DAYANANDA SAGAR UNIVERSITY

PROGRAMME PROJECT REPORT (PPR)

FOR

MASTER OF COMPUTER APPLICATIONS (MCA)

Mode: ONLINE

CENTRE FOR DISTANCE AND ONLINE EDUCATION (CDOE)

DAYANANDA SAGAR UNIVERSITY

BENGALURU

Dayananda Sagar University Bangalore

28/8/25

Registrar

List of Contents

PROGRAMME PROJECT REPORT

Introduction

Technology, Business, and Society have converged like never before. Today's ethos is to develop technological solutions that generate value ethically, drive innovation, and address complex societal challenges. At Dayananda Sagar University (DSU), we recognize the significance of this convergence and have designed our programs to equip students with the skills and knowledge to navigate the dynamic technological landscape.

The Online Master of Computer Applications (MCA) Programme offered from the Centre for Distance and Online Education (CDOE), Dayananda Sagar University imparts knowledge and skill sets to students to achieve this and face real-world technological challenges. It teaches the application of innovative computing practices to current technological situations. It incorporates analysis of contemporary technological issues besides providing a strong theoretical foundation. It provides a collaborative learning environment with dedicated faculty to ensure students achieve their full potential. The online MCA Programme teaches one to work smartly, take the lead in critical technological scenarios, and influence digital transformation more effectively. It teaches one to not only work efficiently but also shape the technological environment and create opportunities for further innovation.

The Online Master of Computer Applications (MCA) Programme is a two-year programme. With the inclusion of various electives, the Programme aims at developing focused technology professionals with a strong understanding of their area of expertise, even as the core subjects equip the students with fundamental computer science theories and concepts along with essential soft skills. The Electives are offered in Artificial Intelligence, Cloud Computing, Cybersecurity, Data Science, Software Engineering, Network Technologies and Mobile Application Development. These help the student emerge as a well-rounded technology professional ready to take up challenges in diverse technological domains from software development to digital infrastructure management. The combination of courses is designed to foster a comprehensive approach and train students to take on leadership roles in global technology ecosystems by providing dual functional electives.

The programme's ethos is to craft technology professionals who can problem-solve and make data-driven decisions, allowing students to develop excellent skills in various computing areas. In a

world that has turned increasingly digital, our programme equips young technology professionals with the skills and knowledge needed to succeed in the digital age. Our online MCA Programme will give students the perfect start to an outstanding career in the ever-evolving world of technology.

1. Programme's Mission and Objectives

The mission of the online MCA Programme from CDOE, DSU firmly believes that theoretical knowledge supported by ample in-depth practical exposure is required to meet the challenges of this ever-changing field. As the world looks towards India for fresh and energetic talent in the field of IT, the Department's mission is to shape the next generation IT professionals with concrete foundation, employable skills along with a touch of human ethical values.

Mission

To emerge as power house of Information Technology and Allied areas developing competent computer professionals to meet the dynamic needs of disruptive technologies.

Objectives

- To create a platform of learning opportunities for students and to enable them to take careers in the technology sector, public and social fields, research and academics.
- To effectively impart to the students the fundamental computer science concepts and principles.
- To enhance the students technical skills, critical thinking skills, and implementation of reasoning and problem-solving models in technological domains.

The objectives of the online MCA programme focus on providing diverse learning opportunities to prepare students for careers across the technology sector, public & social sectors, and also in research and education. The programme emphasizes imparting computer science principles, fostering critical thinking, technical skills, and problem-solving abilities for professional and personal growth.

The programme provides students with a thorough understanding of both regional and international technology landscapes, promoting innovative approaches to technology and strategic decision-making. It emphasizes individual development through reflective learning from projects and internships, builds analytical expertise in key computing domains, and prepares students to excel

in advanced roles within the technology sector. By integrating technological acumen, technical skills, and digital fluency, the programme nurtures industry-ready professionals capable of driving technological innovation and leading teams across organizational boundaries.

The outcomes from the programme focus on developing technical skills, equipping students to thrive in a global digital environment with the right attitude and expertise. Graduates are trained to evaluate technological landscapes, devise strategies for complex technological challenges, and adapt to rapid technological changes. The programme emphasizes ethical principles of technology development, collaborative teamwork, and professional responsibility, alongside effective communication with diverse stakeholders in technological ecosystems. It nurtures technological entrepreneurship and project management abilities, fostering a comprehensive understanding of computer science principles and preparing students to manage technological challenges and opportunities effectively.

2. Relevance of Programme with Dayananda Sagar University Mission and Goals

The Vision of Dayananda Sagar University emphasizes on becoming a leading institution excelling in education, research, innovation, and entrepreneurship. It aims to nurture individuals with exceptional leadership skills, empowering them to address national and global challenges, driving positive change, and contributing to societal development. The focus is on excellence and holistic development to shape future leaders.

The mission statement of Dayananda Sagar University highlights fostering a supportive environment that encourages creativity, innovation, and academic excellence. By aligning all efforts with the institution's vision, it aims to achieve its goals while nurturing intellectual growth, inspiring new ideas, and maintaining a commitment to excellence in every pursuit.

Vision

To be a centre of excellence in education, research & training, innovation & entrepreneurship and to produce citizens with exceptional leadership qualities to serve national and global needs.

Mission

To achieve our objectives in an environment that enhances creativity, innovation and scholarly pursuits while adhering to our vision.

DSU's online MCA programme supports the vision by providing a dynamic learning environment

that encourages technological creativity and innovation. It integrates cutting-edge research, scholarly activities, and practical training to develop technology leaders and innovators who excel in addressing national and global technological challenges. The programme fosters excellence and aligns with the broader institutional aspirations of technological advancement and digital transformation. The programme also reflects the mission of the University by promoting an ecosystem of intellectual growth and technological innovation. It enables students to achieve their goals through a curriculum that emphasizes creativity and problem-solving while cultivating the technical leadership and expertise required to meet the evolving technological and industry needs.

3. Nature of Prospective Target Group of Students

This Online MCA programme from CDOE has been designed for students who may be graduates, mid-career technology professionals, working IT professionals, software developers, system administrators and individuals seeking to update their technical knowledge and transition into advanced technological roles. The programme will offer flexibility, accessibility, and affordability for students to balance their studies along with professional and personal commitments.

The programme is designed such that prospective can get the opportunity to

access high value learning anytime from anywhere and pursue the programme at one's own pace. The programme also fosters professional networking and eliminates the need for relocation or commuting thus making it ideal for diverse student groups. Delivery of the programme through the online mode also contributes towards the aim of India's National Education Policy to achieve a Gross Enrolment Ratio (GER) of 50% by the year 2035.

4. Appropriateness of programme to be conducted in Online mode to acquire specific skills and competence

To enable the students of the online MCA programme to learn, the courses in the programme are delivered through Self-Learning e-Modules, offering a structured and engaging approach to e-learning. These modules are designed to be self-explanatory, providing clear instructions and content to facilitate independent learning. They are self-contained, encompassing all necessary resources within a single unit, and self-directed, allowing students to progress at their own pace. The modules include self-evaluation components to enable students to assess their understanding and progress. This approach ensures students can effectively acquire the prescribed knowledge and

skills while enjoying flexibility, accessibility, and a personalized learning experience tailored to their needs. The course study material for the online programme include are made available through the four-quadrant approach which will be uploaded on the University's Learning Management System (LMS). The components of the four quadrants are:

(i) e-Tutorial: in the form of recorded faculty lecture videos.

(ii) e-Content: Text Materials – e-SLM and access to online library for additional reading materials.

(iii) Discussion forum: For raising of doubts and clarifying the same by the Course Coordinators/Course Mentors assigned to students

(iv) Assessments: Self-Assessments in the form of quizzes, fill in the blanks, matching questions, short answer questions, internal assessments in the form of assignments to reinforce learning.

The online MCA programme from CDOE is thus designed to align perfectly with the online learning format, ensuring effective delivery and engagement. Its structure and content cater to the needs of students learning in the online mode hence making it a suitable choice for those seeking a flexible and accessible higher education opportunity.

5. Instructional Design

5.1 Programme Curriculum

The curriculum of the online MCA Programme has been crafted by computer science and technology experts, thoughtfully incorporating contemporary computing and technological topics alongside courses fostering contemporary technological awareness. The balanced approach in the curriculum design has been done to ensure that students gain modern insights while developing a sense of responsibility toward sustainable technological practices, equipping them with the knowledge and values essential for navigating the contemporary dynamic technological landscape. The curriculum and syllabus are approved by the Board of Studies as per University norms. This ensures the programme meets high-quality standards and aligns with current academic and professional requirements.

MASTER OF COMPUTER APPLICATIONS - MCA

SEM	SN.	COURSE CODE	TITLE OF THE COURSE	CREDITS
1	1	MCAOL101	ADVANCED DATA STRUCTURES AND ALGORITHMS	3
1	2	MCAOL102	ADVANCED PYTHON PROGRAMMING	3
1	3	MCAOL103	ADVANCED OPERATING SYSTEMS	4
1	4	MCAOL104	RELATIONAL DATABASE MANAGEMENT SYSTEMS	3
1	5	MCAOL105	MATHEMATICAL METHODS	4
1	6	MCAOL106	ADVANCED DATA STRUCTURES AND ALGORITHMS LAB	2
1	7	MCAOL107	ADVANCED PYTHON PROGRAMMING LAB	2
1	8	MCAOL108	RDBMS LAB	2
				23

SEM	SN.	COURSE CODE	TITLE OF THE COURSE	CREDITS
2	9	MCAOL201	MACHINE LEARNING WITH PYTHON	3
2	10	MCAOL202	ADVANCED JAVA PROGRAMMING	3
2	11	MCAOL203	DESIGN AND ANALYSIS OF ALGORITHMS	4
2	12	MCAOL204	FULL STACK DEVELOPMENT	3
2	13	MCAOL205	COMPUTER COMMUNICATION NETWORK	3
2	14	MCAOL206	ADVANCED JAVA PROGRAMMING LAB	2
2	15	MCAOL207	MACHINE LEARNING WITH PYHTON LAB	2
2	16	MCAOL208	FULL STACK DEVELOPMENT LAB	2
2	17	MCAOL209X	ELECTIVE – I	4
				26

	ELECTIVE - I
MCAOL2091	CLOUD COMPUTING TECHNOLOGIES
MCAOL2092	BIGDATA ANALYTICS
MCAOL2093	DATA SCIENCE
MCAOL2094	QUANTUM ALGORITHMS
MCAOL2095	BLOCKCHAIN TECHNOLOGIES

SEM	SN.	COURSE CODE	TITLE OF THE COURSE	CREDITS
3	18	MCAOL301	CYBER SECURITY	4
3	19	MCAOL302	EMBEDDED SYSTEMS AND INTERNET OF THINGS	3
3	20	MCAOL303X	ELECTIVE – II	4
3	21	MCAOL304X	ELECTIVE – III	4
3	22	MCAOL305X	ELECTIVE – IV	4
3	23	MCAOL306	DESIGN AND ANALYSIS OF ALGORITHMS LAB	2
3	24	MCAOL307	MINI PROJECT	4
				25

	ELECTIVE - II
MCAOL3031	CLOUD SECURITY & INFRASTRUCTURE
MCAOL3032	DATA ANALYTICS & VISUALIZATION
MCAOL3033	ARTIFICIAL INTELLIGENCE
MCAOL3034	QUANTUM INFORMATION
MCAOL3035	RESEARCH METHODOLOGY & IPR

	ELECTIVE - III
MCAOL3041	IOT
MCAOL3042	DATA MINING
MCAOL3043	DEEP LEARNING
MCAOL3044	QUANTUM ERROR CORRECTION
MCAOL3045	MOBILE APPLICATION DEVELOPMENT TECHNIQUES

	ELECTIVE - IV
MCAOL3051	INDUSTRIAL IOT AND AUTOMATION
MCAOL3052	BUSINESS INTELLIGENCE TECHNOLOGY
MCAOL3053	NATURAL LANGUAGE PROCESSING
MCAOL3054	QUANTUM MACHINE LEARNING
MCAOL3055	SOFTWARE PROJECT MANAGEMENT

SEM	SN.	COURSE CODE	TITLE OF THE COURSE	CREDITS
4	25	MCAOL401X	GENERAL ELECTIVE	4
4	26	MCAOL402	PROJECT	6
				10
			Total Program Credits	**84**

	GENERAL ELECTIVE
MCAOL4011	BUSINESS DATA ANALYTICS
MCAOL4012	INDUSTRIAL SAFETY
MCAOL4013	OPERATIONS RESEARCH

5.2 Programme Detailed Syllabus

Semester: I

Course Name: ADVANCED DATA STRUCTURES AND ALGORITHMS

Credits: 3

Course Description:

This course focuses on fundamental and advanced data structures, including arrays, stacks, queues, linked lists, trees, heaps, and hashing techniques, along with their operations and applications. It also covers advanced algorithms for sorting, graph traversals, shortest paths, and minimum spanning trees, emphasizing their implementation and efficiency. By bridging theory and practical applications, the course equips learners with essential skills for solving complex computational problems.

Course Objectives:

This Course will enable learners to:

- Introduce dynamic memory allocation and C language concepts required for building data structures
- Develop essential skills to construct data structures to store and retrieve data quickly and efficiently.
- Usage of different data structures that support different sets of operations which are suitable for various applications.
- Implement how to insert, delete, search and modify data in any data structure- Stack, Queues, Lists, Trees.

Course Outcomes:

At the end of the course the learner will be able to:

- Define abstract data type and apply to implement list, stack and queue applications
- Demonstrate the concepts of trees, Binary tree, expression tree, Binary search tree and organise the data with applications
- Explain hash functions, open, closed addressing types and rehashing Discuss sorting algorithms - insertion, shell, quick, Heap, Merge.
- Define graph terminologies and Outline applications using Prim’s, Kruskal’s algorithms and relate to real time projects

Unit 1	Introduction to Linear Data Structures: Abstract Data Types (ADTs), Time and Space Complexities, List ADT – Array-Based Implementation, Linked List Implementation,
Unit 2	Advanced Linear Data Structures: Doubly Linked Lists, Circular Linked Lists, Stack ADT: Implementation of Stacks, Queue ADT: Applications of ADT
Unit 3	Introduction to Trees: Trees: Preliminaries, Implementation of Trees, Tree Traversals with an Application, Binary Trees: Implementation
Unit 4	Advanced Trees: Expression Trees, Search Tree ADT, Binary Search Trees, Introduction to AVL and Red-Black Trees, Applications of Trees
Unit 5	Binary and Min-Max Heaps: Binary Heaps, Min-Max Heaps
Unit 6	Advanced Heaps: Leftist Heaps, Binomial Heaps, Fibonacci Heaps, Merging for Binomial Queues, Fibonacci Heap Operations
Unit 7	Hashing Techniques: Fundamentals of Hashing, Hash Functions, Separate Chaining, Open Addressing, Linear Probing, Quadratic Probing, Double Hashing, Rehashing, Extendible Hashing
Unit 8	Advanced Sorting Algorithms: Insertion Sort, Shell Sort, Quick Sort, Heap Sort, Merge Sort,
Unit 9	Graph Representations and Traversals: Representation of Graphs, Graph Traversals, Topological Sort
Unit 10	Shortest Paths and Minimum Spanning Trees: Shortest Path Algorithms: Dijkstra's Algorithm, Graphs with Negative Edge Costs, All Pairs Shortest Path, Minimum Spanning Trees: Prim's and Kruskal's Algorithm

TEXTBOOKS:

- Data structures and applications by Dr. S Nandagopalan
- A.M. Tannenbaum, Y Langsam, M J Augentien “Data Structures using C”, 1st Edition, Pearson, 2019.
- Ellis Horowitz, Susan Anderson-Freed, and Sartaj Sahni, “Fundamentals of Data structures in C”, 2ndEdition, Orient Longman, 2008.
- Brian.W.Kernighan, Dennis.M.Ritchie, “The C Programming Language”, 2ndEdition, Prentice-Hall, 1988.

REFERENCE BOOKS:

- Gilberg & Forouzan, “Data Structures: A Pseudo-code approach with C”, 2ndEdition, Cengage Learning,2014
- Brian W. Kernighan, Dennis Ritchie, “The C Programming Language”, Second Edition,Pearson Education, 2015.
- Brian W. Kernighan, Rob Pike, “The Practice of Programming”, Pearson Education,1999.
- Mark Allen Weiss, “Data Structures and Algorithm Analysis in C”, Second Edition, Pearson Education, 1997.

Course Name: ADVANCED PYTHON PROGRAMMING

Credits: 3

Course Description:

This course introduces Object-Oriented Programming (OOP) in Python, covering core concepts such as classes, objects, inheritance, and polymorphism, along with exception handling. It explores Python's rich ecosystem, including modules like NumPy and Pandas for data manipulation, Matplotlib and Seaborn for data visualization, and libraries for web scraping and file handling. Learners will also learn graphical user interface (GUI) development using Tkinter and advanced frameworks like Flask and Django for web applications. By combining theory and practical implementation, the course prepares learners to build robust, data-driven, and interactive applications.

Course Objectives:

This Course will enable learners to:

- Implement essential operations using Numpy and Pandas module
- Analyse data visualization using Matplotlib and Seaborn
- Implement GUI in Python
- Implement GUI using different platforms like Flask and Django

Course Outcomes:

At the end of the course the learner will be able to:

- Articulate the Object-Oriented Programming concepts such as encapsulation, inheritance and polymorphism as used in Python.
- Apply the fundamentals of Numpy and Pandas in real world applications.
- Analyse data using Python's data visualization and Understand file handling.
- Develop Graphical User Interface using Tkinter.

Unit 1	Advanced OOP in Python: OOP Concepts, Classes and Objects, Class Variables, Inheritance, Types of Inheritance, Polymorphism, Operator Overloading, Type Identification
Unit 2	Iterators, Generators, Decorators, and Exception Handling: Iterators, Generators, Decorators, Errors and Exceptions, Exception Types, Exception Handling using try & except, User defined Exceptions

Unit 3	NumPy and Pandas: NumPy Arrays, Computation on NumPy Arrays, Universal Functions
Unit 4	Pandas for Data Analysis: Pandas Series and Data Frame, Summarizing and Computing Descriptive Statistics, Handling Missing Data, Data Wrangling with Pandas
Unit 5	Data Visualization with Matplotlib and Seaborn: Introduction to Matplotlib, Plot Types: Scatter Plot, Line Plot, Bar Chart, Histogram, Box Plot, Multiple Subplots, Visualization with Seaborn
Unit 6	Web Scraping and File Handling: Web Scraping with Beautiful Soup, File Handling in Python, Kinds of Files, Opening a File, Reading Files, Writing Files
Unit 7	Basics of GUI Programming with Tkinter: Basics of Graphical User Interface, Tkinter Module, Widgets
Unit 8	Advanced GUI Concepts: Models, Views, and Controllers, Customizing the Visual Style, Object-Oriented GUIs, Concept of GUI Mess
Unit 9	Introduction to Flask and Framework: Introduction to Flask, Flask Framework, Building Web Applications with Flask
Unit 10	Introduction to Django and Case Study: Introduction to Django, Django Framework, Case Study Using Flask and Django

TEXTBOOKS:

- Downey, A. B. (2015). Think Python: How to think like a computer scientist (2nd ed.). O'Reilly Media.
- McKinney, W. (2017). Python for data analysis: Data wrangling with pandas, numpy, and ipython (2nd ed.). O'Reilly Media.
- Grinberg, M. (2018). Flask web development: Developing web applications with Python (2nd ed.). O'Reilly Media.

REFERENCE BOOKS:

- Holovaty, A., & Kaplan-Moss, J. (2009). The Django book (2nd ed.). The Django Software Foundation.
- Mark Lutz, "Programming Python", O'Reilly Media, 4th edition, 2010.

Course Name: ADVANCED OPERATING SYSTEMS

Credits: 4

Course Description:

This course provides an in-depth understanding of operating systems, covering their evolution, classifications, and architectures, including monolithic, microkernel, and hybrid structures. It explores key concepts such as process management, interprocess communication, and advanced topics like socket programming and distributed computing. Additionally, the course examines multiprocessor systems, synchronization techniques, and performance optimization. Through theory and practical implementation, learners gain insights into how operating systems manage resources, enable communication, and support modern computing environments.

Course Objectives:

This Course will enable learners to:

- Illustrate the categories and structure of Operating Systems.
- Explain key aspects of process control
- Illustrate concepts related to client/server computing and Network IPC
- Describe the Architecture of Multiprocessor Operating system
- Demonstrate different IPC mechanism with use case.

Course Outcome:

At the end of the course the learner will be able to:

- Define and understand the categories and structure of Operating Systems
- Explain key aspects to ensure effective management and control over processes running within the system of process control
- Illustrate concepts related to client/server computing and Network IPC
- Describe the Architecture of Multiprocessor Operating system with respect to efficient utilization and management multiple processors
- Demonstrate different IPC mechanism with their use case

Unit 1	Introduction to Operating Systems: Classes of Operating Systems, Efficiency, System Performance, User Service, Batch Operating Systems, Multiprogramming Systems, Time Sharing Systems, Real Time Operating Systems, Distributed Operating Systems
Unit 2	Structure and Operations of OS: Operations of an Operating System, Structure of Operating Systems, Monolithic Structure, Layered Design of OS, Command-Line Interface (CLI) and Graphical User Interface (GUI), Types of System Calls (Process Control, File Management, Device Management, Information Maintenance, Communications)
Unit 3	Processes and Programs: Process Management: Process Model, Process Creation and Termination, Process Hierarchies, Process States, Process Implementation, Definition of a Process, Process Lifecycle and Control Blocks, Parent and Child Processes, Process States (New, Ready, Running, Waiting, Terminated), Context Switching and Implementation Details Process Identifiers, fork() Function, wait() and waitpid() Functions, Race Conditions, exec() Functions, Changing User IDs and Group IDs, Security Implications
Unit 4	Process Scheduling and Case Studies: Algorithm Evaluation, CPU Scheduling Objectives and Criteria (CPU Utilisation, Throughput, Turnaround Time, Waiting Time, Response Time), Scheduling Algorithms: First-Come-First-Served (FCFS), Shortest Job First (SJF), Priority Scheduling, Round Robin (RR), Multilevel Queue, Multilevel Feedback Queue, Thread Scheduling Issues, Multiprocessor Scheduling Techniques, Comparative Evaluation of Scheduling Algorithms, Case Studies, Linux Operating System
Unit 5	Threads and Concurrency: Multithreading Models, Thread Libraries, Threading Issues, Threading in Java, Classic Synchronization Problems, User-Level vs. Kernel-Level Threads, Multithreading Models (Many-to-One, One-to-One, Many-to-Many), POSIX Threads, Windows Threads, Java Threads, Thread Cancellation, Signal Handling, Thread Pools, Implementation of Threads in Java, Dining Philosophers Problem, Readers-Writers Problem, Programming Solutions to Synchronization Problems
Unit 6	Process Synchronisation: Race Conditions, Critical Sections, Mutual Exclusion, Peterson's Solution, Synchronization Hardware, Semaphores, Mutexes, Monitors, Message Passing, Atomic Transactions,, Issues with Concurrent Processes Critical Section Problem and Requirements, Software Solutions to Mutual Exclusion (Peterson's Algorithm), Hardware Support for Synchronization (Test-and-Set, Swap), Semaphore Operations and Usage, Mutex Locks and Their Applications, Monitor Constructs for Synchronisation, Inter process Communication via Message Passing, Ensuring Atomicity in Transactions
Unit 7	Inter Process Communication (IPC): Introduction to IPC, Advantages and Disadvantages of IPC, real – Time examples, Pipes, popen() and pclose() Functions Message Queues: Concept of Message Queues, Structured Message Communication, Creating Queues with msgget(), Sending Messages with msgsnd(), Receiving Messages with msgrcv(), Managing Queues with msgctl()

Unit 8	Memory management : Main Memory Management, Swapping, Contiguous Memory Allocation, Paging, Structure of Page Table, Segmentation, Examples, Memory Allocation Strategies (Fixed Partitioning, Dynamic Partitioning), Swapping Processes In and Out of Memory, Contiguous Allocation Methods, Paging Concepts and Page Table Structures (Hierarchical Paging, Hashed Page Tables, Inverted Page Tables), Segmentation and its Implementation, Combining Paging and Segmentation, Practical Examples of Memory Allocation, Semaphores and Shared Memory
Unit 9	Virtual Memory and Advanced Memory Management: Virtual Memory, Demand Paging, Copy-on-Write, Page Replacement Algorithms, Allocation of Frames, Thrashing, Memory-Mapped Files, Allocating Kernel Memory, Benefits of Virtual Memory Systems, Implementation of Demand Paging, Copy-on-Write Optimization, Page Replacement Algorithms (FIFO, LRU, Optimal, Second Chance), Frame Allocation Strategies (Equal, Proportional, Priority-Based), Causes and Solutions for Thrashing, Use of Memory-Mapped Files for File I/O, Kernel Memory Allocation Techniques (Buddy System, Slab Allocator), Memory Management Utilities
Unit 10	Virtualisation: Requirements for Virtualization, Type 1 Hypervisors, Type 2 Hypervisors, Paravirtualization, Memory Virtualization, I/O Virtualization, Virtual Appliances, Virtual Machines on Multicore CPUs, Licensing Issues, Case Study: Windows, Utilities for Monitoring and Managing Memory, Fundamental Concepts of Virtualization, Role and Functionality of Hypervisors, Comparison between Type 1 (Bare Metal) and Type 2 (Hosted) Hypervisors, Principles of Paravirtualization, Techniques for Virtualizing Memory and I/O Devices, Deployment of Virtual Appliances, Managing Virtual Machines on Multicore Processors, Legal and Licensing Considerations in Virtualization, Application of Virtualization in Windows Operating System
Unit 11	Multiprocessor Systems Architecture: Understanding Multiprocessor Systems, Advantages of Multiprocessor Systems, Classification of Multiprocessor Systems, Interconnections in Multiprocessor Systems, Types of Multiprocessor Operating Systems
Unit 12	Kernel Structure and Synchronization: Introduction, Kernel Structure in Multiprocessor OS, Process Synchronization
Unit 13	Network IPC and Socket Programming: Sockets, Socket Descriptors – Advantages and disadvantages- Realtime examples Addressing, Connection Establishment, Data Transfer, Socket Options, Advantages and disadvantages
Unit 14	Client/Server Computing: Distributed Message Passing, Remote Procedure Calls (RPC), Clusters, Windows Vista Cluster Server

TEXTBOOKS:

- Dhananjay M. Dhamdhere, Operating System, A concept-based approach 3rd edition, McGrawHill Education Pvt. Ltd. 2012
- W. Richard Stevens, Stephen A.Rago, Advanced Programming in the UNIX Environment, 3rd Edition, Pearson,2016
- William Stallings: Operating Systems: Internals and Design Principles, 7th Edition, Prentice Hall, 2015.

REFERENCE BOOKS:

- Unix-Concepts and Applications, Sumitabha Das, 4th Edition, Tata McGraw Hill, 2015
- "Operating System Concepts" by Abraham Silberschatz, Greg Gagne, and Peter B. Galvin, Wiley,2018
- "Modern Operating Systems" by Andrew S. Tanenbaum and Herbert Bos, Pearson, 2014

Course Name: RELATIONAL DATABASE MANAGEMENT SYSTEMS
Credits: 3

Course Description:

This course provides a comprehensive understanding of database systems, covering fundamental concepts like data models, relational algebra, SQL, and normalization techniques. It delves into advanced topics such as transaction management, concurrency control, recovery systems, and indexing strategies for efficient query processing. Learners will explore distributed and XML databases, alongside practical programming with PL/SQL, including stored procedures, triggers, and cursors. By combining theoretical foundations with hands-on applications, this course equips learners with the knowledge to design, manage, and optimize modern database systems effectively.

Course Objectives:

This Course will enable learners to:

- Learn the fundamentals of data models, conceptualize and depict a database system using ER diagrams.
- Study the principles to be followed to create an effective relational database and write SQL queries to store/retrieve data to/from database systems.
- Infer the fundamental concepts of transaction processing, concurrency control techniques and recovery procedure.
- Articulate the internal storage structures using different file and indexing techniques and the basics of query processing and optimization.

Course Outcome:

At the end of the course the learner will be able to:

- Apply to Model an application's data requirements using conceptual modelling and design database schemas based on the conceptual model
- Formulate solutions to a broad range of query problems using relational algebra/SQL
- Demonstrate an understanding of normalization theory and apply such knowledge to the normalization of a database
- Explain basic database storage structures, access techniques and query processing

Unit 1	Introduction to Relational Databases: Purpose of Database Systems, Views of Data, Data Models, Database System Architecture, Introduction to Relational Databases, Relational Model, Keys, Need for Database Systems, Data Abstraction Levels (Physical, Logical, View), Hierarchical, Network, and Relational Data Models, Components of DBMS Architecture, Concepts of Relations, Attributes, Tuples, Primary Keys, Foreign Keys, Candidate Keys, Super Keys
Unit 2	Relational Algebra, Calculus, and SQL: Relational Algebra, Relational Calculus, SQL Fundamentals, Advanced SQL Features, Triggers, Embedded SQL, Basic Operations in Relational Algebra (Select, Project, Union, Set Difference, Cartesian Product, Rename), Tuple Relational Calculus and Domain Relational Calculus, SQL Syntax and Commands (SELECT, INSERT, UPDATE, DELETE), Advanced SQL (Joins, Subqueries, Views, Indexes), Implementing Triggers in SQL, Using Embedded SQL in Application Programs
Unit 3	ER Modelling and Functional Dependencies: Entity-Relationship Model, ER Diagrams, Functional Dependencies, Non-Loss Decomposition, Entities, Attributes, Relationships, Symbols and Notations in ER Diagrams, Definition and Importance of Functional Dependencies, Decomposition of Relations for Lossless Join and Dependency Preservation
Unit 4	Normalization and Normal Forms: First Normal Form (1NF), Second Normal Form (2NF), Third Normal Form (3NF), Boyce-Codd Normal Form (BCNF), Multivalued Dependencies, Fourth Normal Form (4NF), Join Dependencies, Fifth Normal Form (5NF), Techniques for Normalizing Relations, Eliminating Redundancy, Preserving Dependencies, Handling Multivalued Dependencies, Advanced Normal Forms and Their Applications
Unit 5	Transaction Concepts and Concurrency Control: Transaction Concepts, ACID Properties, Serializability, Transaction Isolation Levels, Concurrency Control, Lock-Based Protocols, Deadlock Handling, Definition of Transactions, Atomicity, Consistency, Isolation, Durability, Schedules and Serializability, Isolation Levels (Read Uncommitted, Read Committed, Repeatable Read, Serializable), Locks (Shared, Exclusive), Two-Phase Locking Protocol, Deadlock Detection and Prevention
Unit 6	Recovery System: Failure Classification, Recovery Algorithms, Types of Failures (Transaction, System, Media), Log-Based Recovery, Checkpointing, Deferred Update and Immediate Update, Shadow Paging
Unit 7	Storage and File Organization: Overview of Physical Storage Media, RAID, File Organization, Organization of Records in Files, Types of Storage Media (Magnetic Disks, SSDs), RAID Levels and Concepts, Sequential, Heap, and Hash Files, Fixed-Length and Variable-Length Records
Unit 8	Indexing and Query Processing: Indexing and Hashing, Ordered Indices, B+ Tree Index Files, Static Hashing, Dynamic Hashing, Query Processing Overview, Catalogue Information for Cost Estimation, Query Optimization, Primary and Secondary Indices, Structure of B+

	Trees, Hash Functions and Buckets, Steps in Query Processing, Estimating Query Costs, Algorithms for Query Optimization
Unit 9	Distributed and XML Databases: Overview of Distributed Databases, Data Fragmentation, Replication, XML, Databases, XML Schema, Distributed Database Architectures, Horizontal and Vertical Fragmentation, Data Replication Strategies, Storing Data in XML Format, Defining XML Schemas
Unit 10	PL/SQL Programming and Database Integrity: PL/SQL Programming: Stored Procedures, Triggers, Cursors, Database Integrity, Integrity Rules, Domain Rules, Attribute Rules, Relation Rules, Database Rules, Assertions, Triggers, Integrity and SQL, Writing PL/SQL Code, Creating and Using, Stored Procedures and Functions, Implementing Triggers for Automated Actions, Using Cursors for Data Retrieval, Enforcing Integrity Constraints, Implementing Business Rules in SQL

TEXTBOOKS:

- Abraham Silberschatz, Henry F. Korth, S. Sudharshan, "Database System Concepts", Sixth Edition, Tata McGraw Hill, 2014.
- Ramez Elmasri, Shamkant B. Navathe, "Fundamentals of Database Systems", Seventh Edition, Pearson Education, 2017.

REFERENCE BOOKS:

- C. J. Date, A. Kannan, S. Swamynathan, "An Introduction to Database Systems", Eighth Edition, Pearson Education, 2006.
- Raghu Ramakrishnan, Johannes Gehrke, "Database Management Systems", Fourth Edition, Tata McGraw Hill, 2010.
- G. K. Gupta, "Database Management Systems", Tata McGraw Hill, 2011.
- Carlos Coronel, Steven Morris, Peter Rob, "Database Systems: Design, Implementation and Management", Ninth Edition, Cengage Learning, 2011.

Course Name: MATHEMATICAL METHODS

Credits: 4

Course Description:

This course offers a detailed exploration of probability and linear algebra concepts, essential for data science and engineering applications. It covers probability theory, distributions, and hypothesis testing, providing tools for statistical inference. Additionally, the course delves into linear algebra, including vector spaces, matrix decomposition, and analytical geometry, with a focus on practical problem-solving. Advanced topics such as vector calculus and continuous optimization are also addressed, equipping learners with the mathematical foundations needed for complex computational and analytical tasks.

Course Objectives:

This Course will enable learners to:

- Apply statistical methods in data science and machine learning, understand number theoretic cryptography schemes and apply graph theoretic concepts to design algorithms.
- Provide advanced mathematical tools needed in machine learning, data science, cyber security and algorithm design.
- Equip the learners with a working knowledge of probability, statistics, and modelling in the presence of uncertainties.
- Apply mathematics to machine learning and data science
- Develop an intuition and an interest for random phenomena and to introduce both theoretical issues and applications that may be useful in real life.

Course Outcome:

At the end of the course the learner will be able to:

- Understand the application of probability in computer science and information technology.
- Apply probability concepts to solve problems related to algorithms, data structures, and computer systems.
- Perform statistical analysis on data sets using appropriate techniques.
- Interpret the results of statistical analyses in the context of real-world problems.

- Vector calculus and matrix decomposition are applied in MCA for optimizing machine learning algorithms and performing efficient data transformations.

Unit 1	Introduction to Probability: Introduction, Definitions, Empirical definition of probability, Theorems on probability, Addition Theorem, Axioms of probability, Conditional probability, Baye's Theorem
Unit 2	Probability Distributions – I: Random Variables- Introduction, Discrete and continuous random variables-introduction, Discrete Probability Distribution, Continuous probability distributions
Unit 3	Probability Distributions – II: Binomial Distribution, Poisson Distribution, Geometric Distribution, Uniform Distribution, Exponential Distribution
Unit 4	Probability Distributions – III: Joint probability distribution(discrete), Marginal Functions Stochastic Independence, Mean, Variance, Covariance and Correlation
Unit 5	Testing of Hypotheisis – I: Introduction, Random sampling, Sampling Distributions, Sampling Distribution of Means, Testing of Hypothesis Type I and Type II Errors, Significance Level
Unit 6	Testing of Hypotheisis – II: One tailed tests, Two tailed tests, Learners' t- distribution Chi-square distribution
Unit 7	Linear Algebra: Some preliminaries, Elementary transformations, Gauss elimination method, Vector spaces, Linear Dependence and Independence, Rank and Nullity
Unit 8	Analytical Geometry: Norms, Inner Products, Orthogonality, Orthonormality, Inner product of functions

Unit 9	Matrix Decomposition: Determinatns and Trace, Eigenvalues and Eigenvectors, LU Decomposition method, Singular value decomposition
Unit 10	Vector Calculus – I: Vector function of a single variable, Scalar and vector point function, Parital differentiation, Gradient of a scalar function, Gradient of Matrices
Unit 11	Vector Calculus – II: Vector identities, Back propogation and automatic differentiation, Higher order partial derivatives, Multivariate Taylor's series
Unit 12	Continuous Optimization: Optimization using Gradient Descent, Constrained Optimization, Lagrange's multipliers

TEXTBOOKS:

- Mathematics for machine learning, Marc Peter Deisenroth, A. Aldo Faisal, Cheng Soon Ong
- An introduction to the theory of numbers, Ivan Niven, Herbert S Zuckerman, Hugh L Montgomery
- Probability and statistics for engineers (Erwin Miller And John E.Freund), R A Johnson And C.B.Gupta.. 7th edition, Pearson Education / PHI.
- Milton. J. S. and Arnold. J.C., “Introduction to Probability and Statistics”, Tata McGraw Hill, 4 th Edition, 2007.

REFERENCE BOOKS:

- Devore. J.L., “Probability and Statistics for Engineering and the Sciences”, Cengage Learning, New Delhi, 8th Edition, 2012.
- Walpole. R.E., Myers. R.H., Myers. S.L. and Ye. K., “Probability and Statistics for Engineers and Scientists”, Pearson Education, Asia, 8th Edition, 2007.
- Ross, S.M., “Introduction to Probability and Statistics for Engineers and Scientists”,3rd Edition, Elsevier, 2004.

Semester: II

Course Name: MACHINE LEARNING WITH PYTHON

Credits: 3

Course Description:

This course introduces in-depth exploration of fundamental and advanced machine learning concepts. Learners will learn about various learning paradigms including supervised, unsupervised, and reinforcement learning, covering algorithms such as K-Nearest Neighbor, Linear Regression, Support Vector Machines, Decision Trees, and Neural Networks. The curriculum emphasizes practical skills in data preparation, exploratory data analysis, model validation, and introduces advanced topics like deep learning and neural network architectures.

Course Objectives:

This Course will enable learners to:

- Comprehend the concepts of machine learning and types of problem stack led by machine learning.
- Analyse the role of probabilistic methods for machine learning and Reinforcement Learning
- Explore the different supervised learning techniques including ensemble methods
- Learn different aspects of unsupervised learning and reinforcement learning

Course Outcome:

At the end of the course the learner will be able to:

- Design a Decision tree and Random Forest for an application
- Implement Probabilistic Discriminative and Generative algorithms for an application and analyse the results
- Use a tool to implement typical Clustering algorithms for different types of applications
- Design and implement an HMM for a Sequence Model type of application and identify applications suitable for different types of Machine Learning with suitable justification.

Unit 1	Introduction of Machine Learning: Introduction to Machine Learning, Applications, Advantages and Disadvantages, Types of Machine Learning, Machine Learning Life Cycle, Definition of Machine Learning, Real-world Applications Benefits and Limitations Supervised, Unsupervised, Reinforcement Learning, Stages of ML Life Cycle, Real-world Applications, Benefits and Limitations, Supervised, supervised, Reinforcement Learning, Stages of ML Life Cycle of Machine Learning
Unit 2	Data Preparation and Exploratory Data Analysis: Introduction to Scikit-learn, Data Preparation, Exploratory Data Analysis (EDA), Underfitting and Overfitting, setting up Scikit-learn environment, Data Cleaning and Pre-processing Visualization Techniques, Handling Missing Values, Concepts of Underfitting and Overfitting
Unit 3	Reinforcement Learning: Performance Metrics, Elements of Reinforcement Learning, Model-based Learning, Temporal Difference Learning, Defining Performance Metrics for RL, Agent, Environment, Reward, Policy, Differences between Model-based and Model-free Learning, TD Learning Algorithms
Unit 4	Probabilistic Methods and Sequence Models: Probability Density Estimation, Sequence Models, Markov Models, Hidden Markov Models (HMM), Estimating Probability Densities, Understanding Sequence Data, Markov Processes and Properties, Structure and Applications of HMMs
Unit 5	Regression Algorithms: Introduction to Supervised Learning, Linear Regression, Logistic Regression, Least Squares Method, Difference between Regression and Classification, Simple and Multiple Linear Regression, Logistic Regression for Classification, Calculating Least Squares
Unit 6	Classification Algorithms: K-Nearest Neighbour (KNN) Algorithm, Support Vector Machine (SVM) Algorithm, Naive Bayes, Decision Trees, Random Forest, Distance Metrics in KNN, SVM for Classification, Bayes' Theorem in Naive Bayes, Tree Construction and Pruning, Ensemble Methods in Random Forest
Unit 7	Clustering Algorithms: Introduction to Unsupervised Learning, Types of Unsupervised Learning, K-Means Clustering, Cluster Validity, Difference between Supervised and Unsupervised Learning, Applications of Clustering, K-Means Algorithm Steps, Evaluating Clusters using Validity Indices

Unit 8	Dimensionality Reduction and Advanced Techniques: Dimensionality Reduction, Principal Component Analysis (PCA), Cross-Validation, Expectation-Maximization (EM) Algorithm, Importance of Dimensionality Reduction, Steps in PCA, Techniques for Model Validation, EM Algorithm for Parameter Estimation
Unit 9	Fundamentals of Neural Networks: Biological Motivation, Perceptron, Multi-layer Perceptron (MLP), Feed Forward Network, Backpropagation, Activation and Loss Functions, Neuron Structure Comparison, Single-layer Perceptron Limitations, Architecture of MLP, Forward Propagation Process, Backpropagation Algorithm, Common Activation Functions (ReLU, Sigmoid, Tanh)
Unit 10	Deep Learning Architectures and Applications: Deep Learning Overview, Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), Use Cases, Differences between Deep Learning and Traditional Neural Networks, Structure and Applications of CNNs, Understanding RNNs and LSTMs, Real-world Applications (e.g., Image Recognition, Natural Language Processing)

TEXTBOOKS:

- Thomas H.Cormen,Charles E. Leiserson, Ronald LRivest, CliffordStein, "Introduction to Algorithms", MIT Press, England, 2009.
- Alpaydin, E. (2020). Introduction to Machine Learning (4th ed.). MIT Press.
- Murphy, K. P. (2022). Probabilistic Machine Learning: An Introduction. MIT Press.

REFERENCE BOOKS:

- Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press.

Course Name: ADVANCED JAVA PROGRAMMING

Credits: 3

Course Description:

This course provides an extensive journey through Java fundamentals, object-oriented programming concepts, and advanced enterprise development technologies. Learners will master core Java programming elements including data types, control statements, classes, inheritance, exception handling, and multithreading, progressively building sophisticated programming skills. The curriculum advances to database connectivity using JDBC and culminates with modern enterprise application development using Spring Boot, covering critical aspects like dependency injection, application configuration, web deployment, and advanced mapping techniques.

Course Objectives:

This Course will enable learners to:

- Comprehend the foundational principles of Java programming, including syntax, data types, operators, and basic programming concepts.
- Develop programs using object-oriented programming (OOP) principles such as classes, objects, and exception handling.
- Gain hands-on experience with Java tools and frameworks like JDBC for database connectivity and Spring Boot for building web applications.
- Explore advanced Java concepts like metadata, transaction management, and debugging techniques for developing robust applications.
- Build, configure, and deploy Java-based applications efficiently using Maven/Gradle and advanced Spring Boot features.

Course Outcomes:

At the end of the course the learner will be able to:

- Design and implement basic Java programs by applying knowledge of keywords, tokens, I/O operations, and control structures.
- Develop Java programs involving arrays, exception handling, and OOP principles to solve real-world problems.

- Integrate Java programs with relational and NoSQL databases using JDBC for performing CRUD operations.
- Build and deploy Spring Boot applications with structured code, global exception handling, and advanced configuration.
- Debug and optimize Spring Boot applications, utilizing advanced techniques and default configurations to ensure performance and scalability.

Unit 1	The Java Language Basics: The History and Evolution of Java, An Overview of Java, Data Types, Variables, and Arrays
Unit 2	Control Structures and Classes: Operators, Control Statements, Introducing Classes, A Closer Look at Methods and Classes
Unit 3	Object-Oriented Programming in Java: Inheritance, Packages and Interfaces,
Unit 4	Exception Handling and Multithreading: Exception handling, Multithreaded Programming, File Handling and Event Handling
Unit 5	Introduction to JDBC: Introduction to JDBC, JDBC Drivers & Architecture, CRUD Operations Using JDBC, Connecting to Non,Conventional Databases
Unit 6	Advanced JDBC Concepts: Statement Objects, ResultSet, Transaction Processing, Metadata, Data Types, Exceptions
Unit 7	Introduction to Spring Boot: Spring Boot Bootstrapping, Spring Boot Tomcat Deployment
Unit 8	Building Spring Boot Applications: Spring Boot Build Creation, Spring Boot Code Structure, Application Properties, Exception Handling
Unit 9	Advanced Configuration and Debugging: Default Configuration, Exploring the WebConfigurerAdapter, Debugging,
Unit 10	Advanced Features in Spring Boot: Advanced JSON Mapping, Model Attributes

TEXTBOOKS:

- Herbert Schildt: JAVA the Complete Reference, 7th/9th Edition, Tata McGraw Hill, 2007.
- Jim Keogh: J2EE-TheCompleteReference, McGraw Hill, 2007.
- Hortsmann & Cornell, "CORE JAVA 2 ADVANCED FEATURES, VOL II", Pearson

REFERENCE BOOKS:

- Ed Roman, "Mastering Enterprise Java Beans", John Wiley & Sons Inc., 1999.
- Y. Daniel Liang: Introduction to JAVA Programming, 7thEdition, Pearson Education, 2007

Course Name: DESIGN AND ANALYSIS OF ALGORITHMS

Credits: 4

Course Description:

This course provides an in-depth exploration of fundamental problem-solving techniques and advanced algorithmic strategies. Learners will master critical computational approaches including Divide and Conquer, Greedy Methods, Dynamic Programming, and Randomized Algorithms through rigorous theoretical foundations and practical implementations. The curriculum covers complex algorithmic challenges such as sorting techniques (Merge Sort, Quick Sort), graph algorithms (Dijkstra's, Prim's, Kruskal's), optimization problems (Knapsack, Travelling Salesperson), and advanced methods like Heap Sort and Randomized Algorithms.

Course Objectives:

This Course will enable learners to:

- Comprehend algorithm design and analysis provide the theoretical backbone of computer science and are a must in the daily work of the successful programmer.
- Provide a solid background in the design and analysis of the major classes of algorithms.
- Develop their own versions for a given computational task and to compare their performances.
- Develop strategies to incorporate randomness into existing algorithms to improve their performance.
- Formulate dynamic programming solutions for complex problems.

Course Outcomes:

At the end of the course the learner will be able to:

- Describe the basic algorithm design strategies and use them for devising new solutions to various problems.
- Describe the divide-and-conquer paradigm and explain when an algorithmic design situation calls for it. Synthesize divide-and-conquer algorithms.
- Formulate greedy solutions for classic optimization problems.
- Identify appropriate scenarios where the greedy approach is applicable.

- Formulate dynamic programming solutions for complex problems.

Unit 1	Fundamentals of Algorithms: Definition of Algorithms, Properties of Algorithms, Recurrence and Non-Recurrence Algorithms, What is an algorithm?, Input and output specifications, Definiteness, effectiveness, finiteness, Examples of Recurrence and Non-Recurrence Algorithms - Algorithm Specification, Important Problem Types, Steps in algorithm specification, Problem types: String Processing, Graph Problems, Combinatorial Problems
Unit 2	Analysis of Algorithms: Asymptotic Notations, Time and space complexity, Big O notation, Omega and Theta notations, Significance of asymptotic analysis, - Analysis Framework, Notation (o), Importance of analysing algorithms, Measuring space and time complexity, Little-o notation and its significance
Unit 3	Solving Recurrence Relations: Substitution method, Recursion tree method, Master theorem, Examples and practice problems
Unit 4	Fundamental Data Structures: Stacks, Queues, Graphs, Trees, Sets and Dictionaries, Operations and implementations, Applications in Algorithms, Importance in problem-Solving and algorithm Efficiency
Unit 5	Introduction to NP-Hard and NP-Complete Problems: NP-Hard Problems, NP-Complete Problems, P vs. NP problem, Definitions and differences between NP, NP-Hard, and NP-Complete, Examples of NP-Complete problems, Significance in computational theory
Unit 6	Divide-and-Conquer Method: The General Method, Finding Maximum and Minimum Elements, Quick Sort, Merge Sort, Matrix Multiplication, Concept of dividing problems into subproblems, Algorithm for finding max and min, Quick Sort algorithm and analysis, Merge Sort algorithm and analysis, Strassen's Matrix Multiplication method, - Advantages and Disadvantages of Divide and Conquer, Topological Sort, Reducing problem size incrementally, Implementation of topological sorting, Applications in scheduling and dependency resolution, Heaps, Heap Sort, Building and maintaining heaps, Heap operations (insertion, deletion), Using heaps for efficient sorting
Unit 7	Greedy Method Fundamentals: The General Method, Optimal Storage on Tapes, Knapsack Problem, Principles of greedy algorithms, Greedy choice property, Problem of optimal storage on tapes, Greedy solution to the Knapsack problem, Greedy Method Applications: Minimum Spanning Trees, Single Source Shortest Path, Floyd-Warshall's Algorithm, Prim's and Kruskal's algorithms for MST, Dijkstra's algorithm for

	shortest paths, Application of greedy method in graph algorithms, Understanding Floyd-Warshall's algorithm for all pairs shortest paths
Unit 8	Dynamic Programming Concepts: The General Method, All Pairs Shortest Path, Principle of optimality, Overlapping subproblems, Memoization vs. tabulation, Implementation of the All Pairs Shortest Path problem
Unit 9	Dynamic Programming Applications: Optimal Binary Search Tree, Multistage Graphs, constructing an optimal BST, Calculating search costs, Solving multistage graph problems, Cost optimisation in multistage decision processes, Bellman-Ford Algorithm, Reliability Design, Algorithm explanations, Applications in networking and reliability
Unit 10	Backtracking Concepts: The General Method, Solution Space and Tree Organization, constructing solution space trees, Depth-first search strategies, Pruning and bounding techniques, Framework for backtracking algorithms
Unit 11	Backtracking Applications: The Eight Queens Problem, Sum of Subsets Problem, Graph Coloring, Knapsack Problem, Formulating the Eight Queens problem, Recursive solutions, Coloring graphs with minimal colors, Backtracking solution to the Knapsack problem, Analysing time complexity of backtracking algorithms
Unit 12	Branch and Bound Concepts: The General Method, LC Search, LC Branch and Bound, FIFO Branch and Bound, Difference between backtracking and branch and bound, Least Cost (LC) search strategies, Implementing LC branch and bound, FIFO branch and bound approaches
Unit 13	Branch and Bound Applications: 0/1 Knapsack Problem, Traveling Salesperson Problem, Efficiency Considerations, Applying branch and bound to the Knapsack problem, Solving the Traveling Salesperson Problem (TSP), Analysing efficiency and computational complexity, Strategies to improve performance of branch and bound algorithms
Unit 14	Probabilistic and Randomized Algorithms: Probabilistic Algorithms, Randomizing Deterministic Algorithms, Randomizing Quick Sort, Monte Carlo Algorithms, Role of randomness in algorithms, Designing probabilistic algorithms, Analysis of expected performance, Applications where randomized algorithms outperform deterministic counterparts

TEXTBOOKS:

- Introduction to the Design and Analysis of Algorithms, Anany Levitin: 2nd Edition, 2009. Pearson.
- Computer Algorithms/C++, Ellis Horowitz, SatrajSahni and Rajasekaran, 2nd Edition, 2014, Universities Press.
- Algorithms, Kenneth A Berman and Jerome L Paul, Cengage Learning India Pvt Ltd, 2002 edition.

REFERENCE BOOKS:

- Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson, Ronal L. Rivest, Clifford Stein, 3rd Edition, PHI.
- Design and Analysis of Algorithms, S. Sridhar, Oxford (Higher Education)

Course Name: FULL STACK DEVELOPMENT

Credits: 3

Course Description:

This course provides an integrated journey through modern web technologies, covering front-end and back-end programming paradigms. Learners will master web fundamentals including HTML, CSS, and JavaScript, progressing to advanced server-side programming with Node.js and database integration using NoSQL technologies like MongoDB. The curriculum advances to modern front-end frameworks like React.js, exploring component-based architecture and state management, and concludes with cloud computing concepts including containerization, Kubernetes, and cloud infrastructure strategies.

Course Objectives:

This Course will enable learners to:

- Infer the foundational concepts of web development, including HTML, CSS, and JavaScript, for building interactive and responsive web pages.
- Explore server-side development using Node.js and Express, emphasizing RESTful APIs, asynchronous programming, and templating engines.
- Develop a deep understanding of NoSQL databases like MongoDB and integrate them with Node.js applications for full-stack development.
- Gain practical experience with React.js for building dynamic user interfaces, managing state, and handling events in single-page applications.

Course Outcomes:

At the end of the course the learner will be able to:

- Develop well-structured and interactive web pages using HTML, CSS, and JavaScript to enhance user experience.
- Build and deploy server-side applications with Node.js and Express, leveraging RESTful APIs and asynchronous programming for efficient data handling.
- Integrate MongoDB with Node.js applications to perform CRUD operations, manage user sessions, and implement authentication mechanisms.
- Design and implement React.js applications using state management, lifecycle methods, and event handling to create dynamic and scalable interfaces.

Unit 1	Introduction to web and HTML: Server and Client,Communication Protocol (HTTP),Structure of HTML Documents Basic Markup Tags Understanding the roles of servers and clients,How HTTP facilitates web communicationHTML document structure (<!DOCTYPE>, <html>, <head>, <body>) Common HTML tags like headings, paragraphs, links, images.
Unit 2	CSS and JavaScript Basics: Working with Text and Images using CSS, CSS Selectors, CSS Flexbox,JavaScript Basics,Data Types and Variables,Functions,Events, AJAX: GET and POST, Styling text fonts and images with CSS,Using CSS,selectors (class id element selectors) ,Implementing layouts with CSS Flexbox JavaScript syntaxand data types ,Declaring variables (let const var), Writing functions and handling events,Making asynchronous calls using AJAX (XMLHttpRequest Fetch API).
Unit 3	Introduction to Node.js and Express: Introduction to Web Servers ,JavaScript on the Desktop with Node.js, NPM (Node Package Manager),Serving Files with the http Module,Understanding the purpose of web servers, Installing and setting up Node.js,Running JavaScript outside the browser, Managing packages with NPM,Creating a basic HTTP server, Serving static files (HTML, CSS, JS).
Unit 4	Advanced Server-Side Concepts: Introduction to the Express Framework, Server,Side Rendering with Templating Engines,Handling Static Files,Async/Await in Node.js,Fetching JSON from Express,Setting up Express.js applications, Using templating engines like EJS or Pug,Middleware for serving static assets, Writing asynchronous code with async/await,Creating RESTful APIs, Handling JSON data in requests and responses.
Unit 5	NoSQL Databases and MongoDB: Introduction to NoSQL Databases,MongoDB System Overview, Basic Querying with MongoDB Shell,Understanding the difference between SQL and NoSQL databases, Installing and setting up MongoDB, MongoDB data models (documents, collections),Performing CRUD operations in the MongoDB shell.
Unit 6	Integrating MongoDB with Node.js: Request Body Parsing in Express,Node.js MongoDB Connection,Adding and Retrieving Data from MongoDB, Handling Cookies in Node.js, User Authentication with Node.js, Parsing incoming request bodies with body,parser middleware,Connecting to MongoDB using Mongoose or MongoDB driver,Implementing create, read, update, delete (CRUD) operations in Node.js, Managing user sessions with cookies and sessions, Implementing authentication strategies (e.g., JWT, Passport.js).
Unit 7	React.js Fundamentals: Introduction to React.js,ReactDOM,JSX,Components and Props,Setting up a React application using Create React App,Understanding the virtual DOM,Writing JSX syntax Creating functional and class components,Passing data through props.
Unit 8	Advanced React.js Concepts: State and Lifecycle, Handling Events,Lifting State Up, Composition and Inheritance

	Fetch API,Local Storage, Managing component state, Lifecycle methods (componentDidMount, componentDidUpdate, componentWillUnmount),Event handling in React,Sharing state between components, Composing components,Fetching data from APIs, Storing data in the browser's local storage.
Unit 9	Cloud Computing Basics: Overview of Cloud Providers,Virtual Private Cloud (VPC), Scaling Applications, Virtual Machines, Ethernet, and Switches, Understanding services offered by cloud providers (AWS, Azure, Google Cloud),Setting up and configuring a VPC, Horizontal vs. Vertical Scaling, Basics of networking in the cloud environment
Unit 10	Containerization and Orchestration: Docker Containers, Kubernetes Basics, Creating and managing Docker containers, Writing Dockerfiles, Containerizing applications, Introduction to Kubernetes, Deploying and managing containers with Kubernetes.

TEXTBOOKS:

- Web Design with HTML, CSS, JavaScript and JQuery Set Book by Jon Duckett
- Professional Java Script for Web Developers Book by Nicholas C. Zakas
- Learning PHP, MySQL, JavaScript, CSS & HTML5: A Step-by- Step Guide to Creating Dynamic Websites by Robin Nixon
- Web Development with Node and Express by Ethan Brown
- Full stack React: The Complete Guide to ReactJS and Friends, Anthony Accomazzo

Joe Beda ,Kelsey Hightower ,Brendan Burns,"Kubernets Up and Running ",O'Reily Media,1st Edition ,2017.

- Paul Zikopoulos ,Christopher Bienko,Chris Backer , Chris Konarski, Sai Vennam, "Cloud Without Compromise", O'Reily Media ,1 st edition,2021

REFERENCE BOOKS:

- Full-Stack JavaScript Development by Eric Bush.
- Mastering Full Stack React Web Development Paperback – April 28, 2017 by TomaszDyl Kamil Przeorski , Maciej Czarnecki
- Full Stack JavaScript: Learn Backbone.js, Node.js and MongoDB. Copyright © 2015 BY AZAT MARDAN

Course Name: COMPUTER COMMUNICATION NETWORK

Credits: 3

Course Description:

This course provides an in-depth exploration of data communication principles and network technologies, covering fundamental communication models to advanced network layer protocols. Learners will master critical concepts including digital transmission, network architectures, protocol design, and communication technologies across application, transport, network, and data link layers. The curriculum progresses through comprehensive topics such as TCP/IP protocols, socket programming, routing mechanisms, network addressing, error detection techniques, and emerging communication technologies.

Course Objectives:

This Course will enable learners to:

- Demonstrate in depth knowledge of network, basic concepts, networks devices and transmission media, Analog and digital data transmission
- Apply channel allocation, framing, error and flow control techniques.
- Describe the functions of Network Layer i.e. Logical addressing, subnetting & Routing Mechanism.
- Evaluate the different protocols used at application layer i.e. HTTP, FTP L5

Course Outcomes:

At the end of the course the learner will be able to:

- Identify the fundamentals of data communication and digital transmission and bandwidth utilization using various transmission media.
- Analyse the principles of Application protocol layering in modern communication systems.
- Utilize the different Transport Layer function i.e. Port addressing, Connection Management, Error control and Flow control mechanism.
- Illustrate the link layer services and error detection.

Unit 1	Introduction to Data Communication: Communication Model, data communications Data Transmission Concepts: Data Transmission Concepts and Terminology, Analog and Digital, Transmission Impairments
Unit 2	Signal Encoding and Multiplexing: Signal Encoding Techniques: Digital Data and Digital Signals, Multiplexing Techniques
Unit 3	Data Link Layer Fundamentals: Hardware Building Blocks: Nodes and Links, Link Layer Functions
Unit 4	Reliable Transmission and MAC Protocols: Reliable Transmission: StopandWait, Sliding Window, Media Access Control: Ethernet, WiFi, Bluetooth
Unit 5	Network Layer Functions and Switching: Network Layer Functions, Switching and Forwarding: Datagrams, Virtual Circuit Switching, Source Routing
Unit 6	IP Addressing and Internetworking: Internetworking: IPv4, IP Address Classes, Subnetting, CIDR
Unit 7	Network Protocols and Services: ARP, DHCP, ICMP
Unit 8	Routing Algorithms and IPv6: Routing: Distance Vector Routing (DVR), Link State Routing (LSR), BGP, IPv6
Unit 9	Transport Layer Overview and UDP: Transport Layer Services, User Datagram Protocol (UDP), Reliable Byte Stream, Connection Establishment and Termination, Sliding Window, Triggering Transmission, Adaptive Retransmission, TCP Extensions, TCP Congestion Control

Unit 10	Network Applications and Protocols: Principles of Network Applications, Protocol Architecture: TCP/IP, Introduction to Socket Programming, Web and HTTP, FTP, DNS, SMTP, POP, IMAP, PeertoPeer Applications, Performance Metrics: Latency, Delay, Bandwidth Product

TEXTBOOKS:

- Behrouz A. Forouzan, Data Communications and Networking 5E, 5th Edition, Tata McGraw-Hill,2013.

REFERENCE BOOKS:

- Larry L. Peterson and Bruce S. Davie: Computer Networks – A Systems Approach, 4th Edition, Elsevier, 2019
- Nader F. Mir: Computer and Communication Networks, 2nd Edition, Pearson Education, 2015.
- William Stallings, Data and Computer Communication 10 Edition, Pearson Education, Inc., 2014

Course Name: CLOUD COMPUTING TECHNOLOGIES

Credits: 4

Course Description:

This course provides an extensive exploration of cloud technologies, architectural principles, and advanced computational paradigms. Learners will master fundamental concepts including cloud computing reference models, virtualization techniques, distributed computing principles, and service-oriented architectures. The curriculum covers critical domains such as cloud service models (IaaS, PaaS, SaaS), cloud types, virtualization technologies like Docker and Kubernetes, and emerging cloud platforms including Amazon Web Services, Google AppEngine, and Microsoft Azure.

Course Objectives:

This Course will enable learners to:

- Develop a comprehensive understanding of the vision, principles, and motivation behind cloud computing, its architecture, and reference models.
- Explore the historical evolution of computing paradigms, including distributed systems, virtualization, and service-oriented computing.
- Analyse the role of virtualization in enabling cloud services and evaluate various virtualization techniques and platforms.
- Examine different types of cloud architectures, deployment models, and service providers, and assess their suitability for various organizational needs.
- Investigate real-world applications of cloud computing across domains, addressing economic aspects, open challenges, and emerging technologies like OpenStack and Aneka.

Course Outcomes:

At the end of the course the learner will be able to:

- Explain the foundational concepts, definitions, and motivations of cloud computing and its role in modern IT infrastructure.
- Analyse the evolution of computing paradigms, including distributed systems, virtualization, and Web 2.0, and their impact on cloud computing

- Apply virtualization techniques such as full virtualization, para-virtualization, and containerization using platforms like VMware, Xen, and Docker.
- Evaluate cloud service and deployment models (IaaS, PaaS, SaaS, public, private, hybrid clouds) for real-world applications and organizational suitability.
- Demonstrate the use of cloud computing in various domains like healthcare, scientific research, and business, addressing economic aspects and open challenges.

Unit 1	The Vision of Cloud Computing and Defining a Cloud: The Vision of Cloud Computing, defining a Cloud, Understanding the need for cloud computing, Definitions and characteristics of a cloud, Goals and motivations behind cloud computing
Unit 2	A Closer Look and Cloud Computing Reference Model: A Closer Look at Cloud Computing, Cloud Computing Reference Model, Detailed exploration of cloud components, Layers of cloud computing, Service models (IaaS, PaaS, SaaS), Deployment models
Unit 3	Historical Developments in Cloud Computing: Distributed Systems, Virtualization, Web 2.0, Service,Oriented Computing, Utility,Oriented Computing, Evolution of computing paradigms, Role of distributed systems in cloud computing, Impact of virtualization and Web 2.0, Transition from traditional computing to service,oriented and utility computing
Unit 4	Eras of Computing and Computing Models: Eras of Computing, Parallel vs. Distributed Computing, Mainframe computing, Personal computing, Client,server computing, Grid computing, Comparison between parallel and distributed computing models
Unit 5	Elements and Technologies of Distributed Computing: Elements of Distributed Computing, Technologies for Distributed Computing, Nodes, networks, and communication, Middleware, Distributed algorithms, Technologies like RPC, CORBA, RMI, Messaging systems
Unit 6	Introduction to Virtualization: Definition of Virtualization, Characteristics of Virtualized Environments, Purpose and benefits of virtualization, Types of virtualization (server, storage, network, desktop), Isolation and encapsulation
Unit 7	Virtualization Techniques and Cloud Computing: Taxonomy of Virtualization Techniques, Virtualization and Cloud Computing, Full virtualization, Para, virtualization, Hardware, assisted virtualization, Role of virtualization in enabling cloud services, Hypervisors

Unit 8	Pros and Cons of Virtualization and Technology Examples: Advantages and Disadvantages of Virtualization, Technology Examples, Xen: Para, virtualization, VMware: Full Virtualization, Microsoft Hyper,V, Docker and Kubernetes, Benefits: Resource optimization, scalability, flexibility, Challenges: Security, performance overhead, Case studies of virtualization platforms, Containerization with Docker, Orchestration with Kubernetes
Unit 9	Introduction to Cloud Architecture and Reference Model: Cloud Architecture Overview, Cloud Reference Model, Components and layers of cloud architecture, Service models (IaaS, PaaS, SaaS), Deployment models overview
Unit 10	Types of Clouds: Public Clouds, Private Clouds, Hybrid Clouds, Community Clouds, Definitions and characteristics, Use cases and examples, Benefits and challenges of each type, Suitability for different organizations
Unit 11	Economics of the Cloud and Open Challenges: Economics of Cloud Computing, Open Challenges in Cloud Computing, Cost models and pricing, ROI and TCO analysis, Scalability and resource optimization, Security and privacy concerns, Compliance and legal issues, Interoperability and portability challenges
Unit 12	Aneka PaaS and OpenStack: Aneka Platform as a Service, Introduction to OpenStack, Components of OpenStack, Features and architecture of Aneka, Developing applications with Aneka, OpenStack services (Nova, Swift, Cinder, Neutron), Deploying and managing cloud infrastructure with OpenStack
Unit 13	Major Cloud Service Providers: Amazon Web Services (AWS), Google AppEngine, Microsoft Azure, Services offered by AWS (EC2, S3, Lambda, etc.), Features and development with Google AppEngine, Microsoft Azure services and capabilities, Comparing different cloud service providers
Unit 14	Cloud Applications in Various Domains: Scientific Applications, Healthcare, Biology, Geo,Science, Business and Consumer Applications, ARM & ERP, Productivity Tools, Social Networking How cloud computing supports scientific research, Cloud,based solutions in healthcare and biology, Geospatial analysis using the cloud, Enterprise applications in the cloud, Cloud,based productivity and collaboration tools, Social media platforms leveraging cloud technology

TEXTBOOKS:

- RajkumarBuyya, Christian Vecchiola, and ThamaraiSelvi, Mastering Cloud Computing, Tata McGraw Hill, New Delhi, India, 2013.

REFERENCE BOOKS:

- Cloud Computing for Dummies by Judith Hurwitz, R.Bloor, M. Kanfman, F.Halper (Wiley India Edition)
- Cloud Computing: A Practical Approach by J.Vette, Toby J. Vette, Robert Elsenpeter (Tata McGraw Hill)

Course Name: BIG DATA ANALYTICS

Credits: 4

Course Description:

This course provides an in-depth exploration of modern data management and processing technologies. Learners will master critical concepts including big data characteristics, NoSQL databases, Hadoop ecosystem, distributed computing paradigms, and advanced analytics tools like MongoDB, MapReduce, and Apache Spark. The curriculum covers sophisticated data management techniques, exploring diverse technologies such as Hadoop Distributed File System, Hive, Pig, and Spark, with a focus on practical skills in handling large-scale, complex data environments.

Course Objectives:

This Course will enable learners to:

- Provide an in-depth understanding of Big Data concepts, challenges, and its importance in modern technology and business.
- Introduce the architecture, components, and ecosystem of Hadoop and the role of NoSQL databases like MongoDB in Big Data environments.
- Explore programming and data processing frameworks, including MapReduce and Apache Spark, for distributed and large-scale data analysis.
- Familiarize learners with query and scripting tools like Apache Hive and Pig for efficient Big Data management and analysis.
- Develop practical skills in using Big Data tools and technologies for data processing, analytics, and real-time applications.

Course Outcomes:

At the end of the course the learner will be able to:

- Explain the characteristics, classification, and evolution of Big Data, and identify its challenges and applications.
- Compare traditional databases with NoSQL databases and assess their suitability for various Big Data use cases.
- Implement distributed data processing using Hadoop’s HDFS, MapReduce, and related ecosystem tools like Hive and Pig.

- Utilize Apache Spark for large-scale data processing, including transformations, real-time streaming, and machine learning applications.
- Design and develop data analysis workflows using tools like MongoDB, Spark, Hive, and Pig, addressing real-world Big Data challenges.

Unit 1	Introduction to Big Data: Classification of Digital Data, Structured and Unstructured Data, Characteristics of Big Data, Evolution of Big Data, Definition of Big Data, Types of Digital Data, Differences between Structured and Unstructured Data, The 5 V's of Big Data (Volume, Velocity, Variety, Veracity, Value), Historical Evolution of Data, Defining Big Data
Unit 2	Big Data Challenges and Importance: Challenges with Big Data, Other Characteristics of Data, Why Big Data, Data Complexity, Data Quality Issues, Data Storage and Processing Challenges, Importance of Big Data in Modern Business and Technology
Unit 3	Traditional BI vs. Big Data: Traditional Business Intelligence vs. Big Data, Data Warehouse and Hadoop Environment, Limitations of Traditional BI, Comparison between Data Warehouses and Hadoop, Integration of Big Data Technologies with Existing BI Systems
Unit 4	Big Data Analytics Overview: Classification of Analytics, Challenges in Analytics, Importance of Big Data Analytics, Data Science and Data Scientists, Terminologies in Big Data Environments, BASE, Top Analytics Tools, Descriptive, Predictive, and Prescriptive Analytics, Roles and Responsibilities of Data Scientists, Concepts like BASE (Basically Available Soft State Eventual Consistency), Overview of Popular Big Data Tools
Unit 5	Introduction to NoSQL: NoSQL Databases, Comparison of SQL and NoSQL, Types of NoSQL Databases (Document, Key,Value, Columnar, Graph), Use Cases for NoSQL, Advantages and Disadvantages Compared to SQL Databases
Unit 6	Hadoop Basics: RDBMS vs. Hadoop, Distributed Computing Challenges, Hadoop Overview, Limitations of RDBMS for Big Data, Concepts of Distributed Computing, Hadoop's Architecture and Components (HDFS, MapReduce)
Unit 7	Hadoop Components and Ecosystem: Hadoop Distributed File System (HDFS), Processing Data with Hadoop, Managing Resources with Hadoop YARN, Interacting with the Hadoop Ecosystem, HDFS Architecture, Data Storage in HDFS, MapReduce Processing,

	Role of YARN in Resource Management, Overview of Hadoop Ecosystem Tools (Hive, Pig, HBase, etc.)
Unit 8	Introduction to MongoDB: Why MongoDB, Terms Used in RDBMS and MongoDB, Data Types, MongoDB Query Language, Features of MongoDB, Comparison of MongoDB Concepts with RDBMS, BSON Data Types, CRUD Operations in MongoDB, Querying Documents
Unit 9	MapReduce Programming: MapReduce Concepts, Mapper, Reducer, Combiner, Partitioner, Searching, Sorting, Compression, Writing MapReduce Jobs, Data Flow in MapReduce, Role of Combiner and Partitioner, Optimizing MapReduce Jobs, Data Compression Techniques in Hadoop
Unit 10	Introduction to Spark: Introduction to Spark, Downloading and Setting Up Spark, Getting Started with Spark, Features of Apache Spark, Differences Between Spark and Hadoop MapReduce, Spark Installation Steps, Running Basic Spark Applications
Unit 11	Programming with Spark: Programming with RDDs, Working with Key/Value Pairs, Understanding Resilient Distributed Datasets (RDDs), Transformations and Actions, Pair RDDs and Their Operations (reduceByKey, groupByKey, etc.)
Unit 12	Advanced Spark Concepts: Loading and Saving Data, Spark SQL, Spark Streaming, Machine Learning with MLlib, Reading and Writing Data from Various Sources, Using DataFrames and Datasets, Real,Time Data Processing with Spark Streaming, Implementing Machine Learning Algorithms Using MLlib
Unit 13	Apache Hive: Introduction to Hive, Architecture, Data Types, File Formats, Hive Query Language Statements, Partitions and Bucketing, Views and Subqueries, Joins and Aggregations, Group By and Having, RCFile Implementation, Hive User Defined Functions (UDFs), Serialization and Deserialization, Understanding Hive's Role in the Hadoop Ecosystem, Writing SQL,Like Queries in HiveQL, Optimizing Data Storage with Partitions and Bucketing, Implementing UDFs for Custom Processing, Data Serialization Formats in Hive
Unit 14	Apache Pig: Introduction to Pig, Anatomy of Pig, Features and Philosophy, Use Cases for Pig, Pig Latin Overview, Pig Primitive Data Types, Running Pig, Execution Modes of Pig, Understanding Pig's Data Flow Language, Writing Scripts in Pig Latin, Data Types and Schemas in Pig, Running Pig in Local and MapReduce Modes, When to Use Pig Over Other Tools.

TEXTBOOKS:

- Tom White, “Hadoop: The Definitive Guide” Third Edit on, O’reily Media, 2012.
- Michael Mineli, Michele Chambers, Ambiga Dhiraj, "Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's Businesses", Wiley Publications, 2013.
- Holden Karau, Andy Konwinski, Patrick Wendell, and Matei Zaharai, “Learning Spark”, 5th Edition, O’Relly, 2015.
- Ethem Alpaydın, “Introduction to Machine Learning”2nd Edition, The MIT Press Cambridge, Massachusetts London, England

REFERENCE BOOKS:

- Michael Berthold, David J. Hand, "Intelligent Data Analysis”, Springer, 2007.
- Jay Liebowitz, “Big Data and Business Analytics” Auerbach Publications, CRC press (2013)
- Tom Plunkett, Mark Hornick, “Using R to Unlock the Value of Big Data: Big Data Analytics with Oracle R Enterprise and Oracle R Connector for Hadoop”, McGraw-Hill/Osborne Media (2013), Oracle press.

Course Name: DATA SCIENCE

Credits: 4

Course Description:

This course provides an extensive exploration of advanced data analysis techniques and computational methodologies. Learners will master critical concepts including data preprocessing, pattern recognition, classification algorithms, clustering techniques, and sophisticated visualization strategies using Python. The curriculum covers fundamental data mining principles, exploring decision tree induction, Bayesian classification, rule-based systems, and clustering methodologies while emphasizing practical skills in extracting meaningful insights from complex datasets.

Course Objectives:

This Course will enable learners to:

- Comprehend the fundamental concepts, applications, and challenges of data mining and its role in business and technology.
- Acquire knowledge of various data pre-processing techniques for improving the quality of datasets for analysis.
- Explore frequent pattern mining methods and apply algorithms like Apriori, FP-Growth, and ECLAT.
- Learn classification, clustering, and association rule techniques and their real-world applications.
- Master the basics of data visualization using Python and advanced visualization techniques to effectively communicate insights.

Course Outcomes:

At the end of the course the learner will be able to:

- Explain the fundamentals of data mining, including tasks, techniques, and challenges.
- Apply data pre-processing methods such as cleaning, integration, reduction, and transformation to prepare datasets for analysis.
- Develop and implement algorithms like Apriori, FP-Growth, and ECLAT for frequent pattern and association rule mining.

- Evaluate and compare clustering and classification techniques using appropriate metrics such as accuracy, precision, recall, and clustering validation methods.
- Create insightful data visualizations using Python libraries like Matplotlib to represent data effectively and communicate analytical findings.

Unit 1	Introduction to Data Mining: Reasons for using Data Mining, Definition of Data Mining, Types of Data Beneficial for Mining, Patterns and Insights Extracted Through Data Mining, Technologies That Enhance Data Mining, Applications Benefiting from Data Mining, Major Issues in Data Mining, Definition and Importance of Data Mining, Types of Data (Structured, Semi,Structured, Unstructured), Data Mining Tasks (Classification, Clustering, Association Analysis, etc.), Technologies Used in Data Mining (Machine Learning, Statistics, Databases), Applications in Business, Science, Healthcare, etc., Challenges like Scalability, High Dimensionality, Data Quality
Unit 2	Data Pre-processing Overview: An Overview, Data Cleaning, Data Integration, Data Reduction, Data Transformation and Data Discretization, Data Normalization & Noise Removal, Importance of Data Pre-processing, Techniques for Handling Missing Data (Deletion, Imputation), Noise Reduction Methods (Binning, Regression), Data Integration Strategies (Schema Integration, Redundancy Elimination), Data Reduction Techniques (Dimensionality Reduction, Numerosity Reduction), Data Transformation Methods (Normalization, Smoothing, Aggregation), Discretization Methods (Binning, Histogram Analysis)
Unit 3	Basic Concepts in Frequent Pattern Mining: Basic Concepts, Frequent Itemsets, Closed Itemsets, Association Rules, Support and Confidence Measures, Market Basket Analysis, Applications of Frequent Pattern Mining
Unit 4	Frequent Itemset Mining Methods: Frequent Itemset Mining Methods, Apriori Algorithm, FP,Growth Algorithm, ECLAT Algorithm, Candidate Generation and Pruning, Comparison of Algorithms in Terms of Efficiency and Scalability
Unit 5	Pattern Evaluation Methods: Pattern Evaluation Methods, Objective Measures (Lift, Leverage, Conviction), Subjective Measures (Unexpectedness, Actionability), Statistical Significance Testing, Overfitting and Underfitting in Pattern Discovery, Measures for Correlation and Dependency
Unit 6	Introduction to Classification and Decision Trees: Basic Concepts, Decision Tree Induction, Difference Between Classification and Prediction, Decision Tree Construction Process, Splitting Criteria (Information

	Gain, Gini Index, Gain Ratio), Tree Pruning Techniques, Advantages and Disadvantages of Decision Trees
Unit 7	Bayesian and Rule,Based Classification: Bayes Classification Methods, Rule,Based Classification, Naive Bayes Classifier Assumptions and Calculations, Bayesian Belief Networks, Creating Classification Rules, Rule Extraction from Trees and Other Models, Accuracy and Interpretability of Rule,Based Classifiers
Unit 8	Model Evaluation and Lazy Learners: Model Evaluation and Selection, Lazy Learners (Learning from Your Neighbors), Overfitting vs. Underfitting, Cross,Validation Techniques (k,Fold, Leave,One,Out), Performance Metrics (Accuracy, Precision, Recall, F1 Score), ROC Curves and AUC, K,Nearest Neighbor Algorithm, Distance Measures (Euclidean, Manhattan), Impact of k Value on Classification
Unit 9	Introduction to Cluster Analysis: Cluster Analysis, Definition and Objectives of Clustering, Similarity and Dissimilarity Measures, Requirements for Clustering Algorithms (Scalability, Ability to Deal with Different Types of Attributes), Applications in Market Segmentation, Image Processing, Biological Data Analysis
Unit 10	Partitioning and Hierarchical Methods: Partitioning Methods, Hierarchical Methods, K,Means Clustering Algorithm Steps, K,Medoids and PAM Algorithm, Strengths and Weaknesses of Partitioning Methods, Agglomerative vs. Divisive Hierarchical Clustering, Dendrogram Interpretation, Linkage Criteria (Single, Complete, Average)
Unit 11	Density,Based and Grid,Based Methods: Density,Based Methods, Grid,Based Methods, DBSCAN Algorithm Concepts, Identifying Noise and Outliers, OPTICS Algorithm for Variable Density Clusters, Grid,Based Clustering with STING, CLIQUE Algorithm for Subspace Clustering, Advantages of Density and Grid,Based Methods in Handling Complex Data
Unit 12	Evaluation of Clustering: Evaluation of Clustering, Intrinsic Methods (Silhouette Coefficient, SSE), Extrinsic Methods (Purity, Rand Index), Determining the Optimal Number of Clusters, Dealing with Cluster Validity Issues, Comparing Different Clustering Algorithms, Visual Assessment of Cluster Tendency (VAT)
Unit 13	Introduction to Data Visualization with Python: Introduction to Matplotlib, Basic Plotting with Matplotlib, Dataset on Immigration to Canada, Line Plots, Setting Up the Python Environment (Installing Matplotlib), Understanding the Matplotlib Anatomy of a Plot, Loading and Exploring the Dataset, Creating Simple Line Plots, Customizing Plots with Labels, Titles, and Legends, Saving Plots to Files

Unit 14	Advanced Visualization Techniques: Basic Visualization Tools, Area Plots, Histograms, Bar Charts, Specialized Visualization Tools, Pie Charts, Box Plots, Scatter Plots, Bubble Plots, Creating and Customizing Area Plots, Constructing Histograms for Data Distribution Analysis, Developing Bar Charts for Categorical Data, Designing Pie Charts to Represent Proportions, Generating Box Plots for Statistical Summaries, Plotting Scatter and Bubble Plots for Correlation Analysis, Combining Multiple Plots in a Single Figure, Best Practices in Data Visualization for Effective Communication

TEXTBOOKS:

- J. Han, M. Kamber, "Data Mining: Concepts and Techniques", Harcourt India / Morgan Kauffman, 2011.
- Richard Cotton,Learning R, September 2013, O'Reilly Media, Inc., ISBN: 9781449357108

REFERENCE BOOKS:

- K.P. Soman, ShyamDiwakar, V. Ajay: Insight into Data Mining – Theory and Practice, PHI, 2012
- David Hand, Heikki Manila, Padhraic Smyth, "Principles of Data Mining", PHI 2012.
- W.H.Inmon, "Building the Data Warehouse", 3rd Edition, Wiley, 2011.
- Alex Bezon, Stephen J.Smith, "Data Warehousing, Data Mining & OLAP", McGraw-Hill Edition, 2001

Course Name: QUANTUM ALGORITHMS

Credits: 4

Course Description:

This course provides an in-depth exploration of cutting-edge computational paradigms bridging quantum mechanics, advanced mathematics, and computational theory. Learners will master fundamental concepts including quantum information principles, linear algebra foundations, quantum circuit design, and sophisticated quantum algorithmic approaches. The curriculum covers critical domains such as quantum programming languages, quantum gates, state space evolution, and advanced quantum algorithms like factoring and search techniques.

Course Objectives:

This Course will enable learners to:

- Comprehend the evolution, scope, and future developments of quantum computing and its significance in modern technology.
- Explore the foundational models of classical computation, their limitations, and how quantum models overcome these limitations.
- Gain proficiency in linear algebra concepts essential for quantum mechanics and quantum computing.
- Develop an understanding of quantum mechanics principles and their applications in constructing quantum systems.
- Analyse quantum algorithms, circuits, and their efficiency compared to classical algorithms.

Course Outcomes:

At the end of the course the learner will be able to:

- Explain the history, scope, and potential future developments in quantum computing.
- Compare classical computation models (Turing machines, circuits) with quantum computation and identify the limitations of classical models.
- Apply advanced linear algebra concepts, such as eigenvalues, tensor products, and unitary operators, to solve problems in quantum computing.
- Design and implement quantum algorithms and circuits using quantum programming languages such as Qiskit and Q#.

- Analyse quantum algorithms like Shor's and Grover's, evaluate their efficiency, and compare them with classical counterparts for problem-solving.

Unit 1	Introduction to Quantum Computing: History of Quantum Computing, Scope of Quantum Computing, Future of Quantum Computing, Evolution of quantum computing, Key milestones, Current state of quantum technology, Potential future developments.
Unit 2	Classical Computation Models: Models of Computation, Turing Machines, Circuits, Overview of classical computation, Functioning of Turing machines, Logic circuits and Boolean algebra, Limitations of classical models
Unit 3	Analysis of Computational Problems: Complexity Classes, Algorithm Efficiency, Computational Hardness vs. NP problem, Big O notation, Deterministic vs. nondeterministic algorithms, Examples of computationally hard problems.
Unit 4	Quantum Programming Languages: Introduction to Quantum Programming, Qiskit,Q#, Setting up quantum programming environments, Basic syntax and features of Qiskit, Basic syntax and features of Q#,Writing simple quantum programs
Unit 5	Linear Algebra Essentials: Bases and Linear Independence, Linear Operators and Matrices, Vector spaces, Basis vectors ,Matrix algebra, Linear transformations.
Unit 6	Advanced Linear Algebra for Quantum Computing: Pauli Matrices, Inner Products, Eigenvectors and Eigenvalues, Adjoint and Hermitian Operators, Properties of Pauli Matrices, Inner product spaces, Diagonalization of operators, Hermitian and unitary operators in quantum mechanics.
Unit 7	Tensor Products and Operator Decompositions: Tensor Products, Operator Functions, Polar and Singular Value Decompositions Constructing multi-qubit systems, applying functions to operators, Understanding SVD and its applications in quantum computing.
Unit 8	Quantum Mechanics for Quantum Computing: State Space, Evolution, Quantum Measurement, Distinguishing Quantum States, Phase, Composite Systems, Quantum state vectors, Schrödinger equation, Measurement postulates, Probability amplitudes, Quantum superposition, Global and relative phase, Entanglement, Bell states.

Unit 9	Quantum Algorithms and Circuits: Overview of Quantum Algorithms, Quantum Circuit Model, Notable quantum algorithms, Representing algorithms with circuits, Quantum circuit diagrams.
Unit 10	Quantum Gates and Operations: Single Qubit Operations, Controlled Operations, Measurement, Universal Quantum Gates, Pauli-X, Y, Z gates, Hadamard gate, NOT gate, Measurement in different bases, Gate universality.
Unit 11	Simulation of Quantum Systems: Simulating Quantum Phenomena, Quantum Simulation Algorithms, Applications, Hamiltonian simulation, Digital and analog simulation methods, Simulating molecules and materials.
Unit 12	Quantum Fourier Transform and Phase Estimation: The Quantum Fourier Transform (QFT) Phase Estimation, Implementing QFT, Applications of QFT, Phase estimation algorithm, Precision and efficiency.
Unit 13	Algorithms Based on Hidden Subgroup Problems: Order-Finding, Factoring, Period-Finding, Discrete Logarithms, The Hidden Subgroup Problem, Shor's algorithm for integer factorization, Periodicity in quantum algorithms, solving discrete logarithms, Understanding the HSP framework.
Unit 14	Quantum Search Algorithms and Their Limits: The Quantum Search Algorithm, Quantum Search Applications, Quantum Counting, NP-Complete Problem Solutions, Optimality Proofs, Black Box Algorithm Limits, Grover's Algorithm, Database search problem, Algorithm steps and implementation, Speedup over classical search, Quantum counting techniques, Approaches to NP-complete problems, Proofs of optimality, Limitations imposed by quantum no-go theorems

TEXTBOOKS:

- Nielsen, M.A., & Chuang, I.L. (2010). Quantum Computation and Quantum Information. Cambridge University Press. ISBN: 978-1107002173.
- McMahon, D. (2007). Quantum Computing Explained. Wiley-Interscience. ISBN: 978-0470096994.

REFERENCE BOOKS:

- Schumacher, B., & Westmoreland, M. (2010). Quantum Processes, Systems, and Information. Cambridge University Press. ISBN: 978-0521875349.
- Bernhardt, C. (2019). Quantum Computing for Everyone. The MIT Press. ISBN: 978-0262039253.
- Hidary, J.D. (2019). Quantum Computing: An Applied Approach. Springer. ISBN: 978-3030239213.
- Rieffel, E., & Polak, W. (2011). Quantum Computing: A Gentle Introduction. The MIT Press. ISBN: 978-0262015066.

Course Name: BLOCK CHAIN TECHNOLOGIES

Credits: 4

Course Description:

This course provides an in-depth exploration of decentralized digital technologies, covering fundamental principles to advanced implementation strategies. Learners will master critical concepts including blockchain architecture, consensus mechanisms, cryptocurrency fundamentals, and emerging blockchain platforms like Bitcoin, Ethereum, and Hyperledger. The curriculum covers sophisticated domains such as smart contracts, digital tokens, development frameworks, and specialized blockchain applications across various ecosystems.

Course Objectives:

This Course will enable learners to:

- Understand the fundamentals of blockchain technology, its structure, components, and necessity.
- Explore the various data structures, validation methods, and consensus mechanisms used in blockchain.
- Analyse decentralization, disintermediation, and their impact on traditional systems.
- Examine the ecosystems, platforms, and technologies that enable blockchain applications.
- Apply blockchain concepts to real-world scenarios, including Bitcoin, Ethereum, and other blockchain-based platforms.

Course Outcome:

At the end of the course the learner will be able to:

- Explain the structure, components, and need for blockchain technology.
- Analyse the data structures (Merkle Trees, hash functions) and how data is distributed across blockchain networks.
- Compare various consensus mechanisms (PoW, PoS, PoET) used for block validation in blockchain systems.
- Illustrate the concepts of decentralization and disintermediation in blockchain and their impact on peer-to-peer networks.

- Describe the components of the blockchain ecosystem, including storage, communication, and computation layers.

Unit 1	Introduction to Blockchain Technology: Definition of Blockchain, The Structure of Blockchain, Need for Blockchain, Basic Structure and Components of Blockchain.
Unit 2	Data Structures in Blockchain: Data Distribution in Blockchain, Blocks and Chains, Merkle Trees, Hash Functions, How Data is Distributed Across Nodes.
Unit 3	Block Validation and Consensus Mechanisms: Block Validation, Consensus Mechanisms, Validation Processes, Proof of Work (PoW), Proof of Stake (PoS), Proof of Activity, Proof of Elapsed Time, Proof of Burn.
Unit 4	Decentralization and Disintermediation: Decentralization, Disintermediation, Removal of Central Authorities, Peer, to, Peer Networks, Impact on Traditional Systems.
Unit 5	Blockchain Ecosystem Components: Ecosystem Components, Storage, Communication, Computation.
Unit 6	Bitcoin Fundamentals: Bitcoin Working, Bitcoin Transactions, Bitcoin Mining, How Bitcoin Operates, Transaction Lifecycle, Mining Process and Rewards.
Unit 7	Bitcoin Economics and Community: Value of Bitcoin, Community, Politics, and Regulations, Advantages and Disadvantages , Factors Influencing Bitcoin's Value, Regulatory Environment, Benefits and Limitations of Bitcoin.
Unit 8	Ethereum Overview and DApps: Overview of Ethereum, Decentralized Applications (DApps), Differences Between Ethereum and Bitcoin, Smart Contracts, Use Cases of Dapps.
Unit 9	Components of Ethereum: Smart Contracts, Ether, Ethereum Clients, Ethereum Virtual Machine (EVM), Etherscripter, Functionality of Smart Contracts, Role of Ether as Cryptocurrency, Types of Ethereum Clients, Operation of EVM, Introduction to Etherscripter
Unit 10	Digital Tokens and Initial Coin Offerings: Digital Tokens Overview , Initial Coin Offering (ICO) OmiseGO, EOS, Tether, Purpose and Types of Digital Tokens, ICO Process and Regulations, Case Studies of Specific Tokens
Unit 11	Ethereum Wallets and Tools: MetaMask, Wallet Seed, MetaMask Transactions, Mist Wallet, Overview,

	Features, Setting Up MetaMask Wallet, Securing Wallet Seed, Performing Transactions, Features of Mist Wallet.
Unit 12	Truffle Framework for DApp Development: Features of Truffle, Development Truffle Boxes, Community Truffle Box, Setting Up Truffle Utilizing Truffle Boxes for Development, Community Resources and Support.
Unit 13	Hyperledger Platforms: Hyperledger Fabric, Introduction, Fabric vs. Ethereum, Hyperledger Iroha, Features, Hyperledger Sawtooth, Components, Proof of Elapsed Time (PoET), Architecture Hyperledger Fabric, Comparing Fabric with Ethereum, Unique Features of Hyperledger Iroha Components of Sawtooth Platform, Understanding PoET Consensus Mechanism.
Unit 14	Alternative Blockchain Platforms and Future Technologies: Multichain, HydraChain, Future Blockchain Technologies, IOTA, Corda, Chain Core, Blockchain Frameworks,CoCo Framework, Tierion, BigchainDB, Case Study: Blockchain in IoT,Features of Multichain and HydraChain,I nnovations in Future Blockchain Technologies, Overview of Blockchain Frameworks, Application of Blockchain in IoT through Case Study

TEXTBOOKS:

- Mastering Blockchain, Third Edition, Published by Packt Publishing Ltd, Published 2020, Imran Bashir
- Solidity Programming Essentials, First Edition, Published by Packt Publishing Ltd, April 2018 Blockchain for Dummies, Manav Gupta, IBM Limited Edition, John Wiley & Sons, Inc. 2017
- Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller, and Steven Gold Feder. Bitcoin and cryptocurrency technologies: a comprehensive introduction. Princeton University Press, 2016.

REFERENCE BOOKS:

- Joseph Bonneau et al, SoK: Research perspectives and challenges for Bitcoin and cryptocurrency, IEEE Symposium on security and Privacy, 2015.

Semester: III

Course Name: CYBER SECURITY

Credits: 4

Course Description:

This course provides a comprehensive overview of computer security and cybercrime, exploring essential concepts, mechanisms, and policies to safeguard information systems. It covers fundamental security goals, cryptographic techniques, and access control mechanisms, with detailed discussions on RSA cryptosystems, cryptographic hash functions, and discrete logarithms. Learners will delve into network and transport layer security protocols like IPSec and SSL, as well as practical cybercrime methodologies, including social engineering, phishing, and malware attacks. Legal perspectives on cybercrime, both in India and globally, are also examined to provide a holistic understanding of the cyber threat landscape. By the end of the course, learners will gain knowledge and tools to analyse, design, and implement robust security solutions while understanding the legal frameworks surrounding cybercrime.

Course Objectives:

This Course will enable learners to:

- Infer the fundamental principles of computer security, including goals, services, mechanisms, and attacks.
- Explore classical and modern cryptographic techniques, including RSA, cryptographic hash functions, and discrete logarithms.
- Examine security protocols and mechanisms at different layers of the network, such as IPSec, SSL, and VPNs.
- Analyse various aspects of cybercrime, including its classification, planning, and execution, and explore its legal implications.
- Gain knowledge about tools, techniques, and laws for combating cybercrime, with a focus on the Indian IT Act and global perspectives.

Course Outcomes:

At the end of the course the learner will be able to:

- Demonstrate knowledge of the principles and mechanisms of computer security and evaluate potential security threats.

- Apply cryptographic methods and protocols to secure data transmission and storage.
- Implement security strategies at different network layers using appropriate tools and techniques.
- Analyse cybercrimes and propose measures for preventing and mitigating attacks.
- Evaluate the relevance of cyber laws and ethical practices in the context of national and international frameworks.

Unit 1	Overview of Computer Security: An Overview of Computer Security, Security Goals, Security Services, Security Mechanisms, Security Attacks.
Unit 2	Access Control and Security Policies: Access Control Matrix, Policy, Security Policies, Confidentiality Policies, Integrity Policies, Hybrid Policies.
Unit 3	Classical Cryptography: Classical Cryptography, Substitution Ciphers, Permutation Ciphers, Block Ciphers.
Unit 4	RSA Cryptosystem RSA Operations, RSA Performance, RSA Applications, RSA Practical Issues, Public Key Cryptography Standard (PKCS).
Unit 5	Cryptographic Hash Functions: Introduction to Cryptographic Hash, Properties, Construction, Applications, Performance, The Birthday Attack.
Unit 6	Discrete Logarithms and Applications: Discrete Logarithm: Introduction, Diffie-Hellman Key Exchange, Other Applications.
Unit 7	Security at the Network Layer: IPSec: Security at the Network Layer, Security at Different Layers: Pros and Cons, IPSec in Action, Internet Key Exchange (IKE) Protocol, Security Policy and IPSec.

Unit 8	Security at the Transport Layer: Introduction to Security at the Transport Layer, SSL Handshake Protocol, SSL Record Layer Protocol, OpenSSL.
Unit 9	Introduction to Cybercrime: Definition and Origins of the Word, Cybercrime and Information Security, reasons for Cybercriminals, Classifications of Cyber Crimes.
Unit 10	Cybercrime Legal Perspectives Cybercrime: The Legal Perspectives, Cybercrimes: An Indian Perspective, Cybercrime and the Indian ITA 2000, A Global Perspective on Cybercrimes.
Unit 11	Cyber Offenses: Planning Attacks: How Criminals Plan Attacks, Social Engineering, Cyberstalking, Cybercafe and Cybercrime, Botnets (The Fuel for Cybercrime), Attack Vector.
Unit 12	Cybercrime Tools and Methods – I: Proxy Servers and Anonymizers, Phishing, Password Cracking, Keyloggers and Spywares.
Unit 13	Cybercrime Tools and Methods – II: Virus and Worms, Trojan Horses and Backdoors, Steganography, DoS and DDoS Attacks, SQL Injection, Buffer Overflow, Attacks on Wireless Networks.
Unit 14	Cyber Laws: Cybercrime and the Legal Landscape Around the World, Why Do We Need Cyberlaws: The Indian Context, The Indian IT Act.

TEXTBOOKS:

- William Stallings - Cryptography and Network Security 4th edition
- Introduction to Modern Cryptography – Jonathan Katz, Yehuda Lindell – Second Edition.
- Network Security – Charlie Kaufman, Radia Perlman, Mike Speciner – Second Edition
- Fundamentals of Network Security – Eric Maiwald – Mc Graw Hill Edition

- Cryptography, Network Security and Cyber Laws – Bernard Menezes, Cengage Learning,
- 2010 edition
- Cyber Security Understanding Cyber Crimes, Computer Forensics and Legal Perspectives –Nina Godbole, Sunit Belapure, Wiley : April 2011 India Publications Released

REFERENCE BOOKS:

- Cryptography and Network Security: Atul Kahate, Mc Graw Hill Edition.

Course Name: EMBEDDED SYSTEMS AND INTERNET OF THINGS
Credits: 3

Course Description:

This course explores the integration of cloud storage, data analytics, and advanced technologies in IoT applications. It begins with the fundamentals of cloud storage, IoT storage models, and optimization techniques, progressing to IoT data analytics and the use of real-time and edge analytics tools. The curriculum delves into IoT web servers, blockchain technology for IoT security, and cloud computing infrastructure, emphasizing resource management and scalability. It also covers big data management challenges in IoT, alongside applications in smart homes, environments, healthcare, and agriculture. By the end, learners will understand how to leverage IoT technologies to design innovative, efficient, and sustainable solutions for real-world challenges.

Course Objectives:

This Course will enable learners to:

- Infer the principles and architectures of cloud storage, IoT integration, and the fundamentals of data storage, optimization, and redundancy.
- Analyse IoT data using various analytics types (descriptive, predictive, prescriptive) and apply edge analytics and real-time tools for data-driven solutions.
- Explore the design and functionality of IoT web servers, blockchain integration, and scalable security mechanisms.
- Develop smart applications across domains like healthcare, agriculture, and environment using IoT, big data, and cloud infrastructures.

Course Outcomes:

At the end of the course the learner will be able to:

- Examine IoT data analytics frameworks and tools to derive insights for real-time applications.
- Assess blockchain-based solutions and security measures for IoT applications, ensuring decentralized and secure interactions.
- Utilize IoT architectures in developing scalable and efficient web server applications and RESTful web services.

- Design and implement IoT-enabled smart systems in healthcare, agriculture, and environmental monitoring for sustainable solutions.

Unit 1	Introduction to Cloud Storage for IoT: Fundamentals of cloud storage, IoT cloud storage models, Cloud storage architecture, Storage optimization techniques, Data redundancy and backup strategies
Unit 2	Data Analytics Fundamentals in IoT: Introduction to IoT data analytics, Types of analytics (descriptive, predictive, prescriptive), Real-time analytics, Edge analytics, Analytics tools and platforms
Unit 3	Web Servers for IoT Applications: IoT web server architecture, RESTful web services, MQTT servers, CoAP servers, Server security and scalability
Unit 4	Blockchain Technology in IoT: Blockchain fundamentals, Smart contracts, Blockchain-IoT integration, Security aspects, Decentralized applications (DApps)
Unit 5	Cloud Computing Infrastructure: Cloud service models (IaaS, PaaS, SaaS), Cloud deployment models, Cloud-IoT integration, Resource management, Performance optimization
Unit 6	Big Data Management for IoT: Big data characteristics, Data collection and preprocessing, Data storage systems, Data processing frameworks, Scalability challenges
Unit 7	Smart Home Applications: Smart home architecture, Home automation systems, Energy management, Security and surveillance, Integration protocols
Unit 8	Smart Environment Solutions: Environmental monitoring, Pollution control systems, Weather monitoring, Resource management, Sustainable solutions

Unit 9	Smart Healthcare Systems: Healthcare IoT architecture, Patient monitoring systems, Medical device integration, Health data management, Telehealth solutions
Unit 10	Smart Agriculture Applications: Precision agriculture, Crop monitoring systems, Irrigation management, Weather monitoring, Yield optimization

TEXTBOOKS:

- Andrew N Sloss, D. Symes, C. Wright, "ARM System Developers Guide", Morgan Kauffman/ Elsevier,2006.
- Arshdeep Bahga, Vijay Madisetti, "Internet of Things – A hands-on approach",

  Universities Press, 2015

REFERENCE BOOKS:

- Olivier Hersent, David Boswarthick, Omar Elloumi, "The Internet of Things – Key applications and Protocols", Wiley, 2012.
- Microcontroller and Embedded Systems", Pearson Education, Second edition, 2007.

Course Name: CLOUD SECURITY & INFRASTRUCTURE

Credits: 4

Course Description:

This course provides an in-depth study of cloud computing, focusing on its architecture, models, solutions, and security. It begins with the fundamental technologies and system models for distributed and cloud computing, followed by an exploration of cloud service models (IaaS, PaaS, SaaS) and deployment strategies. The curriculum delves into virtualization concepts, tools, and mechanisms, with applications in data centre automation and resource management. Learners will learn about cloud infrastructure design, programming paradigms like MapReduce, and platforms such as Hadoop, AWS, and OpenStack. The course also emphasizes cloud security, covering challenges, governance, risk management, and advanced topics like virtual machine security and identity management, equipping learners with the skills to design and manage secure cloud-based systems.

Course Objectives:

This Course will enable learners to:

- Comprehend cloud computing architectures, models, and reference frameworks, including NIST standards.
- Explore virtualization concepts, types, and mechanisms, and their role in data center automation and cloud infrastructure.
- Analyse and implement cloud-based solutions and programming models such as MapReduce, Hadoop, and cloud software environments.
- Evaluate the security challenges in the cloud environment, including application security, data security, and virtual machine security.
- Develop solutions for managing cloud resources and addressing design challenges like resource provisioning and inter-cloud resource management.

Course Outcomes:

At the end of the course the learner will be able to:

- Implement cloud computing models and ecosystems to solve real-world problems using public and private cloud infrastructures.

- Examine the role of virtualization in resource management and data center automation to optimize cloud operations.
- Assess programming paradigms like MapReduce and tools such as Hadoop to address challenges in distributed computing.
- Utilize cloud software environments like Google App Engine, AWS, and OpenStack to build scalable and efficient cloud applications.
- Design secure cloud architectures by addressing security challenges and implementing identity management, access control, and autonomic security measures.

Unit 1	Cloud Architecture: Technologies for Network-Based System, System Models for Distributed and Cloud Computing, NIST Cloud Computing Reference Architecture
Unit 2	Cloud Model: Cloud Models: Characteristics, Cloud Services, Cloud models (IaaS, PaaS, SaaS), Public vs Private Cloud
Unit 3	Cloud Solutions: Cloud Solutions, Cloud ecosystem, Service management, Computing on demand.
Unit 4	Virtualization - I: Basics of Virtualization, Types of Virtualization, Implementation Levels of Virtualization
Unit 5	Virtualization - II: Virtualization Structures, Tools and Mechanisms, Virtualization of CPU, Memory, I/O Devices
Unit 6	Virtualization - III: Virtual Clusters and Resource management, Virtualization for Data-center Automation.
Unit 7	Cloud Infrastructure: Architectural Design of Compute and Storage Clouds, Layered Cloud Architecture Development

Unit 8	Design Challenges: Design Challenges - Inter Cloud Resource Management, Resource Provisioning and Platform Deployment, Global Exchange of Cloud Resources.
Unit 9	Programming Model: Parallel and Distributed Programming Paradigms, Map Reduce, Twister and Iterative MapReduce
Unit 10	Hadoop: Hadoop Library from Apache, Mapping Applications
Unit 11	Programming Support: Programming Support, Google App Engine, Amazon AWS, Cloud Software Environments, Eucalyptus, Open Nebula, Open Stack, Aneka, CloudSim
Unit 12	Security in the Cloud - I: Security Overview, Cloud Security Challenges and Risks, Software-as-a-Service Security, Security Governance, Risk Management.
Unit 13	Security in the Cloud - II: Security Monitoring, Security Architecture Design, Data Security, Application Security
Unit 14	Security in the Cloud - III: Virtual Machine Security, Identity Management and Access Control, Autonomic Security.

TEXTBOOKS:

- Kai Hwang, Geoffrey C Fox, Jack G Dongarra, "Distributed and Cloud Computing, From Parallel Processing to the Internet of Things", Morgan Kaufmann Publishers, 2012.
- John W.Rittinghouse and James F.Ransome, "Cloud Computing: Implementation, Management, and Security", CRC Press, 2010

REFERENCE BOOKS:

- Toby Velte, Anthony Velte, Robert Elsenpeter, “Cloud Computing, A Practical Approach”, TMH, 2009.
- Kumar Saurabh, “Cloud Computing – insights into New-Era Infrastructure”, Wiley India, 2011.

Course Name: DATA ANALYTICS & VISUALIZATION

Credits: 4

Course Description:

This course provides a detailed introduction to statistical programming with R, covering essential tools and techniques for data analysis, modelling, and visualization. Learners will begin with the basics of R programming, data structures, and preprocessing techniques, progressing to statistical tests, correlation, regression, and advanced hypothesis testing. The course also explores topic modelling, including Latent Dirichlet Allocation (LDA) and its applications in text mining and natural language processing. Additionally, learners will learn about random graphs, their properties, and applications in social networks, epidemiology, and network science. The curriculum emphasizes practical skills through hands-on projects, equipping learners to analyse real-world datasets and present their findings effectively using R.

Course Objectives:

This Course will enable learners to:

- Gain a solid foundation in R programming, including syntax, data types, control structures, and functions.
- Explore and modify datasets using R, focusing on data cleaning, preprocessing, and advanced manipulation techniques.
- Perform basic and advanced statistical tests in R, including T-tests, Chi-squared tests, and correlation analysis.
- Evaluate and tune topic models for optimal performance, using coherence scores and perplexity.
- Apply graph visualization techniques using R packages and analyze network metrics and community detection.

Course Outcomes:

At the end of the course the learner will be able to:

- Utilize basic syntax, control structures, and functions in R to perform statistical computing tasks.

- Conduct data exploration, cleaning, and pre-processing using advanced R packages, and manipulate large datasets effectively.
- Perform and interpret the results of statistical tests and regression models, understanding their assumptions and implications.
- Implement topic modelling techniques and evaluate their performance using coherence scores and other metrics.
- Visualize and interpret the structure and properties of random graphs, applying graph theory to real-world data analysis scenarios.

Unit 1	Introduction to Statistical Programming with R: Introduction to R programming language and statistical computing, Installing and setting up R and RStudio, Basic syntax and operations in R, Understanding data types: Numeric, Integer, Logical, Character, Complex, etc, Control structures: if, else, for loops, while loops, and apply functions, Functions in R: Creating and using custom functions, Handling data files: Reading, writing, and saving data, Data structures in R: Vectors, matrices, data frames, and lists, Transposing data, handling missing values, and naming columns
Unit 2	Working with Data in R: Exploring datasets in R: Loading, viewing, and modifying data, Introduction to data cleaning and preprocessing techniques in R, Handling missing values and outliers in data, Renaming and restructuring data, Advanced data manipulation: Subsetting, merging, and reshaping data, Working with large datasets and efficient data handling techniques, R packages for data manipulation: `dplyr`, `tidyr`, and `datatable’
Unit 3	Statistical Tests in R: Introduction to basic statistical tests in R: Mean, Variance, and Quantile, Implementing the T-test: Equal and unequal variances, paired T-test, Step-by-step walkthrough of T-tests and Chi-squared tests, Performing Chi-squared tests in R: Testing independence and goodness of fit, Distance metrics: Euclidean, Manhattan, and other distance measures, Using R functions to perform these tests and interpreting results
Unit 4	Correlation and Regression Analysis: Introduction to correlation and its significance in statistical analysis, Pearson's and Spearman’s correlation in R, Introduction to regression analysis: Simple linear regression, Understanding multiple regression models and their assumptions, Linear and logistic regression: Models and coefficients, Interpretation of regression coefficients and R squared values, Visualizing regression models: Scatter plots and regression lines, Inference and learning from regression analysis
Unit 5	Advanced Statistical Tests and Distance Metrics: In-depth study of T-tests, Chi-squared tests, and their assumptions, Advanced techniques for hypothesis testing, Understanding the various distance metrics in data analysis, Calculating and interpreting Euclidean distance, Manhattan

	distance, and cosine similarity, Applications of distance metrics in clustering and classification
Unit 6	Introduction to Topic Modeling: Introduction to natural language processing (NLP) and text mining, Overview of topic modeling and its applications, Hidden Markov Models (HMM) for sequence prediction and classification, Latent Semantic Indexing (LSI) for topic discovery, Introduction to Probabilistic Latent Semantic Indexing (PLSI), Latent Dirichlet Allocation (LDA): Theory and implementation, Gibbs Sampling for LDA and understanding its role in topic modeling
Unit 7	Advanced Topic Modeling Techniques: Deep dive into Latent Dirichlet Allocation (LDA) for text classification, Implementing LDA using R and understanding its parameters, Understanding and applying Gibbs Sampling for LDA, Model evaluation techniques: Coherence score, perplexity, Tuning LDA models for optimal performance, Using topic modeling for document clustering and recommendation systems
Unit 8	Introduction to Random Graphs: Introduction to graph theory and its applications in data science, Basic models of random graphs: Erdős–Rényi model, Homogeneous graphs and their properties, Understanding fixed degree sequences in random graphs, Intersectiographs: Theory and applications, Digraphs and hypergraphs: Definitions and characteristics
Unit 9	Random Graphs in Data Science: Analyzing random graphs for social networks and other applications, Understanding graph connectivity, clusters, and communities, Exploring tools and libraries for graph analysis (eg, `igraph`, `statnet`), Applications of random graphs in network science, epidemiology, and biology
Unit 10	Fixed Degree Sequence in Random Graphs: The concept of degree sequences in random graphs, Generating random graphs with a fixed degree sequence, Applications of fixed degree sequence models in social networks and other fields, Comparing random graphs with fixed degree sequences to real-world networks
Unit 11	Advanced Graph Models: Introduction to advanced graph models: Preferential attachment, small-world networks, Scale-free networks and their significance in real-world systems, Understanding random graphs in the context of network dynamics and evolution, Analyzing directed graphs (digraphs) and their applications
Unit 12	Graph Visualization and Tools: Introduction to graph visualization techniques and tools, visualizing networks and graphs using R packages (`igraph`, `ggraph`), Creating interactive graphs with `visNetwork` and other libraries, Visualizing large graphs and networks: Techniques and challenges

Unit 13	Data Analysis with Random Graphs: Case studies of random graphs in data analysis, Applying graph theory to large datasets (social media networks, citation networks), Performance metrics in network analysis: Centrality, betweenness, closeness, Clustering and community detection in random graphs
Unit 14	Final Project and Real-World Applications: Final project: Apply statistical analysis, topic modeling, and random graph theory to a real-world dataset, Implement a comprehensive analysis using R, focusing on data cleaning, modeling, and visualization, Review of best practices in statistical programming, modeling, and graph theory, Presenting results, writing reports, and communicating findings

TEXTBOOKS:

- Crawley, M. J. (2012). The R Book (2nd ed.). Hoboken, NJ: Wiley. A comprehensive guide to programming and statistical analysis using R.
- Wickham, H., & Grolemund, G. (2017). R for Data Science: Import, Tidy, Transform, Visualize, and Model Data. Sebastopol, CA: O'Reilly Media. An introduction to data manipulation and visualization with R.
- James, G., Witten, D., Hastie, T., & Tibshirani, R. (2021). An Introduction to Statistical Learning: With Applications in R (2nd ed.). New York, NY: Springer. A modern take on statistical and machine learning techniques in R.

REFERENCE BOOKS:

- Adler, J. (2010). R in a Nutshell: A Desktop Quick Reference. Sebastopol, CA: O'Reilly Media. A concise reference guide to the R programming language.

Course Name: ARTIFICIAL INTELLIGENCE
Credits: 4

Course Description:

This course provides a comprehensive introduction to Artificial Intelligence (AI), covering its fundamental concepts, techniques, and applications. Learners will explore problem-solving strategies, knowledge representation, reasoning under uncertainty, and game-playing algorithms. The curriculum delves into expert systems, natural language processing, machine learning, and deep learning, with hands-on opportunities to build basic AI models and neural networks. Advanced topics such as robotics, reinforcement learning, and heuristic search techniques are also addressed, along with a focus on AI ethics and responsible practices. The course concludes with a capstone project, enabling learners to apply their knowledge to real-world scenarios and develop innovative AI solutions.

Course Objectives:

This Course will enable learners to:

- Grasp the basics of artificial intelligence, including agent concepts, environments, and problem-solving techniques.
- Learn techniques for knowledge representation, predicate logic, and computable functions.
- Apply algorithms for game playing, such as minimax and alpha-beta pruning, and understand learning concepts in AI.
- Analyse AI applications across various fields and understand emerging trends and ethical challenges.
- Gain practical knowledge in machine learning and deep learning, including training models and creating neural networks.

Course Outcomes:

At the end of the course the learner will be able to:

- Implement informed and uninformed search strategies to solve AI problems.
- Evaluate different techniques and issues in knowledge representation using predicate logic and rules.
- Develop game-playing algorithms like minimax and iterative deepening for AI applications.

- Assess machine learning models using evaluation metrics and improve their performance.
- Analyse the integration of AI with robotics, focusing on control mechanisms and sensor use.

Unit 1	Problem Solving: Introduction to Artificial Intelligence (AI), Concepts of agents and environments, Problem formulation and problem spaces, Search strategies: informed (heuristic) and uninformed search techniques
Unit 2	Knowledge Representation: Issues and techniques in knowledge representation, Predicate logic: representing facts, instances, and relationships, Computable functions and predicates, Resolution and knowledge representation using rules
Unit 3	Uncertainty: Symbolic reasoning under uncertainty, Statistical reasoning methods, Weak and strong slot-and-filler structures
Unit 4	Game Playing and Learning: Overview of game-playing in AI, Minimax algorithm and alpha-beta pruning, Iterative deepening strategies, Introduction to learning concepts in AI
Unit 5	Expert Systems: Architecture and roles of expert systems, Examples such as MYCIN, XOON, DART, Case studies on real-world expert systems, Constructing simple reflex agents with sensors and actuators using Arduino
Unit 6	AI Applications and Trends: Applications of AI in various fields (healthcare, finance, automation), Emerging AI trends such as deep learning and generative models, Social impacts of AI and related ethical concerns, Challenges in deploying AI systems
Unit 7	Natural Language Processing: Introduction to NLP and its significance, Techniques for syntactic and semantic analysis, Basic text processing methods, Real-world applications of NLP, including chatbots and translators
Unit 8	Machine Learning Basics: Overview of machine learning and its types (supervised, unsupervised, reinforcement), Fundamental model training and testing concepts, Key algorithms like linear regression and decision trees, Evaluation metrics for assessing model performance

Unit 9	Deep Learning Fundamentals: Basics of neural networks and deep learning architectures, Overview of convolutional neural networks (CNNs) and recurrent neural networks (RNNs), Training mechanisms: backpropagation and gradient descent Hands-on creation of a basic neural network
Unit 10	Robotics and AI: Introduction to robotics and its integration with AI, Types of robots and their control mechanisms, Use of sensors and actuators in robotic systems, Case studies on applications like autonomous vehicles
Unit 11	Advanced Search Techniques: Overview of advanced search methods, Techniques like simulated annealing and genetic algorithms, Use of heuristic functions for complex problem-solving
Unit 12	Reinforcement Learning: Fundamentals of reinforcement learning (RL), Key concepts such as rewards, policies, and value functions, Algorithms like Q-learning and temporal difference learning, Applications of RL in real-world scenarios (e.g., game AI)
Unit 13	AI Ethics and Regulation: Ethical challenges in AI (bias, transparency, privacy), Overview of AI governance and regulation frameworks, Discussion on creating responsible AI practices
Unit 14	Capstone Project Development: Real-world project development integrating AI knowledge, Application of AI tools and techniques from previous units, Team collaboration and project documentation, Final project presentation and demonstration

TEXTBOOKS:

- Stuart Russell, Peter Norvig, “Artificial Intelligence – A Modern Approach”, 3rd Edition, Pearson Education / Prentice Hall of India, 2010.
- 2Elaine Rich, Kevin Knight, Shivashankar B Nair, “Artificial Intelligence”, 3rd Edition, TMH, 2010.
- Joseph C. Giarratano, Gary D. Riley, ”Expert Systems : Principles and Programming”,4th Edition, 2015.

REFERENCE BOOKS:

- Artificial Intelligence: Foundations of Computational Agents" by David L. Poole and Alan K. Mackworth

Course Name: QUANTUM INFORMATION

Credits: 4

Course Description:

This course provides a deep dive into the principles and applications of quantum computing and quantum information. It begins with the foundations of quantum states, measurements, and operations, followed by an exploration of quantum correlations, entanglement, and teleportation. Learners will learn about quantum gates, circuits, and foundational algorithms such as Shor's and Deutsch-Jozsa algorithms, progressing to advanced topics like quantum search and error correction. The course covers key concepts in quantum information theory, including Shannon and von Neumann entropy, and introduces quantum cryptography and noise analysis. It concludes with discussions on quantum protocols, current challenges, and emerging trends in quantum technologies, equipping learners with a comprehensive understanding of this rapidly evolving field.

Course Objectives:

This Course will enable learners to:

- Gain foundational knowledge of quantum states, density operators, generalized measurements, and quantum operations.
- Comprehend concepts like Bell inequalities, entanglement, Schmidt decomposition, and quantum teleportation.
- Acquire knowledge about the universal set of quantum gates, quantum circuits, and the Solovay-Kitaev theorem.
- Study and implement foundational algorithms such as Deutsch-Jozsa and Shor’s algorithm, as well as advanced algorithms like quantum search.
- Comprehend the basics of quantum information theory, entropy, and quantum cryptography including key distribution and error correction.

Course Outcomes:

At the end of the course the learner will be able to:

- Implement quantum measurement techniques and understand the no-cloning theorem in practical scenarios.

- Evaluate the implications of entanglement and perform quantum teleportation and super-dense coding.
- Construct and analyse quantum circuits using a universal set of gates and the Solovay-Kitaev theorem.
- Assess the efficiency and performance of foundational and advanced quantum algorithms like Shor's and quantum search algorithms.
- Develop and implement quantum cryptographic protocols and quantum error-correcting codes to ensure secure communication.

Unit 1	Quantum States and Measurements: Quantum states, Density operators, Generalized measurements, Quantum operations/channels, No-cloning theorem
Unit 2	Quantum Correlations and Entanglement: Bell inequalities, Entanglement, Schmidt decomposition, Super-dense coding, Quantum teleportation, PPT (Positive Partial Transpose) criterion
Unit 3	Quantum Gates and Circuits: Universal set of gates, Quantum circuits, Solovay-Kitaev theorem
Unit 4	Foundational Quantum Algorithms: Deutsch-Jozsa algorithm, Period-finding algorithm, Factoring and Shor's algorithm
Unit 5	Advanced Quantum Algorithms: Quantum search algorithm, Abelian quantum hidden subgroup problem
Unit 6	Quantum Information Theory Basics: Shannon entropy, Noiseless coding theorem
Unit 7	Von Neumann Entropy and Its Properties: Von Neumann entropy, Properties of von Neumann entropy, Schumacher compression

Unit 8	Noisy Coding Theorem: Noisy-coding theorem
Unit 9	Quantum Cryptography Essentials: Quantum key distribution, Entropic uncertainty relations
Unit 10	Quantum Noise and Error Analysis: Distance measures, Basics of quantum noise
Unit 11	Quantum Error Correction Fundamentals: Knill-Laflamme conditions, Quantum error-correcting codes
Unit 12	Quantum Error-Correcting Code Structures: Hamming bound, Specific quantum codes (e.g., Shor code, Steane code)
Unit 13	Quantum Protocols and Applications: Implementation of quantum protocols (e.g., super-dense coding and teleportation)
Unit 14	Modern Challenges and Future Directions: Current challenges in quantum computing, Potential future developments in quantum algorithms and cryptography

TEXTBOOKS:

- Nielsen, M. A., & Chuang, I. L. (2010). Quantum Computation and Quantum Information: 10th Anniversary Edition. Cambridge University Press.
- Mermin, N. D. (2007). Quantum Computer Science: An Introduction. Cambridge University Press.
- Kaye, P., Laflamme, R., & Mosca, M. (2007). An Introduction to Quantum Computing. Oxford University Press.

REFERENCE BOOKS:

- Preskill, J. (1998). Lecture Notes for Physics 229: Quantum Information and Computation. California Institute of Technology. Retrieved from http://theory.caltech.edu/~preskill/ph229/
- Wilde, M. M. (2017). Quantum Information Theory (2nd ed.). Cambridge University Press.

Course Name: RESEARCH METHODOLOGY & IPR

Credits: 4

Course Description:

This course equips learners with essential skills for conducting research and analysing data effectively. It begins with a review of fundamental statistical concepts and hypothesis testing, followed by an introduction to research fundamentals, including problem formulation, literature surveys, and statistical analysis. Learners will develop academic writing and presentation skills, with a focus on journal papers, reports, and PowerPoint techniques. The course explores data sources, collection methods, and both quantitative and qualitative data analysis. Topics include sampling strategies, survey design, experimental evaluation, and problem analysis. Emphasis is placed on research ethics, responsible practices, and effective project structuring to prepare learners for successful research endeavours.

Course Objectives:

This Course will enable learners to:

- Give an overview of the research methodology and explain the technique of defining a research problem
- Explain carrying out a literature search, its review, developing theoretical and conceptual frameworks and writing a review.
- Explain the details of sampling designs, measurement and scaling techniques and also, different methods of data collections.
- Explain several parametric tests of hypotheses and Chi-square tests.
- Explain the art of interpretation and the art of writing research reports.

Course Outcomes:

At the end of the course the learner will be able to:

- Identify the suitable research methods and articulate the research step in a proper sequence for the given problem.
- Define the problem statement by literature survey and suggest suitable solution for the given problem.
- Analyse the problem and conduct experimental design with the samplings

- Explain the data collection methods from various sources and segregate as the primary and secondary data
- Apply some concepts/section of Copy Right Act /Patent Act /Cyber Law Trademark to the given case study

Unit 1	Review of Statistics: Mean, Mode, Median, Arithmetic mean, Geometric mean, Harmonic mean, Probability, Problem solving, Gaussian distribution, Chi-square distribution, Learner-t distribution, Design of experiments, Hypothesis testing, Identification, Problems on hypothesis testing
Unit 2	Research Fundamentals: Definition of research, Importance of research, Research problem formulation, Literature survey, Problem analysis, Experimental evaluation, Survey techniques, Statistical analysis
Unit 3	Academic Writing: Short and long abstract writing, Journal paper format (international and national), Report writing, English writing skills, Communication skills
Unit 4	Presentation Skills: PowerPoint presentation skills, Other presentation tools and techniques
Unit 5	Research Project Structuring: Project structuring, Research ethics, Literature review techniques
Unit 6	Data Sources and Types: Primary data sources, Secondary data sources, Nature of data
Unit 7	Data Collection Methods: Methods for collecting primary data, Approaches for secondary data collection
Unit 8	Quantitative Data Analysis: Statistical methods, Data interpretation, Use of statistical software

Unit 9	Qualitative Data Analysis: Thematic analysis, Content analysis, Coding methods
Unit 10	Sampling Techniques: Sampling strategies, Selection criteria, Random and non-random sampling
Unit 11	Research Problem Analysis: Problem identification, Problem analysis, Evaluative techniques
Unit 12	Survey Methods: Survey design, Questionnaires, Data gathering methods
Unit 13	Experimental Evaluation: Designing experiments, Conducting experiments, Analysing experimental results
Unit 14	Ethical and Effective Research Practices: Research ethics, Responsible conduct of research, Effective research practices

TEXTBOOKS:

- Research Methodology: Methods and Techniques, C.R. Kothari, Gaurav Garg New Age International 4th Edition, 2018.
- Research Methodology, A Practical and Scientific Approach, Vinayak Bairagi, Mousami V. Munot
- Research Methodology a step-by- step guide for beginners. Ranjit Kumar SAGE Publications Ltd 3rd Edition, 2011
- Garg, B.L., Karadia, R., Agarwal, F. and Agarwal, U.K., 2002. An introduction to Research Methodology, RBSA Publishers.

REFERENCE BOOKS:

- Research Methods: the concise knowledge base Trochim, Atomic Dog Publishing, 2005.
- Donald H.McBurney, Research Methods, 5th Edition, Thomson Learning, ISBN:81-315-0047-0,2006.
- Donald R. Cooper, Pamela S. Schindler, Business Research Methods, 8/e, Tata McGraw-Hill Co.Ltd., 2006.
- Fuzzy Logic with Engg Applications, Timothy J.Ross, Wiley Publications, 2nd Edition, 2004
- Simulated Annealing: Theory and Applications (Mathematics and Its Applications, by P.J. vanLaarhoven & E.H. Aarts[e], 19.

Course Name: IOT

Credits: 4

Course Description:

This course provides a comprehensive overview of the Internet of Things (IoT), covering foundational concepts, applications, and ethical considerations. It begins with an introduction to IoT, exploring its characteristics, design, protocols, and deployment templates. Key IoT applications are examined, including home automation, smart cities, environmental monitoring, energy, retail, logistics, agriculture, industry, and healthcare. Learners will also learn about machine-to-machine (M2M) communication and its differences from IoT, as well as advanced IoT platform design methodologies. The course introduces physical IoT devices like Raspberry Pi, emphasizing device integration and application development. Ethical issues in IoT, such as privacy, environmental impact, and control, are addressed to foster a responsible and optimistic approach to IoT development.

Course Objectives:

This Course will enable learners to:

- Develop and deploy IoT systems using various design methodologies and communication protocols.
- Evaluate the effectiveness of IoT applications in sectors like smart cities, home automation, and environment monitoring.
- Apply knowledge of embedded systems and wireless sensor networks to create functional IoT solutions.
- Assess the capabilities and functionalities of IoT devices, including the use of Raspberry Pi.
- Analyse ethical considerations and privacy concerns related to the deployment of IoT technologies.

Course Outcomes:

At the end of the course the learner will be able to:

- Design comprehensive deployment plans for IoT systems across various application domains.

- Evaluate different IoT communication models and their suitability for specific use cases.
- Implement appropriate IoT protocols for efficient data transmission and device communication.
- Critically assess IoT applications for effectiveness in real-world scenarios such as smart cities and healthcare.
- Assess the security measures and ethical implications of IoT systems, focusing on privacy and environmental impact.

Unit 1	Introduction to IoT: Definition and characteristics of IoT, Physical design of IoT, Things in IoT, IoT protocols, Logical design of IoT, IoT functional blocks, IoT communication models, IoT communication APIs, Wireless sensor networks, Cloud Computing, Big Data Analytics, Communication protocols, Embedded systems, IoT Level-1 to Level-6 deployment templates
Unit 2	Home Automation IoT: Smart lighting, Smart appliances, Intrusion detection, Smoke and gas detectors
Unit 3	IoT for Smart Cities: Smart parking, Smart lighting, Smart roads, Structural health monitoring, Surveillance, Emergency response
Unit 4	IoT in Environment Monitoring: Weather monitoring, Air pollution monitoring, Noise pollution monitoring, Forest fire detection, River flood detection
Unit 5	IoT in Energy Sector: Smart grids, Renewable energy systems, Prognostics
Unit 6	IoT in Retail: Inventory management, Smart payments, Smart vending machines
Unit 7	IoT in Logistics: Route generation and scheduling, Fleet tracking, Shipment monitoring, Remote vehicle diagnostics

Unit 8	IoT in Agriculture: Smart irrigation, Greenhouse control
Unit 9	IoT in Industry: Machine diagnosis and prognosis, Indoor air quality monitoring
Unit 10	IoT in Health and Lifestyle: Health and fitness monitoring, Wearable electronics
Unit 11	IoT and M2M: Introduction to M2M, Differences between IoT and M2M, Software-defined networking (SDN), Network function virtualization (NFV)
Unit 12	IoT Platform Design Methodology: Purpose and requirements specification, Process specification, Domain model specification, Information model specification, Service specification, IoT level specification, Functional view specification, Operational view specification, Device and component integration, Application development
Unit 13	IoT Physical Devices and Endpoints: Definition of an IoT device, Basic building blocks of an IoT device, Raspberry Pi overview, Linux on Raspberry Pi, Raspberry Pi interfaces, Other IoT devices
Unit 14	Ethics in IoT: Characterizing IoT, Privacy concerns, Control and disrupting control, Crowdsourcing, Environmental impact, IoT as a part of the solution, Cautious optimism, Open IoT definition

TEXTBOOKS:

- Bahga, A., & Madisetti, V. (2015). Internet of Things: A Hands-On Approach. VPT.
- Höller, J., Tsiatsis, V., Mulligan, C., Karnouskos, S., Avesand, S., & Boyle, D. (2014). From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence. Academic Press.
- Minerva, R., Biru, A., & Rotondi, D. (2015). Towards a definition of the Internet of Things (IoT). IEEE Internet Initiative.

REFERENCE BOOKS:

- Vermesan, O., & Friess, P. (Eds.). (2014). Internet of Things - From Research and Innovation to Market Deployment. River Publishers.
- Gubbi, J., Buyya, R., Marusic, S., & Palaniswami, M. (2013). Internet of Things (IoT): A Vision, Architectural Elements, and Future Directions. Future Generation Computer Systems, 29(7), 1645-1660.

Course Name: DATA MINING

Credits: 4

Course Description:

This course provides an in-depth understanding of cluster analysis, a key technique in data mining used to group data based on similarities. It begins with the basics of clustering and explores various data types, preprocessing steps, and similarity measures essential for effective clustering. Learners will learn about prominent clustering techniques, including partitioning methods (K-Means, K-Medoids), hierarchical methods, density-based methods (DBSCAN), grid-based approaches, and model-based clustering (Gaussian Mixture Models). The course also addresses challenges in clustering high-dimensional data and introduces strategies to evaluate cluster validity. With a focus on real-world applications, learners will explore how clustering is applied in areas like market segmentation, image processing, and anomaly detection. This comprehensive course is ideal for those looking to enhance their skills in data analysis and pattern recognition.

Course Objectives:

This Course will enable learners to:

- Demonstrate a basic understanding of cluster analysis and its significance in data mining.
- Assess different types of data and preprocessing techniques necessary for effective clustering.
- Analyse various distance measures and their impact on clustering outcomes.
- Apply partitioning and hierarchical clustering methods to different datasets.
- Critically evaluate clustering techniques and their applications in real-world scenarios.

Course Outcomes:

At the end of the course the learner will be able to:

- Implement fundamental clustering techniques to analyse and interpret data.
- Assess the importance of data preparation and similarity measures in clustering.
- Critically evaluate and implement partitioning methods such as k-means and k-medoids.

- Apply advanced clustering techniques including density-based, grid-based, and model-based methods.
- Evaluate the validity of clustering results using various criteria and apply clustering methods to real-world problems.

Unit 1	Introduction to Cluster Analysis: Basic understanding and importance in data mining
Unit 2	Types of Data in Cluster Analysis: Overview of data types (numerical, categorical, mixed-type data) and their impact on clustering.
Unit 3	Data Preparation for Clustering: Steps involved in preprocessing data for clustering, handling missing values, and data normalization.
Unit 4	Similarity and Dissimilarity Measures: Introduction to distance measures (Euclidean, Manhattan, cosine similarity) and their roles in clustering.
Unit 5	Partitioning Methods Overview: partitioning methods and examples such as k-means and k-medoids.
Unit 6	K-Means Clustering: f k-means algorithm, steps for implementation, and convergence criteria.
Unit 7	K-Medoids Clustering: k-medoids clustering and its comparison with k-means.
Unit 8	Hierarchical Clustering Techniques: Overview of hierarchical clustering, agglomerative vs. divisive methods.

Unit 9	Density-Based Clustering: Introduction to density-based methods such as DBSCAN and its advantages.
Unit 10	Grid-Based Clustering Methods: grid-based methods like STING and CLIQUE.
Unit 11	Model-Based Clustering: Overview of model-based methods, including Gaussian Mixture Models.
Unit 12	Clustering High-Dimensional Data: Challenges and strategies for clustering in high-dimensional spaces.
Unit 13	Evaluation of Clustering: Criteria and techniques for assessing cluster validity (e.g., silhouette score, Davies-Bouldin index).
Unit 14	Applications of Clustering in Real-World Scenarios: Use cases of clustering in industries such as market segmentation, image processing, and anomaly detection.

TEXTBOOKS:

- Jain, A. K., & Dubes, R. C. (1988). Algorithms for Clustering Data. Prentice-Hall.
- Kaufman, L., & Rousseeuw, P. J. (2009). Finding Groups in Data: An Introduction to Cluster Analysis. Wiley-Interscience.
- Tan, P. N., Steinbach, M., & Kumar, V. (2018). Introduction to Data Mining (2nd ed.). Pearson

REFERENCE BOOKS:

- Xu, R., & Wunsch, D. (2008). Clustering. Wiley-IEEE Press.
- Everitt, B. S., Landau, S., Leese, M., & Stahl, D. (2011). Cluster Analysis (5th ed.). Wiley

Course Name: DEEP LEARNING

Credits: 4

Course Description:

The Machine Learning and Deep Learning course offers a thorough introduction to AI, covering fundamental concepts like regression, classification, and clustering. Learners will explore neural networks, deep learning architectures, and optimization techniques. The course delves into advanced topics such as CNNs, GANs, RNNs, and transformers, emphasizing their applications and training methods. It also addresses ethical challenges, explainable AI, and AI regulations. The course concludes with a capstone project where learners apply their knowledge to real-world problems, developing and presenting a complete AI solution.

Course Objectives:

This Course will enable learners to:

- Apply the fundamental concepts of machine learning, including regression, classification, and clustering.
- Implement and train basic neural networks, including the perceptron and multilayer feed-forward networks.
- Analyse and evaluate the architecture of deep learning models such as CNNs, RNNs, and GANs.
- Evaluate optimization techniques, including gradient descent variants and advanced optimizers, to improve model performance.
- Analyse ethical challenges in AI, focusing on bias, fairness, privacy, and the importance of explainable AI.

Course Outcomes:

At the end of the course the learner will be able to:

- Implement machine learning algorithms such as regression, classification, and clustering for practical tasks.
- Design, train, and evaluate neural networks using appropriate activation functions and loss functions.
- Analyse and implement deep learning models including CNNs and RNNs, assessing their suitability for specific applications.

- Evaluate optimization techniques and fine-tuning methods to enhance deep learning models.
- Assess the ethical concerns in AI development, including bias, transparency, and fairness, and explore the significance of explainable AI.

Unit 1	A Review of Machine Learning: The Learning Machines, Linear Algebra: Scalars, Vectors, Matrices, Tensors, Probability Basic idea of Regression, Classification, Clustering, LLM, CHATGPT
Unit 2	Neural Networks: The Biological Neuron, The Perceptron, Multilayer Feed Forward Networks Training Neural Networks, Backpropagation Learning, activation Functions: Linear, Sigmoid, Tanh Hard, Tanh Softmax, Rectified Linear Loss Functions: Notation, Regression, Classification, Reconstruction Hyperparameters, Learning Rate, Regularization, Momentum, Sparsity
Unit 3	Deep Networks: Deep Learning concepts, Common architectural principles of deep networks Parameters, Layers, Activation Functions, Loss Functions, Optimization Algorithms, Building Blocks of Deep Networks: RBMs, Autoencoders, Variational Autoencoders
Unit 4	Unsupervised Pretrained Networks: Deep Belief Networks, Generative Adversarial Networks (GANs), CNNs: Biological Inspiration, Intuition, Architecture Overview, Input, Convolutional, Pooling, Fully Connected Layers, Applications of CNNs
Unit 5	Recurrent Neural Networks: Modelling the Time Dimension, 3D Volumetric Input, General RNN Architecture, LSTM Networks, Domain-Specific Applications, Recursive Neural Networks and Applications
Unit 6	Convolutional Neural Networks: Biological inspiration and intuition, CNN architecture: input, convolutional, pooling, fully connected layers, Applications in image and multimedia analysis
Unit 7	Generative Models: Generative Models: Introduction and significance, Deep Belief Networks (DBNs), GANs: Architecture and training, Variational Autoencoders (VAEs)
Unit 8	Optimization in Deep Learning: Overview of optimization techniques, Gradient Descent Variants, Advanced optimizers: Adam, RMSprop, etc., Hyperparameter tuning

Unit 9	Transfer Learning and Fine-Tuning: Concepts of transfer learning, Pretrained models and their adaptation Fine-tuning techniques
Unit 10	Reinforcement Learning: Basics of reinforcement learning, Key concepts: rewards, policies, value functions, Q-learning and temporal difference learning, Applications of RL
Unit 11	Advanced Architectures: Advanced deep learning architectures, Residual Networks (ResNets) Transformer models and self-attention mechanisms
Unit 12	Explainable AI (XAI): Introduction to explainable AI, Importance and techniques of XAI Model interpretability and transparency
Unit 13	AI Ethics and Regulations: Ethical challenges in AI, Privacy, bias, and fairness in AI, Overview of AI regulations and governance
Unit 14	Capstone Project: Project development using AI concepts, Team collaboration and project execution, Final presentation and documentation

TEXTBOOKS:

- Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. Cambridge, MA: MIT Press. A foundational book on deep learning, covering neural networks, optimization, and advanced architectures.
- Bishop, C. M. (2006). Pattern Recognition and Machine Learning. New York, NY: Springer. A comprehensive guide to machine learning techniques, including regression, classification, and clustering.
- Sutton, R. S., & Barto, A. G. (2018). Reinforcement Learning: An Introduction (2nd ed.). Cambridge, MA: MIT Press. A detailed resource for reinforcement learning concepts, including Q-learning and policy optimization.

REFERENCE BOOKS:

- Zhang, Z., Lipton, Z. C., Li, M., & Smola, A. J. (2021). Dive into Deep Learning. Cambridge, UK: Cambridge University Press. An open-source, interactive book with practical implementations of deep learning models.

Course Name: QUANTUM ERROR CORRECTION

Credits: 4

Course Description:

The Quantum Computation and Error Correction course provides a foundational understanding of quantum computing and its associated error correction techniques. It begins with the basics of quantum computation, including qubits, quantum states, and quantum gates. The course then explores various quantum noise models and introduces quantum error-correction codes, such as Shor's Code and CSS Codes. Learners will delve into stabilizer and topological codes, learning about their structures and error-correction capabilities. Advanced topics include Quantum LDPC Codes, quantum bounds, and entanglement-assisted error-correcting codes. The course also covers fault-tolerant quantum computation and advanced error-correction techniques, focusing on real-world implementations and performance metrics.

Course Objectives:

This Course will enable learners to:

- Apply the basic concepts of quantum computation such as qubits, quantum gates, and measurement to solve problems.
- Analyse different quantum noise models, including bit flips, phase flips, and depolarizing channels, and their impact on quantum systems.
- Implement quantum error correction techniques, including Shor’s code and CSS codes, to correct quantum errors.
- Evaluate the concepts of fault-tolerant quantum computation, including error correction and the threshold theorem, for practical applications.
- Create and evaluate advanced quantum error correction codes, including topological codes, LDPC codes, and entanglement-assisted codes.

Course Outcomes:

At the end of the course the learner will be able to:

- Apply quantum error-correction methods, such as Shor’s code and CSS codes, to correct quantum computational errors.
- Analyse and calculate quantum error correction bounds (Quantum Hamming, Singleton, Gilbert-Varshamov) and their applications.

- Evaluate advanced quantum error-correction techniques like entanglement-assisted codes and quantum LDPC codes.
- Assess the efficiency and practicality of fault-tolerant quantum computation, including error thresholds and code performance.
- Design and construct quantum error-correcting codes using techniques like stabilizer codes and topological codes for real-world applications.

Unit 1	Basics of Quantum Computation: Qubits, Quantum States, Quantum Gates, Measurement, Density Matrices, Trace, Partial Trace, The No-Cloning Theorem
Unit 2	Quantum Noise Models: Bit Flips, Phase Flips, Depolarizing Channel, Amplitude Damping, Phase Damping
Unit 3	Quantum Error-Correction: Quantum Codes, The Knill-Laflamme Conditions, Pauli Error Basis, Discretization of Quantum Errors
Unit 4	Constructions: Shor's Code: Definition, Encoding and Decoding, Error Correction
Unit 5	Constructions: CSS Codes: Code Structure, Syndrome Measurement, Error Correction
Unit 6	Stabilizer Codes: Definition, Stabilizer Group, Code Space
Unit 7	Topological Codes: Surface Codes, Colour Codes, Code Properties, Error Correction
Unit 8	Quantum LDPC Codes: Quantum LDPC Codes: Definition, Properties, Encoding and Decoding

Unit 9	Bound - I: Quantum Hamming Bound: Definition, Applications, Bound Calculation
Unit 10	Bound - II: Quantum Singleton Bound: Definition, Applications, Bound Calculation
Unit 11	Bound - III: Quantum Gilbert-Varshamov Bound: Definition, Applications, Bound Calculation
Unit 12	Entanglement-Assisted Quantum Error-Correcting Codes: Entanglement-Assisted Quantum Error-Correcting Codes: Definition, Entanglement Assistance, Code Construction
Unit 13	Fault-Tolerant Quantum Computation: Fault-Tolerance, Threshold Theorem, Error-Correction in Computation
Unit 14	Advanced Quantum Error-Correction Techniques: Error Thresholds, Code Performance, Real-World Implementations

TEXTBOOKS:

- Nielsen, M. A., & Chuang, I. L. (2010). Quantum Computation and Quantum Information (10th ed.). Cambridge University Press.
- Gottesman, D. (2010). Stabilizer Codes and Quantum Error Correction. California Institute of Technology.
- Preskill, J. (2018). Lecture Notes for Physics 229: Quantum Information and Computation. Caltech.

REFERENCE BOOKS:

- Lidar, D. A., & Brun, T. A. (2013). Quantum Error Correction (1st ed.). Cambridge University Press
- Terhal, B. M. (2015). Quantum Error Correction for Beginners. Springer.

Course Name: MOBILE APPLICATION DEVELOPMENT TECHNIQUES

Credits: 4

Course Description:

The Mobile Operating Systems course offers an in-depth exploration of mobile app development, focusing on Android and Flutter platforms. It covers the history and evolution of mobile operating systems, detailing Android's architecture, development tools, and user interface components. Learners will learn multimedia handling, graphics, and data management in Android, along with setting up Flutter, using Dart programming, and building responsive UIs. The course includes state management, navigation, and cloud service integration for both platforms. It concludes with a final project, emphasizing best practices in app development, testing, and deployment.

Course Objectives:

This Course will enable learners to:

- Analyse the architecture and components of mobile operating systems, with a focus on Android OS, to understand their functionality and interaction in app development.
- Implement Android activities, intents, and UI components to develop interactive and efficient mobile applications.
- Design and integrate multimedia features, 2D/3D graphics, and audio/video handling in mobile apps using Android and Flutter frameworks.
- Develop efficient state management and navigation techniques in Flutter to build responsive and user-friendly mobile applications.
- Evaluate performance optimization and testing techniques to improve the efficiency, scalability, and reliability of mobile applications.

Course Outcomes:

At the end of the course the learner will be able to:

- Analyse the architecture of Android OS and develop mobile applications using Android and Flutter frameworks.
- Implement Android and Flutter UI components, manage multimedia features, and integrate backend services for creating functional mobile applications.

- Design state management and navigation in Flutter, enhancing the usability and responsiveness of mobile applications.
- Evaluate app performance and optimize their mobile applications for better efficiency and user experience.
- Develop a full-fledged mobile application as a final project, adhering to best practices in app design, development, and deployment.

Unit 1	Introduction to Mobile Operating Systems: Brief history of embedded device programming, Evolution of mobile apps and app development trends over the past years, Mobile Operating Systems (OS): Overview and significance, Architecture of mobile operating systems, Types of Mobile Operating Systems, Detailed focus on Android OS, Types and features of Android OS, Android OS Architecture: Layers and components, Android Application Development Platforms: IDEs, SDKs, and tools
Unit 2	Activity and Intent in Android: Introduction to Activity and Intent in Android, Activity life cycle and states, Types of Intents: Implicit and Explicit, Debugging Android apps with Android Studio, App testing techniques, Android support libraries and their significance, Understanding Views and Components in Android, Layouts, screen orientation, and user interaction (Buttons, Clickable images, Input controls), Menus and pickers in Android apps
Unit 3	Multimedia and 2D Graphics: Introduction to graphics in mobile apps, Simple graphics with Android: Drawing shapes, colours, and custom views, View and Canvas classes in Android, Working with 2D and 3D graphics, Introduction to OpenGL ES for 3D graphics, Audio handling in Android apps: Playing and streaming audio, Video handling in Android apps: Video playback, formats, and streaming
Unit 4	Android Services and Data Handling: Introduction to Android Services, Location services and working with geolocation, Using Fragment in Android for UI design, Introduction to SQLite database for data storage, SQLite In/Out operations: CRUD operations, Data binding in Android, Content Providers for sharing data across apps
Unit 5	Introduction to Flutter: Introduction to Flutter: Overview and benefits, Setting up a Flutter development environment, Introduction to Dart programming language, Basics of Flutter development, Creating your first Flutter app, Flutter widgets: Building UI components, Styles and animations in Flutter apps
Unit 6	Flutter UI Design and Layout: Advanced Flutter widgets: Text, Images, Icons, etc., Flutter layouts: Row, Column, Stack, and GridView, Building responsive UIs in Flutter for different screen sizes, Flutter themes and design principles, Customizing themes and global app styling, Understanding and implementing Navigation in Flutter

Unit 7	Flutter State Management: Introduction to state management in Flutter, Stateless vs Stateful widgets in Flutter, Managing state locally within a widget, Introduction to popular state management techniques: Provider, Riverpod, Bloc, Handling state across multiple screens and components
Unit 8	Flutter Navigation and Routing: Basics of navigation in Flutter, Navigating between screens using Routes, Passing data between screens in Flutter, Nested navigation and deep linking, Managing app flow using navigation stacks, Advanced navigation techniques in Flutter apps
Unit 9	Data Handling and Integration in Flutter: Working with JSON data in Flutter, Networking in Flutter: API integration, Using HTTP package for RESTful APIs, Persisting data locally with shared preferences and SQLite, Firebase integration with Flutter for real-time data
Unit 10	Flutter Animations: Introduction to animations in Flutter, Using Flutter's built-in animation classes (Tween, AnimationController, etc.), Animating widgets: Transition effects, moving elements, Custom animations in Flutter, Advanced animation techniques: Hero animations, PageRoute transitions, Performance considerations for animations
Unit 11	Advanced Android Features and UI Components: Introduction to RecyclerView and ListView for displaying large datasets, Custom adapters and ViewHolders in Android, Material Design components and UI guidelines, Custom views and widgets in Android, Advanced navigation patterns in Android (Navigation Drawer, Bottom Navigation)
Unit 12	Cloud Services and Backend Integration: Introduction to cloud services for mobile apps, Firebase Cloud Firestore and Firebase Storage, Integration of cloud databases with Android and Flutter apps, Authentication with Firebase: Sign-in methods, OAuth, Push notifications and background services using Firebase
Unit 13	Performance Optimization and Testing: Techniques for optimizing app performance (Memory, Battery, Network), Profiling and monitoring Android and Flutter apps, Android Performance Tools: Profiler, Traceview, Writing unit and integration tests for Android and Flutter, Best practices for app testing and debugging
Unit 14	Final Project and Best Practices: Final project development: Design, develop, and deploy a complete mobile app, Review of best practices in mobile app development, Code quality, refactoring, and version control, Deployment of Android apps to Google Play Store and Flutter apps to various platforms, Continuous integration and delivery in mobile app development

TEXTBOOKS:

- Phillips, B., Stewart, C., & Marsicano, K. (2021). Android programming: The big nerd ranch guide (8th ed.). Big Nerd Ranch.
- Biessek, A. (2020). Flutter for beginners: An introduction to app development with Flutter. Packt Publishing.

REFERENCE BOOKS:

- Moskala, M., & Wojda, I. (2020). Android development with Kotlin. Packt Publishing.
- Miola, A. (2020). Flutter complete reference: Create beautiful, fast, and native-quality apps for iOS and Android. Apress.

Course Name: INDUSTRIAL IOT AND AUTOMATION

Credits: 4

Course Description:

The Industrial Internet of Things (IIoT) course offers a comprehensive exploration of the integration of advanced technologies in manufacturing and industrial systems. Beginning with fundamental concepts of IIoT and cyber manufacturing systems, the course delves into the modelling frameworks for Cyber-Physical Systems (CPS) and Cyber Manufacturing Systems (CMS), highlighting model-based engineering principles and system architecture design. It emphasizes the importance of human-machine interaction, addressing worker integration and human factors in manufacturing. Learners will learn about advanced manufacturing systems, smart applications, and industrial automation, exploring innovations such as smart metering, e-health networks, and city automation systems. The course also covers critical aspects of control systems, IIoT communication networks, and AI applications in manufacturing, providing a solid foundation in data analytics, system integration, and implementation strategies for modern industrial environments.

Course Objectives:

This Course will enable learners to:

- Explain the fundamental concepts of Industrial Internet of Things (IIoT) and Cyber-Physical Systems (CPS) in industrial environments.
- Develop a comprehensive understanding of advanced manufacturing systems and their integration into Industry 4.0.
- Apply knowledge of human-machine interaction strategies in designing effective IIoT systems for industrial applications.
- Analyse and design smart applications in IIoT for sectors like healthcare, automotive, and smart cities.
- Evaluate the role of AI and data analytics in optimizing production processes and industrial automation.

Course Outcomes:

At the end of the course the learner will be able to:

- Demonstrate the application of IIoT and CPS modeling frameworks to real-world manufacturing systems.
- Implement system architectures and integration methods in the context of Industry 4.0 and IIoT.
- Design and manage smart applications, incorporating IIoT communication networks and AI-driven solutions.
- Integrate knowledge systems for enhanced manufacturing intelligence and predictive analytics in production.
- Develop and optimize industrial automation solutions using data-driven methods, ensuring system resilience and performance enhancement.

Unit 1	Introduction to Industrial Internet of Things (IIoT): Industrial IoT Fundamentals, Cyber Manufacturing Systems, Applications of Industrial CPS, Evolution of IIoT
Unit 2	CPS and CMS Modeling Framework: Cyber Physical Engineering Modeling, Manufacturing Process Modeling, Model-Based Engineering Principles, System Architecture Design
Unit 3	Human-Machine Interaction in IIoT: Worker-CPS Integration, User Intervention Strategies, Human Factors in Manufacturing, Workforce Management Systems
Unit 4	Advanced Manufacturing Systems: Innovation Ecosystems, Smart Manufacturing Principles, Digital Transformation, Industry 4.0 Integration
Unit 5	Smart Applications in IIoT: Smart Metering Systems, e-Health Networks, City Automation Systems, Automotive IoT Applications
Unit 6	Industrial Automation Applications: Home Automation Systems, Smart Card Technologies, Plant Automation Solutions, Manufacturing Case Studies

Unit 7	Control Systems in CPS: Supervisory Controllers, System Verification Methods, Component Testing, Performance Assessment Models
Unit 8	CMS and IIoT Architecture: Design Patterns, CPS Manufacturing Systems, Industry 4.0 Implementation, System Integration Methods
Unit 9	Knowledge Integration Systems: Database Management, Machine Vision Integration, Smart Automation Protocols, Knowledge Base Architecture
Unit 10	Production Facility Management: Resiliency Enhancement, CPS Implementation, Facility Monitoring, Production Optimization
Unit 11	IIoT Communication Networks: Network Protocols, Data Transfer Systems, Industrial Network Security, Communication Standards
Unit 12	AI in Manufacturing: Machine Learning Applications, CPS Intelligence, Digital Production Systems, Smart Manufacturing AI
Unit 13	Data Analytics in Production: Big Data Analytics, Manufacturing Intelligence, Condition Monitoring Systems, Predictive Analytics
Unit 14	System Integration and Implementation: Interoperability Standards, Integration Methodologies, Implementation Strategies, Performance Optimization

TEXTBOOKS:

- Xu, L. D., & He, W. (2019). Industrial Internet of Things: Cybermanufacturing systems (1st ed.). Springer.

- Lee, J., Bagheri, B., & Kao, H. A. (2015). A cyber-physical systems architecture for industry 4.0-based manufacturing systems. Manufacturing Letters, 3, 18-23. https://doi.org/10.1016/j.mfglet.2014.12.003

REFERENCE BOOKS:

- Dufresne, F., & Moreau, J. (2018). Smart manufacturing and cyber-physical systems: Applications and new frontiers. Wiley.
- Wang, L., & Xu, X. (2016). Modeling and control of manufacturing processes in Industry 4.0: Advanced technologies and applications. CRC Press.

Course Name: BUSINESS INTELLIGENCE TECHNOLOGY

Credits: 4

Course Description:

This course introduces the critical role of IT in driving business operations and strategic decision-making. It begins with an overview of the business perspective on IT applications, focusing on their significance in achieving enterprise excellence and operational efficiency. Learners will explore various types of digital data, including structured, semi-structured, and unstructured data, and learn the fundamentals of enterprise reporting, along with key tools and technologies used for reporting. The course also covers the strategic importance of Business Intelligence (BI) roadmaps, cost-benefit analysis in IT projects, and risk assessment techniques. Additionally, learners will examine the roles involved in BI projects and advanced data visualization methods to enhance decision-making. Finally, the course discusses the future of BI, focusing on emerging trends and technologies shaping the field.

Course Objectives:

This Course will enable learners to:

- Apply the concepts of business intelligence to enhance decision-making processes in enterprise settings.
- Analyse various types of digital data and determine their suitability for enterprise reporting and BI projects.
- Evaluate key tools and technologies used for enterprise reporting and their application in business environments.
- Design cost-benefit analyses for IT and BI projects to assess the feasibility and value of technological investments.
- Assess business analysis challenges and apply strategies to mitigate risks and improve the effectiveness of BI projects.

Course Outcomes:

At the end of the course the learner will be able to:

- Evaluate the strategic role of IT and BI in achieving business excellence and operational efficiency.

- Demonstrate the ability to apply business intelligence tools and techniques to enhance enterprise reporting.
- Assess business drivers for IT implementation and analyse their impact on BI project success.
- Develop business cases for IT projects by integrating cost-benefit analysis and risk assessment methodologies.
- Analyse and propose solutions for business intelligence challenges in organizations, including advanced data visualization and reporting techniques.

Unit 1	Introduction to Business View in IT: Overview of the business perspective on IT applications, significance in enterprise contexts.
Unit 2	Business Enterprise Excellence: IT contribution to achieving business excellence and operational efficiency.
Unit 3	Digital Data Types: different types of digital data, including structured, semi-structured, and unstructured data.
Unit 4	Basics of Enterprise Reporting: Core principles of enterprise reporting, components, and significance in decision-making.
Unit 5	Reporting Tools and Technologies: Introduction to key tools used for enterprise reporting and their capabilities.
Unit 6	BI Roadmap Overview: BI roadmap and its strategic importance in an organization.
Unit 7	Business Drivers for IT Implementation: business drivers that necessitate IT solutions and BI projects.

Unit 8	Business Analysis Issues: Challenges faced during business analysis, including stakeholder needs and data interpretation.
Unit 9	Cost-Benefit Analysis in IT Projects: Principles and processes of conducting cost-benefit analysis for IT and BI projects.
Unit 10	Risk Assessment and Mitigation: Techniques for assessing risks in IT projects and strategies to mitigate them.
Unit 11	Business Case Assessment Activities: activities involved in developing a business case for IT projects.
Unit 12	Roles Involved in BI Projects: responsibilities of key roles in BI projects, such as project managers and analysts.
Unit 13	Advanced Visualization Techniques: modern visualization tools and methods for enhancing data presentation.
Unit 14	Future of Business Intelligence: trends and technologies shaping the future of BI beyond current practices.

TEXTBOOKS:

- Turban, E., Sharda, R., & Delen, D. (2020). Business intelligence: A managerial perspective on analytics (10th ed.). Pearson.
- Lönnqvist, A., & Männistö, T. (2017). Business intelligence and analytics: Systems for decision support (2nd ed.). Springer.

REFERENCE BOOKS:

- Wixom, B. H., & Watson, H. J. (2020). The new analytics of business intelligence: Driving better decisions (1st ed.). Wiley.
- Ranjan, J. (2015). Business intelligence and analytics: Systems for decision support (3rd ed.). Springer.

Course Name: NATURAL LANGUAGE PROCESSING

Credits: 4

Course Description:

This course provides an in-depth exploration of Natural Language Processing (NLP), focusing on both its foundational concepts and advanced applications. It begins with an introduction to NLP, its core linguistic concepts, and the history of the field, before diving into essential topics like language models, N-grams, and part-of-speech tagging. Learners will also learn about the neural network architectures that power modern NLP systems, including Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM) networks, and advanced models like transformers and attention mechanisms. The course covers key NLP tasks such as syntactic parsing, semantic analysis, information extraction, and machine translation, emphasizing the integration of statistical methods and deep learning. Finally, it explores cutting-edge NLP applications like question answering, sentiment analysis, and dialogue systems, preparing learners for real-world NLP challenges.

Course Objectives:

This Course will enable learners to:

- Apply core linguistic concepts and language processing tasks to develop NLP models and systems.
- Analyse the effectiveness of language models, including N-grams, and evaluate their applications in real-world scenarios.
- Design and implement Part of Speech (POS) tagging systems using machine learning models, including Hidden Markov Models (HMMs).
- Evaluate advanced neural network architectures like Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM), and Transformers for sequential data processing in NLP.
- Develop and apply techniques for machine translation and information extraction to real-world language processing tasks, including named entity recognition and relation extraction.

Course Outcomes:

At the end of the course the learner will be able to:

- Apply various NLP techniques to solve real-world text processing tasks, such as language modelling and POS tagging.
- Evaluate the performance of different NLP models and identify appropriate applications based on model characteristics.
- Demonstrate proficiency in using neural network architectures, including RNNs, LSTMs, and transformers, for processing sequential and structured data in NLP.
- Analyse and apply semantic analysis methods, such as word sense disambiguation and semantic role labelling, to extract meaning from text.
- Design and implement machine translation systems, leveraging modern techniques like attention mechanisms and domain adaptation, for multilingual applications.

Unit 1	Fundamentals of Natural Language Processing: Introduction to NLP, Core Linguistic Concepts, Language Processing Tasks, History of NLP, Machine Learning in NLP
Unit 2	Language Models and N-grams: N-gram Models, Parameter Estimation, Smoothing Techniques, Language Model Evaluation, Model Applications
Unit 3	Part of Speech Tagging: Lexical Syntax, POS Categories, Tagging Algorithms, Hidden Markov Models, Sequence Labeling
Unit 4	Neural Network Foundations: Perceptron Architecture, Activation Functions, Backpropagation, Loss Functions, Network Training
Unit 5	Sequence Processing with RNNs: Recurrent Neural Networks, LSTM Architecture, Memory Cells, Gates Mechanisms, Sequential Learning
Unit 6	Advanced Neural Architectures: Attention Mechanisms, Transformers, Bidirectional Models, Transfer Learning, Fine-tuning Methods

Unit 7	Syntactic Parsing: Context-Free Grammars, Parsing Algorithms, Treebanks, Statistical Parsing, Dependency Parsing
Unit 8	Grammar Formalisms: Probabilistic CFGs, Lexicalized PCFGs, Dependency Grammars, Neural Parsing Models, Parse Evaluation
Unit 9	Semantic Analysis Foundations: Lexical Semantics, Word Sense Disambiguation, Compositional Semantics, Semantic Roles, Meaning Representation
Unit 10	Advanced Semantics: Semantic Role Labeling, Semantic Parsing, Frame Semantics, Event Semantics, Discourse Analysis
Unit 11	Information Extraction: Named Entity Recognition, Relation Extraction, Event Extraction, Temporal Information, IE Evaluation
Unit 12	Machine Translation Basics: Statistical MT, Word Alignment, Phrase-Based Translation, Neural MT Architecture, Translation Evaluation
Unit 13	Advanced Machine Translation: Synchronous Grammars, Attention in MT, Multi-lingual Models, Low-Resource MT, Domain Adaptation
Unit 14	Modern NLP Applications: Question Answering, Text Summarization, Dialogue Systems, Sentiment Analysis, Cross-lingual Applications

TEXTBOOKS:

- Jurafsky, D., & Martin, J. H. (2023). Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition (3rd ed.). Pearson.

- Choudhury, M., & Nasipuri, M. (2021). Natural Language Processing and Text Mining (2nd ed.). Wiley-IEEE Press.

REFERENCE BOOKS:

- Goldberg, Y. (2017). Neural network methods for natural language processing. Morgan & Claypool Publishers.
- Manning, C. D., & Schütze, H. (2020). Foundations of statistical natural language processing (Revised ed.). MIT Press.

Course Name: QUANTUM MACHINE LEARNING

Credits: 4

Course Description:

This course provides a comprehensive exploration of both classical machine learning and quantum computing, focusing on the integration of these fields. It covers the foundations of machine learning, including supervised and unsupervised learning, with practical tools like the Sci-Kit Learn framework, along with quantum computing basics such as quantum states, superposition, and quantum gates. The course delves into the application of linear algebra in machine learning, comparing classical and quantum approaches to problems like Fourier transforms and matrix operations. Learners will explore quantum programming platforms such as Qiskit and Pennylane, and learn how quantum algorithms like Quantum Support Vector Machines and Quantum Neural Networks are transforming machine learning tasks. Additionally, topics like quantum optimization algorithms, variational quantum algorithms, and hybrid classical-quantum systems provide an advanced understanding of the intersection between classical computing and quantum computing in solving complex problems.

Course Objectives:

This Course will enable learners to:

- Analyse the foundational principles of classical machine learning algorithms and their applications
- Evaluate quantum computing concepts such as superposition, entanglement, and quantum gates
- Implement quantum machine learning algorithms on quantum programming platforms like Qiskit and Pennylane
- Compare and contrast classical and quantum methods for supervised and unsupervised learning
- Develop hybrid classical-quantum systems for optimization tasks and demonstrate their potential benefits in machine learning

Course Outcomes:

At the end of the course the learner will be able to:

- Apply classical machine learning techniques and quantum algorithms to solve real-world problems
- Design, implement, and test quantum machine learning models on existing quantum programming platforms
- Analyse and interpret results from both classical and quantum learning systems
- Implementing hybrid classical-quantum solutions for optimization problems and demonstrating improvements over traditional methods
- Assess the applications of quantum computing in machine learning and its future potential in the field

Unit 1	Foundations of Classical Machine Learning: Machine Learning Overview, Basic Principles, Types of Learning, Applications, Sci-Kit Learn Framework
Unit 2	Quantum Computing Fundamentals: Quantum Mechanics Basics, Quantum States, Quantum Gates, Superposition and Entanglement, Quantum Measurements
Unit 3	Quantum Programming Platforms: Qiskit Architecture, Pennylane Framework, Circuit Design, Quantum Gate Operations, Quantum Simulators
Unit 4	Classical Linear Algebra in Machine Learning: Vector Spaces, Matrix Operations, Classical Fast Fourier Transform, Eigenvalue Problems, Linear Transformations
Unit 5	Quantum Linear Algebra: Quantum Fourier Transform, Quantum Phase Estimation, Quantum Matrix Operations, Quantum State Preparation
Unit 6	Classical Supervised Learning: Support Vector Machines, Classification Algorithms, Regression Methods, Model Evaluation, Feature Engineering
Unit 7	Quantum Supervised Learning: Quantum Support Vector Machines, Quantum Classification, Quantum Feature Spaces, Quantum Kernel Methods

Unit 8	Classical Unsupervised Learning: Principal Component Analysis, Clustering Algorithms, Dimensionality Reduction, Data Preprocessing
Unit 9	Quantum Unsupervised Learning: Quantum Principal Component Analysis, Quantum Clustering, Quantum Dimensionality Reduction
Unit 10	Classical Neural Networks: Network Architecture, Activation Functions, Backpropagation, Optimization Methods, Deep Learning
Unit 11	Quantum Neural Networks: Quantum Network Design, Quantum Neurons, Quantum Backpropagation, Parameterized Quantum Circuits
Unit 12	Quantum Approximate Optimization: QAOA Framework, MAX-Cut Problems, Combinatorial Optimization, Circuit Implementation
Unit 13	Variational Quantum Algorithms: VQE Algorithm, H2 Molecule Simulation, Parameter Optimization, Quantum Chemistry Applications
Unit 14	Hybrid Classical-Quantum Systems: Hybrid Architectures, Classical-Quantum Integration, Hybrid Optimization, Error Mitigation Strategies

TEXTBOOKS:

- Sutton, R. S., & Barto, A. G. (2018). Reinforcement learning: An introduction (2nd ed.). MIT Press.
- Cohen, B. A., & Pradhan, D. (2020). Quantum machine learning: An applied approach. Wiley-IEEE Press.

REFERENCE BOOKS:

- Arute, F., et al. (2019). Quantum supremacy using a programmable superconducting processor. Nature, 574(7779), 505–510. https://doi.org/10.1038/s41586-019-1666-5.
- Nielsen, M. A., & Chuang, I. L. (2011). Quantum computation and quantum information (10th anniversary ed.). Cambridge University Press.

Course Name: SOFTWARE PROJECT MANAGEMENT

Credits: 4

Course Description:

This course provides an in-depth understanding of the software development process, focusing on key methodologies, team dynamics, and project management techniques. It covers various software process models, including Waterfall, Prototyping, RAD, and Incremental, along with lifecycle management, milestone management, and project planning methods. Learners will explore requirements engineering, product specification techniques, and project feasibility analysis to ensure the successful development of software systems. Additionally, the course emphasizes software testing strategies, integration and system testing, and the importance of quality management using models like FURPS. By the end of the course, learners will be equipped with the skills to manage software projects effectively, from planning to testing and quality assurance.

Course Objectives:

This Course will enable learners to:

- Evaluate the effectiveness of different software development process models (e.g., Waterfall, Spiral, RAD) in real-world scenarios.
- Analyse the software life cycle phases and their transitions, with a focus on documentation and milestone management.
- Apply project planning methodologies to allocate resources and estimate durations for software development projects.
- Design and implement testing strategies, including unit testing, system testing, and validation.
- Assess software quality management techniques and their impact on ensuring the quality of the final product.

Course Outcomes:

At the end of the course the learner will be able to:

- Evaluate and compare various software development models and their effectiveness in managing complex projects.

- Analyse and optimize project planning and monitoring tools to track project progress efficiently.
- Apply requirements engineering techniques to gather and analyse customer needs for software projects.
- Design and execute comprehensive testing strategies to ensure product quality and reliability.
- Assess software quality and apply best practices for continuous improvement in software development.

Unit 1	Software Development Process Fundamentals: Process Definition, Process Tailoring, Process Improvement, Implementation Discipline, Software Model Identification
Unit 2	Software Process Models: Waterfall Model, Prototyping Model, RAD Model, Incremental Model, Spiral Model, Component Assembly Model
Unit 3	Software Life Cycle Management: Life Cycle Phases, Phase Transitions, Life Cycle Documentation, Life Cycle Activities, Milestone Management
Unit 4	Team Development and Leadership: Software Development Teams, Team Dynamics, Leadership Principles, Communication Strategies, Personality Traits
Unit 5	Project Organization: Organizational Structures, Team Roles, Responsibility Assignment, Project Coordination, Resource Management
Unit 6	Project Planning Methodologies: Top-Down Planning, Bottom-Up Planning, Activity Types, Duration Estimation, Resource Allocation
Unit 7	Project Monitoring Tools: Gantt Charts, PERT Charts, Critical Path Analysis, Schedule Monitoring, Progress Tracking

Unit 8	Project Review and Control: Tracking Meetings, Recovery Plans, Schedule Work, Escalation Procedures, Performance Monitoring
Unit 9	Requirements Engineering: Customer Problem Analysis, Initial Investigation, Information Requirements, Gathering Tools, Product Objectives
Unit 10	Product Specification Techniques: Data Flow Diagrams, Data Dictionary, Structured English, Decision Trees, Decision Tables
Unit 11	Project Feasibility: Technical Feasibility, Economic Feasibility, Operational Feasibility, Schedule Feasibility, Resource Feasibility
Unit 12	Software Testing Strategies: Test Planning, Development Testing, White Box Testing, Black Box Testing, Unit Testing
Unit 13	Integration and System Testing: System Integration, Validation Testing, System Testing, Test Documentation, Test Case Design
Unit 14	Software Quality Management: Quality Measures, FURPS Model, Quality Assurance, Software Reviews, Clean Room Methodology

TEXTBOOKS:

- Sommerville, I. (2011). Software engineering (9th ed.). Addison-Wesley.
- Pressman, R. S., & Maxim, B. R. (2014). Software engineering: A practitioner's approach (9th ed.). McGraw-Hill.

REFERENCE BOOKS:

- Boehm, B. W. (1988). A spiral model of software development and enhancement. ACM SIGSOFT Software Engineering Notes, 11(4), 14-24.
- Humphrey, W. S. (2000). Managing the software process. Addison-Wesley.

Semester: IV

Course Name: BUSINESS DATA ANALYTICS

Credits: 4

Course Description:

This course introduces the fundamentals of business analytics, focusing on its applications in areas such as marketing, sales, human resources, and healthcare. Learners will learn key concepts such as descriptive statistics, data distributions, and data visualization techniques, enabling them to analyse and present data effectively. The course also covers modelling uncertainty, statistical inference, and probability distributions, along with hands-on training in advanced analytics tools like Hadoop and MapReduce. Additionally, learners will explore frameworks like PigLatin, Hive, and JAQL, and learn about NoSQL databases such as Hbase and MongoDB. By the end, learners will gain practical skills in applying business analytics techniques and technologies to solve real-world business problems.

Course Objectives:

This Course will enable learners to:

- Explain the role of business analytics and identify its applications across diverse industries such as marketing, healthcare, and customer support.
- Analyse data using descriptive statistics, visualize insights with dashboards, and summarize distributions using metrics like mean, variance, and correlation.
- Apply probability models, statistical inference methods, and hypothesis testing techniques to model and address uncertainties in data.
- Implement data processing workflows using Hadoop, MapReduce, and related algorithms to solve large-scale data problems.
- Evaluate advanced data analytics frameworks and tools such as Hive, PigLatin, Apache Spark, and NoSQL databases for efficient big data management and analysis.

Course Outcomes:

At the end of the course the learner will be able to:

- Formulate linear programming problems (LPP) and solve them using the graphical method for optimal decision-making.
- Solve linear programming problems using the simplex method and analyse the results.

- Formulate and solve transportation and assignment problems using methods such as the Northwest Corner Rule, Vogel's Approximation, and Hungarian algorithm.
- Apply project management techniques like CPM and PERT to analyse and optimize project schedules.
- Develop solutions for scheduling and sequencing problems in single-server and multi-server environments.

Unit 1	Overview of Business Analytics: Introduction, Drivers for Business Analytics, Applications of Business Analytics: Marketing and Sales, Human Resource, Healthcare
Unit 2	Product Design: Product Design, Service Design, Customer Service and Support, Skills Required for a Business Analyst, Framework for Business Analytics Life Cycle for Business Analytics Process.
Unit 3	Essentials of Business Analytics – I: Descriptive Statistics, Using Data, Types of Data, Data Distribution Metrics: Frequency, Mean, Median, Mode, Range, Variance, Standard Deviation, Percentile, Quartile, z-Score, Covariance, Correlation
Unit 4	Essentials of Business Analytics – II: Data Visualization: Tables, Charts, Line Charts, Bar and Column Chart, Bubble Chart, Heat Map, Data Dashboards.
Unit 5	Modeling Uncertainty and Statistical Inference: Modeling Uncertainty: Events and Probabilities, Conditional Probability, Random Variables –Discrete Probability Distributions
Unit 6	Continuous Probability Distribution: Continuous Probability Distribution, Statistical Inference: Data Sampling, Selecting a Sample
Unit 7	Point Estimation: Point Estimation, Sampling Distributions, Interval Estimation, Hypothesis.
Unit 8	Analytics Using Hadoop: Introducing Hadoop, RDBMS versus Hadoop, Hadoop Overview, HDFS (Hadoop Distributed File System), Processing Data with Hadoop

Unit 9	MapReduce Framework – I: Introduction to MapReduce, Features of MapReduce, Algorithms Using Map-Reduce: Matrix-Vector Multiplication,
Unit 10	MapReduce Framework – II: Relational Algebra Operations, Grouping and Aggregation, Extensions to MapReduce.
Unit 11	Other Data Analytical Frameworks: Overview of Application development Languages for Hadoop, PigLatin
Unit 12	Hive: Hive, Hive Query Language (HQL), Introduction to Pentaho, JAQL
Unit 13	Apache: Introduction to Apache: Sqoop, Drill and Spark, Cloudera Impala
Unit 14	NoSQL Databases: Introduction to NoSQL Databases, Hbase and MongoDB.

TEXTBOOKS:

- Vignesh Prajapati, "Big Data Analytics with R and Hadoop", Packt Publishing, 2013.
- Umesh R Hodeghatta, Umesha Nayak, "Business Analytics Using R – A Practical Approach", Apress, 2017.
- Anand Rajaraman, Jeffrey David Ullman, "Mining of Massive Datasets", Cambridge University Press, 2012.
- Jeffrey D. Camm, James J. Cochran, Michael J. Fry, Jeffrey W. Ohlmann, David R. Anderson, "Essentials of Business Analytics", Cengage Learning, second Edition, 2016.

REFERENCE BOOKS:

- U. Dinesh Kumar, "Business Analytics: The Science of Data-Driven Decision Making", Wiley, 2017.

- A.Ohri, “R for Business Analytics”, Springer, 2012
- Rui Miguel Forte, “Mastering Predictive Analytics with R”, Packt Publication, 2015.

Course Name: INDUSTRIAL SAFETY

Credits: 4

Course Description:

This course provides a comprehensive overview of maintenance engineering, safety management, and preventive practices necessary for ensuring the smooth operation of industrial equipment and systems. The course begins by introducing accident prevention, mechanical and electrical hazards, and the importance of safety protocols in the workplace. It then covers key safety measures, including the Factory Act 1948, fire prevention techniques, and proper equipment maintenance procedures. The fundamentals of maintenance engineering are explored, focusing on types of maintenance, tools, and cost analysis. Learners will also learn about wear and corrosion in industrial machinery, along with prevention methods, lubrication techniques, and fault tracing using decision trees. The course emphasizes periodic and preventive maintenance for machinery such as pumps, air compressors, and electrical motors, and highlights the significance of routine inspections, repairs, and overhauls. By the end of the course, learners will have a solid understanding of how to manage maintenance tasks, prevent equipment failure, and maintain safety standards in various industrial environments.

Course Objectives:

This Course will enable learners to:

- Infer the causes, types, and control measures for accidents and hazards in industrial environments.
- Familiarize with preventive measures outlined in the Factories Act 1948 and safety standards such as fire prevention and safety colour codes.
- Introduce the fundamentals of maintenance engineering, including its functions, tools, and cost analysis.
- Explain the principles of wear, corrosion, and their prevention methods using lubrication and advanced technologies.
- Impart knowledge on fault tracing, periodic maintenance, and preventive maintenance strategies to ensure the reliability and longevity of mechanical and electrical systems.

Course Outcomes:

At the end of the course the learner will be able to:

- Identify causes of accidents, types of hazards, and implement effective control measures to enhance industrial safety.
- Demonstrate knowledge of the Factories Act 1948 and apply safety standards such as fire prevention and guarding mechanisms in industrial settings.
- Explain the principles of maintenance engineering and evaluate the relationship between maintenance costs and equipment replacement economy.
- Apply techniques to prevent wear and corrosion in mechanical systems using appropriate lubricants and corrosion prevention methods.
- Design and implement fault tracing methods and periodic preventive maintenance schedules for mechanical and electrical systems.

Unit 1	Introduction: Accident, causes, types, results and control, mechanical and electrical hazards, types, causes and preventive steps/procedure
Unit 2	Preventive steps – I: Describe salient points of factories act 1948 for health and safety, wash rooms, drinking water layouts, light, cleanliness,
Unit 3	Preventive steps – II: Fire, guarding, pressure vessels, etc, Safety color codes. Fire prevention and firefighting, equipment and methods.
Unit 4	Fundamentals of Maintenance Engineering – I: Definition and aim of maintenance engineering, Primary and secondary functions and responsibility of maintenance department
Unit 5	Fundamentals of Maintenance Engineering – II: Types of maintenance, Types and applications of tools used for maintenance, Maintenance cost & its relation with replacement economy, Service life of equipment.
Unit 6	Wear and Corrosion and Their Prevention Wear- types, causes, effects, wear reduction methods, lubricants-types and applications, Lubrication methods, general sketch

Unit 7	Working Applications: working and applications, i. Screw down grease cup, ii. Pressure grease gun, iii. Splash lubrication, iv. Gravity lubrication, v. Wick feed lubrication vi. Side feed lubrication, vii. Ring lubrication
Unit 8	Corrosion: Definition, principle and factors affecting the corrosion. Types of corrosion, corrosion prevention methods
Unit 9	Fault Tracing – I: Fault tracing-concept and importance, decision tree concept, need and applications, sequence of fault-finding activities
Unit 10	Fault Tracing – II: Show as decision tree, draw decision tree for problems in machine tools, hydraulic, pneumatic, automotive, thermal
Unit 11	Electrical equipment's: Electrical equipment's like, I. Any one machine tool, ii. Pump iii. Air compressor, iv. Internal combustion engine, v. Boiler, vi. Electrical motors, Types of faults in machine tools and their general causes.
Unit 12	Periodic and Preventive Maintenance – I: Periodic inspection-concept and need, degreasing, cleaning and repairing schemes, overhauling of mechanical components, overhauling of electrical motor, common troubles and remedies of electric motor
Unit 13	Periodic and Preventive Maintenance – II: Repair complexities and its use, definition, need, steps and advantages of preventive maintenance. Steps/procedure for periodic and preventive maintenance of: I. Machine tools, ii. Pumps, iii. Air compressors, iv. Diesel generating (DG) sets,
Unit 14	Periodic and Preventive Maintenance – III: Program and schedule of preventive maintenance of mechanical and electrical equipment, advantages of preventive maintenance. Repair cycle concept and importance

TEXTBOOKS:

- Banga, T. R., Sharma, S. C., & Agarwal, N. K. (2013). Industrial Engineering and Management. Khanna Publishers.
- Gupta, A. K. (2017). Maintenance and Reliability Engineering. Laxmi Publications.
- Mishra, R. C. (2006). Reliability and Maintenance Engineering. New Age International Publishers.

- Rao, P. N. (2009). Manufacturing Technology: Foundry, Forming, and Welding (4th ed.). McGraw-Hill Education.

REFERENCE BOOKS:

- Audels, Pump-hydraulic Compressors, Mcgrew Hill Publication, 1978.
- Garg H P,Maintenance Engineering, S. Chand and Company,1987.
- Hans F. Winterkorn, Foundation Engineering Handbook, Chapman & Hall London, 2013.
- Higgins & Morrow, Maintenance Engineering Handbook, Eighth Edition,2008

Course Name: OPERATIONS RESEARCH

Credits: 4

Course Description:

This course offers an in-depth exploration of operations research techniques and business intelligence (BI) applications, equipping learners with the skills needed for effective decision-making and project management in complex business environments. The first half of the course focuses on linear programming and network analysis, covering topics such as formulation of linear programming problems, the Simplex method, transportation and assignment problems, and scheduling and sequencing models. The second half delves into BI project management, including planning, risk management, and the use of analytical models like efficiency analysis, cluster analysis, and outlier detection. Learners will also examine BI applications in various business domains, such as marketing, logistics, and production, along with emerging trends in enterprise reporting and visualization. By the end of the course, learners will have a comprehensive understanding of how to leverage operations research techniques and BI tools to enhance business decision-making.

Course Objectives:

This Course will enable learners to:

- Infer the foundational concepts of linear programming and operations research for effective decision-making.
- Equip with advanced techniques in solving linear programming problems and performing sensitivity analysis.
- Familiarize with network analysis methods, including transportation, assignment, shortest path problems, and project scheduling techniques like CPM and PERT.
- Provide insights into managing business intelligence (BI) projects, including planning, risk management, and efficiency analysis.
- Explore advanced analytical models, BI applications, and enterprise reporting trends for practical implementation in real-world scenarios.

Course Outcomes:

At the end of the course the learner will be able to:

- Formulate and solve linear programming problems using graphical methods, simplex algorithms, and advanced approaches like primal-dual and dual simplex methods.
- Analyse and solve network problems, including transportation, assignment, and shortest path problems, using appropriate algorithms.
- Apply scheduling, sequencing, and inventory control models to optimize resource utilization and decision-making processes.
- Plan and manage business intelligence projects effectively by addressing requirements, risks, and interviewing processes.
- Implement advanced analytical models and BI applications in marketing, logistics, and production, and assess future trends in enterprise reporting and visualization.

Unit 1	Linear Programming: Introduction to Operations Research, Assumptions of Linear Programming Problems, Formulations of Linear Programming Problem, Graphical Method.
Unit 2	Advances in Linear Programming: Solutions to LPP Using Simplex Algorithm, Revised Simplex Method, Primal-Dual Relationships, Dual Simplex Algorithm, Sensitivity Analysis.
Unit 3	Network Analysis – I: Transportation Problems, Northwest Corner Rule, Least Cost Method, Vogel's Approximation Method, Assignment Problem, Hungarian Algorithm.
Unit 4	Network Analysis – II: Shortest Path Problem, Dijkstra's Algorithm, Floyd's Algorithm, Systematic Method, CPM/PERT.
Unit 5	Network Analysis – III: Scheduling and Sequencing, Single Server and Multiple Server Models, Deterministic Inventory Models, Probabilistic Inventory Control Models.
Unit 6	Business Intelligence Project Management: Managing the BI Project, Defining and Planning the BI Project.

Unit 7	Project Planning in BI: Project Planning Activities, General Business Requirements, Project-Specific Requirements.
Unit 8	Interview Process and Risk Management- I: Interviewing Process, Roles and Risks Involved in Project Planning Activities.
Unit 9	Interview Process and Risk Management- II: Interviewing Process, Roles and Risks Involved in Project Planning Activities.
Unit 10	Efficiency Measures: Identification of Good Operating Practices, Cross Efficiency Analysis, Virtual Inputs and Outputs.
Unit 11	Advanced Analytical Models: Other Efficiency Models, Pattern Matching, Cluster Analysis, Outlier Analysis.
Unit 12	BI Applications: Marketing Models, Logistic Models, Production Models, Case Studies.
Unit 13	Business View of IT Applications: Business Enterprise Excellence, Key Purpose of Using IT, Type of Digital Data
Unit 14	Enterprise Reporting and Future Trends: Basics of Enterprise Reporting, Advanced Visualization, Rich Report, BI Road Ahead, Future Beyond Technology.

TEXTBOOKS:

- Hillier, F. S., & Lieberman, G. J. (2021). Introduction to Operations Research (11th ed.). McGraw-Hill Education.
- Taha, H. A. (2017). Operations Research: An Introduction (10th ed.). Pearson Education.

- Ravindran, A., Phillips, D. T., & Solberg, J. J. (2007). Operations Research: Principles and Practice (2nd ed.). Wiley.
- Sharma, J. K. (2013). Operations Research: Theory and Applications (5th ed.). Macmillan Publishers India.

REFERENCE BOOKS:

- Harvey M Wagner, Principles of Operations Research: Prentice Hall of India 2010
- Hitler Libermann, Operations Research: McGraw Hill Pub. 2009
- Pant J C, Introduction to Optimisation: Operations Research, Jain Brothers, Delhi, 2008
- Pannerselvam, Operations Research: Prentice Hall of India 2010
- Taha H A, Operations Research, An Introduction, PHI, 2008

5.3 Duration of the programme

Programme	Level	Duration	Maximum duration for completion	Credits
MCA	Master's Degree	2 years	(2+ 2) years (As per UGC Notification on Specification of Degree, 2014)	84 Credits

5.4 Faculty and support staff requirement

Academic Staff	Number available to meet the required delivery norms
Programme Coordinator	1 member
Course Coordinator	1 member
Course Mentor	1 member per batch of 250 students

5.5 Instructional delivery mechanisms

The instructional delivery mechanisms for the online MCA programme from CDOE, DSU has been designed to ensure an engaging and effective learning experience for students. The CDOE has a team of qualified and experienced faculty and staff for the programme. The proficiency of the faculty team ensures that programs are thoughtfully designed and executed to meet high academic standards. This commitment to quality provides students with a seamless, engaging, and enriching learning experience, specifically tailored to the unique needs and challenges of online education. CDOE creates an environment that supports academic excellence and professional growth, empowering students to succeed in a flexible and accessible educational setting.

CDOE will be having an academic calendar outlining important dates for major events in the semester. This academic calendar will be shared with students at the start of each semester, ensuring clear communication and effective planning for all academic activities.

In addition to providing content through Self-Learning Materials (SLMs), students will have access to a wide range of e-learning resources, including audio and video content, to enhance their understanding of the course material. To ensure student engagement in the programme, various activities will be organized in the form of the following:

- **Webinars and Online Lectures:** Live sessions will be conducted to offer students the opportunity to learn directly from faculty members and engage in real-time discussions. These sessions will cover key topics and provide clarity on different concepts.

- **Discussion Forums:** Students will be encouraged to actively participate in discussion forums designed to stimulate critical thinking and foster open communication. These forums will provide a platform for students to share their perspectives and express their ideas without hesitation. Such forums encourage participation from all students and provide an opportunity for discussion and gaining insights while maintaining a professional decorum.

  Thus, students remain actively involved in the learning process, fostering a collaborative and enriching educational experience that aligns with the objectives of online learning.

- **Assessments:** Continuous Internal Assessments are conducted to support ongoing learning and development. Self-Assessment Questions included in the e-SLMs and quizzes available on the LMS provide regular opportunities for students to conduct periodic evaluations. These quizzes can be taken multiple times, allowing students to refine their understanding and work toward achieving correct answers. This iterative process promotes a deeper understanding of key concepts and strengthens learning outcomes. The flexibility of this approach encourages active participation, helping students identify and address knowledge gaps while building confidence in applying their knowledge effectively. By regularly monitoring progress, students can engage more thoroughly with the course material, ensuring continuous improvement and mastery of the subject matter. The case study pedagogy in the online MCA programme aims to enhance critical thinking, decision-making, and problem-solving skills among students. It encourages real-world application of theoretical concepts, fostering deeper understanding. Students can use the opportunity to analyse complex business scenarios, develop strategic insights, and improve their ability to navigate challenges. Case studies help in cultivating practical knowledge and prepares students for leadership roles.

  Students will be provided access to national portals such as SWAYAM and NPTEL, along with the University's digital library, which will be integrated into the LMS for supplementary reading material. This allows students to explore additional resources beyond the prescribed syllabus. Such access will encourage students to complement the core curriculum but also supports lifelong learning, empowering students to stay updated with the latest developments in their field of study.

5.6 Media resources - Print, Audio or Video, Online, Computer aided:

Students will be getting access to a wide range of e-learning materials, including audio and video content, faculty-led video sessions, virtual classrooms and discussion boards through the

LMS. This will enable students to track their progress in real-time through a personalized dashboard, allowing them to monitor their learning journey.

Students will also be informed about upcoming academic events. Regular notifications will be sent to remind students about upcoming webinars, virtual classes, assignments, and discussion forums. Such notifications will help students to manage their schedules and academic responsibilities.

Additionally, the LMS will facilitate direct communication between students and Course Coordinators/Mentors. Students will be able to raise queries, seek clarification, and receive responses from faculty members. This will foster a supportive learning environment and ensures that students have the necessary guidance and resources to succeed in their studies. The LMS will be a platform to maintain an interactive and engaging online learning experience, enabling students to actively participate in their education while receiving the support they need.

5.7 Student Support Services

Student Support Services of the CDOE, DSU will be providing pre-admission student support services like counselling about the programme including curriculum design, mode of delivery, fee structure and evaluation methods. Post-admission student support services include guiding students towards accessing LMS portal, Academic Calendar and academic delivery. The support services team shall provide support/training in attending the online proctored semester end examination. The support team shall answer to the queries pertaining to conduct of end-semester examinations, evaluation and issue of certificates.

6. Procedure for Admission, Curriculum Transaction and Evaluation

The purpose of Online education by CDOE, DSU is to provide flexible learning opportunities to students to attain qualification, wherever students are not able to attend the regular classroom teaching. Academic programmes offered for such candidates under Online Learning mode will be conducted by CDOE, DSU The programmes/courses are termed Online mode for award of Degree.

Eligibility criteria, programme/course structure, curriculum, evaluation criteria and duration of programme shall be approved by Board of Studies and Academic Council which are based on UGC guidelines.

Candidates seeking admissions in any programme offered by CDOE, DSU shall fill up online application form available informed on the website. Before applying, candidates must check eligibility criteria for the programme. Details about eligibility criteria, programme structure, curriculum, duration, and fee structure are available on the University website.

6.1. Procedure for Admission

6.1.1 Minimum Eligibility Criteria for admission

- Passed any graduation degree (e.g.: B.E. / B.Tech./ B.Sc / B.Com. / B.A./ B. Voc./ BCA etc.,) preferably with Mathematics at 10+2 level or at Graduation level.
- Obtained at least 50% marks (45% marks in case of candidates belonging to reserved category) in the qualifying examination (for students having no Mathematics background need to undergo compulsory bridge course).

Important Instructions:

- Admission granted by the University to the Programme shall be confirmed only for the candidates who fulfil the Admission Eligibility requirement by submitting all the requisite documents and has paid the semester fees.
- All other Admissions granted by the University to the Programme shall be Provisional until the candidate meet the eligibility criteria
- Provisional Admission shall stand cancelled if the candidate does not fulfil Programme eligible criteria within the stipulated time given by the CDOE, DSU.
- The University has the right to make necessary changes from time to time as deemed fit in Eligibility criteria, programme/course structure, curriculum, duration, fee structure and programme announcement dates. All changes will be notified on the website.
- Prior to applying for admissions, candidates are advised to go through the details provided on the University website & the Programme prospectus.

6.1.2. Fee Structure and Financial assistance policy

Suggested Fee for MCA programme is INR 1,20,000/- (One Lakh Twenty Thousand only). Overseas students need to remit the programme fees equivalent in USD to the University.

A scholarship of up to 10% on tuition fees will be provided to Merit students and to students who belong different special categories as defined in the University Policy.

6.2. Curriculum Transactions

6.2.1. Programme Delivery

DSU utilizes modern technology to deliver online programs, ensuring students receive a high standard of education. The faculty at DSU is dedicated to providing expert guidance that promotes the overall development of students. They do more than facilitate learning—they serve as mentors, fostering an engaging environment that enhances student retention and academic growth. The programme is designed with the goal of equipping students with specialized expertise, helping them excel in their chosen fields. Some of the important features are:

- Online academic delivery, ensuring flexibility and accessibility for all students.
- Regular updates and reviews of the curriculum and study materials to keep content current and relevant.
- Live, interactive lectures conducted by CDOE, DSU faculty members and course coordinators, ensuring engagement with students to support them in their learning journey.
- Continuous academic and technical support to assist students throughout in their online learning journey.
- Guidance and mentoring from Course coordinators to help students to navigate any academic challenges.
- Dedicated learning and delivery support from Course mentors.

This approach guarantees a comprehensive and supportive learning experience, where students can focus on their academic outcomes for better professional outcomes. Through these well-structured delivery methods, DSU ensures that each student receives the tools and guidance they need to succeed in their studies and future careers.

6.2.2. Norms for Delivery of Courses in Online Mode

Sl. No.	Credit value of the course	No. of Weeks	No. of Interactive Sessions		Hours of Study Material		Self- Study hours including Assessment etc.	Total Hours of Study (based on 30 hours per credit)
			Synchronouss Online Counselling/ Webinars/ Interactive Live Lectures (1 hour per week)	Discussion Forum/ asynchronous Mentoring(2 hours per week)	e-Tutorial in hours	e-Content hours		
1.	2 Credits	6 weeks	6 hours	12 hours	10	10	22	60
2.	3 Credits	9 weeks	9 hours	18 hours	15	15	33	90
3.	4 Credits	12 weeks	12 hours	24 hours	20	20	44	120

6.2.3. Learning Management System to support Online mode of Course delivery

The LMS platform for the online MCA programme has been specifically designed to help students maximize their potential in their chosen field. It offers a secure and reliable learning environment, accessible on both web and mobile devices, ensuring a consistent and seamless experience. With a user-friendly interface, the platform makes it easy for instructors to design courses, create content, and grade assignments efficiently. Its responsive design delivers an excellent mobile experience, allowing students to access course materials anytime, anywhere.

The LMS platform prioritizes accessibility, ensuring all tools are standards-compliant and easy to navigate, including support for assistive technologies. This ensures an inclusive learning environment for all students at all times such that students have the flexibility to study at their own pace and on their own schedule. The availability of LMS for the online MCA programme encourages students to develop a self-directed approach to learning in the programme.

6.2.4. Course Design

The course content has been carefully designed in accordance with the SWAYAM guidelines, employing the 4-quadrant approach to ensure a seamless and engaging learning experience. This structured approach includes four key components, each crafted to support various aspects of student learning and engagement:

(a) Quadrant-I i.e. e-Tutorial, that contains – Faculty led Video and Audio Contents. These provide visual and auditory explanations of key concepts, offering clear and comprehensive coverage of course topics. The use of video content enhances understanding and helps students grasp complex subjects more easily. Simulations, video demonstrations, Virtual Labs

etc.

(b) Quadrant-II i.e. e-Content to contain illustrations, video demonstrations, documents as required. Curated reading resources, such as articles, case studies, and textbooks, allow students to explore topics in greater detail. These materials complement video lectures and encourage deeper exploration of the subject matter.

(c) Quadrant-III i.e. Discussion forums to raise and clarify doubts on real time basis bythe Course Coordinator(s) and their team. Interactive online discussion platforms enable students to engage with their peers and instructors. These forums foster collaboration, allowing students to share insights, ask questions, and discuss ideas, creating a rich, supportive learning environment.

(d) Quadrant-IV i.e. Self-Assessment, that contains MCQs, Problems, Quizzes,Assignments with solutions and Discussion forum topics. Quizzes, assignments, and tests are integrated throughout the course to help students gauge their understanding and track their progress. These self-assessment tools encourage active learning and allow students to identify areas for improvement.

By utilizing the 4-quadrant approach, the course content ensures a balanced and holistic learning experience that promotes both theoretical knowledge and practical application.

6.2.5. Academic Calendar

The Academic Calendar indicates the timelines for the different academic activity for the Programme in the semester:

Sl. No.	Week	Event(s)
1	Week 1	Induction to the Programme.
2	Week 2-14	Commencement of live lecture sessions.
3	Week 3	Opening of Continuous Internal Assessments (CIA).
4	Week 4	Internal Assignment(s) submission.
5	Week 2-11	Discussion Forums.
6	Week 12	Closure of Internal Assignment(s) submissions.
7	Week 13	Semester End Examination (SEE) - Time Table.
8	Week 13-14	Semester End Examination (SEE) - Registration.
9	Week 15	Exam Admit Card download.
10	Week 16 onwards	Semester End Examination (SEE).
11	Week 17 onwards	Registration for next higher semester.

6.3 Evaluation

Every student shall be assessed for a course through Continuous Internal Assessment (CIA) and Semester End Examination (SEE) as prescribed. CIA and SEE shall respectively have 30:70 percent weightage.

Continuous Internal Assessment (CIA) for Theory Courses shall be conducted for 30 marks in the form of assignments. Semester End Examination Assessment (SEE) for Theory Courses shall be conducted for 70 marks. The SEE question paper shall comprise of objective and descriptive type questions. The SEE will be conducted with technology support as a remote proctored examination.

Continuous Internal Assessment (CIA) for Practical Courses shall be conducted for 30 marks. Semester End Examination (SEE) for Practical shall be conducted for 70 marks. Additional details shall be made available in the respective Practical Guidelines Manual. Students are required to submit their assignments for Continuous Internal Assessment (CIA) and participate in a viva voce for the Semester-End Examination (SEE).

A student's performance in a course shall be judged by taking into account the results of CIA and SEE together. A student has to obtain and satisfy the following conditions to be declared as pass in each course:

(i) minimum 40% of marks in CIA
(ii) minimum 40% of marks in SEE
(iii) minimum 40% of marks in aggregate considering both CIA & SEE

- Students must score minimum 40% marks for project-based courses.
- There shall be no improvement of Continuous Internal Assessment marks if they are above 40%.
- If a student fails in any one component (failure to get 40% marks either in CIA or SEE), then the student will be required to re-appear for that component only (CIA or SEE as the case may be).
- There shall be no improvement of Semester End Examination marks if they are above 40%.

DSU shall be complying as per the prevailing regulatory directions on the conduct of the examinations.

6.3.1 Question Paper Pattern

The Question Paper for the Semester End Examination Assessment (SEE) for Theory Courses

shall be conducted for 70 marks. shall comprise of three sections:

- Section – A for 20 marks comprising TEN Multiple Choice Questions (MCQ) of 2 mark each.
- Section – B for 30 marks of Short Answer type Descriptive Questions of 6 marks each for which a student shall be need to write answers for 5 out of 6 questions.
- Section – C for 20 marks of Long Answer type Descriptive Questions of 10 marks each for which a student shall be need to write answers for 2 out of 3 questions.

Section A (Answer ALL)	**Questions x Marks**	**Marks**
Ten Multiple Choice Questions	10 x 2	20
Section B – Answer **Five** out of **Six** questions		
Descriptive Questions (Short Answers)	5 x 6	30
Section C – Answer **Two** out of **Three** questions		
Descriptive Questions (Long Answers)	2 x 10	20
Total		**70**

6.3.2 Distribution of Marks in Continuous Internal Assessments

The following procedure shall be followed for awarding internal marks for courses. Student must submit two assignments each carrying 30 marks and average of both will be considered as internal assessment marks.

6.3.3 Passing Minimum

The students are considered as passed in a course if they score 40% marks in the Continuous Internal Evaluation (CIA) and Semester-End Examinations (SEE) individually. If a student fails in any one component (failure to get 40% marks either in CIA or SEE), then the student will be required to re- appear for that component only.

6.3.4 Marks and Grades

Marks shall be awarded for both CIA and SEE. The grading will normally be based on CIA and SEE. Relationships among Grades, Grade points and % of marks are listed as per the below mentioned criteria, where F is Fail and IC is ABSENT:

GRADE	GRADE POINTS	DESCRIPTION	% MARKS
O	10	Outstanding	90 to 100
A+	9	Excellent	80 to 89
A	8	Very Good	70 to 79
B+	7	Good	60 to 69
B	6	Above Average	55 to 59
C	5	Average	50 to 54
P	4	Pass	40 to 49
F	0	Fail	< 40
IC	-	In Complete	-

Table: Grade, Points, Grade Description and % of marks

Class Equivalence of Grade points

CGPA	Class/Division
>=5.0 - < 5.75	Pass Class
>=5.75 -<6.75	Second Class
>=6.75 - <7.75	First Class
>=7.75 – 10	First Class with Distinction

- A student will have to ensure a minimum CGPA of 4, to become eligible for the award of the degree.
- A student shall have to re-appear all courses in which they obtain 'F' and 'IC' Grade until a passing grade is obtained.
- 'F' grade denotes failure to obtain minimum passing marks in Continuous Internal Assessment or Semester End Examinations.
- 'IC' grade denotes incomplete performance in any Theory and/or Practical Assessment. It may be awarded in case of absence for CIA or SEE.

- The student can appear for the course/s with ‘F’ and ‘IC’ grade, when exams are conducted subsequently by the University for those Courses.

For a semester:

The SGPA is calculated on the basis of grades obtained in all courses, except audit courses and courses in which F grade or below, registered for in the particular semester.

$$\text{SGPA} = \frac{\text{Points secured in the semester (O − P Grades)}}{\text{Credits registered in the semester, excluding audit courses}}$$

For the entire programme:

The CGPA is calculated on the basis of all pass grades, except audit courses.

$$\text{CGPA} = \frac{\text{Cumulative points secured in all the passed courses (O − P Grades)}}{\text{Cumulative registered credits, excluding audit Courses}}$$

7.1 Requirement of the Laboratory Support and Library Resources

7.1 Laboratory Support

No lab-based courses are offered in this program.

7.2 Library Resources

CDOE, DSU provides an exceptional library facility to support the academic needs of students enrolled in the MCA program. The Central Library at DSU is well-stocked with a vast array of reference books, including key titles relevant to the online MCA curriculum.

For students enrolled in the online mode of education, the University offers digital library access, which provides an extensive collection of e-books, journals, and academic databases. This digital resource ensures that online students have equal access to critical academic content, supporting them in their coursework and research. To further enhance the academic experience, DSU will provide access to educational platforms like SWAYAM, Scopus, and Knimbus, to

the students. These memberships provide access to an extensive range of academic content, including journals, articles, and research papers, enhancing the research capabilities among students.

To ensure a holistic learning experience, DSU has integrated e-learning resources into the LMS for the online MCA program. The LMS serves as a centralized hub for all course materials, including e-books, articles, and other resources. Additionally, e-tutorial lectures are made available, offering students the flexibility to study at their own pace while reinforcing key concepts covered in their courses. This seamless integration of resources ensures that students have everything they need to succeed in their academic journey, regardless of their mode of study.

Beyond traditional academic resources, DSU will organize webinars and other virtual interactions by industry professionals for students. Such events will provide students with valuable insights into contemporary trends, challenges, and best practices in the business world. Through a combination of comprehensive library resources, digital access, and expert industry engagement, the CDOE at DSU will provide online MCA students to strive for academic excellence.

8. Cost Estimate of the Programme and the Provisions

The costs towards the programme study material development and academic delivery system depend on the total programme credits and the number of students. DSU, known for academic excellence, has always complied with the UGC regulations. The programme related expenses towards e-content and IT infrastructure setup cost 50% of the programme fee revenue and the balance for the academic delivery of the programme which depends on the student strength.

9. Quality assurance mechanism and expected programme outcomes

The quality of the online MCA programme is determined by the professionalism of the curriculum, which is designed to meet the demands of the business management profession. A well-structured syllabus, coupled with dedicated efforts and effective course execution, plays a critical role in ensuring the program's success. The primary goal of the online MCA programme is to equip students with comprehensive knowledge and practical skills in management. Additionally, expertise in information communication technology (ICT) gained

through the programme opens up new career opportunities, allowing students to enhance their job prospects and elevate their positions, both in the workplace and in society at large.

The effectiveness of the programme will be assessed through various benchmarks, including the performance of students in their final semester examinations. These results will provide valuable insight into the programme's impact on student learning and skill acquisition. Moreover, continuous feedback from key stakeholders will play an important role in maintaining and enhancing the quality of the programme. By collecting and analysing such feedback, the programme can be refined to better meet the evolving needs of business management students and the corporate sector.

CDOE, DSU has constituted Centre for Internal Quality Assurance (CIQA). DSU to conduct periodic review and assessments and assist CDOE to implement necessary quality measures and effectiveness in programme delivery. CIQA is constantly involved in reviewing all materials prepared by CDOE, including syllabus, SLMs and e-learning content. CIQA will be involved in conducting studies to measure effectiveness of methods adopted for learning. As we proceed further, CIQA will involve in benchmarking quality of academic delivery, and perform various analyses, and guide all stakeholders towards upgrading quality constantly.

Centre for Internal Quality Assurance Committee (CIQAC) chaired by the Vice Chancellor consisting of internal and external experts oversees the functioning of Centre for Internal Quality Assurance and approve the reports generated by Centre for Internal Quality Assurance on the effectiveness of quality assurance systems and processes.

In addition to CIQA, as per the guidelines of National Assessment and Accreditation Council (NAAC), DSU has constituted Internal Quality Assurance Cell (IQAC), in which academicians, industry representatives and other stakeholders are nominated as members. The IQAC is a part of the institution's system and work towards realization of the goals of quality enhancement and sustenance, as quality enhancement is a continuous process. The prime task of the IQAC is to develop a system for conscious, consistent, and catalytic improvement in the overall performance of institutions. The work of the IQAC is the first step towards internalization and institutionalization of quality enhancement initiatives. IQAC's elementary motive is to promote measures for institutional functioning towards quality enhancement through internalization of quality culture and institutionalization of best practices.

The guidelines on quality monitoring mechanism prescribed by the UGC have been adopted by the Centre for Internal Quality Assurance for conducting institutional quality audits, to

promote quality assurance and enhance as well as spread best-in-class practices of quality assurance. University has setup an effective system for collecting feedback from the stakeholders regularly to improve its programmes. The University will conduct self-assessments regularly and use the results to improve its systems, processes etc. and finally quality of programmes.

The Master of Computer Applications (MCA) programme is designed to develop comprehensive computing professionals through a holistic approach to technological education. The programme cultivates advanced computational knowledge, enabling students to apply mathematical and domain-specific principles in solving complex computing challenges. Graduates are equipped with robust analytical skills to identify, research, and resolve intricate technological problems while maintaining a critical understanding of solution design, evaluation, and broader societal implications. The curriculum emphasizes technological proficiency, ethical professional practice, and continuous learning, preparing students to effectively use modern computing tools, adhere to professional standards, and adapt to rapidly evolving technological landscapes. By fostering research capabilities, innovation, and entrepreneurial thinking, the programme develops well-rounded technology professionals who can excel in diverse environments and create value through cutting-edge computing solutions that address individual and societal needs.

Registrar 28/8/25

Dayananda Sagar University
Bangalore