SHIVAJI UNIVERSITY, KOLHAPUR

NAAC A++ Grade with CGPA 3.52

As per

National Education Policy (NEP-2020)

Syllabus for

Bachelor of Science (B.Sc.)

Computer Science (Optional)

(Under Faculty of Science and Technology)

PART-III SEMESTER V & VI

(Syllabus to be implemented from Academic Year 2026-27)

B.Sc. (Computer Science) Part III Semester V & VI

(NEP-2020)

Syllabus to be implemented from Academic year 2026-27

Semester V

B.Sc. (Computer Science) Part - III SEMESTER-V (Duration- Six Month)										
Sr. No.	Course Code	Teaching Scheme			Examination Scheme					
		Theory and Practical			University Assessment (UA)			Internal Assessment (IA)		
		Lectures Per week	Practical Per week	Credit	Maximum Marks	Minimum Marks	Exam minute	Maximum Marks	Minimum Marks	Exam Hours
1	Major IX: Web Development using Python	2	-	2	40	16	90	10	04	20
2	Major X: Artificial Intelligence & Prompt Engineering	2	-	2	40	16	90	10	04	20
3	Major Practical V: Practical based on Major IX and X	-	8	4	80	32	180	20	08	40
4	Major Elective I ✓ Core Java ✓ C#.NET	2	-	2	40	16	90	10	04	20
5	Major Practical I ✓ Core Java Lab ✓ C#.NET Lab	-	4	2	40	16	90	10	04	20
6	Open Elective V: From OE Basket	2	-	2	40	16	90	10	04	20
7	VSC II (Major Specific) Node.js	-	4	2	40	16	90	10	04	20
8	AEC III: Corporate Culture & Communication	2	-	2	40	16	90	10	04	20
9	OJT: On Job Training	-	-	4	80	32	180	20	08	40
	Total (A)			22	440			110		

B.Sc. (Computer Science) Part - III SEMESTER-VI (Duration- Six Month)										
Sr. No.	Course Code	Teaching Scheme			Examination Scheme					
		Theory and Practical			University Assessment (UA)			Internal Assessment (IA)		
		Lectures Per week	Practical Per week	Credit	Maximum Marks	Minimum Marks	Exam minute	Maximum Marks	Minimum Marks	Exam Hours
1	Major XI: Data Analytics with Python	2	-	2	40	16	90	10	04	20
2	Major XII: Big Data Management	2	-	2	40	16	90	10	04	20
3	Major Practical VI: Practical based on Major XI and XII	-	8	4	80	32	180	20	08	40
4	Major Elective II ✓ Advanced Java ✓ ASP.NET	2	-	2	40	16	90	10	04	20
5	Major Practical II ✓ Advanced Java Lab ✓ ASP.NET Lab	-	4	2	40	16	90	10	04	20
6	VSC III (Major Specific) Linux Utilities and Shell Programming	-	4	2	40	16	90	10	04	20
7	SEC III Software Engineering	2	-	2	40	16	90	10	04	20
8	AEC IV: Professional Ethics	2	-	2	40	16	90	10	04	20
9	IKS: IKS for Computer Science	2	-	2	40	16	90	10	04	20
10	FP: Field Project	-	-	2	40	16	90	10	04	20
	Total (B)			22	440			110		

Exit option after Level 6: Students can exit with Bachelor of Science in Computer Science

B.Sc. Computer Science [Optional] (Semester-V) (NEP2.0)

Multiple Entry and Multiple Exit Option (NEP-2020)

PART III SEM V (Level- 5.5)

Title of course: **Web Development using Python**

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

CO1. Understand Python programming fundamentals for web applications
CO2. Design dynamic web pages using Django templates
CO3. Develop database-driven web applications using Django ORM
CO4. Implement user authentication and session management

Unit-I : Python Fundamentals for Web Development (15 Hours)

Introduction to Python- Features and applications of Python for web development, Installing Python and setting up development environment, Python Basics - Python syntax, Variables and data types, Operators and expressions, Control statements (if, for, while), Functions and Modules - Defining functions, Function parameters and return values, Importing and using modules Data Structures- Lists and list operations, Tuples, Dictionaries and dictionary methods, Sets Exception Handling- Try-except blocks, Handling common exceptions, Custom exceptions File Handling - Reading and writing files, Working with different file formats, Virtual Environment- Introduction to virtual environments, Creating and activating virtual environments, Managing dependencies with pip

Unit-II: Django Framework Essentials (15 Hours)

Getting Started with Django - Introduction to Django framework, MVT (Model-View-Template) architecture, Installing Django, Creating Django project and app, Understanding Django project structure, Django settings and configuration, Views and URL Routing: Running Django development server, URL mapping and views, HTTP request and response handling, Django Templates - Django Template Language (DTL), Template syntax and variables, Template inheritance and extending templates, Working with static files (CSS, JavaScript, images) Forms in Django - Creating Django forms, Form validation, Handling GET and POST requests, Processing form data, Models and Database -Introduction to Django ORM, creating models and defining fields,Database configuration (SQLite/MySQL), Migrations (makemigrations and migrate), CRUD operations using ORM, QuerySets and filtering data, Django Admin Interface - Registering models with admin, customizing admin interface, Managing data through admin panel, User Authentication - User authentication and authorization, Implementing login and logout functionality, User registration, Sessions and cookies basics, File Handling in Django - File upload functionality, Serving uploaded files

Reference Books:

1. Antonio Melé, "Django 3 By Example", Packt Publishing, 3rd Edition, 2020
2. William S. Vincent, "Django for Beginners", Welcome to Code, 2022
3. Mark Lutz, "Learning Python", O'Reilly Media, 5th Edition, 2013
4. Gowrishankar S. and Veena A., "Introduction to Python Programming", CRC Press, 1st Edition, 2018, (ISBN 978-0815394372)
5. Dr. Krishna Kumar Mohbey and Dr. Brijesh Bakariya, "An Introduction to Python Programming: A Practical Approach", BPB Publications, 1st Edition, 2021 (ISBN 9789391392062)
6. Dr. R. Nageswara Rao,Dreamtech Press / Wiley India, 3rd Edition: 2020(ISBN: 9789390457151)

Web Sites:

1. Django Official Documentation: https://docs.djangoproject.com/
2. Python Official Documentation: https://docs.python.org/

B.Sc. Computer Science [Optional] (Semester-V) (NEP2.0) (Leel–5.0)

Multiple Entry and Multiple Exit Option (NEP -2.0)

PART III SEM V (Level 5.5)

Title of the Course: Artificial Intelligence & Prompt Engineering

Course Outcomes: After completion of this course, students should be able to:

CO1: Understand core AI concepts, Fundamental AI concepts, intelligent agents, and the scope and applications of Artificial Intelligence.

CO2: Apply classical, heuristic, local, and online search algorithms to solve problems in deterministic, non-deterministic, and unknown environments.

CO3: Design clear, concise, and effective prompts following standard prompt engineering principles and understand the capabilities and limitations of Large Language Models (LLMs)

CO4: Utilize LLMs for text and image generation in various applications and evaluate and optimize prompt effectiveness for specific use cases

Unit I: Artificial Intelligence **(15 Hours)**

Basic Definition and Terminology: Foundation and Evaluation of AI, Scope of AI, Components of AI, Types of AI, Application of AI**,** Intelligent Agent in AI: Types of AI agent, Concept of Rationality, Nature of environment, Structure of agents, Turing Test in AI**,** Search Algorithms in Artificial Intelligence: Properties of Search Algorithms, Types of Search Algorithms**,** Heuristic Search Techniques: Generate-and-Test; Hill Climbing. Properties of A* algorithm, Depth-First Search, Best-First Search, Greedy Best-First, Problem Reduction**,** Beyond Classical Search: Local search algorithms and optimization problem, Local search in continuous spaces, Searching with nondeterministic action and partial observation, Online search agent and unknown environments

UNIT II: Prompt Engineering **(15 Hours)**

Introduction to LLM, Generative AI, Understanding Large Language Models, Capabilities and limitations of LLMs, LLMs in the market (ChatGPT, Claude, Gemini, Llama), Hallucinations. What is prompt engineering, Prompt Ecosystem, prompt anatomy, Importance of prompt design, Basic prompt structure and syntax, tone, style and edge cases, types of prompts (open ended, closed ended, exploratory, multi-modal, contextual), Prompt efficiency, impact of efficiency, strategies for efficient prompt designing, optimizing efficiency, importance of syntax in prompts, best practices for crafting prompts, advanced prompt engineering techniques (Few-shot and zero-shot learning, Prompt tuning and optimization, Handling multi-turn conversations).

Reference Books:

1. Stuart J. Russell & Peter Norvig- “Artificial Intelligence: A Modern Approach” (Pearson, 2020 4th Ed.)
2. Tom Taulli – “Artificial Intelligence Basics: A Non-Technical Introduction” (Apress, 2019 1st Ed.)
3. Melanie Mitchell -“Artificial Intelligence: A Guide for Thinking Humans”(Pilican2020 -1st Ed.)

4. James Phoenix and Mike Taylor - "Prompt Engineering for Generative AI" (O'Reilly, 2024) - 422 pages, comprehensive guide
5. Eric C. Richardson- “Prompt Engineering” (BPB India, 2025, 1st Ed.), ISBN : 978-93-65892-963
6. Mark J. Baars - "AI Prompt Engineering: The 2025 Handbook" (2024) - practical guide
7. Valliappa Lakshmanan and Hannes Hapke - "Generative AI Design Patterns" (O'Reilly, 2024)
8. Yi Zhou - "Prompt Design Patterns" (2024)
9. Samuel Inbaraja S. - "A Practical and Short Textbook of Prompt Engineering" (2024)

B.Sc. Computer Science [Optional] (Semester-V) (NEP2.0) (Leel–5.0)

Multiple Entry and Multiple Exit Option (NEP-2020)

PART III SEM V (Level- 5.5)

Title of course: Major Practical V

Practical Based on Subject I Major IX and X

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

CO1: Develop user-defined functions with parameter passing.

CO2: Implement views, URL mapping, and templates**.**

CO3: Develop a complete mini project integrating all learned concepts.

CO4: Apply Chain-of-Thought, prompt chaining, and self-consistency techniques**.**

CO5: Design multi-step prompt workflows for solving complex real-world problems**.**

Sr. No.	Name of the Practical
1	Write Python programs to demonstrate variables, data types, and operators
2	Implement control statements (if-else, for loop, while loop) with practical examples
3	Create user-defined functions and demonstrate parameter passing
4	Perform operations on lists, tuples, dictionaries, and sets
5	Implement exception handling for common errors (ZeroDivisionError, FileNotFoundError)
6	Create programs for file reading and writing operations
7	Set up virtual environment and install Django using pip
8	Install Django and create a new Django project
9	Create a Django app and configure it in the project settings
10	Create simple views and map URLs to display "Hello World" page
11	Design HTML pages using Django templates with variables and template tags
12	Implement template inheritance (base template and child templates)
13	Add and use static files (CSS, JavaScript, images) in Django templates
14	Create Django forms and implement form validation
15	Create models (e.g., Student, Product) and perform database migrations
16	Implement CRUD operations (Create, Read, Update, Delete) using Django ORM
17	Register models with Django Admin and perform data management operations
18	Implement user registration, login, and logout functionality
19	Create a file upload application to upload and display images/documents
20	Develop a mini project (e.g., Blog System, Library Management, or E-commerce Product Catalog) using Django covering all concepts learned
21	Create basic prompts for different tasks: summarization, translation, and classification
22	Compare zero-shot vs few-shot prompting for sentiment analysis

23	Design role-based prompts (act as a teacher, lawyer, programmer, etc.)
24	Write prompts with specific output format constraints (JSON, table, bullet points)
25	Practice iterative prompt refinement to improve output quality
26	Generate professional emails for different scenarios (job application, complaint, inquiry)
27	Create creative content: short stories, poems, or blog posts on given topics
28	Write prompts for code generation in Python (functions, classes, algorithms)
29	Generate structured data: create quiz questions, survey forms, or datasets
30	Develop prompts for document summarization and key point extraction
31	Implement zero-shot Chain-of-Thought prompting for mathematical word problems
32	Create few-shot CoT prompts for logical reasoning tasks
33	Design multi-step prompts to solve complex problems (planning a trip, analyzing data)
34	Use prompt chaining to break down a complex task into subtasks
35	Implement self-consistency technique by generating multiple outputs and comparing them
36	Write detailed prompts for image generation (landscape, portrait, abstract art)
37	Create prompts with style specifications (photorealistic, cartoon, watercolor, etc.)
38	Use negative prompts to exclude unwanted elements from generated images
39	Design prompts for educational content creation (lesson plans, explanations, examples)
40	Mini Project: Develop a comprehensive prompt engineering solution for a real-world problem (customer service chatbot, content creation assistant, or research helper)

B.Sc. Computer Science [Optional] (Semester-V) (NEP2.0) (Leel–5.0)

Multiple Entry and Multiple Exit Option (NEP -2.0)

PART III SEM V (Level 5.5)

Title of the Course: Major Elective I: Core Java

Course Outcomes: After successful completion of this course, students will able to:

CO1: Use the syntax and semantics of java programming language and basic concepts of OOP.

CO2: Apply the concepts of Multithreading and Exception handling to develop efficient and error free code

CO3: Develop reusable programs using the concepts of inheritance, polymorphism, interfaces and packages.

CO4: Design and program stand-alone Java applications and GUI

Unit I: Introduction to Java and Object Oriented Programming (15 Hours)

Introduction to Java - History of Java and features of Java, Primitive Data Types- Integer (byte, short, int, long), floating point (float, double), char, boolean, Non-Primitive Data Type (String), Java Keywords, variables, constants, Operators- arithmetic, relational, logical, unary, ternary, bitwise, Branching and looping statements, Type casting- Implicit and Explicit, wrapper classes, Command line arguments, Writing simple java program, Compiling and executing Java program (javac, java commands).
Object Oriented Programming - Introduction- Class, Object and methods, Access modifiers and accessibility, Static members, constructors, destructor and this keyword, Encapsulation and Abstraction, Inheritance- Definition and its types single, multilevel, hierarchical, Interface – definition and implementation, Abstract Class – definition and use, Polymorphism- Definition and concepts of method overloading and overriding, Final method and Final Class, Java Packages – introduction, defining packages, CLASSPATH, importing packages, System Packages – java, lang, awt, javax, swing, net, io, util.

Unit – 2: Multithreading, Exception Handling, GUI Programming and Event Handling (15 Hours)

Multithreading and Exception Handling - Introduction to Multithreading, Understanding Threads, Thread Life-Cycle, Creating threads using Thread class & Runnable Interface, Thread Priorities, Exception handling - Fundamentals of exception handling, Exception types, Using try and catch, multiple catch clauses, throw, throws and finally, Built- in exceptions, Creating own exception sub classes.
GUI Programming and Event Handling - Introduction to GUI, Abstract Window Toolkit (AWT), Component and Container, Using Containers - Frame and Panel, Layout Managers - FlowLayout, GridLayout, CardLayout, BorderLayout, AWT Components – Label, Button, TextField, CheckBox, ChekBoxGroup, Event Handling- The Delegation event model, Events, Event sources, Event Listeners, Event classes, Handling mouse and keyboard events, Adapter classes, Inner classes, Anonymous Inner classes.

Reference Books –

1. Programming with Java A Primer, E. Balaguruswamy, Tata McGraw Hill Companies.
2. Java : The Complete Reference, Herbert Schildt, Tata McGraw-Hill
3. Java Programming- Rajendra Salokhe (Aruta Publication)
4. THE Java™ Programming Language, Fourth Edition By Ken Arnold, James Gosling, David Holmes
5. Introduction to Java programming, By Y. Daniel Liang, Pearson Publication.
6. Java How to Program, Sixth Edition, H.M.Deitel and P.J.Deitel, Pearson Education/PHI
7. The Java Tutorials: http://docs.oracle.com/javase/tutorial/

B.Sc. Computer Science [Optional] (Semester-V) (NEP2.0) (Leel–5.0)

Multiple Entry and Multiple Exit Option (NEP -2.0)

PART III SEM V (Level 5.5)

Title of the Course: Major Elective I: C#.NET

Course Outcomes: After successful completion of this course, students will able to:
CO1: Understand framework and architecture of .NET.
CO2: Develop object-oriented applications using C#.
CO3: Understand .NET architecture and C# programming fundamentals.
CO4: Learn graphical user interface (GUI) with windows form controls their properties, methods and events.

Unit I: Introduction to .NET (15 Hours)

An Overview of C#.NET, History and Features of C#.NET, .NET Framework Architecture, Introduction to Integrated Development Environment (IDE), Components of .NET - Common Language Runtime (CLR), Common Language Specification (CLS), Common Type System (CTS), Microsoft Intermediate Language (MSIL or IL), Metadata, Managed and Unmanaged code, Data Types: Value and Reference Types, Type casting, Boxing and Unboxing, JIT Compiler and it's types

Unit II: Object Oriented Concepts & Windows Form Applications (15 Hours)

Object Oriented Concepts - Classes, Objects, Polymorphism, Abstraction, Inheritance and it's types- Single, Multiple, Multilevel, Hierarchical, Parameter Passing Mechanism - 'val' and 'ref', Abstract Classes, Sealed Classes, Partial Classes, Exception Handling, Introduction to Windows Form Applications - Developing GUI Application Using WINFORMS - Basic Controls, Form Controls: Label, Button, Textbox, Checkbox, Radio Button, Timer, Calendar, List Box, Image and overview of remaining all common controls its properties & events.

Reference Books –

1. C# 4.0 The Complete Reference Schildt Mc Graw Hill
2. Inside C# - By Tom Archer, Andrew Whitechapel (Microsoft Pub)
3. Programming in C#- E Balagurusamy
4. Kogent Solutions, C# 2008 Programming covers. NET 3.5 (Black Book), Dreamtech Press

B.Sc. Computer Science [Optional] (Semester-V) (NEP2.0) (Leel–5.0)

Multiple Entry and Multiple Exit Option (NEP -2.0)

PART III SEM V (Level 5.5)

Title of the Course: Major Practical- I: Core Java

Course Outcomes: After successful completion of this course, students will able to:

CO1: Understand the fundamentals of Java programming**,** including data types, operators, control structures, type casting, wrapper classes, and command line arguments.

CO2: Develop and manage multithreaded applications and implement proper synchronization and thread life-cycle management.

CO3: Implement robust programs using exception handling mechanisms, including built-in and user-defined exceptions.

CO4: Design and develop GUI-based applications using AWT**,** and handle events using delegation event model, listeners, adapter classes, inner classes, and anonymous inner classes.

Sr. No.	Name of the Practical
1	Write a Java program to demonstrate all primitive data types and print their values.
2	Write a program to input 3 numbers on command line argument and find maximum number of them
3	Write a Java program to demonstrate branching statements (if, if-else, switch)
4	Write a Java program to demonstrate looping statements (for, while, do)
5	Write a Java program to demonstrate implicit and explicit type casting**.**
6	Write a Java program to demonstrate constructor and this keyword**.**
7	Write a Java program to demonstrate static variables and static methods**.**
8	Write a Java program to demonstrate inheritance**.**
9	Write a Java program to demonstrate method overloading and method overriding**.**
10	Write a Java program to create an interface Shape with the getArea() method. Create three classes Rectangle, Circle, and Triangle that implement the Shape interface. Implement the getArea() method for each of the three classes
11	Write a Java program to demonstrate Abstract class and abstract method**.**
12	Create a package College and add two classes Student and Staff to it. Add appropriate members to the classes.
13	Write a Java program that throws an exception and catch it using a try-catch block
14	Write a Java program to create a thread using: • Extending Thread class • Implementing Runnable interface
15	Write an AWT program to demonstrate FlowLayout, GridLayout, BorderLayout and CardLayout**.**
16	Design a simple GUI with a Label and Button. Each time user clicks the button increase counter and display number of counts on the label.
17	Develop a GUI application using AWT to create a simple calculator using: • Label • TextField • Button • Layout Managers
18	Design a simple GUI for Temperature converter, using appropriate components.

19	Write a java program using swing to create a frame having three text fields. Accept number in first textfield and display previous number in second textfield and next number in the third textfield.
20	Write a Java program that works as a simple calculator. Use a grid layout to arrange buttons for the digits and for the +, -, *, % operations. Add a text field to display the result. Handle any possible exceptions like divide by zero.

B.Sc. Computer Science [Optional] (Semester-V) (NEP2.0) (Leel–5.0)

Multiple Entry and Multiple Exit Option (NEP -2.0)

PART III SEM V (Level 5.5)

Title of the Course: Major Practical- I: C#.NET

Course Outcomes: After successful completion of this course, students will able to:

CO1: Understand the architecture of the .NET Framework including CLR, CTS, CLS, MSIL, Metadata, JIT Compiler, and managed/unmanaged code.

CO2: Apply fundamental C#.NET programming concepts such as data types, type casting, boxing/unboxing, and parameter passing mechanisms.

CO3: Implement Object-Oriented Programming principles including classes, inheritance, polymorphism, abstraction, sealed and partial classes.

CO4: Develop robust applications using structured exception handling mechanisms in C#.NET.

Sr. No.	Name of the Practical
1	Write a C# program to check given number is perfect or not by using command line argument.
2	Write a C# program to demonstrate branching statement.
3	Write a C# program to demonstrate looping statement.
4	Write a console application for swapping of 2 numbers using Pass by Value.
5	Write a console application for swapping of 2 numbers using Pass by Reference.
6	Write a C# program to demonstrate different value types and reference types.
7	Write a C# program to demonstrate type casting
8	Write a C# program to demonstrate boxing and unboxing
9	Write a C# program to demonstrate Polymorphism.
10	Write a C# program to demonstrate Multilevel Inheritance.
11	Write a C# program to demonstrate Hierarchical Inheritance.
12	Write a C# program to demonstrate Interface.
13	Write a C# program to demonstrate sealed class
14	Write a C# program to demonstrate abstract class.
15	Write a C# program to demonstrate partial class
16	Write a C# program to demonstrate exception handling.
17	Write a C# program to create a user-defined exception and handle it in a program.
18	Develop a Windows Form application to perform basic arithmetic operations using Label, TextBox, and Button controls.
19	Create a Windows Form application to demonstrate use of CheckBox, RadioButton, ListBox, and Calendar controls.
20	Develop a Windows Form application using Timer and Image controls and demonstrate event handling.

B.Sc. Computer Science [Optional] (Semester-V) (NEP2.0) (Leel–5.0)

Multiple Entry and Multiple Exit Option (NEP -2.0)

PART III SEM V (Level 5.5)

Title of the Course: VSC-II (P) : Major Specific: Node.js

Course Outcomes: After successful completion of this course, students will able to:

CO1: Develop dynamic web applications using File System, HTTP Server, and Query String modules with proper request-response handling.

CO2: Implement non-blocking I/O operations by applying asynchronous programming techniques including callbacks, promises, and async/await patterns.

CO3: Build real-time applications using Event Emitter class, event-driven architecture, and custom event handling with multiple listeners.

CO4: Design and export custom modules while effectively utilizing core, local, and third-party modules in modular application development.

CO5: Perform complete database operations by establishing MySQL connectivity and executing CRUD commands including insert, select, update, and delete.

CO6: Manipulate server-side data using buffers, global scope, and functions while analyzing the Node.js process model and event loop.

Sr.No.	Name of the Practical
1	Write a Node.js script to display the current Node.js version and operating system platform using the process object.
2	Create a Buffer from the string "Node.js" and convert it back to string after displaying the raw buffer content.
3	Write a Node.js program that prints "Hello" immediately and "World" after a 3-second delay using setTimeout().
4	Create an HTTP server that listens on port 3000 and displays "Server Running" in the browser.
5	Write an HTTP server that displays "Home Page" for /home route and "About Page" for /about route.
6	Create a simple API endpoint /user that returns a JSON object with name, age, and city fields.
7	Write a program using the url module to parse the URL http://localhost/ profile?id=101 and display the pathname and query string.
8	Extract the name and age parameters from the query string "name=John&age=25" using the querystring module.
9	Create a custom module math.js that exports an add(a,b) function and use it in another file app.js.
10	Write a Node.js script to display the current working directory and memory usage using global process object.
11	Create an HTTP server that reads the name parameter from URL query and displays "Hello [name]" in the browser.
12	Install and use the chalk module to print "Success" in green color and "Error" in red color.
13	Create an HTTP server that handles both GET and POST requests and displays appropriate messages for each method.
14	Write an EventEmitter program where a listener receives arguments (name and age) and displays them when an "info" event is emitted.
15	Create an event listener that runs only once using the once() method and verify that it doesn't execute on second emit.
16	Write a Node.js program to connect to MongoDB and insert a single document into a "students" collection.

17	Create a MongoDB connection in Node.js to fetch and display all records from a "products" collection.
18	Write a Node.js program to update the price of a product in MongoDB where the product name is "Laptop".
19	Create a Node.js script to delete a record from the "users" collection where the age is less than 18.
20	Write a MongoDB query in Node.js to fetch records from an "orders" collection sorted by the "amount" field in descending order.

B.Sc. Computer Science [Optional] (Semester-V) (NEP2.0) (Leel–5.0)

Multiple Entry and Multiple Exit Option (NEP -2.0)

PART III SEM V (Level 5.5)

Title of the Course: AEC-II (T): Corporate Culture & Communication

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

CO1. To develop effective professional communication skills
CO2. To understand organizational and cross-cultural workplace behavior
CO3. To enhance ethical, digital, and AI-era communication competence.
CO4. To prepare students for corporate and global work environments.

UNIT I: Introduction to Business Communication (15 Lectures)

Introduction - Meaning and definition of business communication, Objectives and importance in organizations, Types - Internal and External communication, Role of communication in AI-driven businesses. Communication Process and Models - Elements of communication process, Models of Communication Cycle, Characteristics of Professional communication, Barriers to communication, Overcoming communication barriers, Verbal and Non-Verbal Communication - Verbal communication - oral and written Non- verbal communication- body language, kinesics, proxemics, Paralanguage and visual communication -Impact of non-verbal cues in professional settings, Written Business Communication - Principles of effective business writing, Business letters - inquiry, complaint, adjustment, Emails, memos, notices, Report writing - short reports and summaries, AI tools in business writing (email drafting, grammar tools), Oral Communication & Presentation Skills – Meetings- types and conduct, Interviews- preparation and participation, Public speaking and presentation skills, Use of visual aids and AI presentation tools

UNIT II: Concept of Culture in Business (15 Lectures)

Introduction - Meaning and elements of culture Organizational culture -types and significance, Corporate culture in technology-based organizations, Cultural diversity in the workplace, Cross Cultural Communication- High-context and low-context cultures, Hofstede's cultural dimensions Cross-cultural challenges in global business, Communication in multicultural teams ,AI and globalization of workplace communication, Professional Ethics- Meaning and importance of business ethics, Ethical issues in communication, Corporate Etiquette - Workplace behavior, dress code, Email, meeting, and virtual etiquette, Ethics in AI communication and data usage, Digital Communication and AI Workplace Culture: Digital communication tools and platforms, Social media communication for business, Remote work culture and virtual collaboration ,Impact of AI on organizational communication, Future trends in business communication.

References :

1. Bovee, C. L. & Thill, J. V. Business Communication Today
2. Lesikar, R. V. Business Communication: Theory and Application
3. Hofstede, G. Cultures and Organizations
4. Guffey, M. E. Business Communication: Process and Product

B.Sc. Computer Science [Optional] (Semester-V) (NEP2.0) (Leel–5.0)

Multiple Entry and Multiple Exit Option (NEP-2020)

PART III SEM V (Level- 5.5)

Title of course: **OJT**

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

1. Apply theoretical knowledge gained during the academic program to real-world industrial or organizational tasks.
2. Demonstrate professional work ethics, discipline, time management, and responsibility in a workplace environment.
3. Develop technical and practical skills relevant to the specific industry or domain of training.
4. Exhibit effective communication, teamwork, and interpersonal skills while working with professionals.
5. Analyze workplace problems and propose practical solutions using appropriate tools and methodologies.
6. Prepare a comprehensive training report and present learning outcomes, reflecting on professional growth and career readiness.

Guidelines provided by Shivaji University will be followed.

B.Sc. Computer Science [Optional] (Semester-VI) (NEP2.0) (Leel–5.0)

Multiple Entry and Multiple Exit Option (NEP-2020)

PART III SEM VI (Level- 5.5)

Title of course: **Data Analytics with Python**

Course Objectives: After completion of this course, students should be able to:

CO1: Understand the fundamentals of data analysis and its applications
CO2: Utilize NumPy for numerical computing and array operations
CO3: Apply Pandas for data manipulation, cleaning, and analysis
CO4: Implement data visualization techniques using Matplotlib and Pandas

UNIT I: Introduction to Data Analytics and NumPy (15 Hours)

Introduction to Data Analytics - Data Analysis Process, Types of Data Analysis (Descriptive,Diagnostic, Predictive, Prescriptive), Characteristics of Data Analysis, Real-world Applications of Data Analysis, NumPy Fundamentals - Introduction to NumPy library, Installing and importing NumPy, NumPy arrays vs Python lists, NumPy ndarray - Creating NumPy arrays (1D, 2D, 3D), Array attributes (shape, size, dtype), Array indexing and slicing, Array reshaping and resizing, Array Operations - Universal Functions (element-wise operations), Arithmetic operations on arrays, Array-oriented programming, Broadcasting, File I/O and Random Numbers - Reading and writing arrays to files, Generating pseudorandom numbers, Random number distributions, Introduction to Pandas - Overview of Pandas library, Installing and importing Pandas, Pandas data structures: Series and DataFrame, Working with Series and DataFrame - Creating Series and DataFrame objects, Accessing data using indexing and selection, Essential DataFrame operations

UNIT II: Data Manipulation and Visualization (15 Hours)

Data Loading and File Formats - Reading data from text files (CSV, TSV), Writing data to text format, Working with delimited formats, Reading Excel files, Handling different file encodings, Descriptive Statistics: Summarizing data, Computing descriptive statistics (mean, median, mode, standard deviation), Aggregation functions, GroupBy operations, Data Cleaning and Preparation - Handling missing data (detecting, filtering, filling), Data transformation techniques, Removing duplicates, String manipulation and cleaning, Data Visualization with Matplotlib - Introduction to Matplotlib, Creating basic plots (line, scatter, bar), Matplotlib API primer, Customizing plots (labels, titles, legends, colors), Plotting with Pandas - Creating plots directly from DataFrame and Series, Line plots, bar plots, histograms, and box plots, Visualizing data distributions, Combining multiple visualizations

Reference Books:

1. Wes McKinney, "Python for Data Analysis", O'Reilly Media, 3rd Edition, 2022
2. Jake VanderPlas, "Python Data Science Handbook", O'Reilly Media, 2016
3. Joel Grus, "Data Science from Scratch", O'Reilly Media, 2nd Edition, 2019
4. R. Nageswara Rao, "Core Python Programming", Dreamtech Press / Wiley India, 3rd Edition, 2020
5. R. Nageswara Rao, "Machine Learning in Data Science Using Python", Dreamtech Press / Wiley India, 1st Edition, 2022
6. Gowrishankar S. and Veena A., "Introduction to Python Programming", CRC Press, 1st Edition, 2018

Web Sites:

1. NumPy Official Documentation: https://numpy.org/doc/
2. Pandas Official Documentation: https://pandas.pydata.org/docs/
3. Matplotlib Official Documentation: https://matplotlib.org/

B.Sc. Computer Science [Optional] (Semester-VI) (NEP2.0) (Leel–5.0)

Multiple Entry and Multiple Exit Option (NEP-2020)

PART III SEM VI (Level- 5.5)

Title of course: **Big Data Management**

Course Outcomes: After successful completion of this course, students will able to:

CO1. Understand the evolution and basics of Big Data Management.
CO2. Identify components and tools used in Big Data ecosystem.
CO3. Apply basic Big Data processing techniques.
CO4. Compare Big Data storage models and NoSQL databases for business applications.

Unit I: Big Data, Hadoop & HBase (15 Hours)

Introduction & Definition of Big Data, Need of Big Data Management, Sources of Big Data, Characteristics (5 Vs), Evolution of Big Data, Data Warehouse vs Big Data, Structured vs Semi-structured & Unstructured Data, Big Data Life Cycle, Applications of Big Data (Business, Healthcare, Govt), Benefits & ChallengesPrivacy, Security & Compliance (intro level), Hadoop: Introduction & Features, Hadoop Architecture (Master–Slave concept), Hadoop Daemons (NameNode, DataNode, ResourceManager, etc.), HDFS Introduction & Basic Operations, MapReduce - Concept & Architecture, Simple MapReduce example (word count – concept only), Limitations of HDFS, HBase Introduction, HDFS vs HBase, HBase Architecture & Applications, Higher Level Tools: Introduction to Pig, Pig Architecture, Introduction to Hive, Hive Architecture, Hive vs RDBMS

Unit II: Big Data Storage Models & NoSQL (15 Hours)

Big Data Storage Models (Overview) - Key-Value (Dynamo), Document Store (MongoDB, Cassandra – intro), Graph Model (Neo4j – concept), Introduction to NoSQL - Types of NoSQL Databases, RDBMS vs NoSQL, MongoDB, Introduction, CRUD Operations (Insert, Find, Update, Delete),Indexing (basic), NoSQL Database Administration (intro concepts)

Reference Books :

1. Zikopoulos, Eaton, et al. – Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data — McGraw-Hill Osborne Media, 2017
2. Tom E. White – Hadoop: The Definitive Guide — O'Reilly Media, 4th Ed., 2015
3. Alan Gates – Programming Pig — O'Reilly Media, 2st Ed., (2016)
4. Capriolo, Wampler & Rutherglen – Programming Hive — O'Reilly Media, 1st Ed., (2012–2013)
5. Plunkett, Macdonald, et al. – Oracle Big Data Handbook — Oracle Press, 1st Ed.,2014
6. Ajit Singh– Data Modeling with NoSQL Database — (Academic Publisher), 3rd Ed., (2022)

B.Sc. Computer Science [Optional] (Semester-V) (NEP2.0) (Leel–5.0)

Multiple Entry and Multiple Exit Option (NEP-2020)

PART III SEM VI (Level- 5.5)

Title of course: Major Practical VI

Practical Based on Subject I Major XI and XII

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

CO1: Generate and analyze random data using probability distributions**.**

CO2: Perform joining, filtering, grouping, and aggregation operations.

CO3: Understand Hadoop architecture and configure pseudo-distributed mode**.**

CO4: Apply tools from NumPy, Pandas, Matplotlib, Hadoop, Hive, Pig, and MongoDB to solve real-world data problems.

Sr. No.	Name of the Practical
1	Write a NumPy program to create arrays of different dimensions (1D, 2D, 3D) and display their properties
2	Write a NumPy program to perform array indexing, slicing, and reshaping operations
3	Write a NumPy program to split the elements of an array with spaces
4	Write a NumPy program to find the most frequent value in an array
5	Write a NumPy program to generate random numbers from normal distribution and calculate their statistical properties
6	Write a NumPy program to sort a specified number of elements from the beginning of an array
7	Write a NumPy program to calculate the difference between the maximum and minimum values of an array
8	Write a NumPy program to compute the cross product of two vectors
9	Write a Pandas program to convert a NumPy array to a Pandas Series
10	Write a Pandas program to create a DataFrame and convert its first column as a Series
11	Write a Pandas program to join two DataFrames along rows and columns
12	Write a Pandas program to split a DataFrame based on specific conditions
13	Write a Pandas program to import data from Excel and CSV files
14	Write a Pandas program to perform descriptive statistical analysis on a dataset
15	Write a Pandas program to find and replace missing values in a DataFrame using different strategies
16	Write a Pandas program to split datasets into groups and perform aggregation operations
17	Write a program to create line plots using Matplotlib and Pandas
18	Write a program to create bar charts and histograms for data distribution analysis
19	Write a program to create scatter plots and visualize correlations between variables
20	Mini Project: Perform complete data analysis on a real-world dataset including data loading, cleaning, statistical analysis, and visualization
	Install Hadoop in pseudo-distributed mode (or study architecture using demo setup).
22	Create and manage directories in HDFS.
23	Upload and download files in HDFS.
24	Perform basic HDFS operations (ls, put, get, rm).
25	Write a Word Count MapReduce program (Java/Python) – dry run.
26	Execute Word Count on sample text dataset.
27	Analyze output of MapReduce job.

28	Load dataset into Pig and display records.
29	Apply Filter and Projection in Pig.
30	Perform Group By and Aggregation in Pig.
31	Perform Inner Join in Pig.
32	Perform Outer Join in Pig.
33	Create Hive database and tables.
34	Load data into Hive table.
35	Perform SELECT, WHERE queries in Hive.
36	Perform Group By and Order By in Hive.
37	Perform Join in Hive.
38	Create database and collection in MongoDB.
39	Perform CRUD operations in MongoDB.
40	Apply Indexing and query optimization in MongoDB.

B.Sc. Computer Science [Optional] (Semester-VI) (NEP2.0) (Leel–5.0)

Multiple Entry and Multiple Exit Option (NEP -2.0)

PART III SEM VI (Level 5.5)

Title of the Course: Major Elective II: Advanced Java

Course Outcomes: After successful completion of this course, students will able to:

CO1: Develop distributed business applications, develop web pages using advanced server- side programming through servlets and Java server pages.

CO2: Demonstrate approaches for performance and effective coding.

CO3: Learn database programming using Java.

CO4: Study web development concept using Servlet and JSP

Unit – I: Java Swing and JDBC (15 Hours)

Java Swing - Overview of Java Swing and its advantages, Java Swing MVC architecture, Component of Swing - JFrame, JComponent, JLable, JTextfields, JCheckbox, JPanel, JRadiobuttons, JTabbed Pane, JButton, JTree, JTable, JMenu, Difference between AWT and Swing, Event handling in Swing, JDBC - Introduction Java Data Base Connectivity (JDBC), JDBC Architecture and Components, Establishing a connection to a database using DriverManager, Handling database connections - opening and closing connections, Overview of JDBC statements: Statement, PreparedStatement, Executing SQL queries and updates, Retrieving and processing result sets, Types of ResultSet.

Unit – 2: Servlet and Java Server Pages (JSP) (15 Hours)

Servlet - Introduction to Servlet and Servlet Class Hierarchy, Life-Cycle of Servlet, Servlet API, Directory structure of Servlet Application, ServletConfig and ServletContext, Handling HTML form data – get and post method, Session tracking –URL rewriting, Cookies, and HTTpSession.

Java Server Pages (JSP) - Introduction to JSP, Life cycle of JSP, JSP v/s Servlet, Components of JSP: Directives, Tags, Scripting elements – Declarations, Expressions, Scriptlets, Comments, Implicit objects of JSP, Connecting to database.

Reference Books –

1. Programming with Java A Primer, E. Balaguruswamy, Tata McGraw Hill Companies.
2. Java : The Complete Reference, Herbert Schildt, Tata McGraw-Hill
3. Java Programming- Rajendra Salokhe (Aruta Publication)
4. THE Java™ Programming Language, Fourth Edition By Ken Arnold, James Gosling, David Holmes
5. Introduction to Java programming, By Y. Daniel Liang, Pearson Publication.
6. Java How to Program, Sixth Edition, H.M.Deitel and P.J.Deitel, Pearson Education/PHI
7. Java Database Best Practices, by George Reese, O'Reilly
8. Head First Servlets and JSP" by Bryan Basham, Kathy Sierra, and Bert Bates.
9. "Java Swing" by Marc Loy
10. The Java Tutorials: http://docs.oracle.com/javase/tutorial/

B.Sc. Computer Science [Optional] (Semester-VI) (NEP2.0) (Leel–5.0)

Multiple Entry and Multiple Exit Option (NEP -2.0)

PART III SEM VI (Level 5.5)

Title of the Course: Major Elective II: ASP.NET

Course Outcomes: After successful completion of this course, students will able to:

CO1: Understand Web server, HTTP request response architecture.
CO2: Learn Web forms and their controls.
CO3: Learn state management in web forms.
CO4: Understand ADO.NET Architecture with connection oriented and disconnected layer.

Unit – 1: Introduction to ASP.NET (15 Hours)

Introduction to Scripting Languages, ASP.NET Introduction, Features of ASP.NET, Web browser and web server, HTTP request response structure, HTML form elements, GET/POST method, Client-side and Server-side programming, Web form life cycle, Page events, Server Controls: Textbox, Listcontrols, FileUpload, Linkbutton, Imagemap, Image, Imagebutton, Calender, Literal control, Radiobutton, Checkbox, Validation Controls, Navigation controls, Master Page.

Unit – 2: State Management and Database Connectivity (15 Hours)

State Management - Cross page postback property of button, Response.Redirect, Server.transfer, Response.Write, State Management - Session, Application, Global.asax, Caching, Database Connectivity - SqlServer Database, Data controls- Gridview, Listview , FormView, DetailsView, Repeater, Introduction to ADO.Net, ADO.NET Architecture, Connection, Command, DataReader, DataAdapter, DataSet, Connected and Disconected layer of ADO.NET.

Reference Books -

1. ASP .NET-The Complete Reference Tata MacGraw Hill
2. ASP.NET 4 Unleashed by Stephen Walther, Kevin Scott Hoffman, Sams Publishing
3. Microsoft ASP.NET 4.0 Step by Step - George Shepherd, Microsoft Press
4. Mastering ASP.Net - BPB Publication
5. ASP.NET 4.0 Programming- Joydip Kanjilal

B.Sc. Computer Science [Optional] (Semester-VI) (NEP2.0) (Leel–5.0)

Multiple Entry and Multiple Exit Option (NEP -2.0)

PART III SEM VI (Level 5.5)

Title of the Course: Major Practical II: Advanced Java

Course Outcomes: After successful completion of this course, students will able to:

CO1: Understand and explain the concepts of Java Swing architecture (MVC), components, and event handling mechanisms for building desktop applications.

CO2: Design and develop GUI-based applications using Swing components such as JFrame, JTextField, JButton, JTable, JTree, JMenu, and JTabbedPane.

CO3: Analyze and implement connected and disconnected database operations using ResultSet and different ResultSet types.

CO4: Implement session management techniques (URL rewriting, Cookies, HttpSession) and develop dynamic web pages using JSP with database connectivity.

Sr.No.	Name of the Practical
1	Create a simple Swing application using JFrame, JLabel, JTextField, and JButton to display and accept user information.
2	Develop a Swing application demonstrating event handling using ActionListener on a JButton.
3	Design a Student Registration Form using: • JTextField • JCheckBox • JRadioButton • JPanel • JButton Display entered data on button click.
4	Create an application using JTabbedPane with at least three tabs (Profile, Course, Contact).
5	Create a Swing application using JMenuBar and JMenu with menu items like New, Open, Exit.
6	Create a Swing application that displays student records using JTable**.**
7	Develop a Swing program to display hierarchical data (e.g., College → Department → Students) using JTree**.**
8	Create two small programs demonstrating the difference between AWT and Swing components**.**
9	Write a Java program to establish a connection with MySQL/Oracle using DriverManager**.**
10	Create a JDBC program to insert student data into database using Statement**.**
11	Write a program to insert records using PreparedStatement**.**
12	Create a program to retrieve and display records using ResultSet**.**
13	Write a JDBC program to update and delete records from the database.
14	Create a program demonstrating: • TYPE_FORWARD_ONLY • TYPE_SCROLL_INSENSITIVE • TYPE_SCROLL_SENSITIVE
15	Create a simple Servlet that prints “Hello World” in browser.
16	Create a servlet demonstrating: • init() • service() • destroy()

17	Create an HTML form and retrieve form data in Servlet using: • GET method • POST method
18	Develop a Servlet application demonstrating: • URL Rewriting • Cookies • HttpSession
19	Create a JSP page demonstrating: • Declarations • Scriptlets • Expressions • Comments
20	Develop a JSP application to: • Connect to database • Insert record • Display records in tabular format

B.Sc. Computer Science [Optional] (Semester-VI) (NEP2.0) (Leel–5.0)

Multiple Entry and Multiple Exit Option (NEP -2.0)

PART III SEM VI (Level 5.5)

Title of the Course: Major Practical II: ASP.NET

Course Outcomes: After successful completion of this course, students will able to:

CO1: Develop dynamic web applications using ASP.NET Web Forms with server controls, validation controls, navigation controls, and master pages.

CO2: Apply response handling techniques including Response.Write, Response.Redirect, and Server.Transfer in ASP.NET applications.

CO3: Design and implement database-driven web applications using ADO.NET architecture.

CO4: Develop data-centric web applications using data-bound controls such as GridView, ListView, DetailsView, FormView, and Repeater integrated with SQL Server.

Sr. No.	Name of the Practical
1	Write an ASP.Net Program to print a Message on web form.
2	Write an ASP.Net Program to create simple Web Application using two or more web form.
3	Write an ASP.NET Program to set a link for new Page.
4	Write an ASP.NET Program to demonstrate different common Control.
5	Write an ASP.NET program using while or for loop.
6	Write an ASP.NET Program to add the value of Text Box in to Dropdown List and List box controls.
7	Write an ASP.NET Program to delete items from Dropdown list and ListBox.
8	Design a webpage to demonstrate the working of GET and POST methods using HTML form elements.
9	Develop an application to upload an image file and display it on the webpage.
10	Create a webpage that allows users to select a date using Calendar control and display selected date.
11	Create a Master Page with header, footer, and navigation menu. Create at least 3 content pages linked with the master page.
12	Create two webpages where data entered in first page is accessed in second page using CrossPagePostBack property.
13	Create a webpage demonstrating the difference between Response.Write, Response.Redirect and Server.Transfer
14	Create a login form and store username in Session**.** Display welcome message on next page.
15	Create a webpage demonstrating output caching and data caching.
16	Create a Student table in SQL Server and connect it with ASP.NET application.
17	Insert student data into SQL Server using: • SqlConnection • SqlCommand
18	Retrieve and display records using DataReader**.**
19	Retrieve data using DataAdapter and DataSet and display using GridView.
20	Create a complete CRUD application using: • GridView • DetailsView • FormView • Repeater • ListView

B.Sc. Computer Science [Optional] (Semester-VI) (NEP2.0) (Leel–5.0)

Multiple Entry and Multiple Exit Option (NEP -2.0)

PART III SEM VI (Level 5.5)

Title of the Course: VSC-III (P): Major Specific: Linux Utilities and Shell Programming

Course Outcomes: After successful completion of this course, students will able to:

CO1: Learn architecture and basics of Linux Operating System.

CO2: Understand the working and use of NANO editor.

CO3: Learn Regular expressions using metacharacters.

CO4: Learn advanced BASH shell Programming.

Sr. No.	Name of the Practical
1	Demonstration of General-Purpose Utilities.
2	Implement file and directory operations: touch, cp, mv, rm, mkdir, rmdir, cd.
3	Write a shell script that prompts the user for a name of a file or directory and reports if it is a regular file, a directory, or another type of fil. Also perform an ls command against the file or directory with the long listing option.
4	Pattern matching with grep: Search files for patterns using basic regex (*, ?, ^, $, .) and options like -i, -v, -n.
5	Text processing with filters: Use head, tail, wc, and grep to extract and manipulate data files.
6	Advanced filtering: Apply sed for substitutions (s/pattern/replace/g).
7	Basic nano editing: Install nano ,Create, edit, save, and exit files using Ctrl+O, Ctrl+X; practice search (Ctrl+W) and replace. Multi-line operations: Cut (Ctrl+K), paste (Ctrl+U), undo (Alt+U) and create backups (-B).
8	Write a shell script which copies files from one directory to another during copy command.
9	Create a data file which contains given format and perform the given operations on that data file using sed.
10	Write a shell script using grep command to print prime numbers between 1 to 30.
11	Write a shell script to display given year is leap year or not.
12	Write a shell script using if statements to check file exists or not.
13	Write a shell script to calculate the grade of student.
14	Write a shell script to print the Fibonacci series.
15	Write a shell script that check number Armstrong or not.
16	Write a shell script to find out given word contains vowel and also the entered vowel is small case or capital.
17	Write a shell script to read name, sex and marital status and display the same.
18	Write a shell script that display “man”, “bear”,”pig”,”dog”, “cat”, and “sheep” on the screen with each appearing on a separate line. Try to do thins in as few lines as possible.
19	Write a shell script to greet message according to time.
20	Write a shell script that read 5 numbers; count and display total even/odd.

B.Sc. Computer Science [Optional] (Semester-VI) (NEP2.0) (Leel–5.0)

Multiple Entry and Multiple Exit Option (NEP -2.0)

PART III SEM VI (Level 5.5)

Title of the Course: SEC-III: Software Engineering

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

CO1: Understand life cycle models, requirement elicitation techniques, understand the concept of analysis and design of software.
CO2: Acquire a comprehensive understanding of the software development lifecycle and its application in contemporary software engineering practices.
CO3: Develop proficiency in project management methodologies and strategic decision-making for successful software project execution.
CO4: Apply software engineering concepts in software development to develop quality software

Unit- I: Introduction to System and System Design Tools (15 Hours)

Introduction to System - Definition, Elements of System, Characteristics of System, Types of System - Open, Closed, Deterministic, Probabilistic, Manual and Automated, System Development Life Cycle (SDLC), Design - Design Concepts, Design Process and Design Quality, Design Model System Design Tools - Data Flow Diagrams (DFD), Entity-Relationship Diagrams (ERD), Agile Software Development - Agility Principles, Agile methods, Plan-driven and agile development

Unit-II: Introduction to Software Engineering and Testing (15 Hours)

Software Engineering - Definition, importance, principles of software engineering, Software Process Models: Waterfall, Incremental, Prototyping and Spiral model, Software Design: SOLID Design Principles, Software Testing: Definition of Testing, Testing characteristics, Types of Testing, Black-Box Testing, White-Box Testing, Unit testing, Integration testing, Validation Testing and System Testing, Software Quality Assurance -Quality and its Attributes, Quality Control, Quality Assurance, Cost of Quality, SQA Activities, SQA Plan

References:

1. Software Engineering, N.S. Gill, Khanna Publishing House, 2023
2. Software Engineering, Ian Somerville, 9th edition, Pearson education.
3. Software Engineering a Practitioners Approach by S. Pressman & Roger, Seventh Edition, McGraw Hill International Edition.
4. Software Engineering by Sommerville,7th edition, Pearson Publication
5. Software Engineering by K.K. Aggarwal & Yogesh Singh, New Age International Publishers

B.Sc. Computer Science [Optional] (Semester-VI) (NEP2.0) (Leel–5.0)

Multiple Entry and Multiple Exit Option (NEP -2.0)

PART III SEM V (Level 5.5)

Title of the Course: AEC IV: Professional Ethics

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

CO1: Analyze ethical dilemmas using structured ethical decision-making frameworks, considering bias, empathy, and conflicts of interest.

CO2: Apply philosophical and psychological approaches to evaluate ethical behavior, integrity, courage, and emotional intelligence in professional life.

CO3:Assess ethical issues related to research, academics, intellectual property, confidentiality, employee rights, and workplace diversity.

CO4: Evaluate professional responsibilities related to safety, risk management, environmental protection, and sustainable development.

UNIT-I **(15 Hours)**

Foundations of Ethics, Meaning and scope of ethics, Morals, values, and ethics: personal and professional, Individual vs professional ethics, Sense of professional and engineering ethics, Code of professional conduct (with reference to National Society of Professional Engineers), Ethical Decision Making, Ethical dilemmas and moral reasoning, Ethical decision-making models and roadmap, Bias, empathy, and internal obstacles, Conflict of interest and cultural relativism

UNIT-II **(15 Hours)**

Business Ethics and Corporate Behavior, Philosophical approaches to business ethics, Ethical issues in business and organizations, Ethical leadership and corporate governance, Resisting unethical authority and domination, Global business ethics, Psychological and Professional Perspectives, Psychological and philosophical approaches to ethics, Myths about morality, Courage, integrity, emotional intelligence, Ethics in research and academics: academic integrity and intellectual honesty, Ethics in the Workplace, Ethical issues in diverse workplaces, Employee rights, confidentiality, and discrimination, Sustainability and Corporate Skills, Ecology, engineering, and economy, Risk–benefit analysis and risk reduction, Corporate Social Responsibility (CSR), CSR in India and corporate sustainability

References:

1. Subramanian.R. Professional Ethics, Oxford Publication, 2013.
2. Nagarasan. R.S. Professional Ethics and Human Values. New Age International Publications, 2006.
3. Mike W Martin and Roland Schinzinger, Ethics in Engineering,4th edition, Tata McGraw Hill Publishing Company Pvt Ltd, New Delhi,2014

B.Sc. Computer Science [Optional] (Semester-VI) (NEP2.0) (Leel–5.0)

Multiple Entry and Multiple Exit Option (NEP -2.0)

PART III SEM V (Level 5.5)

Title of the Course: IKS

Course Outcomes After completing the course, students will be able to:
CO1: Explain core concepts and philosophy of Indian Knowledge Systems
CO2: Relate ancient Indian logic and mathematics to modern computer science
CO3: Apply IKS-inspired algorithms in computational problem-solving
CO4: Analyze the role of Sanskrit grammar in NLP and AI
CO5: Design ethical and sustainable AI systems using IKS principles
CO6: Evaluate real-world case studies integrating IKS with modern technology

UNIT I **(15 Hours)**

Definition and scope of Indian Knowledge Systems, Importance of ancient knowledge in contemporary science and technology, The IKS corpus: Vedas, Upanishads, Smritis, Sutras, Shastras, Objectives, nature, scope, and unique characteristics of IKS, Unique number representation systems: Bhūta Saṃkhya System, Kaṭapayādi System, Pingala's binary system (Laghu–Guru) and its importance in computing, Sanskrit phonetics and linguistics, Panini's Ashtadhyayi**,** Algorithmic nature of Paninian grammar, Role of Sanskrit in computational linguistics, Morphological analysis and syntax parsing,

UNIT II **(15 Hours)**

Applications: POS tagging, Machine translation, AI-driven language models for Indian languages, Knowledge pyramid and epistemology in IKS, Nyaya logic in knowledge representation, Pramana, Prameya, Samsasya as AI reasoning models, Ethical AI inspired by Indian philosophy, Sustainable and socially responsible AI systems
Case Studies: Compiler Design, Recursion, Encryption Decryption, Computer Graphics, Algorithm Time Complexity, Binary Combinations, Vedic Mathematics for Signal Processing, Paninian Grammar for NLP and AI language models, Machine Learning inspired by Nyaya Logic

References

1. Textbook on IKS by Prof. B Mahadevan, IIM Bengaluru.
2. Bhartiya Knowledge Systems by M.C. Bora, Khanna Publishing House
3. Kautilya The Arthashastra. Translated by R. Shamasastry
4. Kapoor, Kapil, and Singh Avadesh Kumar, Indian Knowledge System Vol.1, DK Print World, Ltd., 2005.
5. Mahadevan B., Bhat V R, Nagendra Pavana R.N., Indian Knowledge System Concepts and Application, PHI Learning Pvt. Ltd., 2022

B.Sc. Computer Science [Optional] (Semester-VI) (NEP2.0) (Leel–5.0)

Multiple Entry and Multiple Exit Option (NEP -2.0)

PART III SEM V (Level 5.5)

Title of the Course: FP

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

1. Identify real-world problems in industry, community, or research environments and define clear project objectives.
2. Apply theoretical knowledge and practical skills acquired during the program to solve field-based problems.
3. Collect, analyze, and interpret data using appropriate tools, techniques, and methodologies.
4. Demonstrate teamwork, leadership, communication, and professional ethics while working in real-time project environments.
5. Prepare structured project reports and deliver effective presentations showcasing findings, outcomes, and recommendations.
6. Evaluate project outcomes critically and suggest improvements or future enhancements based on practical experience.

Common Guidelines will be provided by Shivaji University, Kolhapur