Rajarshi Janak University
Faculty of Science, Technology and Engineering
Office of the Dean

Bachelor of Science Computer Science and Information Technology (BSc CSIT)
Second Year, Fourth Semester

Effective from 2024

Second Year/ Fourth Semester		
Course Code	**Course Title**	**Credit Hours**
CSIT 401	Computer Graphics	3
CSIT 402	Operating System	3
CSIT 403	System Analysis and Design	3
CSIT 404	Java Programming	3
CSIT 405	Web Technology	3

Rajarshi Janak University
Institute of Science, Technology and Engineering
Course of Study for BSc CSIT
(Second Year/ Fourth Semester)

Course Title: Computer Graphics | **Course Code**: CSIT 401
Nature of Course: Theory + Practical | **Full Mark**: 60+20+20
Credit Hours: 3 Cr. | **Pass Marks:** 24+8+8

Course Description

Computer Graphics is an undergraduate course that introduces the fundamental principles, algorithms, mathematical foundations, and programming techniques used to create and manipulate graphical content in computer systems. The course covers 2D and 3D graphics transformations, rasterization algorithms, clipping, object representation, visible surface detection, illumination and shading models, rendering techniques, and modern graphics programming using OpenGL and introductory shader concepts. Through theoretical study and practical laboratory implementation, students develop the ability to design and implement interactive graphical applications used in areas such as visualization, animation, simulation, virtual reality, and game development.

Course Objectives

By the end of the course, students will be able to:

- Understand the theoretical and mathematical foundations of computer graphics.
- Implement classical 2D and 3D graphics algorithms.
- Apply geometric transformations and viewing techniques in both 2D and 3D.
- Gain exposure to contemporary applications in VR/AR and game development.

Course Contents

Unit 1

Introduction to Computer Graphics **[5 hrs.]**

Introduction and Applications of Computer Graphics, Evolution, Overview of graphics hardware (monitors, GPUs, modern display technology: LCD, OLED, VR/AR systems, GPU and Graphics accelerators), Raster and Random Scan Displays and Architecture, Display Processor, Graphics pipeline: From modeling to rendering, Brief history: From raster/CRT to real-time rendering, Input Devices, Graphics Software, Need for machine independent graphics language.

Unit 2

Scan Conversion Algorithms **[7 hrs.]**

Introduction, Basic graphics primitives, Line drawing algorithms: DDA, Bresenham's Line Drawing Algorithm with derivations, numerical example and implementation, Circle and ellipse drawing: Midpoint algorithms with derivations and numerical examples.

Polygons, Polygon fill areas and fill-area primitives, polygon representation and filling algorithms, Inside-outside test, Boundary-fill and Flood-fill algorithm

Unit 3

2D Transformations and Viewing **[6 hrs.]**

Coordinate systems, Basic 2D transformations: Translation, rotation, scaling, reflection, shear, Homogeneous coordinates and matrix representation, Composite transformations, 2D viewing pipeline, Window-to-viewport coordinate transformation.

Clipping: Introduction, clipping Window, Normalization and viewport transformations, Point Lines, Point and Lines Clipping Algorithm (Cohen-Sutherland, Liang-Barsky), Polygon fill area clipping: Sutherland-Hodgeman Polygon Clipping Algorithm.

Unit 4

3D Transformations and Viewing **[5 hrs.]**

3D co-ordinate systems, 3D transformations: Translation, rotation, scaling, affine transformations, composite transformation, 3D viewing pipeline: world, viewing, and screen coordinates, Projection techniques: orthographic, parallel, perspective projections.

Unit 5

3D Object Representations **[6 hrs.]**

Representing Surfaces: Boundary and space partitioning, Polygon Surfaces: Polygon tables, Surface normal and Spatial orientation of surfaces, Plane Equations, Polygon meshes, Wireframe Representation, Blobby Objects,

Curves: Parametric Cubic Curves, Spline Interpolation, Cubic Spline Interpolation, Hermite Curves, Bezier and B-spline curve. Quadric Surfaces: Sphere and Ellipsoid

Unit 6

Solid Modeling **[3 hrs.]**

Solid Modeling Concepts, Sweep representations, Boundary and spatial-partitioning representation, Binary Space Partition Trees (BSP), Octree, Ray Tracing, Rasterization vs ray tracing

Unit 7

Visible Surface Detection **[5 hrs.]**

Hidden Surface Problems, Image Space and Object Space Techniques, Back Face Detection, Depth Buffer (Z - buffer), A- Buffer, Scan-Line Algorithms, Depth Sorting Method (Painter's Algorithm)

Unit 8

Illumination, Shading, and Rendering Techniques **[5 hrs.]**

Introduction to light and color models: RGB, CMY, HSV, illumination models: Ambient, diffuse, specular, Phong reflection, Polygon rendering techniques: Flat shading, Gouraud shading, Phong shading, Fast Phong Shading.

Unit 9

OpenGL and Graphics Programming **[3 hrs]**

OpenGL coordinate systems, Drawing primitives: points, lines, polygons, OpenGL transformations and viewing functions, Using modern OpenGL (shader pipeline basics).

Laboratory Work

The laboratory course consists of implementing the following algorithm using high level programming languages and OpenGL

1. DDA Line Algorithm
2. Bresenham's line drawing algorithm
3. Midpoint Circle Algorithm
4. Midpoint Ellipse Algorithm
5. Basic transformation on 2D including Translation, Rotation and Scaling
6. Simple 3D Object with basic transformations including Translation, Rotation and Scaling
7. Clipping
8. Bezier Curve Generation
9. Hidden surface removal
10. Basic Drawing Techniques in OpenGL

Textbooks / References

1. Donald Hearn & Pauline Baker, ***Computer Graphics with OpenGL***,3rd/4th Edition Pearson Education, 2011.
2. Edward Angel**,** ***Interactive Computer Graphics: A Top-Down Approach with OpenGL***., 5th edition, Pearson Education, 2008
3. Tomas Akenine-Möller et al., ***Real-Time Rendering***, A K Peters/CRC Press.
4. John F. Hughes et al**.,** ***Computer Graphics: Principles and Practice***.

Rajarshi Janak University

Institute of Science ,Technology and Engineering

Course of Study for B.Sc. CSIT

(Second Year/Fourth Semester)

Course Title: Operating System | **Course Code:** CSIT 402

Nature of Course: Theory + Practical | **Full Mark:** 60+20+20

Credit hrs.: 3 | **Pass Mark:** 24+8+8

Course Description:

This course includes the topics that help students understand operating system and its functionality along with its types. It focuses on Process Management, Memory Management, Synchronization, file system, Security, Virtualization and Cloud Operating system.

Course Objectives

By the end of the course, students will:

- Understand the fundamental concepts of operating systems.
- Learn how modern OS manage resources like CPU, memory, storage, and devices.
- Gain practical experience in system-level programming.
- Explore current trends in OS development including containerization, virtualization, and Mobile OS.
- Understand security and performance trade-offs in OS design.

Course Contents

Unit 1

Introduction to Operating Systems and System Structure **[9 hrs]**

Definition, Evolution, and Purpose of OS, Types of Operating Systems, Embedded OS, Mobile OS Architecture,System Structures and Components, Booting Process and Initialization

Unit 2: Process Management **[9 hrs]**

Concept of Process and Threads, Scheduling Concepts and Scheduling Criteria, CPU scheduling algorithms (FCFS, SJF, Round Robin, and Priority Scheduling) , Communication (IPC),Context switching and process synchronization, Synchronization primitives/Mechanisms, System calls and process creation using fork(), exec(), wait(), open(),read(),write().

Unit 3: Synchronization and Deadlocks **[6 hrs]**

Critical section problem, Classical problems: Dining Philosophers, Producer-Consumer, Readers-Writers, Deadlock: Conditions, Prevention, Avoidance (Banker's Algorithm), Detection and Recovery

Unit 4: Memory Management **[6 hrs]**

Memory Allocation and Address Mapping Techniques, Contiguous memory allocation, Paging, Segmentation, Memory Management techniques :page replacement algorithms(FIFO,LRU,Optimal),segmentation,fragmentation,swapping and demand paging, Virtual Memory, Thrashing and working set model

Unit 5: Storage and File System Management **[5 hrs]**

File systems, directory structures, file access methods, file protection mechanism, File allocation (Contiguous, Linked, Indexed),File system mounting and protection, Disk scheduling(FCFS, SSTF, SCAN), Handling Solid State vs Hard Disk Drives in Modern OS,I/O system, Interrupt handling,Polling vs Interrupts,Direct Memory Access (DMA) and Free Space Handling in Operating Systems

Unit 6: Security and Protection **[5 hrs]**

Security goals, Access Control, User authentication and password management, Secure Operating System Models, Operating System Security: Threat Detection, Containment, and Protection, OS-level malware detection and sandboxing, Viruses, Worms, and Malware Protection, Role of OS in System Security.

Unit 7: Operating Systems in Cloud Computing **[5 hrs]**

Virtualization, Virtual Machines and Hypervisors and its types, Containerization, Operating Systems for Cloud Computing, Role of OS in cloud environments

Laboratory Works

Lab works should be done covering all the topics listed below and a small project work should be carried out using the concept learnt in this course. Project should be assigned on Individual Basis. The lab should be conducted using the Linux.

1. Understand and practice basic Linux commands, file permissions, redirection, pipes, and shell scripting in a Linux environment.
2. Create C programs to demonstrate system calls and process creation using fork(), exec(), wait(), getpid(), and getppid().
3. Simulate CPU scheduling algorithms such as FCFS, SJF, Round Robin, and Priority Scheduling using C programs.
4. Implement inter-process communication mechanisms including pipes, message queues, shared memory, and semaphores in Linux.
5. Implement classic process synchronization problems such as Producer-Consumer, Reader-Writer, and Dining Philosophers using synchronization techniques.
6. Simulate memory management techniques including paging, page replacement algorithms, and segmentation in C.
7. Develop C programs to perform file system operations and simulate different file allocation methods in Linux.
8. Simulate disk scheduling algorithms including FCFS, SSTF, SCAN, and LOOK to analyze disk access performance.
9. Create shell scripts to monitor system resources, automate backups, and schedule tasks using cron jobs.
10. Perform Linux user and security management tasks including user creation, permission handling, and privilege management.

Textbooks/ Reference Books:

1. A. Silberschatz, P. B. Galvin, and G. Gagne, ***Operating System Concepts***, 10th ed. Hoboken, NJ, USA: Wiley, 2018.
2. W. Stallings, ***Operating Systems: Internals and Design Principles***, 9th ed. Hoboken, NJ, USA: Pearson, 2018.
3. A. S. Tanenbaum and H. Bos, ***Modern Operating Systems***, 4th ed. Upper Saddle River, NJ, USA: Pearson, 2015.
4. R. Love, ***Linux System Programming: Talking Directly to the Kernel and C Library***, 2nd ed. Sebastopol, CA, USA: O'Reilly Media, 2013.
5. J. Levin, ***Android Internals: A Confectioner's Cookbook*.** Cupertino, CA, USA: Technologeeks Press, 2014.

Rajarshi Janak University
Institute of Science, Technology and Engineering
Course of Study for BSc CSIT
(Second Year/ Fourth Semester)

Course Title: System Analysis and Design **Course Code:** CSIT 403
Nature of Course: Theory + Practical **Full Mark:** 60+20+20
Credit hrs. : 3 Cr. **Pass Mark:** 24+8+8

Course Description:
This course provides knowledge and skills for analyzing, designing, and developing information systems. It introduces the Systems Development Life Cycle (SDLC), project management and planning, requirement analysis, process and data modeling, design issues, implementation and maintenance. The course also covers basics of Object-Oriented Analysis and Design (OOAD) using UML diagrams.

Course Objectives:

Upon successful completion of this course, students will be able to:

- Understand the concepts of system analysis and design.
- Do project management of information systems.
- Plan for system development.
- Perform process and data modeling.
- Design database, forms and interface of a system.
- Understand issues in system implementation and maintenance
- Perform object-oriented based analysis and design of a system

Course Contents:

Unit I: Introduction to System Development [6 Hrs.]
System; Information System; System Analysis and Design; Modern Approach to Systems Analysis and Design; Developing Information Systems; Systems Development Life Cycle; Systems Development Process; Systems Development Life Cycle; System Analyst and its role; Waterfall Model; Prototyping; Spiral Model; Agile Methodologies: eXtreme and Scrum; Origins of Software; Systems Acquisition and Reuse

Unit 2: Project Management [6 Hrs.]
Introduction; Managing the Information Systems Project: Initiating, Planning, Executing and Closing Down the Project; Representing and Scheduling Project Plans: Representing Project Plans, Expected Time Durations Using PERT, Gantt chart, Network Diagram, and Critical Path

Unit 3: Planning [5 Hrs.]
Identifying and Selecting Systems Development Projects: Process, Deliverables and Outcomes; Corporate and Information Systems Planning: Strategic and Systems Planning; Initiating and Planning Systems Development Projects: Process, Deliverables and Outcomes; Assessing Project Feasibility; Types of Feasibility; Building and Reviewing the Baseline Project Plan

Unit 4: System Analysis **[10 Hrs.]**

Performing Requirements Determination; Methods for Requirements Determination: Traditional, Contemporary, and Radical; Requirement Determination using Agile Methodologies, Functional vs. Non-Functional Requirements, Use Cases and Functional Requirements

Process Modeling; Data Flow Diagramming Mechanics: Symbols, Rules, Decompositions and Balancing; Guidelines for Drawing DFDs; Modeling Logic with Decision Tables
Conceptual Data Modeling; Gathering Information for Conceptual Data Modeling; Introduction to E-R Modeling: Entities, Attributes, Keys, Relationships; Conceptual Data Modeling and the E-R Model: Degree of Relationship, Cardinality and Participation in Relationship, Naming Relationships, Associative Entities; Extended E-R Model; Representing Super types and Subtypes

Unit 5: System Design **[8 Hrs.]**

Database Design; Normalization: 1NF, 2NF, 3NF, Transforming E-R Diagrams into Relations, Merging Relations, Physical File and Database Design, Designing Forms and Reports, Formatting Forms and Reports , Designing Interfaces and Dialogues, Interaction Methods and Devices, Designing Interfaces, Designing Dialogues Designing Interfaces and Dialogues in Graphical Environments

Unit 6: System Implementation **[4 Hrs.]**

System Implementation; Software Application Testing; Installation; Documenting the System; Training and Supporting Users; Organizational Issues in Systems Implementation

Unit 7: System Maintenance **[2 Hrs.]**

Maintaining Information Systems; Conducting Systems Maintenance

Unit 8: Object-Oriented Analysis and Design **[4 Hrs.]**

Introduction; Class and Objects; UML Diagrams; Structure and Behavior Diagrams; Use-Case, Class, Object, Activity, Sequence, Component, and Deployment Diagrams

Laboratory/Practical Work:

The laboratory work includes realizing system analysis and design practices. The activities include

- Conducting Requirement Collection through different techniques.
- Performing Feasibility Analysis.
- Drawing Gantt chart and PERT chart using appropriate CASE tool.
- Drawing DFDs for processing modelling using appropriate CASE tool.
- Drawing ERDs for data modeling using appropriate CASE tool.
- Designing Database: Converting ER to Relational Schema.
- Designing form and reports.
- Designing interfaces and dialogues.
- Performing OOAD using UML diagrams using appropriate CASE tool.

In addition to above mentioned lab activities students must complete a group project involving analysis and design phases of system development for some information system. Students should prepare project report including analysis and design of the system.

Text/ Reference Book:

1. Valacich J. S. & George J. F. (2024). ***Modern Systems Analysis and Design***, 10th Ed., Pearson Education Limited.
2. Dennis A., Wixom B. H. & Roth R. M. (2022). ***Systems Analysis and Design,*** 8th Ed., Wiley
3. Tilley S. (2024), ***System Analysis and Design***, 13th Ed., Cengage Learning
4. Tegarden D. P., Samuel B., Lukyanenko R., Dennis A., Wixom B. H. & Roth R. M. (2025). ***Systems Analysis and Design: An Object-Oriented Approach with UML,*** 7th Ed., Wiley

Online Resources:

1. The Unified Modeling Language, https://www.uml-diagrams.org/

Rajarshi Janak University

Faculty of Science, Technology and Engineering

Course of Study for BSc. CSIT

(Second Year/ Fourth Semester)

Course Title: Java Programming | **Course Code:** CSIT 404

Nature of Course: Theory + Practical | **Full Mark:** 60+20+20

Credit hrs.: 3 Cr | **Pass Mark:** 24+8+8

Course Description:

This course introduces students to modern Java programming, focusing on object-oriented principles, Java standard libraries, data persistence, multithreading, and enterprise application development. Students will develop practical skills for backend systems using Servlets/JSP, with a brief introduction to Spring Boot and REST APIs for understanding web fundamentals.

Learning Objectives:

Students will have the ability to apply object-oriented programming concepts in Java to design and develop modular, reusable, and maintainable software solutions. They will gain proficiency in working with collections, generics, exception handling, and file I/O to manage data efficiently and handle program errors gracefully. Students will also acquire the skills to build multithreaded applications. Furthermore, they will develop an understanding of HTTP fundamentals through the use of Servlets and JSP, enabling them to create dynamic web applications.

Course Learning Outcomes:

By the end of the course, students will be able to:

- Write efficient Java programs using OOP concepts like inheritance, polymorphism, and encapsulation.
- Implement data structures such as arrays, lists, and maps in Java.
- Develop multithreaded applications and handle synchronization issues.
- Build dynamic web applications using Java Servlets and JSP.
- Solve complex programming problems through hands-on projects and laboratory exercises.

Course Contents:

Unit 1

Introduction to Java **[5 hrs]**

History and features of Java, Java development environment setup (JDK, JRE, JVM), Java program structure, compilation, and execution, Variables, data types, operators, and control structures, array, strings, Input/output operations, operators; control structures (if, switch, loops)

Unit 2

Object Oriented Programming in Java **[7 hrs]**

Classes and objects, Constructors, this keywords, methods, and method overloading, Inheritance, polymorphism, and encapsulation, Abstract classes and interfaces, Wrapper classes, type casting,Packages and access modifiers.

Unit 3

Exception Handling and File I/O **[5 hrs]**

Types of exceptions; Checked vs unchecked, Exception handling: try-catch-finally, throw and throws , custom exceptions; best practices. File handling: Streams (InputStream, OutputStream, Reader, Writer), reading/writing files, Serialization and deserialization

Unit 4

Java Collections Framework & Generics **[5 hrs]**

Introduction to collections, List, Set, Map interfaces, ArrayList, LinkedList, HashSet, HashMap implementations, Iterators and comparators , Introduction to Generics ,Need for Generics Generic classes and methods ,Type safety and reusability , Generic Collections.

Unit 5

Multithreading & GUI Programming **[6 hrs]**

Thread creation: extending Thread class, implementing Runnable, Thread Priorities ,Thread lifecycle and synchronization, Thread pools and executors, GUI Programming Basics: Introduction to AWT,Swing and JavaFX , Components and Containers, Layout Managers , Basic GUI Components.

Unit 6

Database Connectivity with JDBC **[7 hrs]**

Overview of JDBC and its role in Java applications, JDBC architecture and driver types (focus on Type 4: Thin Driver), Setting up database connectivity (MySQL/PostgreSQL) and JDBC configuration, Managing connections, statements, and result sets, Executing DDL and DML operations in Java, Secure data access using PreparedStatement , Basic transaction control (commit, rollback)

Unit 7

Web Development with Java **[7 hrs]**

Introduction to web development with Java, HTTP basics; request/response cycle, Java Servlets: lifecycle, request, and response handling, JavaServer Pages (JSP): syntax, directives, and scripting elements, MVC architecture in web applications, Session management and filters

Unit 8

Modern Web Development with Spring Boot **[3 hrs]**

Introduction to Spring Boot and its ecosystem, Setting up a Spring Boot project, RESTful web services with Spring Boot.

Laboratory Works

The laboratory work consists of implementing the following programs using JAVA programming language using any available IDE.

1. Setting up Java environment, writing basic Java programs.
2. Implementing classes, objects, and inheritance.
3. Handling exceptions and file operations.

4. Working with Java collections (ArrayList, HashMap).
5. Design a generic class and a generic method to verify compile-time type safety and code reusability.
6. Developing multithreaded program.
7. Create Form-based Applications using Swing.
8. Connect Java to a database using JDBC, perform CRUD operations with PreparedStatement
9. Building a simple web application using Servlets and JSP.
10. Build a web application using the Model-View-Controller (MVC) architectural pattern.
11. Developing a RESTful web service using Spring Boot.

Text /References Book

1. Herbert Schildt, "**Java: The Complete Reference**", 12th Edition, Tata McGraw Hill, 2019.
2. Cary S. Horstmann, "**Core Java Volume I – Fundamentals**", Tenth Edition,Prentice Hall
3. Cary S. Horstmann, "**Core Java Volume II – Fundamentals**", Tenth Edition, Prentice Hall
4. C. Walls, **Spring Boot in Action**. Shelter Island, NY, USA: Manning Publications, 2019.

Online References

1. Oracle, "The Java™ Tutorials." [Online]. Available: https://docs.oracle.com/javase/tutorial/
2. Spring, "Spring Boot Documentation." [Online]. Available: https://spring.io/projects/spring-boot
3. Baeldung, "Baeldung Java and Spring Tutorials." [Online]. Available: https://www.baeldung.com

Rajarshi Janak University
Institute of Science, Technology and Engineering
Course of Study for BSc CSIT
(Second Year/ Fourth Semester)

Course Title: Web Technology **Course Code:** CSIT 405
Nature of Course: Theory + Practical **Full Mark:** 60+20+20
Credit Hrs: 3 Cr. **Pass Mark:** 24+8+8

Course Description:
This course provides a comprehensive overview of web development. It introduces students to the fundamentals of web design and development using HTML, CSS, JavaScript, XML, AJAX and PHP.

Course Objectives:
Upon successful completion of this course, students will be able to:

- Understand the basics of web development.
- Get familiar with UI/UX.
- Design and build responsive web pages using HTML and CSS.
- Perform client-side scripting using JavaScript.
- Utilize AJAX for asynchronous web communication.
- Understand concepts of XML.
- Perform server side scripting using PHP.

Course Content:

Unit 1*:* Introduction to Web Technologies [4 hrs.]
Basics of Internet and WWW, Static vs. Dynamic Web Pages, Web Clients and Servers, HTTP Protocol, HTTP Request and Response, URL Structure and Components, Web Architectures: Single and Multi-tier; Client-Side vs. Server-Side Scripting, Evolution of the Web: Web 1.0, 2.0, 3.0, and 4.0.

Unit 2*:* Web Design Principles and UI/UX [4 hrs.]
Elements of Effective Web Design, Website Planning: Structure, Wireframes, Sitemap, UI/UX Design basics: Look & Feel, Navigation, Accessibility, Browser Compatibility, Bandwidth, caching, and display resolution, Responsive Design, and Mobile-First Design

Unit 3: Hyper Text Markup Language [9 hrs.]
HTML Basics, HTML5 Syntax, Page Structure, Text formatting, Lists, Links and Images, Tables, Forms, Semantic Elements, Multimedia: Audio/Video, Meta tags and SEO basics, Character Entities, DOCTYPE, and Validation, XHTML, HTML5 APIs: Geolocation, Web Storage, Drag and Drop, Canvas

Unit 4: Cascading Style Sheets [6 hrs.]
Introduction to CSS, Syntax, Selectors, Inline, Internal and External CSS, Box Model, Layout, Positioning, Flexbox, Grid, Styling Text, Fonts, Borders, Backgrounds, and Lists, CSS3 Features: Media Queries, Animations, Transitions, Shadows, Best practices for Clean and Maintainable CSS

Unit 5: Client Side Scripting using JavaScript **[10 hrs.]**

Basics: Variables, Data Types, Operators, Functions, Loops, Control Structures, DOM Manipulation and Events, Form Validation, Objects and Arrays, Introduction to ES6 Features: let, const, arrow functions, Introduction to jQuery: Selectors, Events, Effects, AJAX integration, DHTML: Combining HTML, CSS, and JavaScript, React Framework: Environment Setup and Installation, JSX, Components, Props, State, Hooks, Forms, DOM Events, Routes, State Management

Unit 6: XML and AJAX **[4 hrs.]**

Introduction to XML, XML Elements, Nested Elements, Introduction to AJAX: Concepts and Applications, XMLHttpRequest and Fetch API, Asynchronous Data Loading, JSON, JSON Parsing,

Unit 7: Server-Side Scripting using PHP **[8 hrs.]**

Introduction to PHP, Variables, Operators, Looping and Control Structures, Functions, Form Validation, Form Handling, Cookies and Session Management, Class and Objects, Inheritance, Interface, Database Connectivity and CRUD Operations, Exception Handling.

Laboratory / Practical Work:

The laboratory work includes creating web pages including following;

1. Creating web pages using HTML elements.
2. Writing inline, internal and external CSS scripts.
3. Writing JavaScript programs for client side scripting
4. Writing programs using React framework.
5. Creating and accessing XML and JSON documents.
6. Implementing AJAX.
7. Writing programs to implement concepts of PHP as mentioned above.

Text / Reference Books:

1. Duckett J. (2022), ***Front-End Back-End Development with HTML, CSS, JavaScript, jQuery, PHP and MySQL***, Wiley
2. Duckett J. (2022), **PHP and MySQL,** Wiley
3. Frain B. (2022), ***Responsive Web Design with HTML5 and CSS***, Packt Publishing
4. Nixon R. (2021), ***Learning PHP, MySQL & JavaScript***, O'Reilly
5. Schwarzmuller M. (2025), ***React Key Concepts***, Packt Publishing
6. Gookin D. (2019), Dan Gookin's ***Guide to XML and JSON Programming***, Wambooli Press

Online Resources:

1. W3Schools – https://www.w3schools.com
2. Mozilla Developer Network (MDN) – https://developer.mozilla.org
3. React- https://react.dev