MADANAPALLE INSTITUTE OF TECHNOLOGY & SCIENCE

MADANAPALLE

(UGC-AUTONOMOUS)

www.mits.ac.in

DEPARTMENT OF COMPUTER APPLICATIONS

Course Structure

And

Detailed Syllabus (R24)

For the students admitted to

Master of Computer Applications from the academic year 2024-25 batch onwards

Vision and Mission of the Institution

Vision

To become a globally recognized research and academic institution and thereby contribute to technological and socio - economic development of the nation.

Mission

To foster a culture of excellence in research, innovation, entrepreneurship, rational thinking and civility by providing necessary resources for generation, dissemination and utilization of knowledge and in the process create an ambience for practice-based learning to the youth for success in their careers.

Vision and Mission of the Department

Vision

To be the source of producing competent computer application professionals in academic and research activities to serve the industry and society.

Mission

M1: To empower students with knowledge of computer applications through state-of-art infrastructure and curriculum.

M2: To groom students to become competent professionals in emerging technologies with industry specific programs.

M3: To inculcate ethical values, leadership and managerial skills in the students.

PROGRAMME OUTCOMES

The Programme Outcomes (POs) outline the key skills, knowledge, and competencies students must acquire upon completing the Master of Computer Applications (MCA). These outcomes aim to prepare students for successful careers in the rapidly evolving field of computer science and information technology.

PO1 (Foundation Knowledge):

Apply knowledge of mathematics, programming logic and coding fundamentals for solution architecture and problem solving.

PO2 (Problem Analysis):

Identify, review, formulate and analyse problems for primarily focussing on customer requirements using critical thinking frameworks.

PO3 (Development of Solutions):

Design, develop and investigate problems with as an innovative approach for solutions incorporating ESG/SDG goals.

PO4 (Modern Tool Usage):

Select, adapt and apply modern computational tools such as development of algorithms with an understanding of the limitations including human biases.

PO5 (Individual and Teamwork):

Function and communicate effectively as an individual or a team leader in diverse and multidisciplinary groups. Use methodologies such as agile.

PO6 (Project Management and Finance):

Use the principles of project management such as scheduling, work breakdown structure and be conversant with the principles of Finance for profitable project management.

PO7 (Ethics):

Commit to professional ethics in managing software projects with financial aspects. Learn to use new technologies for cyber security and insulate customers from malware

PO8 (Life-long learning):

Change management skills and the ability to learn, keep up with contemporary technologies and ways of working.

PROGRAM EDUCATIONAL OBJECTIVES

The MCA graduates will be able to:

PEO1: Excel in the software industry with the application of comprehensive knowledge and skills.

PEO2: Contribute by building innovative and sustainable solutions to the problems in the IT industry.

PEO3: Become successful professionals, exhibiting social responsibility leading to lifelong learning.

Dept. of Computer Applications

R24 - Course Structure

MCA I Year I Semester

S.No	Course Code	Course Title	L	T	P	C
1	24MATP101	Mathematical Foundations for Computer Applications	3	1	0	4
2	24MCAP101	Database Management Systems	3	1	0	4
3	24MCAP102	Data Structures and Algorithms	3	1	0	4
4	24MCAP103	Python Programming	3	1	0	4
5	24MCAP104	Operating Systems	3	1	0	4
6	24MCAP201	Database Management Systems Laboratory	0	0	3	1.5
7	24MCAP202	Data Structures and Algorithms Laboratory (Using C)	0	0	3	1.5
8	24MCAP203	Python Programming Laboratory	0	0	3	1.5
9		Skill Enhancement Course-I (Refer ANNEXURE – III)	2	0	2	3
		Total	**17**	**5**	**11**	**27.5**

MCA I Year II Semester

S.No	Course Code	Course Title	L	T	P	C
1	24MCAP105	Java Programming	3	1	0	4
2	24MCAP106	Computer Networks and Security	3	1	0	4
3	24MCAP107	Artificial Intelligence	3	1	0	4
4	24MCAP108	Software Engineering	3	1	0	4
5		Discipline Elective – I *(Refer ANNEXURE – II)*	4	0	0	4
6	24MCAP204	Java Programming Laboratory	0	0	3	1.5
7	24MCAP205	Computer Networks and Security Laboratory	0	0	3	1.5
8	24MCAP206	Artificial Intelligence Laboratory	0	0	3	1.5
9		Skill Enhancement Course-II *(Refer ANNEXURE – III)*	2	0	2	3
10		Audit Course *(Refer ANNEXURE – IV)*	2	0	0	0
		Total	**20**	**4**	**11**	**27.5**

MCA II Year I Semester

S. No	Course Code	Course Title	L	T	P	C
1	24MCAP109	Full Stack Web Development	3	1	0	4
2	24MCAP110	Data Science	3	1	0	4
3		**Discipline Elective – II** *(Refer ANNEXURE – II)*	4	0	0	4
4		**Discipline Elective – III** *(Refer ANNEXURE – II)*	4	0	0	4
5		**Open Elective – I** *(Refer ANNEXURE – I)*	3	0	0	3
6	24MCAP207	Full Stack Web Development Laboratory	0	0	3	1.5
7	24MCAP208	Data Science using Python Laboratory	0	0	3	1.5
8		**Skill Enhancement Course-III** *(Refer ANNEXURE – III)*	2	0	2	3
9	24MCAP701	Mini Project	0	0	4	2
10	24MCAP702	Technical Seminar	0	0	4	2
		Total	**19**	**2**	**16**	**29**

MCA II Year II Semester

S.No	Course Code	Course Title	L	T	P	C
1		**Discipline Elective – IV** *(Refer ANNEXURE – II)*	3	0	0	3
2		**Open Elective – II** *(Refer ANNEXURE – I)*	3	0	0	3
3	24MCAP703	Comprehensive Viva	0	0	4	2
4	24MCAP704	Major Project/Internship	0	0	20	10
		Total	**6**	**0**	**24**	**18**

ANNEXURE – I

LIST OF OPEN ELECTIVES

Open Elective – I

(To be offered under MOOC's Category from SWAYAM – NPTEL)

S.No	Course Code	Course Title	Course Offered by the Department of
1	24MBAP3M01	Organizational Behaviour	Management Studies
2	24MBAP3M02	Entrepreneurship	Management Studies
3	24MBAP3M03	Management Accounting	Management Studies
4	24MBAP3M04	Managerial Skills for Interpersonal Dynamics	Management Studies
5	24MBAP3M05	Innovation, Business Models and Entrepreneurship	Management Studies
6	24MBAP3M06	Management Information System	Management Studies

Open Elective – II

S.No	Course Code	Course Title	L	T	P	C
Courses offered in Conventional Mode						
1	24HUMP301	Indian Knowledge System	3	0	0	3
2	24ECEP301	Community Radio Technology	3	0	0	3
3	24MEP301	Automation and Robotics	3	0	0	3
4	24MBAP301	E-Commerce and Digital Markets	3	0	0	3
Course offered under MOOC's Category from SWAYAM – NPTEL						
4	24MBAP3M07	Ethics in Engineering Practice	3	0	0	3
5	24MBAP3M08	E-Business	3	0	0	3
6	24MBAP3M09	AI in Human Resource Management	3	0	0	3
Any new Interdisciplinary Course offered by SWAYAM NPTEL can be appended in future.						

ANNEXURE – II

LIST OF DISCIPLINE ELECTIVES

Discipline Elective – I

S.No	Course Code	Course Title	L	T	P	C
1	24MCAP401	Machine Learning	4	0	0	4
2	24MCAP408	Cyber Security	4	0	0	4
3	24MCAP409	Cloud Computing	4	0	0	4

Discipline Elective – II

S.No	Course Code	Course Title	L	T	P	C
1	24MCAP402	Deep Learning	4	0	0	4
2	24MCAP403	Digital Forensics	4	0	0	4
3	24MCAP404	Edge Computing	4	0	0	4

Discipline Elective – III

S.No	Course Code	Course Title	L	T	P	C
1	24MCAP405	Generative AI	4	0	0	4
2	24MCAP406	Penetration Testing	4	0	0	4
3	24MCAP407	Fog Computing	4	0	0	4

Discipline Elective – IV

S.No	Course Code	Course Title	L	T	P	C
1	24MCAP410	Agile Methodologies	3	0	0	3
2	24MCAP411	Software Project Management	3	0	0	3
3	24MCAP412	DevOps and Microservices	3	0	0	3
4	24MCAP413	Object Oriented Analysis and Design	3	0	0	3
Course offered under MOOC's Category from SWAYAM – NPTEL						
5	24MCAP4M01	Data Analytics with Python	3	0	0	3
Any other new Disciplinary Course which doesn't exist in the Curriculum can be appended in future.						

ANNEXURE – III

LIST OF SKILL ENHANCEMENT COURSES

Skill Enhancement Course-I						
S.No	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1	24ENGP601	Corporate Communication	2	0	2	3
Skill Enhancement Course-II						
S.No	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1	24MCAP601	Frontend Web Development	2	0	2	3
Skill Enhancement Course-III						
1	24MCAP602	Mobile Application Development	2	0	2	3
Any Skill- Enhancement courses can be appended in future.						

ANNEXURE – IV

List of Audit Courses

S.No	Course Code	Course Title	L	T	P	C
1	24MCAP901	Research Methodology and IPR	2	0	0	0

MCA I Year I Semester

Dept. of Computer Applications

MCA I Year I Semester

24MATP101 MATHEMATICAL FOUNDATIONS FOR COMPUTER APPLICATIONS

L	T	P	C
3	1	0	4

Pre-requisite Basic Calculus, Set theory, Relations, Functions, Basics of counting

Course Description:

This course provides a comprehensive understanding of the mathematical concepts and techniques essential for computer applications. It covers a range of topics, including Probability, random variables, probability distributions, propositional logic, modular arithmetic, graph theory and recurrence relations. The course is designed to equip students with the foundational mathematical reasoning skills necessary for advanced studies and system applications in computer science.

Course Objectives:

This course enables students to

1. To extend and formalize the knowledge of the theory of probability and Random Variables.
2. To solve the real time problems by using probability distributions.
3. Develop the skills to formulate the formal logical expressions and construct its validity.
4. Understand Operational and Counting Principles with suitable illustrations.
5. Explore Graph Theoretical concepts with their representing systems and terminologies.

UNIT I PROBABILITY 12 hours

Probability- probability axioms, addition law and multiplicative law of probability, conditional probability, Baye's theorem. Random variables- Discrete and Continuous random variables, probability density functions, properties, mathematical expectations, Joint densities and independence, moment generating function

UNIT II PROBABILITY DISTRIBUTIONS 12 hours

Discrete Probability Distributions – Geometric, Binomial, Poisson distributions and their properties
Continuous Probability Distributions – Gamma, Exponential, Normal distributions and their properties

UNIT III THE FOUNDATION OF MATHEMATICAL REASONING 12 hours

Basic terminologies and logic connectives, Propositional logic, Translation between English and propositional logic, Logical equivalence, Tautologies, Rules of inference for propositional logic, Normal forms.

UNIT IV THE FOUNDATIONS OF OPERATIONAL AND COUNTING PRINCIPLES **12 hours**

Introduction to modular arithmetic, The Euler phi function, Binary Relations, Boolean matrices, Properties of binary relations, Equivalence relations, Order relations, the Hasse diagram, Special elements in a poset, lattices, Properties of Lattices, Basic counting principles, The pigeonhole principle.

UNIT V THE FOUNDATIONS OF GRAPH THEORY **12 hours**

Basic definitions and terminology, Matrix Representation of Graphs, Subgraphs and new graphs from old, Walks, trails, paths, cycles and graph connectivity, Isomorphism of simple graphs, Hamiltonian circuits and Hamiltonian paths, Graph coloring, Trees, Traversal of trees, Spanning subtrees, Minimal spanning trees.

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Understand the concepts of probability and random variables and analyze the real time problems.
CO2: Solve various real time problems through probability distributions.
CO3: Demonstrate proficiency in translating between natural language statements and formal logical expressions through the concepts of inference.
CO4: Utilize modular arithmetic in various computational contexts and apply counting principles to solve many combinatorial problems.
CO5: Construct and apply graph theory models to implement algorithms for traversal and minimal spanning trees.

Text Book(s)

1. Milton. J. S. and Arnold. J.C., "Introduction to Probability and Statistics", Tata McGraw Hill, 4th Edition, 2007.
2. Joseph Khoury, Tale of Discrete Mathematics, A: A Journey Through Logic, Reasoning, Structures and Graph Theory, 1st Edition, 2024, World Scientific Publishing Co Pvt. Ltd.

Reference Books

1. S. C Gupta and V.K Kapoor, Fundamentals of Mathematical Statistics, 11th edition 2010 Sultan Chand & Sons, New Delhi.
2. DR. D.S.C, Mathematical Foundation of Computer Science, 3rd Edition, 2006, Prism Books Pvt Ltd.
3. Gerard O'Regan, Guide to Discrete Mathematics: an accessible introduction to the history, theory, logic and applications, 2nd edition, 2021, Springer.
4. Kevin Ferland, Discrete Mathematics and Applications, 2nd Edition, 2017, CRC Press Taylor & Francis Group.

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

Dept. of Computer Applications

MCA I Year I Semester

24MCAP101 DATABASE MANAGEMENT SYSTEMS

L	T	P	C
3	1	0	4

Pre-requisite None

Course Description:

This course is designed to investigate how database management system techniques are used to design, develop, implement and maintain database applications in organizations.

Course Objectives:

This course enables students to

1. Explain data base concepts and its applications, data base system structure and ER data modeling.
2. Illustrate the relational model, DDL, DML and Relational Algebra.
3. Demonstrate Schema refinement and Normal Forms.
4. Explain transactions and its management, concurrency control schemes and recovery.
5. Familiar with basic database storage structures, indexing methods including B tree and hashing.

UNIT I INTRODUCTION 12 hours

Introduction to data bases, Database system Vs File system, Data abstraction, Instances and Schemas, Database users, Database system structure, Entities, Attributes, Entity sets, Relationships and Relationship sets, Database design and ER diagrams, Specialization and Generalization and participation features of ER model.

UNIT II RELATIONAL MODEL AND QUERY EVALUATION 12 hours

Relational Model Concepts – Relational Algebra – SQL – Basic Queries DDL, DML, DCL, TCL– Complex SQL Queries – Views – Constraints.

UNIT III DATABASE DESIGN 12 hours

Functional Dependencies – Properties of relational Decomposition – First, Second, Third Normal Forms, Dependency Preservation – Boyce/Codd Normal Form – Multi-valued Dependencies and Fourth Normal Form – Join Dependencies and Fifth Normal Form.

UNIT IV TRANSACTION PROCESSING 12 hours

Query Processing-Transaction Processing – Properties of Transactions - Serializability – Transaction support in SQL - Locking Techniques –Validation Techniques — Recovery concepts – Shadow paging – Log Based Recovery – Introduction to PL/SQL & NOSQL

UNIT V FILES AND INDEXING **12 hours**

File operations – Hashing Techniques – Indexing – Single level and Multi-level Indexes – B+ tree – Static Hashing. Case study – Indexed Sequential Access Method (ISAM)

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Understand the basic concepts and the applications of database systems.
CO2: Implement the relational model and construct queries using SQL.
CO3: Apply the relational database design principles.
CO4: Illustrate the concepts of transaction processing, concurrency control and recovery
CO5: Apply the knowledge of database storage structures and indexing methods.

Text Book(s)

1. Abraham Silberschatz, Henry F.Korth and S.Sundarshan "Database System Concepts", Seventh Edition, McGraw Hill, 2019.

Reference Books

1. Ramez Elmasri and Shamkant Navathe, "Fundamentals of Database Systems", Seventh Edition, Pearson Education Delhi, 2017.
2. Raghu Ramakrishnan, ―Database Management Systems‖, Fourth Edition, McGrawHill College Publications, 2015.
3. Lee Chao, "Database Development and Management", Auerbach Publications, 1st edition, 2010.

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

Dept. of Computer Applications

MCA I Year I Semester

24MCAP102 DATA STRUCTURES AND ALGORITHMS

L	T	P	C
3	1	0	4

Pre-requisite None

Course Description:

The course is intended to provide the foundations and practical implementation of Data Structures and algorithms. The primary objective is to ensure that the student evolves into a competent programmer capable of designing, analyzing and implementing the algorithms and data structures for different applications. Also, the student will be equipped with algorithm analysis techniques, the theory of reductions, and the classification of problems into complexity classes like NP.

Course Objectives:

This course enables students to

1. Understand the primary data structure and its applications
2. Identify advanced data structures in problem solving.
3. Analyze various algorithm complexity.
4. Illustrate various sorting and graph algorithms.
5. Apply specific problem-solving techniques in the appropriate application.

UNIT I LINEAR DATA STRUCTURES 12 hours

Need of good data structures and algorithms, Arrays, Stacks, Evaluation of Expressions, Infix to Postfix Conversion, - Queues, Priority Queue, Linked lists, Singly Linked List, Circularly Linked List, Doubly Linked lists

UNIT II NON-LINEAR DATA STRUCTURES 12 hours

Trees: Binary Tree, Binary Search Tree, Tree Traversal, AVL Trees, , Splay Tree, Red-Black Trees, B-trees, B+ Tree, Binary Heap, Skip List, Search and Update Operations on Skip Lists, Graphs: Terminology and Representations – Traversals.

UNIT III ALGORITHMS COMPLEXITY AND ANALYSIS 12 hours

The Role of Algorithms in Computing, analysing algorithms, Growth of functions, Asymptotic notation, recurrence relations, solving recurrence relations: substitution method, recursion-tree method, Master method, analysis of searching techniques: Binary Search.

UNIT IV SORTING AND GRAPH ALGORITHMS 12 hours

Quick Sort, Merge Sort, Heap Sort, Sorting in Linear Time: Counting Sort, Radix Sort- Minimum Spanning Trees (prims and Kruskal's algorithms), Shortest-path Algorithms: Dijkstra's and The Floyd-Warshall algorithm.

UNIT V PROBLEM-SOLVING METHODS AND APPROXIMATION ALGORITHMS **12 hours**

Dynamic Programming Methods: All pairs shortest paths, travelling salesman problem; Backtracking- N-queen problem, Graph coloring; Branch and Bound- Knapsack, Introduction to P, NP, NP-Hard and NP-Complete. NP-Completeness and Reducibility, Cook's Theorem without proof.

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Understand the primary data structure and its implementation.
CO2: Identify and apply appropriate advanced data structures in problem-solving.
CO3: Analyze the complexity of various algorithms in terms of growth functions.
CO4: Implement sorting and shortest-path algorithms.
CO5: Apply algorithm design approaches with real-time applications.

Text Book(s)

1. Cormen, Thomas H., Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. Introduction to algorithms. MIT press, 2022.

2. Rajesh K. Shukla, *Data Structures using C & C++ : As per AICTE,* Wiley India publications, 2019

Reference Books

1. Alsuwaiyel, Muhammad H. *Algorithms: design techniques and analysis*. Vol. 15. World Scientific, 2021.

2. Sedgewick, Robert. An introduction to the analysis of algorithms. Pearson Education India, 2013.

3. McMillan, Michael. *Data structures and algorithms using C*. Cambridge University Press, 2007.

4. Storer, James Andrew. *An introduction to data structures and algorithms*. Springer Science & Business Media, 2012.

5. Soltys-Kulinicz, Michael. *Introduction to the Analysis of Algorithms, An*. World Scientific, 2018.

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

Dept. of Computer Applications

MCA I Year I Semester

24MCAP103 PYTHON PROGRAMMING

L	T	P	C
3	1	0	4

Pre-requisite None

Course Description:

This course provides the students how to write programs in python language to perform different tasks in major concepts such as functions, collections, comprehensions, files, exception handling and Database Operations.

Course Objectives:

This course enables students to

1. Learn the basics of python programming such as variables, data types, operators, etc.
2. Understand the concept of Lambda and user defined functions
3. Examine the use of collections such as string, list, set, tuple, dictionary
4. Analyse the use of comprehensions and files
5. Review the concepts of exceptions and Database Operations

UNIT I FUNDAMENTALS 12 hours

History, Python-Interpreter, Keywords, Variables, Datatypes, Operators, Expressions, Input and Output functions, Type conversion functions, Indentations, Conditionals -if constructs, Loop Structures -For loop, While loop, break and continue statements, Modules, Packages and Libraries

UNIT II LAMBDA AND USER DEFINED FUNCTIONS 12 hours

Lambda functions, User Defined Function definition and function call, parameter passing, Pass and Return statements, Types of functions -Default arguments, Arbitrary arguments using *, Keyword arguments, Arbitrary Keyword arguments using **, Recursive functions, Local, nonlocal and Global variables.

UNIT III COLLECTIONS 12 hours

String operations and functions, Slicing and Indexing operations, format function, multiple inputs using split function, List operations and functions, Tuple operations and functions, Set operations and functions, Dictionary operations and functions, Array and its operations.

UNIT IV COMPREHENSIONS AND FILE HANDLING 12 hours

List comprehensions -for, for with if, for with if and else, nested comprehensions with for, Dictionary comprehensions, Dictionary comprehension with if, Set comprehensions, map, zip, filter and other related functions, File path, opening and closing files, reading and writing files, File position.

UNIT V EXCEPTION HANDLING AND DATABASE CONNECTIVITY 12 hours

Errors and Exceptions, Exception Handling, Multiple Exceptions, raise statement, User Defined Exceptions, Database connectivity - Python MySQL Module - Creating database, Creating tables, Insert, Read, and Update operations.

Dept. of Computer Applications

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Write simple programs in python
CO2: Apply lambda and user defined functions
CO3: Develop programs in collections -string, list, set, tuple, dictionary
CO4: Implement programs using comprehensions and files
CO5: Apply exceptions and OOPS concepts in python applications

Text Book(s)

1. Programming with Python by Kyla Mcmullen, Elizabeth Matthews, June Jamrich Parsons, Cengage Learning 1st Edition 2024
2. Core Python Programming by Nageswara Rao R, Dreamtech Press,3rd edition, 2021

Reference Books

1. Learn Python Programming for Beginners by Flynn Fisher, Aron Chase, 2020
2. Python Programming Guide. Exploring Computer Programming by Sheetal Thakare, Fachbuch, 2020
3. Python: The Complete Reference by Martin C. Brown, McGraw Hill Education 2018

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

Dept. of Computer Applications

MCA I Year I Semester

24MCAP104 OPERATING SYSTEMS

L	T	P	C
3	1	0	4

Pre-requisite None

Course Description:

This course is designed to Explore the fundamental principles of operating systems, including process management, memory allocation, file systems, and advanced topics, through a condensed yet comprehensive five-unit syllabus.

Course Objectives:

This course enables students to

1. Understand the fundamental concepts, functions, and structure of modern operating systems.
2. Impart knowledge on process synchronization mechanisms, CPU scheduling algorithms, and strategies for handling deadlocks in operating systems.
3. Analyze memory management techniques and virtual memory concepts in operating systems.
4. Understand the fundamentals of disk management, RAID structures, and file-system implementation and management techniques in operating systems.
5. Assess the real time and mobile operating systems.

UNIT I INTRODUCTION TO OPERATING SYSTEMS 12 hours

Introduction to operating systems, Types of operating systems, Operating system services, System calls, Types of System calls, System Programs, Operating system operations, Process concepts, Process scheduling, Operations on Processes, Inter-process communication.

UNIT II PROCESS SYCHRONIZATION AND CPU SCHEDULING 12 hours

Process Synchronization: Background, The Critical-section problem, Semaphores, Classic Problems of Synchronization, Monitors, CPU Scheduling: Basic concepts, Scheduling Criteria, Scheduling Algorithms, Deadlocks: Deadlock Characterization, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery from Deadlock.

UNIT III MEMORY MANAGEMENT 12 hours

Background, Swapping, Contiguous Memory Allocation, Segmentation, Paging Virtual Memory: Background, Demand Paging, Copy-on-Write, Page Replacement, Allocation of Frames Thrashing

UNIT IV STORAGE MANAGEMENT 12 hours

Disk Structure, Disk Scheduling, Disk Management, Swap-Space Management, RAID Structure, File-System Interface: File Concept, Access Methods, File-System Mounting, File Sharing, Protection, File-System Implementation, Directory Implementation, Allocation Methods, Free-Space Management.

UNIT V REAL TIME OPERATING SYSTEM 12 hours

Real-time operating systems, Multiprocessor and distributed operating systems, Mobile OS - Android vs IOS layered architecture, Cloud and IoT Operating Systems. Case studies: Analysing Linux and Windows operating systems.

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Understand the key operating system concepts, services, system calls, process management, and inter-process communication.

CO2: Apply synchronization techniques, CPU scheduling algorithms, and manage deadlocks in operating systems.

CO3: Analyse memory management techniques such as segmentation, paging, and virtual memory concepts.

CO4: Implement RAID structures, and effectively handle file-system operations including file access, protection, and space allocation.

CO5: Evaluate and work with real time operating systems.

.

Text Book(s)

1. Avi Silberschatz ,Peter Baer Galvin, Greg Gagne, "Operating System Concepts", 10th edition, 2018. William Stallings, "Operating Systems: Internals and Design Principles", 9th edition,2022.

Reference Books (format change)

1. "Operating Systems Design and Implementation" 3rd edition by Andrew S. Tanenbaum and Albert S. Woodhull.2006.

2. "Operating Systems: Principles and Practice" 2nd edition by Thomas Anderson and Michael Dahlin .2014.

3 "Modern Operating Systems" Fourth edition by Andrew S. Tanenbaum and Herbert Bos 2001

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

Dept. of Computer Applications

MCA I Year I Semester

24MCAP201 DATABASE MANAGEMENT SYSTEMS LABORATORY

L	T	P	C
0	0	3	1.5

Pre-requisite None

Course Description:

This course provides an executive understanding and approach to the technical subject of database management system.

Course Objectives:

This course enables students to

1. Explain DDL, DML,TCL and DCL Commands
2. Implement sub queries, any, all, in, exists, union, intersect and aggregate functions
3. Experiment PL/SQL functions, Procedure functions, Triggers and exceptions.

LIST OF EXPERIMENTS

1. Implementation of DDL commands of SQL with suitable examples
    a. Create table, Alter table, Drop Table
2. Implement DML and DCL Commands by using various examples.
3. QUERIES (ALONG WITH SUB QUERIES) USING ANY, ALL, IN, EXISTS, OTEXISTS, UNION, INTERSECT
4. QUERIES USING AGGREGATE FUNCTIONS (COUNT, SUM, AVG, MAX AND MIN) GROUP BY, HAVING and Creation and dropping of Views..
5. Consider the following schema for Order Database:
    a. SALESMAN(Salesman_id, Name, City, Commission)
    b. CUSTOMER(Customer_id, Cust_Name, City, Grade, Salesman_id)
    c. ORDERS(Ord_No, Purchase_Amt, Ord_Date, Customer_id, Salesman_id)
    d. Write SQL queries to
        1. Count the customers with grades above city's average.
        2. Find the name and numbers of all salesmen who had more than one customer.
        3. List all the salesmen and indicate those who have and do not have customers in their cities (Use UNION operation.)
        4. Create a view that finds the salesman who has the customer with the highest order of a day.
        5. Demonstrate the DELETE operation by removing salesman with id 1000. All his orders must also be deleted.
6. Create a row level trigger for the customers table that would fire for INSERT or UPDATE or DELETE operations performed on the CUSTOMERS table. This trigger will display the salary difference between the old & new Salary.

    CUSTOMERS (ID, NAME, AGE, ADDRESS, SALARY)

7. Create cursor for Employee table & extract the values from the table. Declare the variables ,Open the cursor & extract the values from the cursor. Close the cursor.
    Employee(E_id, E_name, Age, Salary)
8. Write a PL/SQL block of code using parameterized Cursor, that will merge the data available

in the newly created table N_RollCall with the data available in the table O_RollCall. If the data in the first table already exist in the second table then that data should be skipped.

9. Implement functions, procedures in PL/SQL
10. Implement User defined and System defined exceptions.
11. **Mini Project:**

Each student has to implement any one of the following system

i. Employee management system
ii. Student Result information system
iii. Bank Management system
iv. Library information system
v. Hotel management system
vi. Railway reservation system

Note: Each system contains minimum of 6 tables. Each table contains minimum 10 rows.

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Execute DDL, DML and DCL commands.
CO2: Implement PL/SQL procedures, functions, cursors, triggers and exceptions
CO3: Design Relational model for the given system

Text Book(s)

1. Benjamin Rosenzweig, Elena Silvestrova, ORACLE PL/SQL by example, Pearson Education,2023

2. Dr.P.S. Deshpande, SQL & PL/SQL for Oracle 11g, Black Book. Dreamtech Press, 2011

Reference Books

1. Rick F.Vander Lans, Introduction to SQL: Mastering the Relational Database Language, Pearson Education, 2006.

2. Steven Feuerstein with Bill Pribyl, Oracle PL/SQL Programming, O'Reilly Media, Inc. Sixth edition, 2014

3. N.Gehani, Database Book, The: Principles and Practice Using MySQL, Universities Press. 2008

Mode of Evaluation: Continuous Internal Evaluation and End Semester Examination

Dept. of Computer Applications

MCA I Year I Semester

24MCAP202 DATA STRUCTURES AND ALGORITHMS LABORATORY (USING C)

L	T	P	C
0	0	3	1.5

Pre-requisite None

Course Description:

This course introduces on the practical part of Data Structures and Algorithms using C language. This course allows students to implement linear and nonlinear data structures. It gives practical exposure for solving sorting and searching problems. Also, it enables to develop programs using various problem-solving methods.

Course Objectives:

This course enables students to

1. Implement Linear and Non-Linear Data structures.
2. Analyse different types of searching and sorting techniques.
3. Apply various problem-solving methods to real-time Applications

LIST OF EXPERIMENTS

1. Create a Stack and do the following operations.
    a. Push b. Pop c. Peep
2. Create a queue and do the following operations.
    a. Enqueue b. Dequeue c. Display
3. Write a program to convert from infix to postfix expression.
4. Implement the operations on the following data structures:
    a. singly linked list b. circular linked list c. doubly linked list
5. Implement the operations (insert, delete, search) on Binary Search tree.
6. Write a program on Tree Traversal
    a. Pre order b. In order c. Post order
7. Write a program to implement the following trees
    a. Splay Tree b. AVL Tree c. Red-Black Tree
8. Implement the following Divide and Conquer techniques.
    a. Binary search b. Quick sort c. Merge sort
9. Write a program to implement Heap sort for the given list of integer values.
10. Implement Graph traversals.
    a. BFS b. DFS
11. Write a program to find minimum spanning tree using
    a. Prim's method b. Kruskal's method.
12. Implement single source shortest path problem.
13. Implement all pairs shortest path problem.
14. Implement N-queen's problem using backtracking.
15. Write a program to find the solution for a knapsack problem using Branch and Bound

Course Outcomes:

After completion of the course the student will be able to

CO1: Implement Linear and Non-Linear Data structures.

CO2: Demonstrate sorting and searching techniques.

CO3: Apply various problem-solving techniques for complex problems.

Text Book(s)

1. Rajesh K. Shukla, Data Structures using C & C++ : As per AICTE, Wiley India publications, 2019

2. Cormen, Thomas H., Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. Introduction to algorithms. MIT press, 2022

Reference Books

1. Alsuwaiyel, Muhammad H. Algorithms: design techniques and analysis. Vol. 15. World Scientific, 2021.

2. Sedgewick, Robert. An introduction to the analysis of algorithms. Pearson Education India, 2013.

3. McMillan, Michael. Data structures and algorithms using C. Cambridge University Press, 2007.

Mode of Evaluation: Continuous Internal Evaluation and End Semester Examination

Dept. of Computer Applications

MCA I Year I Semester

24MCAP203 PYTHON PROGRAMMING LABORATORY

L	T	P	C
0	0	3	1.5

Pre-requisite Any Programming Language

Course Description:

This course provides the students how to write programs in python language to perform different tasks in major concepts such as functions, collections, comprehensions, files, exception handling and Database Operations.

Course Objectives:

This course enables students to

1. Write algorithms and draw flowcharts using raptor tool for different computational problems
2. Develop Python programs using looping, decision making and user defined functions.
3. Implement Python programs using collections, comprehensions, files, exceptions and Database Operations

LIST OF EXPERIMENTS

SIMPLE PROGRAMS USING IF CONSTRUCTS AND CONTROL STRUCTURES

1.Draw flow charts for the following problems using Raptor Tool

a. To exchange/swap the values of two variables
b. To find factorial of a given number
c. To find the sum of set of numbers
d. To generate Fibonacci sequence
e. To reverse the digits of a number

2.Develop Python scripts for the following

a. To count and display the total number of prime numbers in given two ranges
b. To calculate the sum of digits of given N digit number
c. To generate Armstrong numbers
d. To check the given number for perfect or not
e. To find the factors of the given number

USING LAMBDA AND USER DEFINED FUNCTION

3.Develop Python scripts for the following using Lambda function

a. To create a lambda function that multiplies argument x with argument y
b. To filter, odd and even numbers
c. To square and cube every number in a given list
d. To extract year, month, date and time
e. To find palindromes in a given list of strings

4.Develop Python scripts for the following using User Defined Functions

a. To check the given number for Prime or not
b. To check the given number for odd or even
c. To calculate the length of given string
d. To count vowels in given string
e. To calculate the sum of all items in the given list

USING COLLECTIONS

5.Develop a Python script for String to calculate the length, to reverse, to count and display the Vowels, to remove spaces and to convert from lowercase to uppercase of the given string without using String built-in functions

6.Develop a Python script to implement slicing and indexing operations on String, List, Tuple, Set and Dictionary

7.Develop a Python script for List to sum all the items, to count the total number of items, to check the list is empty or not, to get the largest number and to shuffle & print the given list without using List built-in functions

8.Develop a Python script to implement the Set operations – Union, Difference, Intersection, All & any and Enumerate functions.

9.Develop Python scripts for Tuple to demonstrate packing & unpacking, to check for membership with in & not in operators and to concatenate more than one tuple into a single tuple

10.Develop a Python script for Dictionary to create & add items, to modify existing values using keys, to Update, to print values and to print keys

11.Develop a Python script by using Array to sort the array elements, to add given two matrix values and to transpose the given matrix

USING COMPREHENSIONS AND FILES

12.Develop Python scripts by using Comprehensions to print a list with if, to print a list with if-else, to print a list with nested comprehensions, to print Dictionary items with a condition using zip function and to print Set items

13.Develop a Python script to calculate individual student attendance percentage by using Files

USING EXCEPTION HANDLING AND DATABASE CONNECTIVITY

14.Develop a Python script to validate voting age by using Exceptions

15. Develop a Python script to implement database operations using student marks details.

Course Outcomes:

Upon successful completion of the course, students will be able to
CO1: Draw flow charts for different tasks using raptor tool
CO2: Write python basic programs using conditional, looping structures and functions
CO3: Develop python programs for collections, comprehensions, files, exceptions and Database Operations

Text Book(s)

1. Programming with Python by Kyla Mcmullen, Elizabeth Matthews, June Jamrich Parsons, Cengage Learning 1st Edition 2024
2. Core Python Programming by Nageswara Rao R, Dreamtech Press,3rd edition, 2021

Reference Book(s)

1. Learn Python Programming for Beginners by Flynn Fisher, Aron Chase, 2020
2. Python Programming Guide. Exploring Computer Programming by Sheetal Thakare, Fachbuch, 2020
3. Python: The Complete Reference by Martin C. Brown, McGraw Hill Education 2018

Mode of Evaluation: Continuous Internal Evaluation and End Semester Examination

MCA I Year II Semester

MCA I Year II Semester

24MCAP105 JAVA PROGRAMMING

L	T	P	C
3	1	0	4

Pre-requisite Nil

Course Description:

This course introduces the core and advanced concepts of Java. It covers the core concepts of JVM architecture and OOPS concepts for building platform independent java applications. This course also deals with Implementation of Exception Handling, multi-threading in java programs. It also introduces the advanced concepts like Collections and Collection Framework

Course Objectives:

Students will be able to

1. Understand how to use Java to write Platform independent applications.
2. Analyze use of Java Classes and Objects, methods and constructors.
3. To Apply the concepts of Inheritance, Interfaces and Packages to build java programming.
4. Explore multi-threading, Java Collection Framework concepts.
5. Demonstrate JDBC Connections and execute SQL statements.

UNIT I INTRODUCTION TO JAVA 12 hours

The History of Java, Different versions, Java Virtual Machine Architecture-Class Loader-JRT, Java Buzzwords, Object Oriented Programming and its principles, Java Primitive Data types, Variables, Type Casting: Widening vs Narrowing.

Programming Constructs: Selection, Iteration and Jump Statements. Arrays, Jagged Arrays.

UNIT II CLASSES AND OBJECTS 12 hours

Class Fundamentals, declaring Objects, Introducing Methods, Constructors, this and super Keywords at Various Levels,Static keyword at various levels.

Method Over Loading, Method Overriding, using objects as parameters, Introducing Access Control, understanding static, introducing final, String vs String Buffer.

UNIT III PACKAGES& EXCEPTIONS 12 hours

Interfaces, Multiple Inheritance Issues, Packages and CLASSPATH, Access protection, importing packages, package example, Introducing to ***java. lang*** and ***java.io***packages.

Introduction, Exception handling fundamentals, Exception types, using try and catch, Multiple catch clauses, nested try statements, throw, throws, finally block, Java's Built-in-Exceptions, user defined exception.

UNIT IV MULTI-THREADING& JAVA COLLECTION FRAMEWORK. 12 hours

Multi-Threading: The Java Thread Model, Thread Life Cycle, the Main Thread, Creating a Thread, Multiple threads, Using isAlive(), join(), Thread Synchronization, Collection Framework,-Collection Hierarchy, Iterable Interface, List interface, Array List, Linked List, Vector, Stack, Queue, set

UNIT V JDBC: DATABASE DRIVERS AND CONNECTION MANAGEMENT 12 hours

Type1: JDBC-ODBC driver, Type2: Native-API driver, Type3: Network-Protocol driver,
Type4: Database-Protocol driver. **javax.sql package:** Connection management, Database access, Exceptions and Warnings, JDBC connectivity, Creating and executing SQL statements, prepared statement, Scrollable & Updatable resultset, javax.sql.DataSource Interface.

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Understand the fundamentals of Java for problem solving.
CO2: Implement classes and method overloading concepts.
CO3: Develop user defined package by following security mechanisms
CO4: Implement multi-threaded applications and design applications using Collection Framework.
CO5: Implement JAVA applications using JDBC.

Text Book(s)

1. The Complete Reference Java, 9ed, Herbert Schildt, TMHProgramming in JAVA, Sachin Malhotra, Saurabh choudhary, Second Edition, Oxford

2. Advanced Java Dr.C.Muthu-Shalom Infotech

Reference Books

1. Object oriented programming with JAVA, Essentials and Applications, Raj Kumar Buyya, Selvi, Chu TMH

2. Core Java Volume 1. Fundamentals, 8ed, Cay S.Horstmann, Gray Cornell, Pearson

3. Advanced Java Programming uttam K.Roy-15 th Addition-Oxford Publications.

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

MCA I Year II Semester

24MCAP106 COMPUTER NETWORKS AND SECURITY

L	T	P	C
3	1	0	4

Pre-requisite **None**

Course Description:

This course provides an introduction to computer networks, with a special focus on the Internet architecture and protocols. Topics include layered network architectures, addressing, naming, forwarding, routing, communication reliability, the client-server model, web and email protocols. Network security uses authentication, authorization, data encryption, and intrusion detection techniques. In comparison, cryptography uses encryption, decryption, digital signatures, and key management techniques.

Course Objectives:

This course enables students to

1. Acquire the knowledge in networking concepts and standards.
2. Understand network architectures of Data link layer and Network layer.
3. Analyze the function and design strategy of Transport layer and Application layer.
4. Learn to analyze the security of in-built cryptosystems.
5. Understand Network Layer Security and IPsec.

UNIT I NETWORK BASICS **12 hours**

Introduction- Network Types- Network Topologies- Networking Devices - Reference models: The OSI Reference Model- the TCP/IP Reference Model – Networks Standards - Transmission Media.

UNIT II DATA LINK LAYER & NETWORK LAYER **12 hours**

Introduction to data link layer: Framing-Sliding Window Protocol-Media Access Control-Wired LANs.

Introduction to Network Layer: IP Address Concepts – Switching: Circuit Switching and Packet Switching - Routing Algorithms: Distance vector (RIP), Link state (OSPF) and Path vector (BGP).

UNIT III TRANSPORT LAYER & APPLICATION LAYER **12 hours**

Transport layer protocols: Introduction – Services - Port number - User data gram protocol: UDP applications - Transmission control protocol: TCP services - World Wide Web: HTTPs, Electronic mail-Architecture - Domain Name System:

UNIT IV INTRODUCTION TO COMPUTER AND NETWORK SECURITY **12 hours**

Computer Security Concepts – The OSI Security Architecture – Security Attacks – Security Services and Mechanisms – A Model for Network Security – Classical encryption techniques: Substitution techniques, Transposition techniques, Steganography – Foundations of modern cryptography: Perfect security – Information Theory – Product Cryptosystem – Cryptanalysis.

UNIT V NETWORK LAYER SECURITY 12 hours

Introduction to Network Layer Security; IPSec Protocol - Security Associations (SAs) - Hashed Message Authentication Code (HMAC) - IP Authentication Header - IP ESP - Key Management Protocol for IPSec. Introduction: Cloud Security – Web Security – Wireless Security.

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Understand the Basic Computer Networks Concepts.
CO2: Identify the needs of Data link layer and Network layer.
CO3: Learn the functionalities Transport layer and Application Layer.
CO4: Examine the networks security, threats and vulnerabilities.
CO5: Analyze the Network Layer Security aand IPSec.

Text Book(s)

1. Andrew S.Tannenbaum David J. Wetherall, "Computer Networks" Fifth Edition , Pearson Education 2022.
2. Behrouz A. Ferouzan, Debdeep Mukhopadhyay ―Cryptography & Network Security, 3rd edition, Tata McGraw Hill, 2021.

Reference Books

1. Douglas E. Comer, ―Internetworking with TCP/IP (Volume I) Principles, Protocols and Architecture, Sixth Edition, Pearson Education, 2013.
2. Forouzan, " Data Communication and Networking", Fifth Edition , TMH 2012.
3. William Stallings, ―Data and Computer Communications‖, Tenth Edition, Pearson Education, 2013
4. William Stallings "Network Security Essentials Applications and Standards", Pearson Education., 5th Edition, 2014.

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

Dept. of Computer Applications

MCA I Year II Semester

24MCAP107 ARTIFICIAL INTELLIGENCE

L	T	P	C
3	1	0	4

Pre-requisite None

Course Description:

To understand the importance of AI techniques and its wide range of applications. The course is introduced with basics of AI along with the techniques and algorithms of AI. It also includes applications of AI and Tools of AI.

Course Objectives:

This course enables students to

1. Learn the fundamentals of Artificial Intelligence.
2. Understand the problem solving techniques of Artificial Intelligence.
3. Learn the basic types of logic and knowledge representation.
4. Explain NLP and its applications.
5. Examine various real time smart applications of various domains.

UNIT I INTRODUCTION TO AI 12 hours

AI: What is AI, Turing Test, Evolution of AI, Agents and Environment, Expert Systems, Knowledge representation and reasoning, Uncertain and probabilistic reasoning., Applications of AI.

UNIT II PROBLEM SOLVING IN AI 12 hours

Problem solving as a State Space searching strategies-BFS, DFS, Gaming Problem-Tic-Tac-Toe, Heuristic search techniques: Hill Climbing, Best First Search, Heuristic Functions, A* algorithm, AO* algorithm. Advanced search, Constraint satisfaction problems.

UNIT III KNOWLEDGE REPRESENTATION 12 hours

Propositional Logic: Representation, Inference, Reasoning Patterns, Resolution, Forward and Backward Chaining. First order Logic: Representation, Inference, Reasoning Patterns, Resolution, Forward and Backward Chaining.

UNIT IV NLP & CHATBOTS 12 hours

Natural Language Processing: Introduction, Natural Language Understanding, Components of NLP, Enterprise Applications, NLP Pipeline, NLP libraries, Challenges of NLP - Chatbots: Introduction, how to Build a Chatbot, Architecture of Chatbot, Challenges of building a Chatbot, Industry case studies - Virtual Assistants.

UNIT V SMART APPLICATIONS & AI TOOLS 12 hours

Smart Applications: Smart Manufacturing, Smart Agriculture, Smart Grids, Smart Transportation and Autonomous Vehicles, Smart Homes, Smart Cities.

AI Tools: ChatGPT, Veed, GitHub, Google Gemini, Quill Bot, Midjourney.

Course Outcomes:

Upon successful completion of the course, students will be able to

1. Demonstrate the basic concepts of AI and its applications
2. Illustrate the various AI algorithms
3. Analyze problem using suitable knowledge representation.
4. Implement the NLP based applications, Chatbots.
5. Design smart applications for various domains.

Text Book(s)

1. Elaine Rich and Kevin Knight, Artificial Intelligence, 2nd Edition, 2018.

2. Stuart J. Russell and Peter Norvig, Artificial Intelligence A Modern Approach 3. Joseph Howse, Prateek Joshi, Michael Beyeler - Opencv_ Computer Vision Projects with Python-Packt Publishing (2016)

Reference Books

1. Munesh Chandra Trivedi, A classical approach to Artificial Intelligence, Khanna Publications, 2018.

2. Tom Markiewicz & Josh Zheng, Getting started with Artificial Intelligence, Published by O'Reilly Media,2017.

3. Chandra S.S. & H.S. Anand, Artificial Intelligence and Machine Learning, PHI Publications, 2014.

4. Machine Learning, Rajiv Chopra, Khanna Publishing House, 2018.

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

Dept. of Computer Applications

MCA I Year II Semester

24MCAP108 SOFTWARE ENGINEERING

L	T	P	C
3	1	0	4

Pre-requisite: None

Course Description:

This course aims to provide advanced knowledge and skills in software engineering principles, methodologies, and practices. It covers advanced topics such as software architecture, design patterns, software quality assurance, software testing, and emerging trends in software engineering. The course emphasizes both theoretical concepts and practical applications through projects, and hands-on exercises.

Course Objectives:

This course enables students to,

1. Understand software engineering principles and methodologies.
2. Analyse the various activities and approaches involved in requirement engineering, including elicitation, analysis, specification, and validation.
3. Explore the concept of software modelling and design patterns.
4. Learn advanced techniques for software quality and testing.
5. Analyse the software project management and estimation techniques

UNIT I SOFTWARE ENGINEERING CONCEPTS AND PROCESS MODELS — 12 hours

Defining Software, Software Application Domains, Legacy Software, The Software Process - software myths- -A layered technology– Software process models: A Generic Process Model- Prescriptive Process Models- The Unified Process

UNIT II REQUIREMENT ENGINEERING — 12 hours

Introduction: Characteristics Of a Good Requirement- Types of Software Requirements, Establishing The Groundwork, Requirements Elicitation, Developing Use Cases, Building The Requirements Model, Negotiating Requirements, Validating Requirements

UNIT III SOFTWARE MODELLING AND DESIGN — 12 hours

Requirements analysis, scenario-based modeling, developing an activity diagram, data modeling Concepts, Class-Based Modelling, Flow-Oriented Modeling, UML Diagrams, The Design Process, Design Concepts, The Design Model.

UNIT IV SOFTWARE TESTING AND MAINTENANCE — 12 hours

A Strategic Approach to Software Testing, Strategic Issues, Test Strategies for Conventional Software, White-Box Testing, Basis Path Testing-Control Structure Testing-Black Box Testing-Unit Testing — Integration Testing — Validation Testing - System Testing and Debugging-Maintenance and Reengineering-BPR model-Reengineering process model-Reverse and Forward Engineering.

UNIT V PROJECT MANAGEMENT 12 hours

Software Project Management: Project Management Concepts, Software Measurement Metrics for Software Quality, Software Project Estimation — LOC, FP Based Estimation, COCOMO I & II Model- Make/Buy Decision — Project Scheduling —Earned Value Analysis, Risk Management — Identification, Projection — Risk Management-Risk Identification-RMMM Plan-CASE TOOLS.

Course Outcomes:

Upon successful completion of the course students should be able to:

CO1: Understand the process to be followed in the software development life cycle
CO2: Analyze software requirements and their critical role in software engineering projects.
CO3: Implement various software modelling and design techniques
CO4: Differentiate the testing strategies and maintenance of the software project.
CO5: Analyze software project management and risk management techniques

Text Book(s)

1. Roger Pressman, Software Engineering - A Practitioner's Approach, 9th Edition, Tata McGraw Hill, 2023.

Reference Books

1. Richard Fairley, Software Engineering Concepts, Tata McGraw Hill, New India Edition, 2017.
2. Robert C. Martin, Agile Software Development, Principles, Patterns, and Practices Alan Apt Series, 2011.
3. Wendy Boggs and Michael Boggs, Mastering UML with Rational Rose, Wiley, 2009.
4. Rajib Mall, "Fundamentals of Software Engineering", 5th Edition, PHI, 2018.

Mode of Evaluation: Assignments, Mid-Term Tests, End Semester Examination.

Dept. of Computer Applications

MCA I Year II Semester

24MCAP204 JAVA PROGRAMMING LABORATORY

L	T	P	C
0	0	3	1.5

Pre-requisite None

Course Description:

This course provides the students how to write programs in java language to perform different tasks in major concepts related to OOPS, Handling the Runtime Exceptions, working with String Classes, Creating Threads and implement applications using Multithreading Environment, Java Collection Framework, JDBC.

Course Objectives:

This course enables students to

1. Write basic Java Programs and implement oops concepts in java applications.
2. Read the User Input from various predefined classes and perform different operations on input data.
3. Develop User threads by handling runtime errors and establish Database connectivity.

LIST OF EXPERIMENTS

PROGRAMS ON DATA TYPES, CONDITIONAL STATEMENTS AND LOOPS

Exercise - 1

a) Write a JAVA program to display default value of all primitive data type of JAVA

b) write a java program to illustrate different types of java variables?

Excercise-2

Five Bikers Compete in a race such that they drive at a constant speed which may or may not be the same as the other. To qualify the race, the speed of a racer must be more than the average speed of all 10 racers. Take as input the speed of each racer and print back the speed of qualifying racers.

Excercise-3

a) Write a Java program for creating one base class for student personal details and inherit those details into the sub class of student educational details to display complete student information.

b) Write a java program to display the employee details using Scanner class.

PROGRAMS ON OBJECT ORIENTED CONCEPTS & JAVA STRING CLASSES.

Excercise-4

Write a java program to demonstrate Runtime polymorphism, using subclass and super class hierarchy and override superclass methods using parent child references.

Excercise-5

Write a program to create a class Student along with two method getData(),printData() to get the value through argument and display the data in printData. Create the two objects s1, s2 to declare and access the values from class STtest.

Excercise-6

a) Write a java program to illustrate character string handling methods for a given input string.

b) Write a Java program to reverse a string without using the reverse method of Java's String class

PROGRAMS ON EXCEPTIONS AND MULTI THREADING

Excercise-7

a) Write a java program to handle predefined exceptions in java?

b) Write a JAVA program to check a person is eligible for vote or not using user defined exception

Excercise-8

Write a JAVA program to create an abstract class named Shape that contains two integers and an empty method named printArea(). Provide three classes named Rectangle, Triangle and Circle such that each one of the classes extends the class Shape. Each one of the classes contains only the method printArea() that prints the area of the given Shape.

Excercise-9

a) Write a java program to create a java thread using extending Thread Class and Runnable Interface?

b) Write a java program in which thread sleep for 5 sec and change the name of thread

Excercise-10

Write a java program to solve producer consumer problem in which a producer produces a value and consumer consume the value before producer generate the next value.

PROGRAMS ON JAVA COLLECTIONS, DATA BASE CONNECTIVITY.

Excercise-11

Write a java program to create Array List for Student Courses and perform the following operations

i. Add ii. Access iii. Update iv. Delete

Excercise-12

Write a Java Program to Get Elements of a LinkedList and display on the console.

Excercise-13

Write a Java Program to get First, Last and Maximum element from the Vector.

Excercise-14

Write a Java Program to compare elements in a Collection List and display duplicate elements from the List.

Excercise-15

Write a java program that connects to a database using JDBC and perform the query processing operations

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Write the basic Java Programs and implement oops concepts in java applications.
CO2: Apply Exceptional handling techniques in JAVA applications
CO3: Develop multi-threaded applications and perform Query processing operations.

Text Book(s)

1. The Complete Reference Java, Herbert Scheldt, 13th Edition, Feb 2024
2. Core Java Volume I - Fundamentals, Pearson 12th Edition -December 2023

Reference Books

1. Programming with Java, E Balaguruswsmy,6th Edition, Mar2021
2. Java Programming Fundamentals, Premchand S Nair, CRC press-3rd edition, 2021
3. JAVA: A Beginner's Guide, Herbert Scheldt, Standard Edition

Mode of Evaluation: Continuous Internal Evaluation and End Semester Examination

MCA I Year II Semester

24MCAP205 COMPUTER NETWORKS AND SECURITY LABORATORY

L	T	P	C
0	0	3	1.5

Pre-requisite None

Course Description:

The intention of Network Laboratory is to learn fundamental concepts related to networking. It is equipped with well-connected server and client structured systems to support the study of the concepts and protocols of networking and network security, anonymity and privacy, web security and applied cryptography for the effective implementation of the routing protocols.

Course Objectives:

This course enables students to

1. Learn network fundamentals and commands.
2. Examine various network protocols and security algorithms
3. Analyze the performance of digital signature Scheme and Penetration tools.

LIST OF EXPERIMENTS:

1. Study of different types of Network cables and practically implement the cross wired cable and straight through cable using crimping tool.
2. Study of basic network command and Network configuration commands using CISCO Routers and SWITCHES.
3. Write a HTTP web client program to download a web page using TCP sockets.
4. Simulation of DNS using UDP & TCP sockets.
5. Configure a Network using Distance Vector Routing protocol: RIP.
6. Configure Network using Link State Vector Routing protocol: OSPF, IS-IS.
7. Configure Network using Path Vector Routing protocol: BGP.
8. Perform encryption, decryption using the following substitution techniques.
   (i) Ceaser cipher, (ii) Play fair cipher iii) Hill Cipher iv) Vigenere cipher.
9. Configuration of the ISP-DNS server with host records and MX records so that any internet clients could reach the internet servers for HTTPs, FTP, POP3/IMAP and SMTP services.
10. Configure Site to Site IPsec VPN between two different locations.

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Understand the network fundamentals and commands.
CO2: Apply various network protocols and security algorithms.
CO3: Demonstrate the performance of digital signature Scheme and Penetration tools.

Text Book(s)

1. Andrew S.Tannenbaum David J. Wetherall, “Computer Networks” Fifth Edition , Pearson Education 2022.
2. Behrouz A. Ferouzan, Debdeep Mukhopadhyay ―Cryptography & Network Security, 3rd edition, Tata McGraw Hill, 2021.

Reference Books

1. William Stallings, ―Data and Computer Communications‖, Tenth Edition, Pearson Education, 2013.
2. Douglas R. Stinson," Cryptography: Theory and Practice, Third Edition (Discrete Mathematics and Its Applications), Chapman & Hall/CRC, 2005.
3. William Stallings “Network Security Essentials Applications and Standards", Pearson Education., 5th Edition, 2014.

Mode of Evaluation: Continuous Internal Evaluation and End Semester Examination

Dept. of Computer Applications

MCA I Year II Semester

24MCAP206 ARTIFICIAL INTELLIGENCE LABORATORY

L	T	P	C
0	0	3	1.5

Pre-requisite None

Course Description:

This course introduces the core concepts and techniques of Artificial Intelligence (AI), focusing on algorithm implementation, search strategies, and intelligent agent design. Students will explore problem-solving, knowledge representation, and decision-making in AI systems, supported by hands-on activities and real-world case studies.

Course Objectives:

This course enables students to

1. Explore the methods of implementing algorithms using artificial intelligence techniques
2. Illustrate search algorithms
3. Demonstrate building of intelligent agents.

LIST OF EXPERIMENTS

1. Introduction to AI libraries of Python.
2. Write a program to implement BFS.
3. Write a program to implement DFS.
4. Write a program to implement Tic-Tac-Toe game using Python.
5. Write a program to implement Hill Climbing Algorithm
6. Write a program to implement A* Algorithm
7. Write a program to implement AO* Algorithm
8. Write a program to implement 8 puzzle problem
9. Write a program to implement Water-Jug problem using Python.
10. Write a program to implement Towers of Hanoi problem
11. Build a Chatbot which provides all the information related to MITS college.
12. Build a virtual assistant for Wikipedia using Wolfram Alpha and Python

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Experiment the AI problem solving algorithms.

CO2: Implement Artificial Intelligence elementary Problems.

CO3: Design Chatbot and Virtual Assistant.

Text Book(s)

1. John Zelle, JimLeisy, *Python Programming – An Introduction to computer science*, Ingram , 3rd edition, 2016

2. Ashok Namdev Kamthane and Amit Ashok Kamthane, *Programming and Problem Solving with Python*, McGraw Hill Education; First edition, 2017.

Reference Books

1. Mark Lutz, *Programming Python*, O'Reilly, 3rd Edition, 2006.

2. Wesley J Chun, *Core Python Programming*, Pearson,3rd edition, 2012

3. Brain Draper, Python Programming: A Compatible Guide for Beginners to Master and Become an Expert in python programming Language, CreateSpace Independent Publishing Platform, 2016.

4 Build an AI Assistant with Wolfram Alpha and Wikipedia in Python. *https://medium.com/@salisuwy/build-an-ai-assistant-with-wolfram-alpha-andwikipedia-in-python-d9bc8ac838fe*

Mode of Evaluation: Continuous Internal Evaluation and End Semester Examination

MCA II Year I Semester

Dept. of Computer Applications

MCA II Year I Semester

24MCAP109 FULL STACK WEB DEVELOPMENT

L	T	P	C
3	1	0	4

Pre-requisite 24MCAP602

Course Description:

This course introduces web designing and development methodologies using the front-end development languages such as JavaScript and Mern Stack, React Js, Node JS and Express. Also, this course gives the idea of web development process using Mongo DB

Course Objectives:

This course enables students to

1. Understand JavaScript fundamentals and MERN architecture.
2. Apply React to build interactive UIs with components and state.
3. Analyze and develop REST APIs using Node.js and Express.
4. Implement MongoDB for data storage and retrieval.
5. Optimize full-stack application performance through best practices.

UNIT I JAVASCRIPT AND BASICS OF MERN STACK 12 hours

JavaScript Fundamentals - Objects - Generators, advanced iteration - Modules - DOM tree - Node properties - browser events - Event delegation - UI Events -Forms, controls - Document and resource loading - Mutation observer - Event loop: microtasks and macrotasks - MERN Components- Need for MERN - Server-Less Hello World - Server Setup - nvm - Node.js - npm

UNIT II REACT JS 12 hours

React Introduction - React ES6 - React Render HTML - React JSX - Components -React Classes - Composing Components - Passing Data - Dynamic Composition - React state - setting State - Async State Initialization - Event Handling Communicating from Child to Parent - Stateless Components - Designing components- React Forms - React CSS - React SaaS

UNIT III NODE.JS AND EXPRESS 12 hours

Node.js basics - Local and Export Modules - Node Package Manager - Node.js web server - Node.js File system - Node Inspector - Node.js EventEmitter - Frameworks for Node.js - Express.js Web App - Serving static Resource - Node.js Data Access - Express REST APIs - REST - Resource Based - HTTP Methods as Actions - JSON- Express - Routing - Handler Function - Middleware - The List API

UNIT IV MONGODB 12 hours

MongoDB - MongoDB Basics - Documents - Collections - Query Language - Installation - The mongo Shell - Schema Initialization - MongoDB Node.js Driver - Reading from MongoDB - Writing to MongoDB - CRUD operations - projections - Indexing - Aggregaton - Replication - Sharding - Creating backup – Deployment

UNIT V ADVANCED FEATURES 12 hours

Modularization and Webpack - Routing with React Router - Forms - More Filters in the List API - UI Components - Update API - Delete API - React-Bootstrap - Bootstrap Installation - Navigation - Table and Panel - Forms - Alerts - Modals -Server Rendering - Basic Server Rendering - Handling State - Pagination - Session Handling

Course Outcomes:

Upon successful completion of the course, students will be able to

- **CO1:** Explain core JavaScript concepts and MERN stack architecture
- **CO2:** Develop interactive web interfaces using React components and state management
- **CO3:** Design and implement RESTful APIs using Node.js and Express
- **CO4:** Perform database operations using MongoDB's document-based structure
- **CO5:** Analyze and optimize performance in full-stack MERN applications

Text Books

1. Jon Duckett, "JavaScript & jQuery: Interactive Front-End Web Development", Wiley, 2014.

2. Vasan Subramanian, Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node. Apress, 2019.

Reference Books

1. Mastering Javascript, VedAntani, PACKT publishing, 2019

2. Node JS Web Development, David Herron, PACKT publishing, 2021

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

Dept. of Computer Applications

MCA II Year I Semester

24MCAP110 DATA SCIENCE

L	T	P	C
3	1	0	4

Pre-requisite **24MATP101**

Course Description:

This syllabus provides a comprehensive introduction to Data Science, covering fundamental concepts, tools, and applications. Students will learn data cleaning, analysis, and visualization techniques. The course teaches machine learning models such as regression, classification, and clustering, along with ways to evaluate their performance. It also includes advanced visualization tools like Tableau and Power BI, as well as real-world applications.

Course Objectives:

This course enables students to

1. Define data science concepts and demonstrate Python libraries for data analysis
2. Apply EDA techniques and perform data preprocessing techniques
3. Compare machine learning models and evaluate their performance
4. Design effective visualizations using Tableau/Power BI and interpret geospatial data.
5. Predict trends using time series analysis and relate data science to real-world applications.

UNIT I INTRODUCTION TO DATA SCIENCE 12 Hours

Introduction to Data Science- Definition and scope of data science - Impact of Data Science – Data science workflow and Life Cycle - Data Science Toolkit- Data sources and types - Data collection methods- Introduction to Python libraries for data analysis (NumPy, Pandas, matplotlib, seaborn, Plotly).

UNIT II DATA PREPROCESSING TECHNIQUES 12 Hours

Exploratory Data Analysis (EDA): Descriptive statistics - Data visualization for EDA – Handling missing data, duplicate and outliers - Joining and merging datasets -Data aggregation and summarization.

Data Wrangling and Feature Engineering: Data transformation and reshaping – Scaling-Encoding-Feature extraction and selection - Dimensionality Reduction (PCA and LDA).

UNIT III MODEL EVALUATION & SELECTION 12 Hours

Machine Learning Models: Regression algorithms (linear regression, decision trees) – Regression Model Evaluation metrics- Classification algorithms (logistic regression, random forests)- Classification Model Evaluation metrics - Clustering algorithms (k-means, hierarchical clustering)- Clustering Model Evaluation metrics -Model Selection.

UNIT IV DATA VISUALIZATION 12 Hours

Perceptual and cognitive principles - best practices and design considerations- Data Visualization Tools: **Power BI:** Data Import & Transformation, Data Modelling, Visualizations & Dashboards **Tableau:** Overview & Architecture-tableau interface- Charts-reports-Filters-Dashboard. Geospatial Visualization: Mapping techniques and libraries (GeoPandas) – Visualizing geographic data and spatial relationships.

UNIT V APPLICATIONS 12 Hours

Time Series Analysis and Forecasting: Introduction to time series data – Time Series Forecasting methods (Naïve, AR, MA, ARMA, ARIMA, Exponential Smoothing).

Application of data science: Healthcare-Predictive-Analytics, Retail-Recommendation-Systems, Social-Media-Sentiment-Analysis.

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Understand Data Science fundamentals and use Python libraries for basic data analysis.
CO2: Execute data cleaning, transformation, and feature engineering to optimize dataset quality.
CO3: Build and assess regression, classification, and clustering models.
CO4: Design interactive dashboards and geospatial visualizations.
CO5: Analyze time series data and evaluate real-world applications.

Text Book(s)

1. VanderPlas, Jake. Python data science handbook: Essential tools for working with data. " O'Reilly Media, Inc.", 2016.
2. Murray, Scott. Interactive data visualization for the web: An introduction to designing with D3. " O'Reilly Media, Inc.", 2017.
3. Hyndman, Rob J., and George Athanasopoulos. Forecasting: principles and practice. OTexts, 2018.

Reference Books

1. Wagh, Sanjeev J., Manisha S. Bhende, and Anuradha D. Thakare. Fundamentals of data science. Chapman and Hall/CRC, 2021.
2. Grus, Joel. Data science from scratch: first principles with python. O'Reilly Media, 2019.
3. Larose, Chantal D., and Daniel T. Larose. Data science using Python and R. John Wiley & Sons, 2019.

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

Dept. of Computer Applications

MCA II Year I Semester

24MCAP207 FULL STACK WEB DEVELOPMENT LABORATORY

L	T	P	C
0	0	3	1.5

Pre-requisite 24MCAP602

Course Description:

This course offers a solid foundation in Full Stack Web Development, covering both front-end and back-end technologies. Students will learn essential tools, frameworks, and best practices for building responsive, scalable web applications. Through hands-on projects, they will gain experience in UI design, server-side programming, database integration, and deployment.

Course Objectives:

Students will be able to:

1. Acquire knowledge on web designing using front end tools
2. Implement coding using scripting languages
3. Integrate applications using React and MongoDB

List of Experiments:

1. Write a program to create a simple webpage using HTML.
2. Write a program to build a Chat module using HTML CSS and JavaScript?
3. Build a simple form with validation (check if email is valid before submission).
4. Write a program to create an array of 5 cities and perform the following operations: Log the total number of cities. Add a new city at the end. Remove the first city. Find and log the index of a specific city.
5. Write a program to create a simple calculator Application using React JS
6. Write a program to create a Simple Login form using React JS
7. Write a program to create a counter using ReactJS
8. Create a NodeJS server that serves static HTML and CSS files to the user without using Express.
9. Create a NodeJS server using Express that stores data from a form as a JSON file and displays it in another page. The redirect page should be prepared using Handlebars.
10. Install the MongoDB driver for Node.js. Create a NodeJS server using Express that creates, reads, updates and deletes students' details and stores them in MongoDB database. The information about the user should be obtained from a HTML form
11. Create a simple Sign up and Login mechanism and authenticate the user using cookies. The user information can be stored in MongoDB and the server should be built using NodeJS

and Express Framework.

12. Create an Express API with a /products endpoint to fetch all products. Use fetch in React to call the /products endpoint and display the list of products. Add a POST /products
13. Write a program to create a voting application using ReactJS
14. Develop a leave management system for an organization where users can apply different types of leaves such as casual leave and medical leave. They also can view the available number of days using react application.
15. Create are act application for an online store which consist of registration, login, product information pages and implement routing to navigate through these pages.

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1. Implement webpages using Java script
CO2. Execute the different application using Node JS and Express
CO3. Create Node.JS script with MongoDB

Text Book(s)

1. Deitel, Paul J., and Harvey M. Deitel. *JavaScript for Programmers*. Deitel & Associates, Inc.2024.
2. Subramanian, Vasan. *Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node*. Apress, 2019.

Reference Books

1. VedAntani, Akshat. *Mastering JavaScript*. PACKT Publishing, 2019.

2. Herron, David. *Node.js Web Development*. 6th ed., PACKT Publishing, 2021.

Mode of Evaluation: Continuous Internal Evaluation and End Semester Examination

Dept. of Computer Applications

MCA II Year I Semester

24MCAP208 DATA SCIENCE USING PYTHON LABORATORY

L	T	P	C
0	0	3	1.5

Pre-requisite **24MATP101**

Course Description:

This syllabus provides a comprehensive introduction to Data Science, covering fundamental concepts, tools, and applications. Students will learn data cleaning, analysis, and visualization techniques. The course teaches machine learning models such as regression, classification, and clustering, along with ways to evaluate their performance. It also includes advanced visualization tools like Tableau and Power BI, as well as real-world applications.

Course Objectives:

This course enables students to

1. Use Python libraries for data manipulation, visualization, and analysis.
2. Apply EDA, preprocessing, and model evaluation techniques in machine learning.
3. Create interactive visualizations, analyze geospatial and time series data for real-world insights.

LIST OF EXPERIMENTS

1. Write a program to generate synthetic data (e.g., sales trends) using NumPy arrays and compute descriptive statistics on the dataset.
2. Load a dataset (e.g., Iris Dataset), clean column names, and filter rows based on conditions.
3. Create a dataset and visualize it with different customization options, such as colors, labels, titles, and annotations, to enhance the visual appeal of the following plots by using matplotlib, seaborn, and plotly library.
   (i) bar chart (ii) pie chart (iii) scatter plot and (iv) line plot (v) violin
4. Write a Python program to perform EDA by detecting and handling missing values, outliers, and inconsistencies in a DataFrame.
5. Merge two datasets using inner/left joins and concatenate a third dataset vertically using Pandas merge/join, pivot_table and groupby operations.
   Data Sets: orders.csv (order_id, customer_id, order_date), customers.csv (customer_id, name, email), orders_new.csv (order_id, customer_id, order_date, product_id, quantity, total_price, payment_method).
6. Preprocess the Titanic dataset by scaling numeric features (Age, Fare), encoding categorical variables (Sex, Embarked), and creating new features (FamilySize, Title from Name).
7. Apply PCA and LDA to reduce the Iris dataset to 2 components, visualize the results, and compare their cluster separability.
8. Implement simple and multiple linear regression model to predict a continuous target variable based on input features by using python.

9. Implement logistic regression model to train and evaluate on a labelled dataset by using python.
10. Implement k means clustering algorithm for the given data set using python.
11. Create a Tableau dashboard with a map, bar chart, filters, and trend line using the Superstore dataset.
12. Visualize Indian state populations and major city locations on an interactive map using GeoPandas in Python.
13. Implement the visualization of time series forecasting methods by using Air Passengers dataset: evaluate and plot results.
14. Minor Projects (Self-Study)
    (i) Diabetes risk prediction model (Data Set: Pima Indians dataset)
    (ii) Movie recommendation system (Data Set: MovieLens dataset)
    (iii)Smart Home Energy Consumption Predictor (Dataset: UK Power Networks)

Additional Exercises

1. Implement hierarchical clustering algorithm for the given data set using python.
2. Using the Superstore data Build a Power BI/ Excel report with: A slicer for region, Matrix of sales by sub-category, Tooltip showing profit ratio.

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Apply Data Science Fundamentals
CO2: Develop and Evaluate Machine Learning Models
CO3: Create Visualizations and Solve Real-World Problems.

Text Book(s)

1. VanderPlas, Jake. *Python data science handbook: Essential tools for working with data*. " O'Reilly Media, Inc.", 2016.

2. Murray, Scott. *Interactive data visualization for the web: An introduction to designing with D3*. " O'Reilly Media, Inc.", 2017.

3. Hyndman, Rob J., and George Athanasopoulos. *Forecasting: principles and practice*. OTexts, 2018.

Reference Books

1. Grus, Joel. *Data science from scratch: first principles with python*. O'Reilly Media, 2019.

2. Larose, Chantal D., and Daniel T. Larose. *Data science using Python and R*. John Wiley & Sons, 2019.

Mode of Evaluation: Continuous Internal Evaluation and End Semester Examination

OPEN ELECTIVES

Dept. of Computer Applications

MCA II Year I Semester

24HUMP301 INDIAN KNOWLEDGE SYSTEM

L	T	P	C
3	0	0	3

Pre-requisite NIL

Course Description:

This course explores Bhāratīya (Indian) civilization's contributions to knowledge systems, science, arts, and philosophy. It covers the Saraswatī-Sindhu Civilization, Vedic literature, ancient universities (Takṣaśilā, Nālandā), and scholars like Āryabhaṭa, Caraka, and Pāṇini. Topics include mathematics (zero, pi), astronomy, Āyurveda, metallurgy, and Yoga, along with traditional arts, architecture, and ecology. By the end, students will gain a holistic understanding of Bhāratīya civilization's intellectual, scientific, and cultural legacy and its relevance today.

Course Objectives:

This course enables students to

1. Explain the significance of Bhāratīya heritage in shaping national identity.
2. Compare traditional knowledge (e.g., Vedic math, metallurgy) with contemporary science.
3. Investigate gaps in current research on Bhāratīya sciences (e.g., astronomy, ecology).
4. Assess the feasibility of integrating traditional practices (e.g., organic farming, yoga therapy) into modern industries.
5. Analyse market potential for traditional arts (e.g., classical music, handicrafts) in global economies.

UNIT I BHĀRATĪYA CIVILIZATION AND DEVELOPMENT OF KNOWLEDGE SYSTEM — 9 Hours

Genesis of the land, Antiquity of civilization, On the Trail of the Lost River, Discovery of the Saraswatī River, the Saraswatī-Sindhu Civilization, Traditional Knowledge System, The Vedas, Main Schools of Philosophy (6+3), Ancient Education System, the Takṣaśilā University, the Nālandā University, Alumni, Knowledge Export from Bhārata.

UNIT II ARTS, LITERATURE, AND SCHOLARS — 9 Hours

Art, Music, and Dance, Naṭarāja– A Masterpiece of Bhāratīya Art, Literature, Life and works of Agastya, Lopāmudrā, Ghoṣā, Vālmīki, Patañjali, Vedavyāsa, Yājňavalkya, Gārgī, Maitreyī, Bodhāyana, Caraka, Suśruta, Jīvaka, Nāgārjuna, Kaṇāda, Patañjali, Kauṭīlya, Pāṇini, Thiruvalluvar,
Āryabhaṭa, Varāhamihira, Ādi Śaṅkarācārya, Bhāskarācārya, Mādhavācārya.

UNIT III SCIENCE, ASTRONOMY, AND MATHEMATICS — 9 Hours

Concept of Matter, Life and Universe, Gravity, Sage Agastya's Model of Battery, Velocity of Light, Vimāna: Aeronautics, Vedic Cosmology and Modern Concepts, Bhāratīya Kāla-gaṇanā, Kerala School for Mathematics and Astronomy, History and Culture of Astronomy, Sun, Earth, Moon, and Eclipses, Earth is Spherical and Rotation of Earth, Archaeostronomy; Concepts of Zero and Pi, Number System, Pythagoras Theorem, and Vedic Mathematics..

UNIT IV ENGINEERING, TECHNOLOGY, AND ARCHITECTURE 9 Hours

Pre-Harappan and Sindhu Valley Civilization, Laboratory and Apparatus, Juices, Dyes, Paints and Cements, Glass and Pottery, Metallurgy, Engineering Science and Technology in the Vedic Age and Post-Vedic Records, Iron Pillar of Delhi, Rakhigarhi, Mehrgarh, Sindhu Valley Civilization, Marine Technology, and Bet–Dwārkā.

UNIT V LIFE, ENVIRONMENT, AND HEALTH 9 Hours

Ethnic Studies, Life Science in Plants, Anatomy, Physiology, Agriculture, Ecology and Environment, Āyurveda, Integrated Approach to Healthcare, Medicine, Microbiology, Medicine, Surgery, and Yoga, etc.

Course Outcomes:

Upon successful completion of the course, students will be able to

- **CO1**: Explain the Vedic knowledge system, six schools of philosophy, and India's contributions to global knowledge transmission.
- **CO2**: Compare Bhāratīya artistic traditions (e.g., Naṭarāja sculpture) with global art forms.
- **CO3**: Assess ancient scientific concepts (gravity, Vedic cosmology, Kerala School) against modern theories.
- **CO4**: Evaluate the engineering marvels of Sindhu Valley cities (urban planning, materials science)
- **CO5**: Critique Āyurveda's holistic healthcare model and its relevance today.

Text Book(s)

1. Chauhan, Bhag Chand. *Iks: The Knowledge System of Bharata.* Garuda Prakashan, 2023.
2. Raha, Sibaji, et al. *History of Science in India*. Vol. 1, Part I & II, Vol. VIII, National Academy of Sciences, India and The Ramakrishna Mission Institute of Culture, Kolkata, 2014.

Reference Books

1. Kohle, Pradeep, et al., editors. *Pride of India—A Glimpse of India's Scientific Heritage.* Samskrit Bharati, 2006.
2. Verma, Keshav Dev. *Vedic Physics*. Motilal Banarsidass Publishers, 2012.
3. Soni, Suresh. *India's Glorious Scientific Tradition.* Ocean Books Pvt. Ltd., 2010.

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

Dept. of Computer Applications

Open Elective – II

24ECEP301 COMMUNITY RADIO TECHNOLOGY

L	T	P	C
3	0	0	3

Pre-requisite

Course Description:

This course offers a comprehensive exploration of Community Radio, from foundational concepts to practical implementation. It begins with an introduction to the principles of Community Radio and guides students through the process of establishing a Community Radio Station (CRS). Key topics include Studio Technology, Operations and Management along with detailed instruction in Audio Pre-Production and Post-production techniques. Students will also gain essential knowledge of Radio Transmission technology, including the setup of an FM transmitter. By the end of the course, students will have a thorough understanding of Community Radio principles and the Practical skills required to effectively operate a Community Radio station.

Course Objectives:

This course enables students to

1. Associate the concept of fundamentals in Community Radio in Local Communication and development.
2. Gain knowledge of Studio technology and operations including Soundboards, Microphones, Recording, scheduling, content creation, and team coordination.
3. Develop skills in Audio Pre-production and post-production such as recording, editing, and mixing audio content.
4. Categorize the Radio Transmission Technology comprising signal requirements aligning with factors affecting Coverage and Shadow Areas.
5. Explore Radio Transmission technology essentials and understand the technical aspects of setting up and maintaining an FM transmitter.

UNIT I COMMUNITY RADIO FUNDAMENTALS AND SETUP 9 hours

Introduction to Radio Broadcasting in India - Community Radio: Evolution - Community Radio Policy – Technical principles; Components of a CR Station - Radio Waves and Spectrum - Basics of Electricity - Power Backup and Voltage Stabilization

UNIT II STUDIO TECHNOLOGY & OPERATIONAL PRACTICES 9 hours

Basics of Sound - Analog and Digital Audio - Components of the Audio Chain - Studio Acoustics; Good Engineering Practices for Studio Setup - Studio Equipment: Preventive & Corrective Maintenance - Content Distribution: Alternative Mechanisms

UNIT III AUDIO PRE & POST PRODUCTION 9 hours

Audio Hardware and Field Recording – Microphones - Audio Cables and Connectors - Free and Open-Source Software - Telephony for Radio - Landline Systems - GSM/CDMA - Voice Over Internet Protocol (VoIP); Sound Recording and Editing - Mixing and Mastering - File Formats and Compression Transmission - Storing and Retrieval

UNIT IV RADIO TRANSMISSION TECHNOLOGY 9 hours

Transmission Chain Overview – Live and Pre-recorded Transmission - Principles of FM Transmission – FM Transmitter console- Antenna System - Types of Mast/Towers - Layers of Atmosphere and Radio Wave Propagation - Factors Affecting Coverage and Shadow Areas - Signal Requirements and Coverage Planning Parameters

UNIT V FM TRANSMITTER SETUP 9 hours

Connecting Audio Feed to the Transmitter - Back Panel Connectors - Mounting and Connecting the Transmitter - Probable Causes of Failure of Transmitters - Fault Diagnostics and Corrective Maintenance - Transmitter Operation and Upkeep Issues

Course Outcomes:

Upon the completion of the course, Student will be able to

CO1: Interpret the evolution with a framework of Community Radio with Technical Principles and essential Radio Spectrums.

CO2: Apply Studio Technology and Operational practices with the components of the Audio Chain including Acoustics and Equipment maintenance.

CO3: Conduct Comprehensive Audio Pre & Post-production to operate field Recordings with Hardware and Open-source software to manage sound recording, editing, mixing, mastering, and file compression.

CO4: Infer the principles of FM transmission, Antenna systems, Radio wave propagation and factors affecting coverage.

CO5: Demonstrate knowledge of the connecting audio feeds for Transmitter setup by resolving operational Issues with corrective maintenance.

Text Book(s)

1. Pooja Murada R. Sreedher, “Community Radio in India”, Aakar Books, 2019.
2. Prof. Raj Misra , “Community Radio By the people, For the People”, Orange Books Publication, 2022
   Fraser, Colin, and Sonia Restrepo Estrada, “Community radio handbook”. Paris: Unesco, 2001.

Reference Books

1. Juliet Fox, “Community Radio’s Amplification of Communication for Social Change”, 7th Edition, Palgrave Macmillan (Springer International Publishing.), 2019.
2. Kanchan K. Malik, Vinod Pavarala, “Community Radio in South Asia: Reclaiming the Airwaves”, Routledge India, 2020.
3. Vinod Pavarala and Kanchan K. Malik, “Other voices: the struggle for community radio in India”,Sage Publications India Pvt Ltd, 2007.
4. Michael C. Keith, “The Radio Station: Broadcast, Satellite & Internet”, 7th Edition, Focal Press (Elsevier Inc.), 2007.
5. “Certificate in Community Radio Technology (CCRT)”
   https://www.cemca.org/resources/certificate-community-radio-technology-ccrt-0

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

Dept. of Computer Applications

Open Elective – II

24MEP301 AUTOMATION AND ROBOTICS

L	T	P	C
3	0	0	3

Pre-requisite: NIL

Course Objectives:

This course is designed to:

1. To introduce the fundamental principles of automation and robotic systems.
2. To provide a detailed understanding of sensors, actuators, and embedded hardware used in robotic automation.
3. To explore the role of IoT in enhancing robotic applications with cloud and network-based control.
4. To study different robotic control strategies including path planning and wireless communication.
5. To analyze application domains of automation and robotics in industries, smart environments, and assistive technologies.

UNIT I INTRODUCTION TO AUTOMATION AND ROBOTICS 9 hours

Historical evolution of automation and robotics, Elements of automation systems: sensors, actuators, controllers, communication. Types and classifications of robots: industrial, mobile, service, collaborative. Levels of automation and hierarchy in automated systems. Role of AI and ML in modern robotics (overview).

UNIT II SENSORS, ACTUATORS, AND MICROCONTROLLER SYSTEMS 9 hours

Overview of sensing technologies: ultrasonic, IR, LDR, temperature, gas, motion, force. Actuation mechanisms: electrical, hydraulic, pneumatic; types of motors and control. Interfacing sensors with microcontrollers (theory): ADC, PWM, GPIO. Microcontroller platforms overview: Arduino, ESP32, Raspberry Pi (architecture, features). Importance of timing, signal conditioning, and data acquisition systems.

UNIT III IOT INTEGRATION IN ROBOTICS 9 hours

Architecture of IoT: perception, network, and application layers. IoT protocols: MQTT, CoAP, HTTP/REST, WebSocket – comparison and use cases. Cloud platforms for IoT-enabled robotics: ThingSpeak, Blynk, Google Firebase (functional overview). Edge vs Cloud computing in robotics. Data logging, remote control, and device provisioning (theoretical framework).

UNIT IV ROBOTIC CONTROL, NAVIGATION, AND COMMUNICATION 9 hours

Basics of kinematics and control in mobile robots. Introduction to navigation: line following, obstacle avoidance, indoor localization. Wireless communication methods: Wi-Fi, BLE, LoRa, Zigbee – use in robotic control. Concepts of feedback, open-loop vs closed-loop control, PID (conceptual explanation). Control architectures: centralized, distributed, cloud-based.

UNIT V APPLICATION DOMAINS AND FUTURE TRENDS 9 hours

Industrial automation: pick-and-place robots, robotic arms, AGVs. Smart environments: home automation, surveillance bots, service robots. Assistive and healthcare robotics: autonomous wheelchairs, rehabilitation aids. Emerging trends: swarm robotics, human-robot collaboration (HRC), soft robotics. Ethical considerations, safety standards, and legal frameworks in automation.

Course Outcomes:

At the end of the course, the student will be able to:

CO1: Define core automation and robotics terminologies and categorize various robotic systems. (L1)

CO2: Explain the function and selection of sensors, actuators, and embedded controllers in robotic systems. (L2)

CO3: Interpret IoT architectures and demonstrate their integration with robotic automation. (L3)

CO4: Analyze wireless control, navigation algorithms, and data flow in IoT-based robotic systems. (L3)

CO5: Evaluate the suitability of automation and robotics technologies in various real-world application domains. (L3)

Text Books:

1. Rajesh Singh, Anita Gehlot, "Internet of Things and Automation with ESP32", Wiley India, 2023
2. Debasis Samanta, "Fundamentals of Robotics and Intelligent Systems", Cengage Learning, 2024
3. S. R. Jankiraman, "Embedded Systems and Robotics with Open Source Tools", Universities Press, 2022

Reference Books:

1. Amit Kumar & Mahalik N. P., "IoT and Robotics Applications: Design, Development and Deployment", CRC Press, 2024
2. Jonathan Valvano, "Embedded Systems: Introduction to ARM Cortex-M Microcontrollers", CreateSpace, 2023
3. Rafael C. Gonzalez, "Robotics and Control: Principles and Practice", McGraw-Hill, 2022

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination.

Dept. of Computer Applications

Open Elective - II

24MBAP301 E-COMMERCE AND DIGITAL MARKETS

L	T	P	C
3	0	0	3

Pre-requisite None

Course Description:

The course focusses on e-commerce and is divided into five sections to deliver key aspects of E – commerce such as business models, enablers, and supply chain, Social, Political & Ethical Issues of E-Commerce, E-commerce systems and Digital Markets. Each section delves deeply into various intricacies of doing business over the electronic media.

Course Objectives:

Students will be able to:

1. Introduce the concept of e-business and the business models used in e-commerce.
2. Elucidate about the e-commerce enablers and infrastructure.
3. Enable the students to learn about supply chain management used by e-commerce players.
4. Analyze the socio, political and ethical issues in e-commerce.
5. Develop an insight into e-markets and e-commerce systems.

UNIT I E – COMMERCE BUSINESS MODELS 9 hours

Introduction E-Business - Origin and Need of E-Commerce, – E– commerce v/s Traditional Commerce Factors affecting E -Commerce, Business dimension and technological dimension of E-Commerce, E-Commerce frame work. The Revolution Continues, E-commerce Business Models and Concepts, B2C business models, B2Bmodels, B2G, G2C, Business models for emerging Ecommerce area – customer to customer businessmodel, P2P business model, M-commerce models. IT in business – functional business systems – cross-functional enterprise systems and applications – e- Business models - Enterprise e-Business systems

UNIT II E – COMMERCE ENABLERS 9 hours

E- Commerce enablers, internet and its impact on business strategy Pre and Post Covid-19 Pandemic - industry structure, industry value chain, firm value chain. E-commerce Infrastructure: The Internet, Web, and Mobile Platform.

UNIT III SUPPLY CHAIN MANAGEMENT IN E – COMMERCE 9 hours

B2B E-commerce: Supply Chain Management and Collaborative Commerce. – Introduction to Customer relationship management (CRM) -Building an E-commerce Presence: Web Sites, Mobile Sites, and Apps, E-commerce Marketing Communications -Pre and Post Covid-19 Pandemic

UNIT IV SOCIAL, POLITICAL, AND ETHICAL ISSUES 9 hours

Ethical, Social, and Political Issues in E-commerce, Online Retailing and Services, Online Content and Media, Social Networks, Auctions, and Portals.

UNIT V E-MARKETS **9 hours**

Factions, e-Markets Vs Traditional Market, e-Markets Success factors, e-Market Technology Solutions. E-Procurements: The purchasing process, Developments in IT purchasing, e-Procurement- Models, e- procurement- Solutions – E-Commerce systems: E-Commerce systems – Essential e- Commerce processes – electronic payment processes - e-Commerce application trends – Web store requirements – clicks-and-bricks in e-Commerce.

Course Outcomes:

After Completion of the Course, Students will be able to:

CO1: Understand the concepts of e-business and the business models used in e-commerce.

CO2: Learn about the e-commerce enablers and infrastructure.

CO3: Develop an insight into supply chain management.

CO4: Analyze into the socio, political and ethical issues in e-commerce.

CO5: Develop an understanding of e-markets and e-commerce payment systems.

Textbook(s)

1. Laudon Kenneth C., E-Commerce: Business, Technology, Society, prentice Hall of India, 15th edition, 2019.
2. Bhanver, J.,& Bhanver, K., Click!: The Amazing Story of India's E-commerce Boom and Where it's Headed.: Hachette, 2017.

Reference Books

1. Bhaskar, B. Electronic commerce: Framework, technologies and applications, 3rd edition New Delhi: Tata McGraw Hill Education, 2009.
2. Erisman, P., Six Billion Shoppers: The Companies Winning the Global ECommerce Boom. Macmillan, 2017.
3. Kalakota, R., & Whinston, A. B., Electronic commerce: A manager's guide. New Delhi: Pearson Education, 2009.

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

DISCIPLINE ELECTIVES

Discipline Elective – I

24MCAP401 MACHINE LEARNING

L	T	P	C
4	0	0	4

Pre-requisite 24MATP101, 24MCAP102

Course Description:

This course provides a concise introduction to the fundamental concepts in machine learning and popular machine learning algorithms. This will cover the standard and most popular supervised learning algorithms including linear regression, logistic regression, decision trees, k-nearest neighbor, an introduction to Bayesian learning and the naïve Bayes algorithm, support vector machines and neural networks.

Course Objectives:

This course enables students to

1. Understand the basic concepts and applications of Machine Learning.
2. Explain the predictive models used in machine learning applications.
3. Analyse the classification model respect to real time problems.
4. Recognize the clustering models and Dimensional Reduction Techniques.
5. Experiment the hybrid models intend to increase the accuracy of ML based applications.

UNIT I INTRODUCTION **12 hours**

Learning – Designing a Learning System – Traditional Learning vs Machine Learning – Various types of Machine Learning – Machine Learning workflow – Machine Learning issues and challenges – Machine Learning Applications in real world problems.

UNIT II PREDICTIVE MODELS **12 hours**

Linear Regression Models – Least Squares, single multiple variables, Bayesian Linear Regression, mediant decent, linear classification Models– Discriminant Function –Perception Algorithm, Probabilistic Discriminative Model–Logistic Regressive

UNIT III CLASSIFICATION MODELS **12 hours**

Introduction – Different types of classifiers: Naive Bayes — Logistic Regression — Decision Tree - Support Vector Machine - K-Nearest Neighbour- Evaluation metrics for supervised learning – Issues and challenges – applications.

UNIT IV CLUSTERING MODELS **12 hours**

Partitioning based clustering – Hierarchical based clustering – Density based clustering – Grid based clustering – Mixture Models and EM Algorithm -Evaluation metrics for clustering models - Dimensional Reduction Techniques: PCA – ICA – LDA.

UNIT V ENSEMBLE LEARNING 12 hours

Boosting – AdaBoost Algorithm – Bagging – Random Forest – NoFree-Lunch Theorem – XGBoost Algorithm – Stacking – Voting – Ensemble Diversity – Error Decomposition – Diversity Measures – Evaluating Ensembles of Classifiers.

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Understand the real-world applications that needs machine learning based solutions.
CO2: Examine the predictive machine learning Models.
CO3: Experiment the Classification algorithms in appropriate real-time applications
CO4: Implement the Clustering algorithms for real world problems.
CO5: Apply Ensemble Learning Modes for Real Time Predictions

Text Book(s)

1. Aurelien Geron, Hands on Machine Learning with Scikit -Learn , Keras and Tensor Flow Concepts, Tools and Techniques to build intelligent Systems,2nd Edition by O'Reilly Media Publishers, 2019.
2. Tom Michel , "Machine Learning", Mc.Graw Hill, Indian Edition, 1997.

Reference Books

1. Ethem Alpaydin, Introduction to Machine Learning, 4th edition, MIT Press 2020.

2. Bishop, Christopher M., Pattern Recognition and Machine Learning. Springer-Verlag, 2006.

3. Zhi-Hua Zhou, Ensemble Methods: Foundations and Algorithms, CRC Press, 2012

4. Lior Rokach, Ensemble Learning: Pattern Classification using Ensemble Methods, 2nd ed., World Scientific, 2019.
5. Tom M. Mitchell, Machine Learning, McGraw-Hill Education (India) Private Limited, 2013

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

Dept. of Computer Applications

Discipline Elective – I

24MCAP408 CYBER SECURITY

L	T	P	C
4	0	0	4

Pre-requisite Nil

Course Description:

Cyber-attacks, types of cybercrime, cyber laws and also how to protect them self and ultimately the entire Internet community from such attacks. Interpret and forensically investigate security incidents. Apply policies and procedures to manage Privacy issues.

Course Objectives:

This course enables students to

1. Understand cyber security concepts.
2. Learn cyber laws & concepts of cyber forensics.
3. Study the defensive techniques and privacy methods.
4. Understand cybercrime framework and methodologies
5. Learn firewall methods and use case.

UNIT I INTRODUCTION 12 hours

Introduction to Cyber Security: Basic Cyber Security Concepts, Layers of security, Vulnerability, threat, Harmful acts, Internet Governance – Challenges and Constraints, Computer Criminals, CIA Triad, Assets and Threat, Motive of attackers, Types of Attacks, Cyber Threats-Cyber Warfare, Cyber Crime, Cyber terrorism, Cyber Espionage, Comprehensive Cyber Security Policy.

UNIT II CYBERSPACE 12 hours

Cyberspace and Law**:** Introduction, Cyber Security Regulations, Roles of International Law. The INDIAN Cyberspace, National Cyber Security Policy, Cyber Forensics: Historical background of Cyber forensics - The Need for Computer Forensics, Cyber Forensics and Digital evidence, Forensics Analysis of Email, Digital Forensics Lifecycle, Forensics Investigation, Challenges in Computer Forensics.

UNIT III PRIVACY 12 hours

Privacy Concepts - Privacy principles and policies - Authentication and Privacy - Privacy on the Web - Email-Security - Privacy Impacts of Emerging technologies: Electronic Voting - Cyber Warfare - Privacy Issue, Data Privacy Attacks, Data linking and profiling, privacy policies and specifications, privacy policy languages - Privacy in different domains: Medical, Financial and Educational.

UNIT IV CYBERCRIME 12 hours

Introduction Cybercrime, Proliferation of Mobile and Wireless Devices, Trends in Mobility, Credit card Frauds in Mobile and Wireless Computing Era, Security Challenges Posed by Mobile Devices, Registry Settings for Mobile Devices, Authentication service Security, Attacks on Mobile device.

UNIT V FIREWALL 12 hours

Introduction to network Firewall - Design of Firewalls - Types of Firewalls - Comparison of Firewall types - Example firewall configurations - Network Address Translation (NAT) - Network Defence tools. Firewalls and Packet Filters - Firewall policy - Stateless Vs Stateful Firewalls - Port Forwarding – Case study Firewall Implementation.

Course Outcomes:

CO1: Explain fundamentals of networks security, security architecture, threats and vulnerabilities
CO2: Classify cyber laws and various methodologies in cyber forensics.
CO3: Analyse the basics of cybercrime and computer forensics.
CO4: Differentiate the Privacy and Security issues.
CO5: Implement firewall case to enterprise network.

Text Book(s)

1. William Stallings, "Cryptography and Network Security - Principles and Practice", Seventh Edition, Pearson Education, 2017.
2. Nina Godbole, Sunit Belapure, “Cyber Security: Understanding Cybercrimes, Computer Forensics and Legal Perspectives”, First Edition, Wiley India, 2011.

Reference Books

1. Behrouz A. Ferouzan, Debdeep Mukhopadhyay, "Cryptography and Network Security", 3rd Edition, Tata Mc Graw Hill, 2015.
2. Charles Pfleeger, Shari Pfleeger, Jonathan Margulies, "Security in Computing", Fifth Edition, Prentice Hall, New Delhi, 2015.
3. David Kim, Michael G. Solomon, “Fundamentals of Information Systems Security”, Jones & Bartlett Learning Publishers, 2013.

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

Dept. of Computer Applications

Discipline Elective – I

24MCAP409 CLOUD COMPUTING

L	T	P	C
4	0	0	4

Pre-requisite None

Course Description:
This course describes what is Cloud and the various service delivery models of a Cloud computing architecture, followed by the Cloud management, Performance issues, Hard ware concepts, Storage, Security and Privacy issues.

Course Objectives:
Student will be able to

1. Understand the fundamental ideas behind Cloud Computing, the evolution of the paradigm, its applicability benefits, as well as current and future challenges.
2. Learn the basic ideas and principles in data centre design, cloud management techniques, and cloud software deployment considerations.
3. Study performance metrics across compute, network and storage resources with Cloud delivery and hosting models.
4. Focuses on how to create virtualized storage, access and provisioning Models
5. Understand the Security model in cloud computing.

UNIT I UNDERSTANDING CLOUD COMPUTING 12 hours

Cloud Computing – History of Cloud Computing – Cloud Architecture – Cloud Storage – Why Cloud Computing Matters – Advantages of Cloud Computing – Disadvantages of Cloud Computing – Cloud Services

UNIT II DEVELOPING CLOUD SERVICES 12 hours

Types of Cloud Service Development – Software as a Service – Platform as a Service – Infrastructure as a Service (IaaS)-Web Services – On-Demand Computing – Discovering Cloud Services Development Services and Tools – Amazon Ec2 – Google App Engine – IBM Clouds.

UNIT III CLOUD COMPUTING FOR EVERYONE 12 hours

Centralizing Email Communications – Cloud Deployment Models- Collaborating on Schedules – Collaborating on To-Do Lists – Cloud Computing for the Community – Collaborating on Group Projects and Events.

UNIT IV PROGRAMMING MODEL 12 hours

Parallel and Distributed Programming Paradigms – Map Reduce, Twister and Iterative Map Reduce – Hadoop Library from Apache – Mapping Applications - Programming Support - Google App Engine, Amazon AWS - Cloud Software Environments -Eucalyptus, Open Nebula, Open Stack, Aneka, Cloud Sim.

UNIT V SECURITY IN THE CLOUD 12 hours

Security Overview - Cloud Security Challenges and Risks - Software-as-a-Service Security- Security Governance - Risk Management - Security Monitoring - Security Architecture Design - Data Security - Application Security - Virtual Machine Security - Identity Management and Access Control - Autonomic Security.

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Understand the cloud computing Paradigms and Architecture of Cloud.

CO2: Design different workflows according to cloud services requirements and apply real time applications.

CO3: Analyse the various programming schedule of cloud computing and apply them to solve Problems.

CO4: Implement the different Programming Model using cloud computing tools.

CO5: Assess the legal issues and security risks related cloud resources.

Text Book(s)

1. Michael Miller, "Cloud Computing: Web-Based Applications That Change the Way You Work and Collaborate Online", Que Publishing, August 2022.

2. Cloud Native Architectures, Tom Laszewski, Kamal Arora Erik Farr, Pivum Zonooz, Packt publishing, August 2018

Reference Books

1. Kai Hwang, Geoffrey C Fox, Jack G Dongarra, "Distributed and Cloud Computing, From Parallel Processing to the Internet of Things", Morgan Kaufmann Publishers, 2022.

2. John W.Rittinghouse and James F.Ransome, "Cloud Computing: Implementation, Management, and Security", CRC Press, 2019.

3. Toby Velte, Anthony Velte, Robert Elsenpeter, "Cloud Computing, A Practical Approach", TMH, 2023.

4. Kumar Saurabh, "Cloud Computing – insights into New-Era Infrastructure", Wiley India, 2019.

5. George Reese, "Cloud Application Architectures: Building Applications and Infrastructure in the Cloud" O'Reilly , 2022

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

Discipline Elective – II

24MCAP402 DEEP LEARNING

L	T	P	C
4	0	0	4

Pre-requisite 24MCAP107, 24MCAP401

Course Description:

This course introduces the fundamentals of deep learning, beginning with the basics of neural networks and progressing to advanced architectures like Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs). Learners will explore how these models are applied in real-world scenarios, particularly in computer vision and natural language processing (NLP). Through hands-on projects and practical examples, students will gain a solid understanding of deep learning concepts and techniques essential for modern AI applications.

Course Objectives:

This course enables students to

1. Understand the principles of neural networks and activation neuron functions
2. Identify key concepts and techniques in deep learning and compare their use cases
3. Explore the principles and applications of Convolutional Neural Networks (CNNs)
4. Describe the structure of Recurrent Neural Networks (RNNs)
5. Implement transfer learning techniques for NLP tasks using pre-trained models

UNIT I BASICS OF NEURAL NETWORKS 12 Hours

Neural Network - The Human Brain, Models of a Neuron- Deep Artificial Neural Networks – Loss Functions - Activation Neuron functions-sigmoid, tanh, ReLU, Softmax layer.

UNIT II FUNDAMENTALS OF DEEP LEARNING 12 Hours

Basics of Deep Learning- Origins of Deep Learning - Machine Learning Vs Deep Learning - Fundamental Deep Learning Algorithm – Deep Feedforward Networks - Deep neural networks (DNNs) - Multi-Layer Perceptrons – Applications of DNNs and MLP.

UNIT III CONVOLUTIONAL NEURAL NETWORK 12 Hours

Deep Convolutional Neural Network- Convolution, Filters, pooling, Channel Attention (in CNNs) Pre-trained CNN models- ResNet, AlexNet, VGG16, GoogleNet, U-Net – Applications of CNN and Pre-trained Models- Object Detection & Recognition, Medical Images.

UNIT IV RECURRENT NEURAL NETWORK 12 Hours

Introduction to RNNs – Types of RNNs - Unfolded RNNs, Seq2Seq RNNs – LSTM - Bidirectional LSTMs - Gated Recurrent Units (GRUs), Encoder/ Decoder Architectures - Applications of RNNs & LSTM - Video to Text with LSTM Models.

UNIT V 12 Hours

TRANSFER LEARNING FOR NLP

Introduction to Transfer Learning - Benefits & Use Cases - Traditional ML vs. Transfer Learning - Evolution: Word2Vec →ELMo → BERT/GPT → Modern LLMs - Vision Transformer (ViT**)** - **Case Study:** BERT for sentimental analysis, chatbot, text summarization. Case Study: Deep Learning for Diabetic Retinopathy Detection

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Understand the basics of neural networks and activation functions.
CO2: Explain deep learning concepts and apply them to different applications.
CO3: Analyze various pre-trained CNN models for real-life applications.
CO4: Apply RNNs to solve problems.
CO5: Implement and fine-tune BERT/GPT for custom tasks.

Text Book(s)

1. Navin Kumar Manaswi, Deep Learning with Applications Using Python, Chatbots and Face, Object, and Speech Recognition With TensorFlow and Keras, Apress, 2018
2. Dr. Emily Chen, *Transfer Learning for NLP: From Theory to Deployment with BERT, GPT, and Beyond*, O'Reilly Media, 2024

Reference Books

1. Umberto Michelucci “Applied Deep Learning. A Case-based Approach to Understanding Deep Neural Networks” Apress, 2018.
2. François Chollet “Deep Learning with python”, Manning Publications, 2017 (2nd Edition: 2021)

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

Dept. of Computer Applications

Discipline Elective – II

24MCAP403 DIGITAL FORENSICS

L	T	P	C
4	0	0	4

Pre-requisite 24MATP101, 24MCAP408

Course Description:

This course provides a comprehensive introduction to Digital Forensics, focusing on the investigation of cybercrimes through the identification, acquisition, analysis, and presentation of digital evidence. Students will gain practical skills in handling file systems, memory dumps, network traffic, mobile devices, and cloud environments using industry-relevant tools. Legal and ethical considerations in digital investigations will also be covered.

Course Objectives:

This course enables students to

1. Explain the fundamentals of digital forensics and its role in cybercrime investigations.
2. Apply techniques for identifying, collecting, and preserving digital evidence.
3. Analyze data from file systems, memory, and network sources using forensic tools.
4. Evaluate tools and methodologies for mobile and cloud forensic investigations.
5. Discuss legal, ethical, and procedural requirements for presenting digital evidence in court.

UNIT I INTRODUCTION TO DIGITAL FORENSICS 12 Hours

Definition and scope of digital forensics, Types of digital crimes and evidence, Phases of digital forensics process, Forensic readiness and incident response, Chain of custody and digital evidence handling, Overview of digital forensic tools (EnCase, FTK, Autopsy, etc.

UNIT II FILE SYSTEM AND DISK FORENSICS 12 Hours

Storage media and data acquisition (live and static), File system structures: FAT, NTFS, ext3/ext4, Data recovery techniques: deleted files, slack space, metadata, Disk imaging and cloning techniques, Hashing and integrity verification, Case study: Hard disk analysis using open-source tools

UNIT III MEMORY AND NETWORK FORENSICS 12 Hours

RAM acquisition and analysis, Volatile data collection techniques, Network traffic analysis (Wireshark, TCPDUMP), Detecting network intrusions, Log file analysis and event correlation, Case study: Memory dump investigation.

UNIT IV MOBILE AND CLOUD FORENSICS 12 Hours

Mobile device platforms (Android, iOS), Data acquisition techniques for smartphones, SIM and SD card forensics, Cloud storage investigation challenges, Forensic tools for mobile/cloud (Cellebrite, Oxygen Forensic Suite), Case study: Smartphone forensic investigation

UNIT V LEGAL, ETHICAL, AND REPORT WRITING ASPECTS 12 Hours

Legal issues in digital forensics: admissibility, privacy, jurisdiction, Cyber laws and regulations (Indian IT Act, GDPR basics), Ethics in digital investigation, Documentation and forensic report writing, Expert witness testimony and courtroom procedures, Case study: Legal Implications of Generative AI in Digital Forensics- Case Study: **Digital Forensics in Insider Data Theft Investigation**

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Describe the scope, phases, and tools used in digital forensic investigations.

CO2: Perform forensic analysis of file systems, data recovery, and evidence integrity verification.

CO3: Analyze memory dumps, log files, and network traffic to detect digital intrusions.

CO4: Acquire and examine digital evidence from mobile devices and cloud environments.

CO5: Assess legal, ethical, and procedural compliance in digital evidence handling and reporting.

Text Book(s)

1. Nelson, B., Phillips, A., and Steuart, C., *Guide to Computer Forensics and Investigations*, Cengage Learning, 6th Edition, 2020.
2. Sammons, J., *The Basics of Digital Forensics: The Primer for Getting Started in Digital Forensics*, Syngress, 2nd Edition, 2014.

Reference Books

1. Skulkin, O., Rachkov, Y., Afonin, A., and Kotenko, I., *Mobile Forensics – Advanced Investigative Strategies*, Packt Publishing, 2nd Edition, 2016.
2. Davidoff, S., and Ham, J., *Network Forensics: Tracking Hackers through Cyberspace*, Prentice Hall, 1st Edition, 2012.

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

Dept. of Computer Applications

Discipline Elective – II

24MCAP404 EDGE COMPUTING

L	T	P	C
4	0	0	4

Pre-requisite 24MCAP409

Course Description:

This course introduces the architecture, platforms, protocols, and operational challenges of Edge Computing as a contemporary distributed computing paradigm. It emphasizes the role of edge computing in enabling latency-sensitive applications, its integration with Internet of Things (IoT) ecosystems, and the execution of real-time data analytics at or near the network edge. Learners will explore how edge computing complements cloud services, enhances responsiveness, supports scalability, and addresses security and resource constraints in distributed environments.

Course Objectives:

This course enables students to:

1. Understand the need and evolution of edge computing.
2. Describe edge architectures, platforms, and technologies.
3. Analyze networking, latency, and security challenges at the edge.
4. Create edge-enabled systems integrated with cloud and IoT.
5. Evaluate real-world applications and emerging trends.

UNIT I INTRODUCTION TO EDGE COMPUTING 12 Hours

Definition and significance of edge computing, Evolution: From centralized to decentralized systems, Edge vs Fog vs Cloud, Applications: Smart cities, Healthcare, Retail, Industry 4.0.

UNIT II EDGE ARCHITECTURE AND TECHNOLOGIES 12 Hours

Components: Edge devices, Gateways, MEC, System architecture and edge nodes, Data lifecycle at the edge, Hardware platforms and micro data centers.

UNIT III EDGE PLATFORMS AND FRAMEWORKS 12 Hours

Platforms: AWS Greengrass, Azure IoT Edge, Google Coral, Edge containers and lightweight virtualization, Edge orchestration tools: K3s, Open Horizon, Edge data analytics and AI.

UNIT IV NETWORKING AND SECURITY AT THE EDGE 12 Hours

Protocols: MQTT, CoAP, 5G for edge, Challenges in latency, jitter, bandwidth. Security threats: device authentication, data integrity. Solutions: encryption, trust models, federated learning

.

UNIT V USE CASES AND FUTURE DIRECTIONS 12 Hours

Case studies: Autonomous vehicles, Edge surveillance, Smart factories, Edge-cloud collaboration, Standardization and industry trends, Future research directions in edge computing, case study: Role of Generative AI in Edge Computing

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Understand the principles and scope of edge computing.

CO2: Identify edge-specific architecture and technologies.

CO3: Design edge-based systems for latency-sensitive applications.

CO4: Evaluate networking and security strategies in edge deployments.

CO5: Analyze real-world edge applications and trends.

Text Book(s)

1. Lea, Perry. *Edge Computing: A Primer*. 1st ed., O'Reilly Media, 2022.

2. Mahmood, Zaigham. *Edge Computing: Models, Technologies and Applications*. 1st ed., Springer, 2021.

Reference Books

1. Satyanarayanan, Mahadev. "The Emergence of Edge Computing." *IEEE Computer*, vol. 50, no. 1, 2017.

2. Bonomi, Flavio, et al. "Fog Computing and Its Role in the Internet of Things." *Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing*, 2012.

3. AWS IoT Greengrass Documentation. Amazon Web Services, https://docs.aws.amazon.com/greengrass. Accessed June 2025.

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

Discipline Elective – III

24MCAP405 GENERATIVE AI

L	T	P	C
4	0	0	4

Pre-requisite 24MCAP401

Course Description:

This course introduces the fundamentals of Generative AI and large language models (LLMs), covering prompt engineering and popular AI platforms like OpenAI, Google Gemini, Meta's LLaMA, and Hugging Face. Learners will gain hands-on experience using APIs and tools to build real-world applications, preparing them for industry roles in the growing field of AI-driven solutions.

Course Objectives:

This course enables students to

1. Introduce learners to the core concepts, types, and real-world applications of Generative AI.
2. Develop the ability to craft effective prompts and utilize large language models (LLMs)
3. Use OpenAI APIs to access and deploy AI models in practical applications.
4. Explore the features, usage, and comparative strengths of Google Gemini and Meta's LLaMA models
5. Equip learners with the skills to use the Hugging Face platform for accessing models

UNIT I BASICS OF GENERATIVE AI 12 Hours

NLP Basics: **Tokenization & Embeddings**, Language Modelling - Transformers → GPT-4 - Generative AI - Generative AI technologies- Generative Adversarial Networks (GAN) - Deep Convolutional GAN - Encoder Models -RAG Models - Ethical issues and technical challenges.

UNIT II PROMPT ENGINEERING AND LLM 12 Hours

Introduction to Prompt Engineering- LLM with Prompt Engineering - Introduction to GPT models - Understanding of GPT-3 and GPT-4 – Large Language Models - LLMs in generating text, code, and image - Case Study: Creating a project with LLMS

UNIT III OPEN AI AND API 12 Hours

Introduction to Open AI - Utilizing OpenAI APIs - Setting up and authenticating API usage - Practical exercises using GPT-3/GPT-4 for text generation - Understanding DALL-E and its capabilities in image generation - generate images from textual descriptions

UNIT IV GOOGLE -GEMINI GEN API AND LLAMA 12 Hours

Google Gemini - API key for Gemini - Gemini API and accessing its features - different Gemini models- AI-powered chatbot using Gemini – LLaMA - Features and capabilities of LLaMA - Model Architecture of LLaMA - text generation using LLaMA

UNIT V HUGGING FACE ECOSYSTEM 12 Hours

Hugging Face ecosystem - Transformers library - Hugging Face Models and Tokenizers- pre-trained model to perform sentiment analysis - text generation, named entity recognition, summarization - Hugging Face models with web application. Case Study: **Generative AI for Content Creation at** digital media company

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Understand the fundamentals of NLP, tokenization, embeddings, and language modeling
CO2: Apply Large Language Models (LLMs) for text, code, and image generation
CO3: Recognize OpenAI's ecosystem and API functionalities
CO4: Explain LLaMA's model architecture and its text-generation capabilities
CO5: Analyze the effectiveness of different Hugging Face models for NLP tasks

Text Book(s)

1. *Lewis Tunstall, Leandro von Werra, Thomas Wolf*, **"Natural Language Processing with Transformers"** O'Reilly, 2022
2. *James Phoenix,* **Prompt Engineering for Generative AI,** O'Reilly, 2022

Reference Books

1. **Lewis Tunstall, Leandro von Werra, and Thomas Wolf.** *Natural Language Processing with Transformers: Building Language Applications with Hugging Face*. O'Reilly Media, 2022.
2. **Jason D. Brownlee.** *Generative Adversarial Networks with Python: Deep Learning Generative Models for Image Synthesis and Image Translation*. Machine Learning Mastery, 2019.

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

Discipline Elective – III

24MCAP406 PENETRATION TESTING

L	T	P	C
4	0	0	4

Pre-requisite 24MCAP408

Course Description:

This course introduces penetration testing (PenTest), a key area of cybersecurity focused on simulating attacks to identify system vulnerabilities and assess security. Students gain foundational knowledge in ethical hacking, threat modeling, vulnerability analysis, and exploitation techniques using real-world tools and frameworks, with a strong emphasis on legal and ethical practices.

Course Objectives:

This course enables students to

1. Understand the fundamentals of penetration testing and its role in cybersecurity.
2. Describe the legal/ethical requirements and outline procedures for scoping penetration tests.
3. Differentiate between vulnerability types and assess their potential impact on systems/networks.
4. Simulate controlled cyberattacks to evaluate security defenses.
5. Synthesize test results into professional reports and *design* actionable remediation plans.

UNIT I INTRODUCTION TO ETHICAL HACKING AND PENETRATION TESTING: 12 Hours

Ethical Hacking and Penetration Testing, Exploring Penetration Testing Methodologies, Environmental Considerations, Surveying Different Standards and Methodologies. **Planning and Scoping a Penetration Testing Assessment:** Comparing and Contrasting Governance, Risk, and Compliance Concepts. Regulatory Compliance Considerations. Regulations in the Financial Sector and Healthcare Sector. Payment Card Industry Data Security Standard (PCIDSS).

UNIT II INFORMATION GATHERING AND VULNERABILITY SCANNING 12 Hours

Passive Reconnaissance, Active Reconnaissance, Art of Performing Vulnerability Scans. Email Phishing, Spear Phishing, Whaling, Vishing, Short Message Service (SMS) Phishing, Universal Serial Bus (USB) Drop Key, Watering Hole Attacks, Social Engineering Tools, Social-Engineer Toolkit (SET), Browser Exploitation Framework (BeEF) Call Spoofing Tools

UNIT III EXPLOITING WIRED AND WIRELESS NETWORKS 12 Hours

Exploiting Network-Based Vulnerabilities: DNS Cache Poisoning, SNMP Exploits, SMTP Exploits, SMTP Open Relays. **Exploiting Application-Based Vulnerabilities:** SQL Injection Vulnerabilities, Database Fingerprinting, Booleans in SQL Injection Attacks, Command Injection Vulnerabilities, Lightweight Directory Access Protocol (LDAP) Injection, Session Hijacking, Kerberos Vulnerabilities, Cross-Site Scripting (XSS) Vulnerabilities.

UNIT IV CLOUD, MOBILE, AND IOT SECURITY 12 Hours

Cloud Technologies, Credential Harvesting, Privilege Escalation, Resource Exhaustion and DoS Attacks, Cloud Malware Injection Attacks, Tools and Software Development Kits (SDKs),

Attacking Internet of Things (IoT) Devices, Exploiting Virtual Machines. **Performing Post-Exploitation Techniques:** Command and Control (C2) Utilities, Scheduled Jobs and Tasks, Detection Avoidance, and Enumeration, Post-Exploitation Scanning, Blood Hound, Windows Management Instrumentation for Post-Exploitation Tasks.

UNIT V REPORTING AND COMMUNICATION 12 Hours

Written Reports, Report Contents, Common Themes/Root Causes, Technical Controls Administrative Controls, Importance of Communication during the Penetration Testing Process Communication Triggers, Reasons for Communication, Post-Engagement Cleanup. Tools and Code Analysis: Logic Constructs, Data Structures, Analysis of Scripts and Code Samples for Use in Penetration Testing, Understanding the Different Use Cases of Penetration Testing Tools and Analyzing Exploit Code, Penetration Testing–Focused Linux Distributions: Kali Linux, Parrot OS. Case Study: Penetration Testing for SecureBank's Online Banking Platform

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Explain the principles and methodologies of ethical hacking and penetration testing.
CO2: Analyze social engineering attacks (phishing, vishing, USB drops)
CO3: Identify system and network vulnerabilities using appropriate tools and techniques.
CO4: Demonstrate the ability to simulate basic exploitation techniques in a secure environment.
CO5: Prepare professional security assessment reports with risk analysis and mitigation strategies.

Text Book(s)

1. Stallings, William. *Network Security Essentials: Applications and Standards*. 6th ed., Pearson, 2021.
2. Stallings, William. *Cryptography and Network Security: Principles and Practice*. 8th ed., Pearson, 2020.

Reference Books

1. Cheswick, William R., et al. *Firewalls and Internet Security: Repelling the Wily Hacker*. 2nd ed., Addison-Wesley Professional, 2003.
2. Snader, Jon C. *VPNs Illustrated: Tunnels, VPNs, and IPsec*. Addison-Wesley Professional, 2005.

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

Dept. of Computer Applications

Discipline Elective – III

24MCAP407 FOG COMPUTING

L	T	P	C
4	0	0	4

Pre-requisite **24MCAP409**

Course Description:

This course extends cloud computing capabilities to the edge of the network. Students will explore the basic knowledge, architecture, principles, and applications of fog computing, and understand how it supports with real-time applications in domains such as IoT, and healthcare. Students will learn security, privacy, and data management challenges in fog environments.

Course Objectives:

This course enables students to

1. Explain the fundamental concepts of Fog Computing and its significance in distributed systems.
2. Compare different Fog Computing architectures, protocols, and resource management techniques.
3. Apply Fog Computing solutions to real-world applications
4. Analyze security threats and evaluate privacy mechanisms in Fog Computing environments.
5. Explore emerging trends (AI/ML, 5G integration) and research challenges in Fog Computing.

UNIT I INTRODUCTION TO FOG COMPUTING 12 Hours

Definition – Evolution – Characteristics – Applications - need for Fog Computing - similarities and differences of Fog vs Edge vs Cloud Computing - Issues and challenges - Key characteristics: proximity - low latency - geographic distribution. Use cases: IoT, smart cities, healthcare, industrial automation.

UNIT II FOG COMPUTING ARCHITECTURE AND COMPONENTS 12 Hours

Fog Computing Architecture: Layers and Components - Fog Nodes: Hardware and Software Components. Fog Computing Platforms: Cisco Fog Computing: Overview – Architecture - Components. OpenFog Consortium: Overview – Architecture – Components. Google Cloud IoT Edge: Overview – Architecture – Components.

UNIT III FOG COMPUTING APPLICATION 12 Hours

Smart Cities: Overview, Applications, and Challenges - Smart Agriculture: Overview, Applications, and Challenges - Smart Manufacturing: Overview, Applications, and Challenges - Fog Computing for Video Analytics: Applications, Challenges, and Solutions - Fog Computing for Smart Health Monitoring: Applications, Challenges, and Solutions.

UNIT IV SECURITY AND PRIVACY IN FOG COMPUTING 12 Hours

Threat models and attack surfaces in Fog environments - Authentication and access control mechanisms - Data privacy and encryption techniques - Trust management and blockchain for Fog security - Compliance and regulatory challenges (GDPR, HIPAA).

UNIT V EMERGING TRENDS AND CHALLENGES 12 Hours

AI/ML for Fog resource optimization - 5G and Fog synergy for ultra-low latency - Sustainability: energy-efficient Fog deployments - Open-source tools (OpenStack, FogBus, EdgeX Foundry) - Research challenges: scalability – interoperability – QoS, Case study: Smart Traffic Management System

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Describe Fog Computing, compare it with Cloud/Edge Computing, and list use cases.
CO2: Analyze Fog architectures, protocols, and resource management techniques.
CO3: Design and implement Fog solutions for healthcare, smart cities, and industrial automation.
CO4: Evaluate security threats and recommend privacy mechanisms in Fog environments.
CO5: Explore AI/ML, 5G trends, and discuss research challenges in Fog Computing.

Text Book(s)

1. Assad Abbas, Samee U. Khan, "Fog Computing: Theory and Practice " Wiley Publication, 2020.
2. Mukesh Singhal and Niharika Mishra, "Fog Computing: Concepts, Frameworks and Technologies", Springer, 2022.

Reference Books

1. Amir Vahid Dastjerdi and Rajkumar Buyya, “Fog Computing: Helping the Internet of Things Realize its Potential”, University of Melbourne, 2016.
2. Rajkumar Buyya, Satish Narayana Srirama, “Fog and Edge Computing Principles and Paradigms”, Wiley Publication, 2019.
3. IEEE/ACM papers on Fog Computing.
4. https://www.youtube.com/watch?v=epROHm2IMGo

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

Dept. of Computer Applications

Discipline Elective – IV

24MCAP410 AGILE METHODOLOGIES

L	T	P	C
3	0	0	3

Pre-requisite None

Course Description:

This course aims to provide students with a theoretical as well as practical understanding of agile software development practices and how small teams can apply them to create high-quality software. It also provides a good understanding of software design and a set of software technologies and APIs.

Course Objectives:

This course enables students to

1. Explain the object-oriented techniques used in the real-world software industries
2. Provide students with a theoretical as well as practical understanding of agile software development practices.
3. Understand the software design, technologies and APIs.
4. Examine and demonstrate the agile development environment and its testing techniques.
5. Improve workflows and speed up deployment.

UNIT I INTRODUCTION TO OBJECT ORIENTED ANALYSIS 9 hours

Introduction: SDLC & its variations, Approaches to system development, SSAD, OOAD, Unified Process and its characteristics, Unified Process life cycle, Basics of Object Orientation- OO development and its themes; Modelling and usefulness of OO development

UNIT II AGILE FUNDAMENTALS 9 hours

Introduction to Agile Management – Agile Software Development – Traditional Model vs. Agile Model - Classification of Agile Methods – Agile Manifesto and Principles – Agile Project Management – Agile Team Interactions – Ethics in Agile Teams - Agility in Design, Testing – Agile Documentations – Agile Drivers, Capabilities and Values.

UNIT III AGILE PROCESSES 9 hours

Lean Production - SCRUM, Crystal, Feature Driven Development- Adaptive Software Development - Extreme Programming: Method Overview – Lifecycle – Work Products, Roles and Practices

UNIT IV AGILITY AND KNOWLEDGE MANAGEMENT 9 hours

Agile Information Systems – Agile Decision Making –Development, Acquisition, Refinement, Distribution, Deployment, Leveraging – KM in Software Engineering – Managing Software Knowledge – Challenges of Migrating to Agile Methodologies – Agile Knowledge Sharing – Role of Story-Cards – Story-Card Maturity Model (SMM)

UNIT V INTRODUCTION TO DEVOPS 9 hours

Continuous Delivery, The Principles of Flow Feedback Continual Learning and Experimentation. Selecting which value stream to start with Understanding the Work in our value stream Making it visible Expanding - Design Organization and architecture Outcomes by Integrating Operations.

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Analyse the object-oriented techniques.
CO2: Understand the Agile software development processes
CO3: Perform iterative software development processes: how to plan them, how to execute them
CO4: Review the impact of social aspects on software development success
CO5: To analyse DevOps and its relationship with agile development.

Text Book(s)

1. Konnor Cluster, "Agile Project Management: Learn How To Manage a Project With Agile Methods, Scrum, Kanban and Extreme Programming", Independently Published, 2019.

Reference Books

1. Craig Larman, "Agile and Iterative Development: A Manager's Guide", Pearson Education, Second Impression, 2007

2. Object Oriented Analysis and Design with Applications by Grady Booch Robert, Third Edition, 2009.

3. Shore, James. The Art of Agile Development: Pragmatic guide to agile software development. "O' Reilly Media, Inc.", 2007.

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

Discipline Elective – IV

24MCAP411 SOFTWARE PROJECT MANAGEMENT

L	T	P	C
3	0	0	3

Pre-requisite None

Course Description:

This course provides advanced strategies for managing software projects in dynamic and distributed environments. It integrates both traditional models (Waterfall, V-Model, and CPM) and modern approaches (Agile, DevOps, Scrumban, and AI-driven practices). Emphasis is placed on hybrid project planning, risk analytics, AI/ML-based estimation, real-time monitoring, and virtual team leadership using cutting-edge collaboration tools.

Course Objectives:

This course enables students to:

1. Analyze software project lifecycles (Agile, Hybrid, V-Model) and stakeholder ecosystems.
2. Apply AI/ML-based estimation tools and risk prediction models.
3. Design hybrid project plans using PERT/CPM, Scrumban, and DevOps pipelines.
4. Implement real-time monitoring via EVA, AI-driven dashboards.
5. Lead distributed teams using virtual collaboration tools.

UNIT I FOUNDATIONS & MODERN SPM 9 Hours

Evolution of SPM: Waterfall → Agile → DevOps → AI-Driven PM, Software vs. AI/Data Projects: Key differences. Stakeholder 4.0: Managing clients, remote teams, and AI stakeholders. Problems with software projects, Management Control, Stakeholders, Information and control in organization, overview of project planning.

UNIT II AI-POWERED ESTIMATION & RISK 9 Hours

Estimation 2024: COCOMO III, ML-based effort prediction (e.g., OpenAI Codex). Risk Analytics: Monte Carlo simulations, AI risk prioritization (Qualtrics). Cost-Benefit 2.0: ROI of AI tools (GitHub Copilot, ChatGPT for PM). Workshop: Build a risk matrix using Python/pandas.

UNIT III HYBRID PLANNING & EXECUTION 9 Hours

Objectives of Activity Planning, when to Plan, Project Schedules, Sequencing and Scheduling Activities. Hybrid Models: Scrumban, SAFe, Agile-Waterfall bridges. CPM/PERT++: Critical Chain Method, AI-aided scheduling. Case Study: Failed vs. successful projects.

UNIT IV REAL-TIME MONITORING & AI OPS 9 Hours

EVA 2.0: AI-enhanced Earned Value dashboards (Tableau/Power BI). Change Control: Git workflows, AI-based version conflict resolution. AI Ops: Predictive alerts for delays (e.g., Azure AI). Case Study: AI-driven release management.

UNIT V FUTURE OF TEAMS & LEADERSHIP 9 Hours

Remote Teams 2024: Async collaboration (Slack, Miro), behavioral AI (Humanyze). Motivation 4.0: Gamification, neuroleadership. Ethics in AI PM: Bias, accountability in automated decisions. Role-Play: Virtual Scrum meeting with AI scribe (Otter.ai).

Course Outcomes:

Upon successful completion of the course, students will be able to
CO1: Compare traditional and AI-driven SPM frameworks.
CO2: Deploy ML tools for estimation/risk (e.g., COCOMO III).
CO3: Design hybrid plans (Agile + Waterfall + DevOps).
CO4: Evaluate monitoring with AI dashboards.
CO5: Lead ethical, distributed teams using behavioural AI.

Text Book(s)

1. Hughes, Bob, and Rajib Mall. *Software Project Management*. 6th ed., Tata McGraw Hill, 2024.

2. Puntambekar, A. A., Satish S. Banait, and Dinesh B. Hanchate. *Software Engineering & Project Management*. Technical Publications, 2020.

Reference Books

1. Project Management Institute. *A Guide to the Project Management Body of Knowledge (PMBOK Guide)*. 7th ed., Project Management Institute, 2023.

2. Project Management Institute. *AI in Project Management*. PMI Publications, 2023.

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

Dept. of Computer Applications

Discipline Elective – IV

24MCAP412 DEVOPS AND MICROSERVICES

L	T	P	C
3	0	0	3

Pre-requisite 24MCAP403

Course Description:

This course provides an in-depth exploration of DevOps practices and micro services architecture, aimed at equipping students with the skills necessary to design, deploy, and manage scalable applications. Upon completion, Students will be able to effectively implement DevOps practices and design micro services-based applications, improving their organization's software development lifecycle and operational efficiency.

Course Objectives:

This course enables students to

1. Explain fundamental DevOps concepts and their role in SDLC
2. Apply DevOps tools and practices in software development
3. Analyze Microservices architecture and its implementation
4. Evaluate different architectural styles (monolithic vs microservices)
5. Analyse continuous delivery pipelines for high-velocity deployments

UNIT I DEV OPS TOOLS 9 hours

History of DevOps- DevOps and Software Development Life Cycle – Waterfall Model _Agile Model – DevOps Life Cycle – DevOps Tools: distributed version of control tool Git- Automation testing tools Selenium – report generation –Test NG – User Acceptance Testing – Jenkins

UNIT II INTRODUCTION TO MICROSERVICES 9 hours

Definition of Microservices – Characteristics - Microservices and Containers – Interacting with Other Services – Monitoring and Securing the Services – Containerized Services – Deploying on Cloud-Case Study: Deploying Micro services in the Cloud for Shop Easy – An E-commerce Platform

UNIT III MICROSERVICES ARCHITECTURE 9 hours

Monolithic architecture- Microservices architectural style- Benefits - Drawbacks of Microservices architectural style -decomposing monolithic applications into Micro services- Monolithic Architecture at Edu Learn – An Online Learning Platform.

UNIT IV MICROSERVICES IN DEV OPS ENVIRONMENT 9 hours

Evolution of Micro services and DevOps – Benefits of combining DevOps and Micro services-working of DevOps and Micro services in Cloud environment - DevOps Pipeline representation for a Node JS based Micro services.

UNIT V VELOCITY AND CONTINUOUS DELIVERY 9 hours

Velocity - Delivery Pipeline- test stack - Small/Unit Test – medium /integration testing – system testing- Job of Development and DevOps - Job of Test and DevOps – Job of Op and DevOps. Infrastructure and the job of Ops. **Case Study:** DevOps and Microservices Transformation at medium size technical Corporation.

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Understand about DevOps and the common tools used in DevOps.

CO2: Illustrate how microservices integrate with DevOps pipelines in cloud environments

CO3: Implement containerized microservices with proper monitoring and security

CO4 : Compare monolithic and microservices architectures using case study examples

CO5: Design a complete delivery pipeline with integrated testing strategies

Text Book(s)

1. Tanasseri, N., and R. Rai. *Microservices with Azure: Build Highly Maintainable and Scalable Enterprise-Grade Apps*. Packt Publishing, 2017.

2. Wolff, E. *Microservices: Flexible Software Architecture*. Pearson Education, 2016.

Reference Books

1. James A Scott, A Practical Guide to Microservices and Containers, MapR Data Technologies https://mapr.com/ebook/microservices-and-containers/assets/microservices-andcontainers.pdf

2. Kim, Gene, et al. *The Phoenix Project: A Novel about IT, DevOps*. 5th ed., IT Revolution Press, 2018.

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

Dept. of Computer Applications

Discipline Elective – IV

24MCAP413 OBJECT ORIENTED ANALYSIS AND DESIGN

L	T	P	C
3	0	0	3

Pre-requisite 24MCAP108, 24MCAP403

Course Description:

This course introduces the fundamental principles and methodologies of **Object-Oriented Analysis and Design (OOAD)** for developing robust, scalable, and maintainable software systems. Students will learn how to model real-world problems using object-oriented concepts, apply design patterns, and follow best practices in software architecture.

Course Objectives:

This course enables students to:

1. Understand overview of object-oriented concepts and its life cycle for software development.
2. Learn for modelling the software and to design them using UML diagrams
3. Understand the problem domain and to identify the objects from the problem specification.
4. Apply design axioms and corollaries for the classes and object relational systems.
5. Utilize source tools for Computer Aided Software Engineering

UNIT I INTRODUCTION 9 Hours

An overview – Object basics – Object state and properties – Behaviour – Methods – Messages – Information hiding – Class hierarchy – Relationships – Associations –Aggregations- Identity – Dynamic binding – Persistence – Meta classes – Object oriented system development life cycle.

UNIT II METHODOLOGY AND UML 9 Hours

Introduction – Survey – Rumbaugh, Booch, Jacobson methods – Unified modelling language – Static and Dynamic models – Rational Rose Suite - UML diagrams – Static diagram : Class diagram – Use case diagrams – Behaviour Diagram : Interaction diagram – State chart diagram – Activity diagram - Implementation diagram: Component diagram – Deployment diagram – example - Design of online railway reservation system using UML diagrams - Dynamic modelling – Model organization – Extensibility.

UNIT III OBJECT ORIENTED ANALYSIS 9 Hours

Identifying Use case – Business object analysis – Use case driven object oriented analysis – Use case model – Documentation – Classification – Identifying object, relationships, attributes, methods – Super-sub class – A part of relationships Identifying attributes and methods – Object responsibility – construction of class diagram for generalization, aggregation – example – vehicle class.

UNIT IV OBJECT ORIENTED DESIGN 9 Hours

Design process and benchmarking – Axioms – Corollaries – Designing classes – Class visibility – Refining attributes – Methods and protocols – Object storage and object interoperability – Databases – Object relational systems – Designing interface objects – Macro and Micro level processes – The purpose of a view layer interface-OOUI – MVC Architectural Pattern and Design – Designing the system.

UNIT V CASE TOOLS **9 Hours**

Railway domain : Platform assignment system for the trains in a railway station – Academic domain : Student Marks Analysing System - ATM system - Stock maintenance – Quiz System - E-mail Client system - Cryptanalysis – Health Care Systems. Use Open source CASE Tools: StarUML/ UML Graph for the above case studies.

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Understand the object-oriented concepts and to apply object-oriented life cycle model for a project.
CO2: Design static and dynamic models using UML diagrams.
CO3: Perform object-oriented analysis to identify the objects from the problem specification.
CO4: identify and refine the attributes and methods for designing the object oriented system.
CO5: Utilize open source CASE tools and to apply them in various domains

Text Book(s)

1. Bahrami, Ali. *Object Oriented System Development*. McGraw Hill, 2008.
2. athan, Brahma, and Sarnath Ramnath. *Object-Oriented Analysis, Design and Implementation*. 2nd ed., Universities Press, 2015.

Reference Books

1. Larman, Craig. *Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development*. 3rd ed., Pearson Education, 2005.
2. Booch, Grady, et al. *The Unified Modeling Language User Guide*. 2nd ed., Addison-Wesley, 2005.
3 Bruegge, Bernd, and Allen H. Dutoit. *Object-Oriented Software Engineering Using UML, Patterns, and Java*. 3rd ed., Pearson, 2010.

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

Skill Enhancement Course

Dept. of Computer Applications

Skill Enhancement Course – I

24ENGP601 CORPORATE COMMUNICATION

L	T	P	C
2	0	2	3

Pre-requisite **None**

Course Description:

Course Description: English is practical and it is a must for any institution to provide students with opportunities to indulge in actively applying their language skills. Thus, the Communication Skills Lab facilitates students with adequate opportunities to put their communication skills in use. It also accommodates peer learning by engaging students in various interactive sessions. This lab will be accompanied by a practical lab component.

Course Objectives:

This course enables students able to

1. Comprehend listening tasks and enhance formal conversations and presentations skills
2. Read and write various genres of texts encountered in corporate scenario
3. Prepare and succeed in verbal ability round of job interviews and competitive exams
4. Prepare and succeed in HR interview round of the job selection process
5. Enhance pronunciation and neutralize accent

UNIT I LISTENING AND SPEAKING SKILLS 9 hours

Listening/watching interviews, conversations, documentaries, etc.; Listening to lectures, discussions from TV/Radio/Podcast. Conversational skills (Formal and Informal); Group Discussion; Making effective Power Point presentations.

UNIT II READING AND WRITING SKILLS 9 hours

Reading different genres of texts including newspapers Magazines: creative writing; Writing job applications and resume; Emails; Letters; Memorandum; Reports; Writing abstracts and summaries; Interpreting visual texts.

UNIT III ACCLIMATIZING STUDENTS TO OTHER EXAMS 9 hours

Test of English as a Foreign Language (TOEFL); International English language Testing System (IELTS); Civil Service Examinations; Verbal-ability.

UNIT IV INTERVIEW SKILLS 9 hours

Different types of interviews: Answering questions and offering information; Mock interviews; Body Language.

UNIT V PHONETICS `` 9 hours

Vowels, Consonants, Articulation of sounds, Neutralization of Accent; Word Stress, Sentence Stress and Intonation

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Comprehend audio-visual content in English and improve their conversational and presentation skills

CO2: Understand various genres of texts and write texts used as a part of corporate communication

CO3: Apply verbal ability and reasoning skills in job selections and competitive exams

CO4: Analyze the expectations of recruiters and answer common HR interview questions

CO5: Apply knowledge of phonetics, stress and intonation of English in everyday usage

Text Book(s)

1. Sanjay Kumar and PushpLata; Communication Skills; Oxford University Press, 2012.

2. Sabina Pillai and Agna Fernandez; Soft Skills and Employability Skills;Cambridge University Press, 2018.

3. S.P. Dhanavel; English and Communication Skills for Students of Science and Engineering; Orient Blackswan, 2009

4 M. Ashraf Rizvi; Effective Technical Communication; Tata Mc Graw Hill Co. ltd, 2005.

Reference Books

1. Dr.M.Adithan; *Study Skills for Professional Students in Higher Education*; S.Chand& Co. Pvt., 2014.

2. Guy Brook Hart & Vanessa Jakeman; *Complete IELTS*: Cambridge University Press, 2014.

3. Vanessa Jakeman& Clare Mcdowell; *Action Plan for IELTS:* Cambridge University Press, 2006.

4. Guy Brook Hart; *Instant IELTS*;Cambridge University Press, 2004

5. S.P.Bakshi&Richa Sharma; *Descriptive General English;*Arihant Publications, 2012.

6. Charles Browne, Brent Culligan 7 Joseph Phillips; *In Focus (level 2);* Cambridge University Press.

7. Steven Gershon; *Present Yourself 2* (second edition); Cambridge University Press.

8. Leo Jones; *Let's Talk 3*(second edition); Cambridge University Press.

9. Nutall J. C.; *Reading Comprehension*; Orient Blackswan.

10. *www.cambridgeenglish.org/in/*

11. https://learnenglish.britishcouncil.org/en/english-grammar

12. https://www.rong-chang.com/

Mode of Evaluation: Continuous Internal Evaluation Assignments, Mid Term Tests, End Semester Examination

Skill Enhancement Course – II

24MCAP601 FRONTEND WEB DEVELOPMENT

L	T	P	C
2	0	2	3

Pre-requisite **None**

Course Description:

To understand the basics of frontend web development including HTML, CSS and JavaScript. the students can update their knowledge on technologies. This will help the students to learn the complete set of process-like designing, development and deployment.

Course Objectives:

This course enables students to

1. Understand the Fundamentals of HTML web pages.
2. Apply CSS Styling Techniques, and properties to style web pages effectively, including color manipulation, font styling, and element positioning.
3. Familiarize web page with validation using JavaScript
4. Understand different DOMs and event handling mechanisms.
5. Understand and practice embedded dynamic scripting on client-side Internet

UNIT I UI DESIGN 9 hours

HTML Introduction - Basics- Elements- Attributes- Headings- Paragraph- Tables- Formatting- Links and Images - Lists- Blocks- Layout -Responsive - IFrames -Forms – Form Elements- Introduction to HTML5

1. Write a program to implement Formatting text in HTML (font, color, Heading, Background color)
2. Write a program to Create a Table in HTML.
3. Create Registration Form using HTML Form elements.
4. Use frames such that page is divided into 3 frames 20% on left to show contents of pages, 60% in center to show body of page, remaining on right to show remarks.

UNIT II CASCADING STYLE SHEET 9 hours

Cascading Style Sheet (CSS3): The need for CSS – Basic syntax and structure Inline Styles – Embedding Style Sheets - Linking External Style Sheets - Introduction to CSS3 – Backgrounds - Manipulating text - Margins and Padding - Positioning using CSS - Responsive Web Design - Introduction to LESS/SASS

Add Styles to your Resume using CSS 3 Properties.

1. Write CSS for Fonts, Background, Color, Text
2. Add External, Internal and Inline CSS styles to know the priority.
3. Write an HTML page that contains a selection box with a list of 5 countries, when the user selects a country, its capital should be printed next to the list; Add CSS to customize the properties of the font of the capital (color, bold and font size)

UNIT III OVERVIEW OF JAVASCRIPT 9 hours

Introduction - Core features - Data types and Variables - Operators, Expressions, and Conditional Statements (if, if-else, if-else-if-else), switch-case, loops, (while, for, do-while), Functions, Objects - Array, Date and Math Related Objects.

1. Javascript Program to Generate a Random Number
2. Write a JavaScript conditional statement to sort three numbers.
3. Write JavaScript to demonstrate loops: while, for, do-while JavaScript Program to Find the Sum of Natural Numbers
4. Create a CGPA Calculator in Web Brower using HTML, CSS and JavaScript. Use functions in JavaScript. UNIT IV DOM & Event Handling

UNIT IV DOM AND EVENT HANDLING 9 hours

Document Object Model: getElementByid(), getElementByClassName(), getElementByName(), getElementByTagName(), JS innerHTML Propery - Event Handling - Controlling Windows & Frames and Documents - Form validations- Exception Handling.

1. Write JavaScript to validate the following fields of the Registration page
2. Write a JavaScript program to demonstrate Event Handling.
3. Write a program to demonstrate exception handling in JS.

UNIT V ADVANCED FEATURES OF JAVASCRIPT 9 hours

Browser Management and Media Management – Classes – Constructors – Object-Oriented Techniques in JavaScript – Object constructor and Prototyping - Sub classes and Super classes – Introduction to JSON – JSON Structure –Introduction to jQuery –Introduction to AJAX-Bootstrap - Bootstrap components.

1. Create a Pan Card Validation form using Object Oriented JavaScript, consider the 10th character to be an alphabet.
    a) Get the user's First Name, Last Name and other required fields as input
    b) Assume the last digit of the Pan Number to be an alphabet
    c) Validate the PAN Number.
2. Construct a JSON Structure for a bookstore and validate it using JSON Validator such as http://jsonlint.com/ and parse the Json file to list the books under the category "Fiction".

3. Create a Single Page application allowing to search for a movie and displaying the trailer, poster for various movies.
    a) Create an admin login to upload the trailer, poster, keyword and details of the movie.
    b) Use Bootstrap and JQuery for designing the User Interface.
    c) Form Submission should be handled through Ajax.

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Design a complete, well-organized website using HTML.

CO2: Design interactive websites and dynamically use CSS to optimize user experience

CO3: Develop the dynamic web pages using JAVA script

CO4: Create Dynamic webpages using DOMs and Form Validation.

CO5: Employ object-oriented programming principles.

Text Book(s)

1. Deitel and Deitel and Nieto, Internet and World Wide Web - How to Program, Prentice Hall, 5th Edition, 2011.
2. Java Script for Programmers Paul J. Deitel, Deitel & Associates, Inc.Harvey M. Deitel, Deitel& Associates, Inc.

Reference Books

1. Stephen Wynkoop and John Burke "Running a Perfect Website", QUE, 2nd Edition,1999
2. Web Coding Bible, An Accelerated Course, Chong Lip Phang, 2015

Mode of Evaluation: Continuous Internal Evaluation Assignments, Mid Term Tests, End Semester Examination

Skill Enhancement Course – III

24MCAP602 MOBILE APPLICATION DESIGN AND DEVELOPMENT

L	T	P	C
2	0	2	3

Pre-requisite None

Course Description:

This course offers a thorough foundation in Android development, teaching core concepts like architecture, UI design, threading, and storage systems. Learners will create interactive apps using Recycler View, Material Design, and Intents while exploring advanced features like sensors, databases, and location services. Practical Android Studio exercises and real-world projects build hands-on skills for developing professional mobile applications.

Course Objectives:

This course enables students to

1. Understand the architecture, evolution, and life cycle of Android applications,
2. Design and implement responsive user interfaces using Android UI components
3. Develop robust Android applications by applying threading models
4. Utilize Android's storage mechanisms
5. Integrate device sensors and location-based services to build interactive and context-aware mobile applications.

UNIT I INTRODUCTION ABOUT ANDROID TOOLS 6 Hours

Android Overview – History – Android Versions - Android Stack: Linux, Native Layer and Hardware Abstraction Layer (HAL) –Installing the Android SDK - Anatomy of an Android Project.

1. Create a new Android Project and run a simple "Hello World" app on an emulator or physical device.
2. Design a layout with TextView, ImageView, Button, and EditText.

UNIT II BUILDING USER INTERFACE 6 Hours

Input Components – Text View – Image View – List View and Alert Dialogues – Menus: Popup, Options and Context Menus – Screen Navigation through App Bar – Recycler View –Usage of Intends – Creation of Indents with example program – Lists and Adapters – Types of Adapters.

1. Design a screen with ImageView showing an image resource.
2. Add an Alert Dialog with OK/Cancel buttons in response to a button click
3. Implement navigation between two activities using intents.

UNIT III APPLICATION DESIGN 6 Hours

Threading in Android – AsyncTask – Loaders – AsyncTask Loader –Broadcast Receivers: Custom Broadcasts - Boot Receiver - Alarms and system services –Services: Services Life Cycle – Intent Service – Implementing Intent Service – Notifications: Managing Notifications.

1. Implement an Intent Service and demonstrate its use case.
2. Implement AsyncTask to perform a background operation with UI update.

UNIT IV ANDROID FILE SYSTEM 6 Hours

Android File systems and Files - Action Bar: Preferences and Action Bar - Shared Preferences – App Settings - Databases on Android - SQLite - Content Providers: Overview – Role of Content Providers - Content Provider Example Program – Content Resolver .

1. Design a simple app to create, read, update, and delete records in SQLite database.
2. Build a custom Content Provider and query it via Content Resolver.

UNIT V APPLICATION DEVELOPMENT AND SENSOR 6 Hours

App Widgets: Creation of Application Widgets - Interaction and Animation: Live Wallpaper and Handlers - Sensors: Sensor API in Android - Motion Sensor, Position Sensor, Environmental Sensor, orientation Sensors, Sensor Examples

1. Create a simple App Widget that updates content periodically.
2. Implement a basic animation using Handlers.
3. Develop an app that listens to accelerometer sensor data and displays it in real-time.

Course Outcomes:

Upon successful completion of the course, students will be able to
CO1: Understand the basics of mobile application development and Install android.
CO2: Develop a UI for mobile applications.
CO3: Work with Broadcast Receivers and Services.
CO4: Create Database in Android.
CO5: Build widgets, Wall papers for an android application using sensors

Text Books

1. *Dawn Griffiths, David Griffiths* **"Head First Android Development"** *2nd Edition, 2021,*

2. *Ian Darwin* **"Android Cookbook"** *2nd Edition, O'Reilly, 2022,*

Reference Books

1. *Zigurd Mednieks, Laird Dornin* **"Programming Android"** *2nd Edition, O'Reilly, 2022, , Blake Meike*

2. *Varun Nagpal* **"Android Sensor Programming By Example"** *Packt, 2023,*

Mode of Evaluation: Assignments, Quiz, Seminar, Case study, Mid Term Tests, End Semester Examination.

Audit Course

Audit Course

24MCAP901 RESEARCH METHODOLOGY AND IPR

L	T	P	C
2	0	0	0

Pre-requisite

Course Description:

This course aims has been meticulously designed to provide a comprehensive understanding of the essential aspects of research methodology and the critical realm of Intellectual Property Rights (IPR).

Course Objectives:

This course enables students to

1. Identify and discuss the concepts and finding the criteria of good research.
2. Find sources to data collection.
3. Understand the methodologies of data analysis and reporting.
4. Understand the basics of Intellectual Property Rights.
5. Know the process of patents and filling.

UNIT I INTRODUCTION 6 hours

Introduction, Objectives and types of Research, Research approaches, Overview of research process and design, Criteria of Good Research. Research Tools.

UNIT II DATA COLLECTION AND SOURCES 6 hours

Measurements, Measurement Scales, Questionnaires and Instruments, Sampling and methods, Qualitative and Quantitative Data - Preparing, Exploring, examining and displaying of data.

UNIT III DATA ANALYSIS AND REPORTING 6 hours

Overview of Multivariate analysis, Hypotheses testing and Measures of Association. Presenting Insights and findings using written reports and oral presentation.

UNIT IV INTELLECTUAL PROPERTY RIGHTS 6 hours

Intellectual Property – The concept of IPR, Evolution and development of concept of IPR, IPR development process, Trade secrets, utility Models, IPR & Biodiversity, Role of WIPO and WTO in IPR establishments, Right of Property, Common rules of IPR practices, Types and Features of IPR Agreement, Trademark, Functions of UNESCO in IPR maintenance.

UNIT V PATENTS 6 hours

Patents – objectives and benefits of patent, Concept, features of patent, Inventive step, Specification, Types of patent application, process E-filling, Examination of patent, Grant of patent, Revocation, Equitable Assignments, Licences, Licensing of related patents, patent agents, Registration of patent agents.

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Formulate and Design research problem.
CO2: Understand and Comprehend the Data Collection Methods.
CO3: Perform Data analysis and acquire Insights.
CO4: Understand IPR and follow research ethics.
CO5: Evaluate the Patent Drafting and Filing in research areas.

Text Book(s)

1. Kothari C R, Gaurav Garg, "Research Methodology", New Age International Publishers, 5th Edition, 2023.

Reference Books

1. The Institute of Company Secretaries of India, Statutory body under an Act of parliament, "Professional Programme Intellectual Property Rights, Law and practice", September 2013.

2. Cooper Donald R, Schindler Pamela S and Sharma JK, "Business Research Methods", Tata McGraw Hill Education, 11e (2012).

3. Catherine J. Holland, "Intellectual property: Patents, Trademarks, Copyrights, Trade Secrets", Entrepreneur Press, 2007

4. David Hunt, Long Nguyen, Matthew Rodgers, "Patent searching: tools & techniques", Wiley, 2007.

Mode of Evaluation: Assignments, Mid Term Tests.