Maharashtra Education Society's

Abasaheb Garware College

(Autonomous)

(Affiliated to Savitribai Phule Pune University)

Three Year B.Sc. Degree Program **in Computer Science**

(Faculty of Science and Technology)

T.Y.B.Sc. (Computer Science)

Choice Based Credit System Syllabus
Syllabi under NEP
To be implemented from Academic Year 2025-2026

Structure of the major for Course: Third year B.Sc. (Computer Science)
Major : Computer Science

Year	Semester	Course Type	Course Code	Course Title	Remark	Credit	No. of hours
3	V	Core	CS-301-MJ	Web Technology – I		2	30
			CS-302-MJ	Object Oriented Programming – I		2	30
			CS-303-MJ	Operating Systems		2	30
			CS-304-MJP	Practical based on Web Technology - I		2	4 per batch
			CS-305-MJP	Practical based on Object Oriented Programming – I		2	4 per batch
		Elective	CS-316-MJE	Theoretical Computer Science		2	30
			CS-317-MJE	Network Security		2	30
		VSC	CS-321-VSC	Cloud Computing		2	30
		FP	CS-331-FP	Field Project		2	
	VI	Core	CS-351-MJ	Web Technology – II		2	30
			CS-352-MJ	Object Oriented Programming – II		2	30
			CS-353-MJ	Data Analytics		2	30
			CS-354-MJP	Computer Science Laboratory based on Web Technology – II		2	4 per batch
			CS-355-MJP	Computer Science Laboratory based on Object Oriented Programming – II		2	4 per batch
		Elective	CS-366-MJE	Compiler Construction		2	30
			CS-367-MJE	Software Testing		2	30
		OJT	CS-381-OJT	On job training		4	

Minor : Data Science

Year	Semester	Course Type	Course Code	Course Title	Remark	Credit	No. of hours
3	V	Minor	CS-341-MN	Linear Algebra and Optimization Techniques		2	30
			CS-342-MNP	Practical based on Linear Algebra and Optimization Techniques		2	4 per batch
	VI	Minor	CS-391-MN	Statistical methods for Data Science		2	30
			CS-392-MNP	Practical based on Statistical methods for Data Science		2	4 per batch

Minor : Internet of Things (IoT)

Year	Semester	Course Type	Course Code	Course Title	Remark	Credit	No. of hours
3	V	Minor	CS-343-MN	IoT wireless Communication and Introduction to Cloud technology		2	30
			CS-344-MNP	Practical based on IoT wireless Communication and Introduction to Cloud Technology		2	4 per batch
	VI	Minor	CS-393-MN	IoT and Applications		2	30
			CS-394-MNP	Practical based on IoT and Applications		2	4 per batch

CS-301-MJ: Web Technology – I

Lectures: 30 (Credits-2)

Prerequisites:

- Knowledge of HTML, CSS, JavaScript

Learning Objectives:

1. To understand the fundamentals of web technologies and the client-server communication model.
2. To learn the syntax, structure, and fundamental programming concepts of PHP.
3. To develop skills in using functions and string manipulation techniques in PHP.
4. To explore array structures, operations, and their applications in PHP.
5. To understand file handling and directory management in PHP.

Course Outcomes: On completion of this course, students will be able to:

CO1: Describe the key components of web technologies, including HTTP communication and website types.

CO2: Write basic PHP scripts using variables, data types, operators, and control flow statements.

CO3: Implement functions, variable scope, and string manipulation techniques for efficient program development.

CO4: Use arrays for data storage, retrieval, and processing, including sorting and traversal techniques.

CO5: Perform file and directory operations such as reading, writing, renaming, and managing permissions..

Unit 1: Introduction to Web Technologies **02 (CO1)**

1.1 Qualities of a good website

1.2 Web Client-Server and its Communication

1.3 HTTP request message, HTTP response message

1.4 Types of Websites (Static and Dynamic Websites)

Unit 2: Introduction to PHP **02 (CO2)**

2.1 Lexical structure

2.2 Data types

2.3 Variables, operators and flow control statements

2.4 Including code – require and include

2.5 Comments

Unit 3: Function and String **10 (CO3)**

3.1 Defining and calling a function

3.2 Variable scope – Global variables, static variables

3.3 Function parameters – pass by value, pass by reference, Default parameters, Variable parameters, Missing parameters

3.4 Variable function, Anonymous function

3.5 Types of strings in PHP
3.6 Printing Strings
3.7 Comparing strings
3.8 Manipulating strings
3.9 Regular expressions

Unit 4: Array **10 (CO4)**

4.1 Indexed Vs Associative arrays
4.2 Identifying elements of an array
4.3 Storing data in arrays
4.4 Multidimensional arrays
4.5 Extracting multiple values
4.6 Converting between arrays and variables
4.7 Traversing arrays
4.8 Sorting
4.9 Action on entire arrays
4.10 Using arrays – Set, Stack, Queue

Unit 5: Files and directories **06 (CO5)**

5.1 Opening and Closing, Getting information about file, Read/write to file
5.2 Splitting name and path from file, Rename and delete files
5.3 Reading and writing characters in file
5.4 Reading entire file
5.5 Random access to file data
5.6 Getting information on file
5.7 Ownership and permissions
5.8 Working with directories

Reference Books

1. Programming PHP By Rasmus Lerdorf and Kevin Tatroe, O'Reilly publication
2. Beginning PHP5, Wrox publication
3. PHP for Beginners, SPD publication
4. Mastering PHP BPB Publication
5. PHP cookbook O'Reilly publication

CS-302-MJ: Object Oriented Programming - I

Lectures: 30 (Credits-2)

Prerequisites: Object Oriented Programming concepts

Learning Objectives:

1. Learn how to write, compile, and run Java programs.
2. To learn implementation of object-oriented concepts.
3. To introduce the principles of inheritance and polymorphism; and demonstrate how they relate to the design of abstract classes.
4. To introduce the concepts of exception handling and file handling.
5. To introduce the design of Graphical User Interface using swing controls.

Course Outcomes: On completion of this course, students will be able to:

CO1: Execute basic java programs by accepting input from user using different ways.

CO2: Identify classes, objects, class members and relationships for a given problem.

CO3: Implement different types of inheritance and interfaces.

CO4: Handle abnormal termination of a program using exception handling

CO5: Use Java APIs for GUI development.

Unit 1: Introducing Java **04 (CO1)**

1.1 A Short History of Java
1.2 Features of Java
1.3 Java Environment – Compiler, Interpreter, JVM
1.4 Simple java program
1.5 Types of Comments
1.6 Declaring single and multi-dimensional arrays
1.7 Accepting input using Command line arguments
1.8 Accepting input from console (Using BufferedReader and Scanner class)

Unit 2: Classes and Objects **06 (CO2)**

2.1 Defining a class
2.2 Access Specifiers (public, protected, private, default)
2.3 Array of Objects
2.4 Constructors, Overloading Constructors and Use of 'this' keyword
2.5 static block, static fields and methods
2.6 Concept of Garbage Collection
2.7 Predefined Classes
    2.7.1 Object Class, Methods (equals(), toString(),hashcode(), getClass())
2.8 Creating, Accessing and Using Packages
2.9 Wrapper Classes

Unit 3: Inheritance and Interface **06 (CO3)**

3.1 Inheritance Basics (extends Keyword) and Types of Inheritance

3.2 Superclass, Subclass and use of Super Keyword
3.3 Method Overriding and runtime polymorphism
3.4 Use of final keyword related to method and class
3.5 Use of abstract class and abstract methods
3.6 Defining and Implementing Interfaces
3.7 Runtime polymorphism using interface

Unit 4: Exception and File Handling **06 (CO4)**

4.1 Exception class, Checked And Unchecked Exception
4.2 Catching Exceptions, Multiple Catch Block, Nested try block
4.3 Creating User Defined Exception
4.4 String class (Basic methods), StringBuffer class
4.5 Introduction to Files and Streams Input-OutputStream : FileInput/OutputStream, DataInput/OutputStream
4.6 Reader-Writer : FileReader/Writer, BufferedReader/Writer

Unit 5: Swing **08 (CO5)**

5.1 Basics of Swing
5.2 The MVC Architecture
5.3 Layout Manager and Layouts
5.4 Components – JLabel, JButton, JText, JTextArea, JCheckBox, JRadioButton, JComboBox,
5.5 Event Handling: Event sources, Listeners – ActionListener, ItemListener
5.6 Mouse and Keyboard Event Handling
5.7 Adapters – MouseAdapter, KeyAdapter
5.8 Anonymous inner class

Reference Books:

1. Core Java Volume I - Fundamentals By Cay S. Horstmann, 11th Edition, Prentice Hall, ISBN 978-0-13-516630-7
2. The Complete Reference By Herbert Shildt, 11th Edition, McGraw Hill Education, ISBN 978-260-44023-2
3. Java Beginners Guide By Herbert Shildt, 8th Edition, McGraw-Hill Education ISBN 978-1-260-44021-8
4. Java 2 programming black books, StevenHorlzner

CS-303-MJ: Operating Systems

Lectures – 30 (Credits – 2)

Prerequisites:

- Basic Computer Architecture concepts.
- Basic algorithms and data structure concepts

Learning Objectives:

1. To understand the concept of operation system and its principle
2. To study the various functions and services provided by operating system
3. To understand the notion of process and threads
4. To learn various process scheduling algorithms
5. To learn various memory management techniques
6. To understand the concept of File system management & disk scheduling

Course Outcomes: On completion of this course, students will be able to:

CO 1 : learn the structure and architecture of operating systems.
CO 2 : understand services provided by operating system and how they are provided.
CO 3 : understand the concept of process and thread.
CO 4 : understand and compare various process scheduling algorithms.
CO 5 : understand various memory management techniques and study their advantages and disadvantages.
CO 6 : learn file system management and disk scheduling techniques.

Unit 1: Introduction to Operating Systems **02 (CO 1, CO 2)**

1.1 Operating System Definition
1.2 Computer System Architecture – Single processor, Multiprocessor, Clustered
1.3 Operating System Structure – Simple, Layered, Microkernels, Modules and Hybrid
1.4 Computing Environments- Traditional, mobile, distributed, Client/server, peer to peercomputing
1.5 Operating System services
1.6 User and operating system interface
1.7 Booting
1.8 System calls
1.9 System Programs

Unit 2: Processes and Threads **03 (CO 3)**

2.1 Process Concept – The processes, Process states, Process control block
2.2 Process Scheduling – Scheduling queues, Schedulers, context switch
2.3 Operations on Process – Process creation, Process termination, Synchronization of parent and child
2.4 Threads – Overview, Benefits of threads, Multithreading Models

Unit 3: CPU Scheduling **06 (CO 4, CO 2)**

3.1 Basic Concept

- 3.1.1 CPU-I/O burst
- 3.1.2 CPU scheduler
- 3.1.3 Preemptive scheduling
- 3.1.4 Dispatcher
- 3.1.5 Scheduling Criteria

3.2 CPU Scheduling Algorithms

- 3.2.1 FCFS
- 3.2.2 SJF
- 3.2.3 Priority scheduling
- 3.2.4 Round-robin scheduling
- 3.2.5 Multiple queue scheduling
- 3.2.6 Multilevel feedback queue scheduling

Unit 4: Memory Management **06 (CO 5, CO 2)**

4.1 Introduction

- 4.1.1 Basic hardware
- 4.1.2 Address binding
- 4.1.3 Logical versus physical address space
- 4.1.4 Dynamic loading
- 4.1.5 Dynamic linking and shared libraries

4.2 Swapping

4.3 Contiguous Memory Allocation

- 4.3.1 Memory mapping and protection
- 4.3.2 Memory allocation
- 4.3.3 Fragmentation

4.4 Segmentation memory management mechanism

- 4.4.1 Basic concept
- 4.4.2 Segmentation Hardware

4.5 Paging memory management mechanism

- 4.5.1 Basic Concept
- 4.5.2 Hardware support
- 4.5.3 Protection
- 4.5.4 Shared Pages

Unit 5: Virtual Memory Management **06 (CO 5, CO 2)**

5.1 Overview

5.2 Demand paging

- 5.2.1 Basic concept
- 5.2.2 Performance of demand paging

5.3 Page replacement algorithms

- 5.3.1 FIFO
- 5.3.2 Optimal
- 5.3.3 LRU
- 5.3.4 MFU
- 5.3.5 Second chance

Unit 6: File system Management **07 (CO 6, CO 2)**

- 6.1 File Concept
  - 6.1.1 File attributes
  - 6.1.2 File operations
- 6.2 Types of file storage (Object, file, block)
- 6.3 Access Methods
  - 6.3.1 Sequential
  - 6.3.2 Direct
  - 6.3.3 Other access methods
- 6.4 Directory overview
  - 6.4.1 Single level directory
  - 6.4.2 Two level directory
  - 6.4.3 Tree structure directory
  - 6.4.4 Acyclic graph directory
  - 6.4.5 General graph directory
- 6.5 Allocation Methods
  - 6.5.1 Contiguous allocation
  - 6.5.2 Linked allocation
  - 6.5.3 Indexed allocation
- 6.6 Free Space Management
  - 6.6.1 Bit vector
  - 6.6.2 Linked list
  - 6.6.3 Grouping
  - 6.6.4 Counting

Reference Books:

1. Operating System Concepts, Abraham Silberschatz, Peter Galvin, Greg Gagne, Student Edition, Wiley Asia
2. Operating Systems: Internals and Design Principles, William Stallings, Prentice Hall of India.
3. Advanced Concepts in Operating Systems, M Singhal and NG Shivaratri, Tata McGraw Hill Inc, 2001

CS-304-MJP: Practical based on Web Technology – I

No. of Sessions - 12 (Credits-2)

Assignments:

1. Assignments on basics of PHP.
2. Assignments on function in PHP.
3. Assignments on String manipulations.
4. Assignments on Arrays.
5. Assignments on File and directory handling.

CS-305-MJP: Practical based on Object Oriented Programming – I
No. of Sessions - 12 (Credits-2)

Assignments:

1. Java Tools and IDE, Simple Java Programs
2. Array of Objects and Packages
3. Inheritance and Interfaces
4. Exception and File Handling
5. GUI Designing, Event Handling

CS-316-MJE: Theoretical Computer Science

Lectures: 30 (Credits-2)

Prerequisites

- Mathematical Preliminaries Sets (Subset, Set Operations), Relations (Properties of Relations, Closure of Relations) and Functions
- Discrete Mathematics- Graphs, Trees, Logic and Proof Techniques

Course Objectives

1. To understand the Finite Automata, Pushdown Automata and Turing machine
2. To understand the Regular Language, Context Free Language, Context Sensitive Language and Unrestricted Language.
3. To understand the relation between Automaton and Language

Learning Outcomes: On completion of this course, students will be able to:

CO 1: Understand the use of automata during language design.
CO 2: Relate various automata and Languages.
CO 3: Implement CFG grammer to design the Pushdown Automata machine for formal language problems.
CO 4: design the Turing machine for formal language problems.

Unit 1: Finite Automaton **10 (CO1)**

1.1 Introduction: Symbol, Alphabet, String, Prefix and Suffix of Strings, Formal Language, Operations on Languages
1.2 Deterministic finite Automaton – Definition, DFA as language recognizer, DFA as pattern recognizer
1.3 Nondeterministic finite automaton – Definition and Examples
1.4 NFA To DFA (Myhill Nerode Method)
1.5 NFA with ε- transitions Definition and Examples
1.6 NFA with ε-Transitions to DFA and Examples
1.7 Finite automaton with output – Mealy and Moore machine, Definition and Examples
1.8 Minimization of DFA, Algorithm and Problem using Table Method

Unit 2: Regular Expressions and Languages **05 (CO 2)**

2.1 Regular Expressions (RE): Definition and Example
2.2 Regular Expressions Identities
2.3 Regular language-Definition and Examples
2.4 Conversion of RE to FA-Examples
2.5 Pumping lemma for regular languages and applications
2.6 Closure Properties of regular Languages

Unit 3: Context-Free Grammars and Languages **08 (CO 3)**

3.1 Grammar - Definition and Examples
3.2 Derivation-Reduction - Definition and Examples
3.3 Chomsky Hierarchy
3.4 CFG: Definition and Examples. LMD, RMD, Parse Tree

3.5 Ambiguous Grammar: Concept and Examples
3.6 Simplification of CFG: Removing Useless Symbols, Unit Production, ϵ-production and Nullable Symbol
3.7 Normal Forms: Greibach Normal Form (GNF) and Chomsky Normal Form (CNF)
3.8 Regular Grammar: Definition
3.9 Left linear and Right Linear Grammar-Definition and Example

Unit 4: Push Down Automata **05 (CO 3)**

4.1 Definition of PDA and examples.
4.2 Construction of PDA using empty stack and final State method: Examples using stack method
4.3 Definition DPDA and NPDA, their correlation and Examples of NPDA
4.4 CFG (in GNF) to PDA: Method and examples

Unit 5: Turing Machine **02 (CO 4)**

5.1 The Turing Machine Model, Definition and Design of TM
5.2 Language accepted by TM
5.3 Problems on language recognizers

Reference Books:

1. Introduction to Automata Theory, Languages and Computation, John E. Hopcraft, Rajeev Motwani, Jeffrey D. Ullman, Third Edition, Pearson Education Publication, 2008
2. Introduction to Automata theory, Languages and computation By John E. Hopcroft and JeffreyUllman – Narosa Publishing House, 1995
3. Theory of Computer Science Automata, Languages and Computation, K.L.P. Mishra, N. Chandrasekaran, Publication- Prentice Hall of India, 2008
4. Introduction to Computer Theory Daniel I. A. Cohen – 2nd edition – John Wiley & Sons, 1996
5. Introduction to Languages and The Theory of Computation John C. Martin The McGrawHill, Fourth Edition, 2011

CS-317-MJE: Network Security

Lectures: 30 (Credits - 2)

Prerequisites:

- Networking Protocols & Models – OSI & TCP/IP models, TCP, UDP, and HTTP/S.
- IP Addressing & Security – IPv4, IPv6, subnetting, firewalls, and VPNs.

Learning Objectives:

1. To understand fundamental security concepts, cryptographic techniques, and their applications.
2. To analyze symmetric key encryption algorithms and their modes of operation.
3. To explore asymmetric key cryptography and its use in securing communications.
4. To examine internet security protocols and mechanisms for data protection.
5. To evaluate network security measures, including firewalls, VPNs, and authentication mechanisms.

Course Outcomes: On completion of the course, the student will be able to

CO1: Describe key security principles, cryptographic techniques, and common types of cyber threats.

CO2: Demonstrate the use of symmetric key cryptographic techniques and analyze their security features.

CO3: Implement asymmetric cryptographic methods, digital signatures, and message digests.

CO4: Apply internet security protocols such as SSL/TLS and SHTTP for secure communication.

CO5: Assess network security mechanisms, including firewalls, VPNs, and authentication techniques.

Unit 1: Computer Security **04 (CO1)**

1.1 Security Principles and Need for Security
1.2 Fundamentals of Cryptography
1.3 Plaintext and Ciphertext
1.4 Substitution Techniques and Transposition Techniques
1.5 Symmetric vs. Asymmetric Key Cryptography
1.6 Steganography: Techniques and Applications
1.7 Key Range, Key Size, and Attacks on Ciphertext

Unit 2: Symmetric Key Algorithms **10 (CO2)**

2.1 Introduction to Symmetric Key Cryptography
2.2 Types and Modes of Symmetric Key Algorithms
2.3 Overview of Symmetric Key Cryptography
2.4 Data Encryption Standard (DES) and its Variants
2.5 Diffie-Hellman Key Exchange Algorithm

Unit 3: Asymmetric Key Algorithms **08 (CO3)**

3.1 History and Overview of Asymmetric Key Cryptography
3.2 RSA Algorithm: Working and Applications
3.3 Symmetric vs. Asymmetric Key Cryptography
3.4 Digital Signatures: Concept and Implementation
3.5 Message Digests and Their Uses
3.6 Challenges in Public Key Exchange

Unit 4: Internet Security Protocols **08 (CO4)**

4.1 Digital Certificates and Public Key Infrastructure (PKI)
4.2 Fundamentals of Internet Security
4.3 Secure Socket Layer (SSL) and Transport Layer Security (TLS)
4.4 Secure HyperText Transfer Protocol (SHTTP)
4.5 Time Stamping Protocol (TSP)

Unit 5: Network Security, Firewalls, and Virtual Private Networks (VPNs) **08 (CO5)**

5.1 Introduction to Firewalls and Their Role in Network Security
5.2 IP Security (IPSec) and VPNs
5.3 Intrusion Detection and Prevention Systems
5.4 User Authentication Mechanisms:
5.5 Authentication Basics: Password-Based Authentication, Authentication Tokens, Biometric Authentication, Image-Based Authentication

Reference Books

1. Cryptography and Network Security – Atul Kahate, 4th Edition, McGraw Hill Publication
2. Cryptography and Network Security – William Stallings, 8th Edition, Pearson Publication
3. Network Security Essentials: Applications and Standards – William Stallings, 6th Edition, Pearson Publication
4. Cryptography: Theory and Practice by Douglas Stinson, CRC Press, CRC Press LLC.

CS-321-VSC: Cloud Computing

Lectures: 30 (Credits-2)

Prerequisites:

- Computer Network

Learning Objectives:

1. To understand the concept and evolution of cloud computing.
2. To learn virtualization technology and its role in cloud computing.
3. To explore storage architecture and how to analyze and plan storage solutions.
4. To identify major cloud service providers and their offerings.
5. To understand the design of security architecture in cloud computing.

Course Outcomes: On completion of this course, students will be able to:

CO1: Explain the fundamental concepts of cloud computing, its models, features, and infrastructure components.

CO2: Explore virtualization technologies, their implementation.

CO3: Design and assess storage networks for cloud computing, considering virtualization and storage technologies.

CO4: Identify, evaluate, and deploy cloud services and applications provided by leading cloud platforms.

CO5: Explain cloud security concerns, risk management practices, and security architecture principles in a cloud environment.

Unit 1: Introduction to Cloud Computing **05 (CO1)**

1.1 Cloud computing Overview

1.2 Types of Cloud – public, private, hybrid, community

1.3 Layers of Cloud – IaaS, PaaS, SaaS

1.4 Features of a Cloud Computing

1.5 Cloud Infrastructure Components

Unit 2: Abstraction and Virtualization **06 (CO2)**

2.1 Virtualization Technology

2.2 Types of Virtualization

2.3 Benefits of Virtualization

2.4 Virtualization for Data Center Automation

2.5 Hypervisors and Types of Hypervisors

2.6 Virtual Machine – System and Process VM

2.7 Virtual Machines provision and manageability

2.8 Load Balancing

Unit 3: Storage Network Design **07 (CO4)**

3.1 Architecture of storage, analysis and planning

3.2 Storage network design considerations - NAS and FC SANs, hybrid storage networking technologies (iSCSI, FCIP, FCoE)

3.3 Design for storage virtualization in cloud computing
3.4 Host system design considerations.

Unit 4: Deploying Applications and cloud services **06 (CO5)**

4.1 Cloud service providers
4.2 Microsoft Cloud Services
4.3 Google Cloud Applications
4.4 Exploring SaaS – salesforce.com

Unit 5: Security in the Cloud **06 (CO5)**

5.1 Security Architecture Design
5.2 Cloud Security Challenges and Risks
5.3 Software-as-a-Service Security
5.4 Risk Management – Security Monitoring

Reference Books

1. Cloud Computing: Principles and Paradigms, Editors, Rajkumar Buyya, James Broberg, Andrzej M. Goscinski, Wiley, 2011.
2. Enterprise Cloud Computing - Technology, Architecture, Applications, Gautam Shroff, Cambridge University Press, 2010.
3. Cloud Security: A Comprehensive Guide to Secure Cloud Computing, Ronald L. Krutz, Russell Dean Vines, Wiley- India, 2010.
4. AWS, The ultimate guide from beginners to advanced, Maveric Koston.
5. Microsoft Azure: Planning, Deploying, and Managing Your Data Center in the Cloud, Anthony Puca, Mike Manning, Marshal Copeland, Julian Soh, David Gollob.

CS-331-FP: Field Project

No. of sessions – 12 (Credits – 2)

Prerequisites: Software Engineering

Course Objectives:

1. To understand concepts of Project Management.
2. To know how various tools for development and management of software projects are used to carry out various tasks involved.
3. To learn the importance of project documentation.

Learning Outcomes: On completion of this course, students will be able to:

1. Demonstrate a sound technical knowledge of selected project topic.
2. Apply techniques for project management.
3. Create various documents used during the development of the project and a project report.

Project Implementation Guidelines:

1. Students shall choose anytopic for project work in consultation with project guide, Project In-charge.
2. The students shall work on a Project in a group of **three** students.
3. Students are expected to work on the chosen project during the entire semester.
4. Students shall undertake application oriented/web-based/database-oriented/research-based work.
5. Students shall successfully implement the chosen work. Only a hypothetical / theoretical study shall not be accepted.
6. Students shall choose any appropriate programming language/ platform, computational techniques and tools in consultation with the guide, In-charge.
7. The faculty members from affiliated college shall act as a project guide for each project group with equal distribution of groups amongst each eligible faculty.
8. The guide shall track and monitor the project progress on a weekly basis by considering the workload of 4 laboratory hours per week.
9. The project work shall be evaluated based on the novelty of the topic, scope of the work, relevance to computer science, adoption of emerging techniques/technologies and its real-world application etc.
10. Students shall prepare a project report with the following contents:

    a) Title Page

    b) Certificate

    c) Index Page detailing description of the following with their sub sections:-

    - Title: A suitable title giving the idea about what work is proposed.
    - Introduction: An introduction to the topic giving proper background of the topic.
    - Requirement Specification: Specify Software/hardware/data requirements.
    - System Design details

:Methodology/Architecture/UML/DFD/Algorithms/protocols used (whichever is applicable) - System Implementation: Code implementation.

- System Implementation: Code implementation.
- Results: Test Cases/Tables/Figures/Graphs/Screen shots/Reports etc.
- Conclusion and Future Scope: Specify the Final conclusion and future scope.
- References: Books, web links, research articles etc.

11. The Project report should be prepared in a spiral bound form with adequate number of copies.
12. The Project work and report shall be certified by the concerned Project guide and Head of the department.

Recommended Documentation contents:

Sr. No	Index Name	Page No.
I.	Acknowledgement	
II.	College Certificate	
1.	Introduction I. Existing System II. Need for New System	
2.	Problem Definition	
3.	Proposed System: I. Explanation II. Methodology used	
4.	Scope of the System	
5.	Hardware and Software Requirement	
6.	Fact Finding Techniques	
7.	Feasibility Study I. Operational II. Technical III. Economical	
8.	Diagrams: I. E-R Diagram II. UML Diagrams	
9.	Data Dictionary	
10.	Database Designing	
11.	Screen Designing I. I/O Screen Designing	

	II. Output Formats	
12.	Test Cases Design	
13.	Limitations	
14.	Conclusions and Future Enhancements	
15.	Bibliography, References and Published work (Paper/Book Chapter/Copyright/Patent etc.)	

Evaluation Scheme:

Description	Evaluation	Marks
Continuous Evaluation, Progress Report	Internal	25
Presentation & Project Report	External	10
Demonstration of the Project		10
Viva		05
Total		**50**

Submission of Certified Project Report is mandatory for appearing the Practical Examination (Project).

CS-351-MJ: Web Technologies – II

Lectures: 30 (Credits - 2)

Prerequisites:

- Knowledge of HTML, CSS, JavaScript and basics of PHP

Learning Objectives:

1. To understand web techniques for handling forms, server information, and maintaining state using sessions and cookies.
2. To apply PHP for database interaction and manage relational databases using SQL and PEAR DB.
3. To explore XML and JSON structures.
4. To implement AJAX for handling asynchronous data exchange and basic database interactions using PHP.
5. To learn the fundamentals of the CodeIgniter PHP framework and write simple programs using its core features.

Course Outcomes: On completion of the course, the student will be able to

CO1: Identify and use web techniques such as processing forms, handling server responses, and managing sessions and cookies.

CO2: Implement database interactions using PHP and perform CRUD operations on relational databases with SQL.

CO3: Work with XML and JSON data using PHP.

CO4: Develop AJAX-based scripts to handle asynchronous data exchange and interact with databases using PHP.

CO5: Implement basic programs using CodeIgniter to understand the MVC structure, session handling, and data management.

Unit 1: Web Techniques **08 (CO1)**

1.1 Variables

1.2 Server information

1.3 Processing forms

1.4 Setting response headers

1.5 Maintaining state – session and cookies

Unit 2: Databases **04 (CO2)**

2.1 Using PHP to access a database

2.2 Relational databases and SQL

2.3 PEAR DB basics

Unit 3: PHP and XML **06 (CO3)**

3.1 What is XML?

3.2 XML document Structure

3.3 PHP and XML
3.4 The simple XML extension
3.5 Introduction to JSON
3.6 JSON Data types
3.7 JSON Object
3.8 JSON schema
3.9 JSON Comparison with XML
3.10 Encoding and Decoding JSON in PHP

Unit 4: Ajax **08 (CO4)**

4.1 Introduction of AJAX
4.2 AJAX web application model
4.3 Performing AJAX validation
4.4 Handling XML data using php and AJAX
4.5 Handling JSON data using php and AJAX
4.6 Connecting database using php and AJAX

Unit 5: PHP framework CodeIgniter **04 (CO5)**

5.1 Basic concept of CodeIgniter
5.2 Installing CodeIgniter
5.3 Application Architecture
5.4 MVC Framework
5.5 Libraries Working with databases
5.6 Load external JS and CSS page & redirecting from controller
5.7 Adding JS and CSS, Page redirection
5.8 Loading dynamic data on page & session management, cookies management

Reference Books

1. Programming PHP By Rasmus Lerdorf and Kevin Tatroe, O'Reilly publication
2. Beginning PHP5, Wrox publication
3. PHP for Beginners, SPD publication
4. AJAX Black Book Kogent solution
5. Mastering PHP BPB Publication
6. PHP cookbook O'Reilly publication
7. Professional Codeigniter By Thomas Myer, Wrox Publication
8. Codeigniter 2 CookBook By Rob Foster, PACKT Publication

CS-352-MJ: Object Oriented Programming - II
Lectures: 30 (Credits-2)

Prerequisites: Knowledge of Core Java

Learning Objectives:

1. To understand different various types of data structures.
2. To develop a game application using multithreading.
3. To learn database programming using Java.
4. To study web development concept using Servlet using webserver.
5. To use JSP for developing static and dynamics pages.

Course Outcomes: On completion of this course, students will be able to:

CO1: Implement different data structure while developing applications.

CO2: Built multithreading applications.

CO3: Access open database through Java programs using Java Data Base Connectivity (JDBC) and develop the application.

CO4: Understand and create dynamic web pages, using Servlets and handling of tomcat webserver.

CO5: Create static and dynamic web pages using JSP implicit objects and directives.

Unit 1: Collections **05 (CO1)**

1.1 Introduction to the Collection framework
1.2 List - ArrayList, LinkedList
1.3 Set - HashSet, TreeSet,
1.4 Map - HashMap and TreeMap
1.5 Interfaces such as Iterator, ListIterator, Enumeration

Unit 2: Multithreading **07 (CO2)**

2.1 What are threads?
2.2 Life cycle of thread
2.3 Creating threads - Thread class , Runnable interface
2.4 Thread priorities
2.5 Running multiple threads
2.6 Synchronization and interthread communication

Unit 3: Database Programming **07 (CO3)**

3.1 The design of jdbc
3.2 Types of drivers
3.3 Executing sql statements, query execution
3.4 Metadata – Databasemetadata, ResultSetMetadata
3.5 Scrollable and updatable Resultset

Unit 4: Servlets **06 (CO4)**

4.1 Introduction to Servlets and Hierarchy of Servlets
4.2 Life cycle of a servlet
4.3 Tomcat configuration (Note: Only for Lab Demonstration)
4.4 Handing get and post request (HTTP)
4.5 Handling a data from HTML to a servlet
4.6 Session tracking – Cookies and Http Session

Unit 5: JSP **05 (CO5)**

5.1 Simple JSP program
5.2 Life cycle of a JSP
5.3 Implicit Objects
5.4 Scripting elements – Declarations, Expressions, Scriplets, Comments
5.5 JSP Directives – Page Directive, include directive

Reference Books:

1. Core Java Volume-I-Fundamentals, Eighth Edition, Cay S. Horstmann, Gary Cornell, Prentice Hall, Sun Microsystems Press
2. Core Java Volume-II-Advanced Features, Eighth Edition, Cay S. Horstmann, Gary Cornell, Prentice Hall, Sun Microsystems Press

CS-353-MJ: Data Analytics

Lectures: 30 (Credits-2)

Prerequisites:

- Basic of mathematics and statistics
- Basic programming Knowledge of python
- Knowledge of databases

Learning Objectives:

1. To understand the concept of data analytics, its types, and basic evaluation models used in analyzing data.
2. To explore the modeling process in machine learning and understand different learning approaches and regression techniques.
3. To learn key data mining techniques including frequent pattern mining, association rule generation, and predictive modeling.
4. To analyze social media data using network-based approaches and understand key social media analytics techniques.
5. To understand and apply basic text analytics techniques, including NLP, sentiment analysis, and document summarization.

Course Outcomes: On completion of this course, students will be able to:

CO 1: Understand Data Analytics and its types, know about different models for Data Analysis

CO 2: Use appropriate models of analysis, assess the quality of input, and derive insight from results.

CO 3: Understand different data mining techniques like classification, prediction, clustering and association rule mining

CO 4: Apply modeling and data analysis techniques to the solution of real-world business problems

CO 5: Analyze data, choose relevant models and algorithms for respective applications

Unit 1: Introduction to Data Analytics **05 (CO 1)**

1.1 Concept of data analytics
1.2 Data analysis vs Data analytics
1.3 Types of analytics
  1.3.1 Diagnostic Analytics
  1.3.2 Predictive Analytics
  1.3.3 Prescriptive Analytics
  1.3.4 Exploratory Analysis
  1.3.5 Mechanistic Analysis
1.4 Mathematical models - Concept
1.5 Model evaluation
  1.5.1 Metrics for evaluating classifiers
  1.5.2 Class imbalance - AUC, ROC (Receiver-Operator Characteristic) curves
  1.5.3 Evaluating value prediction models

Unit 2: Machine Learning Overview **08 (CO 2)**

2.1 Introduction to Machine Learning, deep learning, Artificial intelligence
2.2 Applications for machine learning in data science
2.3 The modeling processes
  2.3.1 Selecting a model
  2.3.2 Training the model
  2.3.3 Validating the model,
  2.3.4 Predicting new observations
2.4 Types of machine learning
  2.4.1 Supervised learning
  2.4.2 Unsupervised learning
  2.4.3 Semi-supervised learning
2.5 Regression models
  2.5.1 Linear Regression
  2.5.2 Logistic Regression
  2.5.3 Polynomial Regression
2.6 Concept of classification, clustering

Unit 3: Mining Frequent Patterns, Associations, and Correlations **08 (CO 3)**

3.1 Class/Concept Description: Characterization and Discrimination
3.2 Mining Frequent Patterns, Associations, and Correlations
3.3 Classification and Regression for Predictive Analysis
3.4 Cluster Analysis
3.5 Outlier Analysis
3.6 Mining frequent patterns - Market Basket Analysis.
3.7 Frequent Itemsets, Closed Itemsets, and Association Rules
3.8 Frequent Itemset Mining Methods
3.9 Apriori Algorithm
3.10 Generating Association Rules from Frequent Itemsets
3.11 Frequent pattern growth (FP-growth) algorithm

Unit 4: Social Media Analytics **05 (CO 4, CO 5)**

4.1 Overview of social media analytics
4.2 Social Media Analytics Process, Seven layers of social media analytics, accessing social media data
4.3 Key social media analytics methods
4.4 Social network analysis
4.5 Link prediction, Community detection, Influence maximization, Expert finding, Prediction of trust and distrust among individuals
4.6 Challenges to social media analytics

Unit 5: Text Analytics **04 (CO 4, CO 5)**

5.1 Introduction to Natural Language Processing
5.2 Text Analytic: Tokenization, Bag of words, Word weighting: TF-IDF, n- Grams, stop words, Stemming and lemmatization, synonyms and parts of speech tagging
5.3 Sentiment Analysis
5.4 Document or text summarization
5.5 Trend analytics

Reference Books:

1. Data Science Fundamentals and Practical Approaches, Gypsy Nandi, Rupam Sharma, BPB Publications, 2020.
2. A Hands-On Introduction to Data Science, Chirag Shah, University of Washington Cambridge University Press
3. The Data Science Handbook, Field Cady, John Wiley & Sons, Inc, 2017
4. Data Mining Concepts and Techniques, Jiawei Han, Micheline Kamber, Jian Pei, Morgan Kaufmann, Third Edition, 2012.

CS-354-MJP: Practical based on Web Technology – II

No. of Sessions : 12 (Credits-2)

Assignments:

1. Self-Processing, Sticky forms and File upload
2. COOKIES and SESSIONS
3. File Handling
4. XML documents, JSON object
5. AJAX
6. PHP framework CodeIgniter

CS-355-MJP: Practical based on Object Oriented Programming – II

No. of Sessions: 12 (Credits-2)

Assignments:

1. Collection
2. Multithreading
3. Database programming
4. Servlets
5. JSP

CS-366-MJE: Compiler Construction

Lectures: 30 (Credits-2)

Prerequisites:

- Knowledge of Automata Theory and Languages

Course Objectives:

1. To understand the phases of compiler
2. To understand the design issues of a lexical analyzer and use of LEX tool.
3. To understand design issues of a parser and use of YACC tool.
4. To understand and design code generation and optimization techniques

Learning Outcomes: On completion of this course, students will be able to:

CO 1: Learn the phases of compiler and its working.
CO 2: Understand the process of scanning and LEX tool.
CO 3: Understand the process of parsing of source code. And YACC utility.
CO 4: Learn the semantic analysis phase to performed the type conversion.
CO 5: Learn the conversion code written in source language to machine language.

Unit 1: Introduction **03 (CO 1)**

1.1 Definition of Compiler, Aspects of compilation
1.2 The structure of Compiler
1.3 Phases of Compiler – Lexical Analysis, Syntax Analysis, Semantic Analysis, Intermediate Code generation, code optimization, code generation
1.4 Error Handling
1.5 Introduction to one pass and Multipass compilers, cross compiler, Bootstrapping

Unit 2: Lexical Analysis (Scanner) **04 (CO 2)**

2.1 Review of Finite automata as a lexical analyzer
2.2 Applications of Regular Expressions and Finite Automata (lexical analyzer, searching using RE), Input buffering, Recognition of tokens
2.3 LEX: A Lexical analyzer generator (Simple Lex Program)

Unit 3: Syntax Analysis (Parser) **13 (CO 3)**

5.1 Definition, Types of Parsers
5.2 Top-Down Parser:
  5.2.1 Top-Down Parsing with Backtracking: Method & Problems
  5.2.2 Drawbacks of Top-Down parsing with backtracking
  5.2.3 Elimination of Left Recursion (direct and indirect)
  5.2.4 Need for Left Factoring and examples
5.3 Recursive Descent Parsing: Definition, Implementation of Recursive Descent Parser Using Recursive Procedures
5.4 Predictive [LL (1)] Parser (Definition, Model):
  5.4.1 Implementation of Predictive Parser [LL (1)]
  5.4.2 FIRST & FOLLOW

5.4.3 Construction of LL (1) Parsing Table
5.4.4 Parsing of a String using LL (1) Table

5.5 Bottom-Up Parsers

5.6 Operator Precedence Parser:
5.6.1 Basic Concepts
5.6.2 Operator Precedence Relations form Associativity and Precedence
5.6.3 Operator Precedence Grammar
5.6.4 Algorithm for LEADING & TRAILING (with ex.)
5.6.5 Algorithm for Operator Precedence Parsing (with ex.)
5.6.6 Precedence Functions

5.7 Shift Reduce Parser:
5.7.1 Reduction
5.7.2 Handle
5.7.3 Handle Pruning
5.7.4 Stack Implementation of Shift Reduce Parser (with examples)

5.8 LR Parser:
5.8.1 Model, Types [SLR (1), Canonical LR, LALR]-Method and examples.

5.9 YACC (from Book 3):
5.9.1 Program sections
5.9.2 simple YACC program for expression evaluation

Unit 4: Syntax Directed Definition **05 (CO 4)**

4.1 Syntax Directed Definitions (SDD)
4.1.1 Inherited & Synthesized Attributes
4.1.2 Evaluating an SDD at the nodes of a Parse Tree, Example
4.1.3 Evaluation Orders for SDD's
4.1.4 Dependency Graph

4.2 Ordering the Evaluation of Attributes
4.2.1 S-Attributed Definition
4.2.2 L-Attributed Definition

4.3 Application of SDT

4.4 Construction of syntax trees

4.5 The Structure of a Type

4.6 Translation Schemes
4.6.1 Definition, Postfix Translation Scheme

Unit 5: Code Generation and Optimization **05 (CO 5)**

5.1 Compilation of expression:
5.1.1 Concepts of operand descriptors and register descriptors with example

5.2 Intermediate code for expresions:
5.2.1 postfix notations
5.2.2 Triples
5.2.3 Quadruples
5.2.4 Expression trees

5.3 Code Optimization:
5.3.1 Optimizing transformations

5.3.2 compile time evaluation
5.3.3 elimination of common sub expressions
5.3.4 dead code elimination
5.3.5 frequency reduction
5.3.6 strength reduction

5.4 Three address code:
5.4.1 DAG for Three address code
5.4.2 The Value-number method for constructing DAG's.
5.4.3 Definition of basic block, Basic blocks, and flow graphs
5.4.4 Directed acyclic graph (DAG) representation of basic block.

Reference Books:

1. Compilers: Principles, Techniques, and Tools, Alfred V. Aho, Ravi Sethi, Jeffrey D. Ullman, 2004
2. Principles of Compiler Design By: Alfred V. Aho, Jeffrey D. Ullman, Narosa Publication House, 2002
3. LEX & YACC, 2nd edition, O'reilly Publication, 2012

CS-367-MJE: Software Testing

Lectures: 30 (Credits-02)

Prerequisites: None

Learning Objectives:

1. To provide the knowledge of software testing techniques.
2. To understand how testing methods can be used as an effective tool in quality assurance of software.
3. To provide skills to design test case plan for testing software.
4. To provide knowledge of latest testing methods.

Course Outcomes: On completion of this course, students will be able to:

CO1: To understand fundamental concepts in software testing

CO2: To understand various software testing methods and strategies.

CO3: To design test cases and test plans.

CO4: To understand basic concept of agile testing. CO5: Understand the different automation tools

Unit 1: Introduction to Software Testing 04(CO1)

1.1 Introduction to Software Testing
1.2 Objectives of Software Testing
1.3 Basics of Software Testing – faults, errors and failures
1.4 Principles of testing
1.5 STLC- Software Testing Life Cycle

Unit 2: Software Testing Strategies and Techniques 14(CO2)

2.1 Functional testing vs. non-functional testing
2.2 Types of Testing-Unit Testing, Integration Testing, System Testing, White Box Testing, Black Box Testing, Alpha Testing, Beta Testing, Regression Testing, Performance Testing, User Acceptance Testing (UAT)
2.3 White Box Testing - Basis path testing, Control Structure Testing
2.4 Black Box Testing- Boundary Value Analysis, Equivalence partitioning Differences between BBT & WBT
2.5 Internationalization testing
2.6 Usability and accessibility testing
2.7 Compatibility testing
2.8 Defect Life Cycle, Classification of Defect
2.9 Defect Report

Unit 3: Test Cases and Test Plan **05(CO3)**

3.1 Test case Template
3.2 Write Test Plan for given application with resources required
3.3 Write Test case for Mobile, Web application using Excel
3.4 Prepare Test report for test cases executed

Unit 4: Agile Testing **03(CO4)**

4.1 Introduction to Agile testing
4.2 Difference between traditional testing and Agile testing
4.3 Agile principles and values
4.4 Agile Testing Quadrants

Unit 5: Testing Tool **04(CO5)**

5.1 How to make use of Automation Tools
5.2 Types of Testing Tools
5.3 Demo of Automation Tools

Reference Books:

1. Effective Methods of Software Testing, William E Perry, 3rd Edition, Wiley Publishing Inc.
2. Software Testing Principles and Practices by Srinivasan Desikan, Gopalaswamy Ramesh, Pearson.
3. Software Engineering – A Practitioners Approach, Roger S. Pressman, 7th Edition, Tata McGraw Hill, 20
4. Agile Testing: A Practical Guide for Testers and Agile Teams, Lisa Crispin and Janet Gregory, 1st Edition, Addison-Wesley Professional, 2008

CS-341-MN: Linear Algebra and Optimization Techniques

Lectures: 30 (Credits-2)

Learning Objectives:

1. Understand various concepts of linear algebra and use it as a platform to model physical problems
2. Appreciate analytical and numerical solutions of linear equations
3. Obtain eigenvalues of a matrix using analytical and numerical methods.
4. Use Simplex methods and its variants to solve linear programming problems.

Course Outcomes:

CO1: A student has understood matrix as a basic tool of linear algebra and can use various matrix transformations.

CO2: A student can solve system of linear equations using methods – Gauss Elimination method, Gauss Jordan method, Crammer's Rule.

CO3: Student can solve system of equations using numerical methods.

CO4: Student can find eigenvalues of a matrix – both analytic and numerical methods.

CO5: Student can formulate and solve linear programming problems using Simplex method, Big M method.

Unit 1. Matrices, Vectors, Determinants, Linear Systems 10 (CO1, CO2)

1.1 Overview Matrices, Vectors: Addition and Scalar Multiplication, Matrix Multiplication
1.2 Linear Systems of Equations, Gauss Elimination
1.3 Linear Independence, Rank of a Matrix
1.4 Solutions of Linear Systems: Existence, Uniqueness, Gauss Elimination method
1.5 Second- and Third-Order Determinants
1.6 Determinants, Cramer's Rule
1.7 Inverse of a Matrix, Gauss–Jordan method

Unit 2. Numeric Linear Algebra 6 (CO3)

2.1 Linear Systems: Gauss Elimination
2.2 Linear Systems: Solution by Iteration (Gauss Seidel method and Gauss Jordan method)

Unit 3. Matrix Eigenvalue Problems 4 (CO4)

3.1 The Matrix Eigenvalue Problem: Determining Eigenvalues and Eigenvectors
3.2 Symmetric, Skew-Symmetric, and Orthogonal Matrices

Unit 4. Linear Programming (LP) Problems 10 (CO5)

4.1 Two-variable LP Model
4.2 Properties of Linear Programming Problems (LPP)
4.3 Graphical LP Solution
4.4 LP in standard equation form
4.5 The Simplex Method
4.6 Big M method

Textbooks:

1. Erwin Kreyszig, Advanced Engineering Mathematics, Wiley India, 10th Edition, 2011.
2. Hamdy A Taha, Operations Research: An Introduction, Pearson/PHI, 10th Edition, 2018.

Reference Books:

1. Bernard Kolman, David R. Hill, Elementary Linear Algebra with Applications, Ninth Edition, 2008.
2. Ravindran, Phillips and Solberg, Operations Research: Principles and Practice, John Wiley, Second Edition, 2000.

CS-342-MNP: Practical based on Linear Algebra and Optimization Techniques

No. of Sessions: 12 (Credits-2)

Following Assignments to be done using Python-

1. Introduction to Python (Emphasis on matrices and Linear Algebra)
2. Determinant, Rank, Solving system of linear equations – inbuilt functions
3. Solving system of equations – Gauss elimination method (using row transformations)
4. Matrix eigenvalues problem – inbuilt functions, power method
5. Gauss Seidel method, Gauss Jordan method
6. Solving LPP using Python using PuLP
7. Solving LPP using Python using Scipy

CS-391-MN: Statistical Methods for Data Science

Lectures: 30 (Credits-2)

Learning Objectives:

1. Providing the knowledge of basics of Probability and the distribution related
2. Providing understanding of various sampling techniques related various applications in Data Science
3. Introducing the concepts of Hypothesis testing

Corse Outcomes:

CO1: Students acquired ability to understand the scenarios to apply suitable probability distribution models
CO2: Students have learnt discrete random variable, discrete frequency distributions and application of Chebyshev's theorem.
CO3: Students have learnt continuous random variables and distributions
CO4: Students will get knowledge about sampling distribution
CO5: Students have gained sufficient knowledge related to sampling and hypothesis formulation

Unit 1. Basic concepts of Probability 6(CO1)

1.1 Sample Spaces and Events
1.2 Counting
1.3 Probability
1.4 The Axioms of Probability
1.5 Some Elementary Theorems
1.6 Conditional Probability
1.7 Bayes' Theorem

Unit 2. Discrete random variables, Probability Distributions 8 (CO2)

2.1 Random Variables
2.2 The Binomial Distribution
2.3 The Hyper geometric Distribution
2.4 The Mean and the Variance of a Probability Distribution
2.5 Chebyshev's Theorem
2.6 The Poisson Approximation to the Binomial Distribution

Unit 3. Continuous random variable 6 (CO3)

3.1 Continuous Random Variables
3.2 The Normal Distribution
3.3 The Normal Approximation to the Binomial Distribution
3.4 The Uniform Distribution

Unit 4. Sampling Distribution 6 (CO4)

4.1 Populations and Samples
4.2 The Sampling Distribution of the Mean (known), Central limit theorem
4.3 The Sampling Distribution of the Mean (unknown), t and Chi-square distribution
4.4 The Sampling Distribution of the Variances

4.5 Point Estimation
4.6 Interval Estimation, Confidence intervals

Unit 5. Testing of hypotheses **4 (CO5)**

5.1 Tests of Hypotheses
5.2 Null Hypotheses and Tests of Hypotheses
5.3 Hypotheses Concerning One Mean

Text Book:

1. Richard A., Miller & Freund's Probability and Statistics for Engineers, PHI, 8th Ed., 2011

Reference Books :

1. Paul L. Meyer, Introductory Probability and Statistics Appl., Second Edition, Addison- Wesley, 1970
2. Beaver Beaver, Introduction to Probability and Statistics, Thomson, 12th Ed., 2007.

CS-392-MNP: Practical based on Statistical Methods for Data Science

No. of Sessions: 12 (Credits-2)

Following assignments to be solved manually or using Python.

1. Computation of Probability Bay's Theorem and univariate probability distribution
2. Binomial distribution, Hypergeometric distribution
3. Chebyshev's theorem, Poisson distribution, Approximation of Poisson to Binomial
4. Normal distributions, Uniform distributions
5. Construction of confidence interval for population mean μ for normal distribution (i) σ^2 – known, (ii) σ^2 – unknown
6. Construction of confidence interval for population varianceσ^2 for normal distribution (i) μ – known, (ii) μ – unknown
7. Testing of Hypothesis

CS-343-MN: IoT Wireless Communication and Introduction To Cloud Technology

Lectures: 30 (Credits-2)

Learning Objectives

1. Understand the basic principles of communication systems, types of communication, and error detection and correction techniques.
2. Explain the structure and functioning of cellular technologies, including GSM, GPRS, and the evolution from 1G to 5G.
3. Describe the working and applications of short-range wireless technologies such as Bluetooth, Zigbee, RFID, and GPS.
4. Understand the fundamentals of cloud computing, including service and deployment models, and its relevance to IoT applications.

Course Outcomes: Students will be able to

CO1: Know basics of communication

CO2: Become familiar with working of wireless technologies such as Mobile communication, GSM, GPRS, 3G and 4G Cellular Network Technologies for Data Connections

CO3: Understand working principles of short-range communication application CO4: Get introduced to Cloud Technology

Unit1: Wireless Communication: Cellular Telephony 08(CO1)

1.1 Introduction to Communication: Elements of Communication system, Electromagnetic spectrum,

1.2 Types of communication: simplex, half duplex, full duplex, baseband and broadband, Serial communication: asynchronous and synchronous

1.3 Information Theory: Signal bandwidth and channel bandwidth, rate of information (data rate, baud rate), channel capacity, Nyquist theorem, Signal to noise ratio, Noise Figure, Shannon theorem

1.4 Error handling, hamming code

Unit 2: Wireless Communication: cellular technology 08(CO2)

2.1 Introduction of cellular telephony system: Frequency reuse, handoff strategies, Co-channel and adjacent channel interference, block diagram of mobile handset, Overview of Cellular Telephony generations: 1G to 5G,3G (W-CDMA, UMTS), 4G(LTE)

2.2 GSM: architecture, frame structure, mobility management,

2.3 GPRS: architecture, application

Unit 3: Short Range Wireless Technologies and Location Tracking 08(CO3)

Short range Technologies :

3.1 Bluetooth: Bluetooth architecture, Bluetooth protocol stack, Bluetooth frame Structure, applications of bluetooth

3.2 Zigbee: Architecture, topologies, applications od RFID

3.3 RFID: working of RFID system, types of RFID tags, applications Location Tracking

3.4 GPS system: components of GPS system (space segment, control segment, user segment), Applications of GPS

Unit 4: Introduction to cloud technology 06(CO4)

4.1 Introduction to Cloud Storage . Service modes of cloud: IaaS, PaaS, SaaS

4.2 Deployment models of cloud: Private cloud, public cloud, hybrid cloud and community cloud

4.3 Applications

4.4 Introduction to Amazon Web Services for IoT

Reference books:

1. Wireless Communications Principles and Practice, Rappaport, Pearson Publication
2. Mobile Communications, Jochen Schiller, Pearson publication
3. Internet of Things : Principles and Paradigms, Rajkumar Buyya and Dastjerdi, MK publishers

CS-344-MNP: Practical based on IoT Wireless Communication and Introduction to Cloud Technology

No. of Sessions - 12 (Credits-2)

1. Introduction to Python programming
2. Study of GSM system (Message transmission & Reception)
3. To study working of SIM card in GSM handset
4. Study of GPRS system
5. Study of Zig-bee for one application
6. Study of RFID system
7. Study of Error detection and correction by using Hamming Code technique
8. Study of signal strength measurement.
9. Study of Computer network components: Cables, Connectors, Routers, Switches, Ethernet and related interfacing cards

Compulsory Activity equivalent to two practical: (any one)

1. To study Configuration of IP and MAC address and to study Local Area Network setup
2. To study Amazon web services and Skynet and prepare a report based on the features, advantages, limitations and applications.

CS-393-MN: Internet of Things and Applications

Lectures: 30 (Credits-2)

Prerequisites: None

Learning Objectives

1. Explain the fundamental concepts, architecture, and enabling technologies of the Internet of Things (IoT).
2. Apply IoT design methodology and analyze communication protocols used in IoT systems.
3. Describe the role of cloud platforms in managing and processing IoT data.
4. Design basic IoT applications for domains such as smart homes, agriculture, healthcare, and education.

Course Outcomes: On completion of this course, students will be able to:

CO1: Define the Internet of Things

CO2: IoT design methodology and Analyze protocols for communication among IoT devices

CO3: Understand cloud platforms for IOT

CO4: Design various applications of IOT

Unit I: Internet of Things Concepts **08 (CO1)**

1.1 Definition, Characteristics of IoT, Trends in Adoption of IoT, IoT Devices, IoT Devices Vs Computers,

1.2 Basic Building Blocks.

1.3 Physical Design of IoT: - Things in IoT, Interoperability of IoT Devices, Sensors and Actuators, Need of Analog/Digital Conversion.

1.4 Logical Design of IoT:- IoT functional blocks, IoT Enabling technologies, IoT levels and deployment templates, Applications of IoT

Unit 2: Introduction to IOT design methodology and Protocols **10 (CO2)**

2.1 Design Steps

2.2 Basics of IoT Networking, Networking Components, Internet Structure

2.3 IoT Communication Models and IoT Communication APIs,

2.4 Sensor Networks

2.5 Protocol Standardization for IoT

2.6 M2M and WSN Protocols

2.7 RFID Protocol,

2.8 Modbus Protocol, Zigbee Architecture.

2.9 IP based Protocols: MQTT (Secure), 6LoWPAN, LoRa

Unit3: Cloud Platforms for IOT **06 (CO3)**

3.1 Introduction to Cloud Storage Models, Communication API

3.2 Cloud for IoT

3.3 Introduction to Amazon Web Services for IoT and SkyNet IoT

Unit 4: Applications of IOT **06 (CO4)**

4.1 IOT for Smart home
4.2 Smart agriculture using IOT
4.3 IOT based health care system
4.4 IOT in education

Reference Books

1. Arshdeep Bahga,Vijay Madisetti, "Internet of Things– A hands- on approach", Universities Press, ISBN: 0: 0996025510, 13:978-996025515
2. Olivier Hersent, David Boswarthick, Omar Elloumi, "The Internet of Things: Key Applications and Protocols", 2nd Edition, Wiley Publication,ISBN:978-1-119-99435-0
3. Dawoud Shenouda, Peter Dawoud, "Microcontroller and Smart Home Networks", ISBN: 9788770221566, e-ISBN: 9788770221559
4. Charles Crowell, "Internet of Things for Beginners: An Easy-to-Understand Introduction to IoT", ISBN-13 : 979-8613100194
5. David Hanes, Gonzalo Salgueiro, Robert Barton, Jerome Henry, "IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things", Cisco Press, ISBN-13: 978-1-58714-456-1 ISBN-10: 1-58714-456-5

CS-394-MNP: Practical based on Internet of Things and Applications
No. of Sessions - 12 (Credits-2)

Compulsory Activity equivalent to two practicals:
Study any two latest IoT platforms: features, advantages, limitations and applications.

1. Study of IOT based agriculture system
2. Study of IOT based Temperarture and humidity monitoring system for an office cabin
3. Study of IOT based light monitoring system for smart streets
4. Study of IOT based energy saver at home
5. Study of IOT based patient monitoring system
6. Study of IOT based attendance system (RFID based)

Exploring Internet of Things with Cisco Packet Tracer:

7. Create smart home packet tracker using CISCO packet tracker
8. Study environment controls and IOT things in Packet tracker.

CS-381-OJT: On Job Training

Total duration: 120 Hours (Credits-4)

Learning Objectives:

1. To provide students with practical, hands-on-experience in applying theoretical knowledge to real-world tasks
2. To help students develop and enhance their skills, problem solving abilities and work culture of the industry
3. To foster effective teamwork and collaboration skills
4. To encourage students to build and expand their professional network by interaction with experienced experts and mentors in industry

Course Outcomes: On Completion of this course, student will be able to -

CO1: Enhance the knowledge related to various tools and technologies used in industry CO2: Improve the ability to solve complex problems independently and creatively

CO3: Effectively utilize critical thinking and analytical skills in tackling real world challenges

CO4: Effectively communicate and collaborate skills through interaction with team members and mentors.

CO5: Get an experience in working on projects or related working within industry

CO6: Develop the ability to document process, design, implementation and testing

CO7: Familiar with specific industry domain relevant to internship

CO8: Complete projects and tasks as per the predetermined objectives

Guidelines for On Job Training (OJT)

1. Student must start the OJT/Internship immediately after semester-V examination during the vacation
2. Student are expected to complete the IT related work/project within 120 hours assigned by organization (company/ industry/ consultancy/ institution)
3. The internship work may involve the IT related assignment(s) OR the maintenance of existing project OR the design/development of new project OR equivalent work
4. College should assign the mentors/guides for students to monitor the progress throughout the OJT
5. Students have to submit the weekly progress report duly signed by the concern authorities of organization to the assigned mentor
6. At the end of OJT, students should prepare the documentation and submit a report to the college in prescribed format
7. After completion, the final presentation and documentation will be evaluated by the examination panel