Advanced Data Structures and Applications			
Course Code	**1MCS102**	Scheme	2026
Type of Course	**PCC**	Semester	I
Teaching Hours/Week (L:T:P)	3 : 0 : 0	CIE Marks	50
Total Hours of Pedagogy L /T/P/SL&TW	42 : 0 : 0 : 48	SEE Marks	50
Credits	**3**	Total Marks	100
Examination type (SEE)	Theory	Exam Hours	3

Course outcomes (Skill Set)

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

CO1: Explain heap and priority queue structures, and hashing techniques.
CO2: Construct balanced and multiway search trees for efficient data storage.
CO3: Apply tries and augmented structures for fast data retrieval.
CO4: Apply advanced data structures to solve real-world data problems.

Module-1

Fundamental Heap Structures and Hashing:

Heap Structures: Heaps, Maintaining the Heap Property, Building a Heap, Priority Queues.

Hashing Fundamentals: Direct Address Tables, Hash Tables, Hash Functions

Text Book1: 6.1, 6.2, 6.3, 6.5, 11.1, 11.2, 11.3 Number of Hours:9

Module-2

Advanced Hashing and Priority Structures:

Advanced Hashing: Open Addressing, Perfect Hashing.

Dynamic Hashing: Dynamic Hashing, Bloom Filters.

Advanced Priority Structures: Pairing Heap, Symmetric Min-Max Heaps, Interval Heaps.

Text Book1: 11.4, 11.5

Text Book2: 8.3, 8.4, 9.5, 9.6, 9.7 Number of Hours:8

Module-3

Balanced Trees and Mergeable Heap Structures:

Red-Black Trees: Properties of Red-Black Trees, Rotations, Insertion, Deletion.

Mergeable Heap Structures: Leftist Trees, Binomial Heaps, Fibonacci Heaps.

Text Book1: 13.1, 13.2, 13.3, 13.4

Text Book2: 9.2, 9.3, 9.4 Number of Hours:9

Module-4

Search Trees and Trie Structures:

Adaptive and Multiway Search Trees: Splay Trees, m-Way Search Trees, B-Trees, B+ Trees.

Trie Structures: Digital Search Trees, Binary Tries and Patricia Trees, Multiway Tries.

Text Book2: 10.4, 11.1, 11.2, 11.3, 12.1, 12.2, 12.3 Number of Hours:8

Module-5

Augmented and Specialized Structures:

Augmented Data Structures: Dynamic Order Statistics, How to Augment a Data Structure, Interval Trees.

Applied Trie Structures: Suffix Trees, Tries in Internet Packet Forwarding.

Text Book1: 17.1, 17.2, 17.3
Text Book2: 12.4, 12.5 Number of Hours:8

Textbooks:

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, Introduction to Algorithms, 4th Edition, (2022), MIT Press.
2. Ellis Horowitz and Sartaj Sahni, *Fundamentals of Data Structures in C*, 2nd Edition (2023), Universities Press(India) Private limited.

Reference books / Manuals:

1. Mark Allen Weiss, Data Structures and Algorithm Analysis in C++, 4th Edition, (2014) Pearson Education.
2. Sartaj Sahni, Data Structures, Algorithms and Applications in C++, 2nd Edition, (2011), Universities Press.
3. Peter Brass, Advanced Data Structures, 1st Edition, (2011), Cambridge University Press.

Web links and Video Lectures (e-Resources):

- NPTEL – Advanced Data Structures: https://nptel.ac.in/courses/106102064
- MIT OpenCourseWare – Advanced Data Structures: https://ocw.mit.edu/courses/6-851-advanced-data-structures-spring-2012/
- Visualgo – Interactive Data Structure Visualizations: https://visualgo.net/en
- GeeksforGeeks – Advanced Data Structures Tutorials: https://www.geeksforgeeks.org/dsa/dsa-tutorial-learn-data-structures-and-algorithms/

Assessment Structure (both CIE and SEE)

The weightage assigned to Continuous Internal Evaluation (CIE) and Semester End Examination (SEE) shall be **50% each**. A student shall secure a minimum of 50% of the maximum marks prescribed for CIE and a minimum of **40%** of the maximum marks prescribed for SEE to be eligible for passing.

A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to a subject/course only if he/she secures not less than **50% of the aggregate marks** obtained in CIE and SEE taken together.

Continuous Internal Evaluation:

Continuous Comprehensive Assessments (CCA)+ Internal Assessment Test (IA) = Continuous Internal Evaluation CIE [25+25=50 marks]

A minimum of **two Internal Assessment Tests (IA)** shall be conducted. Each test shall be conducted for a maximum of **25 marks**. The average of the two test marks shall be the marks secured by a student in IA.

In addition, **Continuous Comprehensive Assessment (CCA)** shall be conducted for a total of **25 marks**. It is recommended to include **a maximum of two learning activities** as part of the CCA to foster the **holistic development of students**.

	Data Science and Management			
	Course Code	**1MCS103**	Scheme	2026
	Type of Course	**PCC**	Semester	1
	Teaching Hours/Week (L:T:P)	3:0:0	CIE Marks	50
	Total Hours of Pedagogy L /T/P/SL&TW	42: 0: 0: 48	SEE Marks	50
	Credits	**3**	Total Marks	100
	Examination type (SEE)	Theory	Exam Hours	3

Course outcomes (Skill Set)

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

CO1: Explain the principles of data science, analytics, artificial intelligence, data governance, and their role in data-driven decision making.

CO2: Apply statistical analysis, visualization, predictive analytics, AI techniques, and business analytics tools to solve real-world data-driven problems.

CO3: Analyze business problems using predictive, prescriptive, text, web, and social media analytics to support intelligent decision making.

CO4: Evaluate analytical solutions considering business objectives, ethical issues, privacy, security, responsible AI, and governance principles.

CO5: Design and develop an end-to-end analytics solution for an industrial application by integrating data science methodologies, AI techniques, governance practices, and business strategy.

Module-1

Foundations of Data Science and Analytics: Data-Analytic Thinking, Data Science Engineering Data-Driven Decision Making, Business Problems and Data Science Solutions, Data Mining Process, Business Intelligence and Analytics, Analytics Ecosystem, Applications of Analytics.

Textbook1: Chapters 1.1 to 1.4 and 2.1 to 2.4
Textbook2: Chapter 1.1 to 1.5 Number of Hours: 8

Module-2

Predictive Analytics and Decision Support: Nature of Data, Data Sources, Statistical Analysis, Descriptive Analytics, Data Visualization for Decision Making, Dashboard Concepts, Predictive Modeling, Model Evaluation and Cost-Benefit Analysis.

Textbook 1: Chapters 3.1 to 3.3 and 4.1 to 4.3
Textbook2: Chapter 2.1 to 2.5 Number of Hours:9

Module-3

Artificial Intelligence and Intelligent Decision Systems: Artificial Intelligence Foundations, AI and Analytics, Intelligent Systems, Text Analytics, Web Analytics, Social Media Analytics, Prescriptive Analytics, Decision Automation and Optimization.

Textbook2 : Chapter 4.1 to 4.3 ,5.1 to 5.3 ,6.1 to 6.4 Number of Hours:8

Module-4

	Data Governance, Ethics and Responsible AI: Big Data Concepts, Big Data Ecosystem, Big Data Applications, Privacy Challenges, Security Concerns, Ethical Issues in Analytics, Responsible AI, Future Challenges in AI Governance. **Textbook2:** Chapter 7.1 to 7.3 ,8.1 to 8.5 Number of Hours: 8
	Module-5
	Data Science Strategy and Industrial Applications: Decision Analysis, Analytic Engineering, Deployment of Analytics, Data Science and Business Strategy, Competitive Advantage through Analytics, Data Science Teams, Organizational Adoption of Analytics, Industrial Case Studies in Healthcare, Manufacturing, Finance, and Transportation. **Textbook1:** Chapters 7.1 to 7.3 ,13.1 to 13.4 **Textbook2:** Chapters 1.5 ,8.6 to 8.7 Number of Hours: 9

Suggested Learning Resources: (Text Book/ Reference Book/ Manuals):

Text books:

1. Foster Provost and Tom Fawcett, Data Science for Business: What You Need to Know About Data Mining and Data-Analytic Thinking, ,2nd Edition, O'Reilly Media, USA, 2023.
2. Ramesh Sharda, Dursun Delen and Efraim Turban Business Analytics, Data Science and AI: A Managerial Approach, ,6th Edition, Pearson Education, 2026.

Reference books / Manuals:

1. Thomas H. Davenport and Jeanne G. Harris, Competing on Analytics: The New Science of Winning, Updated Edition, Harvard Business Review Press, 2017.
2. Thomas H. Davenport, Jeanne G. Harris and Robert Morison, Analytics at Work: Smarter Decisions, Better Results, Harvard Business Review Press.
3. Doug Rose, Artificial Intelligence for Business, FT Press (Pearson), 2018.
4. Bernard Marr, Data Strategy: How to Profit from a World of Big Data, Analytics and Artificial Intelligence, 2nd Edition, Kogan Page, 2022.
5. S. Christian Albright and Wayne L. Winston, Business Analytics: Data Analysis and Decision Making, 8th Edition, Cengage Learning, 2024.

Web links and Video Lectures (e-Resources):

- Google Data Analytics: https://grow.google/intl/en_in/data-analytics-course/
- Elements of AI: https://course.elementsofai.com/?utm_source=chatgpt.com
- Data Science: https://skillsbuild.org/students/course-catalog/data-science?utm_source=chatgpt.com
- Data Science: https://www.datadays.org/?utm_source=chatgpt.com

Assessment Structure (both CIE and SEE)

The weightage assigned to Continuous Internal Evaluation (CIE) and Semester End Examination (SEE) shall be **50% each**. A student shall secure a minimum of 50% of the maximum marks prescribed for CIE and a minimum of **40%** of the maximum marks prescribed for SEE to be eligible for passing.

A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to a subject/course only if he/she secures not less than **50% of the aggregate marks** obtained in CIE and SEE taken together.

Continuous Internal Evaluation:

Continuous Comprehensive Assessments (CCA)+ Internal Assessment Test (IA) = Continuous Internal Evaluation CIE [25+25=50 marks]

A minimum of **two Internal Assessment Tests (IA)** shall be conducted. Each test shall be conducted for a maximum of **25 marks**. The average of the two test marks shall be the marks secured by a student in IA.

In addition, **Continuous Comprehensive Assessment (CCA)** shall be conducted for a total of **25 marks**. It is recommended to include **a maximum of two learning activities** as part of the CCA to foster the **holistic development of students**.

	Advanced Database Management Systems			
	Course Code	**1MCS104A**	Scheme	2026
	Type of course	**IPCC**	Semester	01
	Teaching Hours/Week (L:T:P)	3:0:2	CIE Marks	50
	Total Hours of Pedagogy L:T:P:SL&TW	42: 0 : 28 : 50	SEE Marks	50
	Credits	**4**	Total Marks	100
	Type of Examination (IPCC):	Theory- CIE +SEE, and Lab- CIE only.	Exam Hours	03

Course outcomes (Skill Set)

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

CO1: Design advanced relational and object-relational databases with strict integrity constraints.

CO2: Design and optimize high-performance data architectures by utilizing advanced file Structures, hashing, RAID systems, and distributed database techniques.

CO3: Implement scalable Big Data solutions using diverse NoSQL architectures, Hadoop Ecosystem, MapReduce frameworks, and distributed storage systems.

CO4: Build enhanced data models and information retrieval strategies to optimize advanced web search systems.

CO5: Design data warehouses and apply data mining algorithms to extract actionable business Intelligence patterns.

Module-1

Review of Relational Data Model and Relational Database Constraints: Relational model concepts; Relational model constraints and relational database schemas; Update operations, dealing with constraint violations.

Object and Object-Relational Databases: Overview of Object Database Concepts, Object Database Extensions to SQL, The ODMG Object Model and the Object Definition Language ODL, Object Database Conceptual Design, The Object Query Language OQL, Overview of the C++ Language Binding in the ODMG Standard.

Textbook 1: Chapter 5 ,12 Number of Hours: 8

Module-2

Disk Storage, Basic File Structures, Hashing, and Modern Storage Architectures: Introduction, Secondary Storage Devices, Buffering of Blocks, Placing File Records on Disk Operations on Files, Files of Unordered Records (Heap Files), Files of Ordered Records (Sorted Files), Hashing Techniques, Other Primary File Organizations, Parallelizing Disk Access Using RAID Technology, Modern Storage Architectures.

Distributed Database Concepts: Distributed Database Concepts, Data Fragmentation, Replication, and Allocation Techniques for Distributed Database Design, Overview of Concurrency Control and Recovery in Distributed Databases, Query Processing and Optimization in Distributed Databases, Types of Distributed Database Systems, Distributed Database Architectures, Distributed Catalogue Management.

	Textbook 1: Chapter 16 ,23 Number of Hours: 9
	Module-3
	NOSQL Databases and Big Data Storage Systems: Introduction to NOSQL Systems, The CAP Theorem, Document-Based NOSQL Systems and MongoDB, NOSQL Key-Value Stores, Column-Based or Wide Column NOSQL Systems, NOSQL Graph Databases and Neo4j. **Big Data Technologies Based on MapReduce and Hadoop:** What Is Big Data? Introduction to MapReduce and Hadoop, Hadoop Distributed File System (HDFS), MapReduce: Additional Details Hadoop v2 alias YARN, General Discussion. **Textbook 1:** Chapter 24,25 Number of Hours:9
	Module-4
	Enhanced Data Models: Introduction to Active, Temporal, Spatial, Multimedia, and Deductive Databases: Active Database Concepts and Triggers, Temporal Database Concepts, Spatial Database Concepts, Multimedia Database Concepts, Introduction to Deductive Databases. **Introduction to Information Retrieval and Web Search**: Information Retrieval (IR) Concepts, Retrieval Models, Types of Queries in IR Systems, Text pre-processing, Inverted Indexing, Evaluation Measures of Search relevance, web Search and Analysis. Trends in Information Retrieval. **Textbook 1:** Chapter 26 ,27 Number of Hours: 8
	Module-5
	Data Mining Concepts: Overview of Data Mining Technology, Association Rules, Classification, Clustering, Approaches to Other Data Mining Problems, Applications of Data Mining, Commercial Data Mining Tools **Overview of Data Warehousing and OLAP:** Introduction, Definitions, and Terminology, Characteristics of Data Warehouses, Data Modelling for Data Warehouses, building a Data Warehouse, Typical Functionality of a Data Warehouse, Data Warehouse versus Views, Difficulties of Implementing Data Warehouses. **Textbook 1:** Chapter 28 ,29 Number of Hours: 8
	PRACTICAL COMPONENTS OF IPCC
	List of experiments
	SQL QUERIES (Based on DDL statement, Constraints, DML statement, SELECT statement and Views) Note: In all schemas, create the table with necessary constraints (PK, FK, Not Null, Unique and Check Constraints) on SQL prompt and then solve the given queries. 1. **SQL Practical List**: **CUST** (Custno, Custname, Addln1, Addln2, City, State, Phone) **ITEM** (Itemno, Itemname, Itemprice, Qty_On_Hand) **INVOICE** (Invno, Invdate, Custno)

INV_ITEM (Invno, Itemno, Qty_Used)

Create the above four tables along with key constraints. Write an Insert script for insertion of rows with substitution variables and insert Appropriate data.

1. Add a column – "colour" to the Item table.
2. Display the column Item name and Price in sentence form using concatenation.
3. Find the total value of each item (item price * qty).
4. Display the list of customers belonging to "Gujarat" state.
5. Find the customers from "Lalbaug" city of Ahmedabad and Baroda.
6. Find the total, average, highest and lowest unit price of an item.

2. **HOSTEL** (Hno, Hname, Haddress, Total_Capacity, Warden_Nm)

   ROOM (Hno, Rno, Rtype, Location, No_Of_Students, Status)

   CHARGES (Hno, Rtype, Charges)

   STUDENT (Sid, Sname, Saddr, Faculty, Dept, Class, Hno, Rno)

   FEES (Sid, Fdate, Famount)

   The STATUS field tells us whether the room is occupied or vacant. The charges represent the term fees to be paid half yearly. A student can pay either the annual fees at one time or the half yearly fees twice a year.

   Create the above five tables along with key constraints. Write an Insert script for insertion of rows with substitution variables and insert appropriate data.

   1. Display the total number of rooms that are presently vacant.
   2. Display the number of students who are staying in 'double' seated room for each hostel.
   3. Display the warden's name and hostel address of students of 'Computer Science' Department.
   4. Count the total number of 'medical' students who live in the hostel.
   5. Display details about students who have paid the fees in the month of November 2011.

3. **PUBLISHER** (publ_id, publ_name, contact_person, contact_addr contact phone)

   CATEGORY (cat_id, cat_details, max_books, duration)

   BOOK_MASTER (book_id, bname, isbn_no, total_copies, publ_id)

   MEMBER (member_id, mname, cat_id, mem_ship_dt)

   ISSUE (ISSUE_id, member_id, book_id, issu_ret, issue_ret_dt)

   In the above tables, duration is in years and it stores the membership duration for that category.

1. Change the table design of ISSUE table to add a constraint, which will allow only 'I' or 'R' to be entered in the ISSUE_RETcolumn, which stores the action whether the book is being issued or returned.
2. Add a column to the MEMBER table, which will allow us to store the address of the member.
3. Create a table LIBRARY_USERS which has a structure similar to that of the MEMBER table but with no records.
4. Give details about members who have issued books, which contain the word 'DATA' somewhere in their titles.
5. Display the details of books whose all copies are issued.

4. **APPLICANT** (aid, aname, addr, abirth_dt)

   ENTRANCE_TEST (etid, etname, max_score, cut_score)

   ETEST_CENTRE (etcid, location, incharge, capacity)

   ETEST_DETAILS (aid, etid, etcid, etest_dt, score)

   This database is for a common entrance test which can be conducted at a number of centers and can be taken by an applicant on any day except holidays.

   1. Modify the APPLICANT table so that every applicant id has an 'A' before its value. For example, if the applicant id '1123', it should now become 'A1123'.
   2. Display the test centre details where no tests will be conducted.
   3. Display the details about applicants who have the same score as that of "Jaydev" in "Oracle Fundamentals".
   4. Display the details of applicants who have appeared for all the tests.
   5. Display those tests where no applicant has failed.
   6. Display details of entrance test centres which had full attendance between 1st November and 15th November 2011.

5. Consider the relational schema and write down the relational algebra for the following

   STUDENTS(Student_ID, Student_Name, Class, Major)

   COURSE(CourseNo, CourseName, Cr_Hr, Dept)

   SECTION(Sect_ID, CourseNo, Semester, Year, Instructor_Name)

   GRADEREPORT(Student_ID, Sect_ID, Grade)

   PREREQUISITE(CourseNo, Prerequisite_Course)

   1. Create a database named "labworkquestion1" and create the tables mentioned above.
   2. Insert 5 rows into each table.
   3. Display the name and major of all students whose grade is "A".
   4. Delete all tuples in the section relation whose instructor is Mr. Bean.
   5. Find the number of students in each class.

Textbooks:

1. Fundamentals of Database Systems, Ramez Elmasri and Shamkant B. Navathe, 7th Edition, Pearson Education, 2016.

Reference books / Manuals:

1. Database System Concepts Author: Abraham Silberschatz, Henry F. Korth, and S. Sudarshan Publisher: McGraw-Hill Year of Publication: March 2022 (7th Edition) ISBN: 9780078022159

2. Advanced Database Management System Author: Dr. Sanjeev Sharma Publisher: Wiley India Year of Publication: 2024 ISBN: 9789351199380.

3. Database Management Systems, Raghu Ramakrishnan and Johannes Gehrke, McGraw-Hill, 3rd edition, 2013.

Web links and Video Lectures (e-Resources):

- https://www.alessandroferrini.com/posts/relational-data-modeling/
- https://www.geeksforgeeks.org/dbms/definition-and-overview-of-odbms/
- https://www.mongodb.com/resources/basics/databases/what-is-an-object-oriented-database
- https://www.mongodb.com/resources/basics/databases/distributed-database
- https://www.geeksforgeeks.org/introduction-to-nosql/
- https://www.geeksforgeeks.org/dbms/rdms-vs-hadoop/
- https://www.datacamp.com/tutorial/dbms-data-models-explained
- https://learn.microsoft.com/en-us/azure/hdinsight/hadoop/apache-hadoop-introduction
- https://www.coveo.com/blog/top-information-retrieval-techniques-and-algorithms/
- https://peliqan.io/blog/data-warehousing-and-data-mining/
- https://www.itransition.com/business-intelligence/data-warehousing/olap

Assessment Structure (both CIE and SEE)

The weightage assigned to Continuous Internal Evaluation (CIE) and Semester End Examination (SEE) shall be 50% each. A student shall secure a minimum of 50% of the maximum marks prescribed for CIE and a minimum of 40% of the maximum marks prescribed for SEE to be eligible for a pass in the course.

A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to a subject /course only if he/she secures not less than 50% of the aggregate marks obtained in CIE and SEE taken together.

Continuous Internal Evaluation:

(i)Theory component

The CIE marks of 25 shall be earmarked for two tests.

The first test shall be conducted after completing two modules of the syllabus and the second one after completing the rest three modules.

Each test shall be conducted for 25 marks. The average of the two tests scaled down to 25 marks shall constitute the test marks for a maximum of 25 marks.

(ii) Laboratory component

Out of 25 marks, 15 marks shall be assigned for assessment of regular laboratory classwork as per the rubrics listed in Table - ICs/IPCCs1.

The remaining 10 marks shall be based on the practical test conducted by two Internal examiners appointed by the HoD. The allotment of marks shall be as per the rubrics defined for the practical test as indicated in the Table – ICs/IPCCs 1.

To qualify and become eligible to appear for the SEE of the IC/IPCC, a student shall secure at least 13 marks out of 25 in the CIE theory component, and at least 12 marks in the CIE Practical component.

Table – ICs/IPCCs 1 Distribution of marks corresponding to laboratory component

Component	Description	Marks
Allotment of marks for regular classwork		
Ability in Conducting Experiments	Initiative, skill, safety practices, teamwork, and independent handling of equipment during regular lab sessions.	5 marks
Laboratory Record /Journal	Regular and neat maintenance of lab record, accuracy of results, and timely submission.	10 marks
	Subtotal	**15**
Allotment of marks for CIE practical test		
Laboratory Test / Experiment Performance	Ability to set up apparatus, follow procedure, take observations, and obtain correct results in the test experiment.	5 marks
Viva-Voce	Understanding of theory, concepts, and experimental procedure; ability to explain results and answer related questions.	5 marks
	Subtotal	**10**
	Total	**25**

Advances in Computer Networks			
Course Code	**1MCS104B**	Scheme	2026
Type of course	**IPCC**	Semester	1
Teaching Hours/Week (L:T:P)	3: 0: 2	CIE Marks	50
Total Hours of Pedagogy L:T:P:SL&TW	42: 0: 28 :50	SEE Marks	50
Credits	**04**	Total Marks	100
Type of Examination (IPCC):	Theory- CIE +SEE, and Lab- CIE only.	Exam Hours	03

Course outcomes (Course Skill Set)

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

CO1: Apply link-layer protocols, error detection and correction techniques, multiple access protocols, and data centre networking architectures to ensure reliable and efficient data communication.

CO2: Select wireless and mobile networking technologies for modern communication systems.

CO3: Apply multimedia networking protocols to design quality of service aware real-time multimedia communication systems.

CO4: Utilize network virtualization technologies on multiple operating systems enforcing security enhancements.

CO5: Apply Voice and Video over IP protocols to ensure Quality of Service (QoS) based traffic management.

Module-1

The Link Layer: Links, Access Neworks and LANs: Introduction to the Link Layer, The Services Provided by the Link Layer, Where Is the Link Layer Implemented? **Error-Detection and -Correction Techniques:** Parity Checks, Checksumming Methods, Cyclic Redundancy Check (CRC), **Multiple Access Links and Protocols:** Channel Partitioning Protocols, Random Access Protocols**,** Taking-Turns Protocols, DOCSIS: The Link-Layer Protocol for Cable Internet Access, **Data Center Networking, Retrospective: A Day in the Life of a Web Page Request:** Getting Started: DHCP, UDP, IP, and Ethernet, Still Getting Started: DNS and ARP, Still Getting Started: Intra-Domain Routing to the DNS Server, Web Client-Server Interaction: TCP and HTTP.

Text book 1: Chapter 5-5.1, 5.2, 5.3, 5.6, 5.7 Number of Hours:09

Module-2

Wireless and Mobile Networks: Introduction, Wireless Links and Network Characteristics, CDMA, **Cellular Internet Access:** An Overview of Cellular Network Architecture, 3G Cellular Data Networks: Extending the Internet to Cellular Subscribers, On to 4G: LTE, **Mobility Management: Principles:** Addressing, Routing to a Mobile Node, **Mobile IP: Managing Mobility in Cellular Networks:** Routing Calls to a Mobile User, Handoffs in GSM

Text book 1: Chapter 6-6.1, 6.2, 6.4, 6.5, 6.6, 6.7 Number of Hours:09

Module-3
Multimedia Networking: Multimedia Networking Applications, Properties of Video, Properties of Audio, Types of Multimedia Network Applications, **Streaming Stored Video:** UDP Streaming, HTTP Streaming, Adaptive Streaming and DASH, Content Distribution Networks, **Protocols for Real-Time Conversational Applications:** RTP, SIP **Text book 1:** Chapter7-7.1,7.2(Excluding 7.2.5) Number of Hours:08
Module-4
Network Virtualization: VPNs, NATs, And Overlays: Introduction, Virtualization, Virtual Private Networks (VPNs), VPN Tunnelling And IP-in-IP Encapsulation, VPN Addressing And Forwarding, Extending VPN Technology To Individual Hosts, Using A VPN With Private IP Addresses, Network Address Translation (NAT), NAT Translation Table Creation, Variant Of NAT, An Example Of NAT Translation, Interaction Between NAT And ICMP, Interaction Between NAT And Applications, NAT In The Presence Of Fragmentation, Conceptual Address Domains, Linux, Windows, And Mac Versions Of NAT, Overlay Networks, Multiple Simultaneous Overlays. **Text book 2:** Chapter 19-19.1 to 19.18 Number of Hours:08
Module-5
Voice and Video Over IP: Introduction, Digitizing And Encoding, Audio And Video Transmission And Reproduction, Jitter And Playback Delay, Real-time Transport Protocol (RTP), Streams, Mixing, And Multicasting, RTP Encapsulation, RTP Control Protocol (RTCP), RTCP Operation, IP Telephony and Signalling, Quality of Service Controversy, QoS, Utilization, and Capacity, Emergency Services And Pre-emption, IntServ and Resource Reservation, DiffServ and Per-Hop Behaviour, Traffic Scheduling, Traffic Policing And Shaping. **Text book 2:** Chapter 26-26.1 to 26.17 Number of Hours:08
PRACTICAL COMPONENTS OF IPCC
List of experiments
1. Implement Parity Check, Checksum, and Cyclic Redundancy Check (CRC) algorithms for error detection and compare their effectiveness. (**Software Tool:** C/Python) 2. Capture and analyse Ethernet, ARP, IP, TCP, UDP, DNS, and HTTP packets using Wireshark. (**Software Tool:** Wireshark) 3. Simulate Pure ALOHA, Slotted ALOHA, CSMA/CD, and CSMA/CA protocols and compare throughput and collision performance. (**Software Tool:** NS-3 / OMNeT++) 4. Configure an IEEE 802.11 Wireless LAN and evaluate throughput, delay, and packet loss under different traffic loads. (**Software Tool:** Cisco Packet Tracer / NS-3**)**

5. Configure and analyze Bluetooth and ZigBee communication for short-range wireless networking. (**Software Tool:** Cisco Packet Tracer / NS-3**)**

6. Implement Mobile IP concepts and analyze mobility management and handoff performance. (**Software Tool:** NS-3**)**

7. Configure VPN and NAT in a virtual network and study packet forwarding and address translation. (**Software Tool:** Cisco Packet Tracer / GNS3 / Linux**)**

8. Configure audio/video streaming applications and compare UDP streaming with HTTP Adaptive Streaming (DASH). (**Software Tool:** VLC / NS-3**)**

9. Simulate RTP, RTCP, SIP, and QoS mechanisms and measure jitter, latency, throughput, and packet loss. (**Software Tool:** NS-3 / OMNeT++)

10. Implement Quality of Service (QoS) using DiffServ and evaluate its impact on latency, jitter, throughput, and packet loss. (**Software Tool**: Cisco Packet Tracer, GNS3)

Textbooks:

1. James F. Kurose and Keith W. Ross, *Computer Networking: A Top-Down Approach*, 6th Edition, Pearson, 2021.
2. Douglas E. Comer, *Internetworking with TCP/IP, Volume I: Principles, Protocols, and Architecture*, 6th Edition, Pearson Education, 2014.

Reference books / Manuals:

1. Andrew S. Tanenbaum, Nick Feamster, and David J. Wetherall, *Computer Networks*, 6th Edition, Pearson Education, 2021.
2. Network Warrior – by Gary A. Donahue, Practical guide for real-world network administration**.**

Web links and Video Lectures (e-Resources):

- https://nptel.ac.in/courses/106106243
- https://onlinecourses.nptel.ac.in/e-learning/preview/noc26_cs35
- https://www-net.cs.umass.edu/kurose_ross/index.php
- https://www.wireshark.org/docs/wsug_html_chunked/

Assessment Structure (both CIE and SEE)

The weightage assigned to Continuous Internal Evaluation (CIE) and Semester End Examination (SEE) shall be 50% each. A student shall secure a minimum of 50% of the maximum marks prescribed for CIE and a minimum of 40% of the maximum marks prescribed for SEE to be eligible for a pass in the course.

A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to a subject /course only if he/she secures not less than 50% of the aggregate marks obtained in CIE and SEE taken together.

Continuous Internal Evaluation:

(i)Theory component

The CIE marks of 25 shall be earmarked for two tests.

The first test shall be conducted after completing two modules of the syllabus and the second one after completing the rest three modules.

Each test shall be conducted for 25 marks. The average of the two tests scaled down to 25 marks shall constitute the test marks for a maximum of 25 marks.

(ii) Laboratory component

Out of 25 marks, 15 marks shall be assigned for assessment of regular laboratory classwork as per the rubrics listed in Table - ICs/IPCCs1.

The remaining 10 marks shall be based on the practical test conducted by two Internal examiners appointed by the HoD. The allotment of marks shall be as per the rubrics defined for the practical test as indicated in the Table – ICs/IPCCs 1.

Table – ICs/IPCCs 1 Distribution of marks corresponding to laboratory component

Component	Description	Marks
Allotment of marks for regular classwork		
Ability in Conducting Experiments	Initiative, skill, safety practices, teamwork, and independent handling of equipment during regular lab sessions.	5 marks
Laboratory Record / Journal	Regular and neat maintenance of lab record, accuracy of results, and timely submission.	10 marks
	Subtotal	**15**
Allotment of marks for CIE practical test		
Laboratory Test / Experiment Performance	Ability to set up apparatus, follow procedure, take observations, and obtain correct results in the test experiment.	5 marks
Viva-Voce	Understanding of theory, concepts, and experimental procedure; ability to explain results and answer related questions.	5 marks
	Subtotal	**10**
	Total	**25**

*****

Advances in Operating System			
Course Code	**1MCS104C**	Scheme	2026
Type of course	**IPCC**	Semester	I
Teaching Hours/Week (L:T:P)	42: 0: 28	CIE Marks	50
Total Hours of Pedagogy L:T:P:SL&TW	42: 0: 28: 50	SEE Marks	50
Credits	**4**	Total Marks	100
Type of Examination (IPCC):	Theory- CIE +SEE, and Lab- CIE only.	Exam Hours	3

Course outcomes (Course Skill Set)

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

CO1: Apply the principles of real-time operating systems, scheduling algorithms, resource sharing mechanisms, and the different types of distributed systems.

CO2: Analyse and compare centralized, decentralized, and hybrid distributed system architectures, multithreading, virtualization, server clusters, and code migration techniques for distributed applications.

CO3: Apply communication protocols, synchronization algorithms, logical clocks, mutual exclusion techniques, and election algorithms to solve distributed computing problems.

CO4: Evaluate fault tolerance mechanisms, process resilience, reliable communication protocols, distributed commit protocols, and recovery techniques to design dependable distributed systems.

CO5: Build secure distributed applications by incorporating authentication, authorization, secure channels, security management, and distributed file system architectures.

Module-1

Introduction: Introduction to Operating Systems, Goals and Functions of an Operating System, Multiprocessor and Multicore Systems, Clustered Systems, Operating System Services, Resource Management, Computing Environments, Open-Source Operating Systems, Operating System Design Principles, user and Kernel Modes, System Calls.

Textbook 1: Chapter 1, Chapter 2.1, 2.2, 2.3 Number of Hours: 08

Module-2

Architectures: Architectural Styles, Centralized Architectures, Decentralized Architectures, Hybrid Architectures, Threads: Introduction to Threads, Threads in Distributed Systems, The Role of Virtualization in Distributed Systems, Architectures of Virtual Machines, Networked User Interfaces, Client-Side Software for Distribution Transparency. Servers: General Design Issues, Server Clusters, Managing Server Clusters, Approaches to Code Migration, Migration and Local Resources.

Textbook 2: Chapter 2.1,2.2 and Chapter 3 Number of Hours: 08

Module-3

Communication: Fundamentals, Remote Procedure Call, Message-Oriented Communication, Stream-Oriented Communication, Multicast Communication, Synchronization: Clock Synchronization, Logical Clocks, Mutual Exclusion, Global Positioning of Nodes, Election Algorithms.

Textbook2: Chapter 4 and Chapter 6	Number of Hours: 09

Module-4

Fault Tolerance: Introduction To Fault Tolerance, Process Resilience, Reliable Client-Server Communication, Reliable Group Communication, Distributed Commit, Recovery.

Textbook2: Chapter 8	Number of Hours: 08

Module-5

Security: Introduction To Security, Secure Channels, Access Control, Security Management, Distributed File Systems, Architecture, Processes, Communication, Synchronization, Consistency and Replication.

Textbook2: Chapter 9 and Chapter 11.1 to 11.3, 11.5	Number of Hours: 08

PRACTICAL COMPONENTS OF IPCC

List of experiments

1. Study and configure Linux Real-Time Scheduling Policies (SCHED_FIFO and SCHED_RR), Simulate Earliest Deadline First (EDF) Scheduling.
2. Implement the Liu & Layland CPU Utilization Test for periodic real-time tasks and determine whether a task set is schedulable under Rate Monotonic Scheduling (RMS). Implement Priority Inheritance Protocol using Semaphores.
3. Design and implement a GUI-based distributed chat application that enables multiple users to communicate simultaneously over a network using a client-server architecture.
4. Design and implement a simple TCP Client-Server application using socket programming, where the client sends a message to the server and the server responds with an acknowledgment.
5. Implement an RPC-based distributed application that enables a client to invoke procedures on a remote server transparently, without explicitly handling network communication.
6. Implement the Berkeley Clock Synchronization Algorithm for synchronizing clocks of distributed systems.
7. Simulate the Vector Clock Algorithm for maintaining the causal ordering of events in a distributed system.
8. Implement a multithreaded TCP server using POSIX threads (pthread) that can handle multiple client connections simultaneously.
9. Simulate a server cluster and implement Round Robin Load Balancing for

distributing client requests among multiple servers. Develop Transparent Client Interface for Distributed Application.

10. Develop a server-cluster simulation consisting of one load balancer and at least three backend servers. Implement the following load balancing algorithms for reduced response time.

Textbooks:

1. A. Silberschatz, Peter B Galvin, Greg Gagne - Operating System Concepts, Wiley India, 10th Edition, 2023 (ISBN: 978-9357460569)
2. Andrew Tanenbaum Maarten Van Steen - Distributed Systems Principles and Paradigms, 2nd Edition, Pearson, Thirteenth Impression 2025.

Reference books / Manuals:

1. Mukesh Singhal and Niranjan G. Shivaratri, *Advanced Concepts in Operating Systems,* McGraw -Hill, 1st Edition, 2014.
2. George Coulouris, Jean Dollimore, Tim Kindberg, and Gordon Blair, *Distributed Systems: Concepts and Design*, 5th Edition, Pearson, 12th Impression, 2026.
3. Andrew S. Tanenbaum and Herbert Bos, *Modern Operating Systems*, 5th Edition, Pearson, 2024.

Web links and Video Lectures (e-Resources):

1. https://www.youtube.com/playlist?list=PLWkTsO24LpD8ufLfZz2t3e4LSI05ueIOq **:** NPTEL videos for Operating System**.**
2. https://www.youtube.com/watch?v=NzZXz3fJf6o&list=PLShJJCRzJWxhz7SfG4hpaBD5bKOloWx9J **:** NPTEL Link for cloud computing overview.

Assessment Structure (both CIE and SEE)

The weightage assigned to Continuous Internal Evaluation (CIE) and Semester End Examination (SEE) shall be 50% each. A student shall secure a minimum of 50% of the maximum marks prescribed for CIE and a minimum of 40% of the maximum marks prescribed for SEE to be eligible for a pass in the course.

A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to a subject /course only if he/she secures not less than 50% of the aggregate marks obtained in CIE and SEE taken together.

Continuous Internal Evaluation:

(i)Theory component

The CIE marks of 25 shall be earmarked for two tests.

The first test shall be conducted after completing two modules of the syllabus and the second one after completing the rest three modules.

Each test shall be conducted for 25 marks. The average of the two tests scaled down to 25 marks shall constitute the test marks for a maximum of 25 marks.

(ii) Laboratory component

Out of 25 marks, 15 marks shall be assigned for assessment of regular laboratory classwork as per the rubrics listed in Table - ICs/IPCCs1.

The remaining 10 marks shall be based on the practical test conducted by two Internal examiners appointed by the HoD. The allotment of marks shall be as per the rubrics.

Table – ICs/IPCCs 1 Distribution of marks corresponding to laboratory component

Component	Description	Marks
Allotment of marks for regular classwork		
Ability in Conducting Experiments	Initiative, skill, safety practices, teamwork, and independent handling of equipment during regular lab sessions.	5 marks
Laboratory Record / Journal	Regular and neat maintenance of lab record, accuracy of results, and timely submission.	10 marks
	Subtotal	**15**
Allotment of marks for CIE practical test		
Laboratory Test / Experiment Performance	Ability to set up apparatus, follow procedure, take observations, and obtain correct results in the test experiment.	5 marks
Viva-Voce	Understanding of theory, concepts, and experimental procedure; ability to explain results and answer related questions.	5 marks
	Subtotal	**10**
	Total	**25**

Advanced Bigdata Analytics			
Course Code	**1MCS104D**	Scheme	2026
Type of course	**Integrated Professional Core Course-I (IPCC)**	Semester	I
Teaching Hours/Week (L:T:P)	3:0:2	CIE Marks	50
Total Hours of Pedagogy L:T:P:SL&TW	42: 0:28 :50	SEE Marks	50
Credits	**4**	Total Marks	100
Type of Examination (IPCC):	Theory- CIE +SEE, and Lab- CIE only.	Exam Hours	3

Course outcomes (Course Skill Set)

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

CO1: Describe the need of Big data and visualization in real world applications.
CO2: Apply big data analytics frameworks and visualization techniques for decision making
CO3: Analyse the use of big data analytics and visualization for business applications
CO4: Formulate optimized relational and scripting-based queries using Apache Hive and Pig to handle large-scale data ETL and reporting operations

Module-1

Introduction to Big Data Analytics: Definition, Need of Big Data and its characteristics, Classification of Data - Structured, semi structured, and unstructured, Scalability and Parallel Processing, Designing Data Architecture, Data Sources, Quality, Pre-Processing and Storing, Data Storage and Analysis, Big Data Analytics Applications and Case Studies.

Textbook1 – Chapter 1, Sections 1.1–1.7 Number of Hours: 09

Module-2

Introduction to Hadoop: Introduction, Hadoop and its Ecosystem, Hadoop Distributed File System, MapReduce Framework and Programming Model, Hadoop Yarn, Hadoop Ecosystem Tools, Hadoop Distributed File System Basics: HDFS Design Features, Components, HDFS User Commands.

Textbook1 – Chapter 2, Sections 2.1–2.6 Number of Hours: 09

Module-3

MapReduce, Hive and Pig: Introduction, MapReduce Map Tasks, Reduce Tasks and MapReduce Execution, Composing MapReduce for Calculations and Algorithms, Hive, HiveQL, Pig.

Textbook1 – Chapter 4, Sections 4.1–4.6 Number of Hours: 09

Module-4

Spark and Big Data Analytics: Introduction to Data Analysis with Spark-Spark SQL, Using Python with Spark SQL, Data Analysis Operations, Data ETL process, Introduction Analytics, Reporting and Visualization.

Textbook1 – Chapter 5 Number of Hours:09

Module-5
Machine Learning Algorithms for Big Data Analytics: Introduction, Estimating the relationships, Outliers, Variances, Probability Distributions, and Correlations, Regression analysis, Finding Similar Items, Similarity of Sets and Collaborative Filtering, Clustering Analysis. **Textbook1** – Chapter 6 Number of Hours:09
PRACTICAL COMPONENTS OF IPCC
List of experiments

1. Given a raw, noisy dataset containing mixed formats (structured, semi-structured JSON, and unstructured text logs), write a script to handle missing values, filter outliers, detect data duplication, and normalize data types to prepare a clean schema for storage.
2. Write a program to interact with the Hadoop Distributed File System (HDFS). The Program must programmatically create directories, configure replication factors, upload large datasets in blocks, and retrieve block locations and health status of the cluster nodes.
3. Design and implement a MapReduce program to process a large web-server access log file. The Mapper should parse the IP addresses and timestamps, and the Reducer should compute the hourly hit counts for each unique IP address.
4. Write a MapReduce program that executes a *Reduce-Side Join* or *Map-Side Join* on two separate datasets (e.g., a "Users" dataset and a "Transaction Logs" dataset) sharing a common key, simulating a relational database join operation over a distributed network.
5. Create an optimization pipeline in Apache Hive. Write a HiveQL script to create external tables for an e-commerce sales dataset. Implement dynamic partitioning and bucketing based on "Region" and "Year", and write analytical queries to extract top-performing product categories.
6. Write an Apache Pig script (.pig) to load, filter, and transform an unstructured or semi-structured dataset (like a Twitter/X user mention dataset). Use Pig Latin operators (FILTER, GROUP, FOREACH, GENERATE, and ORDER BY) to isolate and count the most trending keywords.
7. Execute interactive SQL queries on distributed Spark clusters. Create temporary and global views from Spark Data Frames, run window functions to find rolling averages

or rank records, and pass the final aggregate outputs to a Python visualization library (like Matplotlib or Seaborn) to generate analytics dashboards.

8. Design and implement an end-to-end ETL pipeline using Apache Spark and PySpark for a available dataset. Read data from CSV/JSON files, perform data cleaning and transformation operations, handle missing values and duplicates, apply aggregations and filtering, and store the processed data in an appropriate format. Create Spark SQL views and execute analytical queries to generate meaningful insights.
9. Implement a regression-based machine learning model using Apache Spark MLlib on available dataset. Perform data pre-processing and feature selection, divide the dataset into training and testing sets, build a Linear Regression model, evaluate its performance using suitable metrics such as MAE, RMSE and R^2, and interpret the results.
10. Implement a clustering and similarity analysis application using Apache Spark MLlib on available dataset. Apply K-Means clustering to group similar records, determine suitable clusters using an appropriate evaluation measure.

Text books:

1. Raj Kamal and Preeti Saxena, "Big Data Analytics Introduction to Hadoop, Spark, and Machine-Learning", McGraw Hill Education, 2018 ISBN: 9789353164966, 9353164966.
2. Douglas Eadline, "Hadoop 2 Quick-Start Guide: Learn the Essentials of Big Data Computing in the Apache Hadoop 2 Ecosystem", 1st Edition, Pearson Education, 2016. ISBN-13: 978- 9332570351

Reference books / Manuals:

1. Tom White, "Hadoop: The Definitive Guide", 4th Edition, O"Reilly Media, 2015.ISBN-13: 978- 9352130672.
2. Seema Acharya, Subhashini Chellappan, Big Data and analytics, Wiley Publications, 2015, ISBN-10: 8126554789, ISBN-13: 978-8126554782.

Web links and Video Lectures (e-Resources):

1. Essential for mastering HDFS commands, YARN configurations, and core MapReduce programming models: https://hadoop.apache.org/docs/stable/
2. Architecture scalability, Hadoop ecosystem tools, Pig, and Hive: https://nptel.ac.in/courses/106105186

Assessment Structure (both CIE and SEE)

The weightage assigned to Continuous Internal Evaluation (CIE) and Semester End Examination (SEE) shall be 50% each. A student shall secure a minimum of 50% of the maximum marks prescribed for CIE and a minimum of 40% of the maximum marks prescribed for SEE to be eligible for a pass in the course.

A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to a subject /course only if he/she secures not less than 50% of the aggregate marks obtained in CIE and SEE taken together.

Continuous Internal Evaluation:

(i)Theory component

The CIE marks of 25 shall be earmarked for two tests.
The first test shall be conducted after completing two modules of the syllabus and the second one after completing the rest three modules.
Each test shall be conducted for 25 marks. The average of the two tests scaled down to 25 marks shall constitute the test marks for a maximum of 25 marks.

(ii) Laboratory component

Out of 25 marks, 15 marks shall be assigned for assessment of regular laboratory classwork as per the rubrics listed in Table - ICs/IPCCs1.
The remaining 10 marks shall be based on the practical test conducted by two Internal examiners appointed by the HoD. The allotment of marks shall be as per the rubrics defined for the practical test as indicated in the Table – ICs/IPCCs 1.
To qualify and become eligible to appear for the SEE of the IC/IPCC, a student shall secure at least 13 marks out of 25 in the CIE theory component, and at least 12 marks in the CIE Practical component.

Table – ICs/IPCCs 1 Distribution of marks corresponding to laboratory component		
Component	**Description**	**Marks**
Allotment of marks for regular classwork		
Ability in Conducting Experiments	Initiative, skill, safety practices, teamwork, and independent handling of equipment during regular lab sessions.	5 marks
Laboratory Record / Journal	Regular and neat maintenance of lab record, accuracy of results, and timely submission.	10 marks
	Subtotal	**15**
Allotment of marks for CIE practical test		
Laboratory Test / Experiment Performance	Ability to set up apparatus, follow procedure, take observations, and obtain correct results in the test experiment.	5 marks
Viva-Voce	Understanding of theory, concepts, and experimental procedure; ability to explain results and answer related questions.	5 marks
	Subtotal	**10**
	Total	**25**

*****

	Artificial Intelligence for IoT			
	Course Code	**1MCS105A**	Scheme	2026
	Type of Course	**PEC**	Semester	I
	Teaching Hours/Week (L:T:P)	3:0:0	CIE Marks	50
	Total Hours of Pedagogy L /T/P/SL&TW	42:0:0:48	SEE Marks	50
	Credits	**03**	Total Marks	100
	Examination type (SEE)	Theory	Exam Hours	03

Course Outcomes(Skill set)

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

CO1: Explain the fundamental concepts of Artificial Intelligence, Internet of Things (IoT), and their integration for developing intelligent IoT systems.

CO2: Apply machine learning, deep learning, and reinforcement learning techniques for processing and analyzing IoT-generated data.

CO3: Develop AI-enabled IoT applications using cloud platforms and intelligent data analytics techniques.

CO4: Analyze AI techniques for processing visual, text, audio, speech, and time-series data generated by IoT devices.

CO5: Apply AIoT concepts to develop solutions for smart homes, healthcare, smart cities, and industrial automation.

Module-1

Foundations and Basic Integrations of IoT and AI

Principles and Foundations of IoT and AI: What is IoT 101? Big data and IoT, Infusion of AI data science in IoT, Tools used.

Machine Learning for IoT: Understanding ML and its Role in IoT, Learning paradigms, Prediction using linear regression, Logistic regression for classification, Classification using support vector machines, Naive Bayes, Decision trees, Ensemble learning.

Text book 1: Chapter 1, 3 Number of Hours: 09

Module-2

Advanced AI Techniques and Their Application in IoT

Deep Learning for IoT: Deep learning 101, Multilayered perceptron for regression and classification, Understanding CNNs, Recurrent neural networks.

Reinforcement Learning for IoT: Introduction to RL, Simulated environments, Introducing Gymnasium, Q-learning, Q-Network.

Text book 1: Chapter 4, 6 Number of Hours: 08

Module-3

Implementing Intelligent IOT Solutions in Diverse Domains-1

AI Cloud Platforms for IoT: Introduction to Cloud Computing for IoT, Cloud Computing Basics, AI Services in Cloud Platforms, Training and Deploying AI Models for IoT on Cloud Platforms

Deep Learning for Time-Series Data from IoT: Introduction to Time-Series Data, The Importance of Time-Series Analysis in IoT,

Text book 1: Chapter 9, 10 Number of Hours: 08

Module-4

Implementing Intelligent IOT Solutions in Diverse Domains-2

Leveraging AI for Visual Data from IoT: A Brief Introduction to Visual Data in IoT Ecosystems, Handling Challenges in Visual Data in IoT, Leveraging AI Techniques for Visual Data in IoT.

AI for Text, Audio and Speech Data from IoT: Introduction to Text, Audio, and Speech Data, Fundamentals of Text Data Processing, Working with Audio Data in IoT.

Text book 1: Chapter 11, 12 Number of Hours: 09

Module-5

Applying AI and IoT in Real-World Scenarios

AI for Personal and Home IoT: Wearable and Fitness Devices, Smartwatch Health Monitoring in Action, Smart Home Devices, AI-driven home automation and intelligence, Case study: An Integrated smart home Implementation

AI for Smart Cities IoT:

AI Applications in Smart Cities: Smart traffic management, Smart parking, Smart waste management, smart policing. AI-Powered Urban Analytics, Adapting IoT for Smart Cities and the Necessary Steps.

Text book 1: Chapter 13, 15 Number of Hours:08

Textbooks:

1. Dr. Amita Kapoor, *Hands-On Artificial Intelligence for IoT*, 2nd Edition, May 2025, Packt Publishing Ltd., Birmingham, UK, ISBN: 978-1-83546-718-3. www.packtpub.com

Reference books :

1. Yuxi (Hayden) Liu, *Artificial Intelligence for IoT Cookbook*, 1st Edition, March 2021, Packt Publishing Ltd., Birmingham, UK, ISBN: 978-1-83898-198-5. www.packtpub.com
2. Rajkumar Buyya and Amir Vahid Dastjerdi (Editors), *Internet of Things: Principles and Paradigms*, 1st Edition, May 2016, Morgan Kaufmann (Elsevier), Cambridge, Massachusetts, USA, ISBN: 978-0-12-805395-9
3. Perry Lea, *Internet of Things for Architects: Architecting IoT Solutions by Implementing Sensors, Communication Infrastructure, Edge Computing, Analytics,*

and Security, 1st Edition, January 2018, Packt Publishing Ltd., Birmingham, UK, ISBN: 978-1-78847-059-9 (Paperback). www.packtpub.com

Web links and Video Lectures (e-Resources):

- NPTEL – Introduction to Internet of Things: https://nptel.ac.in
- NPTEL – Artificial Intelligence and Machine Learning: https://nptel.ac.in
- Google Cloud AI and IoT: https://cloud.google.com
- Coursera – AI and Machine Learning Courses: https://www.coursera.org
- edX – Artificial Intelligence and IoT Courses: https://www.edx.org
- Artificial Intelligence for IoT: https://www.udacity.com/course/aiot-foundations--ud074?utm_source=chatgpt.com

Assessment Structure (both CIE and SEE)

The weightage assigned to Continuous Internal Evaluation (CIE) and Semester End Examination (SEE) shall be **50% each**. A student shall secure a minimum of 50% of the maximum marks prescribed for CIE and a minimum of **40%** of the maximum marks prescribed for SEE to be eligible for passing.

A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to a subject/course only if he/she secures not less than **50% of the aggregate marks** obtained in CIE and SEE taken together.

Continuous Internal Evaluation:

Continuous Comprehensive Assessments (CCA)+ Internal Assessment Test (IA) = Continuous Internal Evaluation CIE [25+25=50 marks]

A minimum of **two Internal Assessment Tests (IA)** shall be conducted. Each test shall be conducted for a maximum of **25 marks**. The average of the two test marks shall be the marks secured by a student in IA.

In addition, **Continuous Comprehensive Assessment (CCA)** shall be conducted for a total of **25 marks**. It is recommended to include **a maximum of two learning activities** as part of the CCA to foster the **holistic development of students**.

	Artificial Intelligence for Cyber Security			
	Course Code	**1MCS105B**	Scheme	2026
	Type of Course	**PEC**	Semester	I
	Teaching Hours/Week (L:T:P)	3:0:0	CIE Marks	50
	Total Hours of Pedagogy L /T/P/SL&TW	42:0:0:48	SEE Marks	50
	Credits	**03**	Total Marks	100
	Examination type (SEE)	Theory	Exam Hours	03

Course outcomes (Skill Set)

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

CO1: Explain the fundamental concepts of AI and ML and their applications in addressing cybersecurity challenges.
CO2: Apply AI and ML techniques to detect, analyze, and mitigate cyber security threats and vulnerabilities.
CO3: Utilize Natural Language Processing (NLP) methods in cybersecurity applications and analyze strategies for defending against adversarial attacks on AI systems.
CO4: Evaluate the ethical, legal, and societal implications of deploying AI technologies in cybersecurity environments.

Module-1

Why Machine Learning and Security? : Cyber Threat Landscape, The Cyber Attacker's Economy , A Marketplace for Hacking Skills, Indirect Monetization, , What Machine Learning Is Not, Adversaries Using Machine Learning, Real-World Uses of Machine Learning in Security , Spam Fighting: An Iterative Approach and Limitations of Machine Learning in Security.

Classifying and Clustering: Machine Learning: Problems and Approaches, Machine Learning in Practice: A Worked Example Training Algorithms to Learn.

Textbook-1: Ch. 1, Ch. 2 (25-36) Number of Hours: 08

Module-2

Classifying and Clustering: Supervised Classification Algorithms, Practical Considerations in Classification Clustering.

Anomaly Detection : When to Use Anomaly Detection Versus Supervised Learning**,** Intrusion Detection with Heuristics, Data-Driven Methods, Feature Engineering for Anomaly Detection, Anomaly Detection with Data and Algorithms, Practical System Design Concerns , Malware Analysis.

Textbook-1: Ch. 2(40-77), Ch. 3 Number of Hours: 09

Module-3

Network Traffic Analysis: Theory of Network Defense**,** Machine Learning and Network Security, Building a Predictive Model to Classify Network Attacks , Natural Language

	Processing in Cybersecurity. **Protecting the Consumer Web:** Types of Abuse and the Data That Can Stop Them, Authentication and Account Takeover, Supervised Learning for Abuse Problems , Clustering Abuse, Further Directions in Clustering. **Textbook-1:** Ch. 5, Ch. 6 Number of Hours: 09
	Module-4
	Production Systems: Defining Machine Learning System Maturity and Scalability, Data Quality**,** Model Quality**,** Performance**,** Maintainability**,** Monitoring and Alerting **and** Security and Reliability. **Textbook-1:** Ch. 7 Number of Hours: 08
	Module-5
	Adversarial Machine Learning: The Importance of Adversarial ML, Security Vulnerabilities in Machine Learning Algorithms, Attack Transferability, Attack Technique: Model Poisoning, Attack Technique: Evasion Attack. **Textbook-1:** Ch. 8 Number of Hours: 08

Textbooks:

1. Clarence Chio and David Freeman,Machine Learning and Security: Protecting Systems with Data and Algorithms, 2018, O'REILLY.

Reference books / Manuals:

1. David Kim, Michael G. Solomon, —Fundamentals of Information Systems Security, Jones & Bartlett Learning Publishers, 2013.
2. Nina Godbole, Sunit Belapure, —Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives||, Wiley Publishers, 2011.
3. "AI in Cybersecurity: Transforming Threat Detection and Response" by Arun Kumar Krishna, 2024.;
4. Artificial Intelligence in Cyber Security" by Leslie F. Sikos, Springer, 2019.

Web links and Video Lectures (e-Resources):

- Google Machine Learning Crash Course:
  https://developers.google.com/machine-learning/crash-course
  Comprehensive introduction to machine learning concepts, algorithms, and practical applications.
- NPTEL – Introduction to Machine Learning:

https://nptel.ac.in/courses/106106139
Covers supervised learning, classification, clustering, and model evaluation.

- MIT OpenCourseWare – Introduction to Machine Learning: https://ocw.mit.edu/courses/6-390-introduction-to-machine-learning-fall-2024/
  Lecture notes, assignments, and video lectures on machine learning fundamentals.
- Stanford CS224N – Natural Language Processing with Deep Learning: https://web.stanford.edu/class/cs224n/
  Useful for the NLP in Cybersecurity component of the course.
- Coursera – Machine Learning for Cyber Threat Detection: https://www.coursera.org/learn/machine-learning-cyber-threat-and-anomaly-detection-mqcsa
  Focuses on anomaly detection, malware analysis, and cybersecurity applications of machine learning

Assessment Structure (both CIE and SEE)

The weightage assigned to Continuous Internal Evaluation (CIE) and Semester End Examination (SEE) shall be **50% each**. A student shall secure a minimum of 50% of the maximum marks prescribed for CIE and a minimum of **40%** of the maximum marks prescribed for SEE to be eligible for passing.

A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to a subject/course only if he/she secures not less than **50% of the aggregate marks** obtained in CIE and SEE taken together.

Continuous Internal Evaluation:

Continuous Comprehensive Assessments (CCA)+ Internal Assessment Test (IA) = Continuous Internal Evaluation CIE [25+25=50 marks]

A minimum of **two Internal Assessment Tests (IA)** shall be conducted. Each test shall be conducted for a maximum of **25 marks**. The average of the two test marks shall be the marks secured by a student in IA.

In addition, **Continuous Comprehensive Assessment (CCA)** shall be conducted for a total of **25 marks**. It is recommended to include **a maximum of two learning activities** as part of the CCA to foster the **holistic development of students**.

Advanced Computer Vision			
Course Code	**1MCS105C**	Scheme	2026
Type of Course	**PEC**	Semester	I
Teaching Hours/Week (L:T:P)	3: 0: 0	CIE Marks	50
Total Hours of Pedagogy L /T/P/SL&TW	42: 0: 0: 48	SEE Marks	50
Credits	**3**	Total Marks	100
Examination type (SEE)	Theory	Exam Hours	3

Course Outcomes:

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

CO1: Explain the basic concepts of computer vision, image formation, and digital cameras.
CO2: Apply image processing, filtering, and optimization techniques to analyze images.
CO3: Use deep learning models such as neural networks and CNNs for image recognition tasks.
CO4: Apply camera geometry and calibration techniques for 3D vision applications.
CO5: Analyze transformation models and multi-view vision techniques for computer vision problems.

Module-1

Introduction: What is computer vision? **Image Formation:** Geometric primitives and transformation, Photometric image formation, Digital Camera.

Textbook-1: Ch. 1.1,2.1,2.2,2.3 Number of Hours: 08

Module-2

Image Processing: Point operators, Linear filtering, Neighborhood operators, Fourier Transform. **Model Fitting and Optimization:** Scattered Data Interpolation, Variational Methods and Regularization.

Textbook-1: Ch. 3.1,3.2,3.3,3.4,4.1,4.2 Number of Hours: 08

Module-3

Deep Learning: Supervised Learning, Unsupervised Learning, Deep Neural Networks, Convolution Neural Networks (CNNs), Training Deep Networks. **Recognition:** Instance Recognition, Image Classification, Object Detection, Semantic Segmentation.

Textbook-1: Ch. 5.1, 5.2, 5.3, 5.4, 6.1, 6.2, 6.3, 6.4 Number of Hours: 08

Module-4

Image Preprocessing and Feature Extraction, The Pinhole Camera, Three Geometric Problems, Homogeneous Coordinates, Learning Extrinsic Parameters, Learning Intrinsic Parameters, Inferring Three-Dimensional World Points, Applications of the Pinhole Camera.

Textbook 2: Ch. 13, 14 Number of Hours: 09

Module-5

Transformation Models and Multi-View Vision: Two-Dimensional Transformation Models, Learning in Transformation Models, Inference in Transformation Models, Multiple Cameras, Models for Shape, Models for Style and Identity, Temporal Models, Models for Visual Words.

Textbook-2: Ch. 15.1–15.3, 16, 17, 18, 19, 20 Number of Hours: 09

Text books:

1. Computer Vision: Algorithms and Applications – Richard Szeliski, 2nd Edition, Springer, 2022
2. Computer Vision: Models, Learning, and Inference – Simon J. D. Prince, 1st Edition, Cambridge University Press,2012

Reference books / Manuals:

1. Aston Zhang et al., Dive into Deep Learning, Latest Online Edition, 2023–2025
2. Kevin P. Murphy, Probabilistic Machine Learning: An Introduction, 2022.

Web links and Video Lectures (e-Resources):

- Computer Vision: Algorithms and Applications (2nd Edition, 2022) – Online: https://szeliski.org/Book/
- Dive into Deep Learning – https://d2l.ai/
- Neural Networks and Deep Learning – http://neuralnetworksanddeeplearning.com/
- Stanford CS231n Course – http://cs231n.stanford.edu/
- MIT OpenCourseWare – Computer Vision – https://ocw.mit.edu/
- Coursera Computer Vision (University of Michigan) – https://www.coursera.org/
- OpenCV Documentation – https://docs.opencv.org/
- PyTorch Tutorials – https://pytorch.org/tutorials/
- TensorFlow Tutorials – https://www.tensorflow.org/tutorials

Assessment Structure (both CIE and SEE)

The weightage assigned to Continuous Internal Evaluation (CIE) and Semester End Examination (SEE) shall be **50% each**. A student shall secure a minimum of 50% of the maximum marks prescribed for CIE and a minimum of **40%** of the maximum marks prescribed for SEE to be eligible for passing.

A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to a subject/course only if he/she secures not less than **50% of the aggregate marks** obtained in CIE and SEE taken together.

Continuous Internal Evaluation:

Continuous Comprehensive Assessments (CCA)+ Internal Assessment Test (IA) = Continuous Internal Evaluation CIE [25+25=50 marks]

A minimum of **two Internal Assessment Tests (IA)** shall be conducted. Each test shall be conducted for a maximum of **25 marks**. The average of the two test marks shall be the marks secured by a student in IA.

In addition, **Continuous Comprehensive Assessment (CCA)** shall be conducted for a total of **25 marks**. It is recommended to include **a maximum of two learning activities** as part of the CCA to foster the **holistic development of students**.

Data Analytics for Fraud Detection			
Course Code	**1MCS105D**	Scheme	2026
Type of Course	**PEC**	Semester	I
Teaching Hours/Week (L:T:P)	3: 0: 0	CIE Marks	50
Total Hours of Pedagogy L /T/P/SL&TW	42: 0: 0: 48	SEE Marks	50
Credits	**3**	Total Marks	100
Examination type (SEE)	Theory	Exam Hours	3

Course Outcomes:

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

CO1: Explain the concepts, types, and challenges of fraud detection using data analytics.

CO2: Apply data preprocessing, visualization, and analytical techniques for fraud analysis.

CO3: Develop fraud detection models using statistical and machine learning techniques.

CO4: Evaluate fraud detection solutions using predictive analytics, graph analytics, and real-world case studies.

Module-1

Introduction to Fraud Analytics: Fundamentals and characteristics of fraud, types of fraud, fraud detection and prevention, role of big data in fraud detection, data-driven fraud detection, fraud-detection techniques, fraud cycle, fraud analytics process model, role of fraud data scientists, data-analytic thinking, business problems and data science solutions, supervised and unsupervised analytical approaches, and overview of the data mining process.

Textbook-1: Chapter 1; **Textbook-2:** Chapters 1, 2 Number of Hours: 08

Module-2

Data Collection, Sampling and Preprocessing: Types of data sources, merging data sources, sampling, types of data elements, visual data exploration and exploratory statistical analysis, Benford's Law, descriptive statistics, treatment of missing values and outliers, identification of red flags, data standardization and categorization, weights of evidence coding, variable selection, Principal Component Analysis, RIDIT and PRIDIT analysis, and segmentation.

Textbook-1: Chapter 2; **Textbook-2:** Chapter 2 Number of Hours: 08

Module-3

Descriptive Analytics and Outlier Detection: Descriptive analytics for fraud detection, fundamentals of anomaly and outlier detection, graphical and statistical outlier detection procedures, probabilistic and statistical models for outlier detection, linear models, proximity-based outlier detection, clustering techniques for fraud detection, high-dimensional outlier detection, and one-class Support Vector Machines for identifying anomalous observations.

Textbook-1: Chapter 3; **Textbook-3:** Chapters 1–5 Number of Hours: 08

Module-4

Predictive Analytics for Fraud Detection: Fundamentals of predictive modelling, target definition, supervised learning, logistic regression, variable selection, decision trees, neural networks, Support Vector Machines, ensemble methods, multiclass classification, model generalization and overfitting, model validation, evaluation of classification models, handling skewed and imbalanced datasets, and performance evaluation of fraud detection models.

Textbook-1: Chapter 4; **Textbook-2:** Chapters 3, 4, 5, 7 Number of Hours: 09

Module-5

Social Network Analytics, Fraud Model Management and Governance: Social networks and their application to fraud detection, fraud as a social phenomenon, homophily, neighbourhood-based metrics, community mining for identifying groups of fraudsters, bipartite networks, fraud model lifecycle, model representation, selection of cases for investigation, fraud alert and case management, visual analytics, backtesting analytical fraud models, model design and documentation, data quality, privacy in fraud analytics, economic considerations, modelling extensions, Internet of Things in fraud analytics, and corporate fraud governance.

Textbook-1: Chapters 5, 6, 7; **Textbook-3:** Chapter 12 Number of Hours: 09

Text books:

1. Bart Baesens, Véronique Van Vlasselaer and Wouter Verbeke, *Fraud Analytics Using Descriptive, Predictive, and Social Network Techniques: A Guide to Data Science for Fraud Detection*, John Wiley & Sons, 1st Edition, 2015.
2. Foster Provost and Tom Fawcett, *Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking*, O'Reilly Media, 2013.ISBN 9781449374273.
3. Charu C. Aggarwal, *Outlier Analysis*, 2nd Edition, Springer, 2017

Reference books / Manuals:

1. Jiawei Han, Micheline Kamber, and Jian Pei, Data Mining: Concepts and Techniques, 4th Edition, Morgan Kaufmann, 2023.
2. Trevor Hastie, Robert Tibshirani, and Jerome Friedman, The Elements of Statistical Learning, 2nd Edition (Corrected), Springer.
3. Ian H. Witten, Eibe Frank, Mark A. Hall, and Christopher J. Pal, Data Mining: Practical Machine Learning Tools and Techniques, 4th Edition, Morgan Kaufmann, 2016.
4. Christopher M. Bishop and Hugh Bishop, Deep Learning: Foundations and Concepts, Springer, 2024.
5. Kevin Murphy, Probabilistic Machine Learning: An Introduction, MIT Press, 2022

Web links and Video Lectures (e-Resources):

- Kaggle Datasets (Fraud Detection) – Public datasets for credit card fraud, insurance fraud, transaction analytics, and machine learning projects.
- Scikit-learn Documentation – Documentation and tutorials for machine learning algorithms used in fraud detection.
- Scikit-learn MOOC (Inria) – Free hands-on course covering data preprocessing, classification, anomaly detection, and model evaluation.
- TensorFlow Tutorials – Official tutorials for deep learning models applicable to fraud analytics.
- FATF (Financial Action Task Force) – International standards, reports, and guidance on anti-money laundering (AML) and financial crime.
- WEKA Documentation – Machine learning toolkit useful for experimentation with fraud detection datasets.
- Python Data Science Handbook – Free online reference for NumPy, Pandas, Matplotlib, and Scikit-learn.
- Pandas Documentation – Official documentation for data preprocessing and analytics.
- NumPy Documentation – Numerical computing library used in data analytics.
- Matplotlib Documentation – Visualization techniques for exploratory fraud analysis.

Video Lectures

- NPTEL – Data Analytics and Machine Learning Courses
- MIT OpenCourseWare – Data Science and Machine Learning
- Stanford Online – Machine Learning Resources
- DeepLearning.AI Machine Learning Specialization (Coursera) – Covers supervised learning, anomaly detection, model evaluation, and practical ML workflows.
- TensorFlow YouTube Channel
- Google Developers – Machine Learning Crash Course
- DataTalks.Club YouTube Channel – Practical data analytics and machine learning tutorials.
- freeCodeCamp – Data Science and Machine Learning

Assessment Structure (both CIE and SEE)

The weightage assigned to Continuous Internal Evaluation (CIE) and Semester End Examination (SEE) shall be **50% each**. A student shall secure a minimum of 50% of the maximum marks prescribed for CIE and a minimum of **40%** of the maximum marks prescribed for SEE to be eligible for passing.

A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to a subject/course only if he/she secures not less than **50% of the aggregate marks** obtained in CIE and SEE taken together.

Continuous Internal Evaluation:

Continuous Comprehensive Assessments (CCA)+ Internal Assessment Test (IA) = Continuous Internal Evaluation CIE [25+25=50 marks]

A minimum of **two Internal Assessment Tests (IA)** shall be conducted. Each test shall be conducted for a maximum of **25 marks**. The average of the two test marks shall be the marks secured by a student in IA.

In addition, **Continuous Comprehensive Assessment (CCA)** shall be conducted for a total of **25 marks**. It is recommended to include **a maximum of two learning activities** as part of the CCA to foster the **holistic development of students**.

Software Engineering and Web Applications			
Course Code	**1MCS106A**	Scheme	2026
Type of Course	**PEC/Elective**	Semester	1
Teaching Hours/Week (L:T:P)	3: 0: 0	CIE Marks	50
Total Hours of Pedagogy L /T/P/SL&TW	42:0:0:48	SEE Marks	50
Credits	**3**	Total Marks	100
Examination type (SEE)	Theory	Exam Hours	3

Course outcomes (Skill Set)

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

CO1: Apply software architecture, component-based development, and service-oriented principles in designing large-scale software systems.
CO2: Analyse software reliability, security, quality, and dependability using appropriate engineering techniques and metrics.
CO3: Apply software evolution, configuration management, and process improvement practices to enhance software maintainability.
CO4: Apply web engineering principles to design effective web application architecture, navigation, and user interfaces.
CO5: Apply web engineering practices to develop, test, deploy, and maintain secure and high-quality web applications.

Module-1

Software Architecture and Reuse Engineering: Architectural Design – Architectural Design Decisions, Architectural Views, Architectural Patterns, Application Architectures , Component-Based Software Engineering – Components and Component Models, CBSE Process, Component Composition, Service-Oriented Software Engineering – Service-Oriented Architecture (SOA), RESTful Services, Service engineering, Service Composition.

Textbook 1: Chapters 6.1-6.4, 16.1-16.3,18.1-18.4 Number of Hours:8

Module-2

Software Dependability and Security: Dependable systems: Dependability Properties, Sociotechnical systems, Redundancy and diversity, Dependable processes, Reliability Engineering – Availability and Reliability, Reliability Requirements, Fault-Tolerant Architectures, Reliability Measurement, Security Engineering – Security and dependability Security Requirements, Secure Systems Design, Security Testing and Assurance.

Textbook 1: Chapters 10.1-10.4,11.1 -11.5,13.1-13.5 Number of Hours:9

Module-3

Software Evolution and Configuration Management: Software Evolution – Evolution Processes, Legacy Systems, Software Maintenance, Quality Management – Software Quality, Quality Assurance, Software Standards, Reviews and Inspections, Software Measurement. Configuration Management – Version Management, System Building, Change Management, Release Management.

Textbook 1: Chapters 9.1-9.3, 24.1-24.5,25.1-25.4 Number of Hours:9

Module-4
Web Application Analysis and Design: Introduction to Web Applications, Characteristics of Web Applications, Web Application Analysis, Content Design, Architectural Design for Web Applications, Navigation Design, Interface Design, User Experience Design, Responsive Web Design Principles, Modeling Web Applications. **Textbook 2:** Chapters 3,4,5,6,7 Number of Hours:8
Module-5
Web Application Development, Testing and Maintenance: Web Application Construction, Web Components, Web Services Integration, Testing Web Applications, Functional Testing, Performance Testing, Quality Assurance for Web Applications, Web Application Security, Deployment Strategies, Web Application Maintenance and Evolution. **Textbook 2:** Chapters 8,9,10,11,12 Number of Hours:8

Suggested Learning Resources: (Text Book/ Reference Book/ Manuals):

Text books:

1. Software Engineering, Ian Sommerville, Pearson Education, 10th Edition, 2016.
2. Web Engineering A Practitioner's Approach Roger S. Pressman and David Lowe, McGraw-Hill Higher Education, 1st Edition, 2008.

Reference books / Manuals:

1. Software Architecture in Practice Len Bass, Paul Clements, and Rick Kazman, Addison-Wesley Professional, 4th Edition, 2021.
2. Web Engineering Gerti Kappel, Birgit Pröll, Siegfried Reich, and Werner Retschitzegger, Wiley, 2006.
3. DevOps A Software Architect's Perspective Len Bass, Ingo Weber, and Liming Zhu, Addison-Wesley Professional, 2015.

Web links and Video Lectures (e-Resources):

- https://ocw.mit.edu/search/?q=software+engineering
- https://developer.mozilla.org/en-US/docs/Learn_web_development
- https://www.w3schools.com/
- https://www.freecodecamp.org/learn/
- https://www.youtube.com/@TraversyMedia
- https://www.youtube.com/@NetNinja

Assessment Structure:

Assessment Structure (both CIE and SEE)

The weightage assigned to Continuous Internal Evaluation (CIE) and Semester End Examination (SEE) shall be **50% each**. A student shall secure a minimum of 50% of the maximum marks prescribed for CIE and a minimum of **40%** of the maximum marks prescribed for SEE to be eligible for passing.

A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to a subject/course only if he/she secures not less than **50% of the aggregate marks** obtained in CIE and SEE taken together.

Continuous Internal Evaluation:

Continuous Comprehensive Assessments (CCA)+ Internal Assessment Test (IA) = Continuous Internal Evaluation CIE [25+25=50 marks]

A minimum of **two Internal Assessment Tests (IA)** shall be conducted. Each test shall be conducted for a maximum of **25 marks**. The average of the two test marks shall be the marks secured by a student in IA.

In addition, **Continuous Comprehensive Assessment (CCA)** shall be conducted for a total of **25 marks**. It is recommended to include **a maximum of two learning activities** as part of the CCA to foster the **holistic development of students**.

Advances in Blockchain Technology			
Course Code	**1MCS106B**	Scheme	2026
Type of Course	**PEC**	Semester	I
Teaching Hours/Week (L:T:P)	3:0:0	CIE Marks	50
Total Hours of Pedagogy L /T/P/SL&TW	42:0:0:48	SEE Marks	50
Credits	**03**	Total Marks	100
Examination type (SEE)	Theory	Exam Hours	03

Course Outcomes:

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

- CO1: Analyze blockchain architecture, decentralization principles, and cryptographic foundations.
- CO2: Evaluate advanced consensus protocols and blockchain scalability solutions.
- CO3: Implement smart contracts and decentralized applications using Ethereum.
- CO4: Compare enterprise blockchain frameworks and their application domains.
- CO5: Assess blockchain security, privacy-preserving mechanisms, and emerging decentralized technologies.

Module-1

Blockchain Fundamentals and Distributed Ledger Technology: Introduction to Blockchain, Distributed Systems, CAP Theorem and FLP Impossibility, Decentralization using Blockchain, Types of Blockchains, Consensus Mechanisms, Cryptographic Primitives, Hash Functions, Public Key Cryptography, Digital Signatures, Merkle Trees.

TextBook1: Chapter- 1, 2, 3 Number of Hours: 09

Module-2

Advanced Consensus and Scalability Mechanisms: Byzantine Generals Problem, Proof of Work (PoW), Proof of Stake (PoS), Delegated Proof of Stake (DPoS), Practical Byzantine Fault Tolerance (PBFT), Proof of Authority (PoA), Federated Consensus, Blockchain Scalability Challenges, Segregated Witness (SegWit), State Channels, Lightning Network, Sidechains.

TextBook1: Chapter- 5, 11 Number of Hours: 08

Module-3

Ethereum, Smart Contracts and Decentralized Applications: Ethereum Architecture, Ethereum Virtual Machine (EVM), Accounts and Transactions, Gas Mechanism, Smart Contracts, Solidity Fundamentals, Smart Contract Deployment, Decentralized Applications (DApps), Decentralized Autonomous Organizations (DAOs), Token Standards.

TextBook1: Chapter- 13, 14, 15 Number of Hours: 08

Module-4

Enterprise Blockchain Platforms: Enterprise Blockchain Concepts, Hyperledger Project, Hyperledger Fabric Architecture, Membership Service Provider, Channels and Chaincode, Hyperledger Sawtooth, Hyperledger Iroha, Corda Architecture, Quorum Platform, Enterprise Use Cases.

TextBook1: Chapter- 16, 17, 18 Number of Hours: 08

Module-5

Security, Privacy and Emerging Trends: Blockchain Security, 51% Attacks, Sybil Attacks, Eclipse Attacks, Privacy in Blockchain, Zero-Knowledge Proofs, Ring Signatures, Confidential Transactions, Decentralized Finance (DeFi), Non-Fungible Tokens (NFTs), Web3 Ecosystem, Blockchain and IoT, Blockchain and Artificial Intelligence.

TextBook1: Chapter- 4, 12, 19, 20, 21 Number of Hours: 09

Textbooks:

1. Imran Bashir, *Mastering Blockchain*, 4th Edition, Packt Publishing, 2023.

Reference books:

1. Andreas M. Antonopoulos and Gavin Wood, *Mastering Ethereum: Building Smart Contracts and DApps*, 1st Edition, O'Reilly Media, Inc., USA, 2018. ISBN: 978-1491971949.
2. Daniel Drescher, *Blockchain Basics: A Non-Technical Introduction in 25 Steps*, 2nd Edition, Apress, Berkeley, CA, USA, 2020. ISBN: 978-1484254127.
3. Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller and Steven Goldfeder, *Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction*, 1st Edition, Princeton University Press, Princeton, New Jersey, USA, 2016. ISBN: 978-0691171692.

Web links and Video Lectures (e-Resources):

- Ethereum: https://ethereum.org/en/developers/docs
- Hyperledger: https://hyperledger-fabric.readthedocs.io
- Blockchain Basics: https://www.ibm.com/topics/blockchain
- Blockchain Foundation Training: https://training.linuxfoundation.org/blockchain

Assessment Structure (both CIE and SEE)

The weightage assigned to Continuous Internal Evaluation (CIE) and Semester End Examination (SEE) shall be **50% each**. A student shall secure a minimum of 50% of the maximum marks prescribed for CIE and a minimum of **40%** of the maximum marks prescribed for SEE to be eligible for passing.

A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to a subject/course only if he/she secures not less than **50% of the aggregate marks** obtained in CIE and SEE taken together.

Continuous Internal Evaluation:

Continuous Comprehensive Assessments (CCA)+ Internal Assessment Test (IA) = Continuous Internal Evaluation CIE [25+25=50 marks]

A minimum of **two Internal Assessment Tests (IA)** shall be conducted. Each test shall be conducted for a maximum of **25 marks**. The average of the two test marks shall be the marks secured by a student in IA.

In addition, **Continuous Comprehensive Assessment (CCA)** shall be conducted for a total of **25 marks**. It is recommended to include **a maximum of two learning activities** as part of the CCA to foster the **holistic development of students**.

Mobile and Web Application Security			
Course Code	**1MCS106C**	Scheme	2026
Type of Course	**PEC**	Semester	1
Teaching Hours/Week (L:T:P)	3: 0: 0	CIE Marks	50
Total Hours of Pedagogy per semester L /T/P/SL&TW:	42: 0: 0: 48	SEE Marks	50
Credits	**03**	Total Marks	100
Examination type (SEE)	Theory	Exam Hours	03

Course outcome (Course Skill Set)

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

1. Describe principles of secure design and development for web and mobile applications.
2. Analyze vulnerabilities and attacks affecting modern application ecosystems.
3. Apply security mechanisms for authentication, authorization, data protection, and secure communications.
4. Perform security testing and vulnerability assessment of applications.
5. Investigate DevSecOps practices and emerging trends in application security.

Module-1

Secure Application Engineering and Threat Modeling

Introduction to application security; security goals and principles; attack surfaces and trust boundaries; threat modeling methodologies (STRIDE, DREAD); secure software development life cycle (SSDLC); risk assessment; security requirements engineering; OWASP Web Top-10 and Mobile Top-10; secure coding principles.

Textbook1: Chapter 1, Chapter 2, Chapter 3

Textbook2: Chapter 1 Number of Hours : 08

Module-2

Web Application Security

HTTP/HTTPS security; authentication and authorization mechanisms; session management; cookies and token security; Same-Origin Policy; Cross-Site Scripting (XSS); Cross-Site Request Forgery (CSRF); SQL Injection; file upload vulnerabilities; REST API security; Content Security Policy and security headers.

Textbook1: Chapter 4, Chapter 5, Chapter 6,Chapter 7, Chapter 10

Number of Hours : 08

Module - 3
Mobile Application Security Android application architecture; permission model; secure data storage; cryptographic APIs; secure inter-process communication; secure networking and TLS; code signing; reverse engineering; obfuscation techniques; malware and privacy threats; mobile application testing. **Textbook2:** Chapter 2, Chapter 3, Chapter 4, Chapter 5, Chapter 6 Number of Hours : 09
Module-4
Application reconnaissance and mapping; vulnerability discovery; authentication attacks; session attacks; access-control vulnerabilities; Static Application Security Testing (SAST); Dynamic Application Security Testing (DAST); vulnerability scanning; secure code review; penetration testing methodologies; remediation and reporting. **Textbook1:** Chapter 8, Chapter 9 Number of Hours : 09
Module-5
DevSecOps and Emerging Trends **DevSecOps principles**; CI/CD security; secure containerized deployments; microservices and API ecosystem security; cloud-native application security; privacy engineering; AI-assisted security tools; zero-trust principles; case studies of major web and mobile security breaches; current research challenges. **Textbook1**: Chapter 11, Chapter 12 Number of Hours : 08
Text books: 1. Andrew Hoffman, Web Application Security: Exploitation and Countermeasures for Modern Web Applications, 2nd Edition, O'Reilly Media (Indian Edition available through Shroff Publishers), 2024. 2. Pragati Ogal Rai, Shubham Asthana, Android Application Security Essentials, Packt Publishing, 2015. **Reference books / Manuals:** 1. Dafydd Stuttard and Marcus Pinto, The Web Application Hacker's Handbook, , Wiley, 2011. 2. Joshua J. Drake, Zach Lanier, Collin Mulliner, Pau Oliva Fora, Stephen A. Ridley, Georg Wicherski , Android Hacker's Handbook, Wiley, 2014. 3. OWASP Web Security Testing Guide and Mobile Application Security Testing Guide (latest editions, freely available online).

Web links and Video Lectures (e-Resources):

- OWASP Foundation, https://owasp.org , OWASP Top -10, Secure Coding, Testing Guides
- https://owasp.org/www-project-web-security-testing-guide/, Web Application Security Testing
- https://owasp.org/www-project-mobile-app-security/ : Mobile Security Standards
- https://www.youtube.com/@OWASPGlobal OWASP Foundation YouTube Channel
- NPTEL courses on cyber security and privacy

Assessment Structure (both CIE and SEE)

The weightage assigned to Continuous Internal Evaluation (CIE) and Semester End Examination (SEE) shall be **50% each**. A student shall secure a minimum of 50% of the maximum marks prescribed for CIE and a minimum of **40%** of the maximum marks prescribed for SEE to be eligible for passing.

A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to a subject/course only if he/she secures not less than **50% of the aggregate marks** obtained in CIE and SEE taken together.

Continuous Internal Evaluation:

Continuous Comprehensive Assessments (CCA)+ Internal Assessment Test (IA) = Continuous Internal Evaluation CIE [25+25=50 marks]

A minimum of **two Internal Assessment Tests (IA)** shall be conducted. Each test shall be conducted for a maximum of **25 marks**. The average of the two test marks shall be the marks secured by a student in IA.

In addition, **Continuous Comprehensive Assessment (CCA)** shall be conducted for a total of **25 marks**. It is recommended to include **a maximum of two learning activities** as part of the CCA to foster the **holistic development of students**.

Data Engineering			
Course Code	**1MCS106D**	Scheme	2026
Type of Course	**PEC**	Semester	1
Teaching Hours/Week (L:T:P)	3:0:0	CIE Marks	50
Total Hours of Pedagogy L /T/P/SL&TW	42: 0: 0: 48	SEE Marks	50
Credits	**3**	Total Marks	100
Examination type (SEE)	Theory	Exam Hours	3

Course outcomes (Skill Set)

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

CO1: Explain the concepts of data engineering, data architecture, data engineering lifecycle, and the roles and responsibilities of data engineers in modern organizations.
CO2: Apply data engineering principles to select appropriate technologies, architectures, and lifecycle components for developing scalable, reliable, and maintainable data platforms.
CO3: Analyze data generation, storage, ingestion, transformation, and serving techniques to support efficient data processing and analytics requirements.
CO4: Evaluate data engineering solutions by considering data quality, scalability, security, privacy, governance, technology selection, and operational requirements in modern data ecosystems.
CO5: Evaluate modern data engineering tools, platforms, and emerging technologies to recommend appropriate solutions for diverse data engineering applications and organizational requirements.

Module-1

Foundations of Data Engineering and Data Architecture: Data Engineering Defined, The Data Engineering Lifecycle, Evolution of the Data Engineer, Data Engineering and Data Science, Data Engineering Skills and Activities, Data Maturity and the Data Engineer, Business Responsibilities, Technical Responsibilities, Data Engineers Inside an Organization, Internal-Facing versus External-Facing Data Engineers, Data Engineering Lifecycle Components, Designing Good Data Architecture, Principles of Good Data Architecture, Scalability, Security, Loosely Coupled Systems, Reversible Decisions, Architecture Leadership.

Textbook 1: Chapters 1.1–1.5 and 2.1–2.6 Number of Hours: 8

Module-2

Data Engineering Lifecycle and Technology Selection: The Data Engineering Lifecycle, Source Systems, Storage, Ingestion, Transformation, Serving Data, Security, Data Management, DataOps, Orchestration, Software Engineering, Choosing Technologies Across the Data Engineering Lifecycle, Evaluating Technology Choices, Build versus Buy Decisions, Cloud and On-Premises Considerations, Total Cost of Ownership, Vendor Selection and Technology Strategy.

Textbook 1: Chapters 3.1–3.8 and 4.1–4.5 Number of Hours:9

Module-3

Data Generation, Storage, and Ingestion: Data Generation in Source Systems, APIs, Application Databases, OLTP and OLAP Systems, Change Data Capture, Logs, Message Queues and Event Streams, Data Sharing, Storage Fundamentals, File Storage, Block Storage, Object Storage, Distributed Storage Systems, Data Warehouses, Data Lakes, Data Lakehouses, Data Platforms, Data Ingestion, Batch and Stream Ingestion, ETL and ELT, Serialization and Deserialization, Throughput, Reliability, Scalability, Data Migration.

Textbook 1: Chapters 5.1–5.6, 6.1–6.5, and 7.1–7.6 Number of Hours:8

Module-4

Queries, Transformation, and Data Serving: Queries, Data Modeling, Normalization and Denormalization, Data Transformation, Data Integration, Data Quality, Data Pipelines, Analytics Engineering, Serving Data for Analytics, Business Analytics, Operational Analytics, Embedded Analytics, Machine Learning Support, Data Products, Data Mesh, Query Federation, Semantic and Metrics Layers, Reverse ETL, Data Sharing and Self-Service Analytics.

Textbook 1: Chapters 8.1–8.10 Number of Hours: 8

Module-5

Security, Privacy, and Future Trends in Data Engineering: Security Fundamentals, Security Policies, Principle of Least Privilege, Encryption, Logging, Monitoring and Alerting, Network Access, Data Privacy, Governance Considerations, Shared Responsibility in Cloud Environments, Data Backup and Recovery, Emerging Data Engineering Trends, Modern Data Stack, Live Data Stack, Streaming Pipelines, Real-Time Analytics, Data and Machine Learning Integration, Future Roles and Responsibilities of Data Engineers.

Textbook1: Chapters 9.1–9.8 and 10.1–10.6 Number of Hours: 9

Suggested Learning Resources: (Text Book/ Reference Book/ Manuals):

Text books:

1. Joe Reis and Matt Housley, Fundamentals of Data Engineering: Plan and Build Robust Data Systems, O'Reilly Media, 2022.

Reference books / Manuals:

1. Martin Kleppmann and Chris Riccomini, Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems, 2nd Edition, O'Reilly Media, 2026.
2. James Densmore, Data Pipelines Pocket Reference: Moving and Processing Data for Analytics, 1st Edition, O'Reilly Media, 2021.
3. Tyler Akidau, Slava Chernyak, and Reuven Lax, Streaming Systems: The What, Where, When, and How of Large-Scale Data Processing, 1st Edition, O'Reilly Media, 2018.
4. Ralph Kimball and Margy Ross, The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling, 3rd Edition, John Wiley & Sons, 2013.

Web links and Video Lectures (e-Resources):

- Data Engineering Zoomcamp: https://www.youtube.com/@DataTalksClub/playlists
- Data Engineering Zoom camp GitHub Repository: https://github.com/DataTalksClub/data-engineering-zoomcamp
- Datatalks.club: https://www.youtube.com/@DataTalksClub?utm_source=chatgpt.com

Assessment Structure (both CIE and SEE)

The weightage assigned to Continuous Internal Evaluation (CIE) and Semester End Examination (SEE) shall be **50% each**. A student shall secure a minimum of 50% of the maximum marks prescribed for CIE and a minimum of **40%** of the maximum marks prescribed for SEE to be eligible for passing.

A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to a subject/course only if he/she secures not less than **50% of the aggregate marks** obtained in CIE and SEE taken together.

Continuous Internal Evaluation:

Continuous Comprehensive Assessments (CCA)+ Internal Assessment Test (IA) = Continuous Internal Evaluation CIE [25+25=50 marks]

A minimum of **two Internal Assessment Tests (IA)** shall be conducted. Each test shall be conducted for a maximum of **25 marks**. The average of the two test marks shall be the marks secured by a student in IA.

In addition, **Continuous Comprehensive Assessment (CCA)** shall be conducted for a total of **25 marks**. It is recommended to include **a maximum of two learning activities** as part of the CCA to foster the **holistic development of students**.

Software Automation Testing Laboratory			
Course Code	**1MCSL107A**	Scheme	2026
Type of course	Professional Core Course Lab	Semester	I
Teaching Hours/Week (L:T:P)	0: 0: 2	CIE Marks	50
Total Hours of Pedagogy L:T:P:SL&TW	0: 0: 28 :2	SEE Marks	50
Credits	**1**	Total Marks	100
Type of Examination	**Practical**	Exam Hours	3

Course outcomes (Skill Set)

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

1. Implement automated UI and API test scripts using Playwright and WireMock to validate software functionality.
2. Analyze automation test results, identify defects, evaluate test coverage, and troubleshoot failures using appropriate testing tools and reports.
3. Evaluate automation frameworks, testing strategies, and test artifacts to improve software quality and testing efficiency.
4. Develop a scalable end-to-end automation testing framework by integrating UI testing, API mocking, reporting, and continuous testing practices for real-world applications.

Note:

1. The laboratory syllabus consists of PART-A and PART-B. While PART-A has 6-8 conventional experiments, PART-B has 6 typical open-ended experiments. The maximum marks for the laboratory course are 100.
2. Both PART-A and PART-B are considered for CIE and SEE.
3. Students have answer 1(one) question from PART-A and 1(one) question from PART-B.
   a. The questions set for SEE shall be from among the experiments under PART-A. It is evaluated for **80 marks** out of the maximum 100 marks.
   b. The open-ended question set for SEE shall be any other open-ended question and not selected from the experiments under PART-A. It shall be evaluated for **20 marks.**

For Continuous Internal Evaluation (CIE) during the semester, classwork shall typically include open-ended questions from **Part-B**, along with other similar questions designed to enhance the students' skills.

PART – A
List of Experiments

1. Develop and execute an automated test using Playwright to validate the login functionality of a web application with valid and invalid credentials. Analyze the execution results to identify failed scenarios and evaluate the effectiveness of the implemented automation script in improving software quality.
2. Design and implement automated tests using Playwright to validate the search and filtering functionality of an e-commerce application with multiple input

combinations. Analyze the test results for different scenarios and evaluate the robustness and maintainability of the automation scripts.

3. Develop an end-to-end automated test suite using Playwright to validate shopping cart and checkout functionalities of an e-commerce application. Analyze the execution outcomes and evaluate the test suite based on coverage, reliability, and execution efficiency.
4. Configure WireMock to simulate a REST API using predefined stub responses for GET requests. Analyze the mocked API responses under different request conditions and evaluate the effectiveness of service virtualization in supporting application testing.
5. Create WireMock stubs to simulate POST requests with valid and invalid input data for a REST API. Analyze the application's behavior using mocked responses and evaluate the suitability of WireMock for testing applications when backend services are unavailable.
6. Integrate Playwright with WireMock to automate the testing of a web application that interacts with mocked REST APIs. Analyze the interaction between the user interface and mocked services and evaluate the reliability, repeatability, and efficiency of the integrated automation testing approach.

PART – B

OPEN ENDED EXPERIMENTS

Design and Implement an End-to-End Automated Testing Framework for a Web Application: Develop a scalable automated testing framework for any open-source web application (banking, education, healthcare, etc.) using Playwright. Identify critical user workflows, automate the test scenarios, execute them under different conditions, and analyze the test outcomes.

1. **Design and implement a data-driven automated testing framework using Playwright /** Selenium / Available open source **for a selected web application.** Identify critical workflows and execute the same test scenarios using multiple sets of valid, invalid, boundary, and unexpected input data. Implement parameterization, test fixtures, reusable functions, and appropriate assertions. Analyze the test results to identify defects and evaluate the maintainability and reusability of the framework.
2. **Develop an automated testing solution using Playwright /** Selenium / Available open source **to evaluate a selected web application across multiple browsers and viewport configurations.** Identify critical functional and UI scenarios, execute them across different browser engines and screen sizes, capture failures and screenshots, and compare the test results. Analyze browser-specific and responsive behaviour and evaluate the effectiveness of automated cross-browser testing.
3. **Design and implement a service virtualization framework using WireMock for a selected REST-based application.** Identify different API response conditions such as successful responses, validation errors, authentication failures, timeouts, unavailable services, and server errors. Create suitable WireMock stubs and test how the application responds to these conditions. Analyze the results and evaluate the effectiveness of API contract and negative testing.

4. **Develop an automated testing framework using Playwright /** Selenium / Available open source **that generates comprehensive test execution reports for a selected web application.** Execute a suitable set of functional test cases, capture screenshots and traces for failed scenarios, categorize failures, and analyze test execution time, pass/fail statistics, and coverage. Develop a mechanism to identify recurring failures and evaluate the effectiveness of automated reporting in software quality assessment.
5. **Design and implement a Continuous Testing pipeline for a web application by integrating Playwright-based automated tests with a CI/CD platform such as GitHub Actions or Jenkins.** Configure automated test execution whenever application changes are introduced, collect test results and artifacts, and configure the pipeline to identify and report failures. Analyze the benefits and limitations of continuous testing in improving software quality and reducing regression testing effort.
6. **Design and implement a scalable end-to-end quality engineering framework for a selected open-source web application by integrating UI automation using Playwright, API simulation using WireMock, reusable test components, test data management, automated reporting, and continuous test execution.** Identify critical business workflows, develop functional and negative test scenarios, execute the tests under different conditions, analyze failures and coverage, and evaluate the scalability, reliability, maintainability, and efficiency of the proposed framework.

Textbooks:

1. Paul C. Jorgensen, The Craft of Software Testing: Managing the Testing Process, *3rd* Edition, Auerbach Publications.
2. Alan Richardson, *Selenium WebDriver Recipes in C#, 2nd Edition*, Leanpub
3. Srinivasan Desikan and Gopalaswamy Ramesh, *Software Testing: Principles and Practices*, Pearson Education.

Reference books / Manuals:

1. Evan Burchard, The Web Application Hacker's Handbook*, Wiley.*

Web links and Video Lectures (e-Resources):

- https://playwright.dev
- https://www.wiremock.io/webinars/developer-portals
- https://playwright.dev/community/learn-videos

Assessment Structure (both CIE and SEE)

The weightage assigned to Continuous Internal Evaluation (CIE) and Semester End Examination (SEE) shall be 50% each. A student shall secure a minimum of 50% of the maximum marks prescribed for CIE and a minimum of 40% of the maximum marks prescribed for SEE to be eligible for a pass in the course.

A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to a subject /course only if he/she secures not less than 50% of the aggregate marks obtained in CIE and SEE taken together.

Continuous Internal Evaluation (CIE):

For the Professional Core Course Laboratory, the maximum marks prescribed are 100. Out of these, 50 marks are allotted for Continuous Internal Evaluation (CIE) and 100 marks for the Semester End Examination (SEE). The SEE marks obtained out of 100 shall be proportionately scaled down to 50 marks. The final result shall be declared based on the aggregate of the CIE marks (50) and the scaled-down SEE marks (50), totalling 100 marks."

Out of 50 marks, 40 marks shall be assigned for assessment of regular laboratory PART -A classwork, and 10 marks for Part – B. (Table 1)

Out of 40 marks, 25 marks shall be based on the regular classwork and allotment of marks shall be as per rubrics listed in Table 1.

The remaining 15 marks shall be based on the practical test conducted by two Internal examiners appointed by the HoD. The allotment of marks shall be as per the rubrics defined for the practical test as indicated in the Table 1.

The open-ended question set for CIE shall be any other open-ended question and not selected from the experiments under PART-A. It shall be evaluated for 10 marks. The allotment of marks shall be as per rubrics listed in Table 2

Table – 1 Distribution of marks corresponding to laboratory component

Component	Description	Marks
Allotment of marks for regular classwork		
Ability in Conducting Experiments	Initiative, skill, safety practices, teamwork, and independent handling of equipment during regular lab sessions.	10 marks
Laboratory Record / Journal	Regular and neat maintenance of lab record, accuracy of results, and timely submission.	15 marks
	Subtotal	**25**
Allotment of marks for CIE practical test		
Laboratory Test / Experiment Performance	Ability to set up apparatus, follow procedure, take observations, and obtain correct results in the test experiment.	5 marks
Viva-Voce	Understanding of theory, concepts, and experimental procedure; ability to explain results and answer related questions.	10marks
	Subtotal	**15**
	Total	**40**

Table 2 Distribution of marks corresponding to PART -B of the laboratory component Evaluation Rubrics for continuous internal evaluation CIE). Maximum marks:10				
Criteria and Guiding factors	**Students' Level and Rubrics** Excellent: E, Good: G, Fair: F Needs Improvements: NI			
	E	**G**	**F**	**NI**
(1) Problem Understanding **Criteria:** Is the problem understood correctly and framed appropriately? (i)Clarity in interpreting the given problem. (ii)Identification of relevant concepts and constraints. (iii) Ability to rephrase or model the problem logically.	2.0	1.5	1.0	0.5
(2) Conceptual Knowledge **Criteria**: Are the chosen methods or concepts valid for this type of problem? (i)Application of appropriate theories, principles, or laws. (ii)Integration of multiple concepts where necessary. (iii)Accuracy in stating relevant equations or models.	2.0	1.5	1.0	0.5
(3) Approach and Strategy **Criteria**: Is the approach well thought out and methodical? (i)Logical structure and progression. (ii)Use of diagrams, sketches, or flowcharts if applicable. (iii)Identification of assumptions and boundary conditions.	2.0	1.5	1.0	0.5
(4) Execution and Accuracy **Criteria**: Are the calculations correct? Is the final solution reasonable? (i) Correct application of formulas/methods. (ii)Accuracy of calculations or simulations (if any). (iii)Unit consistency and result verification.	2.0	1.5	1.0	0.5
(5) Innovation or Depth **Criteria**: Has the student shown original thinking or alternative methods? (i)Novel or efficient solutions. (ii)Consideration of alternate approaches. (iii)Interdisciplinary integration or practical extensions.	1	0.75	0.5	0.25
(6) Presentation and Clarity **Criteria**: Is the answer easy to follow and professionally written? (i)Organization and neatness. (ii)Clarity of explanation, labelling, and steps. (iii) Use of proper technical terminology.	1	0.75	0.5	0.25
Total	10	7.5	5.0	2.5
Note: (i)The marks shown for Excellent, Good, Fair, Needs Improvement are indicative. The evaluator may also assign intermediate scores depending on the quality of work. (ii)If the total score results in a fraction, it should be rounded off to the next higher integer.				

(iii)The listed criteria are guidelines; evaluators may also consider other authentic dimensions (e.g., depth of insight, critical thinking, and practical awareness, recognition of limitations, or justified compromises.
(iv)To award the full10 marks in the excellent category, the student must demonstrate reasonably good performance across all six criteria; otherwise, proportionally lower marks may be given.

****

	Computer Vision Laboratory			
	Course Code	**1MCSL107B**	Scheme	2026
	Type of course	Professional Core Course Lab	Semester	I
	Teaching Hours/Week (L:T:P)	0:0:2	CIE Marks	50
	Total Hours of Pedagogy L:T:P:SL:TW	0 : 0 : 28 : 2	SEE Marks	50
	Credits	**01**	Total Marks	100
	Type of Examination	**Practical**	Exam Hours	3

Course outcomes (Skill Set)

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

CO1: Apply basic image processing techniques such as gray scale conversion, filtering, histogram analysis, and segmentation.

CO2: Implement edge, corner, and feature detection methods to extract important information from images.

CO3: Apply feature matching techniques to compare and analyze images.

CO4: Develop and evaluate basic deep learning models (CNN) for image classification tasks.

CO5: Design and implement computer vision solutions for real-world applications such as object detection, face recognition, and motion analysis

Note:

1. The laboratory syllabus consists of PART-A and PART-B. While PART-A has 6 conventional experiments, PART-B has 6 typical open-ended experiments. The maximum marks for the laboratory course are 100.
2. Both PART-A and PART-B are considered for CIE and SEE.
3. Students have to answer 1(one) question from PART-A and 1(one) question from PART-B.

a. The questions set for SEE shall be from among the experiments under PART-A. It is evaluated for **70 marks** out of the maximum 100 marks.

a. The open-ended question set for SEE shall be any other open-ended question and not selected from the experiments under PART-A. It shall be evaluated for **30 marks.**

4. For Continuous Internal Evaluation (CIE) during the semester, classwork shall typically include open-ended questions from **Part-B**, along with other similar questions designed to enhance the students' skills.

PART – A

List of experiments

1. Write a program to read an image, convert it into grayscale, display its histogram, and perform geometric transformations such as translation, rotation, and scaling.
2. Write a program to apply smoothing filters (Average and Gaussian) and detect edges using Sobel and Canny edge detection methods.
3. Write a program to detect corner points using Harris Corner Detection and perform feature matching between two images using ORB descriptors.
4. Write a program to segment an image using thresholding techniques and analyze the segmented regions.

5. Write a program to build and train a Convolutional Neural Network (CNN) for image classification using a standard dataset such as MNIST or CIFAR-10.
6. Write a program to estimate camera intrinsic and extrinsic parameters using calibration images and perform basic 3D point reconstruction or depth estimation.

PART – B

OPEN ENDED EXPERIMENTS

Open-ended experiments are a type of laboratory activity where the outcome is not predetermined and students are given the freedom to explore, design, and conduct the experiment based on the problem statements as per the concepts defined by the course coordinator. It encourages creativity, critical thinking, and inquiry-based learning.

1. Design and develop a Real-Time Object Detection System using computer vision and deep learning techniques for detecting and localizing objects in images or videos.
2. Design and develop a Face Recognition or Face Mask Detection System using feature extraction and deep learning techniques.
3. Design and develop an Image Segmentation System for medical, satellite, or industrial images and evaluate its performance.
4. Design and develop an Image Stitching (Panorama Creation) System using feature detection, matching, and geometric transformations.
5. Design and develop an Optical Flow-Based Motion Detection and Tracking System for video analysis applications.
6. Design and develop an Image Classification System using Transfer Learning with pre-trained deep learning models and benchmark datasets.

Suggested Learning Resources: (Text Book/ Reference Book/ Manuals):

Text Books:

1. Computer Vision: Algorithms and Applications – Richard Szeliski, 2nd Edition, Springer, 2022.
2. Computer Vision: Models, Learning, and Inference – Simon J. D. Prince, 1st Edition, Cambridge University Press, 2012.
3. Deep Learning – Ian Goodfellow, Yoshua Bengio, Aaron Courville, MIT Press, 2016.

Reference Books / Manuals:

1. Dive into Deep Learning – Aston Zhang et al., Latest Online Edition (2023–2025).
2. Probabilistic Machine Learning: An Introduction – Kevin P. Murphy, MIT Press, 2022.
1. Learning OpenCV 4 Computer Vision with Python – Joseph Howse, Joe Minichino, Packt, 2020.

Web links and Video Lectures (e-Resources):

- Computer Vision: Algorithms and Applications (Online Book)
- Dive into Deep Learning (Interactive Book)
- Neural Networks and Deep Learning
- Stanford CS231n: Convolutional Neural Networks for Visual Recognition
- MIT OpenCourseWare – Computer Vision
- OpenCV Documentation
- PyTorch Tutorials
- TensorFlow Tutorials
- Coursera Computer Vision Courses

Assessment Structure (both CIE and SEE)

The weightage assigned to Continuous Internal Evaluation (CIE) and Semester End Examination (SEE) shall be 50% each. A student shall secure a minimum of 50% of the maximum marks prescribed for CIE and a minimum of 40% of the maximum marks prescribed for SEE to be eligible for a pass in the course.

A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to a subject /course only if he/she secures not less than 50% of the aggregate marks obtained in CIE and SEE taken together.

Continuous Internal Evaluation (CIE):

For the Professional Core Course Laboratory, the maximum marks prescribed are 100. Out of these, 50 marks are allotted for Continuous Internal Evaluation (CIE) and 100 marks for the Semester End Examination (SEE). The SEE marks obtained out of 100 shall be proportionately scaled down to 50 marks. The final result shall be declared based on the aggregate of the CIE marks (50) and the scaled-down SEE marks (50), totalling 100 marks."

Out of 50 marks, 40 marks shall be assigned for assessment of regular laboratory PART -A classwork, and 10 marks for Part – B. (Table 1)

Out of 40 marks, 25 marks shall be based on the regular classwork and allotment of marks shall be as per rubrics listed in Table 1.

The remaining 15 marks shall be based on the practical test conducted by two Internal examiners appointed by the HoD. The allotment of marks shall be as per the rubrics defined for the practical test as indicated in the Table 1.

The open-ended question set for CIE shall be any other open-ended question and not selected from the experiments under PART-A. It shall be evaluated for 10 marks. The allotment of marks shall be as per rubrics listed in Table 2.

Table – 1 Distribution of marks corresponding to laboratory component		
Component	**Description**	**Marks**
Allotment of marks for regular classwork		
Ability in Conducting Experiments	Initiative, skill, safety practices, teamwork, and independent handling of equipment during regular lab sessions.	10 marks
Laboratory Record / Journal	Regular and neat maintenance of lab record, accuracy of results, and timely submission.	15 marks
	Subtotal	**25**
Allotment of marks for CIE practical test		
Laboratory Test / Experiment Performance	Ability to set up apparatus, follow procedure, take observations, and obtain correct results in the test experiment.	5 marks
Viva-Voce	Understanding of theory, concepts, and experimental procedure; ability to explain results and answer related questions.	10marks
	Subtotal	**15**
	Total	**40**

Table 2 Distribution of marks corresponding to PART -B of the laboratory component				
Evaluation Rubrics for continuous internal evaluation CIE). Maximum marks:10				
Criteria and Guiding factors	**Students' Level and Rubrics**			
	Excellent: E, Good: G, Fair: F Needs Improvements: NI			
	E	**G**	**F**	**NI**
(1) Problem Understanding **Criteria:** Is the problem understood correctly and framed appropriately? (i)Clarity in interpreting the given problem. (ii)Identification of relevant concepts and constraints. (iii) Ability to rephrase or model the problem logically.	2.0	1.5	1.0	0.5
(2) Conceptual Knowledge **Criteria**: Are the chosen methods or concepts valid for this type of problem? (i)Application of appropriate theories, principles, or laws. (ii)Integration of multiple concepts where necessary. (iii)Accuracy in stating relevant equations or models.	2.0	1.5	1.0	0.5

(3) Approach and Strategy **Criteria**: Is the approach well thought out and methodical? (i)Logical structure and progression. (ii)Use of diagrams, sketches, or flowcharts if applicable. (iii)Identification of assumptions and boundary conditions.	2.0	1.5	1.0	0.5
(4) Execution and Accuracy **Criteria**: Are the calculations correct? Is the final solution reasonable? (i) Correct application of formulas/methods. (ii)Accuracy of calculations or simulations (if any). (iii)Unit consistency and result verification.	2.0	1.5	1.0	0.5
(5) Innovation or Depth **Criteria**: Has the student shown original thinking or alternative methods? (i)Novel or efficient solutions. (ii)Consideration of alternate approaches. (iii)Interdisciplinary integration or practical extensions.	1	0.75	0.5	0.25
(6) Presentation and Clarity **Criteria**: Is the answer easy to follow and professionally written? (i)Organization and neatness. (ii)Clarity of explanation, labelling, and steps. (iii) Use of proper technical terminology.	1	0.75	0.5	0.25
Total	10	7.5	5.0	2.5

Note:
(i)The marks shown for Excellent, Good, Fair, Needs Improvement are indicative. The evaluator may also assign intermediate scores depending on the quality of work.
(ii)If the total score results in a fraction, it should be rounded off to the next higher integer.
(iii)The listed criteria are guidelines; evaluators may also consider other authentic dimensions (e.g., depth of insight, critical thinking, and practical awareness, recognition of limitations, or justified compromises.
(iv)To award the full10 marks in the excellent category, the student must demonstrate reasonably good performance across all six criteria; otherwise, proportionally lower marks may be given.

****

	Advanced Data Structures Laboratory			
	Course Code	**1MCSL107C**	Scheme	2026
	Type of course	**Professional Core Course Laboratory**	Semester	I
	Teaching Hours/Week (L:T:P)	0 : 0 : 2	CIE Marks	50
	Total Hours of Pedagogy L:T:P:SL&TW	0 : 0 : 28 : 2	SEE Marks	50
	Credits	**1**	Total Marks	100
	Type of Examination	**Practical**	Exam Hours	3

Course outcomes (Skill Set)

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

CO1: Make use of appropriate advanced data structures to solve complex, real-world computational problems.
CO2: Develop code to perform data operations and validate results.
CO3: Make use of appropriate programming tools and development environments to implement, test, and debug advanced data structure applications.
CO4: Build a technical document consisting of experiment setup, design, implementation, and results with inferences.

Note:

1. The laboratory syllabus consists of PART-A and PART-B. While PART-A has 6 conventional experiments, PART-B has 6 typical open-ended experiments. The maximum marks for the laboratory course are 100.
2. Both PART-A and PART-B are considered for CIE and SEE.
3. Students have answer 1(one) question from PART-A and 1(one) question from PART-B.
   a. The questions set for SEE shall be from among the experiments under PART-A. It is evaluated for **80 marks** out of the maximum 100 marks.
   b. The open-ended question set for SEE shall be any other open-ended question and not selected from the experiments under PART-A. It shall be evaluated for **20 marks.**

For Continuous Internal Evaluation (CIE) during the semester, classwork shall typically include open-ended questions from **Part-B**, along with other similar questions designed to enhance the students' skills.

PART – A
List of Experiments

1. Task Priority Selection System (Heap)

- Insert tasks with priorities
- Identify and execute highest priority task

2. Job Queue Merging System (Leftist Tree)

- Merge two job queues

- Maintain priority ordering after merge

3. Adaptive Search System (Splay Tree)

- Insert and search elements
- Frequently accessed elements move to root

4. Structured Data Indexing (B-Tree)

- Insert keys into structured index
- Maintain sorted hierarchical organization

5. Dynamic Key Storage System (Extendible Hashing)

- Insert records into hash buckets
- Handle overflow using directory expansion

6. Social Group Formation System (DSU)

- Perform union operations
- Check connectivity between members

PART – B

OPEN ENDED EXPERIMENTS

Open-ended experiments are a type of laboratory activity where the outcome is not predetermined and students are given the freedom to explore, design, and conduct the experiment based on the problem statements as per the concepts defined by the course coordinator. It encourages creativity, critical thinking, and inquiry-based learning.

1. Concept 1: Stock Monitoring System (Heap)

   Track minimum and maximum stock values dynamically

2. Concept 2: Event List Combination System (Leftist Tree)

   Merge multiple event lists into one ordered list

3. Concept 3: Recently Used Item Tracker (Splay Tree)

   Track frequently accessed elements using self-adjusting behavior

4. Concept 4: Search Index Verification System (B-Tree)

   Search for existence of keys in structured index

5. Concept 5: Bucket Distribution System (Extendible Hashing)

   Observe record distribution across dynamically growing buckets

6. Concept 6: Network Connectivity Checker (DSU)

   Determine whether two nodes belong to the same connected group

Textbooks:

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, Introduction to Algorithms, 3rd Edition, MIT Press.
2. Ellis Horowitz and Sartaj Sahni, *Fundamentals of Data Structures in C*, 2nd Edition (2023), Universities Press(India) Private limited.

Reference books / Manuals:

1. Mark Allen Weiss, Data Structures and Algorithm Analysis in C++, 4th Edition, (2014) Pearson Education.
2. Sartaj Sahni, Data Structures, Algorithms and Applications in C++, 2nd Edition, (2011), Universities Press.
3. Peter Brass, Advanced Data Structures, 1st Edition, (2011), Cambridge University Press.

Web links and Video Lectures (e-Resources):

NPTEL – Advanced Data Structures:
https://nptel.ac.in/courses/106102064

MIT OpenCourseWare – Advanced Data Structures:
https://ocw.mit.edu/courses/6-851-advanced-data-structures-spring-2012/

VisuAlgo – Data Structure Visualization:
https://visualgo.net/en

GeeksforGeeks – Advanced Data Structures:
https://www.geeksforgeeks.org/advanced-data-structures/

GeeksforGeeks – Heap, Trees, Hashing, DSU (C/C++ Implementations):
https://www.geeksforgeeks.org/data-structures/

Programiz – Data Structures and Algorithms:
https://www.programiz.com/dsa

OnlineGDB – Online C/C++ Compiler:
https://www.onlinegdb.com/online_c++_compiler

Assessment Structure (both CIE and SEE)

The weightage assigned to Continuous Internal Evaluation (CIE) and Semester End Examination (SEE) shall be 50% each. A student shall secure a minimum of 50% of the maximum marks prescribed for CIE and a minimum of 40% of the maximum marks prescribed for SEE to be eligible for a pass in the course.

A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to a subject /course only if he/she secures not less than 50% of the aggregate marks obtained in CIE and SEE taken together.

Continuous Internal Evaluation (CIE):

For the Professional Core Course Laboratory, the maximum marks prescribed are 100. Out of these, 50 marks are allotted for Continuous Internal Evaluation (CIE) and 100 marks for the Semester End Examination (SEE). The SEE marks obtained out of 100 shall be proportionately scaled down to 50 marks. The final result shall be declared based on the aggregate of the CIE marks (50) and the scaled-down SEE marks (50), totalling 100 marks."

Out of 50 marks, 40 marks shall be assigned for assessment of regular laboratory PART -A classwork, and 10 marks for Part – B. (Table 1)

Out of 40 marks, 25 marks shall be based on the regular classwork and allotment of marks shall be as per rubrics listed in Table 1.

The remaining 15 marks shall be based on the practical test conducted by two Internal examiners appointed by the HoD. The allotment of marks shall be as per the rubrics defined for the practical test as indicated in the Table 1.

The open-ended question set for CIE shall be any other open-ended question and not selected from the experiments under PART-A. It shall be evaluated for 10 marks. The allotment of marks shall be as per rubrics listed in Table 2

Table – 1 Distribution of marks corresponding to laboratory component

Component	Description	Marks
Allotment of marks for regular classwork		
Ability in Conducting Experiments	Initiative, skill, safety practices, teamwork, and independent handling of equipment during regular lab sessions.	10 marks
Laboratory Record / Journal	Regular and neat maintenance of lab record, accuracy of results, and timely submission.	15 marks
	Subtotal	**25**
Allotment of marks for CIE practical test		
Laboratory Test / Experiment Performance	Ability to set up apparatus, follow procedure, take observations, and obtain correct results in the test experiment.	5 marks
Viva-Voce	Understanding of theory, concepts, and experimental procedure; ability to explain results and answer related questions.	10marks
	Subtotal	**15**
	Total	**40**

Table 2 Distribution of marks corresponding to PART -B of the laboratory component				
Evaluation Rubrics for continuous internal evaluation CIE). Maximum marks:10				
Criteria and Guiding factors	**Students' Level and Rubrics**			
	Excellent: E, Good: G, Fair: F Needs Improvements: NI			
	E	**G**	**F**	**NI**
(1) Problem Understanding **Criteria:** Is the problem understood correctly and framed appropriately? (i)Clarity in interpreting the given problem. (ii)Identification of relevant concepts and constraints. (iii) Ability to rephrase or model the problem logically.	2.0	1.5	1.0	0.5
(2) Conceptual Knowledge **Criteria**: Are the chosen methods or concepts valid for this type of problem? (i)Application of appropriate theories, principles, or laws. (ii)Integration of multiple concepts where necessary. (iii)Accuracy in stating relevant equations or models.	2.0	1.5	1.0	0.5
(3) Approach and Strategy **Criteria**: Is the approach well thought out and methodical? (i)Logical structure and progression. (ii)Use of diagrams, sketches, or flowcharts if applicable. (iii)Identification of assumptions and boundary conditions.	2.0	1.5	1.0	0.5
(4) Execution and Accuracy **Criteria**: Are the calculations correct? Is the final solution reasonable? (i) Correct application of formulas/methods. (ii)Accuracy of calculations or simulations (if any). (iii)Unit consistency and result verification.	2.0	1.5	1.0	0.5
(5) Innovation or Depth **Criteria**: Has the student shown original thinking or alternative methods? (i)Novel or efficient solutions. (ii)Consideration of alternate approaches.	1	0.75	0.5	0.25

(iii)Interdisciplinary integration or practical extensions.				
(6) Presentation and Clarity **Criteria**: Is the answer easy to follow and professionally written? (i)Organization and neatness. (ii)Clarity of explanation, labelling, and steps. (iii) Use of proper technical terminology.	1	0.75	0.5	0.25
Total	10	7.5	5.0	2.5

Note:

(i)The marks shown for Excellent, Good, Fair, Needs Improvement are indicative. The evaluator may also assign intermediate scores depending on the quality of work.

(ii)If the total score results in a fraction, it should be rounded off to the next higher integer.

(iii)The listed criteria are guidelines; evaluators may also consider other authentic dimensions (e.g., depth of insight, critical thinking, and practical awareness, recognition of limitations, or justified compromises.

(iv)To award the full10 marks in the excellent category, the student must demonstrate reasonably good performance across all six criteria; otherwise, proportionally lower marks may be given.

****

	Micro Services Development Laboratory			
	Course Code	**1MCSL107D**	Scheme	2026
	Type of course	**Professional Core Course Lab**	Semester	I
	Teaching Hours/Week (L:T:P)	0:0:2	CIE Marks	50
	Total Hours of Pedagogy L:T:P:SL&TW	0 : 0 : 28 : 2	SEE Marks	50
	Credits	**01**	Total Marks	100
	Type of Examination	Practical	Exam Hours	3

Course outcomes (Skill Set)

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

CO1: Explain the architecture and components of a Spring Boot microservice application and configure the development environment.

CO2: Develop RESTful microservices with database connectivity, validation, and exception handling.

CO3: Implement communication, service discovery, and request routing among microservices.

CO4: Deploy, document, and test microservices using containerization and API testing tools.

CO5: Develop a simple distributed microservices application by integrating multiple microservices.

Note:

1. The laboratory syllabus consists of PART-A and PART-B. While PART-A has 6 conventional experiments, PART-B has 6 typical open-ended experiments. The maximum marks for the laboratory course are 100.
2. Both PART-A and PART-B are considered for CIE and SEE.
3. Students have to answer 1(one) question from PART-A and 1(one) question from PART-B.

a. The questions set for SEE shall be from among the experiments under PART-A. It is evaluated for **70 marks** out of the maximum 100 marks.

a. The open-ended question set for SEE shall be any other open-ended question and not selected from the experiments under PART-A. It shall be evaluated for **30 marks.**

4. For Continuous Internal Evaluation (CIE) during the semester, classwork shall typically include open-ended questions from **Part-B**, along with other similar questions designed to enhance the students' skills.

Pre-Requisite Exercise 1 (not to be included in evaluation)

Development Environment Setup and Spring Boot Project Initialization

Objective

To configure a complete microservices development environment and create a basic Spring Boot application capable of exposing RESTful web services.

Description

Configure the software tools required for microservices development, including the Java Development Kit (JDK), Maven, an Integrated Development Environment (IDE), a relational database, Postman, Docker, and Git.

Generate a Spring Boot project using Spring Initializr and configure the required project dependencies for building RESTful applications.
Establish database connectivity, build and execute the application, develop a basic RESTful web service, and verify the successful execution of REST APIs using a web browser and Postman.

Note

This exercise provides the essential software environment and development workflow required for all subsequent laboratory experiments.
Students shall be able to

- Install and configure Java Development Kit.
- Configure Maven for dependency management.
- Install and configure an IDE.
- Generate Spring Boot projects using Spring Initializr.
- Build and execute Spring Boot applications.
- Configure database connectivity.
- Test REST APIs using Postman.
- Manage project source code using Git.

Deliverables

- Successfully configured development environment.
- Executable Spring Boot application.
- Working REST endpoint.
- Maven project structure.
- Git repository containing the project.

Pre-Requisite Exercise 2 (not to be included in evaluation)

Development of a Reusable Spring Boot Microservice Framework

Objective

To develop a reusable Spring Boot microservice framework implementing a layered architecture that can serve as the foundation for all subsequent laboratory experiments.

Description

Develop a reusable Spring Boot framework by implementing the standard project structure and a layered architecture comprising controller, service, repository, entity, DTO, configuration, validation, and exception-handling components.

Configure database persistence, dependency management, application properties, and API documentation to create a robust and maintainable microservice framework.

Implement a sample REST endpoint to validate the framework and ensure proper integration of all architectural layers.

The framework developed in this exercise shall be reused and extended in all subsequent laboratory experiments, enabling students to focus on implementing microservices concepts rather than repeatedly creating project infrastructure.

Students shall be able to
Design a layered microservice architecture.
Implement Controller, Service, Repository, Entity, and DTO layers.
Configure database persistence using Spring Data JPA.
Implement input validation.
Develop global exception handling mechanisms.
Configure application properties.
Generate API documentation using Swagger/OpenAPI.
Create a reusable project template.
Framework Components

The reusable framework shall contain:

Maven project structure
Configuration package
Controller package
Service package
Repository package
Entity package
DTO package
Exception package
Validation support
Database configuration
Swagger/OpenAPI configuration
Sample REST endpoint
Deliverables
Fully functional reusable Spring Boot framework.
Standard package structure.
Configured database connectivity.
Sample CRUD-ready architecture.
Swagger-generated API documentation.
Reusable project template for subsequent experiments.

PART – A
List of experiments

Experiment 1
Development of a Product Management Microservice

Objective To develop a RESTful Product Management microservice by extending the reusable Spring Boot microservice framework with database persistence, validation, and exception handling.

Description Extend the reusable Spring Boot microservice framework developed during the pre-requisite exercises to implement a Product Management microservice. Design the Product entity and corresponding repository, service, controller, and DTO classes following the layered architecture.

Implement RESTful CRUD operations for managing product information using Spring Data JPA and a relational database. Incorporate input validation and global exception handling to ensure reliable and maintainable service implementation.

Test the REST APIs using Postman and verify the correctness of all CRUD operations and error-handling mechanisms.

Students shall be able to
Extend an existing Spring Boot microservice framework.
Design entity and DTO classes.
Implement repository, service, and controller layers.
Develop RESTful CRUD APIs.
Configure database persistence using Spring Data JPA.
Apply input validation.
Implement global exception handling.
Test REST APIs using Postman.

Deliverables

Product Management microservice.
Database schema and entity classes.
Functional CRUD REST APIs.
Validated request handling.
Tested API collection.

Experiment 2

Development of a Customer Management Microservice

Objective : To develop an independent Customer Management microservice using the reusable Spring Boot framework and implement RESTful services for customer information management.

Description: Develop a Customer Management microservice by extending the reusable framework created during the pre-requisite exercises. Design the Customer entity together with the corresponding repository, service, controller, and DTO components using the standard layered architecture.

Implement RESTful CRUD APIs for customer management with persistent storage using Spring Data JPA. Apply validation and exception-handling mechanisms to ensure reliable processing of user requests. Verify all APIs using Postman and evaluate the correctness of the implemented microservice.

Students shall be able to

Develop an independent microservice.
Design layered application components.
Implement CRUD services.
Manage persistent customer data.
Apply validation techniques.
Handle application exceptions.

Test REST APIs.

Deliverables

Customer Management microservice.
Functional REST APIs.
Persistent customer database.
Validated API responses.
Tested Postman collection.

Experiment 3

Development of Communicating Microservices

Objective: To develop two independent microservices and implement synchronous inter-service communication using REST-based service invocation mechanisms.

Description: Develop two lightweight microservices by extending the reusable Spring Boot framework. Design suitable REST APIs for each microservice and implement synchronous communication using Spring REST Client or OpenFeign to exchange information between them.

Verify the correctness of service interactions by invoking APIs and exchanging JSON data. Implement appropriate exception handling to manage communication failures and validate the responses using Postman.

Students shall be able to

Develop multiple independent microservices.
Implement synchronous inter-service communication.
Exchange data between services.
Handle communication exceptions.
Test service interactions using Postman.

Deliverables

Two independent microservices.
Functional REST-based service communication.
Successful data exchange between services.
Tested Postman collection.

Experiment 4

Development of Service Discovery and API Gateway

Objective :To configure service registration, service discovery, and centralized API routing in a distributed microservices environment.

Description: Develop two simple microservices using the reusable framework and configure them to register with a Service Discovery Server. Configure an API Gateway to route client requests dynamically to the registered services.

Verify successful service registration, service discovery, and gateway-based request routing. Test different request scenarios using Postman and evaluate the behaviour of the distributed application.

Students shall be able to
Configure a Service Discovery Server.
Register microservices dynamically.
Configure API Gateway routing.
Route client requests through the gateway.
Verify service registration and discovery.
Deliverables
Service Discovery Server.
Registered microservices.
Configured API Gateway.
Successful gateway-based routing.

Experiment 5
Containerization and Deployment of Microservices.

Objective: To containerize and deploy multiple microservices using Docker and Docker Compose.

Description: Develop one or more microservices using the reusable framework and containerize them by creating Docker images and Dockerfiles. Configure Docker Compose to deploy the microservices together with the required database services.

Verify successful deployment, container networking, and application execution in the containerized environment. Evaluate the portability and deployment characteristics of the application.

Students shall be able to
Create Dockerfiles.
Build Docker images.
Configure Docker Compose.
Deploy multiple containers.
Verify container networking and execution.
Deliverables
Dockerfiles.
Docker images.
Docker Compose configuration.
Running containerized microservices.

Experiment 6
API Documentation, Functional Testing, and Performance Evaluation.

Objective: To document, test, and evaluate RESTful microservices using API documentation and software testing tools.

Description: Develop a microservice using the reusable framework and generate API documentation using Swagger/OpenAPI. Perform functional testing using Postman by executing various request scenarios and validating the correctness of API responses.

Conduct basic performance evaluation using JMeter or an equivalent testing tool by measuring response time and throughput under different workloads. Analyse the observed performance characteristics and document the test results.

Students shall be able to
Generate API documentation.
Perform functional API testing.
Design API test cases.
Conduct basic performance evaluation.
Analyse application performance.
Deliverables
Swagger/OpenAPI documentation.
Functional API test cases.
Performance testing report.
Performance analysis results.

PART – B
OPEN ENDED EXPERIMENTS

Experiment 1
Design and Development of an E-Commerce Microservices Application

Objective: To design and develop a distributed e-commerce application using multiple microservices and contemporary microservices technologies.

Description: Design and develop an e-commerce application by identifying suitable business services such as Product Management, Customer Management, Order Management, Inventory Management, and Payment Processing. Develop the required microservices using the reusable Spring Boot framework and implement REST-based communication among them.
Integrate service discovery, API gateway, database persistence, and containerized deployment. Test the application using Postman and generate API documentation to validate the correctness of the distributed system.

Students shall be able to
Design a distributed microservices architecture.
Identify service boundaries.
Develop multiple collaborating microservices.
Integrate service discovery and API gateway.
Deploy and validate the complete application.

Deliverables
Complete e-commerce microservices application.
Integrated distributed services.

API documentation.
Containerized deployment.
Functional testing report.

Experiment 2
Design and Development of a Smart Healthcare Management System

Objective : To design and develop a distributed healthcare management application using multiple collaborating microservices.

Description:Design and develop a healthcare management system by identifying services such as Patient Management, Doctor Management, Appointment Scheduling, Billing, and Medical Records. Develop the required microservices using the reusable framework and implement communication among services.

Configure service discovery, API gateway, database persistence, and deployment using Docker Compose. Test and document the developed application.

Students shall be able to
Model healthcare services as microservices.
Develop integrated service-oriented applications.
Configure distributed application infrastructure.
Test and document the developed solution.
Deliverables
Healthcare microservices application.
Integrated services.
API documentation.
Deployment configuration.
Test report.

Experiment 3
Design and Development of a Secure Microservices Application

Objective : To design and develop a secure microservices application by implementing authentication and authorization mechanisms.

Description: Design and develop a distributed microservices application by integrating authentication and authorization using Spring Security and JSON Web Token (JWT) or equivalent security mechanisms. Protect REST APIs, manage user authentication, and authorize access to secured resources.
Test secured APIs using Postman and verify authentication, authorization, and protected resource access under different user scenarios.

Students shall be able to
Design secure RESTful services.
Implement authentication mechanisms.
Configure authorization rules.
Secure distributed microservices.

Test secured APIs.

Deliverables

Secure microservices application.
Authentication and authorization implementation.
Protected REST APIs.
Security testing report.

Experiment 4

Design and Development of a Cloud-Native Microservices Solution

Objective: To design, develop, deploy, and evaluate a complete cloud-native microservices solution for a real-world application.

Description: Identify a suitable real-world problem and design a cloud-native microservices architecture by decomposing the application into independent services. Develop the required microservices using the reusable framework and integrate service communication, service discovery, API gateway, containerization, API documentation, and testing.

Deploy the complete application using Docker Compose, perform functional testing, and evaluate the overall behaviour, scalability, and maintainability of the developed solution.

Students shall be able to

Design complete cloud-native applications.
Develop integrated distributed systems.
Apply microservices architectural principles.
Deploy containerized applications.
Evaluate the developed solution.

Deliverables

Complete cloud-native application.
Distributed microservices architecture.
Containerized deployment.
API documentation.
Functional testing report.
Project report.

Experiment 5

Design and Development of a Smart Campus Microservices Application

Objective: To design and develop a distributed smart-campus application using independently deployable microservices.

Description: Design and develop a smart-campus system by identifying appropriate services such as Student Management, Attendance, Classroom/Laboratory Management, Asset Management, Notification, and Maintenance Request services. Develop suitable microservices and integrate them using REST-based or event-driven communication. Incorporate database persistence, service discovery, API gateway, security where appropriate, and containerized deployment. Test and document the complete system.

Students shall be able to: Identify microservice boundaries from a real-world problem; design collaborating services; integrate multiple microservices; configure distributed infrastructure; deploy and validate a complete application.

Deliverables: Smart-campus microservices application; architecture diagram; integrated services; API documentation; containerized deployment; testing report.

Experiment 6
Design and Development of a Library Management Microservices Application

Objective: To design and develop a simple Library Management application using multiple microservices.

Description: Design and develop a Library Management application consisting of suitable services such as Book Management, Member Management, and Issue/Return Management. Develop the required microservices using Spring Boot and implement REST-based communication among them. Configure database persistence and test the integrated application using Postman.

Students shall be able to:
Identify services in a simple application.
Develop independent Spring Boot microservices. Implement REST APIs for different services. Establish communication between microservices. Store and retrieve application data. Test the integrated application using Postman.

Deliverables:
Library Management microservices application.
Minimum three functional microservices.
REST APIs for the identified services.
Database configuration.
Postman test collection.
Brief architecture diagram.

Suggested Learning Resources: (Text Book/ Reference Book/ Manuals):

Text Books

1. Chris Richardson, Microservices Patterns: With Examples in Java, Manning Publications, 2018.
2. Sam Newman, Building Microservices: Designing Fine-Grained Systems, 2nd Edition, O'Reilly Media, 2021.
3. Laurentiu Spilca, Spring Start Here: Learn What You Need and Learn It Well, Manning Publications, 2021.

Reference Books

1. Craig Walls, Spring Boot in Action, Manning Publications, 2016.
2. Mark Richards and Neal Ford, Fundamentals of Software Architecture: An Engineering Approach, O'Reilly Media, 2020.
3. Eberhard Wolff, Microservices: Flexible Software Architecture, Addison-Wesley Professional, 2017.
4. Brendan Burns, Designing Distributed Systems: Patterns and Paradigms for Scalable, Reliable Services, O'Reilly Media, 2018.
5. Spring Boot Reference Documentation (Official).
6. Spring Cloud Reference Documentation (Official).
7. Docker Documentation (Official).

Web links and Video Lectures (e-Resources):

Web Resources

1. Spring Boot Official Documentation
2. Spring Boot Project Home and Getting Started Guides
3. Spring Boot Docker Guide
4. Docker Java Language Guide
5. Docker Spring Boot Samples
6. Spring Cloud Documentation
7. Spring Initializr
8. Postman Learning Center
9. Apache JMeter User Manual
10. OpenAPI Initiative Documentation

Suggested Video Resources

1. Java Brains (Spring Boot and Microservices)
2. Spring Developer YouTube Channel (Official)
3. Amigoscode (Spring Boot and Microservices)
4. TechWorld with Nana (Docker, Kubernetes and DevOps)
5. Programming Techie (Spring Boot Microservices)
6. Daily Code Buffer (Spring Boot and Spring Cloud)
7. CodeJava (Spring Boot Tutorials)
8. freeCodeCamp.org (Java and Spring Boot Courses)

Laboratory Requirements

The laboratory shall be equipped with the necessary hardware and software infrastructure to support the development, deployment, testing, and evaluation of microservices-based applications.

Hardware Requirements

- Desktop/Laptop computers with minimum 8 GB RAM (16 GB recommended)
- Multi-core processor (Intel Core i5/AMD Ryzen 5 or equivalent)
- Minimum 20 GB free storage
- LAN/Wi-Fi connectivity
- Internet connectivity for software installation and documentation access (or offline repositories where applicable)

Software Requirements

- Operating System: Windows/Linux/macOS
- Java Development Kit (JDK)
- Spring Boot
- Spring Cloud
- Maven
- IntelliJ IDEA Community Edition / Eclipse IDE / Visual Studio Code
- PostgreSQL or MySQL
- Git
- Docker and Docker Compose
- Postman
- Swagger/OpenAPI
- Apache JMeter (or equivalent)

Assessment Structure (both CIE and SEE)

The weightage assigned to Continuous Internal Evaluation (CIE) and Semester End Examination (SEE) shall be 50% each. A student shall secure a minimum of 50% of the maximum marks prescribed for CIE and a minimum of 40% of the maximum marks prescribed for SEE to be eligible for a pass in the course.

A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to a subject /course only if he/she secures not less than 50% of the aggregate marks obtained in CIE and SEE taken together.

Continuous Internal Evaluation (CIE):

For the Professional Core Course Laboratory, the maximum marks prescribed are 100. Out of these, 50 marks are allotted for Continuous Internal Evaluation (CIE) and 100 marks for the Semester End Examination (SEE). The SEE marks obtained out of 100 shall be

proportionately scaled down to 50 marks. The final result shall be declared based on the aggregate of the CIE marks (50) and the scaled-down SEE marks (50), totalling 100 marks."

Out of 50 marks, 40 marks shall be assigned for assessment of regular laboratory PART -A classwork, and 10 marks for Part – B. (Table 1)
Out of 40 marks, 25 marks shall be based on the regular classwork and allotment of marks shall be as per rubrics listed in Table 1.
The remaining 15 marks shall be based on the practical test conducted by two Internal examiners appointed by the HoD. The allotment of marks shall be as per the rubrics defined for the practical test as indicated in the Table 1.
The open-ended question set for CIE shall be any other open-ended question and not selected from the experiments under PART-A. It shall be evaluated for 10 marks. The allotment of marks shall be as per rubrics listed in Table 2

Table – 1 Distribution of marks corresponding to laboratory component		
Component	**Description**	**Marks**
Allotment of marks for regular classwork		
Ability in Conducting Experiments	Initiative, skill, safety practices, teamwork, and independent handling of equipment during regular lab sessions.	10 marks
Laboratory Record / Journal	Regular and neat maintenance of lab record, accuracy of results, and timely submission.	15 marks
	Subtotal	**25**
Allotment of marks for CIE practical test		
Laboratory Test / Experiment Performance	Ability to set up apparatus, follow procedure, take observations, and obtain correct results in the test experiment.	5 marks
Viva-Voce	Understanding of theory, concepts, and experimental procedure; ability to explain results and answer related questions.	10marks
	Subtotal	**15**
	Total	**40**

Table 2 Distribution of marks corresponding to PART -B of the laboratory component				
Evaluation Rubrics for continuous internal evaluation CIE). Maximum marks:10				
Criteria and Guiding factors	**Students' Level and Rubrics**			
	Excellent: E, Good: G, Fair: F Needs Improvements: NI			
	E	**G**	**F**	**NI**
(1) Problem Understanding **Criteria:** Is the problem understood correctly and framed appropriately? (i)Clarity in interpreting the given problem. (ii)Identification of relevant concepts and constraints. (iii) Ability to rephrase or model the problem logically.	2.0	1.5	1.0	0.5
(2) Conceptual Knowledge **Criteria**: Are the chosen methods or concepts valid for this type of problem? (i)Application of appropriate theories, principles, or laws. (ii)Integration of multiple concepts where necessary. (iii)Accuracy in stating relevant equations or models.	2.0	1.5	1.0	0.5
(3) Approach and Strategy **Criteria**: Is the approach well thought out and methodical? (i)Logical structure and progression. (ii)Use of diagrams, sketches, or flowcharts if applicable. (iii)Identification of assumptions and boundary conditions.	2.0	1.5	1.0	0.5
(4) Execution and Accuracy **Criteria**: Are the calculations correct? Is the final solution reasonable? (i) Correct application of formulas/methods. (ii)Accuracy of calculations or simulations (if any). (iii)Unit consistency and result verification.	2.0	1.5	1.0	0.5
(5) Innovation or Depth **Criteria**: Has the student shown original thinking or alternative methods? (i)Novel or efficient solutions. (ii)Consideration of alternate approaches. (iii)Interdisciplinary integration or practical extensions.	1	0.75	0.5	0.25
(6) Presentation and Clarity **Criteria**: Is the answer easy to follow and professionally written? (i)Organization and neatness. (ii)Clarity of explanation, labelling, and steps.	1	0.75	0.5	0.25

(iii) Use of proper technical terminology.				
Total	10	7.5	5.0	2.5

Note:
(i)The marks shown for Excellent, Good, Fair, Needs Improvement are indicative. The evaluator may also assign intermediate scores depending on the quality of work.
(ii)If the total score results in a fraction, it should be rounded off to the next higher integer.
(iii)The listed criteria are guidelines; evaluators may also consider other authentic dimensions (e.g., depth of insight, critical thinking, and practical awareness, recognition of limitations, or justified compromises.
(iv)To award the full10 marks in the excellent category, the student must demonstrate reasonably good performance across all six criteria; otherwise, proportionally lower marks may be given.

****