GURU NANAK COLLEGE (AUTONOMOUS)

A Sikh Minority Institution | Unit of Guru Nanak Educational Society (Regd.)
Affiliated to University of Madras | Approved by AICTE
Accredited at A++ Grade by NAAC (CGPA 3.68/4)
Guru Nanak Salai, Velachery, Chennai - 600042

SCHOOL OF INFORMATION TECHNOLOGY

MASTER OF COMPUTER APPLICATIONS

(SEMESTER PATTERN WITH CHOICE BASED CREDIT SYSTEM)

(For the PG Batch of 2026-28 and thereafter)

LOCF - LEARNING OUTCOME BASED CURRICULUM FRAMEWORK

PREAMBLE

1. ABOUT THE PROGRAMME

Master of Computer Applications is a 2 years postgraduate programme in computer science and its applications. The programme is designed to provide students with a strong foundation in computer science theory as well as practical skills in software development, programming languages, database management, networking, and other related areas.

The curriculum of an MCA program includes subjects like programming languages (Python , R, J2EE, Java, etc.), Data Structures and Algorithms, Computer Networks, Advanced Database Management Systems, Software Testing, Web Technologies, Data Science, Artificial Intelligence, Machine Learning and more.

MCA graduates pursue careers as Software Developers, Systems Analysts, Database Administrators, Web Developers, Software Engineers, and other roles in the IT industry. The demand for skilled professionals with expertise in computer applications continues to grow, making MCA a popular choice for those seeking to advance their careers in technology. The programme emphasizes the application of software technology to solve mathematical, computing, communications/networking, and commercial problems. The curriculum enables mastery of the fundamentals of computing and provides an opportunity to gain in-depth knowledge of a broad range of topics.

In Guru Nanak College, the sanctioned intake of students is 60. The Programme has Well-experienced and Qualified staff. Periodical Syllabus Revision made based on recommendations of Academic and Industry experts to suit the current Industrial needs. The Programme has State-of-the-art infrastructure, a well-equipped lab, and a Library. The Programme guides the students for International Certifications in IT. Periodic technical seminars and workshops by experts and alumni are organized by the Programme to give deep insights into the latest technology in demand and current real-world scenarios. Periodic Industrial Visits are organised to provide exposure to the current needs and work style of the IT industry. Peer learning is encouraged. Fast learners are encouraged to do In-House projects

2. VISION

To impart essential knowledge in Information technology to the students, and enhance their ability to apply the knowledge gained and be successful in their professional and social life, contribute to their progress and thereby get involved in the upliftment of the society.

3. MISSION

- Equip the students with global technological skills in Information Technology that enhance them to be innovative, have lateral thinking, and good at problem-solving.
- Increase Industry - Institute Interaction to enlighten the students about the required skills to be successful in their careers.
- Train and develop the students as IT Professionals with Confidence, Competence, Commitment, and Character.

4. PROGRAMME EDUCATIONAL OBJECTIVES (PEOS)

1. Technical and Professional Competence: Postgraduates will acquire advanced knowledge in computer applications, software development, data management, and emerging technologies, enabling them to excel in IT industries, academia, and research.
2. Analytical and Problem-Solving Skills: Postgraduates will apply computational thinking, algorithms, and modern tools to design and implement innovative solutions for complex real-world problems across diverse domains.
3. Ethics and Social Responsibility: Postgraduates will uphold professional ethics, integrity, and social responsibility, contributing to sustainable technological development and addressing societal challenges through IT solutions.
4. Leadership, Communication, and Collaboration: Postgraduates will demonstrate effective communication, teamwork, and leadership skills, preparing them to manage projects, collaborate across disciplines, and lead in dynamic organizational environments.
5. Adaptability, Research, and Lifelong Learning: Postgraduates will engage in continuous learning, research, and professional development, adapting to evolving global trends in computing, artificial intelligence, and digital transformation.

5. PROGRAMME OUTCOMES(POS)

1. **Foundation Knowledge**: Apply knowledge of mathematics, programming logic and coding fundamentals for solution architecture and problem solving.
2. **Problem Analysis:** Identify, review, formulate and analyse problems for primarily focussing on customer requirements using critical thinking frameworks.
3. **Development of Solutions**: Design, develop and investigate problems with as an innovative approach for solutions incorporating ESG/SDG goals.
4. **Modern Tool Usage**: Select, adapt and apply modern computational tools such as development of algorithms with an understanding of the limitations including human biases.

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

6. **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.

7. **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

8. **Life-long learning:** Change management skills and the ability to learn, keep up with contemporary technologies and ways of working.

6. PEO – PO MAPPING

	PEO1	PEO2	PEO3	PEO4	PEO5
PO1	3	2	–	–	1
PO2	2	3	1	1	2
PO3	3	3	2	1	2
PO4	3	2	1	–	2
PO5	1	1	1	3	2
PO6	2	1	2	3	1
PO7	2	1	3	1	2
PO8	1	2	1	1	3

7. CHOICE BASED CREDIT SYSTEM (CBCS)

The College follows the CBCS with Grades under the Semester pattern. Each course is provided with a credit point based on the quantum of subject matter, complexity of the content and the hours of teaching allotted. This is done after a thorough analysis of the content of each course by the members of the Board of Studies and with the approval of the Academic Council. Students are also offered a variety of Job oriented Elective courses as part of the curriculum. All PG Students must complete and obtain certification in one NPTEL / SWAYAM-MOOC of their choice before the end of the fourth semester. The course will carry 2 credits and is a mandatory requirement for the award of the degree. Students can earn extra credits by opting for Certificate Courses.

The evaluation method under CBCS involves a more acceptable grading system that reflects the personality of the student. This is represented as Cumulative Grade Point Average (CGPA) and Grade Point Average (GPA) which are indicators of the Academic Performance of the student. It provides students with a scope for horizontal mobility and empowers them with the flexibility of learning at their convenience.

Eligibility for Admission

Candidates admitted to the first year of the Master of Computer Applications should have passed B.C.A/B.Sc. in Computer Science or any other equivalent degree OR passed B.Sc./B.Com/BA with Mathematics at 10 + 2 level or at graduation level (with Optional bridge course in Mathematics), provided they have undergone the course under 10+2+3 pattern and obtained at least 50% of marks (45 % marks in case of candidates belonging to reserved category) in the qualifying examination shall be eligible for admission to the M.C.A. Programme. Students applying for the PG programme should have taken the UG degree in the relevant subject from a recognized university as per the norms of the University of Madras.

Duration of the Course

The PG programme is of two years duration with four semesters. The period from June to November is termed as the odd semester and the period from December to April is referred to as the even semester. Each semester must compulsorily have 90 working days before the students appear for the final End Semester Exam.

Course of Study

The main course of study for the Masters' Degree shall consist of the following:

PART - A: Consisting of (a) Core subjects (b) Elective courses related to the main subject of study, (c) Extra disciplinary subjects and (d) Main project.

PART – B : (a) Ability Enhancement Skills (I, II, III & IV Semesters) , (b) Summer Internship (After II Semester), (c) 1 MOOC Course /NPTEL

8. COURSE STRUCTURE

The MCA Programme consists of **17** Core courses with **3-4** credits for each course, **3** Elective courses of **3** credits and **1** EDE course with **3** credits each and **4** Ability Enhancement Skills courses with **2** credits each. Internship is a compulsory component with **2** credits. MOOC Course/ NPTEL is a compulsory component with **2** credits, and Project is a compulsory course with **11** credits.

Consolidated Credit Structure for all the 2 years

Course component		No. of Courses	Credits
Part A	Core	**17**	58
	Elective	**3**	9
	EDE	**1**	3
	Project	**1**	11
Part B	Ability Enhancement Skills	**4**	8
	Internship	**1**	2
	MOOC Course/ NPTEL	**1**	2
Total		**28**	**93**

Credit Distribution for Each Semester:

Semester I Course Component		Subject	Credit	Hours per week	Marks		Total
					Internal	External	
Part A	Core - I	Operating Systems with Linux	4	5	50	50	100
	Core - II	Advanced Database Management Systems	4	5	50	50	100
	Core - III	Advanced Java Technologies	4	5	50	50	100
	Elective I	Cyber Security and Networking / Cybersecurity and Ethical Hacking / Ethical Hacking and Penetration Testing	3	5	50	50	100
	Core - IV	Practical - I : ADBMS Lab	3	4	50	50	100
	Core - V	Practical - II: Advanced Java Technologies Lab	3	4	50	50	100
Part B	Ability Enhancement Skills - I	Communication and Presentation Skill	2	2	50	50	100
	Total		23	30			

Semester II Course Component		Subject	Credit	Hrs/ Week	Marks		Total
					Internal	External	
Part A	Core - VI	Python Programming	4	5	50	50	100
	Core - VII	Data Structures and Algorithms	4	5	50	50	100
	Core – VIII	Block Chain Technologies	3	4	50	50	100
	Core - IX	Artificial Intelligence	3	4	50	50	100
	EDE- I	Statistics for Data Analytics	3	4	50	50	100
	Core – X	Practical III : Python Programming Lab	3	3	50	50	100
	Core – XI	Practical IV : Data Structures and Algorithms Lab	3	3	50	50	100
Part B	Ability Enhancement Skills - II	Quantitative Aptitude	2	2	50	50	100
Total			25	30			

Semester III		Subject	Credit	Hrs/ Week	Marks		Total
Course Component					Internal	External	
Part A	CORE - XII	Machine Learning	4	5	50	50	100
	Core-XIII	Full Stack Software Development	4	5	50	50	100
	Core-XIV	Cloud Computing and DevOps	3	4	50	50	100
	Core-XV	R programming for Statistical Analytics	3	4	50	50	100
	Elective-II	Agile Software Development / Software Engineering / Software Project Management	3	4	50	50	100
	Core-XVI	Practical V : R programming for Statistical Analytics Lab	3	3	50	50	100
	Core-XVII	Practical VI : Full Stack Software Development Lab	3	3	50	50	100
Part B	Ability Enhancement Skills - III	Data Visualization	2	2	50	50	100
	MOOC / NPTEL	MOOC / NPTEL	2	-	-	-	-
	Internship	Internship	2	-	-	-	-
Total			29	30			

Semester IV		Subject	Credit	Hrs/ Week	Marks		Total
Course Component					Internal	External	
Part A	Elective-III	Edge Computing/ Parallel Computing/ Distributed Computing	3	4	50	50	100
	Project	Project and Viva Voce	11	23	20	60+20	100
Part B	Ability Enhancement Skills - IV	Architectural Design	2	3	50	50	100
Total			16	30			
Total Credits			93				

9. EXAMINATION

Continuous Internal Assessment (CIA) will be for 50 percent and End Semester Examination (ESE) will be for 50 percent. The CIA marks for the two credit courses (Ability Enhancement skill courses) coming under Part B, shall be awarded based on the nature of the course or at the course instructor's discretion. (Assignment/project/presentation).

Continuous Internal Assessment (CIA)

Every semester will have a mid-semester examination which will be conducted on completion of 45 working days in each semester. A Model exam for three hours duration will be conducted on completion of 80 working days in each semester. For the courses coming under Part B ESE will be conducted in MCQ pattern. Internship credits will be given in semester III after verification of documents by the respective Heads.

The schedule for these tests is as follows:

CIA	Schedule	Syllabus Coverage
Mid Semester Examination	After **45** working days of the Semester	50%
Model Examination	After **85** working days of the Semester	100%

The components for the CIA (Theory & Practical) are as follows:

Continuous Internal Assessment - Components			
Assessment Type	**Nature**	**Maximum Marks**	**Weightage**
Mid-Semester Examination	Pen and Paper test for 90 minutes	50	10
Model Examination	Written examination for 180 minutes	100	10
Assignment			5
Workbook / recordbook / notebook			5
Class activity 1			5
Class activity 2			5
Class activity 3			5
Attendance	Marks will be given based on the attendance percentage of respective course		5
Total			50

The Mode of Evaluation for class activity can be based on the following parameters (these are suggestions, the faculty can add more if needed)

1) Quiz with MCQ
2) Case-Study analysis
3) Debate
4) Seminar/ Presentation
5) Peer/Self-Assessment
6) Role play
7) Scrap book
8) Monograph
9) Puzzle
10) Group Discussion
11) Group activity
12) Surprise class tests

A record of all such assessment procedures will be maintained by the department and is open for clarification. Students will have the right to appeal to the Principal in case of glaring disparities in marking. CIA marks for practical subjects will be awarded by the respective faculty based on the performance of the student in the model practical examination, observation notebook, submission of record books, regularity and attendance for the practical classes. The attendance particulars for practical classes will be maintained by the concerned faculty.

The marks for attendance will be awarded as per the following:

Percentage of General Attendance	**Marks Awarded**
90-100	5
75-89	4
60-74	3
<60	0

End Semester Examinations (ESE)

After the completion of a minimum of 90 working days each semester, the End Semester Examinations will be conducted. Examinations for all UG and PG programmes will be held for all courses in November/December and April/May. Practical examinations will be conducted only during the end of the odd / even semester before, during or after the commencement of the theory exam. The schedule for ESE Practicals will be notified by the Controller of Examinations in consultation with the Dean Academics.

Mode of Evaluation

METHODS OF EVALUATION		
Internal Evaluation	Mid Sem Exam (10)	50 Marks
	Model Exam (10)	
	Assignment (10)	
	Class activity (15)	
	Attendance (5)	
External Evaluation	End Semester Examination	50 Marks
Total		**100 Marks**

Method of assessment

Remembering (K1)	• The lowest level of questions requires students to recall information from the course content • Knowledge questions usually require students to identify information in the textbook. • Suggested Keywords: **Choose, Define, Find, How, Label, List, Match, Name, Omit, Recall, Relate, Select, Show, Spell, Tell, What, When, Where, Which, Who, Why**
Understanding (K2)	• Understanding off acts and ideas by comprehending organizing, comparing, translating, interpolating and interpreting in their own words. • The questions go beyond simple recall and require students to combined at altogether • Suggested Keywords: **Classify, Compare, Contrast, Demonstrate, Explain, Extend, Illustrate, Infer, Interpret, Outline, Relate, Rephrase, Show, Summarize, Translate**
Application (K3)	• Students have to solve problems by using / applying a concept learned in the classroom. • Students must use their knowledge to determine a exact response. • Suggested Keywords: **Apply, Build, Choose, Construct, Develop, Experiment with, Identify, Interview, Make use of, Model, Organize, Plan, Select, Solve, Utilize**
Analyze (K4)	• Analyzing the question is one that asks the students to breakdown something into its component parts. • Analyzing requires students to identify reasons causes or motives and reach conclusions or generalizations. • Suggested Keywords: **Analyze, Assume, Categorize, Classify, Compare, Conclusion, Contrast, Discover, Dissect, Distinguish, Divide, Examine, Function, Inference, Inspect, List, Motive, Relationships, Simplify, Survey, Take part in, Test for, Theme**
Evaluate (K5)	• Evaluation requires an individual to make judgment on something. • Questions to be asked to judge the value of an idea, a character , a work of art, or a solution to a problem. • Students are engaged in decision-making and problem–solving. • Evaluation questions do not have single right answers. • Suggested Keywords: **Agree, Appraise, Assess, Award, Choose, compare, Conclude, Criteria, Criticize, Decide, Deduct, Defend, Determine, Disprove, Estimate, Evaluate, Explain, Importance, Influence, Interpret, Judge, Justify, Mark, Measure, Opinion, Perceive, Prioritize, Prove, Rate, Recommend, Rule on, Select, Support, Value**
Create (K6)	• The questions of this category challenge students to get engaged in creative and original thinking. • Developing original ideas and problem solving skills • Suggested Keywords: **Adapt, Build, Change, Choose, Combine, Compile, Compose, Construct, Create, Delete, Design, Develop, Discuss, Elaborate, Estimate, Formulate, Happen, Imagine, Improve, Invent, Make up, Maximize, Minimize, Modify, Original, Originate, Plan, Predict, Propose, Solution, Solve, Suppose, Test, Theory**

BRIDGE COURSE

UNIT-I: Language Evolution

Machine Language, Assembly Language, High-Level Language. Translators: Compiler, Interpreter and Assembler, Linker, Loader, Study of Programming Language (Procedure Oriented, Object-Base, Event-Base), Programming Construction Tools Problem Analysis, Process Analysis, Conceptual Development of Solution. Development Tools: Algorithm: Types of Algorithm, Algorithm of Analysis, Advantage and Disadvantage of Algorithm Flowcharts: Concept, Symbols, Rules of Flowcharting, Advantage and Disadvantage of Flowchart.

UNIT –II: Introduction to Operating System

Operating System: Introduction, Purpose, Function, and Role of Operating System. Types of OS: Concepts of Batch, Multi Programmed, Time Sharing, Parallel, Real-Time, and Distributed. Operating System Structure System Components: Process, Main Memory, File I/O System, Secondary Storage Management, Networking, Protection System, Command Interpreter System, Operating System Services, System Call. Process and Job Control: Process and Types of Process, Process State.

UNIT-III: Computer Programming Concepts

Basics of Programming Language: Use of Character Set, Meaning of Keywords and Identifiers, Role of Data Types, Constants and Variables. Importance of Casting, Different Types of Operators and their Precedence, Expressions, Conditional Statements (One-Way, Two-Way, and Multi-Way Conditional), Looping Statements (for, while, do-while), Usage of break, continue, goto, and exit Statements.

UNIT-IV: Basic DBMS Concepts

Introduction: - Overview of database management system, limitations of data processing Environment, database approach, data independence, three levels of abstraction, DBMS structure. DMBS Model, DBMS Vs RDBMS Entity Relation Model: Entity, attributes, keys, relations, cardinality, participation, weak entities, ER diagram,

BOOKS

1. Pradeep K. Sinha and PritiSinha " Computer Fundamentals", BPB, 2007, ISBN-10:81-7656-752-3/13:978-81-7656-752-7
2. ChetanShrivastava " Fundamentals of Information Technology", Kalyani publishers,2002, ISBN-81-7663-576-6
3. DrMadhulikaJain,"Information Technology Concept", BPB,2006, ISBN -81-7656-276-9
4. E. Balguruswami, "Programming in ANSI C", TMH, 2009, ISBN-978-0-07-064822-7/0-07-064822-0
5. S.B Kishor," Information Technology", Das GanuPrakashan, ISBN 979-81-921757-9-9

SEMESTER - I

PROGRAMME: MCA	**BATCH:2026-28**
PART: A	**COURSE COMPONENT: CORE - I**
COURSE NAME: Operating Systems with Linux	**COURSE CODE: 26PMCA301**
SEMESTER: I	**MARKS:100**
CREDITS: 4	**TOTAL HOURS: 75**
THEORY	

COURSE OBJECTIVE:

To analyze and apply operating system concepts, resource management techniques, Linux system operations, distributed computing principles, security mechanisms, and Agile practices for designing efficient, secure, and reliable computing environments.

COURSE OUTCOMES:

1. Explain the concepts, structure, services, architectures, and types of operating systems including Linux operating system fundamentals and system commands.

2. Apply process management, process synchronization, CPU scheduling algorithms, inter-process communication, and Linux process management commands for efficient system operation.

3. Analyze deadlock handling techniques, memory management schemes, virtual memory concepts, and page replacement algorithms to improve resource utilization and system performance.

4. Examine file system implementation, disk management, RAID structures, secondary storage organization, and Linux file management system calls for secure and efficient data storage

5. Analyze distributed operating system concepts and security mechanisms, and apply Agile sprint planning techniques for collaborative management and secure deployment of distributed computing environments.

UNIT I **15 HOURS**

OVERVIEW: **Operating system:** Concept, Components of operating systems, operating system operations, protection and security, computing Environment Abstract view of OS: user view, system view, operating system services, **System calls**: concept, types of system calls**, Computer system Architecture:** Single processor systems, Multiprocessor systems, Types of operating systems: Batch operating systems, Multi programmed operating systems, Time shared operating systems, Real-time operating systems, Distributed operating systems **Process Management:** Process concept, operation on a process, cooperating process, IPC, Threads **Linux operating systems**: Introduction to Linux OS, Basic commands of Linux OS**.**

UNIT II **15 HOURS**

PROCESS MANAGEMENT: Process Synchronization: Introduction, Critical section problem, Synchronization hardware, semaphores, semaphore implementation, classical problem of synchronization, critical Region, Monitors, **CPU scheduling**: Basic concept, scheduling criteria, Scheduling algorithms, Multi-level queue Scheduling, Multilevel Feedback scheduling. **Linux operating systems:** Process management commands and system calls

UNIT III **15 HOURS**

DEADLOCK: Deadlock characterization, Resource Allocation graph, Deadlock prevention, avoidance, Detection and recovery and Bankers Algorithm. Memory management: Main memory, Contiguous memory allocation, Fragmentation, Paging and Segmentation. Virtual memory: Demand paging, page Replacement, Page Replacement Algorithm, Thrashing. Linux operating systems: Memory management commands and system calls

UNIT IV **15 HOURS**

File Devices and secondary storage management: File system interface: Concepts, access methods, Directory and disk structure, File system structure, File system Implementation, File system implementation, Directory Implementation, Allocation methods, Free space management, **Secondary Storage:** Mass storage structure, Disk structure, - RAID structure and RAID Levels. **Linux operating systems**: File management commands and system calls

UNIT V **15 HOURS**

DISTRIBUTED OPERATING SYSTEM: Distributed Computing System: Introduction – Advantages - Introduction to Distributed Computing Environment (DCE) **– Distributed Operating System:** Introduction - Issues in Designing a Distributed Operating System- Security : Potential Attacks – Cryptography- Authentication – Access Control – Digital Signatures. Case Study on : Agile sprint planning

RECOMMENDED TEXTBOOKS:

1. Abraham Silberschatz; Peter B. Galvin; Greg Gagne, Operating System Concepts, 9th Edition, John Wiley & Sons (Asia) Pvt Ltd
2. Pradeep K. Sinha, Distributed Operating Systems, IEEE Computer Society Press, 1997.
3. M.G. Venkateshmurthy, "Introduction to Unix and shell programming ", Pearson Education India, New Delhi, 2009.

REFERENCE BOOKS:

1. Sriram Narayan, Introduction to Agile Methods, InfoQ,2015
2. William Stallings, Operating Systems Internals and Design Principles, 7th Edition, Prentice Hall.
3. Andrew S. Tanenbaum ; Herbert Bos , Modern Operating Systems, 4th Edition , Pearson Education, Inc.

E-LEARNING RESOURCES:

1. https://www.google.com/search?q=operating+systems
2. https://www.javatpoint.com/linux-tutorial
3. https://www.usm.unimuenchen.de/people/puls/lessons/intro_general/Linux/Linux_for_beginners.pdf

GUIDELINES TO THE QUESTION PAPER SETTERS
QUESTION PAPER PATTERN

SECTION	QUESTION COMPONENT	NUMBERS	MARKS	TOTAL
A	*Answer all the questions*	*1-10*	*2*	*20*
B	*Answer any 5 out of 7 questions*	*11-17*	*7*	*35*
C	*Answer any 3 out of 5 questions(each in 1200 words)*	*18-22*	*15*	*45*
				100

BREAK UP OF QUESTIONS

UNITS	SECTION A	SECTION B	SECTION C
I	2	2	1
II	2	1	1
III	2	2	1
IV	2	1	1
V	2	1	1
TOTAL	10	7	5
	SECTION A - 10	**SECTION B - 7**	**SECTION C - 5**

PO – CO MAPPING

Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1.	**3**	**3**	**2**	**3**	**1**	**1**	**2**	**3**
CO2.	**3**	**3**	**3**	**3**	**2**	**2**	**2**	**3**
CO3	**3**	**3**	**3**	**3**	**1**	**2**	**2**	**3**
CO4	**3**	**3**	**3**	**3**	**2**	**2**	**3**	**3**
CO5	**3**	**3**	**3**	**3**	**3**	**3**	**3**	**3**
Average Contribution	3	3	2.8	3	1.8	2	2.4	3

PO – CO QUESTION PAPER MAPPING

CO No:	COURSE OUTCOMES	POs ADDRESSED	COGNITIVE LEVEL (K1 to K6)
CO1	Explain the concepts, structure, services, architectures, and types of operating systems including Linux operating system fundamentals and system commands.	1,2,4,8	1 to 5
CO2	Apply process management, process synchronization, CPU scheduling algorithms, inter-process communication, and Linux process management commands for efficient system operation.	1,2,3,4,8	1 to 5
CO3	Analyze deadlock handling techniques, memory management schemes, virtual memory concepts, and page replacement algorithms to improve resource utilization and system performance.	1,2,3,4,8	1 to 5

CO4	Examine file system implementation, disk management, RAID structures, secondary storage organization, and Linux file management system calls for secure and efficient data storage	1,2,3,4,7,8	1 to 5
CO5	Analyze distributed operating system concepts and security mechanisms, and apply Agile sprint planning techniques for collaborative management and secure deployment of distributed computing environments.	1,2,3,4,5,6,7,8	1 to 5
K1= Remember, K2= Understand, K3= Apply, K4=Analyse, K5= Evaluate, K6= Create			

Continuous Internal Assessment - Components			
Assessment Type	**Nature**	**Maximum Marks**	**Weightage**
Mid-Semester Examination	Pen and Paper test for 90 minutes	50	10
Model Examination	Written examination for 180 minutes	100	10
Assignment	Any topic related to the course	10	5
Workbook/recordbook/ notebook	Handwritten Class Notes	5	5
Class activity 1	Quiz with MCQ,	10	5
Class activity 2	Seminar / Presentation	10	5
Class activity 3	Surprise Class Tests	10	5
Attendance	Marks will be given based on the attendance percentage of respective course		5
Total			50

PROGRAMME: MCA	**BATCH:2026-28**
PART: A	**COURSE COMPONENT: CORE - II**
COURSE NAME: Advanced Database Management Systems	**COURSE CODE: 26PMCA302**
SEMESTER: I	**MARKS:100**
CREDITS: 4	**TOTAL HOURS: 75**
THEORY	

COURSE OBJECTIVE:

To develop the ability to analyze, design, implement, and manage relational and NoSQL database systems by applying data modeling techniques, normalization principles, SQL/PLSQL programming, database security practices, and modern database management approaches for scalable and reliable data-driven applications

COURSE OUTCOMES:

1. Explain the concepts of the relational model, ACID properties, entity-relationship modeling, constraints, keys, and the transformation of E-R diagrams into relational schemas for effective database design.
2. Apply functional dependencies, normalization techniques, and SQL data definition, manipulation, and transaction control commands to design and maintain consistent relational databases.
3. Construct SQL queries using filtering, sorting, aggregate functions, joins, subqueries, views, and access control mechanisms to retrieve and manage database information effectively.
4. Develop PL/SQL programs incorporating control structures, cursors, subprograms, triggers, and exception handling mechanisms to implement database applications and automate database operations.
5. Evaluate the suitability of NoSQL and MongoDB technologies for scalable database applications by analyzing document models, querying techniques, sharding strategies, Agile project management approaches, and security, privacy, and ethical governance considerations.

UNIT I **15 HOURS**

Introduction to Relational Model : Definition of Relation, Tuple, Attribute, Domain, Database Schema – ACID Properties - **Database Design and E-R Model:** Overview of the Design Process, The Entity-Relationship Model : Entity Sets, Relationship sets, Attributes – Constraints : Mapping Cardinalities, Keys : Super Key, Candidate Key, Primary Key, Foreign Key – Entity- Relationship Diagrams: Basic Structure, Mapping cardinality, Roles, Weak Entity Sets- Reduction of E-R Diagram to Relational Schemas – Entity-Relationship Design issues – Extended E-R features : Specialization, Generalization, Attribute Inheritance, Aggregation, Reduction to Relational Schema

UNIT II **15 HOURS**

Functional Dependencies – Anomalies in Databases(Insertion, Updation and Deletion). Normal forms based on Primary keys (1NF, 2NF,3NF)- Structured Query Language : Overview, Data types, Basic Schema Definition (DDL Commands – CREATE, ALTER, DROP, TRUNCATE) – Modification of data in the Database (DML Commands – INSERT, UPDATE, DELETE) – Transactions (TCL Commands – COMMIT, ROLLBACK, SAVEPOINT).

UNIT III **15 HOURS**

SQL Query (Select Statement): Basic Query operations and operations that include Filtering and Sorting, Aggregate function, Group By and Having clauses – Joins: Inner Join, Outer Join, Self Join – Nested Subqueries and Correlated Subqueries – Views – Authorization and Access Control

UNIT IV **15 HOURS**

Overview of PL/SQL: Advantages – Main Features – PL/SQL Language Fundamentals: Database Character Set- Lexical Units: Delimiters, Identifiers, Literals, Comments-Declarations –Assigning values to Variables – Expressions – PL/SQL Control Statements –Cursor Overview: Implicit Cursors, Explicit Cursors – Processing Query result sets(Handling Single- Row Result Sets) – PL/SQL Subprograms: Reasons to use Subprograms, Subprogram Parts, Return Statement, Subprogram Parameter Passing Methods, Subprogram Parameter Modes, Default Values for IN Subprogram Parameters- PLSQL Triggers : Overview- Reasons to user Triggers , DML Triggers, Trigger Design Guidelines, Trigger Enabling and Disabling- PL/SQL Error Handling: Overview of Exception Handling, Predefined Exceptions, User-Defined Exceptions, Raising Exceptions Explicitly, RAISE_APPLICATION_ERROR Procedure, Unhandled Exceptions.

UNIT V **15 HOURS**

NoSQL and document databases: Scalability , Implementations (Document Databases, Key Value Stores, Wide column Stores, Graph Stores), What is missing in NoSQL databases – Mongo DB : Database Structure : Documents, Collections, Thinking in Documents and referencing documents, Embedding documents, Document Design strategy, sharding – MongoDB installation on Windows- The Mongo Shell: searching for help, database creation, dropping databases, collections, capped collections – Manipulating Documents : Simple data retrieval, Inserting documents, updating a document, Deleting a document – Data Retrieval : Querying a Collection, Projections, Sorting, limiting the output – Case Study on : **Agile Database Project Management,** Database Security, Privacy & Ethical Governance

RECOMMENDED TEXTBOOKS:

1. Henry F.Korth, Abraham Silberschatz, S. Sudharshan,"*Database System Concepts"*, 6th Edition, 2011, McGrawHill.(Unit – 1,2,3)
2. Elmasri, Navathe, " Fundamentals of Database Systems", 7th Edition, 2016, Pearson (Unit- 2)
3. Oracle Database Database PL/SQL Language Reference (Unit-4)
4. Zoran Maksimovic, MongoDB 3, Syncfusion, Techchnology Resource Portal (Unit -5)

REFERENCE BOOKS:

1. Watt, A. and Eng, N., *Database Design – 2nd Edition*, BCcampus Open Education, Open Textbook (Free Download).
2. Raghu Ramakrishnan ,"Database Management Systems", McGraw Hill Publishing Company Limited, 2004.

E-LEARNING RESOURCES:

1. https://www.tutorialspoint.com/plsql/index.htm
2. https://www.javatpoint.com/dbms-tutor

GUIDELINES TO THE QUESTION PAPER SETTERS
QUESTION PAPER PATTERN

SECTION	QUESTION COMPONENT	NUMBERS	MARKS	TOTAL
A	*Answer all the questions*	*1-10*	*2*	*20*
B	*Answer any 5 out of 7 questions*	*11-17*	*7*	*35*
C	*Answer any 3 out of 5 questions(each in 1200 words)*	*18-22*	*15*	*45*
				100

BREAK UP OF QUESTIONS

UNITS	SECTION A	SECTION B	SECTION C
I	2	2	1
II	2	1	1
III	2	2	1
IV	2	1	1
V	2	1	1
TOTAL	10	7	5
	SECTION A - 10	**SECTION B - 7**	**SECTION C - 5**

PO – CO MAPPING

Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1.	3	3	3	2	1	1	2	2
CO2.	3	3	3	3	1	1	2	2
CO3	3	3	3	3	1	1	3	2
CO4	3	3	3	3	2	1	3	2
CO5	3	3	3	3	3	3	3	3
Average Contribution	3	3	3	2.8	1.6	1.4	2.6	2.2

PO – CO QUESTION PAPER MAPPING

CO No:	COURSE OUTCOMES	POs ADDRESSED	COGNITIVE LEVEL (K1 to K6)
CO1	**Explain** the concepts of the relational model, ACID properties, entity-relationship modeling, constraints, keys, and the transformation of E-R diagrams into relational schemas for effective database design.	1,2,3	1 to 5
CO2	**Apply** functional dependencies, normalization techniques, and SQL data definition, manipulation, and transaction control commands to design and maintain consistent relational databases.	1,2,3,4	1 to 5

CO3	**Construct** SQL queries using filtering, sorting, aggregate functions, joins, subqueries, views, and access control mechanisms to retrieve and manage database information effectively.	1,2,3,4,7	1 to 5
CO4	**Develop** PL/SQL programs incorporating control structures, cursors, subprograms, triggers, and exception handling mechanisms to implement database applications and automate database operations.	1,2,3,4,7	1 to 5
CO5	*Evaluate the suitability of NoSQL and MongoDB technologies for scalable database applications by analyzing document models, querying techniques, sharding strategies, Agile project management approaches, and security, privacy, and ethical governance considerations.*	1,2,3,4,5,6,7,8	1 to 5
K1= Remember, K2= Understand, K3= Apply, K4=Analyse, K5= Evaluate, K6= Create			

Continuous Internal Assessment - Components			
Assessment Type	**Nature**	**Maximum Marks**	**Weightage**
Mid-Semester Examination	Pen and Paper test for 90 minutes	50	10
Model Examination	Written examination for 180 minutes	100	10
Assignment	Any topic related to the course	10	5
Workbook/recordbook/ notebook	Handwritten Class Notes	5	5
Class activity 1	Quiz with MCQ,	10	5
Class activity 2	Seminar / Presentation	10	5
Class activity 3	Surprise Class Tests	10	5
Attendance	Marks will be given based on the attendance percentage of respective course		5
Total			50

PROGRAMME: MCA	**BATCH:2026-28**
PART: A	**COURSE COMPONENT: CORE - III**
COURSE NAME: Advanced Java Technologies	**COURSE CODE: 26PMCA303**
SEMESTER: I	**MARKS:100**
CREDITS: 4	**TOTAL HOURS: 75**
THEORY	

COURSE OBJECTIVE:

To develop the ability to design, develop, and deploy secure, database-driven, and enterprise web applications using object-oriented Java programming, multithreading, JDBC, Servlets, JSP, Spring Boot, Agile project management practices, and ethical software engineering principles to address real-world computing challenges.

COURSE OUTCOMES:

1. Apply Java programming concepts including data types, operators, control structures, classes, objects, constructors, methods, strings, and command-line arguments to develop object-oriented applications.
2. Develop Java applications using inheritance, packages, interfaces, and exception handling mechanisms to promote code reusability, abstraction, and robust error management.
3. Develop multithreaded and database-driven Java applications using thread management, synchronization techniques, and JDBC programming for efficient data processing and resource utilization.
4. Construct dynamic web applications using Servlets, JSP, MVC architecture, session management, and HTTP request-response handling mechanisms.
5. Evaluate and develop Spring Boot-based enterprise applications by integrating RESTful services, MVC architecture, Agile project management practices, and secure and ethical software development principles

UNIT I **15 HOURS**

An Overview of Java- Data types, Variables and Arrays – Operators- Control Statements - classes and Class Fundamentals – Declaring Objects – Introducing Methods – Constructors – constructor overloading-‘this’ Keyword – METHODS AND CLASSES: Overloading Methods – Objects as Parameters – Returning Objects – Access Control – Static and Final - Exploring the String Class – StringBuffer Class-Command Line Arguments

UNIT II **15 HOURS**

INHERITANCE: Inheritance Basics – Using Super – Creating Multilevel Hierarchy – Method Overriding - Abstract Classes – Final with Inheritance – Packages and Interface :Packages and Member Access , importing packages Interfaces- Implementing and Extending Interfaces. EXCEPTION HANDLING: Exception Types – Using try, catch, throw, throws, finally.

UNIT III **15 HOURS**

Multithreaded Programming -The Java Thread Model- The Main Thread- Creating a Thread- Creating Multiple Threads- Thread Priorities- Synchronization- Interthread Communication. JDBC Programming: JDBC Architecture, Types of JDBC Drivers, Introduction to major JDBC Classes and Interface, Creating simple JDBC Application, Types of Statement (Statement Interface, Prepared Statement, Callable Statement), Exploring Result Set Operations

UNIT IV **15 HOURS**

SERVLET AND JSP: Introduction to Web Applications and Java EE – HTTP Protocol and Request/Response Model – Servlet Lifecycle: init(), service (), destroy () – Handling Form Data (GET, POST methods) – Introduction to JSP (Java Server Pages) – JSP Directives, Scripting Elements, Action Tags – MVC Architecture using Servlet and JSP – Session Management: Cookies and HttpSession

UNIT V **15 HOURS**

Modern Web Applications using Spring Boot: Introduction to Spring Framework and Spring Boot, Spring Boot Architecture and Project Setup, Creating RESTful Web Services with Spring Boot, Spring Boot MVC and Template Integration, Building a Simple CRUD Application, Spring Boot DevTools and Application Deployment – Case Study on : **Agile Software Project Management for Enterprise Java Applications - Secure and Ethical Enterprise Application Development**

.

RECOMMENDED TEXTBOOKS:

1. Herbert Schildt - The Complete Reference Java - Tata McGraw Hill Publishing Company Limited Edition 11, 2018.
2. Java ™ Servlet Programming- Jason Hunter with William Crawford, Published by O'Reilly & Associates, Inc.
3. Database Programming with JDBC and Java , O'Reilly & Associates, Second Edition
4. Craig Walls, "Spring in Action" - Manning Publications, 4th Edition (2014).

REFERENCE BOOKS:

1. Rajib Mall, Fundamentals of Software Engineering, 5th Edition, PHI Learning Pvt. Ltd., 2018.
2. Murachs, Java Servlet and JSP, 2nd Edition, Joel Murach Andrea Steelman, Mike Murach & Associates Inc.
3. Kogent Solution Inc, "Java 6 Programming Black Book", Dreamtech Press, 20l7.
4. J2EE the Complete Reference, First Edition by Jim Keogh, Tata McGraw Hill, 20l2.

E-LEARNING RESOURCES:

1. https://docs.oracle.com/javase/tutorial/
2. www.roseindia.net
3. https://www.edx.org/learn/java
4. https://docs.oracle.com/en/middleware/standalone/weblogic-server/14.1.1.0/wbapp/basics.html

GUIDELINES TO THE QUESTION PAPER SETTERS
QUESTION PAPER PATTERN

SECTION	QUESTION COMPONENT	NUMBERS	MARKS	TOTAL
A	*Answer all the questions*	*1-10*	*2*	*20*
B	*Answer any 5 out of 7 questions*	*11-17*	*7*	*35*
C	*Answer any 3 out of 5 questions(each in 1200 words)*	*18-22*	*15*	*45*
				100

BREAK UP OF QUESTIONS

UNITS	SECTION A	SECTION B	SECTION C
I	2	2	1
II	2	1	1
III	2	2	1
IV	2	1	1
V	2	1	1
TOTAL	10	7	5
	SECTION A - 10	**SECTION B - 7**	**SECTION C - 5**

PO – CO MAPPING

Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1.	3	3	3	2	1	1	1	2
CO2.	3	3	3	3	1	1	2	2
CO3	3	3	3	3	2	1	2	2
CO4	3	3	3	3	2	1	3	2
CO5	3	3	3	3	3	3	3	3
Average Contribution	3	3	3	2.8	1.8	1.4	2.2	2.2

PO – CO QUESTION PAPER MAPPING

CO No:	COURSE OUTCOMES	POs ADDRESSED	COGNITIVE LEVEL (K1 to K6)
CO1	**Apply** Java programming concepts including data types, operators, control structures, classes, objects, constructors, methods, strings, and command-line arguments to develop object-oriented applications.	1,2,3	1 to 5
CO2	**Develop** Java applications using inheritance, packages, interfaces, and exception handling mechanisms to promote code reusability, abstraction, and robust error management.	1,2,3,4	1 to 5

CO3	**Develop** multithreaded and database-driven Java applications using thread management, synchronization techniques, and JDBC programming for efficient data processing and resource utilization.	1,2,3,4	1 to 5
CO4	**Construct** dynamic web applications using Servlets, JSP, MVC architecture, session management, and HTTP request-response handling mechanisms.	1,2,3,4,7	1 to 5
CO5	***Evaluate and develop Spring Boot-based enterprise applications by integrating RESTful services, MVC architecture, Agile project management practices, and secure and ethical software development principles***	1,2,3,4,5,6,7,8	1 to 5
K1= Remember, K2= Understand, K3= Apply, K4=Analyse, K5= Evaluate, K6= Create			

Continuous Internal Assessment - Components			
Assessment Type	**Nature**	**Maximum Marks**	**Weightage**
Mid-Semester Examination	Pen and Paper test for 90 minutes	50	10
Model Examination	Written examination for 180 minutes	100	10
Assignment	Any topic related to the course	10	5
Workbook/recordbook/ notebook	Handwritten Class Notes	5	5
Class activity 1	Quiz with MCQ,	10	5
Class activity 2	Seminar / Presentation	10	5
Class activity 3	Surprise Class Tests	10	5
Attendance	Marks will be given based on the attendance percentage of respective course		5
Total			50

PROGRAMME: MCA	**BATCH:2026-28**
PART: A	**COURSE COMPONENT: ELECTIVE - I**
COURSE NAME: Cyber Security and Networking	**COURSE CODE: 26PMCA3E1**
SEMESTER: I	**MARKS:100**
CREDITS: 3	**TOTAL HOURS: 60**
THEORY	

COURSE OBJECTIVE:

To enable learners to understand, apply, analyze, evaluate, and design secure computing environments by integrating cyber security fundamentals, network architectures, TCP/IP protocols, network security mechanisms, cryptographic techniques, authentication systems, and Agile cyber security project management practices for ensuring confidentiality, integrity, and availability of information systems.

COURSE OUTCOMES:

1. Explain the fundamentals of cyber security, cyber threats and attacks, data communication concepts, network topologies, transmission media, and the OSI reference model for secure communication systems.
2. Apply TCP/IP protocol suite concepts, IP addressing, subnetting, ARP/RARP mechanisms, VLANs, and VPN technologies to configure and manage secure network communications.
3. Analyze network security requirements, security policies, network attacks, and secure communication protocols to identify vulnerabilities and enhance network protection mechanisms.
4. Evaluate cryptographic techniques, public key infrastructures, digital certificates, and secure communication mechanisms for protecting data confidentiality, integrity, and authenticity.
5. Evaluate the effectiveness of hash functions, message authentication codes, digital signature schemes, authentication protocols, and Agile cyber security project management practices in ensuring secure, reliable, and efficient information systems.

UNIT I **12 HOURS**

Introduction to Cyber security and Threats: Understanding Cyber security: What, Why, and Scope, Cyber Threats and Attacks (Malware, Phishing, Ransomware, Components- Data Communications: protocols and standards- Basic Concepts: Line Configuration- Topology- Transmission mode- Categories of Networks Transmission Mode: Guided Media-OSI Layer

UNIT II **12 HOURS**

TCP/IP Protocol Suite: Overview of TCP/IP – Network Layer- Classes – Subnetting – ARP – RARP- VLANs –VPN - Types of VPNs – Site-to-site VPN, Remote-access VPN. VPN Protocols – PPTP, L2TP, OpenVPN, IPSec

UNIT III **12 HOURS**

Network Security Protocols and Monitoring Tools: Fundamentals of Network Security- Introduction to network security, Types of Network Attacks-Network Security Policies. Network Protocols and Security –TCP/IP Security – Secure Communication Protocols-Wireless Security

UNIT IV **12 HOURS**

Block ciphers & public key cryptography: Encryption Techniques – Symmetric (AES, DES) vs. Asymmetric (RSA, ECC). Digital Signatures & Certificates – PKI, SSL/TLS certificates, certificate authorities. Secure Email & Data Transmission – PGP, S/MIME, Secure File Transfer Protocols (SFTP)

UNIT V **12 HOURS**

Hash functions and digital signatures: Authentication requirement – Authentication function – MAC – Hash function – Security of hash function and MAC –MD5 - SHA - HMAC – CMAC - Digital signature and authentication protocols – DSS – EI Gamal – Schnorr. Case Study on : Agile Cyber Security Project Management

RECOMMENDED TEXTBOOKS:

1. William Stallings, Cryptography and Network Security, 6th Edition, Pearson Education, March 2013.
2. Data communication and Networking 6th Edition by Author Behrouz a Forouzan
3. Cyber Security for beginners by Raef Meeuwisse
4. Security Warrior by Cyrus Piekari, Anton Chuvakin, 2nd ed., Oreilly Publishers, 2005.
5. Cyber security essentials by Charles J. Brooks, Christopher Grow, Philip Craig and Donald Short

REFERENCE BOOKS:

1. Wiley, D., Amado, M., Ashton, K., Ashton, S., Bostwick, J., Nan, V., Nisse, T., Randall, D., Clements, G., Drysdale, J., Francis, J., Harrison, B., Rino, J., Robinson, J., & Snyder, A. (2018). *Project Management for Instructional Designers*. David Wiley. (Open Educational Resource). (Marshall University Research Portal)
2. Behrouz A. Ferouzan, "Cryptography & Network Security", Tata Mc Graw Hill, 2007.
3. Man Young Rhee, "Internet Security: Cryptographic Principles", "Algorithms and Protocols", WileyPublications, 2003.
4. Charles Pfleeger, "Security in Computing", 4th Edition, Prentice Hall of India, 2006.
5. Ulysess Black, "Internet Security Protocols", Pearson Education Asia, 2000.
6. Charlie Kaufman and Radia Perlman, Mike Speciner, "Network Security, Second Edition, Private Communication in Public World", PHI 2002.
7. Bruce Schneier and Neils Ferguson, "Practical Cryptography", First Edition, Wiley Dreamtech India Pvt Ltd, 2003.
8. Douglas R Simson "Cryptography – Theory and practice", First Edition, CRC Press, 1995.

GUIDELINES TO THE QUESTION PAPER SETTERS
QUESTION PAPER PATTERN

SECTION	QUESTION COMPONENT	NUMBERS	MARKS	TOTAL
A	*Answer all the questions*	*1-10*	*2*	*20*
B	*Answer any 5 out of 7 questions*	*11-17*	*7*	*35*
C	*Answer any 3 out of 5 questions(each in 1200 words)*	*18-22*	*15*	*45*
				100

BREAK UP OF QUESTIONS

UNITS	SECTION A	SECTION B	SECTION C
I	2	2	1
II	2	1	1
III	2	2	1
IV	2	1	1
V	2	1	1
TOTAL	10	7	5
	SECTION A - 10	**SECTION B - 7**	**SECTION C - 5**

PO – CO MAPPING

Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1.	3	3	1	2	1	1	3	2
CO2.	3	3	3	3	1	1	3	2
CO3	3	3	3	3	1	1	3	2
CO4	3	3	3	3	1	1	3	2
CO5	3	3	3	3	3	3	3	3
Average Contribution	3	3	2.6	2.8	1.4	1.4	3	2.2

PO – CO QUESTION PAPER MAPPING

CO No:	COURSE OUTCOMES	POs ADDRESSED	COGNITIVE LEVEL (K1 to K6)
CO1	**Explain** the fundamentals of cyber security, cyber threats and attacks, data communication concepts, network topologies, transmission media, and the OSI reference model for secure communication systems.	1,2,7	1 to 5
CO2	**Apply** TCP/IP protocol suite concepts, IP addressing, subnetting, ARP/RARP mechanisms, VLANs, and VPN technologies to configure and manage secure network communications.	1,2,3,4,7	1 to 5

CO3	**Analyze** network security requirements, security policies, network attacks, and secure communication protocols to identify vulnerabilities and enhance network protection mechanisms.	1,2,3,4,7	1 to 5
CO4	**Evaluate** cryptographic techniques, public key infrastructures, digital certificates, and secure communication mechanisms for protecting data confidentiality, integrity, and authenticity.	1,2,3,4,7	1 to 5
CO5	Evaluate the effectiveness of hash functions, message authentication codes, digital signature schemes, authentication protocols, and Agile cyber security project management practices in ensuring secure, reliable, and efficient information systems.	1,2,3,4,5,6,7,8	1 to 5
K1= Remember, K2= Understand, K3= Apply, K4=Analyse, K5= Evaluate, K6= Create			

Continuous Internal Assessment - Components			
Assessment Type	**Nature**	**Maximum Marks**	**Weightage**
Mid-Semester Examination	Pen and Paper test for 90 minutes	50	10
Model Examination	Written examination for 180 minutes	100	10
Assignment	Any topic related to the course	10	5
Workbook/recordbook/ notebook	Handwritten Class Notes	5	5
Class activity 1	Quiz with MCQ,	10	5
Class activity 2	Seminar / Presentation	10	5
Class activity 3	Surprise Class Tests	10	5
Attendance	Marks will be given based on the attendance percentage of respective course		5
Total			50

PROGRAMME: MCA	**BATCH:2026-28**
PART: A	**COURSE COMPONENT: ELECTIVE - I**
COURSE NAME: Cyber Security and Ethical Hacking	**COURSE CODE: 26PMCA3E5**
SEMESTER: I	**MARKS:100**
CREDITS: 3	**TOTAL HOURS: 60**
THEORY	

COURSE OBJECTIVE:

To apply, analyze, and evaluate ethical hacking methodologies, reconnaissance techniques, system and network security assessment tools, web application security testing, and password auditing mechanisms to identify vulnerabilities, assess security risks, and recommend appropriate countermeasures for securing computing environments

COURSE OUTCOMES:

1. Apply ethical hacking methodologies, reconnaissance techniques, vulnerability assessment procedures, and penetration testing phases to identify and assess security weaknesses in computing systems.
2. Analyze information gathered through reconnaissance and social engineering techniques using appropriate security tools and recommend suitable countermeasures to mitigate security risks.
3. Analyze Windows security architecture, user account vulnerabilities, password attacks, registry manipulation techniques, and system security weaknesses to assess potential threats and protection mechanisms.
4. Apply network scanning, enumeration, domain analysis, port investigation, wireless security assessment, and network vulnerability identification techniques to evaluate network security posture.
5. Evaluate web application vulnerabilities, browser-based attack vectors, password cracking techniques, and security testing methods to recommend effective measures for securing web applications and authentication systems.

UNIT I **12 HOURS**

Introduction to Ethical Hacking and Methodology Introduction to Ethical Hacking: Planning and performing attacks - Five stages of Ethical Hacking – Ways to conduct Ethical Hacking - Types of hacking – Hacker types Hacking Methodology: Setting the stage for testing - Seeing what others see - Scanning systems - Determining what's running on open ports - Assessing vulnerabilities - Penetrating the system

UNIT II **12 HOURS**

Reconnaissance and Social Engineering Reconnaissance: HTTrack - Google Directives – Harvester - Whois – Netcraft –Host – Fierce and other tools to extract information from DNS - Extracting Information From E-mail Servers – MetaGooFil – ThreatAgent Social Engineering: Implications of

Social Engineering - Performing Social Engineering attacks – Social Engineering countermeasures

UNIT III **12 HOURS**

Windows Hacking and Security- Security architecture of Windows – Windows User Account Architecture and Attack – Windows vulnerabilities – Detecting null sessions - Cracking BIOS and Windows Password –Changing Windows Visuals – Editing the OS – Registry - Other system files – Checking share permissions

UNIT IV **12 HOURS**

Network Hacking Network infrastructure vulnerabilities - Scanning, poking and prodding the network – IP address – DNS – NslookUp – Port scanning and surfing - Sockets – PING – Netstat – Getting information about a domain – FTP Port and using FTP Client – FTP Commands –Detecting common Router, Switch and Firewall weaknesses – Discovering wireless network attacks and taking countermeasures -

UNIT V **12 HOURS**

Web Hacking and Cracking Passwords Web Hacking: Get, Post and Head methods – Hacking from Web Browser – Post Dial Up Screen hacking – Web security testing tools – Web Vulnerabilities – Minimizing Web security risks Cracking Passwords: Password vulnerabilities – Cracking passwords – Password cracking Count measures.

RECOMMENDED TEXTBOOKS:

1. Ankit Fadia, Ethical Hacking, Second Edition, Macmillan India Ltd, 2006
2. Kevin Beaver, Hacking for Dummies, Fifth Edition, Wiley, 2016
3. CEH official Certfied Ethical Hacking Review Guide, Wiley India Edition, 2015.

REFERENCE BOOKS:

1. Kenneth C.Brancik, Insider Computer Fraud, Auerbach Publications Taylor & Francis Group, 2008.
2. Patrick Engebretson, The Basics of Hacking and Penetration Testing, Second Edition,Syngress, 2013

GUIDELINES TO THE QUESTION PAPER SETTERS
QUESTION PAPER PATTERN

SECTION	QUESTION COMPONENT	NUMBERS	MARKS	TOTAL
A	*Answer all the questions*	*1-10*	*2*	*20*
B	*Answer any 5 out of 7 questions*	*11-17*	*7*	*35*
C	*Answer any 3 out of 5 questions(each in 1200 words)*	*18-22*	*15*	*45*
				100

BREAK UP OF QUESTIONS

UNITS	SECTION A	SECTION B	SECTION C
I	2	2	1
II	2	1	1
III	2	2	1
IV	2	1	1
V	2	1	1
TOTAL	10	7	5
	SECTION A - 10	**SECTION B - 7**	**SECTION C - 5**

PO – CO MAPPING

Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1.	3	3	3	3	2	1	3	3
CO2.	2	3	3	3	2	1	3	3
CO3	3	3	3	3	1	1	3	3
CO4	3	3	3	3	2	1	3	3
CO5	3	3	3	3	2	1	3	3
Average Contribution	2.8	3	3	3	1.8	1	3	3

PO – CO QUESTION PAPER MAPPING

CO No:	COURSE OUTCOMES	POs ADDRESSED	COGNITIVE LEVEL (K1 to K6)
CO1	Apply ethical hacking methodologies, reconnaissance techniques, vulnerability assessment procedures, and penetration testing phases to identify and assess security weaknesses in computing systems.	1,2,3,4,7,8	1 to 5
CO2	Analyze information gathered through reconnaissance and social engineering techniques using appropriate security tools and recommend suitable countermeasures to mitigate security risks.	2,3,4,7,8	1 to 5
CO3	Analyze Windows security architecture, user account vulnerabilities, password attacks, registry manipulation techniques, and system security weaknesses to assess potential threats and protection mechanisms.	1,2,3,4,7,8	1 to 5
CO4	Apply network scanning, enumeration, domain analysis, port investigation, wireless security assessment, and network vulnerability identification techniques to evaluate network security posture.	1,2,3,4,7,8	1 to 5
CO5	Evaluate web application vulnerabilities, browser-based attack vectors, password cracking techniques, and security testing methods to recommend effective measures for securing web applications and authentication systems.	1,2,3,4,7,8	1 to 5
K1= Remember, K2= Understand, K3= Apply, K4=Analyse, K5= Evaluate, K6= Create			

Continuous Internal Assessment - Components			
Assessment Type	**Nature**	**Maximum Marks**	**Weightage**
Mid-Semester Examination	Pen and Paper test for 90 minutes	50	10
Model Examination	Written examination for 180 minutes	100	10
Assignment	Any topic related to the course	10	5
Workbook/recordbook/ notebook	Handwritten Class Notes	5	5
Class activity 1	Quiz with MCQ,	10	5
Class activity 2	Seminar / Presentation	10	5
Class activity 3	Surprise Class Tests	10	5
Attendance	Marks will be given based on the attendance percentage of respective course		5
Total			50

PROGRAMME: MCA	**BATCH:2026-28**
PART: A	**COURSE COMPONENT: ELECTIVE - I**
COURSE NAME: Ethical Hacking and Penetration Testing	**COURSE CODE: 26PMCA3E6**
SEMESTER: I	**MARKS:100**
CREDITS: 3	**TOTAL HOURS: 60**
THEORY	

COURSE OBJECTIVE:

To develop the ability to analyze, evaluate, and apply ethical hacking and penetration testing methodologies, security assessment techniques, exploit development concepts, web application security testing, malware analysis, and client-side attack detection mechanisms for identifying, assessing, and mitigating cybersecurity vulnerabilities in computing environments.

COURSE OUTCOMES:

1. Analyze ethical hacking principles, vulnerability assessment methodologies, penetration testing processes, and social engineering attack techniques to identify security weaknesses and recommend appropriate countermeasures.
2. Apply physical penetration testing techniques, insider threat assessment methods, and Metasploit framework tools to evaluate organizational security and demonstrate controlled exploitation scenarios.
3. Analyze the planning, execution, and reporting phases of penetration testing and evaluate Linux and Windows exploit development techniques for identifying system vulnerabilities and security risks.
4. Analyze web application security vulnerabilities, including OWASP Top Ten risks, SQL Injection, Cross-Site Scripting, and vulnerability analysis techniques to assess application security posture and recommend mitigation strategies.
5. Evaluate client-side browser exploits and malware analysis techniques to investigate emerging threats, assess attack vectors, and propose effective defensive mechanisms for secure computing environments

UNIT I **12 HOURS**

Introduction Ethics of Ethical Hacking: Why you need to understand your enemy's tactics, recognizing the gray areas in security, Vulnerability Assessment and Penetration Testing. Penetration Testing and Tools: Social Engineering Attacks: How a social engineering attack works, conducting a social engineering attack, common attacks used in penetration testing, preparing yourself for face-to-face attacks,defending against social engineering attacks.

UNIT II **12 HOURS**

Physical Penetration Attacks: Need of Physical Penetration, Conducting a Physical Penetration, Common Ways into a Building, Defending Against Physical Penetrations. Insider Attacks: Conducting an Insider Attack, Defending Against Insider Attacks. Metasploit: The Big Picture, Getting Metasploit, Using the Metasploit Console to Launch Exploits, Exploiting Client-Side Vulnerabilities with Metasploit, Penetration Testing with Metasploit's Meterpreter, Automating and Scripting Metasploit, Going Further with Metasploit

UNIT III **12 HOURS**

Managing a Penetration Test: Planning a Penetration Test, Structuring a Penetration Testing Agreement, Execution ofa Penetration Test, Information Sharing During a Penetration Test, Reporting the Results of a Penetration Test.Basic Linux Exploits: Stack Operations, Buffer Overflows, Local Buffer Overflow Exploits, Exploit Development Process. Windows Exploits: Structured Exception Handling (SEH), Understanding Windows Memory Protections (XP SP3, Vista 7 And Server 2008), Bypassing Windows Memory Protections.

UNIT IV **12 HOURS**

Web Application Security Vulnerabilities: Overview of Top Web Application Security Vulnerabilities, Injection Vulnerabilities, Cross-Site Scripting Vulnerabilities, The Rest of the OWASP Top Ten, SQL Injection Vulnerabilities, Cross-Site Scripting Vulnerabilities Vulnerability Analysis: Passive Analysis: Source Code Analysis, Binary Analysis.

UNIT V **12 HOURS**

Client-Side Browser Exploits: Why Client-Side Vulnerabilities are Interesting, Internet Explorer Security Concepts, History of Client- Side Exploits and Latest Trends, Finding New Browser-Based Vulnerabilities, Heap Spray to Exploit, Protecting Yourself from Client-Side Exploit. Malware Analysis: Collecting Malware and Initial Analysis: Malware, Latest Trends in Honeynet Technology, Catching Malware: Setting the Trap, Initial Analysis of Malware

RECOMMENDED TEXTBOOKS:

1. Gray Hat Hacking - The Ethical Hackers Handbook, Allen Harper, Shon Harris, Jonathan Ness, Chris Eagle, Gideon Lenkey, and Terron Williams, 3rd Edition, Tata McGraw-Hill.
2. The Web Application Hacker's Handbook-Discovering and Exploiting Security flaws, Dafydd Suttard, Marcus pinto, 1st Edition, Wiley Publishing.
3. Penetration Testing: Hands-on Introduction to Hacking", Georgia Weidman, 1st Edition, No Starch Press.

REFERENCE BOOKS:

1. The Pen Tester Blueprint-Starting a Career as an Ethical Hacker ", L. Wylie, Kim Crawly, 1st Edition, Wiley Publications.

GUIDELINES TO THE QUESTION PAPER SETTERS
QUESTION PAPER PATTERN

SECTION	QUESTION COMPONENT	NUMBERS	MARKS	TOTAL
A	*Answer all the questions*	*1-10*	*2*	*20*
B	*Answer any 5 out of 7 questions*	*11-17*	*7*	*35*
C	*Answer any 3 out of 5 questions(each in 1200 words)*	*18-22*	*15*	*45*
				100

BREAK UP OF QUESTIONS

UNITS	SECTION A	SECTION B	SECTION C
I	2	2	1
II	2	1	1
III	2	2	1
IV	2	1	1
V	2	1	1
TOTAL	10	7	5
	SECTION A - 10	**SECTION B - 7**	**SECTION C - 5**

PO – CO MAPPING

Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1.	3	3	3	3	2	1	3	2
CO2.	3	3	3	3	2	1	3	2
CO3	3	3	3	3	3	3	3	2
CO4	3	3	3	3	2	1	3	2
CO5	3	3	3	3	2	1	3	3
Average Contribution	3	3	3	3	2.2	1.4	3	2.2

PO – CO QUESTION PAPER MAPPING

CO No:	COURSE OUTCOMES	POs ADDRESSED	COGNITIVE LEVEL (K1 to K6)
CO1	**Analyze** ethical hacking principles, vulnerability assessment methodologies, penetration testing processes, and social engineering attack techniques to identify security weaknesses and recommend appropriate countermeasures.	1,2,3,4,7	1 to 5
CO2	**Apply** physical penetration testing techniques, insider threat assessment methods, and Metasploit framework tools to evaluate organizational security and demonstrate controlled exploitation scenarios.	1,2,3,4,7	1 to 5
CO3	**Analyze** the planning, execution, and reporting phases of penetration testing and **evaluate** Linux and Windows exploit development techniques for identifying system vulnerabilities and security risks.	1,2,3,4,5,6,7	1 to 5
CO4	**Analyze** web application security vulnerabilities, including OWASP Top Ten risks, SQL Injection, Cross-Site Scripting, and vulnerability analysis techniques to assess application security posture and recommend mitigation strategies.	1,2,3,4,7	1 to 5
CO5	**Evaluate** client-side browser exploits and malware analysis techniques to investigate emerging threats, assess attack vectors, and propose effective defensive mechanisms for secure computing environments.	1,2,3,4,7,8	1 to 5
K1= Remember, K2= Understand, K3= Apply, K4=Analyse, K5= Evaluate, K6= Create			

Continuous Internal Assessment - Components			
Assessment Type	**Nature**	**Maximum Marks**	**Weightage**
Mid-Semester Examination	Pen and Paper test for 90 minutes	50	10
Model Examination	Written examination for 180 minutes	100	10
Assignment	Any topic related to the course	10	5
Workbook/recordbook/ notebook	Handwritten Class Notes	5	5
Class activity 1	Quiz with MCQ,	10	5
Class activity 2	Seminar / Presentation	10	5
Class activity 3	Surprise Class Tests	10	5
Attendance	Marks will be given based on the attendance percentage of respective course		5
Total			50

PROGRAMME: MCA	**BATCH:2026-28**
PART: A	**COURSE COMPONENT: CORE - IV**
COURSE NAME: Practical - I : ADBMS Lab	**COURSE CODE: 26PMCA304P**
SEMESTER: I	**MARKS:100**
CREDITS: 3	**TOTAL HOURS: 45**
PRACTICAL	

COURSE OBJECTIVE:

To enable students to apply and develop database programming skills using SQL, PL/SQL, and MongoDB for designing, implementing, automating, and managing relational and NoSQL database solutions through queries, stored programs, cursors, triggers, exception handling, and data manipulation operations..

COURSE OUTCOMES:

1. Construct relational database tables and perform SQL operations including joins, data manipulation, and query processing to manage and retrieve data effectively.
2. Develop PL/SQL stored procedures and functions using parameter passing mechanisms to automate database operations and business logic implementation.
3. Implement PL/SQL exception handling and cursor mechanisms to process query result sets and develop robust database applications.
4. Develop database triggers using PL/SQL to enforce business rules, validate data, and automate database events.
5. Develop and evaluate MongoDB database solutions by creating databases, managing collections and documents, performing CRUD operations, and executing queries for effective NoSQL data management.

EXERCISES:

1. Create tables of your choice and exhibit different types of Joins and DML Command usage.
2. Create a Stored Procedure using PL/SQL to increase the Basic pay of an employee with a specific amount.
3. Create a Stored Procedure that uses a single exception handler to handle the predefined exception NO_DATA_FOUND, which can occur in SELECT INTO statements.
4. Create a Stored Procedure that uses a single Exception Handler to handle the User Defined Exception.
5. Create a Stored Procedure that uses multiple Exception Handlers to handle predefined exception and user defined exception.
6. Create a PL/SQL program to exhibit Processing query result set with Explicit Cursors
7. Create an explicit cursor that accepts two parameters, and then uses it in an CURSOR FOR LOOP statement to display the wages paid to employees who earn more than a specified wage in a specified department
8. Create a Function procedure using PL/SQL to calculate and return the Net pay of an employee. The function must receive the necessary inputs as parameters.
9. Create a Trigger using PL/SQL to validate the Basic pay given in Insert Query. Allow insertion only if Basic Pay is greater than 5000.

10. Create a Trigger that uses Conditional Predicates to Detect Triggering Statement.
11. Create a Trigger using PL/SQL to exhibit INSTEAD OF Trigger usage.
12. MongoDB Installation and Database Creation
13. Working with Documents and Collections
14. CRUD Operations in MongoDB
15. Querying, Projection, and Sorting in MongoDB

PO- CO MAPPING

COs \ POs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**
CO1.	3	2	3	2	1	–	1	1
CO2.	3	2	3	3	1	–	1	1
CO3.	3	3	3	3	1	–	2	1
CO4.	3	3	3	3	1	–	3	1
CO5.	2	3	3	3	1	–	1	3
Average	**2.8**	**2.6**	**3.0**	**2.8**	**1.0**	**–**	**1.6**	**1.4**

Continuous Internal Assessment - Components			
Assessment Type	**Nature**	**Maximum Marks**	**Weightage**
Mid-Semester Examination	Practical examination with programming/problem-solving tasks for 90 minutes	50	10
Model Examination	Practical examination with programming/problem-solving tasks for 180 minutes	100	10
Assignment	Assignment based on experimental concepts	10	5
Workbook/recordbook/ notebook	Handwritten record of experiments, observations, results, and conclusions	10	5
Class activity 1	Debugging Test	5	5
Class activity 2	Debugging Test	5	5
Class activity 3	Quiz with MCQ	5	5
Attendance	Marks will be given based on the attendance percentage of respective course		5
Total			50

PROGRAMME: MCA	**BATCH:2026-28**
PART: A	**COURSE COMPONENT: CORE - V**
COURSE NAME: Practical - II: Advanced Java Technologies Lab	**COURSE CODE: 26PMCA305P**
SEMESTER: I	**MARKS:100**
CREDITS: 3	**TOTAL HOURS: 45**
PRACTICAL	

COURSE OBJECTIVE:

To enable students to design, develop, and deploy object-oriented, database-driven, and web-based applications using Java, JDBC, Servlets, JSP, multithreading, and Spring Boot frameworks for solving real-world software development problems.

COURSE OUTCOMES:

1. Develop Java programs using object-oriented programming concepts such as classes, constructors, method overloading, inheritance, packages, and string manipulation to solve computational problems.
2. Implement exception handling and multithreading mechanisms in Java applications to improve program reliability and concurrent execution.
3. Develop database-driven Java applications using JDBC for data retrieval, insertion, modification, and update operations.
4. Develop dynamic web applications using Servlets and JSP technologies to manage user interactions, session tracking, cookies, authentication, and database connectivity.
5. Develop and deploy enterprise web applications using Spring Boot, RESTful APIs, MVC architecture, and Thymeleaf templates for modern application development.

EXERCISES:

1. Program to check if a given number is palindrome or not.
2. Program to calculate the area of shapes using method overloading.
3. Program to demonstrate constructor overloading with default arguments
4. Program to demonstrate String class and its methods.
5. Program to demonstrate Multiple Inheritance
6. Program to handle the Exception using try and multiple catch block.
7. Program to Create a Package and access the member variables of classes contained in the package.
8. Program to create a Thread that implements Runnable Interface.
9. Program to execute select query using JDBC
10. Java program to insert, modify, update value in SQL table using JDBC.
11. Java Program to create HTTP servlet and display a Welcome message.
12. Java program to retrieve the details from login form and display using Servlet.
13. Java Program to handle session tracking using servlet
14. Write a JSP application that checks the login credentials and display appropriate message back to the Client?
15. JSP program to perform database operations.

16. JSP program to create cookies.
17. Create a basic Spring Boot application that displays “Welcome to Spring Boot” on a browser.
18. Create a RESTful API with endpoints.
19. Use Spring Boot MVC to create a web form for student details and display submitted data on another page using Thymeleaf.

PO – CO MAPPING

Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1.	3	3	3	3	2	1	2	2
CO2.	3	3	3	3	2	2	2	2
CO3	3	3	3	3	2	2	2	2
CO4	3	3	3	3	2	2	3	2
CO5	3	3	3	3	3	2	2	3
Average Contribution	3	3	3	3	2.2	1.8	2.2	2.2

Continuous Internal Assessment - Components			
Assessment Type	**Nature**	**Maximum Marks**	**Weightage**
Mid-Semester Examination	Practical examination with programming/problem-solving tasks for 90 minutes	50	10
Model Examination	Practical examination with programming/problem-solving tasks for 180 minutes	100	10
Assignment	Assignment based on experimental concepts	10	5
Workbook/recordbook/ notebook	Handwritten record of experiments, observations, results, and conclusions	10	5
Class activity 1	Debugging Test	5	5
Class activity 2	Debugging Test	5	5
Class activity 3	Quiz with MCQ	5	5
Attendance	Marks will be given based on the attendance percentage of respective course		5
Total			50

PROGRAMME: MCA	**BATCH:2026-28**
PART: B	**COURSE COMPONENT: Ability Enhancement Skills - I**
COURSE NAME: Communication and Presentation Skill	**COURSE CODE: 26PSKL401**
SEMESTER: I	**MARKS:100**
CREDITS: 2	**TOTAL HOURS: 30**
SKILL	

COURSE OBJECTIVE:

To equip students with the ability to communicate, collaborate, and present ideas effectively using verbal, non-verbal, written, digital, and presentation skills while adhering to professional and ethical communication practices.

COURSE OUTCOMES:

1. Explain the fundamentals of effective communication, including listening, speaking, reading, and writing (LSRW) skills, and evaluate the impact of technology on communication processes.
2. Apply appropriate verbal, non-verbal, written, visual, digital, and group communication techniques in professional and organizational contexts.
3. Analyze various communication barriers and determine suitable strategies to overcome them for effective interpersonal and organizational communication.
4. Demonstrate ethical communication practices, active listening, emotional intelligence, professionalism, cultural sensitivity, and constructive feedback in diverse professional environments.
5. Develop and deliver effective presentations using appropriate tools, audio-visual aids, and audience engagement techniques while managing questions, interruptions, and feedback confidently.

UNIT I

Essentials of Effective Communication : Communication Skills-LSRW- Characteristic features of LSRW-Consequences of Ineffective Communication-Impact of technology on Communication

UNIT II

Types of Communication : Verbal Communication – Non-verbal Communication- Visual Communication - Written Communication-Group Communication-Digital Communication-Formal and Informal Communication-Vertical-Horizontal-Diagonal Grapevine

UNIT III

Barriers in Communication : Physical Barriers - Language Barriers - Social and Cultural Barriers - Psychological Barriers - Semantic Barriers - Interpersonal Barriers - Technological Barriers- Means to overcome the various barriers to Communication

UNIT IV

Etiquettes and Ethical Practices in Communication : Listening - Clarity and Conciseness - Professional Tone - Timeliness - Constructive Feedback-Transparency-Professionalism-

Accountability-Confidentiality-Cultural Sensitivity- Emotional Intelligence-Empathy-Social Intelligence-Social Etiquettes- Appreciation and Gratitude

UNIT V

Presentation Skills : Types of Presentation- Preparing a presentation-Do's and Don'ts while giving a presentation- Managing tools for presentation-Using Prompts-Making effective uses of Audio/Visual aid during presentation-Dealing with Questions, Interruptions and Pauses- Practical: Participating in Mock presentations

RECOMMENDED TEXTBOOKS:

1. Monippally, Matthukutty, M. Business Communication Strategies. New Delhi: Tata McGraw-Hill Publishing Company Ltd., 2001.
2. Peter, Francis. (2012) Soft Skills and Professional Communication. New Delhi: Tata McGraw Hill.
3. Raman, Meenakshi & Prakash Singh (2012) Business Communication Oxford University Press

REFERENCE BOOKS:

1. Gallo, Maria. D (2018) Stop Lecturing Start Communicating: The Public Speaking Survival Guide for Business Kindle Edition
2. Hasson, Gill. (2012) Brilliant Communication Skills. Great Britain: Pearson Education.
3. Patil, Shailesh (2020) Handbook on Public Speaking, Presentation & Communication Skills: Principles & Practices to create high impact presentations & meaningful conversations, Chennai, Notion Press Media Pvt Ltd.

E-LEARNING RESOURCES:

1. https://uwaterloo.ca/centre-for-teaching-excellence/catalogs/tip-sheets/effective- communication-barriers-and-strategies
2. https://www.coursera.org/articles/presentation-skills
3. https://positivepsychology.com/how-to-improve-communication-skills/

The end semester exam will have viva voce for 50 marks and internal 50 marks

SEMESTER - II

PROGRAMME: MCA	**BATCH:2026-28**
PART: A	**COURSE COMPONENT: CORE -VI**
COURSE NAME: Python Programming	**COURSE CODE: 26PMCA306**
SEMESTER: II	**MARKS:100**
CREDITS: 4	**TOTAL HOURS: 75**
THEORY	

COURSE OBJECTIVE:

To develop the ability to design, implement, analyze, and evaluate data-driven solutions using Python programming, data analytics, machine learning, and deep learning techniques, while employing modern computational tools, agile project management practices, ethical principles, and secure AI methodologies to solve real-world problems.

COURSE OUTCOMES:

1. Apply Python programming constructs, data structures, functions, modules, and control statements to develop solutions for basic data analysis problems.
2. Utilize Python data science libraries such as NumPy, Pandas, Matplotlib, Seaborn, and SciPy to perform data manipulation, transformation, and visualization tasks.
3. Analyze datasets using data cleaning, preprocessing, exploratory data analysis, statistical techniques, and visualization methods to identify patterns and derive meaningful insights.
4. Implement and evaluate machine learning models including K-Nearest Neighbors, Naïve Bayes, Linear Regression, Multiple Linear Regression, and Logistic Regression for predictive analytics applications.
5. Design, develop, and evaluate intelligent machine learning solutions using neural networks, clustering and deep learning techniques, while applying agile project management practices, ethical principles, responsible AI guidelines, and secure machine learning methodologies to address real-world data-driven problems

UNIT I **15 HOURS**

Introduction to Data Analysis: Role of Python in Modern Data Science-Real-World Applications of Data Analysis-Overview of Tools and Libraries- Python Syntax and Variables-Data Types: Strings, Numbers, Lists, Dictionaries Control Flow: Loops and Conditionals-Writing Reusable Code with Functions-Importing and Using Modules-Python Basics for Data Analysis- Python Syntax and Variables - Data Types: Strings, Numbers, Lists, Dictionaries- Control Flow: Loops and Conditionals - Writing Reusable Code with Functions- Importing and Using Modules.

UNIT II **15 HOURS**

Essential Python Libraries-NumPy: Numerical Computing Basics-Pandas: Introduction to Data Frames- Matplotlib and Seaborn: Visualization Primer-SciPy: Scientific Computing Tools- Working with NumPy Arrays-Creating and Manipulating Arrays-Array Operations and Broadcasting-Indexing, Slicing, and Reshaping-Statistical Functions (Mean, Median, Variance)- Data Manipulation with Pandas-Series vs. Data Frames: Key Differences - Reading/Writing Data (CSV, Excel, SQL) -

Handling Missing Data (dropna , fillna)- Filtering, Sorting, and Grouping Data - Merging and Joining Datasets.

UNIT III **15 HOURS**

Data Cleaning and Pre-processing - Identifying and Removing Duplicates - Outlier Detection and Treatment- Data Normalization and Standardization- Time Series Data in Pandas-Basic Visualization with Matplotlib - Line, Bar, and Scatter Plots - Histograms and Boxplots -Customizing Titles, Legends, and Annotations- Creating Subplots and Multi-Panel Figures- Exploratory Data Analysis (EDA) - Descriptive Statistics (Mean, Median, Skewness)- Correlation Analysis (Pearson, Spearman)- Identifying Trends and Patterns-Case Study: EDA on a Real Dataset

UNIT IV **15 HOURS**

The Data Science Process-Data Acquisition-Data Preparation-Data Exploration -Data Modelling-Data Presentation-Machine Learning - K-Nearest Neighbors-Introduction to K Nearest Neighbors-How to create and test the K Nearest Neighbor classifier-Another Application-Naive Bayes-Applications of Naive Bayes-How to Build a Basic Model Using Naive Bayes in Python-Regression-Introduction to Labels and Features-Features-Simple and Multiple Linear Regression-Logistic Regression-Generalized Linear Models

UNIT V **15 HOURS**

Neural Networks – Perceptrons- Backpropagation – Clustering - Introduction to Clustering - Example of Clustering -Running K-means with Scikit-Learn - Implementation of the Model - Bottom-up Hierarchical Clustering - K-means Clustering- Network Analysis-Classification -TensorFlow on Windows- Create a Neural Network Model - Deep Learning Case Studies-Bank Churn Modelling-Sentiment Analysis- Case Study on : **Agile-Based Machine Learning Project Management and Team Collaboration, Responsible AI, Ethics, and Secure Machine Learning**

RECOMMENDED TEXTBOOKS:

1. Aniket Jain, Python for Data Analysis: A Beginner's Guide to Data Science, 2025
2. Daniel Nedal & Peters Morgan, INTRODUCTION TO MACHINE LEARNING WITH PYTHON, Davies Company, 2018

REFERENCE BOOKS:

1. Noah Gift, Alfredo Deza and Alexey Grigorev, *Practical MLOps: Operationalizing Machine Learning Models*, O'Reilly Media, 2021. (Openly accessible online through O'Reilly Learning Platform).

GUIDELINES TO THE QUESTION PAPER SETTERS
QUESTION PAPER PATTERN

SECTION	QUESTION COMPONENT	NUMBERS	MARKS	TOTAL
A	*Answer all the questions*	*1-10*	*2*	*20*
B	*Answer any 5 out of 7 questions*	*11-17*	*7*	*35*
C	*Answer any 3 out of 5 questions(each in 1200 words)*	*18-22*	*15*	*45*
				100

BREAK UP OF QUESTIONS

UNITS	SECTION A	SECTION B	SECTION C
I	2	2	1
II	2	1	1
III	2	2	1
IV	2	1	1
V	2	1	1
TOTAL	10	7	5
	SECTION A - 10	**SECTION B - 7**	**SECTION C - 5**

PO – CO MAPPING

Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1.	3	3	3	3	1	1	1	2
CO2.	3	3	3	3	1	1	1	2
CO3	3	3	3	3	1	1	2	3
CO4	3	3	3	3	2	1	2	3
CO5	3	3	3	3	3	3	3	3
Average Contribution	3	3	3	3	1.6	1.4	1.8	2.6

PO – CO QUESTION PAPER MAPPING

CO No:	COURSE OUTCOMES	POs ADDRESSED	COGNITIVE LEVEL (K1 to K6)
CO1	**Apply** Python programming constructs, data structures, functions, modules, and control statements to develop solutions for basic data analysis problems.	1,2,3,4	1 to 5
CO2	**Utilize** Python data science libraries such as NumPy, Pandas, Matplotlib, Seaborn, and SciPy to perform data manipulation, transformation, and visualization tasks.	1,2,3,4	1 to 5

CO3	**Analyze** datasets using data cleaning, preprocessing, exploratory data analysis, statistical techniques, and visualization methods to identify patterns and derive meaningful insights.	1,2,3,4,8	1 to 5
CO4	**Implement and evaluate** machine learning models including K-Nearest Neighbors, Naïve Bayes, Linear Regression, Multiple Linear Regression, and Logistic Regression for predictive analytics applications.	1,2,3,4,8	1 to 5
CO5	Design, develop, and evaluate intelligent machine learning solutions using neural networks, clustering and deep learning techniques, while applying agile project management practices, ethical principles, responsible AI guidelines, and secure machine learning methodologies to address real-world data-driven problems	1,2,3,4,5,6,7,8	1 to 5
K1= Remember, K2= Understand, K3= Apply, K4=Analyse, K5= Evaluate, K6= Create			

Continuous Internal Assessment - Components			
Assessment Type	**Nature**	**Maximum Marks**	**Weightage**
Mid-Semester Examination	Pen and Paper test for 90 minutes	50	10
Model Examination	Written examination for 180 minutes	100	10
Assignment	Any topic related to the course	10	5
Workbook/recordbook/ notebook	Handwritten Class Notes	5	5
Class activity 1	Quiz with MCQ,	10	5
Class activity 2	Seminar / Presentation	10	5
Class activity 3	Surprise Class Tests	10	5
Attendance	Marks will be given based on the attendance percentage of respective course		5
Total			50

PROGRAMME: MCA	**BATCH:2026-28**
PART: A	**COURSE COMPONENT: CORE - VII**
COURSE NAME: Data Structures and Algorithms	**COURSE CODE: 26PMCA307**
SEMESTER: II	**MARKS:100**
CREDITS: 4	**TOTAL HOURS: 75**
THEORY	

COURSE OBJECTIVE:

To enable students to understand, apply, analyze, and evaluate fundamental data structures and algorithm design techniques by integrating arrays, linked lists, stacks, queues, trees, graphs, divide-and-conquer, greedy, dynamic programming, and backtracking approaches, along with complexity analysis, NP-completeness concepts, and Agile algorithm development with ethical software engineering practices for designing efficient and responsible computational solutions to real-world problems

COURSE OUTCOMES:

1. Explain fundamental concepts of data structures, algorithms, pseudocode representation, asymptotic notations, complexity analysis, and searching techniques for evaluating algorithm efficiency.
2. Implement linear data structures such as stacks, queues, and linked lists and apply them to solve computational problems.
3. Demonstrate and apply tree and graph data structures, including traversal techniques and classical graph algorithms, to solve real-world computational problems.
4. Analyze and design efficient solutions using divide-and-conquer and greedy strategies for sorting, optimization, and shortest path problems.
5. Evaluate optimal solutions for complex computational problems using dynamic programming, backtracking techniques, and NP-completeness theory, including problems such as 0/1 Knapsack, Traveling Salesman Problem, N-Queens, and Graph Coloring, while integrating Agile algorithm development practices and ethical software engineering principles for efficient and responsible system design.

UNIT I **15 HOURS**

Data Structures: Basic Terminologies – Algorithms: Definition, Pseudocode Representation – Time complexity and space complexity - efficiency of algorithms - O-notation - Omega notation and Theta Notation Arrays: One dimensional & multidimensional arrays – Searching

UNIT II **15 HOURS**

Stacks: Representation - Operations – Applications. Queues: Representation – Operations – Applications. Linked List: Single Linked List, Double Linked List, Circular Linked List, Applications of Linked List.

UNIT III **15 HOURS**

Trees: Basic Terminologies - Binary trees: Representation, Operations, Traversals, Types - Applications of Trees. Graphs: Basic Terminologies, Representation, Operations, Traversals – Applications: Shortest path problem, Topological sorting, Minimum Spanning trees.

UNIT IV **15 HOURS**

Divide and Conquer: General method – binary search - merge sort - quick sort--Recursive algorithms, analysis of non-recursive and recursive algorithms, solving recurrence equations. Greedy Method: General method – knapsack problem – minimum spanning tree algorithms – single source shortest path algorithm – scheduling - Huffman codes

UNIT V **15 HOURS**

Dynamic Programming: 0/1 Knapsack and Traveling salesman problem – Backtracking: n- Queen problem - Graph Colouring - Introduction to NP-Hard and NP-Completeness –Case Study on : **Agile Algorithm Development and Ethical Software Engineering Lifecycle**

RECOMMENDED TEXTBOOKS:

1. Ellis Horowitz and Sartaj Sahni,Susan Anderson-Freed, Fundamentals of Data Structures in C (Second Edition), 2008.
2. Ellis Horowitz, Sartaj Sahni, Sanguthevar, Rajasekaran, Fundamentals of ComputerAlgorithms, Galgotia Publications Pvt. Ltd., 2008.
3. Debasis Samanta, Classic Data Structures, Second Edition, Prentice-Hall of India, Pvt. Ltd.,India, 2009.

REFERENCE BOOKS:

1. Thomas H. Cormen, Charles E. Leiserson, Ronald Rivest, Clifford Stein, Introduction to Algorithms , MIT Press, 2009
2. Dinesh P Mehta & Sartaj Sahni, Handbook of Data Structures and Applications, Chapman and Hall. 2005.

E-LEARNING RESOURCES:

1. https://www.geeksforgeeks.org/data-structures/
2. https://ocw.mit.edu/courses/6-006-introduction-to-algorithms-fall-2011/

GUIDELINES TO THE QUESTION PAPER SETTERS
QUESTION PAPER PATTERN

SECTION	QUESTION COMPONENT	NUMBERS	MARKS	TOTAL
A	*Answer all the questions*	*1-10*	*2*	*20*
B	*Answer any 5 out of 7 questions*	*11-17*	*7*	*35*
C	*Answer any 3 out of 5 questions(each in 1200 words)*	*18-22*	*15*	*45*
				100

BREAK UP OF QUESTIONS

UNITS	SECTION A	SECTION B	SECTION C
I	2	2	1
II	2	1	1
III	2	2	1
IV	2	1	1
V	2	1	1
TOTAL	10	7	5
	SECTION A - 10	**SECTION B - 7**	**SECTION C - 5**

PO – CO MAPPING

Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1.	3	3	2	3	1	1	2	3
CO2.	3	3	3	3	2	1	2	3
CO3	3	3	3	3	2	1	2	3
CO4	3	3	3	3	2	1	2	3
CO5	3	3	3	3	3	3	3	3
Average Contribution	3	3	2.8	3	2	1.4	2.2	3

PO – CO QUESTION PAPER MAPPING

CO No:	COURSE OUTCOMES	POs ADDRESSED	COGNITIVE LEVEL (K1 to K6)
CO1	**Explain** fundamental concepts of data structures, algorithms, pseudocode representation, asymptotic notations, complexity analysis, and searching techniques for evaluating algorithm efficiency.	1,2,4,8	1 to 5
CO2	**Implement** linear data structures such as stacks, queues, and linked lists and apply them to solve computational problems.	1,2,3,4,8	1 to 5
CO3	**Demonstrate and apply** tree and graph data structures, including traversal techniques and classical graph algorithms, to solve real-world computational problems.	1,2,3,4,8	1 to 5
CO4	**Analyze and design** efficient solutions using divide-and-conquer and greedy strategies for sorting, optimization, and shortest path problems.	1,2,3,4,8	1 to 5
CO5	**Evaluate** optimal solutions for complex computational problems using dynamic programming, backtracking techniques, and NP-completeness theory, including problems such as 0/1 Knapsack, Traveling Salesman Problem, N-Queens, and Graph Coloring, while integrating Agile algorithm development practices and ethical software engineering principles for efficient and responsible system design.	1,2,3,4,5,6,7,8	1 to 5
K1= Remember, K2= Understand, K3= Apply, K4=Analyse, K5= Evaluate, K6= Create			

Continuous Internal Assessment - Components			
Assessment Type	**Nature**	**Maximum Marks**	**Weightage**
Mid-Semester Examination	Pen and Paper test for 90 minutes	50	10
Model Examination	Written examination for 180 minutes	100	10
Assignment	Any topic related to the course	10	5
Workbook/recordbook/ notebook	Handwritten Class Notes	5	5
Class activity 1	Quiz with MCQ,	10	5
Class activity 2	Seminar / Presentation	10	5
Class activity 3	Surprise Class Tests	10	5
Attendance	Marks will be given based on the attendance percentage of respective course		5
Total			50

PROGRAMME: MCA	**BATCH:2026-28**
PART: A	**COURSE COMPONENT: CORE - VIII**
COURSE NAME: Block Chain Technologies	**COURSE CODE: 26PMCA308**
SEMESTER: II	**MARKS:100**
CREDITS: 3	**TOTAL HOURS: 60**
THEORY	

COURSE OBJECTIVE:

To enable students to analyze, and evaluate blockchain-based systems by applying blockchain architectures, platforms, design patterns, security mechanisms, performance models, and agile development practices to develop effective decentralized solutions for real-world applications.

COURSE OUTCOMES:

1. Explain the architecture, functionality, non-functional properties, and key features of major blockchain platforms including Bitcoin, Ethereum, and Hyperledger Fabric.
2. Analyze the characteristics of different blockchain models and evaluate their suitability for real-world applications such as supply chain management and international money transfer systems.
3. Apply blockchain architectural principles and design methodologies to develop blockchain-based solutions for software systems.
4. Design blockchain-enabled applications by incorporating appropriate interaction, data management, security, and smart contract structural patterns.
5. .Evaluate the performance, dependability, security, and transaction management aspects of blockchain systems, and apply agile project development practices for the design and management of decentralized applications (DApps) to support informed architectural decision-making.

UNIT I **12 HOURS**

Introduction: What is Blockchain – Blockchain-Based Applications – Blockchain Functionality – Blockchain Non-functional properties – Blockchain Architecture Design. Existing Blockchain Platforms: Bitcoin – Ethereum – Hyperledger Fabric – Other Representative Bitcoin Platforms

UNIT II **12 HOURS**

Varieties of Blockchains: Fundamental properties of Blockchain – Decentralization – Ledger Structure – Consensus Protocol – Block Configuration – Auxiliary Blockchains – Anonymity – Incentives. Example Use Cases**:** Agricultural Supply Chains and International Money Transfers (A brief discussion; Context; Key Non-functional Requirements; Conventional Technology being used, Blockchain Solution)

UNIT III **12 HOURS**

Blockchain in Software Architecture: Blockchain as an Architectural Element - Blockchain as Storage Element - Blockchain as Computational Element - Blockchain as Communication Mechanism - Blockchain as an Asset Management and Control Mechanism - Integrating Blockchain into a System as

a Component, Design Process for Applications on Blockchain: Evaluation of Suitability - Design Process for Blockchain-Based Systems

UNIT IV **12 HOURS**

Blockchain Patterns: Patterns on Interacting with the External World - Data Management Patterns - Security Patterns - Contract Structural Patterns

UNIT V **12 HOURS**

Performance: Performance Characteristics of Blockchain - Architectural Performance Modelling - Predicting Latency for Blockchain-Based Systems - Architectural Decision-Making, Dependability and Security: Confidentiality – Integrity – Safety – Maintainability - Availability and Reliability - Aborting and Retrying Blockchain Transactions in Ethereum and Bitcoin – Case Study on : Agile Blockchain Project Development & DApp Management

RECOMMENDED TEXTBOOKS:

1. Xiwei Xu , Ingo Weber, Mark Staples, 2019, Architecture for Blockchain Applications,Springer.
2. Blockchain IBM limited edition by Manav Gupta

REFERENCE BOOKS:

1. Antonopoulos, A. M., & Wood, G. (2018). *Mastering Ethereum: Building Smart Contracts and DApps*. O'Reilly Media.
2. "Serious Cryptography": A practical Introduction to modern Encryption by Jean-Phillippe, Aumasson.0"classification Techniques For Intrusion and Detection" by p.Amudha, S.Sivakumar
3. "Access control, Authentication,Public Key Infrastucture" 2nd Edition By Mike Chapple.

E-LEARNING RESOURCES:

1. https://onlinecourses.nptel.ac.in/noc24_cs15

GUIDELINES TO THE QUESTION PAPER SETTERS
QUESTION PAPER PATTERN

SECTION	QUESTION COMPONENT	NUMBERS	MARKS	TOTAL
A	*Answer all the questions*	*1-10*	*2*	*20*
B	*Answer any 5 out of 7 questions*	*11-17*	*7*	*35*
C	*Answer any 3 out of 5 questions(each in 1200 words)*	*18-22*	*15*	*45*
				100

BREAK UP OF QUESTIONS

UNITS	SECTION A	SECTION B	SECTION C
I	2	2	1
II	2	1	1
III	2	2	1
IV	2	1	1
V	2	1	1
TOTAL	10	7	5
	SECTION A - 10	**SECTION B - 7**	**SECTION C - 5**

PO – CO MAPPING

Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1.	3	3	2	3	1	1	2	3
CO2.	3	3	3	3	1	2	3	3
CO3	3	3	3	3	2	2	2	3
CO4	3	3	3	3	2	2	3	3
CO5	3	3	3	3	3	3	3	3
Average Contribution	3	3	2.8	3	1.8	2	2.6	3

PO – CO QUESTION PAPER MAPPING

CO No:	COURSE OUTCOMES	POs ADDRESSED	COGNITIVE LEVEL (K1 to K6)
CO1	**Explain** the architecture, functionality, non-functional properties, and key features of major blockchain platforms including Bitcoin, Ethereum, and Hyperledger Fabric.	1,2,4,8	1 to 5
CO2	**Analyze** the characteristics of different blockchain models and evaluate their suitability for real-world applications such as supply chain management and international money transfer systems.	1,2,3,4,7,8	1 to 5
CO3	**Apply** blockchain architectural principles and design methodologies to develop blockchain-based solutions for software systems.	1,2,3,4,8	1 to 5
CO4	**Design** blockchain-enabled applications by incorporating appropriate interaction, data management, security, and smart contract structural patterns.	1,2,3,4,7,8	1 to 5
CO5	.**Evaluate** the performance, dependability, security, and transaction management aspects of blockchain systems, and **apply** agile project development practices for the design and management of decentralized applications (DApps) to support informed architectural decision-making.	1,2,3,4,5,6,7,8	1 to 5
K1= Remember, K2= Understand, K3= Apply, K4=Analyse, K5= Evaluate, K6= Create			

Continuous Internal Assessment - Components			
Assessment Type	**Nature**	**Maximum Marks**	**Weightage**
Mid-Semester Examination	Pen and Paper test for 90 minutes	50	10
Model Examination	Written examination for 180 minutes	100	10
Assignment	Any topic related to the course	10	5
Workbook/recordbook/ notebook	Handwritten Class Notes	5	5
Class activity 1	Quiz with MCQ,	10	5
Class activity 2	Seminar / Presentation	10	5
Class activity 3	Surprise Class Tests	10	5
Attendance	Marks will be given based on the attendance percentage of respective course		5
Total			50

PROGRAMME: MCA	**BATCH:2026-28**
PART: A	**COURSE COMPONENT: CORE - IX**
COURSE NAME: Artificial Intelligence	**COURSE CODE: 26PMCA309**
SEMESTER: II	**MARKS:100**
CREDITS: 3	**TOTAL HOURS: 60**
THEORY	

COURSE OBJECTIVE:

To enable students to understand, apply, analyze, evaluate, artificial intelligence systems by integrating intelligent agents, state-space search strategies, knowledge representation techniques, logical inference mechanisms, non-monotonic reasoning, and expert systems, along with Agile AI project management and deployment lifecycle practices for developing intelligent, efficient, and real-world problem-solving applications.

COURSE OUTCOMES:

1. Explain the fundamental concepts of Artificial Intelligence, intelligent agents, problem-solving techniques, state space representation, and production systems for defining AI problem-solving frameworks.
2. Apply uninformed and informed search strategies such as BFS, DFS, A*, Hill Climbing, Best First Search, and AO* algorithm to solve AI-based problem spaces.
3. Analyze knowledge representation techniques including predicate logic, rules, frames, inheritance, and constraint propagation for effective knowledge modeling in intelligent systems.
4. Evaluate logical inference mechanisms including forward and backward reasoning, first-order logic, and rule-based deduction systems for knowledge-based problem solving.
5. Evaluate the effectiveness of non-monotonic reasoning, minimalist reasoning, and expert system methodologies—including knowledge representation, expert system shells, and knowledge acquisition techniques—along with Agile AI project management practices for developing intelligent, reliable, and real-world decision-support systems

UNIT I **12 HOURS**

Problems and Search: What is AI – AI problems – AI Techniques – Level of Model – **Intelligent Agents:** Agents and Environments – Concept of Rationality – The nature of Environments – Structure of agents – Problem solving Agents – Problem formulation Problems**, Problem Space and Search:** Defining the problem as a State Space Search – Production Systems

UNIT II **12 HOURS**

Searching –Searching for solutions - Uninformed search Strategies- Breadth First Search, Depth First Search Heuristic Search Techniques: Generate-and-Test - Hill Climbing – - Best First Search- the A* algorithm - Problem Reduction: AND –OR graphs – the AO* algorithm

UNIT III **12 HOURS**

Knowledge Representation: Knowledge Representation Issues Representation and Mappings – approaches to knowledge representation – Using Predicate Logic: Representing Instance and Isa Relationship – Computable functions and Predicates- representing knowledge using rules- Rule based deduction systems-Reason under uncertainty – Frames and Inheritance – Constraint Propagation

UNIT IV **12 HOURS**

Representing Knowledge using Rules: Procedural vs. Declarative Knowledge – Logic programming - Forward vs backward reasoning – First order logic –Inference in first order logic

UNIT V **12 HOURS**

Introduction to Non-monotonic Reasoning – Minimalist Reasoning - Expert Systems: Introduction, basic concepts, the structure of expert systems, the human element in expert systems how expert systems work, Representing and using Domain Knowledge – Expert System shells – Knowledge Acquisition – Case Study on : . **Agile AI Project Management and Deployment Lifecycle**

RECOMMENDED TEXTBOOKS:

1. Kevin Knight, Elaine Rich, B. Nair, "Artificial Intelligence (SIE)", Mc-Graw Hill Publication.

REFERENCE BOOKS:

1. Stuart Russell, Peter Norvig, Artificial Intelligence: A Modern Approach (3rd Edition), Pearson, 2010
2. Denis Rothman, "Artificial Intelligence By Example", Packt Publishing Ltd.

E-LEARNING RESOURCES:

1. https://www.tutorialspoint.com/artificial_intelligence/artificial_intelligence_overview.htm
2. https://www.javatpoint.com/artificial-intelligence-ai
3. https://www.tutorialspoint.com/tensorflow/tensorflow_understanding_artificial_intelligence.ht m
4. https://aima.cs.berkeley.edu/

GUIDELINES TO THE QUESTION PAPER SETTERS
QUESTION PAPER PATTERN

SECTION	QUESTION COMPONENT	NUMBERS	MARKS	TOTAL
A	*Answer all the questions*	*1-10*	*2*	*20*
B	*Answer any 5 out of 7 questions*	*11-17*	*7*	*35*
C	*Answer any 3 out of 5 questions(each in 1200 words)*	*18-22*	*15*	*45*
				100

BREAK UP OF QUESTIONS

UNITS	SECTION A	SECTION B	SECTION C
I	2	2	1
II	2	1	1
III	2	2	1
IV	2	1	1
V	2	1	1
TOTAL	10	7	5
	SECTION A - 10	**SECTION B - 7**	**SECTION C - 5**

PO – CO MAPPING

Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1.	3	3	3	3	3	2	3	3
CO2.	3	3	3	3	3	2	3	3
CO3	3	3	3	3	2	2	3	3
CO4	3	3	3	3	2	2	3	3
CO5	3	3	3	3	3	3	3	3
Average Contribution	3	3	3	3	2.6	2.2	3	3

PO – CO QUESTION PAPER MAPPING

CO No:	COURSE OUTCOMES	POs ADDRESSED	COGNITIVE LEVEL (K1 to K6)
CO1	**Explain** the fundamental concepts of Artificial Intelligence, intelligent agents, problem-solving techniques, state space representation, and production systems for defining AI problem-solving frameworks.	1,2,3,4,5,7,8	1 to 5
CO2	**Apply** uninformed and informed search strategies such as BFS, DFS, A*, Hill Climbing, Best First Search, and AO* algorithm to solve AI-based problem spaces.	1,2,3,4,5,7,8	1 to 5
CO3	**Analyze** knowledge representation techniques including predicate logic, rules, frames, inheritance, and constraint propagation for effective knowledge modeling in intelligent systems.	1,2,3,4,7,8	1 to 5
CO4	**Evaluate** logical inference mechanisms including forward and backward reasoning, first-order logic, and rule-based deduction systems for knowledge-based problem solving.	1,2,3,4,7,8	1 to 5
CO5	**Evaluate** the effectiveness of non-monotonic reasoning, minimalist reasoning, and expert system methodologies—including knowledge representation, expert system shells, and knowledge acquisition techniques—along with Agile AI project management practices for developing intelligent, reliable, and real-world decision-support systems	1,2,3,4,5,6,7,8	1 to 5
K1= Remember, K2= Understand, K3= Apply, K4=Analyse, K5= Evaluate, K6= Create			

Continuous Internal Assessment - Components			
Assessment Type	**Nature**	**Maximum Marks**	**Weightage**
Mid-Semester Examination	Pen and Paper test for 90 minutes	50	10
Model Examination	Written examination for 180 minutes	100	10
Assignment	Any topic related to the course	10	5
Workbook/recordbook/ notebook	Handwritten Class Notes	5	5
Class activity 1	Quiz with MCQ,	10	5
Class activity 2	Seminar / Presentation	10	5
Class activity 3	Surprise Class Tests	10	5
Attendance	Marks will be given based on the attendance percentage of respective course		5
Total			50

PROGRAMME: MCA	**BATCH:2026-28**
PART: A	**COURSE COMPONENT: EDE- I**
COURSE NAME: Statistics for Data Analytics	**COURSE CODE: 26PMCA3E2**
SEMESTER: II	**MARKS:100**
CREDITS: 3	**TOTAL HOURS: 60**
THEORY AND PROBLEMS	

COURSE OBJECTIVE:

To enable students to understand, apply, analyze, and evaluate statistical concepts including measures of central tendency and dispersion, probability theory, correlation and regression, probability distributions, and hypothesis testing for both large and small samples, and to integrate Agile Statistical Data Analysis Lifecycle (ASDAL) for structured, iterative, and collaborative statistical decision-making in real-world data-driven problem-solving.

COURSE OUTCOMES:

1. Explain and apply measures of central tendency, dispersion, and probability theory including conditional probability, independence, and Bayes' theorem to solve basic statistical and probabilistic problems
2. Analyze and evaluate relationships between variables using correlation and regression techniques, including correlation coefficient, rank correlation, and regression models to interpret statistical associations and predictions
3. Apply and solve problems using discrete and continuous probability distributions such as Binomial, Poisson, and Normal distributions for modeling real-world stochastic processes.
4. Examine and test statistical hypotheses using large sample techniques, sampling distributions, standard error concepts, and parametric tests including Z-test and F-test for population parameter inference.
5. Apply and evaluate small sample statistical inference techniques, including t-test and chi-square test for hypothesis testing of means, variances, and categorical association, and integrate Agile Statistical Data Analysis Lifecycle (ASDAL) practices to perform iterative, structured, and team-oriented statistical decision-making for real-world data analysis problems.

UNIT I **12 HOURS**

Measures of Averages: Arithmetic Mean and its Merits and Demerits - Median and its Merits and Demerits - Mode and its Merits and Demerits - **Measures of Dispersion:** Quartile Deviation and its Merits and Demerits – Mean Deviation and its Merits and Demerits – Standard Deviation and its Merits and Demerits - **Probability:** Sample Space- Events-Axiomatic approach to Probability - Conditional Probability – Independent events – Bayes formula

UNIT II **12 HOURS**

Correlation: Definitions of Correlation, Positive Correlation, Negative Correlation, No Correlation – Limitations of Correlation - Correlation Coefficient -Properties of correlation coefficient – Rank Correlation **– Regression:** Definition, Difference between correlation and regression – Regression lines - Regression Coefficient – Properties of Regression coefficient

UNIT III **12 HOURS**

Binomial Distribution: Definition, Binomial frequency distribution (Simple problems based on Binomial Distribution) – **Poisson Distribution:** Definition, Poisson Frequency Distribution (Simple problems based on Binomial Distribution) – **Normal Distribution:** Definition, Characteristics of Normal Distribution (Problems based on Area Property of Normal Distribution)

UNIT IV **12 HOURS**

Large Samples: Definitions of Population; Sample; Parameter; Statistic; Sampling Distribution; Standard Error; Null Hypothesis; Alternate Hypothesis, Types of Error – Hypothesis Testing Procedure (One Tail and Two Tail Test), significance level , - **Large Sample Tests:** Test for specified Mean – Test for Equality of two means- **Small Samples – F-test:** Introduction- Test of Equality of Two Population Variances.

UNIT V **12 HOURS**

Small Samples – t-test: Introduction, Uses, Properties – Test for a specified Mean – Test for significance for the difference between the two population means when population standard deviations are not known - **Small Samples – Chi Square-test:** Introduction, uses – Test of Independence of Attributes – Test for a specified Population Variance- Case Study on : **Agile Statistical Data Analysis Lifecycle (ASDAL)**

RECOMMENDED TEXTBOOKS:

1. P.R. VITTAL & MALINI, "Statistical and Numerical Methods", Margham Publications.

REFERENCE BOOKS:

1. Hadley Wickham, Mine Çetinkaya-Rundel, Garrett Grolemund, R for Data Science (2nd Edition), O'Reilly Media (Open Online Edition / Tidyverse Team), 2023
2. S.C. Gupta & V.K.Kapoor, "Fundamentals of Mathematical Statistics", Sultanchand & Sons
3. Dr. S.P. Gupta, "Statistical Methods", Sultanchand & Sons

E-LEARNING RESOURCES:

1. https://www.tutorialspoint.com/statistics/arithmetic_mean.htm
2. https://www.w3schools.com/statistics/index.php
3. https://www.listendata.com/p/statistics-tutorials.html
4. https://r4ds.hadley.nz/

GUIDELINES TO THE QUESTION PAPER SETTERS
QUESTION PAPER PATTERN

SECTION	QUESTION COMPONENT	NUMBERS	MARKS	TOTAL
A	*Answer all the questions*	*1-10*	*2*	*20*
B	*Answer any 5 out of 7 questions*	*11-17*	*7*	*35*
C	*Answer any 3 out of 5 questions(each in 1200 words)*	*18-22*	*15*	*45*
				100

BREAK UP OF QUESTIONS

UNITS	SECTION A		SECTION B		SECTION C	
	THEORY	PROBLEM	THEORY	PROBLEM	THEORY	PROBLEM
I	2		1	1	-	1
II	2			1	-	1
III	2			1	-	1
IV	2		2		-	1
V	2		1		-	1
TOTAL	**10**		**4**	**3**	**-**	**5**
	SECTION A - 10		**SECTION B - 7**		**SECTION C - 5**	

PO – CO MAPPING

Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1.	3	3	3	3	2	1	2	3
CO2.	3	3	3	3	2	1	2	3
CO3	3	3	3	3	2	1	2	3
CO4	3	3	3	3	2	1	2	3
CO5	3	3	3	3	3	3	3	3
Average Contribution	3	3	3	3	2.2	1.4	2.2	3

PO – CO QUESTION PAPER MAPPING

CO No:	COURSE OUTCOMES	POs ADDRESSED	COGNITIVE LEVEL (K1 to K6)
CO1	**Explain and apply** measures of central tendency, dispersion, and probability theory including conditional probability, independence, and Bayes' theorem to solve basic statistical and probabilistic problems	1,2,3,4,8	1 to 5

CO2	**Analyze and evaluate** relationships between variables using correlation and regression techniques, including correlation coefficient, rank correlation, and regression models to interpret statistical associations and predictions	1,2,3,4,8	1 to 5
CO3	**Apply and solve** problems using discrete and continuous probability distributions such as Binomial, Poisson, and Normal distributions for modeling real-world stochastic processes.	1,2,3,4,8	1 to 5
CO4	**Examine and test** statistical hypotheses using large sample techniques, sampling distributions, standard error concepts, and parametric tests including Z-test and F-test for population parameter inference.	1,2,3,4,8	1 to 5
CO5	**Apply and evaluate** small sample statistical inference techniques, including t-test and chi-square test for hypothesis testing of means, variances, and categorical association, and integrate Agile Statistical Data Analysis Lifecycle (ASDAL) practices to perform iterative, structured, and team-oriented statistical decision-making for real-world data analysis problems.	1,2,3,4,5,6,7,8	1 to 5
K1= Remember, K2= Understand, K3= Apply, K4=Analyse, K5= Evaluate, K6= Create			

Continuous Internal Assessment - Components			
Assessment Type	**Nature**	**Maximum Marks**	**Weightage**
Mid-Semester Examination	Pen and Paper test for 90 minutes	50	10
Model Examination	Written examination for 180 minutes	100	10
Assignment	Any topic related to the course	10	5
Workbook/recordbook/ notebook	Handwritten Class Notes	5	5
Class activity 1	Develop the Problem solving skill (Time Completion -3 Marks, Total number of question solved - 7Marks)	10	5
Class activity 2	Quiz with MCQ	10	5
Class activity 3	Surprise Class Test	10	5
Attendance	Marks will be given based on the attendance percentage of respective course		5
Total			50

PROGRAMME: MCA	**BATCH:2026-28**
PART: A	**COURSE COMPONENT: CORE - X**
COURSE NAME: Practical III : Python Programming Lab	**COURSE CODE: 26PMCA310P**
SEMESTER: II	**MARKS:100**
CREDITS: 3	**TOTAL HOURS: 45**
PRACTICAL	

COURSE OBJECTIVE:

To apply Python programming techniques, machine learning algorithms, data preprocessing methods, and search/sorting algorithms to develop, implement, and evaluate computational solutions for data-driven problem-solving and predictive analytics applications

COURSE OUTCOMES:

1. **Apply** Python programming constructs, data preprocessing techniques, and data handling methods to prepare datasets for machine learning applications.
2. **Implement** supervised and unsupervised machine learning algorithms, including Linear Regression, Multiple Linear Regression, Logistic Regression, K-Nearest Neighbors, Decision Trees, and Apriori, using Python libraries.
3. **Analyze** the performance and suitability of machine learning models by interpreting results, validating outputs, and comparing algorithmic approaches for different datasets.
4. **Develop** Python-based solutions employing classical search, sorting, and concept learning algorithms such as Linear Search, Binary Search, Merge Sort, Selection Sort, Insertion Sort, FIND-S, and Candidate-Elimination.
5. **Design and evaluate** intelligent predictive models using machine learning techniques and decision-tree-based classification algorithms to solve real-world data-driven problems.

EXERCISES

1. Write a Python program for Data Preprocessing
2. Write a Python program for Implementing A prior algorithm in Python
3. Write a Python program for Implementing Linear Regression using Python
4. Write a Python program for Implementing Multiple Linear Regression in Python
5. Write a Python program for Implementing Logistic Regression in Python
6. Write a Python program for Implementing Nearest Neighbors in Python
7. Write a Python program for Implementing Decision Tree in Python
8. Write a Python Program to perform Linear Search
9. Write a Python Program to perform Binary Search
10. Write a Python Program to perform Merge sort
11. Write a Python Program to selection sort
12. Write a Python Program to perform insertion sort
13. Write a Python program to Implement and demonstrate the FIND-S algorithm for finding the most specific hypothesis based on a given set of training data samples. Read the training data from a .CSV file.

14. For a given set of training data examples stored in a .CSV file, implement and demonstrate the Candidate-Elimination algorithm to output a description of the set of all hypotheses consistent with the training examples
15. Write a program to demonstrate the working of the decision tree based ID3 algorithm. Use an appropriate data set for building the decision tree and apply this knowledge to classify a new sample.

PO – CO MAPPING

Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1.	3	3	2	3	1	1	1	2
CO2.	3	3	3	3	1	1	1	2
CO3	3	3	3	3	1	1	2	3
CO4	3	3	3	3	1	1	1	2
CO5	3	3	3	3	2	1	2	3
Average Contribution	3	3	2.8	3	1.2	1	1.4	2.4

Continuous Internal Assessment - Components			
Assessment Type	**Nature**	**Maximum Marks**	**Weightage**
Mid-Semester Examination	Practical examination with programming/problem-solving tasks for 90 minutes	50	10
Model Examination	Practical examination with programming/problem-solving tasks for 180 minutes	100	10
Assignment	Assignment based on experimental concepts	10	5
Workbook/recordbook/ notebook	Handwritten record of experiments, observations, results, and conclusions	10	5
Class activity 1	Debugging Test	5	5
Class activity 2	Debugging Test	5	5
Class activity 3	Quiz with MCQ	5	5
Attendance	Marks will be given based on the attendance percentage of respective course		5
Total			50

PROGRAMME: MCA	**BATCH:2026-28**
PART: A	**COURSE COMPONENT: CORE - XI**
COURSE NAME: Practical IV : Data Structures and Algorithms Lab	**COURSE CODE: 26PMCA311P**
SEMESTER: II	**MARKS:100**
CREDITS: 3	**TOTAL HOURS: 45**
PRACTICAL	

COURSE OBJECTIVE:

To enable students to apply, analyze, and design efficient data structures and algorithmic solutions by implementing linear and non-linear data structures, searching and sorting techniques, graph and tree traversal methods, and classical algorithm design paradigms such as divide-and-conquer, greedy, dynamic programming, and backtracking to solve computational problems with optimal time and space efficiency.

COURSE OUTCOMES:

1. **Apply** linear, binary, and Fibonacci search algorithms and evaluate their performance in terms of key comparisons for efficient data retrieval.

2. **Implement** and analyze stack, queue, circular queue, and priority queue operations, along with singly, doubly, and circular linked list data structures for solving computational problems.

3. **Demonstrate** tree and graph traversal techniques (inorder, preorder, postorder, BFS, DFS) and apply shortest path algorithms such as Dijkstra's algorithm to solve network and optimization problems.

4. **Design and evaluate** algorithmic solutions using divide-and-conquer techniques for binary search and sorting problems, and greedy techniques for optimization problems such as knapsack.

5. **Develop and analyze** advanced algorithmic solutions using dynamic programming and backtracking techniques to solve complex problems such as Traveling Salesman Problem and 8-Queens problem.

EXERCISES :

1. Searching Algorithms (with the number of key comparisons) - Sequential, Binary and Fibonacci search algorithms
2. Evaluation of arithmetic expression.
3. Stack, Queue, Circular queue, priority queue
4. Singly Linked List, Doubly Linked List, Circular Linked List.
5. Tree Traversal techniques.
6. Graph Traversal techniques.
7. Dijkstra's Algorithm to obtain the shortest paths
8. Binary search using Divide-and-Conquer technique.
9. Sorting algorithms using Divide-and-Conquer technique.

10. Knapsack using Greedy technique.
11. Traveling Salesman algorithms using Dynamic Programming technique.
12. 8 Queens with the design of Backtracking.

PO – CO MAPPING

Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1.	3	3	3	3	2	1	2	3
CO2.	3	3	3	3	2	1	2	3
CO3	3	3	3	3	2	1	2	3
CO4	3	3	3	3	2	1	2	3
CO5	3	3	3	3	2	1	2	3
Average Contribution	3	3	3	3	2	1	2	3

Continuous Internal Assessment - Components			
Assessment Type	**Nature**	**Maximum Marks**	**Weightage**
Mid-Semester Examination	Practical examination with programming/problem-solving tasks for 90 minutes	50	10
Model Examination	Practical examination with programming/problem-solving tasks for 180 minutes	100	10
Assignment	Assignment based on experimental concepts	10	5
Workbook/recordbook/ notebook	Handwritten record of experiments, observations, results, and conclusions	10	5
Class activity 1	Debugging Test	5	5
Class activity 2	Debugging Test	5	5
Class activity 3	Quiz with MCQ	5	5
Attendance	Marks will be given based on the attendance percentage of respective course		5
Total			50

PROGRAMME: MCA	**BATCH:2026-28**
PART: B	**COURSE COMPONENT: Ability Enhancement Skills - II**
COURSE NAME: Quantitative Aptitude	**COURSE CODE: 26PSKL404**
SEMESTER: II	**MARKS:100**
CREDITS: 2	**TOTAL HOURS: 30**
SKILL	

COURSE OBJECTIVE:

To enable students to understand, apply, and analyze fundamental quantitative aptitude concepts including number systems, arithmetic operations, ratios, percentages, time-work and time-distance problems, interest calculations, algebraic reasoning, permutations and probability, and data interpretation techniques, for developing strong logical, analytical, and problem-solving skills required for competitive examinations and computational decision-making in computer applications.

COURSE OUTCOMES:

1. Explain and apply fundamental numerical concepts including numbers, HCF, LCM, fractions, simplification, square roots, cube roots, and averages to solve basic quantitative problems efficiently.
2. Solve and analyze numerical problems involving ages, percentages, profit and loss, ratio and proportion, partnership, surds, indices, and chain rule to develop logical quantitative reasoning skills.
3. Apply and evaluate concepts of time and work, pipes and cisterns, time and distance, and problems on trains to compute and interpret real-world rate-based problems.
4. Analyze and compute problems involving boats and streams, alligation, simple and compound interest, logarithms, and geometric measurements (area, volume, surface area) for quantitative decision-making.
5. Evaluate and interpret advanced quantitative and analytical problems involving probability, permutations and combinations, calendars, clocks, races, stocks and shares, discounting, logical reasoning series, and graphical

UNIT I

Numbers, HCF, LCM, Decimal Fractions, Simplification, Square Roots, Cube roots, Averages.

UNIT II

Problems on Numbers and Ages, Surds, Indices, Percentages, Profit and Loss, Ratio and Proportion, Partnership, Chain Rule.

UNIT III

Time and Work, Pipes and Distances. Time and Distance, Problems on Trains.

UNIT IV

Boats and Streams, Alligation, Simple Interest, Compound Interest, Logarithms, Area, Volume and Surface Area

UNIT V

Races and Games of Skill, Calendar, Clocks, Stocks and Shares, Permutation and Combination, Probability, True

discount, Banker's Discount, Height and Distances, Odd man out - Series, Tabulation, Bar Graphs, Pie Charts, Line Graphs

RECOMMENDED TEXTBOOKS:

1. R.S. Aggarwal, "Quantitative Aptitude for Competitive Examinations", Seventh Revised Edition, S. Chand and Co. Ltd., New Delhi, 2005.Communication Techniques and Skills by R. K. Chadha; DhanpatRai Publications, New Delhi

REFERENCE BOOKS:

1. Barron's Guide for GMAT, Galgotia Publications, New Delhi, 2006Excellent General English- R.B.Varshnay, R.K. Bansal, Mittal Book Depot, Malhotra

The end semester exam will have a written exam for 100 marks (50 MCQs will be given)

SEMESTER - III

PROGRAMME: MCA	**BATCH:2026-28**
PART: A	**COURSE COMPONENT: CORE - XII**
COURSE NAME: Machine Learning	**COURSE CODE: 26PMCA312**
SEMESTER: III	**MARKS:100**
CREDITS: 4	**TOTAL HOURS: 75**
THEORY	

COURSE OBJECTIVE:

To develop the ability to design, build, evaluate, and deploy machine learning and deep learning models using appropriate data preparation, classification, regression, clustering, optimization, and neural network techniques, while applying modern MLOps practices, Responsible AI principles, ethics, security, and governance for solving real-world problems.

COURSE OUTCOMES:

1. Explain machine learning concepts, learning paradigms, data preparation techniques, performance measures, and challenges involved in developing machine learning solutions.
2. Apply data preprocessing, feature engineering, model training, evaluation, and classification techniques to build and improve machine learning models.
3. Analyze regression and optimization techniques, including gradient descent methods and regularization approaches, for developing predictive machine learning models.
4. Implement decision tree and clustering algorithms to solve classification, prediction, and unsupervised learning problems using appropriate machine learning techniques.
5. .Develop and evaluate Artificial Neural Network models using Keras, including perceptron and multi-layer perceptron architectures for regression and classification tasks, and apply Agile MLOps practices along with Responsible AI, AI ethics, security, and governance principles for reliable and ethical machine learning deployment.

UNIT I **15 HOURS**

The Machine Learning Landscape: Introduction, Types of Machine Learning Systems, Supervised/Unsupervised Learning, Batch and Online Learning, Instance-Based Versus Model- Based Learning, Main Challenges of Machine Learning, Non-representative Training Data, Poor-Quality Data, Underfitting the Training Data, Testing and Validating - Frame the Problem, Select a Performance Measure, Check the Assumptions, Create a Test Set, Visualizing Geographical Data-correlation- Prepare the Data for Machine Learning Algorithms

UNIT II **15 HOURS**

Data Cleaning- Handling Text and Categorical Attributes- Feature Scaling- Transformation Pipelines- Select and Train a Model- Training and Evaluating on the Training Set- Fine-Tune Your Model- Grid Search- Randomized Search. Classification: MNIST- Training a Binary Classifier- Performance Measures- Measuring Accuracy Using Cross-Validation- Confusion Matrix- Precision and Recall. Multiclass, multilabel, and multioutput classification

UNIT III **15 HOURS**

Training Models: Linear Regression- The Normal Equation- Computational Complexity- Gradient and batch gradient Descent- Stochastic Gradient Descent- Mini-batch Gradient Descent- Polynomial Regression- Learning Curves- Ridge Regression- Lasso Regression-Logistic Regression- Estimating Probabilities- Training and Cost Function- Decision Boundaries

UNIT IV **15 HOURS**

Decision Trees: Training and Visualizing a Decision Tree- Making Predictions- Estimating Class Probabilities- The CART Training Algorithm- Unsupervised Learning Techniques- Clustering- K-Means-Limits of K-Means-Using clustering for image segmentation, preprocessing, Semi- supervised Learning- DBSCAN

UNIT V **15 HOURS**

Neural Networks and Deep Learning - Introduction to Artificial Neural Networks with Keras- Biological to Artificial Neurons-Biological Neurons-Logical Computations with Neurons- The Perceptron- Multi-Layer Perceptron and Back propagation- Regression MLPs- Classification MLPs- Implementing MLPs with Keras – Case Study on : **Agile Machine Learning Project Management and MLOps - Responsible AI, AI Ethics, Security, and Governance**

RECOMMENDED TEXTBOOKS:

1. Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow by Aurélien Géron,O'Reilly Media, Inc.

REFERENCE BOOKS:

1. Swarup Panda, *Scalable Artificial Intelligence Systems: Cloud-Native, Edge-AI, MLOps, and Governance for Real-World Deployment,* Deep Science Publishing, 2025.
2. Andrew P. McMahon, Machine Learning Engineering with Python, Packt Publishing, 2020
3. Machine Learning, "Step-by-Step Guide to Implement Machine Learning Algorithms with Python", By Rudolph Russell, 2018.
4. Introduction to Machine Learning Second Edition by Ethem Alpaydın, The MIT Press - Cambridge, Massachusetts-London.

E-LEARNING RESOURCES:

1. https://www.javatpoint.com/machine-learning
2. https://www.w3schools.com/python/python_ml_getting_started.asp
3. https://www.geeksforgeeks.org/machine-learning/

GUIDELINES TO THE QUESTION PAPER SETTERS
QUESTION PAPER PATTERN

SECTION	QUESTION COMPONENT	NUMBERS	MARKS	TOTAL
A	*Answer all the questions*	*1-10*	*2*	*20*
B	*Answer any 5 out of 7 questions*	*11-17*	*7*	*35*
C	*Answer any 3 out of 5 questions(each in 1200 words)*	*18-22*	*15*	*45*
				100

BREAK UP OF QUESTIONS

UNITS	SECTION A	SECTION B	SECTION C
I	2	2	1
II	2	1	1
III	2	2	1
IV	2	1	1
V	2	1	1
TOTAL	10	7	5
	SECTION A - 10	**SECTION B - 7**	**SECTION C - 5**

PO – CO MAPPING

Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	2	3	1	1	1	3
CO2	3	3	3	3	1	1	1	2
CO3	3	3	3	3	1	1	1	2
CO4	3	3	3	3	2	1	1	2
CO5	3	3	3	3	3	2	3	3
Average Contribution	3	3	2.8	3	1.6	1.2	1.4	2.4

PO – CO QUESTION PAPER MAPPING

CO No:	COURSE OUTCOMES	POs ADDRESSED	COGNITIVE LEVEL (K1 to K6)
CO1	Explain machine learning concepts, learning paradigms, data preparation techniques, performance measures, and challenges involved in developing machine learning solutions.	1,2,4,8	1 to 5
CO2	Apply data preprocessing, feature engineering, model training, evaluation, and classification techniques to build and improve machine learning models.	1,2,3,4	1 to 5
CO3	Analyze regression and optimization techniques, including gradient descent methods and regularization approaches, for developing predictive machine learning models.	1,2,3,4	1 to 5
CO4	**Implement** decision tree and clustering algorithms to solve classification, prediction, and unsupervised learning problems using appropriate machine learning techniques.	1,2,3,4	1 to 5
CO5	Develop and evaluate Artificial Neural Network models using Keras, including perceptron and multi-layer perceptron architectures for regression and classification tasks, and apply Agile MLOps practices along with Responsible AI, AI ethics, security, and governance principles for reliable and ethical machine learning deployment.	1,2,3,4,5,7,8	1 to 5
K1= Remember, K2= Understand, K3= Apply, K4=Analyse, K5= Evaluate, K6= Create			

Continuous Internal Assessment - Components			
Assessment Type	**Nature**	**Maximum Marks**	**Weightage**
Mid-Semester Examination	Pen and Paper test for 90 minutes	50	10
Model Examination	Written examination for 180 minutes	100	10
Assignment	Any topic related to the course	10	5
Workbook/recordbook/ notebook	Handwritten Class Notes	5	5
Class activity 1	Quiz with MCQ,	10	5
Class activity 2	Seminar / Presentation	10	5
Class activity 3	Surprise Class Tests	10	5
Attendance	Marks will be given based on the attendance percentage of respective course		5
Total			50

PROGRAMME: MCA	**BATCH:2026-28**
PART: A	**COURSE COMPONENT: CORE - XIII**
COURSE NAME: Full Stack Software Development	**COURSE CODE: 26PMCA313**
SEMESTER: III	**MARKS:100**
CREDITS:4	**TOTAL HOURS: 75**
THEORY	

COURSE OBJECTIVE:

To enable students to understand, apply, analyze, and design full stack web applications by integrating JavaScript fundamentals, React-based front-end development, Node.js and Express server-side programming, and MongoDB database management, along with advanced features such as routing, APIs, state management, server-side rendering, and secure authentication mechanisms, for developing scalable, efficient, and ethically designed MERN-based web solutions.

COURSE OUTCOMES:

1. Explain and apply JavaScript fundamentals, DOM manipulation, event handling, asynchronous execution (event loop), and MERN stack basics including Node.js, Express, React, and MongoDB to build foundational web applications.
2. Design and develop dynamic and responsive user interfaces using React concepts such as components, JSX, props, state management, event handling, forms, and CSS-based styling techniques.
3. Implement and analyze server-side applications using Node.js and Express by developing RESTful APIs, handling routing, middleware, and integrating data access operations for scalable backend systems.
4. Construct and evaluate MongoDB-based NoSQL database solutions by performing CRUD operations, schema design, querying, aggregation, replication, sharding, and deployment strategies for scalable data management.
5. Evaluate advanced full stack web applications using React-based routing, modular UI components, state handling, server-side rendering, API operations (update/delete), and MongoDB features such as aggregation and pagination, while integrating secure authentication mechanisms (Google Sign-In, session handling) and applying secure full stack development and ethical web engineering lifecycle principles for reliable and scalable web systems.

UNIT I **15 HOURS**

JAVASCRIPT AND BASICS OF MERN STACK 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 macro tasks - MERN Components - React - Node.js - Express - MongoDB - Need for MERN - Server-Less Hello World - Server Setup - nvm - Node.js – npm

UNIT II **15 HOURS**

REACT 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 **15 HOURS**

NODE.JS AND EXPRESS 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 - Automatic Server Restart - Testing - The Create API - Using the List API - Using the Create API- Error Handling - Template Engine

UNIT IV **15 HOURS**

MONGODB 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 - Aggregation - Replication - Sharding - Creating backup – Deployment.

UNIT V **15 HOURS**

ADVANCED FEATURES 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 - MongoDB Aggregate - Pagination - Higher Order Components - Search Bar - Google Sign In - Session Handling-Case Study on : **Secure Full Stack Development and Ethical Web Engineering Lifecycle**

RECOMMENDED TEXTBOOKS:

1. Pro MERN Stack, Full Stack Web App Development with Mongo, Express, React, and Node, Vasan Subramanian, A Press Publisher, 2019

REFERENCE BOOKS:

1. Andrew Hoffman, Web Application Security: Exploitation and Countermeasures, O'Reilly Media, 2020
2. Beginning MERN Stack Build and Deploy a Full Stack MongoDB, Express, React, Node.js App , Greg Lim, 2021

E-LEARNING RESOURCES:

1. http://tutorialsteacher.com
2. https://reactjs.org
3. https://nodejs.org
4. www.Expressjs.com
5. www.mongodb.com
6. https://www.oreilly.com/library/view/web-application-security/9781492053118/

GUIDELINES TO THE QUESTION PAPER SETTERS
QUESTION PAPER PATTERN

SECTION	QUESTION COMPONENT	NUMBERS	MARKS	TOTAL
A	*Answer all the questions*	*1-10*	*2*	*20*
B	*Answer any 5 out of 7 questions*	*11-17*	*7*	*35*
C	*Answer any 3 out of 5 questions(each in 1200 words)*	*18-22*	*15*	*45*
				100

BREAK UP OF QUESTIONS

UNITS	SECTION A	SECTION B	SECTION C
I	2	2	1
II	2	1	1
III	2	2	1
IV	2	1	1
V	2	1	1
TOTAL	10	7	5
	SECTION A - 10	**SECTION B - 7**	**SECTION C - 5**

PO – CO MAPPING

Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1.	3	3	3	3	2	1	2	3
CO2.	3	3	3	3	3	2	2	3
CO3	3	3	3	3	3	3	2	3
CO4	3	3	3	3	2	3	2	3
CO5	3	3	3	3	3	3	3	3
Average Contribution	3	3	3	3	2.6	2.4	2.2	3

PO – CO QUESTION PAPER MAPPING

CO No:	COURSE OUTCOMES	POs ADDRESSED	COGNITIVE LEVEL (K1 to K6)
CO1	**Explain and apply** JavaScript fundamentals, DOM manipulation, event handling, asynchronous execution (event loop), and MERN stack basics including Node.js, Express, React, and MongoDB to build foundational web applications.	1,2,3,4,8	1 to 5
CO2	**Design and develop** dynamic and responsive user interfaces using React concepts such as components, JSX, props, state management, event handling, forms, and CSS-based styling techniques.	1,2,3,4,5,8	1 to 5

CO3	**Implement and analyze** server-side applications using Node.js and Express by developing RESTful APIs, handling routing, middleware, and integrating data access operations for scalable backend systems.	1,2,3,4,5,6,8	1 to 5
CO4	**Construct and evaluate** MongoDB-based NoSQL database solutions by performing CRUD operations, schema design, querying, aggregation, replication, sharding, and deployment strategies for scalable data management.	1,2,3,4,6,8	1 to 5
CO5	**Evaluate** advanced full stack web applications using React-based routing, modular UI components, state handling, server-side rendering, API operations (update/delete), and MongoDB features such as aggregation and pagination, while integrating secure authentication mechanisms (Google Sign-In, session handling) and applying secure full stack development and ethical web engineering lifecycle principles for reliable and scalable web systems.	1,2,3,4,5,6,7,8	1 to 5
K1= Remember, K2= Understand, K3= Apply, K4=Analyse, K5= Evaluate, K6= Create			

Continuous Internal Assessment - Components			
Assessment Type	**Nature**	**Maximum Marks**	**Weightage**
Mid-Semester Examination	Pen and Paper test for 90 minutes	50	10
Model Examination	Written examination for 180 minutes	100	10
Assignment	Any topic related to the course	10	5
Workbook/recordbook/ notebook	Handwritten Class Notes	5	5
Class activity 1	Quiz with MCQ,	10	5
Class activity 2	Seminar / Presentation	10	5
Class activity 3	Surprise Class Tests	10	5
Attendance	Marks will be given based on the attendance percentage of respective course		5
Total			50

PROGRAMME: MCA	**BATCH:2026-28**
PART: A	**COURSE COMPONENT: CORE - XIV**
COURSE NAME: Cloud Computing and DevOps	**COURSE CODE: 26PMCA314**
SEMESTER: III	**MARKS:100**
CREDITS:3	**TOTAL HOURS: 60**
THEORY	

COURSE OBJECTIVE:

To analyze, design, implement, secure, and evaluate cloud-based solutions using AWS services, infrastructure automation, DevOps and DevSecOps practices, FinOps principles, cloud security governance, and sustainable cloud computing strategies to achieve scalable, reliable, cost-effective, and secure software deployment and operations

COURSE OUTCOMES:

1. Analyze the fundamental concepts, service models, scalability features, reliability mechanisms, and business benefits of AWS cloud computing for deploying and managing cloud-based solutions.
2. Apply AWS services to create and manage cloud infrastructure, including user accounts, billing controls, virtual machines, storage resources, databases, and load balancers for real-world applications.
3. Implement infrastructure automation, virtual machine management, and cloud application deployment using AWS CLI, SDKs, CloudFormation, and Elastic Beanstalk in a cloud environment.
4. Design and secure cloud-based solutions by configuring IAM policies, security groups, VPCs, serverless applications using AWS Lambda, and deployment automation tools.
5. Evaluate and apply DevOps, DevSecOps, FinOps, cloud cost optimization, sustainable cloud computing, and cloud security governance practices to enhance software delivery, operational efficiency, cost effectiveness, security, and sustainability in cloud-based environments.

UNIT I **12 HOURS**

Amazon Web Services-cloud computing- Hosting a web shop-Implementing a highly available system-Innovative and fast-growing platform- Services solve common problems-Enabling automation-Flexible capacity (scalability)- Built for failure (reliability)- Reducing time to market- Global infrastructure-Free Tier-Pay-per-use opportunities- Comparing alternatives-Exploring and Interacting with AWS services

UNIT II **12 HOURS**

Creating an AWS account-Signing In-Creating a key pair-Create a billing alarm to keep track of your AWS bill- simple example: WordPress- Creating and exploring your infrastructure- Resource groups-Load balancer-MySQL database- Network filesystem- Using virtual machines: EC2 and exploring them-Launching a virtual machine- Connecting to your virtual machine

UNIT III **12 HOURS**

Monitoring and debugging a virtual machine-Changing the size of a virtual machine-Starting a virtual machine in another data center. Programming your infrastructure - Infrastructure as Code- Installing and configuring the CLI- Controlling virtual machines with SDK: nodecc- nodecc create and list virtual machines-CloudFormation, Elastic Beanstalk,-Deploying applications in a flexible cloud environment

UNIT IV **12 HOURS**

Creating a virtual machine and run a deployment script on startup with AWS CloudFormation- Deploying a simple web application with AWS Elastic Beanstalk-Components of AWS Elastic Beanstalk- Securing your system: IAM, security groups, and VPC- Securing your AWS account-Creating a private network in the cloud: Amazon Virtual Private Cloud (VPC)- Executing your code with AWS Lambda- Building a website health check with AWS Lambda.

UNIT V **12 HOURS**

DevOps Definition- DevOps for Developers –Term DevOps- Introduction to DevOps-Tradition and Agile Project Settings- Dev vs OPS – conflicts during and after deployments- conflicts in performance. Operations as bottleneck: Horizontal Optimization, Operation and ITSM- DevOps to rescue-essence of DevOps: values and processes, Tools.- Case Study on: **Cloud Cost Optimization, FinOps and Sustainable Cloud Computing, DevSecOps and Cloud Security Governance**

RECOMMENDED TEXTBOOKS:

1. Amazon Web Services in Action, Second Edition ,MICHAEL WITTIG ANDREAS WITTIG FOREWORD BY BEN WHALEY MANNING.
2. DevOps for Developers, Michael Hutterman, Apress

REFERENCE BOOKS:

1. Beginning Serverless Computing
2. Developing with Amazon Web Services,Microsoft Azure, and Google Cloud. Amazon Web Services™ For Dummies® Published by: John Wiley & Sons.
3. The Open University, *Cloud Computing for Business*, OpenLearn, Open Educational Resources (OER),

GUIDELINES TO THE QUESTION PAPER SETTERS
QUESTION PAPER PATTERN

SECTION	QUESTION COMPONENT	NUMBERS	MARKS	TOTAL
A	*Answer all the questions*	*1-10*	*2*	*20*
B	*Answer any 5 out of 7 questions*	*11-17*	*7*	*35*
C	*Answer any 3 out of 5 questions(each in 1200 words)*	*18-22*	*15*	*45*
				100

BREAK UP OF QUESTIONS

UNITS	SECTION A	SECTION B	SECTION C
I	2	2	1
II	2	1	1
III	2	2	1
IV	2	1	1
V	2	1	1
TOTAL	10	7	5
	SECTION A - 10	**SECTION B - 7**	**SECTION C - 5**

PO – CO MAPPING

Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1.	3	3	3	3	1	2	1	3
CO2.	3	3	3	3	2	3	2	2
CO3	3	3	3	3	2	3	2	3
CO4	3	3	3	3	2	2	3	3
CO5	2	3	3	3	3	3	3	3
Average Contribution	2.8	3	3	3	2	2.6	2.2	2.8

PO – CO QUESTION PAPER MAPPING

CO No:	COURSE OUTCOMES	POs ADDRESSED	COGNITIVE LEVEL (K1 to K6)
CO1	**Analyze** the fundamental concepts, service models, scalability features, reliability mechanisms, and business benefits of AWS cloud computing for deploying and managing cloud-based solutions.	1,2,3,4,8	1 to 5
CO2	**Apply** AWS services to create and manage cloud infrastructure, including user accounts, billing controls, virtual machines, storage resources, databases, and load balancers for real-world applications.	1,2,3,4,6	1 to 5
CO3	**Implement** infrastructure automation, virtual machine management, and cloud application deployment using AWS CLI, SDKs, CloudFormation, and Elastic Beanstalk in a cloud environment.	1,2,3,4,6,8	1 to 5
CO4	**Design and secure** cloud-based solutions by configuring IAM policies, security groups, VPCs, serverless applications using AWS Lambda, and deployment automation tools.	1,2,3,4,7,8	1 to 5
CO5	Evaluate and apply DevOps, DevSecOps, FinOps, cloud cost optimization, sustainable cloud computing, and cloud security governance practices to enhance software delivery, operational efficiency, cost effectiveness, security, and sustainability in cloud-based environments.	2,3,4,5,6,7,8	1 to 5
K1= Remember, K2= Understand, K3= Apply, K4=Analyse, K5= Evaluate, K6= Create			

Continuous Internal Assessment - Components			
Assessment Type	**Nature**	**Maximum Marks**	**Weightage**
Mid-Semester Examination	Pen and Paper test for 90 minutes	50	10
Model Examination	Written examination for 180 minutes	100	10
Assignment	Any topic related to the course	10	5
Workbook/recordbook/ notebook	Handwritten Class Notes	5	5
Class activity 1	Quiz with MCQ,	10	5
Class activity 2	Seminar / Presentation	10	5
Class activity 3	Surprise Class Tests	10	5
Attendance	Marks will be given based on the attendance percentage of respective course		5
Total			50

PROGRAMME: MCA	**BATCH:2026-28**
PART: A	**COURSE COMPONENT: CORE - XV**
COURSE NAME: R programming for Statistical Analytics	**COURSE CODE: 26PMCA315**
SEMESTER: III	**MARKS:100**
CREDITS:3	**TOTAL HOURS: 60**
THEORY	

COURSE OBJECTIVE:

To enable students to understand, apply, analyze, and evaluate data science concepts, R programming techniques, data wrangling, visualization, statistical methods, and machine learning algorithms for solving real-world problems, while integrating Agile data science project management practices and secure, ethical data analytics principles for effective, responsible, and reproducible decision-making.

COURSE OUTCOMES:

1. Explain and apply fundamental concepts of data science, business analytics, and data-driven problem-solving approaches, including descriptive, diagnostic, predictive, and prescriptive analytics for understanding real-world business problems.
2. Implement and analyze basic R programming constructs and data exploration techniques using vectors, matrices, data frames, dplyr operations, and missing value handling for effective data preprocessing and manipulation.
3. Demonstrate and apply data wrangling techniques in R, including reshaping datasets, merging, grouping, and performing date-time operations for structured data transformation and analysis.
4. Analyze and visualize datasets using string manipulation, regular expressions, and ggplot2-based visualization techniques, along with performing feature engineering methods for improving data quality and model readiness.
5. Evaluate statistical and machine learning models using probability theory, inferential statistics, and supervised/unsupervised learning techniques such as linear regression, logistic regression, decision trees, and K-means clustering, while integrating Agile data science project management practices and secure, ethical data analytics principles in R for responsible and efficient data-driven decision-making.

UNIT I **12 HOURS**

The Data Science Landscape- The need to understand data science- Understanding the business problem - Process of approaching your Data-Checking reproducibility and Consistency-Making hypotheses and looking for Evidence-Analytical skills you need -Types of business Analytics-Descriptive analytics-Diagnostic analytics- Predictive analytics-Prescriptive analytics

UNIT II **12 HOURS**

R Basics: Objects-Vectors-Attributes-Matrices-Arrays-Class-Lists-Data-Frames-Functions-Loading data-Saving data-R conventions-Exploring Data: the Gap minder data-Introducing dplyr-Renaming columns-Selecting columns-Filtering rows-De-duplicating data-Arranging data-Adding new columns-Working with factors-Handling missing values-Detecting missing data visually using naniar library-Treating missing values-Deletions-Imputations

UNIT III **12 HOURS**

Wrangling Data: Pivoting data: Long and wide format data-Converting wide to long format data-Grouping and merging and summarizing data-Binding Data-Joins: Inner, outer, left, and right. Working with Dates: Creating date and date time Objects-Retrieving date and time Components-Modifying date and time components-electricity production Data-Calculations with dates and times-Durations-Periods-Interval

UNIT IV **12 HOURS**

Manipulating Strings: Essential string Manipulations-Changing case- Regular expressions-Grouping in regular expressions-Greedy and non-greedy matching- Visualizing Data: ggplot2 template and aesthetics -Visualizing distributions-Bar plots-Histograms- Density plots-Visualizing proportions-Pie charts - Stacked bar plots -Visualizing associations-Boxplot-Scatter plot-Visualizing time series using line chart- Feature Engineering: Understanding feature engineering-Transforming numeric data-Z-score scaling-Min-max scaling-Binning-Feature clipping

UNIT V **12 HOURS**

Statistics and Probability: Population sampling-Descriptive statistics - Measures of central tendency-Measures of dispersion –Probability -Inferential statistics. Introducing ML: Framing a ML problem-Supervised and Unsupervised learning -Reinforcement learning- Linear regression - Logistic regression- Decision trees - Classification trees – Clustering - K-means clustering –Case Study on : **Agile Data Science Project Management using R , Secure and Ethical Data Analytics in R**

RECOMMENDED TEXTBOOKS:

1. Abhishek Das, Data Science Essentials with R, BPB Publications, India, 2025

REFERENCE BOOKS:

1. Thomas Mailund, Aarhus, Beginning Data Science in R 4 Data Analysis, Visualization, and Modelling for the Data Scientist, 2nd ed., Thomas Mailund 2022.
2. Hadley Wickham, Mine Çetinkaya-Rundel, Garrett Grolemund, R for Data Science (2nd Edition – Online Open Textbook), O'Reilly Media (Open Online Edition, 2023

E-LEARNING RESOURCES:

1. https://r4ds.hadley.nz/

GUIDELINES TO THE QUESTION PAPER SETTERS
QUESTION PAPER PATTERN

SECTION	QUESTION COMPONENT	NUMBERS	MARKS	TOTAL
A	*Answer all the questions*	*1-10*	*2*	*20*
B	*Answer any 5 out of 7 questions*	*11-17*	*7*	*35*
C	*Answer any 3 out of 5 questions(each in 1200 words)*	*18-22*	*15*	*45*
				100

BREAK UP OF QUESTIONS

UNITS	SECTION A	SECTION B	SECTION C
I	2	2	1
II	2	1	1
III	2	2	1
IV	2	1	1
V	2	1	1
TOTAL	10	7	5
	SECTION A - 10	**SECTION B - 7**	**SECTION C - 5**

PO – CO MAPPING

Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1.	3	3	3	3	2	1	2	3
CO2.	3	3	3	3	2	1	2	3
CO3	3	3	3	3	2	1	2	3
CO4	3	3	3	3	2	1	2	3
CO5	3	3	3	3	3	3	3	3
Average Contribution	3	3	3	3	2.2	1.4	2.2	3

PO – CO QUESTION PAPER MAPPING

CO No:	COURSE OUTCOMES	POs ADDRESSED	COGNITIVE LEVEL (K1 to K6)
CO1	**Explain and apply** fundamental concepts of data science, business analytics, and data-driven problem-solving approaches, including descriptive, diagnostic, predictive, and prescriptive analytics for understanding real-world business problems.	1,2,3,4,8	1 to 5
CO2	**Implement and analyze** basic R programming constructs and data exploration techniques using vectors, matrices, data frames, dplyr operations, and missing value handling for effective data preprocessing and manipulation.	1,2,3,4,8	1 to 5
CO3	**Demonstrate and apply** data wrangling techniques in R, including reshaping datasets, merging, grouping, and performing date-time operations for structured data transformation and analysis.	1,2,3,4,8	1 to 5
CO4	**Analyze and visualize** datasets using string manipulation, regular expressions, and ggplot2-based visualization techniques, along with performing feature engineering methods for improving data quality and model readiness.	1,2,3,4,8	1 to 5
CO5	Evaluate statistical and machine learning models using probability theory, inferential statistics, and supervised/unsupervised learning techniques such as linear regression, logistic regression, decision trees, and K-means clustering, while integrating Agile data science project management practices and secure, ethical data analytics principles in R for responsible and efficient data-driven decision-making.	1,2,3,4,5,6,7,8	1 to 5
K1= Remember, K2= Understand, K3= Apply, K4=Analyse, K5= Evaluate, K6= Create			

Continuous Internal Assessment - Components			
Assessment Type	**Nature**	**Maximum Marks**	**Weightage**
Mid-Semester Examination	Pen and Paper test for 90 minutes	50	10
Model Examination	Written examination for 180 minutes	100	10
Assignment	Any topic related to the course	10	5
Workbook/recordbook/ notebook	Handwritten Class Notes	5	5
Class activity 1	Quiz with MCQ,	10	5
Class activity 2	Seminar / Presentation	10	5
Class activity 3	Surprise Class Tests	10	5
Attendance	Marks will be given based on the attendance percentage of respective course		5
Total			50

PROGRAMME: MCA	**BATCH:2026-28**
PART: A	**COURSE COMPONENT: ELECTIVE - II**
COURSE NAME: Agile Software Development	**COURSE CODE: 26PMCA3E3**
SEMESTER: III	**MARKS:100**
CREDITS:3	**TOTAL HOURS: 60**
THEORY	

COURSE OBJECTIVE:

To analyze, plan, implement, and evaluate Agile software development and organizational transition practices by applying Scrum-based project management, Agile engineering techniques, DevSecOps principles, secure coding practices, modern collaboration tools, and sustainable product development strategies to deliver high-quality, secure, customer-centric, and value-driven software solutions

COURSE OUTCOMES:

1. Analyze Agile development principles, Scrum workflow, product backlog management, transition barriers, and organizational readiness factors to facilitate successful Agile adoption.
2. Apply Agile transition planning techniques, including transition team formation, goal setting, backlog creation, sprint planning, reviews, and retrospectives to manage organizational Agile transformation.
3. Develop Agile transition strategies by preparing budgets, communication plans, training programs, role-based learning tracks, and coaching mechanisms to support Agile team effectiveness.
4. Evaluate facilities requirements, pilot project selection criteria, Agile tools, and customer involvement practices to establish an effective Agile enterprise environment.
5. Plan, monitor, and evaluate Agile software projects by applying release planning, cost estimation, backlog management, DevSecOps practices, secure coding principles, and sustainable product development strategies to deliver secure, high-quality, and value-driven software solutions.

UNIT I **12 HOURS**

Introduction - ***Why Agile?:*** Myths about Agile Development - Reasons to Stay Away from Agile.- How Your Organization Will Benefit from Agile ? - ***Transitional Concepts***: What Is Agile Development? – Workflow - Product Backlog (Prioritization , Grooming) – Sprinting - Beta Test - Defects- Workflow Summary- The Product Backlog -***Transition Barriers*****:** People Barriers - Organizational Barriers - ***Management in an Agile Environment*****:** Getting Management Ready for the Transition - An Effective Organizational Arrangement for Scrum Teams.- Hiring New Scrum Team Members- The Care and Feeding of Scrum Teams

UNIT II **12 HOURS**

Create the Transition Team : The Transition Team Lead-The Executive Sponsor - ***Define the Organizational Agreements*****:** Document the Agreements - ***Determine Transition Goals*****:** Measuring and Monitoring the Transition- Improved Productivity through Reduced Defects -Improved Performance through Increased Feature Value- Setting Control Limits to Manage Corrective Actions-Avoiding Organizational Dysfunction Caused by Measurements - ***Create the Transition Backlog*** **:** Bringing the Transition Team Together - ***Beginning Transition Sprint 1:*** The Sprint Schedule- The Transition Backlog - The Structure of the Transition Backlog -Grooming: Reducing Backlog Items to Sprint Size - Sprint Planning - Building the Sprint Backlog - Sprint Reviews for Transition Sprints-

Sprint Retrospectives for Transition Sprints.

UNIT III **12 HOURS**

Create the Transition Budget*:* Training and Coaching - Software and Hardware - Facilities.- Travel - People - ***Develop the Transition Communication Plan****:* Project Information Storage -***Create the Training Plan:*** Basic Concepts-Agile Project Structure-Roles in an Agile Project- Matching Skills to Roles- Skills Become Training Modules-The Training Modules-Defining the Tracks-Executing the Tracks-The Role of Coaches in the Agile Transition-Overload the Scrum Teams- Scrum Masters and Scrum Product Owners.

UNIT IV **12 HOURS**

Facilities Planning: Team Rooms -Setting Up a Team Room -Employee Directories- Employee Telephones-Private Spaces-Server Rooms-The Facilities Plan ***- Selecting the Pilot Projects for the Agile Transition:*** Define Your Goals-Set Organizational Expectations- Selecting Your Pilot Project-Obstacles to a Successful Pilot Project-Setting Your Project Up for Success ***- Tools in the Agile Enterprise:*** Continuous Integration/Build - Automated Testing - Sprint and Backlog Management - Team Communication ***Managing Customer Involvement:*** Selecting the Right Customer - Managing the Involved Customer

UNIT V **12 HOURS**

Agile Project Management—Getting Started: Scheduling in an Agile Project-Scheduling Challenges-Determining the Project's Estimated Costs- Planning and Staffing-Specialization and the Unbalanced Backlog- Architecture Definition- Unprepared Backlog Items - Getting Your Project Started.-: Creating the Release Goals, Create/Update Defined Processes and Policies, Create/Update the DONEness Definition, Determine Team Staffing, Prepare the Product Backlog for Grooming, Create the Sprint Schedule, Sprint Schedules and Large Project Considerations, The Unwanted Stabilization Sprint, When the Automated Testing Isn't Sufficient, Begin Backlog Grooming Work Sessions - ***Agile Project Management:*** Monitoring, Reporting, and Controlling - Burning Down the Product Backlog- The Release Plan-Feature Completion- Controlling the Project.- Case Study on **Agile Software Engineering with DevSecOps and Secure Coding, Agile Product Development for Sustainable and Responsible Software Systems**

RECOMMENDED TEXTBOOKS:

1. Schiel, J. (2010). Enterprise-Scale Agile Software Development. CRC Press / Taylor & Francis Group. ISBN: 978-1-4398-0321-9.

REFERENCE BOOKS:

1. Armando Fox and David Patterson, *Engineering Software as a Service: An Agile Approach Using Cloud Computing*, Open Access Edition, Strawberry Canyon LLC, Latest Edition. (Freely downloadable online)
2. Schwaber, K., & Beedle, M. (2002). Agile Software Development with Scrum. Prentice Hall.
3. Cohn, M. (2004). User Stories Applied: For Agile Software Development. Addison-Wesley Professional.
4. Cohn, M. (2005). Agile Estimating and Planning. Prentice Hall.
5. Derby, E., & Larsen, D. (2006). Agile Retrospectives: Making Good Teams Great. Pragmatic Bookshelf.

GUIDELINES TO THE QUESTION PAPER SETTERS
QUESTION PAPER PATTERN

SECTION	QUESTION COMPONENT	NUMBERS	MARKS	TOTAL
A	*Answer all the questions*	*1-10*	*2*	*20*
B	*Answer any 5 out of 7 questions*	*11-17*	*7*	*35*
C	*Answer any 3 out of 5 questions(each in 1200 words)*	*18-22*	*15*	*45*
				100

BREAK UP OF QUESTIONS

UNITS	SECTION A	SECTION B	SECTION C
I	2	2	1
II	2	1	1
III	2	2	1
IV	2	1	1
V	2	1	1
TOTAL	10	7	5
	SECTION A - 10	**SECTION B - 7**	**SECTION C - 5**

PO – CO MAPPING

Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1.	2	3	3	2	3	2	1	3
CO2.	2	3	3	3	3	3	1	3
CO3	1	3	3	2	3	3	2	3
CO4	1	3	3	3	3	2	1	3
CO5	3	3	3	3	3	3	3	3
Average Contribution	1.8	3	3	2.6	3	2.6	1.6	3

PO – CO QUESTION PAPER MAPPING

CO No:	COURSE OUTCOMES	POs ADDRESSED	COGNITIVE LEVEL (K1 to K6)
CO1	**Analyze** Agile development principles, Scrum workflow, product backlog management, transition barriers, and organizational readiness factors to facilitate successful Agile adoption.	2,3,5,8	1 to 5
CO2	**Apply** Agile transition planning techniques, including transition team formation, goal setting, backlog creation, sprint planning, reviews, and retrospectives to manage organizational Agile transformation.	2,3,4,5,6,8	1 to 5

CO3	**Develop** Agile transition strategies by preparing budgets, communication plans, training programs, role-based learning tracks, and coaching mechanisms to support Agile team effectiveness.	2,3,5,6,8	1 to 5
CO4	**Evaluate** facilities requirements, pilot project selection criteria, Agile tools, and customer involvement practices to establish an effective Agile enterprise environment.	2,3,4,5,8	1 to 5
CO5	Plan, monitor, and evaluate Agile software projects by applying release planning, cost estimation, backlog management, DevSecOps practices, secure coding principles, and sustainable product development strategies to deliver secure, high-quality, and value-driven software solutions.	1,2,3,4,5,6,7,8	1 to 5
K1= Remember, K2= Understand, K3= Apply, K4=Analyse, K5= Evaluate, K6= Create			

Continuous Internal Assessment - Components			
Assessment Type	**Nature**	**Maximum Marks**	**Weightage**
Mid-Semester Examination	Pen and Paper test for 90 minutes	50	10
Model Examination	Written examination for 180 minutes	100	10
Assignment	Any topic related to the course	10	5
Workbook/recordbook/ notebook	Handwritten Class Notes	5	5
Class activity 1	Quiz with MCQ,	10	5
Class activity 2	Seminar / Presentation	10	5
Class activity 3	Surprise Class Tests	10	5
Attendance	Marks will be given based on the attendance percentage of respective course		5
Total			50

PROGRAMME: MCA	**BATCH:2026-28**
PART: A	**COURSE COMPONENT: ELECTIVE - II**
COURSE NAME: Software Engineering	**COURSE CODE: 26PMCA3E7**
SEMESTER: III	**MARKS:100**
CREDITS:3	**TOTAL HOURS: 60**
THEORY	

COURSE OBJECTIVE:

To analyze, apply, and evaluate software engineering principles, process models, agile methodologies, requirements engineering techniques, software design practices, quality assurance frameworks, configuration management, and project management metrics for the systematic development, assessment, and maintenance of high-quality software systems.

COURSE OUTCOMES:

1. Explain and compare software engineering principles, software process models, and development methodologies to select appropriate process frameworks for software development projects.
2. Apply agile software development practices, teamwork strategies, software engineering principles, and collaborative tools in the development of software systems.
3. Analyze software requirements using elicitation techniques, use cases, UML-based analysis models, and requirements engineering practices to develop effective requirement specifications.
4. Design and evaluate software architectures and quality assurance mechanisms by applying software design concepts, quality models, review techniques, and SQA practices.
5. Evaluate and manage software quality, reliability, configuration management, and project metrics to ensure effective software project execution and continuous process improvement.

UNIT I **12 HOURS**

Defining Software-Software Application Domains - Legacy Software- Changing Nature of Software- The Software Process- Software Engineering Practice - Software Development Myths - A Generic Process Model : Defining a Framework Activity-Identifying a Task Set- Process Patterns- Process Assessment and Improvement- The Waterfall Model: Incremental Process Models- Evolutionary Process Models- Specialized Process Models: Component-Based Development- The Formal Methods Model -Aspect-Oriented Software Development. Personal and Team Process Models

UNIT II **12 HOURS**

Agility: Cost of Change- Agile Process- Extreme Programming- Other Agile Process Models: Scrum- Dynamic Systems Development Method- Agile Modeling- Agile Unified Process- Tool Set for the Agile Process- Characteristics of a Software Engineer: Software Team and Team Structure- Agile Teams- Impact of Social Media- Software Engineering Using the Cloud- Collaboration Tools. Software Engineering Knowledge: Core Principles- Principles That Guide Each Framework Activity- Work Practices.

UNIT III **12 HOURS**

Requirements Engineering: Establishing the Groundwork- Eliciting Requirements- Developing Use Cases- Building the Analysis Model- Requirements Analysis- Scenario-Based Modeling- UML Models That Supplement the Use Case- Identifying Analysis Classes- Specifying Attributes- Defining Operations- Class-Responsibility-Collaborator Modeling- Associations and Dependencies- Creating a Behavioral Model: Identifying Events with the Use Case- State Representations- Patterns for

Requirements Modeling- Requirements Modeling for Web and Mobile Apps.

UNIT IV **12 HOURS**

The Design Process- Design Concepts- The Design Model- quality concepts: Software Quality- Garvin's Quality Dimensions- McCall's Quality Factors- ISO 9126 Quality Factors- Software Quality Dilemma- Cost Impact of Software Defects- Defect Amplification and Removal- Review Metrics and Their Use- Formal Technical Reviews- software quality assurance: Elements of Software Quality Assurance

UNIT V **12 HOURS**

SQA Processes and Product Characteristics- SQA Tasks, Goals, and Metrics- Statistical Software Quality Assurance- Software Reliability- The ISO 9000 Quality Standards- The SQA Plan- Software Configuration Management- Version Control- Change Control- Impact Management- Status Reporting- Content Management- Project management concepts: people, product, process- Metrics in the Process and Project Domains- Software Measurement.

RECOMMENDED TEXTBOOKS:

1. Roger. S. Pressman, 2015, Software Engineering A Practitioner‟s approach, Eighth Edition,Tata McGraw-Hill, New Delhi.

GUIDELINES TO THE QUESTION PAPER SETTERS
QUESTION PAPER PATTERN

SECTION	QUESTION COMPONENT	NUMBERS	MARKS	TOTAL
A	*Answer all the questions*	*1-10*	*2*	*20*
B	*Answer any 5 out of 7 questions*	*11-17*	*7*	*35*
C	*Answer any 3 out of 5 questions(each in 1200 words)*	*18-22*	*15*	*45*
				100

BREAK UP OF QUESTIONS

UNITS	SECTION A	SECTION B	SECTION C
I	2	2	1
II	2	1	1
III	2	2	1
IV	2	1	1
V	2	1	1
TOTAL	10	7	5
	SECTION A - 10	**SECTION B - 7**	**SECTION C - 5**

PO – CO MAPPING

Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1.	3	3	3	2	2	3	2	3
CO2.	3	3	3	3	3	3	2	3
CO3	3	3	3	3	2	2	1	2
CO4	3	3	3	3	2	2	3	2
CO5	3	3	3	3	2	3	3	3
Average Contribution	3	3	3	2.8	2.2	2.6	2.2	2.6

PO – CO QUESTION PAPER MAPPING

CO No:	COURSE OUTCOMES	POs ADDRESSED	COGNITIVE LEVEL (K1 to K6)
CO1	**Explain and compare** software engineering principles, software process models, and development methodologies to select appropriate process frameworks for software development projects.	1,2,3,6,8	1 to 5
CO2	**Apply** agile software development practices, teamwork strategies, software engineering principles, and collaborative tools in the development of software systems.	1,2,3,4,5,6,8	1 to 5
CO3	**Analyze** software requirements using elicitation techniques, use cases, UML-based analysis models, and requirements engineering practices to develop effective requirement specifications.	1,2,3,4	1 to 5
CO4	**Design and evaluate** software architectures and quality assurance mechanisms by applying software design concepts, quality models, review techniques, and SQA practices.	1,2,3,4,7	1 to 5
CO5	**Evaluate and manage** software quality, reliability, configuration management, and project metrics to ensure effective software project execution and continuous process improvement.	1,2,3,4,6,7,8	1 to 5
K1= Remember, K2= Understand, K3= Apply, K4=Analyse, K5= Evaluate, K6= Create			

Continuous Internal Assessment - Components			
Assessment Type	**Nature**	**Maximum Marks**	**Weightage**
Mid-Semester Examination	Pen and Paper test for 90 minutes	50	10
Model Examination	Written examination for 180 minutes	100	10
Assignment	Any topic related to the course	10	5
Workbook/recordbook/ notebook	Handwritten Class Notes	5	5
Class activity 1	Quiz with MCQ,	10	5
Class activity 2	Seminar / Presentation	10	5
Class activity 3	Surprise Class Tests	10	5
Attendance	Marks will be given based on the attendance percentage of respective course		5
Total			50

PROGRAMME: MCA	**BATCH:2026-28**
PART: A	**COURSE COMPONENT: ELECTIVE - II**
COURSE NAME: Software Project Management	**COURSE CODE: 26PMCA3E8**
SEMESTER: III	**MARKS:100**
CREDITS:3	**TOTAL HOURS: 60**
THEORY	

COURSE OBJECTIVE:

To analyze, plan, evaluate, and manage software projects by applying project planning, estimation, scheduling, risk management, resource allocation, contract management, team leadership, and ethical decision-making principles to ensure successful and responsible software project execution

COURSE OUTCOMES:

1. Analyze software project requirements, stakeholder needs, project characteristics, and feasibility factors to develop effective project plans and evaluate project viability using appropriate estimation and cost-benefit techniques.
2. Apply suitable software process models, effort estimation techniques, and activity scheduling methods to plan and manage software development projects efficiently.
3. Analyze and evaluate project risks, resource allocation strategies, project progress metrics, and change control mechanisms to ensure successful project execution and monitoring.
4. Apply contract management principles, team organization strategies, leadership practices, and decision-making techniques to manage software project teams and stakeholder relationships effectively.
5. Evaluate ethical, legal, privacy, and security issues associated with information technology systems and recommend responsible practices for professional IT conduct.

UNIT I **12 HOURS**

Introduction to Software Project Management- Software project versus other types of project- problems- management control- Stakeholders- Requirement Specification – Information and control in organizations, step wise project planning- Select-identify scope and objectives identify project infrastructure- Analyse project characteristics- products and activities- Estimate effort for each activity- Allocate resources- Review/ publicize plan- Execute plan and lower levels of planning. Project evaluation- Strategic assessment- technical assessment cost-benefit analysis- cash flow forecasting- cost-benefit evaluation techniques risk evaluation.

UNIT II **12 HOURS**

Selection of an appropriate project approach- choosing technologies technical plan contents list- choice of process models- structured methods-rapid application development- waterfall model - process model- spiral model software prototyping- ways of categorizing prototypes- tools- incremental delivery- selecting process model -Software effort estimation- introduction where- problems with over and under estimates- the basis for software estimating software effort estimation technique- expert judgment- COCOMO -Activity Planning- sequencing and scheduling activities- network planning models- representing lagged activities- forward pass- backward pass identifying the critical path.

UNIT III **12 HOURS**

Risk Management- nature of risk- managing- identification-analysis reducing- evaluating- z values. Resource allocation- nature of resources requirements- scheduling- critical paths- counting the cost- resource schedule cost schedule- scheduling sequence. Monitoring and control- creating the frame work- collecting the data- visualizing the progress- cost monitoring- earned value- prioritizing, monitoring- Change control.

UNIT IV **12 HOURS**

Managing contracts- types of contract- stages in contract placement terms of a contract-contract management- acceptance. Managing people and organizing teams- organizational behaviour background- selecting the right person for the job- instruction in the best methods-motivation- decision making leadership- organizational structures

UNIT V **12 HOURS**

Ethics in Information Technology – an Overview of Ethics – Ethics for IT professionals and IT users, Computer and internet crime – privacy workplace monitoring – advanced surveillance technology.

RECOMMENDED TEXTBOOKS:

1. Hughes and M. Cottrell, 2005, Software Project Management, 4th Edition, Tata McGraw Hill, New Delhi.

REFERENCE BOOKS:

1. K. Heldman, 2005, Project Management Professionals, 3 rd. Edition, Wiley Dreamtech
2. Bhforooz& Hudson, 2004, Software Engineering, Oxford Press.

GUIDELINES TO THE QUESTION PAPER SETTERS
QUESTION PAPER PATTERN

SECTION	QUESTION COMPONENT	NUMBERS	MARKS	TOTAL
A	*Answer all the questions*	*1-10*	*2*	*20*
B	*Answer any 5 out of 7 questions*	*11-17*	*7*	*35*
C	*Answer any 3 out of 5 questions(each in 1200 words)*	*18-22*	*15*	*45*
				100

BREAK UP OF QUESTIONS

UNITS	SECTION A	SECTION B	SECTION C
I	2	2	1
II	2	1	1
III	2	2	1
IV	2	1	1
V	2	1	1
TOTAL	10	7	5
	SECTION A - 10	**SECTION B - 7**	**SECTION C - 5**

PO – CO MAPPING

Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1.	2	3	3	2	2	3	2	2
CO2.	2	3	3	3	2	3	1	2
CO3	2	3	3	3	3	3	2	3
CO4	1	3	3	2	3	3	3	2
CO5	1	3	2	2	2	2	3	3
Average Contribution	1.6	3	2.8	2.4	2.4	2.8	2.2	2.4

PO – CO QUESTION PAPER MAPPING

CO No:	COURSE OUTCOMES	POs ADDRESSED	COGNITIVE LEVEL (K1 to K6)
CO1	**Analyze** software project requirements, stakeholder needs, project characteristics, and feasibility factors to develop effective project plans and evaluate project viability using appropriate estimation and cost-benefit techniques.	2,3,6	1 to 5
CO2	**Apply** suitable software process models, effort estimation techniques, and activity scheduling methods to plan and manage software development projects efficiently.	2,3,4,6	1 to 5
CO3	**Analyze and evaluate** project risks, resource allocation strategies, project progress metrics, and change control mechanisms to ensure successful project execution and monitoring.	2,3,4,5,6,8	1 to 5
CO4	**Apply** contract management principles, team organization strategies, leadership practices, and decision-making techniques to manage software project teams and stakeholder relationships effectively.	2,3,5,6,7	1 to 5
CO5	**Evaluate** ethical, legal, privacy, and security issues associated with information technology systems and recommend responsible practices for professional IT conduct.	2,7,8	1 to 5
K1= Remember, K2= Understand, K3= Apply, K4=Analyse, K5= Evaluate, K6= Create			

Continuous Internal Assessment - Components			
Assessment Type	**Nature**	**Maximum Marks**	**Weightage**
Mid-Semester Examination	Pen and Paper test for 90 minutes	50	10
Model Examination	Written examination for 180 minutes	100	10
Assignment	Any topic related to the course	10	5
Workbook/recordbook/ notebook	Handwritten Class Notes	5	5
Class activity 1	Quiz with MCQ,	10	5
Class activity 2	Seminar / Presentation	10	5
Class activity 3	Surprise Class Tests	10	5
Attendance	Marks will be given based on the attendance percentage of respective course		5
Total			50

PROGRAMME: MCA	**BATCH:2026-28**
PART: A	**COURSE COMPONENT: CORE - XVI**
COURSE NAME: Practical V : R programming for Statistical Analytics Lab	**COURSE CODE: 26PMCA316P**
SEMESTER: III	**MARKS:100**
CREDITS:3	**TOTAL HOURS: 45**
PRACTICAL	

COURSE OBJECTIVE:

To enable students to apply R programming techniques and analytical methods for data preprocessing, visualization, statistical analysis, and the implementation of machine learning models to solve data-driven problems effectively

COURSE OUTCOMES:

1. Apply R programming constructs, data structures, and statistical functions to develop programs for mathematical computations, data manipulation, and analytical tasks.
2. Perform data preprocessing, visualization, and exploratory data analysis using R packages and graphical techniques to derive meaningful insights from datasets.
3. Implement machine learning algorithms such as Linear Regression, Multiple Linear Regression, Logistic Regression, K-Nearest Neighbors, Decision Trees, and Apriori for predictive and descriptive analytics.
4. Analyze datasets using appropriate statistical measures, visualization methods, and data mining techniques to identify patterns, relationships, and trends.
5. Develop data-driven solutions using R programming by integrating data handling, preprocessing, visualization, and machine learning techniques for real-world analytics applications.

EXERCISES:

1. Write an R program to construct a calculator.
2. R program for Fibonacci series
3. R program to find whether a given number is prime or not.
4. R Programming Using ggplot() to construct (Histogram, Scatter plots, box plots etc)
5. Data Pre-processing in R
6. Implementing A prior algorithm in R
7. Implementing Linear Regression R
8. Implementing Multiple Linear Regression in R
9. Implementing Logistic Regression in R
10. Implementing Nearest Neighbors in R
11. Implementing Decision Tree in R
12. Write a R program to create a Data frames which contain details of employees anddisplay the details.
13. Write a R program to add and delete a row from the data frame
14. Write a R program to add and delete a column from the data frame
15. R Program to implement basic statistical operations.

PO – CO MAPPING

Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1.	**3**	**3**	**3**	**3**	1	1	1	**2**
CO2.	**3**	**3**	**3**	**3**	1	1	1	**2**
CO3	**3**	**3**	**3**	**3**	1	1	1	**3**
CO4	**3**	**3**	**3**	**3**	1	1	1	**3**
CO5	**3**	**3**	**3**	**3**	**2**	**2**	1	**3**
Average Contribution	3	3	3	3	1.2	1.2	1	2.6

Continuous Internal Assessment - Components			
Assessment Type	**Nature**	**Maximum Marks**	**Weightage**
Mid-Semester Examination	Practical examination with programming/problem-solving tasks for 90 minutes	50	10
Model Examination	Practical examination with programming/problem-solving tasks for 180 minutes	100	10
Assignment	Assignment based on experimental concepts	10	5
Workbook/recordbook/ notebook	Handwritten record of experiments, observations, results, and conclusions	10	5
Class activity 1	Debugging Test	5	5
Class activity 2	Debugging Test	5	5
Class activity 3	Quiz with MCQ	5	5
Attendance	Marks will be given based on the attendance percentage of respective course		5
Total			50

PROGRAMME: MCA	**BATCH:2026-28**
PART: A	**COURSE COMPONENT: CORE - XVII**
COURSE NAME: Practical VI : Full Stack Software Development Lab	**COURSE CODE: 26PMCA317P**
SEMESTER: III	**MARKS:100**
CREDITS:3	**TOTAL HOURS: 45**
PRACTICAL	

COURSE OBJECTIVE:

To enable students to apply, analyze, and design full stack web applications by integrating modern JavaScript, React-based front-end development, Node.js and Express back-end services, and MongoDB database operations, including RESTful API handling, state management, routing, session handling, and deployment of scalable and responsive web solutions using contemporary MERN stack technologies.

COURSE OUTCOMES:

1. Apply modern JavaScript features, DOM manipulation, browser events, modules, and advanced iteration concepts to develop interactive front-end web applications.
2. Design and develop dynamic user interfaces using React components, JSX, state management, props handling, event handling, forms, and CSS/SaaS styling techniques.
3. Implement and integrate server-side applications using Node.js and Express by developing RESTful APIs and handling client-server communication effectively.
4. Construct and manage NoSQL databases using MongoDB by performing CRUD operations, querying data, and exploring advanced features such as aggregation, replication, and sharding.
5. Develop and evaluate full-stack MERN applications with features such as routing, session handling, server-side rendering, and asynchronous state management for scalable web solutions.

EXERCISES:

1. Modern JavaScript features-based programming
2. Simple programs in MERN environment
3. REACT based programming
4. Exploring stateless components
5. Designing components with React CSS and SaaS
6. Node and Express based web development Handling of various APIs associated with Node.js
7. Setting up MongoDB and handling data manipulation
8. Querying the MongoDB databases
9. Exploring Agrégation, Replication, Sharding and other features in MongoDB
10. Experiments on React Router and various APIs
11. Server rendering and handling states
12. Session handling
13. Simple exercise on JavaScript Objects, Generators, advanced iteration, and Modules.
14. Working with DOM tree, node properties, browser events, UI Events, Forms, controls,
15. Document and resource loading, Mutation observer, microtasks and macrotasks.
16. Front end UI development with React JSX, components, React Classes, Composing
17. Components, React state, Async State, Event Handling, Stateless Components. Working with React Forms - React CSS - React SaaS
18. Application backend development with Node.js and Express
19. Handling NoSQL data using MongoDB and manipulating data and management

PO – CO MAPPING

Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1.	3	3	3	3	2	1	2	3
CO2.	3	3	3	3	3	2	2	3
CO3	3	3	3	3	3	3	2	3
CO4	3	3	3	3	2	3	2	3
CO5	3	3	3	3	3	3	3	3
Average Contribution	3	3	3	3	2.6	2.4	2.2	3

Continuous Internal Assessment - Components			
Assessment Type	**Nature**	**Maximum Marks**	**Weightage**
Mid-Semester Examination	Practical examination with programming/problem-solving tasks for 90 minutes	50	10
Model Examination	Practical examination with programming/problem-solving tasks for 180 minutes	100	10
Assignment	Assignment based on experimental concepts	10	5
Workbook/recordbook/ notebook	Handwritten record of experiments, observations, results, and conclusions	10	5
Class activity 1	Debugging Test	5	5
Class activity 2	Debugging Test	5	5
Class activity 3	Quiz with MCQ	5	5
Attendance	Marks will be given based on the attendance percentage of respective course		5
Total			50

PROGRAMME: MCA	**BATCH:2026-28**
PART: B	**COURSE COMPONENT: Ability Enhancement Skills - III**
COURSE NAME: Data Visualization	**COURSE CODE: 26PSKL406**
SEMESTER: III	**MARKS:100**
CREDITS:2	**TOTAL HOURS: 30**
SKILL	

COURSE OBJECTIVE:

To enable students to apply data visualization techniques using Tableau by connecting, transforming, analyzing, and visualizing data through charts, geographical maps, calculated fields, and interactive dashboards for effective data-driven decision-making.

COURSE OUTCOMES:

1. Explain the fundamentals of data analytics and Tableau concepts, including data connections, measures, dimensions, and field classifications used in data visualization.
2. Apply data integration techniques in Tableau by connecting, accessing, and joining data from files, servers, and cloud-based sources for visualization tasks.
3. Construct appropriate visualizations using Tableau charts such as bar charts, line charts, pie charts, histograms, and scatter plots to represent and analyze data effectively.
4. Analyze geographical data using Tableau mapping techniques, including filled maps, symbol maps, density maps, and visualization features to derive location-based insights.
5. Design interactive dashboards by utilizing calculated fields, formatting options, and multiple visualizations to communicate analytical insights effectively.

UNIT I

Tableau Foundations: The cycle of analytics-Connecting to Data-Measures & Dimensions-Discrete and Continuous Fields

UNIT II

Working with Data in Tableau : Connecting to data on a file-Connecting to data on a server-Connecting to data in the cloud-Joining Tables

UNIT III

CHARTS : Bar Chart , Line Chart, Pie Chart, Histogram- Scatter Plot

UNIT IV

Geographical Visualization : Filled maps-Symbol Maps-Density Maps-Using Show Me

UNIT V

Calculated field – Formatting and Dashboard Interface : Working with String Functions-Basic Arithmetic Calculations-Building the Dashboard- Combining multiple visualizations into a dashboard

RECOMMENDED TEXTBOOKS:

1. Joshua N.Milligan, "Learning Tableau 2019", 3rd Edition, Packt .

PROGRAMME: MCA	**BATCH:2026-28**
PART: B	**COURSE COMPONENT: Internship**
COURSE NAME: Internship	**COURSE CODE: 26PINT401**
SEMESTER: III	**MARKS:100**
CREDITS: 2	**TOTAL HOURS: -**
SKILL	

COURSE OBJECTIVE:

To facilitate Experiential Learning for a deeper understanding of theoretical concepts in a real-world environment

COURSE OUTCOMES:

1. **Apply academic knowledge** to real-world IT problems.
2. Develop **domain knowledge**, **multidisciplinary, and professional skills**.
3. **Gain industry exposure** through practical experience.
4. **Practice ethical and sustainabl**e computing approaches
5. Enhance **employability skills.**

COURSE DESCRIPTION:

- **Duration of the Internship**: Internships are a mandatory part of PG curricula in the College. The minimum duration of the internship is 4 to 6 Weeks after the second semester, during summer vacation.
- **Letter of Acceptance**: Students need to obtain a Letter of Acceptance from the external organization by submitting a requisition letter from the Institution's letterhead.
- **Certificate / Letter of Completion**: The certificate / letter of completion must be submitted by the student after the completion of his / her internship training. The Certificate/letter of completion shall be printed on the Company's / Organization letter head.
- **Proforma for Evaluation of the Intern**: The intern is evaluated by both the external internship guide and internal internship coordinator.
- **Feedback Form**: The student must complete the online feedback through his / her login credential. It shall serve to clarify the internship educational purpose and ensure an understanding of the total learning experience.
- **Students have to present their learning soon after the college is reopened.**
- **The course will be evaluated after the completion of the third semester.**

SEMESTER - IV

PROGRAMME: MCA	**BATCH:2026-28**
PART: A	**COURSE COMPONENT: ELECTIVE - III**
COURSE NAME: Edge Computing	**COURSE CODE: 26PMCA3E4**
SEMESTER: IV	**MARKS:100**
CREDITS: 3	**TOTAL HOURS: 60**
THEORY	

COURSE OBJECTIVE:

To enable learners to understand, analyze, and apply the principles of modern and emerging computing paradigms including edge, cloud, and distributed systems, and to design and evaluate secure, scalable, and intelligent edge computing solutions integrated with IoT, data analytics, blockchain technologies, and Agile–DevOps practices for real-world applications across diverse domains.

COURSE OUTCOMES:

1. Explain the fundamental computing paradigms including parallel, distributed, cluster, grid, utility, cloud, and emerging models such as fog, edge, and ubiquitous computing to understand modern computing ecosystems.
2. Apply the concepts of edge computing architecture and IoT-enabled systems using appropriate tools, networking models, and simulators to analyze edge-based environments.
3. Analyze edge data analytics techniques, machine learning integration, and security mechanisms to evaluate data processing, storage, and protection in edge computing systems.
4. Evaluate blockchain concepts, architectures, and platforms and integrate blockchain with edge computing to assess decentralized and secure computing solutions.
5. Evaluate edge computing-based solutions by analyzing real-world use cases and healthcare/industrial IoT case studies, integrating Agile–DevOps lifecycle practices along with security, ethics, and blockchain-based trust management for reliable and scalable edge analytics systems

UNIT I **12 HOURS**

Computing Paradigms: Introduction to Computing- The Major Impacts of Computing- Parallel computing: Shared Memory Systems, Distributed Memory Systems, Hybrid Model – Distributed Computing – Cluster Computing: High-Performance Clusters, Load Balancing Clusters, High Availability Clusters – Utility Computing – Grid Computing- Cloud Computing : Characteristics of Cloud Environment, Cloud Models – Other Computing Paradigms: Ubiquitous Computing, Jungle Computing, Fog Computing

UNIT II **12 HOURS**

Edge Computing and its Essentials: Introduction – Edge Computing Architecture: Edge Devices, Edge Server Cluster, Cloud Server – Background Essentials: IoT Devices: Mobile Phone-Based Sensors, Medical Sensors, Neural Sensors, Environmental and Chemical Sensors, Radio Frequency Identification, Actuators – Networking Architecture- Network Management and Control - Edge Computing State of the Art Interfaces and Devices: Middleware, Application Interfaces – Edge Computing Simulators: PureEdgeSim, IOTSIM-EDGE, IFOGSIM, EDGECLOUDSIM.

UNIT III **12 HOURS**

Edge Analytics: Types of Data – Data Analytics – Goals of Data Analytics – Domains benefiting from Big Data Analytics – Real time applications of Data Analytics – Phases of Data Analytics – Types of

Data Analytics: Descriptive, Diagnostic, Predictive, Prescriptive – Edge Data Analytics – Potential of Edge Analytics – Architecture of Edge Analytics – Machine Learning for Edge Devices – Edge Analytics: Case Study16.**Edge Data Storage Security:** Data Security – Data Confidentiality-Authentication – Privacy Preserving Schemes – Edge-Based Attack Detection and Prevention

UNIT IV **12 HOURS**

Blockchain and Edge Computing Systems: History of Blockchain – Distributed Ledger technology – Role of P2P Architecture in Blockchain – Blockchain Cryptography- Characteristics of Blockchain- Benefits and Limitations of Blockchain- Types of Blockchain – Blockchain Architecture and Fundamentals- Blockchain Platforms: Ethereum, Hyperledger- Polkadot Network. - Edge Computing with Blockchain.

UNIT V **12 HOURS**

Edge Computing Use Cases and Case Studies: Use Cases – Edge Computing High-Potential Use cases: Autonomous Vehicles, Smart Cities, Industrial Automation, Network Functions, Gaming, Content Delivery, Financial Sector, Augmented Reality, Healthcare Sector- Realization of Edge Computing in Healthcare, ensuring Storage Security: Devices and Setup, Case Study 1: pulse Oximeter to detect ARDS in Edge server, Case Study 2: Blood Pressure Monitor to predict Hypotension in Edge Server, Case Study 3 : Body Composition Scale to detect Heat Index in Edge Server, Use Case – Edge Computing / Analytics in Industrial IOT -Case Study on : **Edge Computing Project Lifecycle using Agile + DevOps- Security, Ethics, and Trust Management in Edge-Blockchain Systems**

RECOMMENDED TEXTBOOKS:

1. K. Anitha Kumari, G.Sudha Sadasivam, D. Dharani, M.Niranjanamurthy, Edge Computing: Fundamentals, Advances and Applications, CRC Press, 1st Edition, 2022.
2. Seong-eun Yoo, Taehong Kim, Youngsoo Kim, Edge/Fog Computing Technologies for IoT Infrastructure, MDPI Books, 2021

REFERENCE BOOKS:

1. Kevin Lee and Ka Lok Man, *Edge Computing for Internet of Things*, MDPI, 2022
2. Flavio Bonomi, Rodolfo Milito, Preethi Natarajan, Jiang Zhu, Fog Computing and Its Role in the Internet of Things (IoT), IEEE Communications Society (IEEE Journal / White Paper Series), 2012
3. Arshdeep Bahga, Vijay Madisetti, Blockchain Technology: Principles and Applications, VPT (Virtually Published Textbook / Open Access Book), 2017

E-LEARNING RESOURCES:

1. https://www.tutorialspoint.com/edge-computing/index.htm
2. https://www.geeksforgeeks.org/computer-networks/edge-computing/
3. https://www.dropbox.com/s/0b1g8q9k7z3x8qk/Blockchain_Technology_Principles_and_Applications.pdf?dl=0
4. https://www.researchgate.net/publication/260245812_Fog_Computing_and_Its_Role_in_the_Internet_of_Things

GUIDELINES TO THE QUESTION PAPER SETTERS
QUESTION PAPER PATTERN

SECTION	QUESTION COMPONENT	NUMBERS	MARKS	TOTAL
A	*Answer all the questions*	*1-10*	*2*	*20*
B	*Answer any 5 out of 7 questions*	*11-17*	*7*	*35*
C	*Answer any 3 out of 5 questions(each in 1200 words)*	*18-22*	*15*	*45*
				100

BREAK UP OF QUESTIONS

UNITS	SECTION A	SECTION B	SECTION C
I	2	2	1
II	2	1	1
III	2	2	1
IV	2	1	1
V	2	1	1
TOTAL	10	7	5
	SECTION A - 10	**SECTION B - 7**	**SECTION C - 5**

PO – CO MAPPING

Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1.	3	3	2	2	1	1	1	3
CO2.	3	3	3	3	2	1	2	3
CO3	3	3	3	3	1	1	3	3
CO4	3	3	3	3	1	1	3	3
CO5	3	3	3	3	3	3	3	3
Average Contribution	3	3	2.8	2.8	1.6	1.4	2.4	3

PO – CO QUESTION PAPER MAPPING

CO No:	COURSE OUTCOMES	POs ADDRESSED	COGNITIVE LEVEL (K1 to K6)
CO1	Explain the fundamental computing paradigms including parallel, distributed, cluster, grid, utility, cloud, and emerging models such as fog, edge, and ubiquitous computing to understand modern computing ecosystems.	1,2,8	1 to 5
CO2	Apply the concepts of edge computing architecture and IoT-enabled systems using appropriate tools, networking models, and simulators to analyze edge-based environments.	1,2,3,4,8	1 to 5

CO3	Analyze edge data analytics techniques, machine learning integration, and security mechanisms to evaluate data processing, storage, and protection in edge computing systems.	1,2,3,4,7,8	1 to 5
CO4	Evaluate blockchain concepts, architectures, and platforms and integrate blockchain with edge computing to assess decentralized and secure computing solutions.	1,2,3,4,7,8	1 to 5
CO5	***Evaluate edge computing-based solutions by analyzing real-world use cases and healthcare/industrial IoT case studies, integrating Agile–DevOps lifecycle practices along with security, ethics, and blockchain-based trust management for reliable and scalable edge analytics systems***	1,2,3,4,5,6,7,8	1 to 5
K1= Remember, K2= Understand, K3= Apply, K4=Analyse, K5= Evaluate, K6= Create			

Continuous Internal Assessment - Components			
Assessment Type	**Nature**	**Maximum Marks**	**Weightage**
Mid-Semester Examination	Pen and Paper test for 90 minutes	50	10
Model Examination	Written examination for 180 minutes	100	10
Assignment	Any topic related to the course	10	5
Workbook/recordbook/ notebook	Handwritten Class Notes	5	5
Class activity 1	Quiz with MCQ,	10	5
Class activity 2	Seminar / Presentation	10	5
Class activity 3	Surprise Class Tests	10	5
Attendance	Marks will be given based on the attendance percentage of respective course		5
Total			50

PROGRAMME: MCA	**BATCH:2026-28**
PART: A	**COURSE COMPONENT: ELECTIVE - III**
COURSE NAME: Parallel Computing	**COURSE CODE: 26PMCA3E9**
SEMESTER: IV	**MARKS:100**
CREDITS: 3	**TOTAL HOURS: 60**
THEORY	

COURSE OBJECTIVE:

To analyze, apply, and evaluate parallel and distributed computing concepts, architectures, programming models, communication mechanisms, synchronization techniques, and distributed system services for designing efficient, scalable, secure, and fault-tolerant computing solutions

COURSE OUTCOMES:

1. Analyze the fundamental concepts, goals, and challenges of parallel computing and distinguish between sequential, concurrent, and parallel execution models for computational problem solving.
2. Evaluate parallel computer architectures, memory organization techniques, and processor classification schemes to determine their suitability for high-performance computing applications.
3. Apply parallel algorithm design principles, decomposition strategies, synchronization mechanisms, and performance metrics to develop efficient parallel solutions for computational problems.
4. Analyze distributed system architectures, communication models, middleware services, and process management mechanisms to understand the operation of distributed computing environments.
5. Evaluate distributed object-based systems with respect to naming, synchronization, fault tolerance, security, and real-world distributed applications to ensure reliable and scalable system design.

UNIT I **12 HOURS**

Introduction to Parallel Computing: The Idea of Parallelism, Power and potential of parallelism, examining sequential and parallel programs, Scope and issues of parallel and distributed computing, Goals of parallelism, Parallelism and concurrency using multiple instruction streams.

UNIT II **12 HOURS**

Parallel Architecture: Pipeline architecture, Array processor, Multi-processor architecture, Systolic architecture, Dataflow architecture, Architectural classification schemes, Memory access classification, Memory Issues: Shared vs. distributed, Symmetric multiprocessing (SMP), SIMD, Vector processing, GPU co-processing, Flynn's Taxonomy, Instruction Level support for parallel programming, Multiprocessor caches and Cache Coherence, Non-Uniform Memory Access (NUMA).

UNIT III **12 HOURS**

Parallel Algorithm Design Principles and Programming: Need for communication and coordination/synchronization, Scheduling and contention, Independence and partitioning, Task- Based Decomposition, Data Parallel Decomposition, Characteristics of task and interaction, Load balancing, Data Management, parallel algorithm models, Sources of overhead in parallel programs, Performance metrics for parallel algorithm implementations, Parallel algorithmic patterns like divide and conquer, Map and Reduce, Specific algorithms like parallel Merge Sort, Parallel graph Algorithms

UNIT IV **12 HOURS**

Architectures Of Distributed Systems - Architectural Styles – System Architectures -Architectures Versus Middleware - Self-Management In Distributed Systems - Processes - Threads - Virtualization - Clients -Servers - Communication - Remote Procedure Call - Message-Oriented Communication - Stream-Oriented Communication - Multicast Communication

UNIT V **12 HOURS**

Distributed Object Based Systems - Architecture - Processes - Communication - Naming- Synchronization - Fault Tolerance - Security - Distributed System Examples - File Systems And Web Based Systems

RECOMMENDED TEXTBOOKS:

1. Ananth Grama, Anshul Gupta, and George Karypis, Vipin Kumar; Introduction to Parallel Computing; Addition Wesley; 2nd Edition; 2003.
2. A.S. Tanenbaum; Distributed Operating Systems; Create Space Independent Publishing Platform; 3rd edition;2017

REFERENCE BOOKS:

1. Introduction To Parallel Programming, Steven Brawer, Academic Press
2. Introduction To Parallel Processing, M. Sasikumar, Dinesh Shikhare and P. Ravi Prakash,PHI
3. Randy Chow, T. Johnson, Distributed Operating Systems and Algorithms, Addison Wesley
4. Ian Foster: Designing and Building Parallel Programs – Concepts and tools for Parallel Software Engineering, Pearson Publisher, 1st Edition, 2019.
5. Parallel Programming in C with MPI and OpenMP Michael J. Quinn, McGrawHill Higher Education.

E-LEARNING RESOURCES:

1.https://www.youtube.com/watch?v=qbQCQ0U6H0o&list=PLbMVogVj5nJQRvzENlvMKA9q70ScSRZBQ

GUIDELINES TO THE QUESTION PAPER SETTERS
QUESTION PAPER PATTERN

SECTION	QUESTION COMPONENT	NUMBERS	MARKS	TOTAL
A	*Answer all the questions*	*1-10*	*2*	*20*
B	*Answer any 5 out of 7 questions*	*11-17*	*7*	*35*
C	*Answer any 3 out of 5 questions(each in 1200 words)*	*18-22*	*15*	*45*
				100

BREAK UP OF QUESTIONS

UNITS	SECTION A	SECTION B	SECTION C
I	2	2	1
II	2	1	1
III	2	2	1
IV	2	1	1
V	2	1	1
TOTAL	10	7	5
	SECTION A - 10	**SECTION B - 7**	**SECTION C - 5**

PO – CO MAPPING

Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1.	3	3	2	2	1	1	1	2
CO2.	3	3	3	3	1	1	1	2
CO3	3	3	3	3	2	2	1	2
CO4	3	3	3	3	2	1	2	2
CO5	3	3	3	3	2	1	3	3
Average Contribution	3	3	2.8	2.8	1.6	1.2	1.6	2.2

PO – CO QUESTION PAPER MAPPING

CO No:	COURSE OUTCOMES	POs ADDRESSED	COGNITIVE LEVEL (K1 to K6)
CO1	**Analyze** the fundamental concepts, goals, and challenges of parallel computing and distinguish between sequential, concurrent, and parallel execution models for computational problem solving.	1,2	1 to 5
CO2	**Evaluate** parallel computer architectures, memory organization techniques, and processor classification schemes to determine their suitability for high-performance computing applications.	1,2,3,4	1 to 5
CO3	**Apply** parallel algorithm design principles, decomposition strategies, synchronization mechanisms, and performance metrics to develop efficient parallel solutions for computational problems.	1,2,3,4	1 to 5
CO4	**Analyze** distributed system architectures, communication models, middleware services, and process management mechanisms to understand the operation of distributed computing environments.	1,2,3,4	1 to 5
CO5	**Evaluate** distributed object-based systems with respect to naming, synchronization, fault tolerance, security, and real-world distributed applications to ensure reliable and scalable system design.	1,2,3,4,7,8	1 to 5
K1= Remember, K2= Understand, K3= Apply, K4=Analyse, K5= Evaluate, K6= Create			

Continuous Internal Assessment - Components			
Assessment Type	**Nature**	**Maximum Marks**	**Weightage**
Mid-Semester Examination	Pen and Paper test for 90 minutes	50	10
Model Examination	Written examination for 180 minutes	100	10
Assignment	Any topic related to the course	10	5
Workbook/recordbook/ notebook	Handwritten Class Notes	5	5
Class activity 1	Quiz with MCQ,	10	5
Class activity 2	Seminar / Presentation	10	5
Class activity 3	Surprise Class Tests	10	5
Attendance	Marks will be given based on the attendance percentage of respective course		5
Total			50

PROGRAMME: MCA	**BATCH:2026-28**
PART: A	**COURSE COMPONENT: ELECTIVE - III**
COURSE NAME: Distributed Computing	**COURSE CODE: 26PMCA3E10**
SEMESTER: IV	**MARKS:100**
CREDITS: 3	**TOTAL HOURS: 60**
THEORY	

COURSE OBJECTIVE:

To analyze, apply, and evaluate the principles of distributed systems, synchronization and communication algorithms, mutual exclusion and recovery mechanisms, consensus protocols, and cloud computing technologies for designing reliable, scalable, and fault-tolerant distributed computing environments

COURSE OUTCOMES:

1. Analyze the fundamental concepts, architectures, communication models, execution paradigms, and design challenges of distributed systems to explain the behavior of distributed computations.
2. Apply logical clock mechanisms, message ordering techniques, group communication protocols, and snapshot algorithms to maintain consistency and coordination in distributed systems.
3. Analyze and evaluate distributed mutual exclusion and deadlock detection algorithms for ensuring resource sharing and fault-free operation in distributed environments.
4. Evaluate consensus, agreement, checkpointing, and rollback recovery algorithms to ensure reliability, fault tolerance, and recovery in distributed systems.
5. Apply cloud computing concepts, virtualization techniques, scalability mechanisms, and cloud service models to design and manage cloud-based distributed computing solutions.

UNIT I **12 HOURS**

Introduction: Definition-Relation to Computer System Components – Motivation – Message -Passing Systems versus Shared Memory Systems – Primitives for Distributed Communication – Synchronous versus Asynchronous Executions – Design Issues and Challenges; A Model of Distributed Computations: A Distributed Program – A Model of Distributed Executions – Models of Communication Networks – Global State of a Distributed System.

UNIT II **12 HOURS**

Logical Time: Physical Clock Synchronization: NTP – A Framework for a System of Logical Clocks – Scalar Time – Vector Time; Message Ordering and Group Communication: Message Ordering Paradigms – Asynchronous Execution with Synchronous Communication – Synchronous Program Order on Asynchronous System – Group Communication – Causal Order – Total Order; Global State and Snapshot Recording Algorithms: Introduction – System Model and Definitions – Snapshot Algorithms for FIFO Channels

UNIT III **12 HOURS**

Distributed Mutual exclusion Algorithms: Introduction – Preliminaries – Lamport's algorithm – Ricart Agrawala's Algorithm — Token-Based Algorithms – Suzuki-Kasami's Broadcast Algorithm; Deadlock Detection in Distributed Systems: Introduction – System Model – Preliminaries – Models of Deadlocks – Chandy-Misra-Haas Algorithm for the AND model and OR Model.

UNIT IV **12 HOURS**

Consensus and Agreement Algorithms: Problem Definition – Overview of Results – Agreement in a Failure-Free System(Synchronous and Asynchronous) – Agreement in Synchronous Systems with Failures; Checkpointing and Rollback Recovery: Introduction – Background and Definitions – Issues in Failure Recovery – Checkpoint-based Recovery – Coordinated Checkpointing Algorithm – – Algorithm for Asynchronous Checkpointing and Recovery

UNIT V **12 HOURS**

Definition of Cloud Computing – Characteristics of Cloud – Cloud Deployment Models – Cloud Service Models – Driving Factors and Challenges of Cloud – Virtualization – Load Balancing – Scalability and Elasticity – Replication – Monitoring – Cloud Services and Platforms: Compute Services – Storage Services – Application Services

RECOMMENDED TEXTBOOKS:

1. Kshemkalyani Ajay D, Mukesh Singhal, "Distributed Computing: Principles, Algorithms and Systems",CambridgePress,2011.

2. Mukesh Singhal, Niranjan G Shivaratri, "Advanced Concepts in Operating systems", McGraw Hill Publishers, 1994.

REFERENCE BOOKS:

1. George Coulouris, Jean Dollimore, Time Kindberg, "Distributed Systems Concepts and Design", Fifth Edition, Pearson Education, 2012.

2. Pradeep L Sinha, "Distributed Operating Systems: Concepts and Design", Prentice Hall of India, 2007.

3. Tanenbaum A S, Van Steen M, "Distributed Systems: Principles and Paradigms", Pearson Education, 2007.

4. Liu M L, "Distributed Computing: Principles and Applications", Pearson Education, 2004.

5. Nancy A Lynch, "Distributed Algorithms", Morgan Kaufman Publishers, 2003.

6. Arshdeep Bagga, Vijay Madisetti, " Cloud Computing: A Hands-On Approach", Universities Press, 2014.

GUIDELINES TO THE QUESTION PAPER SETTERS
QUESTION PAPER PATTERN

SECTION	QUESTION COMPONENT	NUMBERS	MARKS	TOTAL
A	*Answer all the questions*	*1-10*	*2*	*20*
B	*Answer any 5 out of 7 questions*	*11-17*	*7*	*35*
C	*Answer any 3 out of 5 questions(each in 1200 words)*	*18-22*	*15*	*45*
				100

BREAK UP OF QUESTIONS

UNITS	SECTION A	SECTION B	SECTION C
I	2	2	1
II	2	1	1
III	2	2	1
IV	2	1	1
V	2	1	1
TOTAL	10	7	5
	SECTION A - 10	**SECTION B - 7**	**SECTION C - 5**

PO – CO MAPPING

Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1.	3	3	3	2	1	1	1	2
CO2.	3	3	3	3	2	1	1	2
CO3	3	3	3	3	1	1	2	2
CO4	3	3	3	3	1	2	2	3
CO5	3	3	3	3	2	2	2	3
Average Contribution	3	3	3	2.8	1.4	1.4	1.6	2.4

PO – CO QUESTION PAPER MAPPING

CO No:	COURSE OUTCOMES	POs ADDRESSED	COGNITIVE LEVEL (K1 to K6)
CO1	**Analyze** the fundamental concepts, architectures, communication models, execution paradigms, and design challenges of distributed systems to explain the behavior of distributed computations.	1,2,3	1 to 5
CO2	**Apply** logical clock mechanisms, message ordering techniques, group communication protocols, and snapshot algorithms to maintain consistency and coordination in distributed systems.	1,2,3,4	1 to 5
CO3	**Analyze** and **evaluate** distributed mutual exclusion and deadlock detection algorithms for ensuring resource sharing and fault-free operation in distributed environments.	1,2,3,4	1 to 5
CO4	**Evaluate** consensus, agreement, checkpointing, and rollback recovery algorithms to ensure reliability, fault tolerance, and recovery in distributed systems.	1,2,3,4,8	1 to 5
CO5	**Apply** cloud computing concepts, virtualization techniques, scalability mechanisms, and cloud service models to design and manage cloud-based distributed computing solutions.	1,2,3,4,8	1 to 5
K1= Remember, K2= Understand, K3= Apply, K4=Analyse, K5= Evaluate, K6= Create			

Continuous Internal Assessment - Components			
Assessment Type	**Nature**	**Maximum Marks**	**Weightage**
Mid-Semester Examination	Pen and Paper test for 90 minutes	50	10
Model Examination	Written examination for 180 minutes	100	10
Assignment	Any topic related to the course	10	5
Workbook/recordbook/ notebook	Handwritten Class Notes	5	5
Class activity 1	Quiz with MCQ,	10	5
Class activity 2	Seminar / Presentation	10	5
Class activity 3	Surprise Class Tests	10	5
Attendance	Marks will be given based on the attendance percentage of respective course		5
Total			50

PROGRAMME: MCA	**BATCH:2026-28**
PART: A	**COURSE COMPONENT: PROJECT**
COURSE NAME: Project and Viva Voce	**COURSE CODE: 26PMCA318**
SEMESTER: IV	**MARKS:100**
CREDITS: 11	**TOTAL HOURS:**
VIVA VOCE	

COURSE OBJECTIVE:

To enable students to apply, analyze, and synthesize academic concepts to real-world problem situations, develop innovative and research-based solutions, and effectively communicate outcomes through structured documentation and presentations, thereby strengthening problem-solving, analytical, and lifelong learning competencies.

COURSE OUTCOMES:

1. Apply domain-specific theories, principles, and methodologies to design and execute meaningful projects.
2. Identify complex problems and develop innovative, practical, and multidisciplinary solutions.
3. Demonstrate the ability to conduct research, analyse data, and interpret findings to support informed decision-making.
4. Integrate ethical practices, sustainability, and social responsibility into project work, contributing positively to society.
5. Present and document project outcomes, while cultivating adaptability and continuous learning.

COURSE DESCRIPTION:

- The project work is an essential part of the postgraduate curriculum, designed to provide students with hands-on experience in their field of study. It encourages the application of theoretical knowledge to real-world problems, fostering research skills, critical thinking, and creativity. The project is typically undertaken in the final year of the PG programme.

- Students are expected to work on real-time projects. The project work is to be carried out either in a software industry or in an academic institution for the entire semester.

- Project work shall be carried out individually in an R&D section of any Industry or University or in the Institute in which the candidate is studying. The Project Work/Dissertation report shall be submitted through the guides/supervisors to the Head of the Department.

- The report should typically be between 40-60 pages, excluding references and appendices.

- The report should be neatly bound and presented.

PROGRAMME: MCA	**BATCH:2026-28**
PART: B	**COURSE COMPONENT: Ability Enhancement Skills - IV**
COURSE NAME: Architectural Design	**COURSE CODE: 26PSKL410**
SEMESTER: IV	**MARKS:100**
CREDITS: 2	**TOTAL HOURS: 30**
SKILL	

COURSE OBJECTIVE:

To enable learners to understand UML concepts and effectively apply UML modeling tools to design and develop software architectural representations such as use case, structural, behavioral, component, and deployment diagrams for real-world software systems

COURSE OUTCOMES:

1. Explain the fundamental concepts of UML and interpret various UML diagrams such as Use Case, Sequence, Activity, State Chart, Class, Component, and Deployment diagrams used in software architectural design.
2. Demonstrate the use of UML modeling tools for project management, diagram creation, and editing of modeling elements in software design environments.
3. Design Use Case, Sequence, and Class diagrams using UML tools to model system requirements and structural architecture effectively.
4. Develop Activity and State Chart diagrams using UML tools to represent dynamic behavior and workflow of software systems.
5. Construct Component and Deployment diagrams using UML tools to model system architecture and deployment structure in real-world applications.

UNIT I

Introduction to UML DIAGRAMS: Use Case Diagram – Sequence Diagram, Activity Diagram – State Chart Diagram – Class Diagram – Component Diagram – Deployment Diagram.

UNIT II

Introduction to UML TOOL - Managing Project - Editing Elements and Diagrams

UNIT III

Modelling with UML TOOL: Designing Use Case Diagram – Sequence Diagram - Class Diagram

UNIT IV

Modelling with UML TOOL: Designing Activity Diagram – State Chart Diagram

UNIT V

Modelling with UML TOOL: Designing Component Diagram – Deployment Diagram

RECOMMENDED TEXTBOOKS:

1. **UML for the IT Business Analyst 2nd Edition** by Howard Podeswa (Author)
2. https://buildmedia.readthedocs.org/media/pdf/staruml/latest/staruml.pdf
3. https://docs.staruml.io/

REFERENCE BOOKS:

1. **Software Architecture in Practice (SEI Series in Software Engineering) 3rd Edition,** by Len Bass (Author), Paul Clements (Author), Rick Kazman (Author)
2. **Head First Object-Oriented Analysis and Design 1st Edition** by Brett D.McLaughlin (Author), Gary Pollice (Author), Dave West (Author)

E-LEARNING RESOURCES:

1. https://www.tutorialspoint.com/software_architecture_design/architecture_models.htm
2. https://www.tutorialspoint.com/uml/index.htm