VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

School of Computer Science Engineering and Information Systems

Master of Computer Applications

(M.C.A)

Curriculum

(*2025-2026 onwards*)

INDEX

School of Computer Science Engineering and Information Systems

Master of Computer Applications

Programme Credit Structure	Credits
University Core Courses	37
Open Elective Courses	03
Professional Core Courses	28
Professional Elective Courses	12
Total Graded Credit Requirement	**80**

University Core Courses

S.No.	Course Code	Course Title	L	T	P	C
1	PAENG501	Technical Report Writing	1	0	4	3
2	PAMCA697	Project Work	0	0	0	8
3	PAMCA698	Internship I/ Dissertation I	0	0	0	10
4	PAMCA699	Internship II/ Dissertation II	0	0	0	10
5	PASTS601	Competitive Coding I	3	0	0	3
6	PASTS602	Competitive Coding II	3	0	0	3
					Total Credits	**37**

Open Elective Course

S.No.	Course Code	Course Title	L	T	P	C
1	PASTS501	Qualitative and Quantitative Skills Practice I	3	0	0	3

Professional Core Courses

S.No.	Course Code	Course Title	L	T	P	C
1	PAMAT501	Probability and Statistics	3	1	0	4
2	PAMCA501	Data Structures and Algorithms	3	0	2	4
3	PAMCA502	Java Programming	3	0	2	4
4	PAMCA503	Database Management Systems	3	0	2	4
5	PAMCA504	Machine Learning	3	0	2	4
6	PAMCA505	Software Engineering	3	1	0	4
7	PAMCA506	Full Stack Web Development	3	0	2	4
					Total Credits	**28**

Professional Elective Courses

S.No.	Course Code	Course Title	L	T	P	C
1	PAMCA601	Cloud Computing	3	1	0	4
2	PAMCA602	Python for Data Science	3	0	2	4
3	PAMCA603	Mobile Application Design and Development	3	0	2	4
4	PAMCA604	Cybersecurity	3	1	0	4
5	PAMCA605	Deep Learning	3	0	2	4
6	PAMCA606	Generative AI	3	0	2	4
7	PAMCA607	Mining of Massive Dataset	3	0	2	4
8	PAMCA608	System Design	3	1	0	4

University Core Courses

Course Code	Course Title	L	T	P	C
PAENG501	Technical Report Writing	1	0	4	3
Pre-requisite	NIL	**Syllabus Version**			
		1.0			

Course Objectives

To develop communicative competence in students.
To apply advanced technical communication principles in practical writing tasks for diverse professional contexts.
To execute research and analytical skills to produce coherent and evidence-based technical documents through practical exercises.

Course Outcomes

Apply the principles of effective report writing.
Demonstrate the ability to draft and present technical reports.
Produce key components of technical documents, including sections focusing on purpose, audience, and structure, through writing assignments.
Create technical summaries and abstracts by practicing techniques for summarizing and paraphrasing complex information.

Module:1	**Fundamentals of Technical Writing**	**3 hours**

Introduction to Technical Writing: Definition and typical forms (reports, instructions, proposals); Key Factors in Technical Writing: Purpose, Audience, and Tone; General Writing Basics: Clarity, fluency, effectiveness; The Process of Writing: Pre-writing, Writing, and Post-writing stages; Organization in Writing: Having an outline, using introductions, headings, lists, figures, and summaries.

Module:2	**Technical Grammar**	**2 hours**

Concord; Tense Shifts

Module:3	**Introduction to Reports and Report Structure Front Matter**	**2 hours**

Reports in Organizations: Role and importance of reporting in corporate and industrial segments.; Purpose of Reports: Conveying decisions, facts, and information accurately and up-to-date.; Report Structure Overview: Division into Front Matter, Main Body, and Back Matter; Components of the Front Matter

Module:4	**Report Structure Main Body and Back Matter**	**4 hours**

Components of the Main Body: Introduction, Discussion or Description, Conclusions, and Recommendations; Writing Introduction, Discussion or Description, Conclusions and Recommendations, Methods of Reporting: The Letter Method and The Schematic Method; Routine Reports: Nature, frequency, and function in organizations (often statistical, fixed intervals).

Module:5	**Technical Proposals**	**2 hours**
Punctuation Right words and phrases; avoiding cliches, jargons, foreign words and phrases, ambiguity, redundancy, circumlocution - Developing hints		
Module:6	**Contemporary Issues**	**2 hours**
Guest Lecture		
	Total Lecture Hours:	**15 hours**
Text Book(s)		
Kumar. S & Pushplata. , "**Effective Communication Skills.** ", New Delhi OUP, **2018** Muralikrishna and Sunita Mishra, "**Communication Skills For Engineers**", Pearson, 2nd Edition, **2011** Shirley Mathew, "**Effective Communication Skills.**", Nirali Prakashan, **2025**		
Reference Books		
Indicative Experiments		
1. Introduction to the Technical Writing Introduction to Technical Report Writing**;** Analyzing and identifying the characteristics of effective technical writing in sample documents.; Short writing exercises focusing on clarity, conciseness, and identifying purpose, audience, and tone in simple technical scenarios. **Activity:** Reviewing and providing feedback on short technical descriptions.		2 hours
2. Analyzing Audience and Context for Practical Writing Applying reader and stakeholder analysis techniques to specific engineering report scenarios; Drafting content segments tailored for different technical and non-technical audiences (e.g., writing an executive summary for managers vs. a technical description for fellow engineers); Simulating audience needs assessment based on given project descriptions. **Activity**: Developing a 'reader profile' for a major report project.		4 hours
3. **Technical Grammar and Style in Practice** Intensive practical exercises on complex grammatical structures, sentence syntax, and common errors in technical writing; Exercises in applying principles of clarity, conciseness, and precision to improve technical sentences and paragraphs; Practicing the appropriate use of active and passive voice in different report sections. **Activity:** Peer-editing session focusing on grammar, mechanics of writing, and technical style.		2 hours

4. The Writing Process and Report Outlining Lab Practicing prewriting techniques for complex technical topics: brainstorming, mind mapping, and systematic outlining; Developing detailed hierarchical outlines for a major technical report project, including main and sub-points; Planning content organization based on report type (e.g., feasibility, empirical research, lab report) and audience needs. **Activity**: Group exercise to compare and refine report outlines.	4 hours
5. **Writing the Report Front Matter and Introduction** Detailed practical guidance on drafting all components of the report Front Matter; Creating a professional Title Page, Forwarding Letter/Preface, Acknowledgements, and Table of Contents for the ongoing report project; Drafting effective Abstracts and Summaries (Executive Summaries) based on provided technical content, focusing on capturing the essence. **Activity:** Writing the Introduction section of the report project, including background, scope, and objectives.	4 hours
6. Writing the Report Body: Data, Descriptions, and Discussion Techniques for presenting data, technical descriptions, and analysis in the main body; Drafting sections of the report body focusing on presenting organized data and technical details; Writing the "Discussion" section, focusing on interpreting results and explaining findings based on provided or self-generated data. **Activity:** Integrating data and analysis points into the draft of the report body.	4 hours
7. **Writing the Report Body: Conclusions and Recommendations** Practical methods for drawing logical conclusions and formulating actionable recommendations; Drafting the "Conclusions" section based on the data and discussion from Module 6; Developing clear and practical recommendations based on the conclusions, considering the report's purpose and audience. **Activity:** Writing the "Recommendations" section of the report project.	4 hours
8. **Transcribing Visuals** Using charts, graphs and tables; Transcribing visuals that are clear, accurate, and effectively support the report's text; Integrating created visuals into the report draft, ensuring proper placement, captions, and referencing within the text. **Activity:** Peer review focusing on the effectiveness and integration of visuals.	2 hours

9. **Report Back Matter, Condensation, and Final Review** Practical session on creating the Back Matter: Appendices, Bibliography, Glossary, and Index (if applicable). Emphasis on consistent citation and referencing styles; Compiling Appendices and formatting a Bibliography for the report project; Practicing techniques for précis writing and summarization to condense longer texts. **Activity:** Final review of the complete report draft, focusing on overall structure, flow, formatting, and coherence.	2 hours
10. **Presentation Skills** Short Presentations, Formal Presentation with PPT Analytical Presentation of Charts, Graphs and Tables **Activity**: Presentations – Individual and Group	2 hours
Total Laboratory Hours:	**60 hours**
Mode of Evaluation : Continuous Assessment Test, Digital Assignment, Quiz, Final Assessment Test, Lab Continuous Assessment, Lab Final Assessment, Presentation	
Recommended by Board of Studies :	16-05-2025
Approved by Academic Council : No. 78	12-06-2025

Course Code	Course Title	L	T	P	C
PAMCA698	Internship I/ Dissertation I	0	0	20	10
Pre-requisite	**NIL**	**Syllabus Version**			
		1.0			

Course Objectives :

1. To provide hands-on learning experience in the design and development of suitable products and processes relevant to the chosen field.
2. To enhance technical skill sets required for solving real-world problems in the domain of study.
3. To develop research orientation and analytical thinking abilities for addressing advanced challenges in the field.

Course Outcomes :

1. Formulate real-life problem statements with appropriate assumptions and constraints.
2. Perform comprehensive literature and patent searches to gather relevant information in the area of interest.
3. Develop and implement suitable solution methodologies, including experimentation, design, analysis, and iterative improvements.
4. Analyze results through error analysis, benchmarking, and cost estimation to validate the solution.
5. Synthesize findings to derive scientific conclusions and effectively communicate the outcomes through technical reports and presentations.

Module Content	**(Project Duration : One Semester)**

1. Dissertation may be a theoretical analysis, modeling & simulation, experimentation & analysis, prototype design, fabrication of new equipment, correlation and analysis of data, software development, applied research and any other related activities.
2. Dissertation should be individual work.
3. Carried out inside or outside the university, in any relevant industry or research institution.
4. Publications in the peer reviewed Journals / International Conferences will be an added advantage.

Mode of Evaluation: Assessment on the project – Dissertation report to be submitted, presentation, project reviews and Final Oral Viva Examination.

Recommended by Board of Studies			
Approved by Academic Council	**No.**	**Date**	

Course Code	Course Title	L	T	P	C
PAMCA699	Internship II/ Dissertation II	0	0	20	10
Pre-requisite	**NIL**	**Syllabus Version**			
		1.0			

Course Objectives :

1. To provide hands-on learning experience in the design and development of suitable products and processes relevant to the chosen field.
2. To enhance technical skill sets required for solving real-world problems in the domain of study.
3. To develop research orientation and analytical thinking abilities for addressing advanced challenges in the field.

Course Outcomes :

1. Formulate real-life problem statements with appropriate assumptions and constraints.
2. Perform comprehensive literature and patent searches to gather relevant information in the area of interest.
3. Develop and implement suitable solution methodologies, including experimentation, design, analysis, and iterative improvements.
4. Analyze results through error analysis, benchmarking, and cost estimation to validate the solution.
5. Synthesize findings to derive scientific conclusions and effectively communicate the outcomes through technical reports and presentations.

Module Content	**(Project Duration : One Semester)**

1. Dissertation may be a theoretical analysis, modeling & simulation, experimentation & analysis, prototype design, fabrication of new equipment, correlation and analysis of data, software development, applied research and any other related activities.
2. Dissertation should be individual work.
3. Carried out inside or outside the university, in any relevant industry or research institution.
4. Publications in the peer reviewed Journals / International Conferences will be an added advantage.

Mode of Evaluation: Assessment on the project – Dissertation report to be submitted, presentation, project reviews and Final Oral Viva Examination.

Recommended by Board of Studies			
Approved by Academic Council	**No.**	**Date**	

Course Code	Course Title	L	T	P	C
PASTS601	Competitive Coding I	3	0	0	3
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

Equip learners with an in-depth understanding of linear data structures (linked lists, stacks, queues, deques, etc.). Enabling them to design, implement, and optimize these structures for various problem-solving scenarios. Emphasis is placed on both theory (complexities, operations) and hands-on coding to master data structure manipulation and usage.

Course Outcomes

By the end of this course, participants will be able to:

- Confidently implement and utilize linked lists, stacks, queues, and deques
- Analyse and optimize the time/space complexities of linear data structure operations (insertion, deletion, traversal)
- Apply problem-solving strategies to real-world tasks and competitive coding challenges, leveraging linear data structures effectively
- Integrate debugging, testing, and best coding practices when working with linear data structures

Module:1	Linked List-1	6 hours

Introduction to Linked Lists: Definition, structure, advantages; Types of Linked Lists: Singly, Doubly, Circular.
Basic Operations: Creating a linked list; Insertion (at beginning, middle, end); Deletion (from beginning, middle, end); Traversal (iterative, recursive).

Module:2	Linked List-2	6 hours

Advanced Topics: Reversing a linked list (iterative, recursive); Detecting and removing loops; Merging and splitting linked lists; Intersection of two linked lists.

Module:3	Stack-1	5 hours

Introduction to Stack: Definition, LIFO property, real-world applications (e.g., undo functionality, browser history); Static vs dynamic stacks.
Operations on Stack: Push, pop, peek; Checking stack overflow and underflow.
Implementation: Using arrays; Using linked lists.

Module:4	Stack-2	4 hours

Applications of Stack: Balancing parentheses in expressions; Converting infix expressions to postfix; Evaluating postfix expressions; Solving backtracking problems (e.g., maze solving).

Module:5	Queue	7 hours

Introduction to Queue: Definition, FIFO property; Types of Queues: Simple, Circular, Double-Ended.
Operations on Queue: Enqueue (insertion), Dequeue (removal), Peek (front element); Checking queue overflow and underflow.
Implementation: Using arrays; Using linked lists.
Applications of Queue: Task scheduling (e.g., CPU scheduling, print queue); Breadth-First Search (BFS) implementation; Managing resources in real-time systems (e.g., job queues).

Module:6	Priority Queue	5 hours

Introduction to Priority Queue: Definition, importance in task prioritization; Differences between Priority Queue and Simple Queue.
Operations on Priority Queue: Enqueue and Dequeue operations based on priority.
Implementation: Using arrays; Using heaps (Binary Heaps: Min-Heap and Max-Heap).

Module:7	Hash map	9 hours
Introduction to Hash map: Definition, key-value pair structure, advantages over arrays. Hash Functions and Collision Handling: Direct addressing, simple hash functions, modulo operation; Collision resolution techniques: Chaining, Linear Probing, Quadratic Probing. Applications of Hash map: Frequency counting (e.g., counting occurrences of elements); Caching mechanisms (e.g., implementing LRU cache); Finding duplicates in arrays or strings; Solving anagram problems; Building indexes for fast lookup.		
Module:8	**Interview Prep - Networking**	**3 hours**
Networking Fundamentals: OSI model TCP/IP basics Network Protocols: HTTP/HTTPS DNS and DHCP FTP and SMTP - Practical Interview Questions: - Debugging network issues, Firewall, DNS, Ping and Trace route - Explaining client-server architecture - Designing scalable systems		
Total Lecture hours:		**45 hours**

Text Book	
1.	A Textbook of Data Structures and Algorithms 1: Mastering Linear Data Structures – 1st Edition by G A Vijayalakshmi Pai – Wiley Publication
Reference Books	
1.	Data Structure and Algorithms Made Easy by Narasimha Karumanchi -5th edition

Professional Core Courses

Course Code	Course Title	L	T	P	C
PAMAT501	Probability and Statistics	3	1	0	4
Pre-requisite		**Syllabus Version**			
		1.0			

Course Objectives

1. Apply probability and statistical concepts to solve real-world data analysis problems.
2. Analyze data distributions and relationships using appropriate statistical measures.
3. Evaluate hypothesis testing methods to support data-driven decision-making and predictive modeling.

Course Outcomes

1. Apply probability concepts and random variable distributions to model uncertainties in real-world data.
2. Select and apply suitable probability distributions for practical and experimental scenarios.
3. Interpret relationships among variables using correlation and regression analysis.
4. Perform statistical hypothesis tests using large sample techniques.
5. Use parametric and non-parametric methods for hypothesis testing in small or non-normal samples.

Module:1	**Foundations of Probability and Random Variables**	**12 hours**

Probability – The axioms of probability – Conditional probability – Multiplication rule – Theorem of total probability – Bayes theorem – Independence of events.
Random Variables – Discrete and continuous random variables – Probability mass, probability density and cumulative distribution functions – 2 Dimensional Random Variable – Joint distributions – Marginal and conditional distributions – Product moments – Covariance.
Moments – Moment generating functions – Characteristic function. (Concepts Only)

Module:2	**Probability Distributions**	**9 hours**

Discrete distributions – Binomial, Poisson, Geometric.
Continuous distributions – Uniform, Exponential, Gamma, Weibull, Beta, Normal distributions.

Module:3	**Correlation and Regression Analysis**	**7 hours**

Mathematical expectation – Correlation – Rank Correlation - linear regression – Partial correlation – Multiple correlation – Multiple linear regression.

Module:4	**Statistical Inference – Large Samples**	**7 hours**

Sampling distributions – Estimation of parameters – Statistical hypothesis testing.
Large sample tests based on Normal distribution:- Single mean - Difference of means - Single proportion - Difference of proportions - Difference of standard deviations

Module:5	**Statistical Inference – Small Samples and Non-parametric Tests**	**8 hours**

Small sample tests:Tests based on t, F distributions for mean, variance and proportion – Chi-square test – Contingency table – Goodness of fit – Analysis of Variance – 1 way (RBD) – 2 way (CRD), Latin SquareNon-parametric tests:- Sign test - Rank sum test - Run test - Kruskal-Wallis test - Mann-Whitney U test - Median test - Kolmogorov-Smirnov test - DarlingTests

Module:6	Contemporary Issues	2 hours
Guest Lecture from Industry and R & D organizations		
	Total Lecture Hours:	**45 hours**
Text Book(s)		
1. Richard A Johnson, "Probability and Statistics for Engineers", Pearson Education Ltd, Malaysia., 9th Edition, 2018 2. Gupta, S.P. and Kapoor, V.K., "Fundamentals of Mathematical Statistics", Sultan Chand & Sons, New Delhi., 12th Edition, 2020		
Reference Books		
1. Douglas C. Montgomery and George C. Runger, "**Applied Statistics and Probability for Engineers** ", John Wiley & Sons., 6th Edition, **2016** 2. Ronald E Walpole, Raymond H Myers, Sharaon L Myers, and Keying Ye, "**Probability & Statistics for Engineers and Scientists**", Prentice Hall, Delhi, 9th Edition, **2011** **3.** Robert V. Hogg, J.W. McKean, and Allen T. Craig, "**Introduction to Mathematical Statistics**", Pearson Education, Asia, 7th Edition, **2012**		
Mode of Evaluation: Continuous Assessment Test, Digital Assignment, Quiz, Final Assessment Test, Seminar		
Recommended by Board of Studies :		02-06-2025
Approved by Academic Council : No. 78		12-06-2025

Course Code	Course Title	L	T	P	C
PAMCA501	Data structures and Algorithms	3	0	2	4
Pre-requisite		**Syllabus Version**			
		1.0			

Course Objectives

1. Introduce foundational concepts in data structures and algorithm analysis.
2. Describe linear and non-linear data structures with practical applications.
3. Apply algorithm design strategies including divide and conquer, greedy, and dynamic programming.

Course Outcomes

1. Analyze time and space complexity of algorithms using asymptotic notation.
2. Compare searching, sorting, and nonlinear data structure techniques.
3. Design optimal solutions using graph-based data structures.
4. Apply greedy strategies to solve computational problems.
5. Apply dynamic programming strategies to solve computational problems.

Module:1	Algorithmic Analysis and Elementary Data Structures	9 Hours

The Role of Algorithms in Computing - Analyzing and Designing Algorithms - Characterizing Running Time - Asymptotic Notations - Formal Definition, Standard Notations and Common Functions - Solving Recurrence - Substitution Methods – Stack and its Applications - Queue – Different Types of queue - Linked List - Types – Applications of Linked List.

Module:2	Sorting Techniques and Trees	9 Hours

Sorting - Insertion sort, Shell sort, Merge sort, Radix sort, Heap sort - Searching -Linear Search, Binary Search - Hashing - Hash Functions, Closed Addressing - Open Addressing, Linear Probing, Quadratic Probing, Double Hashing, Rehashing - Preliminaries - Binary Trees - Expression Tree - Binary Search Tree - AVL Tree - B-Tree.

Module:3	Graph Algorithms	9 Hours

Representation of Graphs - Graph Traversals - Depth First search - Breadth First Search - Minimum Spanning Trees - The Algorithms of Kruskal and Prim - Single Source Shortest Path - Bellman-Ford Algorithm - Dijkstra‘s Algorithm

Module:4	Greedy Algorithms	7 Hours

Greedy Algorithms - Huffman Code - Bin Packing - Divide and Conquer - Quicksort - Strassen's Matrix Multiplication

Module:5	Dynamic Programming	9 hours

Backtracking - N-Queens Problem, Longest Common Subsequence, Dynamic Programming - Knapsack Problem - Ordering Matrix Multiplication - All Pairs Shortest Path Problem - Floyd-Warshall Algorithm

Module:6	Contemporary Issues	2 hours

Guest Lecture from Industry and R & D organizations

	Total Lecture hours:	45 hours

Text Books	
1	Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein, "Introduction to Algorithms", 2022, 4'" Edition, MIT Press, USA.
Reference Books	
1.	Mark Allen Weiss, “Data Structures and Algorithm Analysis in C++", 2019, 4" Edition, Pearson Education, Delhi.
2.	Michael T. Goodrich, Roberto Tamassia, and Michael H. Goldwasser, “Data structures and Algorithms in Python", 2013, John Wiley and Sons, Inc., United States of America.
Mode of Evaluation: CAT, Written assignment, Quiz, FAT, Project, Seminar, Group discussion	
List of Experiments (Indicative)	
1.	Implement stack functions using arrays
2.	Implementation of queues
3.	Implement queue functions using arrays
4.	Implementation of circular queue
5.	Reversing a queue
6.	Reverse a singly linked list and Merge two linked list
7.	Remove duplicate nodes from sorted linked list
8.	Program to find size of doubly linked list
9.	Rotate circular linked list by n nodes
10.	Find nth node from the end of circular linked list
11.	Determine whether the given two binary trees are identical or not
12.	Implement backtracking using depth first search
13.	Detecting a cycle in the graph
14.	Determine the height of a binary search tree
15.	Identify if the given binary search tree is valid or not.
16.	Implement kruskal's algorithm and prim's algorithm
17.	Implement floyd-warshall algorithm
18.	Implement knapsack problem
19.	Implementation of Quick sort and Merge sort
Total hours:	**30 hours**
Mode of Evaluation: Continuous assessment /FAT / Oral examination and others	
Recommended by Board of Studies	
Approved by Academic Council	**No.** **Date**

Course Code	Course Title	L	T	P	C
PAMCA502	Java Programming	3	0	2	4
Pre-requisite		**Syllabus Version**			
		1.0			

Course Objectives

1. Demonstrate core object-oriented principles and Java libraries.
2. Implement multithreading, exception handling, and collections.
3. Build GUI and web applications using JavaFX, JDBC, and Spring Boot.

Course Outcomes

1. Apply object-oriented principles to develop Java applications.
2. Develop multithreaded and exception-handling features in Java programs.
3. Build GUI-based database applications using JavaFX and JDBC.
4. Create web applications using Spring Boot and Hibernate.
5. Design microservice-based applications for distributed environments.

Module:1	**Java Fundamentals and OOP Concepts**	**9 hours**
Java Features - Data Types, Variables and Arrays - Operators - Control Statements - Classes and Methods - Inheritance - Packages and Interfaces		
Module:2	**Core Java Features and Advanced Programming**	**9 hours**
Exception Handling - String Handling - Wrapper Classes and Autoboxing - Multithreading - Files and Streams - Serialization - Annotations		
Module:3	**GUI, Database, and Networking**	**9 hours**
JavaFX - Event Handling - JDBC - Networking - Generics - Lambda Expressions - Collections - Stream API - Concurrency Utilities		
Module:4	**Enterprise Java and Spring Boot**	**8 hours**
MVC Architecture - Anatomy of Web Application - Servlets (API, Life Cycle) - GenericServlet - HttpServlet - ServletConfig - ServletContext - Session Management		
Module:5	**Persistence, Microservices, and Industry Trends**	**8 hours**
Spring Boot Basics - Core Features - Creating Web Application - Querying for Data with Spring Boot - Persistence - Hibernate - Integrating Hibernate with SpringBoot - Defining Microservices - Creating Cooperating Microservices - Deploying using Docker		
Module:6	**Contemporary Issues**	**2 hours**
Guest Lecture from Industry and R & D organizations		
	Total Lecture hours:	**45 hours**

Text Books

1	Herbert Schildt, Danny Coward, “Java: The Complete Reference”, 2023, 13th Edition, McGraw Hill Publications

Reference Books	
1.	Greg L. Turnquist, “Learning Spring Boot 3.0: Simplify the development of production-grade applications using Java and Spring”, 2022, 3rd Edition, Packt Publishing
2.	Catalin Tudose, “Java Persistence with Spring Data and Hibernate”, 2023, 2nd Edition, Manning Pubns Co
3.	Magnus Larsson, “Microservices with Spring Boot and Spring Cloud: Build Resilient and Scalable Microservices Using Spring Cloud, Istio, and Kubernetes”, 2021, 2nd Edition, Packt Publishing
Mode of Evaluation: CAT, Written assignment, Quiz, FAT, Project, Seminar, Group discussion	

List of Experiments (Indicative)		**Hours**
1	Basic programs	2 hours
2	Implementation of package, interface and abstract class	4 hours
3	Test any five of standard exceptions and user-defined custom exceptions in Java	2 hours
4	Threads creation and design applications by extending the thread class / implementing the Runnable interface. Application of multithreading in Java	2 hours
5	Using Collections and Streams for working with user defined objects	4 hours
6	Design GUI-based Java application using JavaFX controls and JDBC	4 hours
7	Design and implement networking applications	2 hours
8	Implementation of servlet programs – session management	4 hours
9	MVC driven Web Application Development using Springboot and Hibernate	4 hours
10	Develop, deploy and test Microservices using Springboot	2 hours
	Total hours:	**30 hours**

Mode of Evaluation: Continuous assessment /FAT / Oral examination and others			
Recommended by Board of Studies			
Approved by Academic Council	**No.**	**Date**	

Course Code	Course Title	L	T	P	C
PAMCA503	Database Management Systems	3	0	2	4
Pre-requisite		**Syllabus Version**			
		1.0			

Course Objectives

1. Design relational databases using data modeling and normalization techniques.
2. Apply relational theory and structured query language for data management.
3. Explore database architecture, indexing, transactions, and advanced models.

Course Outcomes

1. Design normalized relational schemas using ER and EER modeling.
2. Apply relational algebra and normalization techniques for schema optimization.
3. Develop complex queries using SQL and PL/SQL for data manipulation.
4. Implement transaction and concurrency control strategies.
5. Evaluate specialized models like NoSQL, distributed, temporal, and spatial databases.

Module:1	Fundamentals of DBMS and Data Modeling	9 hours

Introduction - Database Definition, Environment - Database Architecture - Data Models, Entity-Relationship Modeling, Enhanced Entity-Relationship Modeling, The Relational Model - Terminology, Integrity Constraints, Logical Database Design, ER and EER to Relational Mapping

Module:2	Relational Theory and Normalization	9 hours

Relational Algebra - Binary, Unary Operators, Aggregate and GROUP BY, Data Redundancy and Update Anomalies, Functional Dependencies, Process of Normalization - 1NF - 2NF - 3NF - Functional Dependencies - Inference Rules, Minimal sets of Functional Dependencies - BCNF - 4NF - 5NF, Lossless-Join Algorithms of Normal forms

Module:3	SQL, PL/SQL and Query Processing	9 hours

SQL and PL/SQL: Definition, Views, Data Manipulation and Transaction, Control Languages, Advanced SQL - Declarations, Control Statements, Exception, Cursor, Subprograms, Triggers and ADT, Query processing - Query Optimization, Heuristic-based Approach - Rule-based Optimization

Module:4	Physical Database Design and Transaction Management	8 hours

Indexing—Sparse and Dense, BTree Vs Hash, Tuning-Physical Layout and Query Rewriting, Properties of Transaction, Concurrency Control - The Need for Concurrency Control, Serializability and Decomposition, Locking and Timestamp Methods, Multiversion Techniques, Recovery - Need, Techniques

Module:5	Advanced Database Models	8 hours

Temporal Database Concepts - Spatial Database Concepts - Distributed Database Concepts - NoSQL Databases - Introduction to MongoDB - Distributed CRUD Operations and Sharding

Module:6	Contemporary Issues	2 hours
Guest Lecture from Industry and R & D organizations		
	Total Lecture hours:	**45 hours**

Text Books	
1	Ramez Elmasri and Shamkant B. Navathe, "Fundamentals of Database Systems", 2016, 7th Edition, Pearson Education, Delhi.
2	Thomas Connolly, Carolyn Begg, "Database Systems: A Practical Approach to Design, Implementation and Management", 2015, 6th Edition, Pearson Education.
Reference Books	
1.	Abraham Silberschatz, Henry F. Korth, S. Sudarshan, "Database System Concepts", 2020, 7th Edition, McGraw Hill.
2.	Raghu Ramakrishnan, Johannes Gehrke, "Database Management Systems", 2007, 3rd Edition, McGraw Hill.
3.	Shannon Bradshaw, Eoin Brazil, Kristina Chodorow, "MongoDB: The Definitive Guide: Powerful and Scalable Data Storage", 2019, 3rd Edition, O'Reilly.
Mode of Evaluation: CAT, Written assignment, Quiz, FAT, Project, Seminar, Group discussion	

List of Experiments (Indicative)		**Hours**
1	**Database Creation** Viewing all Databases - Creating a Database - Viewing all Tables in a Database - Creating Tables - Dropping / Truncating / Renaming Tables, Creating Views, Set the New Constraints to the Table - Drop the Constraints / Modify Constraints, etc.	6 Hours
2	**Database Manipulation** Inserting / Updating / Deleting Records in a Table - Using Transaction Control Commands - Commit, Rollback and Save point	2 Hours
3	**ET Operators and Built-in Functions** Union, Intersection, Minus, and Queries involving Date Functions - String Functions and Math Functions	2 Hours
4	**Complex Queries (Nested and Join Queries)** Join Queries - Inner Join, Outer Join - Subqueries - With IN clause - With EXISTS clause	4 Hours
5	**PL/SQL Programs** Variables, Constants, Loops, Conditional Statements, Cursor, Procedure, and Functions	8 Hours
6	**No SQL Databases** Mongo DB - Create, CRUD operation.	2 Hours
7	**Design and develop business applications using SQL, PL/SQL and No SQL.**	6 Hours
	Total hours:	**30 hours**

Mode of Evaluation: Continuous assessment /FAT / Oral examination and others			
Recommended by Board of Studies			
Approved by Academic Council	**No.**	**Date**	

Course Code	Course Title	L	T	P	C
PAMCA504	Machine Learning	3	0	2	4
Pre-requisite		**Syllabus Version**			
		1.0			

Course Objectives

1. To gain knowledge of ML systems and algorithms.
2. To design and train models using various learning paradigms.
3. To develop skills in ensemble, reinforcement, and neural learning.

Course Outcomes

1. Demonstrate core principles of machine learning algorithms.
2. Classify data using supervised and unsupervised algorithms.
3. Integrate ensemble and reinforcement learning techniques.
4. Reduce data dimensionality using statistical techniques.
5. Train neural networks to model learning-based representations.

Module:1	**Introduction**	**8 Hours**

Machine Learning and its Applications — Learning Problems — Designing a Learning Systems — Perspectives and Issues in Machine Learning - Version Spaces -— Finite and Infinite Hypothesis Spaces — PAC Learning - Learning a Class from Examples - VC Dimension — Noise - Learning Multiple Classes

Module:2	**Supervised and Unsupervised Learning Algorithms**	**10 Hours**

Regression: Linear Regression, Multiple Linear Regression, Logistic Regression — Bayes Classification – Decision Trees – Support Vector Machines – K-Nearest Neighbors –Introduction - K-Means Clustering - Expectation Maximization Algorithm - Supervised Learning after Clustering - Hierarchical Clustering - Density Based Clustering - Evaluation Metrics - Association Rule Learning.

Module:3	**Combining Multiple Learners and Reinforcement Learning**	**9 Hours**

Generating Diverse Learners - Model Combination Schemes - Voting - Error Correcting Output Codes - Bagging - Boosting - The Mixture of Experts - Stacking - Random Forest Classifier - Single State Case - K-Armed Bandit - Elements of Reinforcement Learning - Model Based Learning - Temporal Difference Learning - Generalization - Partially Observable States

Module:4	**Dimensionality Reduction**	**7 hours**

Principal Component Analysis - Feature Embedding - Factor Analysis - Canonical Correlation Analysis - Linear Discriminant Analysis -

Module:5	**Neural Networks**	**9 hours**

Introduction to Neural Networks — Perceptron — Multilayer Perceptron - Back propagation algorithm – Training Procedures – Improving Convergence – Loss Functions and Optimization - Deep Learning – Convolution Neural Networks – Recurrent Neural Networks.

Module:6	**Contemporary Issues**	**2 hours**

Guest Lecture from Industry and R & D organizations

	Total Lecture hours:	**45 hours**

Text Books	
1	Ethem Alpaydin, “Introduction to Machine Learning", 2020, 4" Edition, MIT press.
Reference Books	
1.	Mitchell, Tom M., “Machine Learning", 2007, Vol. 1, McGraw-Hill, New York.
2.	Marsland, Stephen, “Machine Learning: an Algorithmic Perspective", 2015, 2nd Edition, Chapman and Hall/CRC.
3.	Mohri, Mehryar, AfshinRostamizadeh, and Ameet Talwalkar, “Foundations of Machine Learning", 2018, 2nd Edition, MIT press.
4.	Doane, David P., and Lori E. Seward, “Applied Statistics in Business and Economics", 2016, 5th Edition, Mcgraw-Hill.
Mode of Evaluation: CAT, Written assignment, Quiz, FAT, Project, Seminar, Group discussion	

List of Experiments (Indicative)		**Hours**
1.	Simple Linear Regression — Multiple Linear Regression	4 Hours
2.	Decision Tree Classification Algorithm - Entropy - Gini Index	2 Hours
3.	Naive Bayes Classification - Maximum Likelihood	2 Hours
4.	Classification and Regression Trees — Regression Trees	4 Hours
5.	Support Vector Machines - Linear Kernel Functions — Non Linear Kernal Functions and Kernel Functions	4 Hours
6.	K-Nearest neighbor Classification Algorithm	4 Hours
7.	Bagging , Boosting and Random Forest Classification	4 Hours
8.	K-Means Clustering	2 Hours
9.	Hierarchical — Agglomerative - Divisive Clustering	2 Hours
10.	K-Armed Bandit - Model Based Learning	2 Hours
	Total hours:	**30 hours**

Mode of Evaluation: Continuous assessment /FAT / Oral examination and others			
Recommended by Board of Studies			
Approved by Academic Council	**No.**	**Date**	

Course Code	Course Title	L	T	P	C
PAMCA505	Software Engineering	3	1	0	4
Pre-requisite	**NIL**	**Syllabus Version**			
		1.0			

Course Objectives

1. Compare traditional and agile software development models.
2. Document software requirements and design using UML tools.
3. Demonstrate principles of software quality, testing, and DevOps deployment.

Course Outcomes

1. **Compare** traditional and agile software development processes.
2. **Document** software requirements using UML and design principles.
3. **Apply** quality assurance practices to meet software standards.
4. **Construct** effective software testing strategies.
5. **Deploy** software using DevOps tools and CI/CD pipelines.

Module:1	Fundamentals of Software Engineering and Agile Development	9 hours

Nature of Software, Software Engineering – Need, Importance and its Characteristics · Software Processes – Generic Process Model · Prescriptive Processes · Model Specialized, Unified Process, Classical Evolutionary models, Personal and Team Process Model · Software Project Metrics, Agile Process · Agile Principles · Adaptive Software Development · Extreme Programming · Scrum · Dynamic Systems Development Method · Crystal · Feature Driven Development · Lean Software Development · Agile Modelling · Agile Unified Process

Module:2	Requirements and Object-Oriented Design Engineering	12 hours

Requirements Engineering, Establishing the Groundwork, Requirements Gathering, Developing Use Cases, Building the Analysis Model, Negotiating Requirements, Validating Requirements · Requirements Monitoring, Validating Requirements, Design Concepts · Architectural Design · Object Oriented Design using UML · Interactions · Use Case Diagrams · Interaction Diagrams · Activity Diagrams · State Machines · Processes and Threads · Time and Space · State Chart Diagrams · Components · Deployment Diagram

Module:3	Software Quality Assurance and Management	8 hours

Software Quality · Quality Factors · The Software Quality Dilemma · Achieving Software Quality, Reviews - Criteria for Types of Reviews - Informal Reviews - Formal Technical Reviews, Software Quality Assurance · Elements of Software Quality Assurance · SQA Processes and Product Characteristics · SQA Tasks, Goals, and Metrics · Formal Approaches to SQA · Statistical Software Quality Assurance · Software Reliability · The ISO 9000 Quality Standards · The SQA Plan

Module:4	Software Testing and Validation	7 hours

Strategic Approach To Software Testing, Planning and Recordkeeping, Test-Case Design, White Box Testing, Black-Box Testing, Object-Oriented Testing, Integration Testing, Artificial Intelligence and Regression Testing, Integration Testing in the OO Context, Validation Requirements, Testing Patterns

Module:5	**DevOps and Software Deployment**	**7 hours**
DevOps: Motivation · Cloud as A Platform · Operations · Deployment Pipeline · Continuous Integration and Testing · Deployment · Crosscutting Concerns: Monitoring · Logging · Security Audits-Other IIeII's · Business Considerations – Case study – Migrating to Micro services		
Module:6	**Contemporary Issues**	**2 hours**
Guest Lecture from Industry and R & D organizations		
	Total Lecture hours:	**45 hours**

Text Books	
1	Roger S. Pressman and Bruce Maxim, "Software Engineering – A Practitioner's Approach", 2023, 9e Edition, McGraw Hill Higher Education.
Reference Books	
1.	Ian Sommerville, "Software Engineering", 2017, 10th Edition, Addison-Wesley.
2.	Grady Booch, James Rumbaugh, and Ivar Jacobson, "The Unified Modeling Language User Guide", 2019, Addison-Wesley.
3.	Len Bass, Ingo Weber, and Liming Zhu, "DevOps: A Software Architect's Perspective", 2017, Pearson Education.

Mode of Evaluation: CAT, Written assignment, Quiz, FAT, Project, Seminar, Group discussion			
Recommended by Board of Studies			
Approved by Academic Council	**No.**	**Date**	

Course Code	Course Title	L	T	P	C
PAMCA506	Full Stack Web Development	3	0	2	4
Pre-requisite	**Nil**	**Syllabus Version**			
		1.0			

Course Objectives

1. Build responsive UIs using HTML, CSS, and JavaScript.
2. Develop front-end applications using React and Angular frameworks.
3. Integrate backend services using Node.js, Express, and MongoDB.

Course Outcomes

1. Design responsive web pages using HTML, CSS, and JavaScript.
2. Develop frontend interfaces using ReactJS.
3. Create backend APIs using Node.js and Express.
4. Connect MongoDB with web applications for persistent storage.
5. Build full-stack apps using MERN/MEAN stack and deploy them.

Module:1	**HTML, CSS and Client-side Scripting**	**11 hours**

Introduction to HTML5 - Headings, Linking, Images, Lists, Tables, Forms, New HTML5 Form Input Types, Page Structure Elements - Introduction to CSS - Inline Styles, Embedded Style Sheets, Linking External Style Sheets, Backgrounds, Element Dimensions, Box Model and Text Flow, Positioning, Media Types and Media Queries, Drop Down Menus Selectors, Text Shadows, Box Shadows- JavaScript - Introduction, Operators - Control Statements - Functions, Dialog boxes - Arrays - Objects - Document Object Model - JavaScript Event Handling, Form Processing - Ajax - JSON

Module:2	**React Framework**	**10 hours**

Environment Set up – JSX – React DOM – Props – State – Controlled Components – Custom Hooks – Reusable React Component – Conditional Rendering – Advanced and Impossible States – Fetching and Re-fetching – Async/Await – Styling in React- React Routers- Server Rendering - Authentication and Authorization - Deployment – Mongoose and Redux libraries

Module:3	**Web Server - Node JS**	**5 hours**

Getting Started with Node.js Installing Node.js, Working with Node Packages, Creating a Node.js Application - Using Events, Listeners, Timers, and Callbacks in Node.js - Implementing Event emitter, Implementing Callbacks - Accessing the File System from Node.js - Implementing HTTP Services in Node.js - Saving Time with Express - The Request and Response Objects - Form Handling - Sending Client Data to Server, Form Handling with Express - Cookies and Sessions – Middleware Function Calls

Module:4	**MongoDB**	**5 hours**

NoSQL Database - MongoDB Basics Documents, Collections, Database, Query Language, Installation, The Mongo Shell - MongoDB CRUD Operations - MongoDB Node.js Driver using Mongoose

Module:5	**Front End framework - Angular**	**12 hours**

Reactive Development Paradigm – Advanced Angular Architecture – Setting up your Development Environment – Creating basic Angular App – Developing high quality UX with Angular Material – Forms Observables and Subjects

Module:6	Contemporary Issues	2 hours
Guest Lecture from Industry and R & D organizations		
	Total Lecture hours:	**45 hours**

Text Books	
1	Paul Deitel, Harvey Deitel and Abbey Deitel, "Internet and World Wide Web: How To Program", 2018, 5th Edition, Pearson.
2	Vasan Subramanian, Pro MERN "Stack: Full Stack Web App Development with Mongo, Express, React, and Node", 2019, 2nd Edition, Apress.
Reference Books	
1.	Brad Dayley, Brendan Dayley, Caleb Dayley, "Node.js, MongoDB, and Angular Web Development", 2017, 2nd Edition, Addison Wesley, Oreilly.
2.	Ethan Brown, "Web Development with Node and Express", 2019, 2nd Edition, O'Reilly Media Inc..
3.	Shyam Seshadri, Brad Green, "AngularJS: Up and Running: Enhanced Productivity with Structured Web Apps", 2014, 1st Edition, O'Reilly Media Inc..

Mode of Evaluation: CAT, Written assignment, Quiz, FAT, Project, Seminar, Group discussion

List of Experiments (Indicative)		**Hours**
1.	Design interactive web pages using HTML5	2
2.	Design responsive web pages using CSS	2
3.	Implement Client-Side Validations and Dynamic Object Manipulations with Events using Javascript	4
4.	Implement Asynchronous Client, server, and data storage interactions using AJAX and JSON.	4
5.	Develop component-based User Interface using REACT JS	2
6.	Develop Single Page Applications using Angular framework.	2
7.	Design web applications with dynamic routing using Node JS, and Express framework	2
8.	Develop a three-tier web application model and data manipulations using Node Js, Express, and Mongo DB.	4
9.	Develop a three-tier web application File upload and Email.	4
10.	Develop web applications with Session and Cookies.	4
	Total hours:	**30 hours**

Mode of Evaluation: Continuous assessment /FAT / Oral examination and others

Recommended by Board of Studies			
Approved by Academic Council	**No.**	**Date**	

Professional Elective Courses

Course Code	Course Title	L	T	P	C
PAMCA601	Cloud Computing	3	1	0	4
Pre-requisite		**Syllabus Version**			
		1.0			

Course Objectives

1. Trace the evolution of cloud computing and virtualization technologies.
2. Explore cloud architectures, service models, and deployment platforms.
3. Examine security frameworks and design trusted cloud environments.

Course Outcomes

1. Analyze the convergence of computing paradigms and virtualization in cloud environments.
2. Design scalable and resilient cloud infrastructures with modern techniques.
3. Implement cloud-native applications using DevOps and container orchestration.
4. Apply cloud security principles to protect cloud resources.
5. Evaluate the integration of edge computing and IoT with cloud services.

Module:1	Cloud Computing Evolution and Virtualization	8 hours

Overview of Computing Paradigm- Convergence of technologies -Evolution of cloud computing- Benefits and Challenges- NIST Cloud Computing Reference Architecture- Cloud Deployment Model- Cloud Service Model- Virtualization -Types of Virtualizations -Hypervisors- -Virtual Machines- VM Migration- Virtualization of CPU, Memory, I/O Devices- Network Virtualization- Storage Virtualization- Desktop Virtualization.

Module:2	Cloud Computing Infrastructure Setup and Management	10 hours

Resource Pooling, Sharing and Provisioning- Data Center - Standardization, Automation and Optimization-Multi-tenancy- Types- Resource Provisioning Approaches-Zero Downtime Architecture - Scaling in Cloud Computing- Scaling Strategies- - Capacity Planning - Load Balancing- Categories of Load Balancing -Content Delivery Network - Cloud Computing Services- Amazon Webservices, Microsoft Azure, Google Cloud Platform-Opensource Support for Cloud.

Module:3	Cloud Computing Programming Paradigms	9 hours

MapReduce Paradigm & HDFS- Microservices- Serverless Computing and Event Processing – DevOps – DevOps lifecycle- Infrastructure as Code - Provisioning with Terraform-Version control and Git - Continuous Integration -Automated Testing in DevOps - Containers and Orchestration- Deployment with Docker and Kubernetes

Module:4	Cloud Computing Security	8hours

Threat, Vulnerability, and Risk — Network Level Security—Host Level Security- Application-Level Security- Information Security- Identity Management and Access Control - Zero Trust Security Model- Cloud Security Design Principles- Cloud Security Management Frameworks— Security-As-A-Service— Privacy in Cloud Environment.

Module:5	**Edge Computing and IoT**	**8 hours**
Edge Computing & Fog Hierarchy - Mobile Computing - Smart Objects- Unique Identification of Things- IoT Architecture and Communication Protocols - Enabling Technologies for the Internet of Things- Cloud Services for IoT- Case Study- Innovative Applications of IoT.		
Module:6	**Contemporary Issues**	**2 hours**
Guest Lecture from Industry and R & D organizations		
	Total Lecture hours:	**45 hours**

Text Books	
1	Douglas E. Comer, "The Cloud Computing Book: The Future of Computing Explained", 2021, 1st Edition, CRC Press, Florida.
2	Sandeep Bhowmik, "Cloud Computing", 2017, 1st Edition, Cambridge University Press,UK
Reference Books	
1.	Ekambar Kumar Singirikonda, "DevOps Automation Cookbook",2024,1st Edition, BPB Publications, India
2.	Rajkamal," Internet of Things -Architecture and Design Principles",2022, 2nd Edition, McGraw Hill Education,US
3.	K.Chandrasekaran, " Essentials of Cloud Computing", 2015, 1st Edition, CRC Press,Florida.

Mode of Evaluation: CAT, Written assignment, Quiz, FAT, Project, Seminar, Group discussion			
Recommended by Board of Studies			
Approved by Academic Council	**No.**	**Date**	

Course Code	Course Title	L	T	P	C
PAMCA602	Python for Data Science	3	0	2	4
Pre-requisite		**Syllabus Version**			
		1.0			

Course Objectives

1. Develop Python programs using OOP, functions, and file operations.
2. Analyze real-world data using statistical and probabilistic techniques.
3. Utilize Python libraries for data preprocessing, visualization, and modeling.

Course Outcomes

1. Develop Python programs using OOP, data structures, and file handling.
2. Apply mathematical and statistical techniques to real-world data.
3. Transform and clean datasets for analysis.
4. Visualize data using Python libraries like Matplotlib and Seaborn.
5. Solve practical problems using machine learning tools and datasets.

Module:1	Python Programming Basics and File Handling	10 hours

Introduction to Python: Features, Environment Setup, Data Types and Variables, Control Flow Structures, Python Data Structures: Lists, Tuples, Sets, Dictionaries. list/dict comprehensions. Functions: Defining and Calling Functions, Function Arguments: Positional, Keyword, Default, Return Values, Lambda Functions, Recursive Functions, Function Decorators, Modules and Packages: Creating and Importing Custom Modules, Built-in Modules: math, os, sys, random, etc. Virtual Environments, Basics of Object-Oriented Programming: Classes, Objects, Methods, File Handling: Reading/Writing Files, File Paths, Exception Handling: try, except, else, finally, multiple exceptions.

Module:2	Mathematical Foundations for Data Science	7 hours

Introduction to Data Science & Problem Typology, Linear Algebra Essentials: Vectors and Matrices, Matrix Operations and Factorizations, Descriptive Statistics: Mean, Median, Mode, Variance, Standard Deviation, Structured and Unstructured Data. Optimization Methods for Data Science: Basics of Optimization, Convex vs. Non-Convex Functions, Gradient Descent, Cost Functions and Loss Minimization.

Module:3	Probability and Statistical Inference	8 hours

Basics of Probability: Conditional Probability, Bayes Theorem, Distributions: Normal, Binomial, Poisson, Statistical Inference: Sampling, Hypothesis Testing, Outlier Detection and Handling, Understanding and Plotting Distributions, Univariate, Bivariate, Multivariate Statistical Analysis.

Module:4	EDA and Python Libraries	9 hours

Data Acquisition and Preprocessing, Handling Missing Values and Outliers, Feature Engineering and Dimensionality Reduction, Data Quality and Cleaning Techniques, Exploratory Data Analysis (EDA): Concepts and Steps, NumPy: Arrays, Operations, Pandas: Series, DataFrames, Indexing, Grouping, Merging.

Module:5	Data Visualization and Introduction to Data Science Tools	9 hours
Introduction to Data Visualization, Matplotlib and Seaborn: Line, Bar, Histogram, Heatmap, Pairplot, Plotly for Interactive Visualization, Introduction to Open Source Tools: R, Scilab, Octave, Weka, scikit-learn Basics: Data Splitting, Model Training, Prediction.		
Module:6	**Contemporary Issues**	**2 hours**
Guest Lecture from Industry and R & D organizations		
	Total Lecture hours:	**45 hours**

Text Books

1	Gowrishankar S and Veena A, "Introduction to Python Programming", 2019, 1st Edition, CRC Press, Taylor & Francis Group, Boca Raton, FL.
2	R. V. Hogg, J. W. McKean and A. Craig, "Introduction to Mathematical Statistics", 2019, 8th Edition, Pearson Education, India.

Reference Books

1	Ani Adhikari and John DeNero, "Computational and Inferential Thinking: The Foundations of Data Science", 2019, GitBook.
2	Carl Shan, Henry Wang, William Chen, Max Song. "The Data Science Handbook: Advice and Insight from 25 Amazing Data Scientists." 2016, The Data Science Bookshelf.

Mode of Evaluation: CAT, Written assignment, Quiz, FAT, Project, Seminar, Group discussion

List of Experiments (Indicative)		Hours
1	Solve problems using operators, control structures.	2
2	Work with Python data structures (Lists, Tuples, Dictionaries, Sets).	4
3	Implement user-defined functions, OOP, exception handling, and file operations.	6
4	Explore and download datasets from UCI, Kaggle.	2
5	Handle missing data and perform outlier detection.	2
6	Apply normalization techniques and compute descriptive statistics.	2
7	Create data visualizations using Matplotlib and Seaborn.	2
8	Analyze tweets or fake news spread using text-based data.	2
9	Perform EDA on a real-world dataset.	2
10	Build and evaluate a simple Linear Regression model using scikit-learn.	2
11	Implement a recommendation system based on transactional data.	2
12	Behavioral segmentation using clustering or statistical profiling.	2
	Total hours:	**30 hours**

Mode of Evaluation: Continuous assessment /FAT / Oral examination and others

Recommended by Board of Studies			
Approved by Academic Council	**No.**	**Date**	

Course Code	Course Title	L	T	P	C
PAMCA602	Mobile Application Design and Development	3	0	2	4
Pre-requisite		**Syllabus Version**			
		1.0			

Course Objectives

1. Design mobile UIs using Flutter and Dart programming.
2. Implement state management, navigation, and data handling techniques.
3. Deploy optimized and tested mobile applications across platforms.

Course Outcomes

1. Build cross-platform apps using Flutter and Dart.
2. Design responsive user interfaces using Flutter widgets.
3. Apply state management and navigation techniques.
4. Integrate APIs, cloud, and AI/ML features into mobile apps.
5. Deploy apps to Android and iOS platforms with testing and optimization.

Module:1	**Introduction to Mobile App Development**	**8 hours**

Overview of mobile platforms (Android, iOS, hybrid) - Introduction to Flutter & Dart: Features, advantages, limitations - Flutter architecture, installation, and setup - First Flutter app: Structure and workflow

Module:2	**Dart Programming Essentials**	**8 hours**

Dart syntax, variables, data types - Control structures: loops, conditionals - Functions, classes, OOP concepts in Dart - Error handling and debugging

Module:3	**Flutter UI Development**	**9 hours**

Mobile UI Principles – UI Widgets: Stateless vs Stateful - Layouts: Single-child, multi-child, custom layouts - Theming and styling - User input, forms, and validation

Module:4	**State Management & Navigation**	**9 hours**

State management approaches (setState, Provider, Riverpod, Bloc) - Navigation and routing - Data persistence: local storage (SQLite, SharedPreferences), file I/O - Integrating REST APIs and JSON parsing

Module:5	**Emerging Trends & App Deployment**	**9 hours**

Integrating AI/ML (using TensorFlow Lite, ML Kit) - CI/CD pipelines (GitHub Actions for Flutter) - Cloud integration (Firebase, AWS Amplify) - Testing, debugging, and performance optimization - App deployment: Play Store & App Store

Module:6	**Contemporary Issues**	**2 hours**

Guest Lecture from Industry and R & D organizations

	Total Lecture hours:	**45 hours**

Text Books

1	Eric Windmill, "Flutter in Action", Manning Publications, 2020.
2	Rap Payne, "Beginning App Development with Flutter", Apress, 2025

Reference Books	
1.	Simone Alessandria, Flutter Cookbook: 100+ Step-by-Step Recipes for Building Cross-Platform, Professional-Grade Apps with Flutter 3.10.x and Dart 3.x, Packt Publishing. 2023
2.	Alberto Miola, "Flutter Complete Reference 2.0", Independently Published, 2023
Mode of Evaluation: CAT, Written assignment, Quiz, FAT, Project, Seminar, Group discussion	

List of Experiments (Indicative)		**Hours**
1.	Setting up Flutter and Dart environment; running a sample app.	2
2.	Dart programming basics: variables, functions, OOP.	2
3.	Building UI with basic widgets: Text, Image, Button.	2
4.	Designing layouts: Row, Column, Stack, ListView, GridView.	2
5.	Implementing forms and input validation.	2
6.	State management using setState and Provider.	3
7.	Navigation and routing between screens.	3
8.	Local data storage using SQLite and SharedPreferences.	3
9.	Consuming REST APIs and displaying dynamic data.	3
10.	Integrating Firebase for authentication and real-time database.	3
11.	App testing, debugging, and performance profiling with Mockito.	3
12.	Packaging and deploying the app to Android and iOS devices.	2
	Total hours:	**30 hours**

Mode of Evaluation: Continuous assessment /FAT / Oral examination and others			
Recommended by Board of Studies			
Approved by Academic Council	**No.**	**Date**	

Course Code	Course Title	L	T	P	C
PAMCA604	Cyber Security	3	1	0	4
Pre-requisite	**Nil**	**Syllabus version**			
		1.0			

Course Objectives:

1. Identify cyber threats, attacks, and system vulnerabilities.
2. Apply security tools and techniques for system and data protection.
3. Interpret cybersecurity laws, governance models, and ethical considerations.

Course Outcomes:

1. Identify cyberattacks, vulnerabilities, and their classifications.
2. Mitigate threats using countermeasures and security practices.
3. Apply standards and cyber laws to enhance system security.
4. Investigate identity theft and phishing using practical tools.
5. Formulate organizational cybersecurity policies and strategies.

Module:1	**Introduction to Cyber and Cyber Offenses**	**8 hours**

Definition and Scope - Risks - Threats - Classifications and Categories of Cybercrimes - The Legal Perspectives Indian Perspective - Global Perspectives - overview of Indian Cyber Laws- CIA Triad - Attack Vectors and Cyber Offenses - Social Engineering, Cyber Stalking, Cyber Cafe and Cybercrimes - Bot Nets

Module:2	**Cybercrime - Mobile and Wireless Devices**	**8 hours**

Trends in Mobility - Credit Card Frauds in Mobile and Wireless Computing Era - Security Challenges Posed by Mobile Devices - Authentication Service Security - Attacks on Mobile Phone - Organizational Measures and Security Policies - Identity and Access Management - Architecture - IAM Standards

Module:3	**Cybercrime Techniques and Tools**	**9 hours**

Introduction - Password Cracking - Keyloggers and Spywares - Viruses and Worms - Trojan Horses and Backdoors - Steganography - DoS and DDoS Attacks - SQL Injection - Buffer Overflow - Tools - John the Ripper, Hashcat, Refog, Spyrix, Zeus Trojan, Conficker Worm, OpenPuff, Steghide, SQLMap, Aircrack-ng, Kismet - Attacks on Wireless Networks - Web Threats and End Point Security - Organizational Guidelines and Incident Handling

Module:4	**Phishing and Identity Theft**	**9 hours**

Phishing - Methods and Techniques - Spear Phishing - Types of Phishing - Phishing Toolkit - Spy Phishing -Toolkits and Phishing Countermeasures - Identity Theft - Personal Identifiable Information - Types and Techniques – Countermeasures - Case Study - Identify Theft - Social Media Security and Privacy Challenges - Protecting People's Privacy

Module:5	**Cybercrimes -** Legal, Ethical, and Strategic Dimensions	**9 hours**

The Legal Perspectives - Need of Cyberlaw - the Indian IT Act - Amendments to the Indian IT Act - Digital Signature and Cybercrime Punishment - Ethical and Psychological Aspects - Hacker Mindset and Skills of Hackers and Other Cybercriminals - Sociology of Cybercriminals - Ethics in Cyberspace - Cyber Terrorism and Information Warfare - Intellectual Property in Cyberspace (IPR Issues) - Disaster Recovery and Threat Mitigation - Case Study on IPR Crime in India

Module 6	Contemporary Issues	2 Hours
Guest Lecture from Industry and R & D organizations		
	Total Lecture hours:	**45 hours**

Text Books	
1.	**Cybersecurity and Cyberwar: What Everyone Needs to Know,** P.W. Singer & Allan Friedman, Oxford University Press, 2024
2.	Cybersecurity and Cybercrime: An Introduction (2nd Edition), Thomas J. Holt, Adam M. Bossler, Kathryn C. Seigfried-Spellar, Routledge, 2023 (Latest Edition).

Reference Books	
1.	**Cyber Law: The Indian Perspective",** Pavan Duggal, LexisNexis India, 5th Edition (2023)
2.	Yuri Diogenes, Erdal Ozkaya, "Cybersecurity - Attack and Defense Strategies", 2018, 2nd Edition, Packt Publishers.
3	Nina Godbole, Sunit Belapure, "Cyber Security - Understanding Cybercrimes, Computer Forensics and Legal Perspectives", 2018, 1st Edition, Wiley.

Mode of Evaluation: CAT, Written assignment, Quiz, FAT, Project, Seminar, Group discussion			
Recommended by Board of Studies			
Approved by Academic Council	**No.**	**Date**	

Course Code	Course Title	L	T	P	C
PAMCA605	Deep Learning	3	0	2	4
Pre-requisite	**Nil**	**Syllabus version**			
		v.1.0			

Course Objectives:

1. Understand fundamental deep learning concepts.
2. Compare deep learning architectures such as CNNs and RNNs.
3. Apply transfer learning, attention mechanisms, and transformers to real-world tasks.

Course Outcomes:

1. Apply deep neural network concepts to practical problems.
2. Differentiate among CNN architectures for vision tasks.
3. Implement transfer learning using pre-trained models.
4. Integrate transformers and attention mechanisms into NLP and vision tasks.
5. Evaluate deep learning model performance and suggest improvements.

Module:1	**Introduction to neural networks and deep neural networks**	**9 hours**

Neural Networks Basics - - Activation functions- Loss function - Hyperparameters – Gradient Descent - Variants of GD: (Batch GD, Stochastic GD, Mini-batch Gradient Descent)- Back Propagation- AdaDelta, AdaGrad, RMSProp and Adam Optimization - Hyperparameter tuning - Batch Normalization – Regularization - Under-fitting Vs Over-fitting

Module:2	**Convolutional Neural Network (CNN)**	**9 hours**

Fundamentals of Convolutional Neural Networks - Image Augmentation – CNN operations – Architecture Padding, Strides- Pooling - Fully Connected Layers - Simple Convolution Network – Deep Convolutional Models – AlexNet, VGGNet, NiN, GoogleNet, ResNet, DenseNet

Module:3	**Region based CNN & Generative Adversarial Networks (GAN)**	**8 hours**

Region based CNN – Fast RCNN - You Only Look Once – Single Shot Detector , Encoder - Decoder architecture - Generative Adversarial Networks (GANs): Vanilla GAN, DC-GAN

Module:4	**Recurrent networks**	**9 hours**

Introduction to Recurrent Neural Networks - Text Augmentation – Components of RNN – Working principle of RNN - Bidirectional RNNs, Sequence-to-Sequence Architectures- Long-Term Dependencies - Echo State Networks - Long Short-Term Memory (LSTM) and Other Gated RNNs

Module:5	**Advanced Deep Learning Models**	**8 hours**

Queries, Keys, and Values - Attention Pooling by Similarity - Attention Scoring Functions - Attention Mechanisms, Multi head Attention - Self-Attention and Positional Encoding, Auto encoders - Transformers - Bidirectional Encoder Representations from Transformers (BERT), Generative Pre-trained Transformers, GPT and T5

Module:6	**Contemporary Issues**	**2 hours**

Guest Lecture from Industry and R & D Organizations

	Total Lecture hours:	**45 hours**

Text Books	
1	Ian Goodfellow, Yoshua Bengio and Aaron Courville, “Deep learning”, 2016, MIT Press.
2	Zhang, A., Lipton, Z. C., Li, M., & Smola, A. J. (2023). Dive into deep learning. Cambridge University Press.
Reference Books	
1.	Dipanjan Sarkar, Raghav Bali, Tamoghna Ghosh, “Hands-On Transfer Learning with Python”, 2018, First edition, Packt Publishing
2.	John D. Kelleher, “Deep Learning”, 2019, First edition, The MIT Press
3.	Charu C. Aggarwal, “Neural Networks and Deep Learning: A Textbook”, 2018, First edition, Springer
Mode of Evaluation: CAT, Written assignment, Quiz, FAT, Project, Seminar, Group discussion	

List of Experiments (Indicative)		**Hours**
1.	Implement a basic neural network with forward and backward propagation	3
2.	Compare different optimization techniques (SGD, RMSProp, Adam) and apply regularization (L1, L2, Dropout).	3
3.	Design and train a CNN model on the MNIST or CIFAR-10 dataset	3
4.	Use models like VGG16, ResNet for classification on a custom dataset using transfer learning	3
5.	Implement object detection using YOLOv5 or SSD and evaluate performance	3
6.	Implement various GAN to generate synthetic images	3
7.	Build a simple RNN model for text sentiment classification	3
8.	Implement a sequence-to-sequence model for translation tasks	3
9.	Develop deep learning models to classify movie reviews from the IMDB dataset using LSTM and Bi-directional LSTM	3
10.	Build a Transformer model for text classification using the IMDB sentiment analysis dataset	3
	Total hours:	**30 hours**

Mode of Evaluation: Continuous assessment /FAT / Oral examination and others			
Recommended by Board of Studies			
Approved by Academic Council	**No.**	**Date**	

Course Code	Course Title	L	T	P	C
PAMCA606	Generative AI	3	0	2	4
Pre-requisite	**NIL**	**Syllabus Version**			
		1.0			

Course Objectives

1. Describe foundational concepts and evolution of generative AI.
2. Implement generative models using autoencoders, GANs, and transformers.
3. Evaluate generative AI applications with ethical considerations.

Course Outcomes

1. Summarize the evolution and impact of Generative AI.
2. Construct generative models using VAEs, GANs, and diffusion techniques.
3. Implement transformer architectures and tokenization.
4. Fine-tune generative models using RLHF and evaluation metrics.
5. Develop generative applications across text, code, and multimodal domains.

Module:1	**Introduction to Generative AI**	**5 hours**

Definition and scope of Generative AI - Historical evolution and milestones - Comparison with traditional AI techniques - Applications across domains – art, text, audio, video, code - Ethical considerations – bias, copyright, misinformation

Module:2	**Generative Models – Foundations**	**10 hours**

Probabilistic modeling and latent variables - Autoencoders – structure and reconstruction - Variational Autoencoders (VAEs) – sampling and generation - Generative Adversarial Networks (GANs) – architecture and training - Diffusion models – basics and applications

Module:3	**Transformers and Large Language Models (LLMs)**	**10 hours**

Introduction to Transformers – self-attention, positional encoding - Pretraining techniques – masked and causal language modelling - Overview of popular LLMs – GPT, BERT, T5, LLaMA - Tokenization and embeddings – WordPiece, BPE - Scaling laws and model architecture

Module:4	**Training, Fine-Tuning, and Evaluation**	**10 hours**

Pretraining vs. fine-tuning strategies - Supervised fine-tuning, prompt tuning, and adapter layers - Reinforcement Learning with Human Feedback (RLHF) - Evaluation metrics – BLEU, ROUGE, perplexity, human evaluation - Dataset curation and data-centric AI

Module:5	**Applications and Deployment of Generative AI**	**8 hours**

Text generation – story writing, summarization, chatbots - Code generation – GitHub Copilot, Code LLMs - Multimodal generation – text-to-image (DALL·E, Stable Diffusion), text-to-speech - Deployment frameworks – Hugging Face, LangChain, ONNX - Responsible AI and future trends

Module:6	**Contemporary Issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Books	
1	Generative Deep Learning: Teaching Machines to Paint, Write, Compose, and Play by David Foster (Second Edition, 2023)
2	Tunstall, L., Von Werra, L., & Wolf, T. (2022). Natural language processing with transformers. " O'Reilly Media.
3	Heaton, J. (2018). Ian goodfellow, yoshua bengio, and aaron courville: Deep learning: The mit press, 2016, 800 pp, isbn: 0262035618. Genetic programming and evolvable machines, 19(1), 305-307.
Reference Books	
1.	TECKENTRUP, A. L. (2023). Transformers for Natural Language Pro-cessing. By Denis Rothman. SIAM Review Vol. 65, Issue 1 (March 2023), 65(1), 327.- Kamath, U., Keenan, K., Somers, G., & Sorenson, S. (2024).
2.	Large Language Models: A Deep Dive.Alammar, J., & Grootendorst, M. (2024). Hands-on large language models: language understanding and generation. " O'Reilly Media, Inc.".
Mode of Evaluation: CAT, Written assignment, Quiz, FAT, Project, Seminar, Group discussion	

Course Code	Course Title	L	T	P	C
PAMCA607	Mining of Massive Datasets	3	0	2	4
Pre-requisite		**Syllabus Version**			
		1.0			

Course Objectives

1. Preprocess large-scale datasets using scalable frameworks.
2. Apply distributed, streaming, and frequent pattern mining techniques.
3. Perform graph analytics and implement ML at scale using Spark and Hadoop.

Course Outcomes

1. Preprocess large datasets for mining and analysis.
2. Build scalable models using classifiers and clustering algorithms.
3. Discover frequent patterns and near-duplicate items in massive datasets.
4. Develop distributed data pipelines using Hadoop and Spark.
5. Analyze graphs using link analysis and community detection algorithms.

Module:1	Big-Data Fundamentals & MapReduce	6 hours

Data-mining applications at scale, Hadoop ecosystem overview: HDFS architecture, MapReduce programming model, Writing MapReduce jobs, Job optimizations: combiners, custom partitioners, chaining.

Module:2	Mining Data Streams	9 hours

Stream-processing challenges & one-pass constraints, Sampling techniques: reservoir sampling, Approximate counting: Bloom filters, Flajolet-Martin algorithm, Window models: sliding, tumbling, decaying windows, DGIM algorithm for count-distinct in decaying windows.

Module:3	Locality-Sensitive Hashing & Frequent Itemset	9 hours

Similarity measures: Jaccard, cosine; document shingling & MinHash, LSH banding technique for near-duplicate detection, Market-basket Analysis & Apriori, FP-Growth, Scalable, limited-pass algorithms for massive datasets, Streaming frequent-itemset algorithms.

Module:4	Link Analysis and Mining of Social Networks	9 hours

Graph representations: adjacency lists, property graphs, PageRank & Topic-Sensitive PageRank, handling link spam, Community detection: Label Propagation, modularity maximization, k-core decomposition, Clustering coefficients & motif counting, Graph-processing frameworks: Spark GraphX, Pregel model.

Module:5	Large Scale Machine Learning	10 hours

Distributed ML on Spark MLlib, Classification: logistic regression, linear SVM, decision trees, random forests, Regression: linear and tree-based models at scale, Clustering: K-Means, BFR, CURE, streaming clustering, Spectral clustering: Laplacian matrices & Eigen-decomposition, Model evaluation & hyperparameter tuning in distributed settings.

Module:6	Contemporary Issues	2 hours

Guest Lecture from Industry and R & D organizations

	Total Lecture hours:	45 hours

Text Books	
1	Jure Leskovec, Anand Rajaraman, Jeffrey Ullman, "Mining of Massive Datasets", Standford Press, 3rd Edition 2020
2	Streaming Data - Understanding the real-time pipeline, Andrew G. Psaltis ,May 2017
Reference Books	
1.	Jiawei Han, Micheline Kamber, Jian Pei **, Data Mining: Concepts and Techniques,** Elsevier/Morgan Kaufmann publishers, 4th Edition, 2022
2.	Tyler Akidau, Slava Chernyak, and Reuven Lax "Streaming Systems: The What, Where, When, and How of Large-Scale Data Processing" O'Reilly Media, 2018
Mode of Evaluation: CAT, Written assignment, Quiz, FAT, Project, Seminar, Group discussion	

List of Experiments (Indicative)		**Hours**
1.	Set up a pseudo-distributed Hadoop cluster; upload, replicate, and manage files in HDFS.	4
2.	Write a MapReduce job for word-count and inverted index construction.	4
3.	Write a MapReduce job to inner-join two datasets by respondent ID.	2
4.	Load text into Spark RDDs; apply map, filter, flatMap, and collect summary stats.	2
5.	Use Spark SQL/DataFrames for ETL: ingest, select/filter/aggregate, and write back.	4
6.	Implement Flajolet-Martin and DGIM in Spark Streaming for distinct- and window-counts.	2
7.	Implement Bloom filters and Count-Min Sketch for approximate membership and frequency.	2
8.	Shingle documents; apply MinHash and banded LSH to detect near-duplicates.	2
9.	Run Apriori and FP-Growth on transaction data with Spark MLlib.	2
10.	Compute PageRank and detect communities (Label Propagation) in Spark GraphX.	2
11.	Train and evaluate logistic regression and decision-tree models in Spark MLlib.	2
12.	Execute K-Means, BFR, CURE, and spectral clustering in Spark MLlib.	2
	Total hours:	**30 hours**

Mode of Evaluation: Continuous assessment /FAT / Oral examination and others			
Recommended by Board of Studies			
Approved by Academic Council	**No.**	**Date**	

Course Code	Course Title	L	T	P	C
PAMCA608	System Design	3	1	0	4
Pre-requisite		**Syllabus Version**			
		1.0			

Course Objectives

1. To understand the foundations of system design and distributed data management
2. To build scalable, secure and resilient application services
3. To apply system design principles to real-time and large-scale architectures

Course Outcomes

1. Recognize the key principles of system design with focus on non-functional requirements.
2. Apply concepts of scalable and distributed architectures to build efficient application services.
3. Analyze secure and resilient system design strategies for access control, recovery and threat mitigation.
4. Design solutions for large scale system challenges.
5. Develop real-time architecture for modern applications.

Module:1	Foundations of System Design	9 hours

Data Systems Architecture - Nonfunctional Requirement - Reliability and Fault Tolerance – Scalability – Maintainability – Replication- Sharding– Issues with Distributed Systems

Module:2	Building Scalable Application Services	8 hours

Scalable Systems - Distributed Systems Architectures - Concurrent Systems - Application Services - Distributed Caching - Asynchronous Messaging - Serverless Processing Systems – Microservices - Scalable Event Driven Processing - Stream Processing Systems

Module:3	Secure and Resilient System	8 hours

Secure System Design – Intersection of Security and Reliability – Adversaries and Threat Models – Design for Least Privilege and Secure Access - Zero Trust Architecture - Designing for a Changing Landscape - Design for Resilience - Design for Recovery - Mitigating Denial-of-Service Attacks.

Module:4	Large-Scale System Design	9 hours

Rate Limiter - Unique ID generator - URL Shortener - Web Crawler – Notification System – Chat System – News Feed System - Search Autocomplete - Multimedia Sharing - Cloud Storage

Module:5	Real-Time Service Architectures	9 hours

Proximity Service, Nearby Users, Map/Geolocation - Distributed Message Queue - Event Aggregation - Metrics Monitoring - Object Storage - Distributed Email - Gaming/Leaderboard – Payment System - Digital Wallet - Stock Exchange

Module:6	Contemporary Issues	2 hours
	Total Lecture hours:	**45 hours**

Text Books

1	Martin Kleppmann, Chris Riccomini, Designing Data-Intensive applications: The Big Ideas Behind Reliable, Scalable, and Maintainable systems, Second Edition, O'Reilly Media Inc, 2026. (Early release)
2	Ian Gorton, Foundations of Scalable Systems: Designing Distributed Architectures, O'Reilly, 2024.

Reference Books

1.	Heather Adkins, Betsy Beyer, Paul Blankinship, Piotr Lewandowski, Ana Oprea, Adam Stubblefield, Building Secure and Reliable Systems, O'Reilly Media, Inc., 2020.
2.	Alex Xu, System Design Interview - An Insider's Guide, Second Edition, Independently Published, 2020.
3.	Alex Xu, Sahn Lam, System Design Interview - An Insider's Guide" Volume 2, Byte Code LLC, 2022.
4.	Brendan Burns, Designing Distributed Systems, O'Reilly, 2022.
5.	Ford, Neal, Mark Richards, Pramod Sadalage, and Zhamak Dehghani Software Architecture: The Hard Parts. O'Reilly Media, 2021.
6.	Geewax, John J, API Design Patterns. Manning Publications Co., 2021.

Mode of Evaluation: Quiz, Design Project, Case Study, Seminar, CAT and FAT			
Recommended by Board of Studies			
Approved by Academic Council	**No.**	**Date**	