"Dissemination of Education for Knowledge, Science and Culture" -Shikshanmaharshi Dr. Bapuji Salunkhe



# Shri Swami Vivekanand Shikshan Sanstha's **VIVEKANAND COLLEGE, KOLHAPUR**

(AN EMPOWERED AUTONOMOUS INSTITUTE)

# DEPARTMENT OF COMPUTER SCIENCE Three/Four- Years UG Programme Department/Subject Specific Core or Major (DSC)

# **NEP- 1.0**

# Curriculum, Teaching and Evaluation Structure

(as per NEP-2020 Guidelines)

## for

# **B.Sc.-III Computer Science**

# Semester-V & VI

(With effect from the academic year 2025-26)

### **Department of Computer Science**

#### **B.Sc.:** Program Outcomes (POs):

- **PO 1: Disciplinary Knowledge:** Graduates will gain in-depth understanding in their specific major or discipline, mastering the foundational principles and theories, as well as advanced concepts. Execute theoretical and practical knowledge developed from the specific curriculum.
- **PO 2: Problem-Solving Skills:** Graduates will learn to use their knowledge to identify, analyze and solve problems related to their field of study.
- **PO 3: Analytical Skills:** Graduates will gain the ability to collect, analyze, interpret, and apply data in a variety of contexts. They might also learn to use specialized software or equipment.
- **PO 4: Research Skills and Scientific Temper:** Graduates might learn how to design and conduct experiments or studies, analyze results and draw conclusions. They might also learn to review and understand academic literature.
- **PO 5: Environment and Sustainability:** Possess a sympathetic awareness of the environment while conducting research and scientific studies and focus on sustainable social development.

#### **B.Sc. in Computer Science: Program Specific Outcomes (PSOs):**

- **PSO1:** To obtain sound knowledge in the theory, principles and applications of Computer Science.
- **PSO2:** Develop and build strong problem solving, analyzing and decision-making skills and abilities to solve multidisciplinary problems of various domains associated with Computer Science.
- **PSO3:** Apprised with current trends and concepts of Computer Science and its applications in order to get competence for getting professional status.
- **PSO4:** Ability to understand the computing needs of multidisciplinary problems and to formulate and solve these by applying programming skills, development tools and environments.
- **PSO5:** Integrate knowledge of Computer Science with associated subjects like mathematics, statistics, electronics etc. to build and explore problem solving concepts.
- **PSO6:** Develop skills and knowledge (communication, problem solving, interviewing, creativity, team work, group discussion, aptitude etc.) to make or to build competence among the stakeholders to achieve the career in different fields and at a different level (Industry personnel, academician, researcher, entrepreneur etc).
- **PSO7:** Understand professional and ethical responsibilities in order to work at different positions in organizations and at a societal context.

Sr. No.	Course Abbr.	Course code	Course Name	Teac Sch Hours	ching eme s/week	Exan	Course Credits			
				TH	PR	ESE	CIE	PR	Marks	
Semester-V										
1	DSC-IX	DSC03CSC51	Computer Network	2	_	40	10	-	50	2
2	DSC-X	DSC03CSC52	Software Engineering	2	-	40	10	-	50	2
3	DSC-XI	DSC03CSC53	Internet Technologies-I	2	-	40	10	-	50	2
4	DSE-I	DSE03CSC51	Introduction to Java Programming	2	-	40	10	-	50	2
		DSE03CSC52	PHP Programming						50	2
5	VSC-PR-IV	VSC03CSC59	Introduction to Data Science using Python	-	4	-	-	25	25	2
6	FP	FPR03CSC51	Field Project		4			50	50	2
7	DSC-PR-V	DSC03CSC59	DSC Computer Science Lab-5	-	12	-	-	75	75	6
8	MIN-IX	MIN03CSC51	Software Engineering-I	2	-	40	10	-	50	2
9	MIN-PR-V	MIN03CSC59	Software Engineering Lab-I	-	4	-	-	25	25	2
		/ Total	10	24	200	50	175	425	22	
			Semester-	VI						
1	DSC-XIII	DSC03CSC61	Semester- Advanced Computer Network	<b>VI</b> 2	-	40	10	-	50	2
1	DSC-XIII DSC-XIV	DSC03CSC61 DSC03CSC62	Semester- Advanced Computer Network Object Oriented Software Engineering	<b>VI</b> 2 2 2	-	40 40	10 10	-	50 50	2 2
1 2 3	DSC-XIII DSC-XIV DSC-XV	DSC03CSC61 DSC03CSC62 DSC03CSC63	Semester-Advanced ComputerNetworkObject Oriented SoftwareEngineeringInternet Technologies-II	<b>VI</b> 2 2 2 2 2	-	40 40 40	10 10 10	-	50 50 50	2 2 2
1 2 3 4	DSC-XIII DSC-XIV DSC-XV DSE-II	DSC03CSC61 DSC03CSC62 DSC03CSC63 DSE03CSC61	Semester-         Advanced Computer         Network         Object Oriented Software         Engineering         Internet Technologies-II         Advanced Java         Programming         Advanced DUD	<b>VI</b> 2 2 2 2 2 2		40 40 40 40	10 10 10 10	-	50 50 50 50	2 2 2 2 2
1 2 3 4	DSC-XIII DSC-XIV DSC-XV DSE-II	DSC03CSC61 DSC03CSC62 DSC03CSC63 DSE03CSC61 DSE03CSC62	Semester-'Advanced ComputerNetworkObject Oriented SoftwareEngineeringInternet Technologies-IIAdvanced JavaProgrammingAdvanced PHPProgramming	<b>VI</b> 2 2 2 2 2 2	-	40 40 40 40	10 10 10 10	-	50 50 50 50	2 2 2 2 2
1 2 3 4 5	DSC-XIII DSC-XIV DSC-XV DSE-II VSC-PR-VI	DSC03CSC61 DSC03CSC62 DSC03CSC63 DSE03CSC61 DSE03CSC62 VSC03CSC69	Semester-Advanced ComputerNetworkObject Oriented SoftwareEngineeringInternet Technologies-IIAdvanced JavaProgrammingAdvanced PHPProgrammingMachine Learning for DataScience	VI 2 2 2 2 2	- - - -	40 40 40 40	10 10 10 10 -		50 50 50 50 25	2 2 2 2 2 2
1 2 3 4 5 6	DSC-XIII DSC-XIV DSC-XV DSE-II VSC-PR-VI OJT	DSC03CSC61         DSC03CSC62         DSC03CSC63         DSE03CSC61         DSE03CSC62         VSC03CSC69         OJT03CSC61	Semester-Advanced Computer NetworkObject Oriented Software EngineeringInternet Technologies-IIAdvanced Java ProgrammingAdvanced PHP ProgrammingMachine Learning for Data ScienceOn Job Training	VI 2 2 2 2 2 -	- - - 4 4	40 40 40 -	10 10 10 -	- - - 25 50	50 50 50 50 25 50	2 2 2 2 2 2 2 2 2
1 2 3 4 5 6 7	DSC-XIII DSC-XIV DSC-XV DSE-II VSC-PR-VI OJT DSC-PR-VI	DSC03CSC61         DSC03CSC62         DSC03CSC63         DSE03CSC61         DSE03CSC62         VSC03CSC69         OJT03CSC61         DSC03CSC69	Semester-Advanced ComputerNetworkObject Oriented SoftwareEngineeringInternet Technologies-IIAdvanced JavaProgrammingAdvanced PHPProgrammingMachine Learning for DataScienceOn Job TrainingDSC Computer ScienceLab-6	VI 2 2 2 2	- - - 4 4 12	40 40 40 -	10 10 10 10 -	- - - 25 50 75	50 50 50 50 25 50 75	2 2 2 2 2 2 2 2 2 6
1 2 3 4 5 6 7 8	DSC-XIII DSC-XIV DSC-XV DSE-II VSC-PR-VI OJT DSC-PR-VI MIN-X	DSC03CSC61         DSC03CSC62         DSC03CSC63         DSE03CSC61         DSE03CSC62         OJSC03CSC69         OJT03CSC61         DSC03CSC69         MIN03CSC61	Semester-Advanced Computer NetworkObject Oriented Software EngineeringInternet Technologies-IIAdvanced Java ProgrammingAdvanced PHP ProgrammingMachine Learning for Data ScienceOn Job TrainingDSC Computer Science Lab-6Software Engineering-II	VI 2 2 2 - - 2 - 2 - 2	- - - 4 4 12 -	40 40 40 - - 40 -	10 10 10 - - - 10	- - - 25 50 75 -	50         50         50         50         50         25         50         75         50	2 2 2 2 2 2 2 2 2 6 2
1 2 3 4 5 6 7 8 9	DSC-XIII DSC-XIV DSC-XV DSE-II VSC-PR-VI OJT DSC-PR-VI MIN-X MIN-PR-VI	DSC03CSC61         DSC03CSC62         DSC03CSC63         DSE03CSC61         DSE03CSC62         OJT03CSC63         DSC03CSC63         MIN03CSC61         MIN03CSC69	Semester-Advanced Computer NetworkObject Oriented Software EngineeringInternet Technologies-IIAdvanced Java ProgrammingAdvanced PHP ProgrammingMachine Learning for Data ScienceOn Job TrainingDSC Computer Science Lab-6Software Engineering-IISoftware Engineering Lab-II	VI 2 2 2 - - 2 - 2 - 2 - - 2 - - - 2 - -	- - - 4 4 12 - 4	40 40 40 - - 40 -	10 10 10 - - 10 -	- - 25 50 75 - 25	50         50         50         50         50         25         50         75         50         25	2 2 2 2 2 2 2 6 2 2 2 2 2 2 2 2 2 2 2 2

Thind V. C **X7 0- X7** 

Abbreviations: TH-Theory, PR-Practical, PRO- Project, SEE- Semester End Examination, CIE-Continuous Internal Examination

Minimum passing for 10 marks Internal evaluation Note:

Minimum passing for 40 marks Theory paper Minimum passing for 25 marks Practical

- = 16 marks
- = 10 marks

= 04 marks

Passing percentage for Democracy, Election and Good Governance (DEGG) and Environmental Studies papers should be 40% Separate passing for each Head - SEE, CIE and Practicals

# **Semester -V**

#### B. Sc. Part – III Semester -V COMPUTER SCIENCE DSC-IX: DSC03CSC51: Computer Network

Credits: 02 Theory: 30hrs.

**Course Outcomes:** After the completion of the course the student will be able to:

**CO1:** Explain the fundamental concepts of computer networks and network architectures.

**CO2:** Apply network communication techniques using various transmission media and devices.

**CO3:** Analyze different transport and application layer protocols for efficient network

**CO4:** Evaluate emerging network security technologies and modern networking trends.

#### **Unit -1: Introduction to Computer Networks:**

Components of Data Communication, Standards and Organizations (IEEE, IETF, ISO, ITU etc.)

**Network Classifications:** LAN, MAN, WAN, PAN, **Network Topologies:** Bus, Star, Ring, Mesh, Hybrid. Network Protocols and Layered Network Architecture, Overview of OSI Reference Model & TCP/IP Protocol Suite. Introduction to Network Security: Common Terms, Firewalls, Virtual Private Networks (VPN).

#### Unit -2: OSI Model and Network Devices:

**Physical Layer:** Cabling (Twisted Pair, Coaxial, Fiber Optic), Network Interface Cards (NIC), Transmission Media Devices: Repeater, Hub, Bridge, Switch, Router, Gateway.

**Data Link Layer:** Framing Techniques, Error Detection and Correction, Flow Control Protocols (Stop-and-Wait, Sliding Window), Shared Media Protocols: CSMA/CD, CSMA/CA.

**Network Layer:** Virtual Circuits & Datagram Approach, IP Addressing (IPv4 & IPv6), Subnetting, CIDR, Routing Algorithms (Adaptive & Non-Adaptive).

**Transport Layer:** Transport Services, TCP & UDP Protocols, Connection Establishment, Flow & Congestion Control.

Application Layer: Application Layer Protocols & Services: DNS (Domain Name System)

HTTP, HTTPS, WWW, Telnet, FTP, SMTP, POP3, IMAP, Introduction to Secure Communications (TLS, SSL).

#### Unit -3: Introduction to Network Security & Emerging Trends: (5 Lectures)

Overview of Cybersecurity & Cryptography, Firewalls & Intrusion Detection Systems (IDS/IPS), Wireless & Mobile Networks: Basics of Wi-Fi, Bluetooth, 5G Networks, Introduction to Cloud Networking & IoT Security, Introduction to Software-Defined Networking (SDN).

#### **Reference Books:**

1. "Data Communications and Networking" – Behrouz A. Forouzan (McGraw Hill)

- 2. "Computer Networks", A.S.Tanenbaum, Pearson Education, Fourth Edition.
- 3. "Fundamentals of Computer Networks" V.S. Bagad & I.A. Dhotre (Technical Publications)
- 4. "Internetworking with TCP/IP" Douglas E. Comer (Pearson)
- 5. "Data and Computer Communications" William Stallings (Pearson)

#### (15 Lectures)

(10 Lectures)

Marks-50

#### B. Sc. Part – III Semester -V COMPUTER SCIENCE DSC-X: DSC03CSC52: Software Engineering

Credits: 02 Theory: 30hrs. Marks-50

**Course Outcomes:** After the completion of the course the student will be able to:

**CO1:** Understand fundamental concepts of software engineering and system analysis.

**CO2:** Apply project management techniques and software quality assurance methods.

**CO3:** Analyze software deployment strategies and maintenance techniques.

**CO4:** Evaluate emerging trends in software development and their real-world applications.

#### Unit -1: Fundamentals of Software Engineering & System Analysis: (10 Lectures)

Introduction to Software Engineering: Definition, Importance, and Characteristics of Software, System Development Life Cycle (SDLC). Classical Models: Waterfall, Spiral, V-Model, Agile. Modern Software Development Trends (DevOps, CI/CD).

System Analysis & Requirements Engineering: Definition of a System, Elements & Characteristics, Types of Systems (Open, Closed, Static, Dynamic, etc.).

Requirement Analysis & Specification: Functional & Non-functional Requirements, Requirement Gathering Techniques (Interviews, Surveys, Prototyping), Software Design & Architecture. Design Approaches: Top-down, Bottom-up.

#### Unit -2: Software Project Management & Quality Assurance:(10 Lectures)

Project Planning & Scheduling (Gantt Chart, PERT, CPM), Estimation Techniques (COCOMO, Function Point Analysis), Risk Management in Software Projects. Version Control & Change Management (Git, GitHub).

Software Quality Assurance (SQA): Software Quality Factors (Reliability, Maintainability, Usability), Software Testing & Debugging: Testing Levels: Unit, Integration, System, Acceptance, Testing Types: White Box, Black Box, Regression, Security Testing. Manual vs. Automated Testing (Selenium, JUnit, PyTest), Alpha & Beta Testing.

#### Unit -3: Software Deployment, Maintenance & Case Studies

Change Over Strategies (Direct, Parallel, Phased, Pilot), Software Maintenance (Corrective, Adaptive, Perfective, Preventive), Reverse Engineering & Reengineering. Emerging Trends in Software Engineering: Agile Methodologies (Scrum, Kanban, XP), Cloud-Based Software Development, DevOps & Continuous Integration/Continuous Deployment (CI/CD), Security in Software Development (OWASP, Secure SDLC) Case Studies & Industry Applications: College Admission System, Library Management System, Bank Management System, E-commerce System (Amazon-like Platforms), Cloudbased Applications (Google Drive, Microsoft Azure, AWS).

#### **Reference Books:**

- 1. Roger S. Pressman & Bruce R. Maxim Software Engineering: A Practitioner's Approach
- 2. Software Engineering 9th Edition by Ian Sommerville
- 3. K.K. Aggarwal & Yogesh Singh Software Engineering
- 4. Pankaj Jalote Software Engineering: A Precise Approach
- 5. Dr. S. L. Gupta & Seema Gupta Software Engineering and Quality Assurance

#### B. Sc. Part – III Semester -V COMPUTER SCIENCE DSC-XI: DSC03CSC53: Internet Technologies-I Credits: 02 Theory: 30hrs. Marks-50

**Course Outcomes:** After the completion of the course the student will be able to:

**CO1:** Explain the core concepts and architecture of Flask as a micro-framework.

- **CO2:** Develop dynamic web applications using Jinja2 templating and frontend integration.
- CO3: Implement secure and interactive user form handling in Flask applications.

CO4: Design and deploy simple Flask-based web applications with user interaction.

#### Unit -1: Introduction to Flask Understanding Flask as a Micro Framework: (10 Lectures)

Characteristics of Flask: Lightweight, modular, flexible, and extensible. Who Uses Flask? Real-world applications and industry adoption. Setup Tools and pip: Installing Python, Installing Flask. Working with Virtual Environments (venv/virtualenv): Creating a new virtual environment, Activating and deactivating virtualenv, Adding and removing packages within virtualenv. Application Structure: Initialization of a Flask App, Routes and View Functions, Server Startup & Running Flask Applications. The Request-Response Cycle, Application and Request Contexts, Request Dispatching, Handling Responses in Flask, Command-Line Options with Flask-Script, Building the First Simple Flask Application.

#### Unit -2: Jinja2 Templating & Frontend Integration:

Introduction to the Jinja2 Template Engine, Rendering Templates in Flask, Jinja2 Fundamentals: Variables, Expressions, and Output, Comments and Escaping, Control Structures (Loops, Conditionals), Using Filters for Data Formatting, Template Inheritance & Includes: Modularizing templates for better maintainability, Using Twitter Bootstrap with Flask: Flask-Bootstrap for responsive UI, Custom Error Pages (404, 500, etc.), Managing Static Files in Flask: Handling CSS, JavaScript, and images, Linking Static Assets in Templates.

#### **Unit -3: Handling Forms & User Interaction:**

Cross-Site Request Forgery (CSRF) Protection in Forms, Creating & Managing Form Classes in Flask, HTML Rendering of Forms, Form Handling in View Functions: Retrieving user input

Processing form data, Redirects & User Sessions, Flask Flash Messages for User Notifications

Server-Side Form Validation: Ensuring valid user input, Custom validation methods, Creating Custom Fields & Validation Rules.

#### **Reference Books:**

- 1. "Flask Web Development" Ramesh B. Arunachalam (BPB Publications)
- 2. "Python Web Development with Flask" Parth Patel (BPB Publications)
- 3. "Flask Web Development: Developing Web Applications with Python" Miguel Grinberg (O'Reilly Media)
- 4. "Python and Flask for Web Development" Samuel Dauzon, Aidas Bendoraitis (Packt Publishing)
- 5. "Flask By Example" Gareth Dwyer (Packt Publishing)

#### 7 | Syllabus

#### (10 Lectures)

#### (10 Lectures)

ser interaction.

#### B. Sc. Part – III Semester -V COMPUTER SCIENCE

#### DSE-I: DSE03CSC51: Introduction to Java Programming Credits: 02 Theory: 30hrs. Marks-50

**Course Outcomes:** After the completion of the course the student will be able to:

**CO1:** Understand the fundamentals of Java programming and its comparison with other languages.

**CO2:** Apply object-oriented programming concepts to develop modular and reusable Java programs.

**CO3:** Analyze and handle exceptions efficiently to ensure robust application development. **CO4:** Design interactive Graphical User Interfaces (GUI) using AWT and Swing.

#### **Unit -1: Introduction to Java Programming Language:**

History of Java, Features of Java, Comparison of Java and C++, Java Environment, Java Tools – jdb, javap, javadoc, Java IDEs Structure of java program, First java program, Types of Comments, Data types, Variables, Operators, Keywords, Naming Convention, Declaring 1D, 2D array, Decision Making (if, switch),Looping(for, while), Type Casting, Accepting input using Command line argument, Accepting input from console.

#### Unit -2: Object Oriented Concepts in Java:

Defining Your Own Classes, Access Specifiers (public, protected, private, default), Array of Objects, Constructor, Overloading Constructors and use of 'this' Keyword, static block, static Fields and methods, Object class methods, String Class, Inner class, Packages, Wrapper Classes, Garbage Collection, Memory allocation for objects, Constructor, Implementation of Inheritance, use of super keyword, Implementation of Polymorphism, Method Overloading, Method Overriding, Nested and Inner classes, Use of final keyword related to method and class, abstract class and abstract methods, Defining and Implementing Interfaces, Object Cloning.

#### Unit -3: Exception Handling, GUI components using AWT and Swing: (12 Lectures)

Exception types, Using try catch and multiple catch, Nested try, throw, throws and finally, Creating User defined Exceptions, Assertions, Basics of AWT and Swing, their Difference, Layout Manager, Layouts, Components: JButton, JLabel, JText, JTextArea, JCheckBox and JRadioButton, JList, JComboBox, JMenu and JPopupMenu Class, JMenuItem and JCheckBoxMenuItem, JRadioButtonMenuItem, JScrollBar, Dialogs (Message, confirmation, input), JFileChooser, JColorChooser, Event Handling: Event sources, Listeners, Mouse and Keyboard Event Handling, Adapters.

#### **Reference Books:**

- 1. Core Java2 Volume I-Fundamentals by Cay S. Horstmann, Gary Cornell
- 2. Effective Java Programming Language Guide by Joshua Bloch
- 3. Java, The Complete Reference by Herbert Schildt
- 4. Java 2 programming black books by Steven Horlzner
- 5. Programming with Java, A primer by E. Balagurusamy

#### (10 Lectures)

# (8 Lectures)

#### B. Sc. Part – III Semester -V COMPUTER SCIENCE DSE-I: DSE03CSC52: PHP Programming Credits: 02 Theory: 30hrs.

**Course Outcomes:** After the completion of the course the student will be able to:

**CO1:** Understand the fundamentals of PHP and web development concepts.

- **CO2:** Develop dynamic web applications using PHP forms, session management, and file handling.
- **CO3:** Integrate PHP with MySQL for database-driven web applications.
- CO4: Implement advanced PHP programming techniques, including OOP and API development.

#### **Unit -1: Introduction to PHP & Web Development:**

Basics of Web Development: Client-Server Architecture, HTTP Protocol, Introduction to PHP: History, Features, Installation & Setup, PHP Syntax and Basic Programming. Constructs: Variables, Data Types, Constants, Operators and Expressions in PHP, Control Structures: Conditional Statements (if, else, switch), Loops (for, while, do-while, foreach), PHP Functions: User-Defined and Built-in Functions

#### Unit -2: Working with Forms, Sessions, and File Handling:

Handling Form Data: GET vs. POST Methods, Validating User Input and Sanitization Session Management: Cookies, Sessions, Authentication Mechanisms, File Handling in PHP: Reading, Writing, Uploading Files, Error Handling and Exception Handling in PHP, Introduction to Regular Expressions in PHP

#### Unit -3: Database Connectivity and Advanced PHP Concepts:

Introduction to MySQL and PHP Integration, Connecting to a MySQL Database using MySQLi and PDO, Performing CRUD Operations (Create, Read, Update, Delete), Introduction to AJAX with PHP, Introduction to Object-Oriented Programming (OOP) in PHP: Classes, Objects, Inheritance, Encapsulation, Basics of PHP Frameworks (e.g., Laravel Overview), Introduction to API Development in PHP.

#### **Reference Books:**

- 1. "PHP and MySQL Web Development" Luke Welling & Laura Thomson
- 2. "Learning PHP, MySQL & JavaScript" Robin Nixon
- 3. "Modern PHP: New Features and Good Practices" Josh Lockhart
- 4. "PHP: A Beginner's Guide" Vikram Vaswani
- 5. "PHP and MySQL Web Development" Narayana Rao & Kogent Learning Solutions

(10 Lectures)

(10 Lectures)

(10 Lectures)

Marks-50

#### B. Sc. Part – III Semester -V COMPUTER SCIENCE VSC-PR: VSC03CSC59: Introduction to Data Science using Python Credits: 02 Practical: 30hrs. Marks-25

#### **Course Outcomes:**

CO1: Understand fundamental concepts of data analyticsCO2: Perform data preprocessing and cleaning techniquesCO3: Analyze data using descriptive and exploratory techniquesCO4: Apply predictive analytics techniques

**Practical 1:** Understanding Populations and Samples **Objective:** Learn the difference between populations and samples and apply sampling techniques.

**Practical 2:** Working with Different Data Types **Objective:** Understand and categorize different types of data (numerical, categorical, ordinal, nominal).

**Practical 3:** Data Collection and Cleaning **Objective:** Collect data from various sources and clean it for analysis.

**Practical 4:** Calculating and Visualizing Measures of Central Tendency **Objective:** Compute and compare Mean, Median, and Mode for a given dataset and visualize distributions.

**Practical 5:** Data Analytics Life Cycle Implementation **Objective:** Understand different phases of data analytics life cycle through a case study.

**Practical 6:** Descriptive vs. Predictive Analytics **Objective:** Compare descriptive and predictive analytics using real data.

**Practical 7:** Data Pre-processing Techniques **Objective:** Implement data pre-processing techniques to improve data quality.

**Practical 8:** Exploratory Data Analysis (EDA) Basics **Objective:** Perform EDA to discover insights from data.

**Practical 9:** Data Visualization Techniques **Objective:** Use different visualization techniques to represent data insights.

**Practical 10:**Tools for Data Analytics **Objective:** Explore popular tools used in data analytics.

#### B. Sc. Part – III Semester -V COMPUTER SCIENCE DSC-PR-V: DSC03CSC59: Computer Science Lab-5

#### Credits: 06 Practical: 30hrs. Marks-75

#### Group A (Practical's based on Computer Network):

#### 1. Study of Different Types of Network Cables and Practically Implement the Cross-wired Cable and Straight-through Cable Using Clamping Tool

- (i) Construct one straight-through cable and one cross-wired cable.
- (ii) Test both cables with a LAN tester to ensure they are wired correctly.
- (iii) Check the functionality of these cables by connecting devices (PC to Switch or Router) and verify network communication.

#### 2. Study of Network Devices in Detail (Switch, Hub, Router, etc.)

- (i) Demonstrate the connection of a hub, switch, and router in a simple LAN setup.
- (ii) Test communication between different devices in the setup using ping and traceroute.

#### 3. Study of Network IP

- (i) Configure an IP address on a computer.
- (ii) Practice calculating subnets using a given IP address.
- (iii) Verify the network connectivity using ping and ipconfig commands.

#### 4. Connect the Computers in Local Area Network (File/Printer Sharing)

- (i) Set up the physical connection and ensure that each computer is in the same subnet.
- (ii) Test network connectivity between the computers by sending pings from one to another.
- (iii) Share files between the computers via Windows File Sharing or Samba.

#### 5. Study of Basic Network Commands and Network Configuration

- (i) Use ping to check connectivity between devices.
- (ii) Use ipconfig to verify IP configuration and identify potential network issues.
- (iii) Use tracert to trace the route of data packets from one machine to another.
- (iv) Use netstat to view active network connections.

#### **Group B** (Practical's based on Software Engineering): **1. Practical on SDLC Models & Requirement Gathering Techniques**

- (i) Prepare a requirement specification document outlining functional and non-functional requirements.
- (ii) Discuss and choose the appropriate SDLC model for the chosen system.

#### 2. Practical on Project Planning & Scheduling (Using Gantt Chart)

- (i) Prepare the project schedule using a Gantt chart that includes all milestones, task dependencies, and deadlines.
- (ii) Identify potential risks that may affect project completion and propose strategies to manage them.

#### 3. Practical on Software Testing (White Box & Black Box Testing)

- (i) Write a test report for both White Box and Black Box testing.
- (ii) Identify any defects or issues found during testing and suggest improvements.

#### 4. Practical on Version Control and GitHub

- (i) Use Git to track changes and handle conflicts between branches.
- (ii) Set up a GitHub repository and collaborate with peers (if possible) to complete the project.
- (iii) Submit a commit log showing all changes and contributions made throughout the project lifecycle.

#### 5. Practical on Software Maintenance

- (i) Prepare a maintenance report that includes the types of maintenance performed, the issues identified, and the solutions implemented.
- (ii) Discuss the maintenance challenges faced and the measures taken to ensure the system's stability and security.

#### Group C (Practical's based on Flask):

- 1. Create a web application to print a message 'Hi! This my First Flask Practical'
- 2. Create a web application to demonstrate dynamic URL.
- 3. Create a web application to demonstrate how int, string float variable value can be passed through dynamic URL.
- 4. Create a web application to pass integer, string and float values through dyanamic URL and display it in tabular format.
- 5. Create a web application to demonstrate how to use Jinja varialbes.
- 6. Create a web application to demonstrate Jinja if-elif-else and for statements.
- 7. Create a web application to demonstrate Jinja Filters.

#### Group D (Practical's based on Java Programming):

- 1. WAP to find the largest of n natural numbers.
- 2. WAP to find whether a given number is prime or not.
- 3. Write a menu driven program for following:
  - a. to display a Fibonacci series
  - b. to compute Factorial of a number
  - c. to check whether a given number is odd or even.
  - d. to check whether a given string is palindrome or not.
- 4. WAP to print the sum and product of digits of an Integer and reverse the Integer.
- 5. Write a program to create an array of 10 integers. Accept values from the user in that array. Input another number from the user and find out how many numbers are equal to the number passed, how many are greater and how many are less than the number passed.
- 6. Write a program that will prompt the user for a list of 5 prices. Compute the average of the prices and find out all the prices that are higher than the calculated average.
- 7. Write a program in java to input N numbers in an array and print out the Armstrong numbers from the set.
- 8. Write java program for the following matrix operations:
  - a. Addition of two matrices
  - b. Summation of two matrices
  - c. Transpose of a matrix
  - d. Input the elements of matrices from user.
- 9. Write a java program that computes the area of a circle, rectangle and a Cylinder using function overloading.

- 10. Write a Java for the implementation of Multiple inheritance using interfaces to calculate the area of a rectangle and triangle.
- 11. Write a java program to create a frame window in an Applet. Display your name, address and qualification in the frame window.
- 12. Write a program that reads two integer numbers for the variables a and b. If any other character except number (0-9) is entered then the error is caught by NumberFormatException object. After that ex.getMessage() prints the information about the error occurring causes.
- 13. Write a program for the following string operations:
  - a. Compare two strings
  - b. Concatenate two strings
  - c. Compute length of a string
- 14. Create a class called Fraction that can be used to represent the ratio of two integers. Include appropriate constructors and methods. If the denominator becomes zero, throw and handle an exception

#### B. Sc. Part – III Semester -V COMPUTER SCIENCE MIN-IX: MIN03CSC51: Software Engineering-I Credits: 02 Theory: 30hrs. Marks-50

**Course Outcomes:** After the completion of the course the student will be able to:

**CO1:** Understand the fundamental concepts of software engineering and system analysis.

**CO2:** Apply project management and software quality assurance techniques.

**CO3:** Analyze software testing and debugging techniques to ensure software quality.

**CO4:** Evaluate deployment strategies and emerging trends in software engineering.

# Unit -1: Fundamentals of Software Engineering & System Analysis: (10 Lectures) Introduction to Software Engineering: Definition, Importance, and Characteristics of Software, System Development Life Cycle (SDLC). Classical Models: Waterfall, Spiral, V-Model, Agile. Modern Software Development Trends (DevOps, CI/CD). System Analysis & Requirements Engineering: Definition of a System, Elements & Characteristics, Types of Systems (Open, Closed, Static, Dynamic, etc.). Requirement Analysis & Specification: Functional & Non-functional Requirements, Requirement Gathering Techniques (Interviews, Surveys, Prototyping), Software Design & Architecture. Design Approaches: Top-down, Bottom-up.

# Unit -2: Software Project Management & Quality Assurance: (10 Lectures)

Project Planning & Scheduling (Gantt Chart, PERT, CPM), Estimation Techniques (COCOMO, Function Point Analysis), Risk Management in Software Projects. Version Control & Change Management (Git, GitHub).

Software Quality Assurance (SQA): Software Quality Factors (Reliability, Maintainability, Usability), Software Testing & Debugging: Testing Levels: Unit, Integration, System, Acceptance, Testing Types: White Box, Black Box, Regression, Security Testing. Manual vs. Automated Testing (Selenium, JUnit, PyTest), Alpha & Beta Testing.

#### Unit -3: Software Deployment, Maintenance & Case Studies (10 Lectures)

Change Over Strategies (Direct, Parallel, Phased, Pilot), Software Maintenance (Corrective, Adaptive, Perfective, Preventive), Reverse Engineering & Reengineering. Emerging Trends in Software Engineering: Agile Methodologies (Scrum, Kanban, XP), Cloud-Based Software Development, DevOps & Continuous Integration/Continuous Deployment (CI/CD), Security in Software Development (OWASP, Secure SDLC) Case Studies & Industry Applications: College Admission System, Library Management System, Bank Management System, E-commerce System (Amazon-like Platforms), Cloudbased Applications (Google Drive, Microsoft Azure, AWS).

#### **Reference Books:**

- 1. Roger S. Pressman & Bruce R. Maxim Software Engineering: A Practitioner's Approach
- 2. Software Engineering 9th Edition by Ian Sommerville
- 3. K.K. Aggarwal & Yogesh Singh Software Engineering
- 4. Pankaj Jalote Software Engineering: A Precise Approach
- 5. Dr. S. L. Gupta & Seema Gupta Software Engineering and Quality Assurance

#### B. Sc. Part – III Semester -V COMPUTER SCIENCE MIN-PR-V: MIN03CSC59: Software Engineering Lab-I Credits: 02 Practical: 30hrs. Marks-25

#### 1. Practical on SDLC Models & Requirement Gathering Techniques

- (iii) Prepare a requirement specification document outlining functional and non-functional requirements.
- (iv) Discuss and choose the appropriate SDLC model for the chosen system.

#### 2. Practical on Project Planning & Scheduling (Using Gantt Chart)

- (iii)Prepare the project schedule using a Gantt chart that includes all milestones, task dependencies, and deadlines.
- (iv) Identify potential risks that may affect project completion and propose strategies to manage them.

#### 3. Practical on Software Testing (White Box & Black Box Testing)

(iii) Write a test report for both White Box and Black Box testing.

(iv) Identify any defects or issues found during testing and suggest improvements.

#### 4. Practical on Version Control and GitHub

- (iv) Use Git to track changes and handle conflicts between branches.
- (v) Set up a GitHub repository and collaborate with peers (if possible) to complete the project.
- (vi) Submit a commit log showing all changes and contributions made throughout the project lifecycle.

#### **5. Practical on Software Maintenance**

- (iii) Prepare a maintenance report that includes the types of maintenance performed, the issues identified, and the solutions implemented.
- (iv) Discuss the maintenance challenges faced and the measures taken to ensure the system's stability and security.

# **Semester -VI**

#### B. Sc. Part – III Semester -VI COMPUTER SCIENCE DSC-XIII: DSC03CSC61: Advanced Computer Network Credits: 02 Theory: 30hrs. Marks-50

# Course Outcomes: After the completion of the course the student will be able to: CO1: Understand Advanced Networking and Security Concepts. CO2: Manage Internet Services and Secure Network Communications. CO3: Implement Intranet and Cloud Networking Services. CO4: Apply Authentication and Directory Services in Network Security. Unit -1: Advanced Networking & Security: TCP/IP for System Administrators, Basic Network Configurations (IP, Subnet, Routing)

CP/IP for System Administrators, Basic Network Configurations (IP, Subnet, Routing) Linux Firewall (Netfilter, iptables, firewall), VPN & Secure Network Access. Network Authentication: Kerberos, OpenLDAP, System and Network Security Essentials.

#### Unit -2: Internet Services & Protocols:

Domain Name System (DNS) – Structure, Configuration, Web Hosting & Apache Web Server Configuration. Email Services: SMTP, POP, IMAP, Security in Email, Secure Communication: SSH (Secure Shell), TLS, SSL. Network File Transfer Services: FTP, SFTP, NFS. Authentication & Directory Services: OpenLDAP Server, Samba, LDAP.

#### Unit -3: Intranet Services & Cloud Networking:

Network File Systems (NFS, DFS, Samba), Dynamic Host Configuration Protocol (DHCP) Database Servers & Web Technologies: MySQL, LAMP Applications, Virtualization & Cloud Computing Basics, Introduction to Container Networking (Docker, Kubernetes) Introduction to Network Automation & Monitoring Tools (Wireshark, Nagios, Zabbix)

#### **Reference Books:**

- 1. "Linux Administration: A Beginner's Guide" Wale Soyinka (McGraw Hill)
- 2. "Computer Networks", A.S.Tanenbaum, Pearson Education, Fourth Edition.
- 3. "Computer Networks: Principles, Technologies, and Protocols for Network Design" Natalia Olifer & Victor Olifer (Pearson)
- 4. "TCP/IP Illustrated, Volume 1: The Protocols" W. Richard Stevens (Addison-Wesley)
- 5. "Linux Firewalls: Attack Detection and Response" Michael Rash (No Starch Press)

(10 Lectures)

#### B. Sc. Part – III Semester -VI COMPUTER SCIENCE DSC-XIV: DSC03CSC62: Object Oriented Software Engineering Credits: 02 Theory: 30hrs. Marks-50

**Course Outcomes:** After the completion of the course the student will be able to:

**CO1:** Understand Object-Oriented Principles & UML Modeling.

**CO2:** Apply Object-Oriented Analysis & Design Techniques.

**CO3:** Implement System Design using OOAD.

**CO4:** Integrate Object-Oriented Models with Database Systems.

Unit -1: Fundamentals of Object-Oriented Analysis & UML: (10 Lectures) Introduction to Object-Oriented Concepts: Class & Object, Encapsulation, Inheritance, and Polymorphism, Aggregation and Composition. Introduction to Object-Oriented Analysis & Design (OOAD), Need for OOAD, Object-Oriented vs. Traditional Software Development, Advantages of Object-Oriented Modeling, Unified Modeling Language (UML) Basics, Overview of UML & Conceptual Model, UML Architecture and Diagrams, UML Notations and Elements.

#### Unit -2: Object-Oriented Modeling & Design Patterns:

Unified Process & UML Diagrams: Unified Process Model & Phases (Inception, Elaboration, Construction, Transition), UML Structural Diagrams: Class Diagram, Object Diagram, Package Diagram, Component Diagram, Deployment Diagram. UML Behavioral Diagrams: Use Case Diagram, Activity Diagram, Sequence Diagram, Statechart Diagram, Advanced Object-Oriented Design Concepts. System Design Process & Model Partitioning: Concurrency & Subsystem Allocation, Task, Data, and Resource Management, Design Patterns & Best Practices.

#### Unit -3: Object-Oriented Implementation:

Mapping Object Model to Database Schema: Object-Relational Mapping (ORM), Relational vs. NoSQL Database Design for OO Applications, Popular ORM Tools (Hibernate, Django ORM, SQLAlchemy), Object-Oriented Software Testing: Testing Strategies for OO Systems Unit Testing, Integration Testing, System Testing, Object-Oriented Test Case Design, Emerging Trends in OO Software Development. Agile & DevOps in OOAD, Microservices and Distributed Object-Oriented Systems, Cloud-Native Object-Oriented Applications, AI and Automation in OO Software Engineering.

#### **Reference Books:**

- 1. "Object-Oriented Modeling and Design" Rambaugh, Michael Blaha, William Premerlani, Frederick Eddy, William Lorensen
- 2. "Software Engineering" Sommerville & Pankaj Jalote
- 3. "Object-Oriented Programming with C++" E. Balagurusamy
- 4. "Applying UML and Patterns" Craig Larman
- 5. "The Unified Modeling Language User Guide" Grady Booch, James Rumbaugh, Ivar Jacobson

(10 Lectures)

#### B. Sc. Part – III Semester -VI COMPUTER SCIENCE DSC-XV: DSC03CSC63: Internet Technologies-II Credits: 02 Theory: 30hrs. Marks-50

**Course Outcomes:** After the completion of the course the student will be able to:

**CO1:** Integrate and Manage Databases in Flask Applications.

**CO2:** Implement User Authentication and Security Measures.

**CO3:** Utilize Flask for Secure Web Application Development.

**CO4:** Deploy and maintain Flask Applications in Production Environments.

#### Unit -1: Database Integration & Management:

Overview of SQL & MySQL Databases. Introduction to Python Database Frameworks Working with Flask-MySQL: Database Configuration in Flask, Connecting Flask with MySQL, Model Definition & Relationships, Performing Database Operations: Creating Tables, Inserting, Updating, and Deleting Records, Querying Data using Flask, Using Databases in View Functions, Interacting with Databases from the Python Shell.

#### Unit -2: User Authentication & Security:

Authentication Extensions for Flask, Building an Authentication System: Creating an Authentication Blueprint, User Authentication using Flask-Login, Handling User Logins & Sessions: Preparing the User Model for Logins, Protecting Routes & Restricting Access, Adding Login and Logout Functionality, Implementing User Registration: New User Registration Flow, Adding a Registration Form, Storing & Validating User Data.

#### **Unit -3: Application Deployment & Production:**

Understanding Deployment Workflow, Logging & Debugging Errors in Production, Cloud Deployment Strategies: Deploying Flask on Cloud Platforms (Heroku, PythonAnywhere), Deploying Flask Apps with Hosting Services, Handling Production vs Development Environments, Deploying Application Updates & Maintenance.

#### **Reference Books:**

- 1. "Flask Web Development" Ramesh B. Arunachalam (BPB Publications)
- 2. "Python Web Development with Flask" Parth Patel (BPB Publications)
- 3. "Flask Web Development: Developing Web Applications with Python" Miguel Grinberg (O'Reilly Media)
- 4. "Python and Flask for Web Development" Samuel Dauzon, Aidas Bendoraitis (Packt Publishing)
- 5. "Flask By Example" Gareth Dwyer (Packt Publishing)

#### (10 Lectures)

#### (10 Lectures)

#### **B. Sc. Part – III Semester - VI COMPUTER SCIENCE**

DSE-II: DSE03CSC61: Advanced Java Programming Theory: 30hrs. Credits: 02 Marks-50

**Course Outcomes:** After the completion of the course the student will be able to:

**CO1:** Understand the concepts of multithreading and networking in Java.

**CO2:** Apply Java Database Connectivity (JDBC) for database operations.

CO3: Develop dynamic web applications using Java Server Pages (JSP) and servlets.

**CO4:** Apply Java web technologies to real-world applications and projects.

#### **Unit -1: Multithreding and Collection in Java**

Introduction to Multithreading, Creating and Managing Threads, Thread Priorities, Thread Synchronization and Inter-thread Communication, Java Concurrency Framework, Collection Overview, Collection Interfaces, Collection classes, User defined classes in collection, Comparators, Legacy classes and interfaces.

#### Unit -2: Java Database Connectivity (JDBC)

Overview of JDBC architecture, JDBC drivers and database connections, Executing SQL Queries with JDBC, CRUD operations with JDBC, Batch processing and transaction management, Connection Pooling and Data Source, Implementing connection pooling, Using Data Source for efficient database connections, Advanced JDBC Features, Callable Statements and stored procedures, Result Set types

#### Unit -3: Java Server Pages (JSP) and Java Servlets

Introduction to JSP, JSP Life Cycle, JSP Elements, JSP Implicit Objects, JSP Form Handling and Request Processing, JSP Session Management, JSP Exception Handling, JSP Expression Language (EL), JSP Standard Tag Library (JSTL), JSP Database Connectivity (JDBC), Introduction to Servlets, Setting Up the Servlet Environment, Servlet API and Interfaces, Handling Requests and Responses, Servlet Session Management, Servlet Communication, Servlet Filters and Listeners, JDBC Connectivity in Servlets

#### **Reference Books:**

- 1. "Java: The Complete Reference" Herbert Schildt
- 2. "Core Java Volume I & II" Cay S. Horstmann
- 3. "Head First Servlets and JSP" Bryan Basham, Kathy Sierra, Bert Bates
- 4. "Java Concurrency in Practice" Brian Goetz
- 5. "Java Network Programming" Elliotte Rusty Harold

#### (10 Lectures)

#### (10 Lectures)

#### **B. Sc. Part – III Semester - VI COMPUTER SCIENCE** DSE-II: DSE03CSC62: Advanced PHP Programming Credits: 02 Theory: 30hrs.

**Course Outcomes:** After the completion of the course the student will be able to:

**CO1:** Develop and integrate web services using PHP.

**CO2:** Utilize PHP frameworks to build scalable web applications.

**CO3:** Implement secure and efficient PHP coding practices.

CO4: Design and implement software using PHP design patterns.

#### Unit -1: PHP and Web Services:

Introduction to Web Services and APIs, Consuming and Creating RESTful APIs with PHP, JSON and XML Data Handling in PHP, Authentication and Security in Web Services (JWT, OAuth), cURL and HTTP Requests in PHP, WebSocket Programming in PHP for Real-Time Applications.

#### **Unit -2: PHP Frameworks and Performance Optimization:**

Introduction to PHP Frameworks (Laravel, CodeIgniter, Symfony), MVC Architecture in PHP Frameworks, Routing, Middleware, and Templating Engines, Database Handling with ORM (Eloquent, Doctrine), Performance Optimization in PHP: Caching, Profiling, and Code Optimization Techniques, Deployment and Hosting of PHP Applications (Apache, Nginx, Docker, Cloud Platforms), Secure Coding Practices in PHP (SQL Injection, XSS, CSRF Prevention)

#### **Unit -3: PHP Design Patterns:**

Introduction to Design Patterns in PHP, Understanding Design Patterns: Importance and Benefits, Categories of Design Patterns: Creational, Structural, Behavioral, Implementing Design Patterns in PHP, Creational Design Patterns, Singleton Pattern, Factory Pattern, Abstract Factory Pattern, Builder Pattern, Prototype Pattern, Structural Design Patterns, Adapter Pattern, Decorator Pattern, Composite Pattern, Facade Pattern, Proxy Pattern, Behavioral Design Patterns, Observer Pattern, Strategy Pattern, Command Pattern, Template Method Pattern, Dependency Injection Pattern, Practical Implementation and Use Cases, Applying Design Patterns in Real-World PHP Applications, Using Design Patterns in PHP Frameworks (Laravel, Symfony), Best Practices and Performance Considerations.

#### **Reference Books:**

- 1. "PHP and MySQL Web Development" Luke Welling & Laura Thomson
- 2. "Learning PHP, MySQL & JavaScript" Robin Nixon
- 3. "Modern PHP: New Features and Good Practices" Josh Lockhart
- 4. "PHP: A Beginner's Guide" Vikram Vaswani
- 5. "PHP and MySQL Web Development" Narayana Rao & Kogent Learning Solutions

#### (10 Lectures)

(10 Lectures)

(10 Lectures)

Marks-50

### **B. Sc. Part – III Semester -VI COMPUTER SCIENCE**

#### VSC-PR-VI: VSC03CSC69: Machine Learning for Data Science Credits: 02 Practical: 30hrs. Marks-25

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

CO1: Understand fundamental concepts of Machine Learning and its various types.

- CO2: Apply Machine Learning models for regression and classification tasks.
- **CO3:** Analyze clustering techniques and optimize model performance.

CO4: Evaluate and optimize Machine Learning models for real-world applications.

#### **Practical Assignments**

1)Consider the given table Student as numpy array and solve the following questions.

Roll No	Sub1	Sub2	Sub3
1	80	76	81
2	70	87	82
3	90	85	93
4	89	90	97
5	79	91	79

- 1. Find out average marks of Sub1, Sub2 and Sub3 and display subject with maximum average marks.
- 2. Display student roll numbers who has scored maximum marks in subject1.
- 3. Display rolls numbers of student who has scored minimum marks in Sub2.
- 4. Display roll numbers of student who has maximum average marks.
- 5. Display counts of the student those who have scored marks more than average in Sub3.
- 2) Download Titanic dataset (titanic.csv) from the UCI repository/Kaggle. Solve following questions to analyse the data using pandas.
- 1. Display the male female count of the passengers.
- 2. Display the Survived and Not Survived passengers count.
- 3. Display class wise count of survived passengers.
- 4. Display gender wise survival percentage of passengers.
- 5. Display class wise count of survived Female passengers.
- 6. Display class wise Fare of passengers.
- 7. Display class wise count of passenger travelling without paying fare.

3) Consider the following dataset where Experience (in Years) and Salary (in Lacks) given. Implement

the Simple Linear Regression machine learning algorithm to predict the salary for given experience value 10. Visualize the straight fit line with original and predicted data distribution. Display intercept, slope and model score.

Experience	1	1	2	3	4	4	5	6	6	7
Salary	2.1	2.5	3.1	3	3.8	3.2	4.3	3.9	4.4	4.8

4) Consider the following dataset which consists Aptitude, Communication marks and Class (Speaker, Leader, Intel) values for 15 students. Use KNN classification algorithm to classify given values of Aptitude (2) and Communication (5) marks to classes Speaker, Leader and Intel with K's values 1,2,3

and similarity measure using Euclidean Distance.

Apti.	2	2	7	7	8	4	5	3	8	6	6	6	6	9	5
Comm.	5	6	6	2.5	6	7	3	5.5	3	5.5	4	7	2	7	4.5
Class	SK	SK	LD	IT	LD	SK	IT	SK	IT	LD	IT	LD	IT	LD	IT

5) Consider the following dataset and implement the K-Means Clustering machine learning algorithm to find out number of clusters. Use rule  $\sqrt{n/2}$  to consider the value of K-clusters.

Х	1	2	2	3	4	5
Y	1	1	3	2	3	5

#### B. Sc. Part – III Semester -VI COMPUTER SCIENCE DSC-PR-VI: DSC03CSC69: Computer Science Lab-6 Credits: 06 Practical: 30hrs. Marks-75

#### Group A (Practical's based on Advanced Computer Network):

#### 1. Configure a Network Topology Using Packet Tracer Software

- (i) Design a simple network with at least two routers, two switches, and four computers.
- (ii) Configure static IP addresses on each machine and enable routing on routers.
- (iii) Test network connectivity using the ping command.
- (iv) Demonstrate the routing and switching behavior in Packet Tracer.

#### 2. Study of DNS and Domain Name Resolution

- (i) Use the nslookup command to query DNS records.
- (ii) Configure a DNS server in Packet Tracer or on a real machine.
- (iii)Perform DNS lookups for common websites (e.g., nslookup google.com).

#### 3. Wi-Fi Network Setup and Security Configuration

- (i) Configure a Wi-Fi router with an appropriate SSID, security protocol, and password.
- (ii) Test connectivity by connecting wireless devices to the network.
- (iii) Set up MAC address filtering and other security measures on the router.
- (iv) Test the network performance using speed test tools and troubleshoot connection issues.

#### 4. Wi-Fi Network Setup and Security Configuration

- (i) Configure a Wi-Fi router with an appropriate SSID, security protocol, and password.
- (ii) Test connectivity by connecting wireless devices to the network.
- (iii) Set up MAC address filtering and other security measures on the router.
- (iv) Test the network performance using speed test tools and troubleshoot connection issues.

#### Group B (Practical's based on Object Oriented Software Engineering):

- 1. To study and draw various UML diagrams.
- 2. To illustrate the use of class diagrams.
- 3. To draw an activity diagram and use case diagram for ATM and Library Management System.
- 4. Draw Object Diagram for ATM System.
- 5. Development of State Transition Diagram.
- 6. Draw ER Diagram for Hospital Management System

#### Group C (Practical's based on Internet Technology-II):

- 1. Create a web application to demonstrate how MySQL database is connected to Flask.
- 2. Create a web application with CRUD operations.
- 3. Create a web application to demonstrate Session and Cookies in Flask.
- 4. Create a web application to demonstrate use of multiple tables from MySQL database.
- 5. Create a web application to demonstrate report preparation using FPDF.
- 6. Practical to demonstrate website hosting.

#### Group D (Practical's based on Advanced Java Programming):

- 1. Java program to demonstrate the Multithreding Concepts
- 2. Java program to demonstrate the JDBC CRUD Operations with MySQL
- 3. Create a simple JSP page with expressions, declarations, and actions.
- 4. Java program to design User Login System using JSP
- 5. Java program to design Servlet-Based Online Form Processing

#### B. Sc. Part – III Semester -VI COMPUTER SCIENCE

#### MIN-X: MIN03CSC61: Software Engineering-II Theory: 30hrs.

Credits: 02

**Course Outcomes:** After the completion of the course the student will be able to:

**CO1:** Understand Object-Oriented Principles & UML Modeling.

CO2: Apply Object-Oriented Analysis & Design Techniques.

**CO3:** Implement System Design using OOAD & Design Patterns.

**CO4:** Integrate Object-Oriented Models with Database Systems.

#### Unit -1: Fundamentals of Object-Oriented Analysis & UML:

Introduction to Object-Oriented Concepts: Class & Object, Encapsulation, Inheritance, and Polymorphism, Aggregation and Composition. Introduction to Object-Oriented Analysis & Design (OOAD), Need for OOAD, Object-Oriented vs. Traditional Software Development, Advantages of Object-Oriented Modeling, Unified Modeling Language (UML) Basics, Overview of UML & Conceptual Model, UML Architecture and Diagrams, UML Notations and Elements.

#### **Unit -2: Object-Oriented Modeling & Design Patterns:**

Unified Process & UML Diagrams: Unified Process Model & Phases (Inception, Elaboration, Construction, Transition), UML Structural Diagrams: Class Diagram, Object Diagram, Package Diagram, Component Diagram, Deployment Diagram. UML Behavioral Diagrams: Use Case Diagram, Activity Diagram, Sequence Diagram, Statechart Diagram, Advanced Object-Oriented Design Concepts. System Design Process & Model Partitioning: Concurrency & Subsystem Allocation, Task, Data, and Resource Management, Design Patterns & Best Practices.

#### **Unit -3: Object-Oriented Implementation:**

Mapping Object Model to Database Schema: Object-Relational Mapping (ORM), Relational vs. NoSQL Database Design for OO Applications, Popular ORM Tools (Hibernate, Django ORM, SQLAlchemy), Object-Oriented Software Testing: Testing Strategies for OO Systems Unit Testing, Integration Testing, System Testing, Object-Oriented Test Case Design, Emerging Trends in OO Software Development. Agile & DevOps in OOAD, Microservices and Distributed Object-Oriented Systems, Cloud-Native Object-Oriented Applications, AI and Automation in OO Software Engineering.

#### **Reference Books:**

- 1. "Object-Oriented Modeling and Design" Rambaugh, Michael Blaha, William Premerlani, Frederick Eddy, William Lorensen
- 2. "Software Engineering" Sommerville & Pankaj Jalote
- 3. "Object-Oriented Programming with C++" E. Balagurusamy
- 4. "Applying UML and Patterns" Craig Larman
- 5. "The Unified Modeling Language User Guide" Grady Booch, James Rumbaugh, Ivar Jacobson

Marks-50

#### (10 Lectures)

#### (10 Lectures)

#### B. Sc. Part – III Semester -VI COMPUTER SCIENCE MIN-PR-VI: MIN03CSC69: Software Engineering Lab-II Credits: 02 Practical: 30hrs. Marks-25

- 1. To study and draw various UML diagrams.
- 2. To illustrate the use of class diagrams.
- 3. To draw an activity diagram and use case diagram for ATM and Library Management System.
- 4. Draw Object Diagram for ATM System.
- 5. Development of State Transition Diagram.
- 6. Draw ER Diagram for Hospital Management System

### **Question Paper Format:**

Seat No.

Day:

Ques. Paper Code

[8]

#### **VIVEKANAND COLLEGE, KOLHAPUR**

(AN EMPOWERED AUTONOMOUS INSTITUTE) B.Sc. Part- III (Computer Science) (Semester-V) Examination..... Course Code and Name: DSC03CSC51: Computer Network Time: 1.30 hours Date: --/--/ Marks : 40

#### **Instructions:**

1) All the questions are compulsory.

2) Figures to the right indicate full marks.

3) Draw neat labeled diagrams wherever necessary.

4) Use of log table/calculator is allowed.

#### **Q. 1. Select correct alternative (One mark each):**

i)	Xyzabcdefghijklmno	р		
ii)	a) Xyzabcdefghijklmno	b) p	c)	d)
iii)	a) Xyzabcdefghijklmno	b) p	c)	d)
iv)	a) Xyzabcdefghijklmno	b) p	c)	d)
v)	a) Xyzabcdefghijklmno	b) p	c)	d)
vi)	a) Xyzabcdefghijklmno	b) p	c)	d)
vii)	a) Xyzabcdefghijklmno	b) p	c)	d)
viii)	a) Xyzabcdefghijklmno	b) p	c)	d)
	a)	b)	c)	d)

#### Q.2. Attempt any TWO (Eight marks each):

- i) Xyzabcdefghijklmnop.
- ii) Xyzabcdefghijklmnop.
- iii) Xyzabcdefghijklmnop.

[16]

#### Q.3. Attempt any FOUR (Four marks each):

- i) Xyzabcdefghijklmnop.
- ii) Xyzabcdefghijklmnop.
- iii) Xyzabcdefghijklmnop.
- iv) Xyzabcdefghijklmnop.
- v) Xyzabcdefghijklmnop.
- vi) Xyzabcdefghijklmnop.

.....

[16]