ISBN 978-93-325-1584-0 3. William Stallings, Operating Systems: Internals and Design Principles, Pearson Education ,9th Edition,2018, ISBN-13 978-9352866717	

	4. Achyut S. Godbole, Operating Systems, Tata McGraw Hill, McGraw Hill Education, Third Edition, 2017, ISBN-13978-0070702035
Course Outcomes	**Upon completion of the course, students will be able to:** **CO1:** Describe the core concepts, structure, and process scheduling, synchronization. **CO2:** Explain Deadlocks and memory management techniques effectively. **CO3:** Apply file system operations and shell scripting to manage and automate system tasks. **CO4:** Use cloud, parallel, distributed, and virtualization concepts for scalable computing.

Name of the Programme : B.E in Computer Science and Engineering (Artificial Intelligence and Machine Learning)
Course Code : AIM-209
Title of the Course : Modern Operating Systems Lab
Number of Credits : 1
Effective from AY : 2025-26

Pre-requisites for the Course:	Basic knowledge of programming	
Course Objectives:	**This course will enable the students to:** • Demonstrate CPU scheduling and synchronization problems in operating systems. • Simulate deadlock handling and page replacement algorithms • Implement disk scheduling algorithms and use shell scripts with basic commands for system tasks. • Develop applications using Android and cloud.	
Content:	**List of Programs**	**No of Hours**
	1. Implementation of non-pre-emptive CPU Scheduling algorithms 2. Implementation of pre-emptive CPU Scheduling algorithms 3. Implementation of Producer - Consumer Problem using Semaphores. 4. Simulate algorithm for deadlock prevention. 5. Simulate algorithm for Deadlock avoidance using Banker's Algorithm. 6. Simulate memory allocation methods: (i) Best Fit (ii) Worst Fit and (iii) Next Fit. 7. Implementation of page replacement algorithm. 8. Implementation of Disk Scheduling algorithms. 9. Implementation of Basic Shell Commands. 10. Implementation of Simple Shell Scripts. 11. Implementation of virtualization on cloud. 12. Mini project - Android App Development.	**30**
Pedagogy:	Integration of instructional learning, constructive thinking, inquiry-based, collaborative, experiential, and problem-solving approaches.	

References/ Readings:	**Text Books:** 1. Abraham Silberschatz, Peter B Galvin, Gerg Gagne; The Operating System Concepts; Wiley Publications, Global Edition, 2025, ISBN 9789357460569. 2. Sumitabha Das, UNIX Concepts and Applications, McGraw Hill Education, Fourth Edition, 2006, ISBN-13 978-0-07-063546-3. 3. Rajkumar Buyya , Christian Vecciola, S. Thamrai Selvi, Mastering Cloud Computing , 2nd Edition, 2024,McGraw Hill, ISBN-13: 978-9355329509 4. B.M. Harwani, Android Programming Unleashed, Pearson, First Edition,2013, ISBN 978-93-325-1584-0 Reference Books: **5.** Andrew S. Tanenbaum, Herbert Bos,Modern Operating Systems, Fourth Edition,2016, ISBN 978-93-325-7577-6 6. William Stallings, Operating Systems: Internals and Design Principles, Pearson Education ,9th Edition,2018, ISBN-13 978-9352866717 7. Achyut S. Godbole, Operating Systems, Tata McGraw Hill, McGraw Hill Education, Third Edition, 2012, ISBN 13- 9781259083990.
Course Outcomes:	**Upon completion of the course, students will be able to:** **CO1:** Demonstrate non-pre-emptive and pre-emptive CPU scheduling algorithms. **CO2:** Simulate deadlock handling algorithms and page replacement algorithms. **CO3:** Implement disk scheduling algorithms and execute shell scripts and basic shell commands for system tasks. **CO4:** Develop basic android application and explore virtualization on cloud.

Name of the Programme : **B.E in Computer Science and Engineering (Artificial Intelligence and Machine Learning)**
Course code : **XXXXXX**
Title of the course : **Mathematical Foundation for AI & ML-II**
Number of Credits : **4**
Effective from AY : **2025-26**

Pre-requisites for the Course:	Knowledge of differential calculus, matrix theory and elementary probability theory.	
Course Objectives:	This course will enable students to: 1. Understand probability spaces, discrete and continuous random variables, and special distributions in the context of artificial intelligence and machine learning. 2. Understand sampling distributions and the Central Limit Theorem's role in estimating means, and explore hypothesis testing concepts and various tests for means, in relation to artificial intelligence and machine learning. 3. Gain exposure to Vector and Matrix Calculus, which is an essential tool for solving optimization problems in artificial intelligence and machine learning. Units 4. Learn unconstrained and constrained optimization methods used in artificial intelligence and machine learning.	
Contents:		**No of Hours**
Unit - 1	**Probability Distributions**: Probability Space, Discrete and Continuous Random Variables and probability Distributions, Some Special Distributions (Bernoulli, Binomial, Poisson, Geometric, Exponential, Uniform, and Gaussian distributions), Moment Generating Function (computation of first two moments of specific probability distributions).	**11**
Unit - 2	**Sampling Distributions**: Definition of Sampling distribution, Sampling distribution of the Mean with variance known, Central Limit Theorem (Sampling distribution of the Mean for a Large Sample when variance is not known). **Confidence Interval Estimation**: Concept, Confidence interval for the mean of a normal population with variance known, Large Sample Confidence interval for a population mean (variance not known). Tests of Hypotheses: Statistical Hypotheses, Null Hypothesis, alternative hypothesis, Type I and Type II Errors, Level of Significance, Test on the mean of a normal population with variance known, Large Sample Test on a Population mean (variance unknown).	**12**

Unit - 3	**Differential Calculus**: Partial Derivative, Directional Derivative, Gradient (Total Derivative) of scalar valued Functions of several variables, Jacobian (Total Derivative) of vector valued Functions of several variables, Hessian of scalar valued functions of several variables, Linearization and Multivariate Taylor Series, Derivative (Gradient) of scalar valued functions of matrices.	**11**
Unit - 4	**Unconstrained Optimization**: Gradients and Stationary Points, Second Derivative Test for functions of several variables (up to three variables), Gradient Descent method, Stochastic Gradient Descent method. **Constrained Optimization**: Lagrange's Multiplier Method, Standard form of linear programming problem, Basic Simplex algorithm.	11

Pedagogy:	Interactive, reflective, and inquiry-based methods, with a strong emphasis on critical thinking and problem-solving skills.
References/ Readings:	**Text Books** 1. Douglas C. Montgomery, George C. Runger, "Applied Statistics and Probability for Engineers", Fourth Edition, Wiley-India, 2012, ISBN 9788126523153. 2. E. K. P. Chong and S. H. Zak, "An Introduction to Optimization with applications to machine learning", 4th Edition, John Wiley and Sons Pvt. Ltd., 2013, ISBN 9781118279014. 3. George B. Thomas, Maurice D. Weir Naval, Joel Hass, Christopher Heil "Thomas' Calculus - Early Transcendentals", Thirteenth Edition, Pearson, 2014, ISBN-10: 0-321-88407-8. 4. Marc Peter Deisenroth, A. Aldo Faisal, Cheng Soon Ong, "Mathematics for Machine Learning", Cambridge University Press, 2020, ISBN-10: 110845514X. **Reference Books** 1. Jay Dawani, "Hands-On Mathematics for Deep learning", First Edition, Packt Publishing Ltd., 2020, ISBN-10: 1838647295. 2. Jorge Nocedal, Stephen J. Wright, "Numerical optimization", 2nd Edition, Springer, 2006, ISBN-10: 0387303030. 3. Richard A. Johnson, Irwin Miller, John Freund, "Miller and Freund's Probability and Statistics for Engineers", Eight Edition, Pearson India Education services Pvt. Ltd., 2018, ISBN-10: 0321640772.

Course Outcomes:	On completion of the course students should be able to: **CO1:** Describe various probability distributions to model uncertainties arising in artificial intelligence and machine learning. **CO2:** Formulate and test statistical hypotheses to validate artificial intelligence and machine learning models **CO3:** Apply vector and matrix calculus techniques to effectively compute gradients required to train models arising in artificial intelligence and machine learning. **CO4:** Analyze various optimization techniques to solve problems arising in artificial intelligence and machine learning.

Professional Electives

Name of the Programme : B.E in Computer Science and Engineering (Artificial Intelligence and Machine Learning)
Course Code : AIM-225
Title of the Course : Design and Analysis of Algorithm
Number of Credits : 3
Effective From AY : 2025-26

Pre-requisites for the Course:	Fundamentals of data structures and basic computer programming.	
Course Objectives:	**This course will enable students to:** 1. Understand recursion and algorithm thinking. 2. Analyze different paradigms such as Divide & Conquer, Greedy, and Dynamic Programming. 3. Differentiate between complexity classes such as P and NP and describing their relationships and significance. 4. Demonstrate how different algorithm design approaches are used to solve various classes of engineering problems.	
Contents:		**No. of Hours**
Unit - 1	**Algorithms & Algorithm Strategy:** Algorithm, classification of Algorithms, how to design an algorithm. **Complexity of Algorithms**: Analysis of algorithms, Complexity, the asymptotic Notations, Relational Properties to be applied on Asymptotic Notations, Standard Notations and common functions, Basic Efficiency Classes, Logarithmic Rules, Recurrence Relation.	**13**
Unit - 2	**Divide and Conquer Algorithms:** Introduction, General Divide & Conquer Recurrence, Binary search, Finding Maximum and Minimum, Merge sort technique, Quick sort technique. **Greedy Algorithm:** Optimization Problem, Greedy Algorithm, Knapsack Problem, Job Sequencing problem with deadlines, Optimal storage on tapes, Optimal merge patterns, Single source shortest paths.	**12**
Unit - 3	**Dynamic programming:** Introduction, Matrix Chain Multiplication, 0/1 Knapsack Problem, Travelling Salesman Problem. **Graph Theory:** Minimum Spanning Tree, Single Source Shortest Path, All Pairs Shortest Path Problem, The Floyd-Warshall Algorithm.	**10**

Unit - 4	**Backtracking Algorithms:** Introduction, N-Queen Problem, Sum of subsets, Graph Coloring, Hamiltonian Cycle. **String-Matching Algorithms:** String Matching, Notations and Terminology, Lemma, Naïve String matching algorithm, Rabin-Karp Algorithm. **P and NP Problems:** Introduction, Polynomial (P) Problems, Nondeterministic Polynomial (NP) Problems, NP hard problems.	**10**
Pedagogy:	Interactive, reflective, and inquiry-based methods, with a strong emphasis on critical thinking and problem-solving skills.	
References/ Readings:	**Text Books** 1. Shefali Singhal and Neha Garg: Analysis and Design of Algorithms, BPB Publications, First Edition, 2018, ISBN-10: 9386551896. 2. Ellis Horowitz, Sartaj Sahni, S. Rajasekharan: Fundamentals of Computer Algorithms, Universities Press, 2nd Edition, 2008, ISBN-10: 8173716129. 3. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein: Introduction to Algorithms, PHI, 3rd Edition, 2009, ISBN-13: 9780262033848 **Reference Books** 1. Michael T Goodrich and Roberto Tamassia: Algorithm Design: Foundations, Analysis, and Internet Examples, Second Edition, Wiley, 2002, ISBN-10: 0471383651. 2. Gilles Brassard and Paul Bratley: Fundamentals of Algorithmic, PHI, 1996, ISBN-10:0133350681. 3. Jon Kleinberg and ÉvaTardos: Algorithm Design, 1st Edition, Pearson, 2005, ISBN-10: 0321295358. 4. Alfred V. Aho, John E. Hopcroft & Jeffrey D. Ullman: Design and Analysis of Computer Algorithms, Addison-Wesley Publishing Company, First Edition, 1974,ISBN-10: 0201000296 5. S. Sridhar: Design and Analysis of Algorithms, Oxford University Press (India),First Edition, 2014,ISBN-10: 0198093691	
Course Outcomes:	Upon completion of the course, students will be able to: **CO 1:** Describe recursion and algorithm thinking. **CO 2:** Explain various paradigms such as Divide & Conquer, Greedy, and Dynamic Programming. **CO 3:** Analyze classes such as P and NP and explain their relationships and significance within computational theory. **CO 4:** Demonstrate how the different algorithm design approaches are used to solve various classes of engineering problems.	

Name of the Programme : B.E in Computer Science and Engineering (Artificial Intelligence and Machine Learning)
Course Code : AIM-226
Title of the Course : Design and Analysis of Algorithm Lab
Number of Credits : 1
Effective From AY : 2025-26

Pre-requisites for the Course:	Basic knowledge of data structures and computer programming	
Course Objectives:	The course will enable students to: 1. Understand the time complexity of various algorithms. 2. Apply divide-and-conquer, greedy, and dynamic programming techniques to solve classical algorithmic problems. 3. Implement fundamental algorithms for graph theory. 4. Develop proficiency in problem-solving and algorithm design for computationally challenging problems.	
Contents:	**List of Programs**	**No of Hours**
	1. Implementation of time complexity of various algorithms. 2. Implementation of Merge Sort technique using Divide and Conquer Algorithm. 3. Implementation of Quick Sort technique using Divide and Conquer Algorithm. 4. Implementation of Knapsack problem using Greedy Algorithm. 5. Implementation of 0/1 knapsack Problem using Dynamic Programming. 6. Implementation of Travelling Salesman Problem using Dynamic Programming. 7. Implementation of Floyd-Warshall Algorithm. 8. Implementation of N-Queen Problem. 9. Implementation of Graph Coloring. 10. Implementation of Hamiltonian Cycle. 11. Implementation of Pattern Matching Algorithms. 12. Case study on P and NP, NP Hard class problems.	**30**
Pedagogy:	Integration of instructional learning, constructive thinking, inquiry-based, collaborative, experiential, and problem-solving approaches.	
	Text Books 1. Ellis Horowitz, Sartaj Sahni, S. Rajasekharan: Fundamentals of Computer Algorithms, Universities Press, 2nd Edition, 2008, ISBN-10: 8173716129. 2. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein: Introduction to Algorithms, PHI, 3rd Edition, 2009, ISBN-13: 9780262033848	

References/ Readings:	**Reference Books** 1. Michael T Goodrich and Roberto Tamassia: Algorithm Design: Foundations, Analysis, and Internet Examples, Second Edition, Wiley, 2002, ISBN-10: 0471383651. 2. Gilles Brassard and Paul Bratley: Fundamentals of Algorithmic, PHI, 1996, ISBN-10:0133350681. 3. Jon Kleinberg and ÉvaTardos: Algorithm Design, 1st Edition, Pearson, 2005, ISBN-10: 0321295358. 4. Alfred V. Aho, John E. Hopcroft & Jeffrey D. Ullman: Design and Analysis of Computer Algorithms,Addison-Wesley Publishing Company,First Edition, 1974,ISBN-10: 0201000296 5. S. Sridhar: Design and Analysis of Algorithms,Oxford University Press (India),First Edition, 2014,ISBN-10: 0198093691
Course Outcomes:	Upon completion of the course, students will be able to: **CO1:** Explain algorithm performance by calculating time complexity for different problem-solving approaches. **CO2:** Demonstrate the ability to choose suitable algorithmic strategies for a given computational problem using divide-and-conquer, greedy, and dynamic programming techniques. **CO3:** Apply algorithmic techniques to model and solve key graph and combinatorial challenges, such as shortest paths, Hamiltonian cycles, coloring problems, and N-Queens. **CO4:** Evaluate algorithm correctness and efficiency through practical implementation and problem-solving exercises.

Name of the Programme : **B.E in Computer Science and Engineering (Artificial Intelligence and Machine Learning)**
Course Code : **AIM-227**
Title of the Course : **Fundamentals of Software Engineering**
Number of Credits : **3**
Effective from AY : **2025-26**

Pre-requisites for the Course:	Basic knowledge of computer programming	
Course Objectives:	This course enables students to: 1. Understand the foundational concepts of software engineering and compare various software development process models. 2. Explain requirements engineering principles to analyse, document, and validate software system requirements. 3. Apply major software design techniques used in developing structured and object-oriented software solutions. 4. Demonstrate knowledge of coding practices, software testing techniques, and software project metrics for evaluating software quality and project performance.	
Contents:		**No of Hours**
Unit - 1	**Basics of software engineering**: Principles of software engineering, Software characteristics, Software applications, Phases of software engineering, Software process, project, and product, Process assessment, Software process, Capability Maturity Model. **Software Life Cycle Models**: Waterfall Model, Spiral Model, Incremental model, Prototyping model, Object-oriented model, Agile model, Rapid Application Development model, Iterative enhancement model, V-Model, Extreme Programming.	**11**
Unit - 2	**Software requirements:** Functional requirements, Non-functional requirements, User requirements, System requirements, Software Requirements Document. **Software Requirements Engineering Process:** Feasibility study, Requirements elicitation and analysis, Requirements validation techniques, Software prototyping, Requirements management.	**11**
Unit - 3	**Basics of software design:** Architectural design, Component-level design, User interface design, Fundamental design concepts, Design techniques. **Object-Oriented Modelling using UML:** UML overview, nature and purpose of models. Use case diagrams, class diagrams, activity diagram, sequence diagram, interaction diagram.	**11**

Unit - 4	**Coding techniques:** Structured coding techniques, Coding style, Coding methodology, Code verification techniques, Coding tools, Code documentation, Code standards and guidelines. **Software Testing Strategies:** Strategic approach to software testing, Testing strategies for conventional software, Black box testing, White box testing, Validation testing, System testing, Debugging. **Software Metrics:** Software quality metrics, Metrics for analysis model, Metrics for design model, Metrics for source code, Metrics for testing, Metrics for software maintenance.	**12**
Pedagogy:	Interactive, reflective, and inquiry-based methods, with a strong emphasis on critical thinking and problem-solving skills.	
References/ Readings	**Text books** 1. Kundu, Shakti. Modern Software Engineering Guidebook. BPB Publications, 2024. ISBN-10: 9355519801 2. Rumbaugh, James; Ivar Jacobson; and Grady Booch. The Unified Modeling Language Reference Manual. 2nd ed. Addison-Wesley (Pearson Education), 2004. ISBN-10: 0321245628. 3. Pressman, Roger S. Software Engineering: A Practitioner's Approach. 6th ed. McGraw-Hill, 2005. ISBN-10: 007301933X. **Reference books** 1. Mohapatra, Hitesh, & Rath, Amiya Kumar. Fundamentals of Software Engineering (2nd ed.). BPB Publications, 2025. ISBN-10: 9365893380 2. Jalote, Pankaj. Software Project Management in Practice. Addison-Wesley, 2002. ISBN-10: 0201737213. 3. Schach, Stephen R. Object-Oriented and Classical Software Engineering. 8th ed. McGraw-Hill, 2011. ISBN-10: 0073376183. 4. Rumbaugh, James, Michael Blaha, William Premerlani, Frederick Eddy, and William Lorensen. Object-Oriented Modelling and Design. Prentice Hall, 1991. ISBN-10: 0136298419. 5. Sommerville, Ian. Software Engineering. 10th ed. Pearson, 2016. ISBN-10: 1292096136	
Course Outcomes:	Upon completion of the course, students will be able to: **CO1:** Explain software engineering preliminaries and software development process models. **CO2:** Describe requirements engineering to software systems. **CO3:** Apply Software Design Techniques. **CO4:** Analyze Coding, Testing techniques and project metrics concepts.	

Name of the Programme : B.E in Computer Science and Engineering (Artificial Intelligence and Machine Learning)
Course Code : AIM-228
Title of the Course : Fundamentals of Software Engineering Lab
Number of Credits : 1
Effective From AY : 2025-26

Pre-requisites for the Course:	Basic knowledge of computer programming	
Course Objectives:	This course will enable students to: 1. Understand software engineering concepts to develop small to medium-scale software systems. 2. Apply software development process models and follow best practices for requirement analysis, design, coding, and testing. 3. Implement practical experience with software design, coding, testing, and project metrics. 4. Develop teamwork, documentation, and problem-solving skills in real software projects.	
Content:	**List of Programs/Experiments**	**No of Hours**
	1. Case study on Software Development Process Models. 2. Develop IEEE SRS document. 3. Design Level 0, Level 1 and Level 2 Dataflow Diagram. 4. Design Data Dictionary. 5. Design Use Case and Class Diagram. 6. Design Activity Diagram. 7. Design Sequence Diagram. 8. Design Interaction Diagram. 9. Design and develop Frontend, 10. Develop Backend and Integrate. 11. Case study on software testing strategies. 12. Mini Project	**30**
Pedagogy:	Integration of instructional learning, constructive thinking, inquiry-based, collaborative, experiential, and problem-solving approaches.	
References/ Readings:	**Text books** 1. Kundu, Shakti. Modern Software Engineering Guidebook. BPB Publications, 2024. ISBN-10: 9355519801	

	2. Rumbaugh, James; Ivar Jacobson; and Grady Booch. The Unified Modeling Language Reference Manual. 2nd ed. Addison-Wesley (Pearson Education), 2004. ISBN-10: 0321245628. 3. Pressman, Roger S. Software Engineering: A Practitioner's Approach. 6th ed. McGraw-Hill, 2005. ISBN-10: 007301933X. **Reference books** 1. Mohapatra, Hitesh, & Rath, Amiya Kumar. Fundamentals of Software Engineering (2nd ed.). BPB Publications, 2025. ISBN-10: 9365893380. 2. Jalote, Pankaj. Software Project Management in Practice. Addison-Wesley, 2002. ISBN-10: 0201737213. 3. Schach, Stephen R. Object-Oriented and Classical Software Engineering. 8th ed. McGraw-Hill, 2011. ISBN-10: 0073376183. 4. Rumbaugh, James, Michael Blaha, William Premerlani, Frederick Eddy, and William Lorensen. Object-Oriented Modelling and Design. Prentice Hall, 1991. ISBN-10: 0136298419. 5. Sommerville, Ian. Software Engineering. 10th ed. Pearson, 2016. ISBN-10: 1292096136

Course Outcomes:	Upon completion of the course, students will be able to: **CO1:** Describe the ability to analyze software requirements and produce a structured requirements specification. **CO2:** Design software solutions using appropriate design techniques (UML diagrams). **CO3:** Implement, test, and debug software programs following coding standards and best practices. **CO4:** Apply software project metrics and testing techniques to evaluate software quality.