Academic Year: 2024-25

Skill Enhancement Course-I

S.No.	Sem & Section	Title	Date (From -to)
1.	III SEM CSE-A,B,C&D	Python Programming	06/08/2024 -12/08/2024

Skill Enhancement Course-II

S.No.	Sem & Section	Title	Date (From -to)
1.	IV SEM CSE-A,B,C&D	Full Stack Development-I	20/12/2024 -12/04/2025

Skill Oriented Course-III

S.No.	Sem & Section	Title	Date (From -to)
1.	V SEM CSE-A,B,C&D	Cyber Security & Block chain	23/09/2024 - 26/09/2024, 28/09/2024 and 30/09/2024
		GenAI & Prompt Engineering	
		Django & Flask Framework	

Skill Oriented Course-IV

S.No.	Sem & Section	Title	Date (From -to)
1.	VI SEM CSE-A,B,C&D	React JS +Next JS + SQL	03/03/2025 -08/03/2025
		Spring Boot+MySQL	

Skill Oriented Course-V

S.No.	Sem & Section	Title	Date (From -to)
1.	VII SEM CSE-A,B,C&D	Aptitude & Verbal	06/08/2024 - 10/08/2024

Python Programming

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

CO1: Illustrate basic concepts and control structures in python Programming. **(K3)**
CO2: Demonstrate functions and packages. **(K3)**
CO3: Construct python programs using structured data types. **(K3)**
CO4: Develop programs on Files, Exception handling and OOPs Concepts. **(K3)**
CO5: Construct programs for Data Analysis using Num Py and Pandas. **(K3)**

Course Contents:

- Basics of Python Programming
- Control Flow Statements
- Functions, Strings and Lists
- Python Data Structures : Dictionaries, Tuples , and Sets
- Files and Object-Oriented Programming
- Introduction to Data Science

Full Stack Development-I

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

CO1: Illustrate HTML elements and their attributes for designing static web pages. **(K3)**

CO2: Apply appropriate CSS styles to HTML elements. **(K3)**

CO3: Demonstrate JavaScript Pre-defined and User-Defined Objects. **(K3)**

CO4: Develop dynamic web pages and validate forms using JavaScript. **(K3)**

Experiments covering the Topics:

- Lists, Links and Images
- HTML Tables, Forms and Frames
- HTML 5 and Cascading Style Sheets, Types of CSS
- Selector forms
- CSS with Color, Background, Font, Text and CSS Box Model
- Applying JavaScript - internal and external, I/O, Type Conversion
- JavaScript Conditional Statements and Loops, Pre-defined and User-defined Objects
- JavaScript Functions and Events
- Node.js

Cyber Security & Block chain

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

CO1: Understand core cyber security concepts and identify various cyber threats.. **(K2)**
CO2: Implement network and application security measures to mitigate vulnerabilities. **(K3)**
CO3: Discuss cryptographic principles and their role in blockchain technology. **(K3)**
CO4: Develop and deploy smart contracts and decentralized applications on blockchain platforms. **(K3)**
CO5: Analyze and apply security auditing and risk management techniques to ensure compliance with industry standards. **(K3)**

Course Contents:

- Introduction to Cyber Security
- Network Security & Wireless Networks
- Secure Software Development & Web Application Security
- Introduction to Cryptography
- Cyber Security Practices
- Hands-on Session Cyber Security
- Introduction to Blockchain
- Cryptography and Consensus Algorithms
- Smart Contracts and Decentralized Applications
- Hands-on Session Smart Contracts and Decentralized Applications
- Blockchain Platforms and Frameworks
- Use cases on Blockchain

Gen AI & Prompt Engineering

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

CO1: Explain the fundamentals of Generative AI and Machine Learning. **(K2)**
CO2: Apply classification algorithms and evaluate their performance. **(K3)**
CO3: Design and implement neural networks for deep learning tasks. **(K3)**
CO4: Produce images from text prompts using prebuilt text-to-image models. **(K3)**
CO5: Develop a basic chatbot using large language models. **(K3)**

Course Contents:

- Introduction to Gen AI
- Introduction to ML
- Performance Metrics in ML
- ML with Hands-on
- Deep Learning Overview
- ANN with Hands-on
- CNN with Hands-on
- RNN with Hands-on
- NLP Fundamentals
- NLP Hands-on
- GANs with Hands-on
- AEs with Hands-on
- Exploration of Hugging Face
- Pre-built Text-to-Image Models
- Hands-on Activity
- LLMs
- Prompt Engineering
- Custom Chatbot using Hugging Face Text-to-Text Models
- Custom Chatbot using APIs

Django & Flask Framework

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

CO1: Explain Django and Flask project structure and workflow. **(K2)**
CO2: Demonstrate Django & Flask Templates. **(K3)**
CO3: Illustrate Django ORM & SQLAlchemy ORM. **(K3)**
CO4: Develop User Authentication and Authorization in Django and Flask. **(K3)**
CO5: Demonstrate Django REST Framework and Flask Extensions. **(K3)**

Course Contents:

- Introduction to Flask Framework
- Introduction to Django Framework
- Hands-on with Django Framework
- Hands-on with Flask Framework
- Django Models and Database Integration
- Working with Databases in Flask
- Performing CRUD Operations
- Working with Django Forms
- User Authentication and Authorization
- User Authentication in Flask
- Flask Extensions and API Development
- Django REST Framework and Deployment

React JS +Next JS + SQL

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

CO1: Explain the fundamental concepts of React JS, Next JS, and SQL including component structure, routing mechanisms, and database operations. **(K2)**

CO2: Demonstrate the use of React hooks, Next.js data fetching methods, and SQL queries for data retrieval and manipulation.. **(K3)**

CO3: Apply state management, routing, and component-based design to develop responsive user interfaces using React JS and Next JS. **(K3)**

CO4: Develop a full-stack application integrating React/Next.js frontend with a SQL-based backend, using appropriate API endpoints.. **(K3)**

CO5: Apply client-side and server-side data fetching techniques using Fetch API, SWR, and Next.js methods to interact with backend services and databases.. **(K3)**

React JS:

- **Introduction to React & Setup**
  - What is React? Why use it?
  - Setting up a development environment (Node.js, npm/yarn, VS Code).
  - Creating a basic React application (create-react-app).
  - Understanding JSX.
- **Components & Props**
  - Functional vs. Class components.
  - Props: Passing data between components.
  - Component composition.
  - Basic Styling.
- **State & Lifecycle**
  - Managing component state using useState.
  - Understanding component lifecycle methods (useEffect).
  - Event handling.
- **Lists & Keys, Conditional Rendering**
  - Rendering lists of data.
  - Using keys for efficient rendering.
  - Conditional rendering based on state.
  - Basic forms.
- **Hooks & Context API**
  - Deep dive into common hooks (useState, useEffect, useContext).
  - Using the Context API for state management.
  - Custom hooks.
- **Routing (React Router)**
  - Setting up React Router.
  - Creating routes and navigation.
  - Passing parameters in routes.

Next JS

- **Introduction to Next.js & Setup**
  - What is Next.js? Why use it?
  - Setting up a Next.js project.
  - Understanding pages and routing.
  - File-based routing.
- **Data Fetching (getServerSideProps, getStaticProps)**
  - Server-side rendering (SSR) and static site generation (SSG).
  - Fetching data using getServerSideProps and getStaticProps.
  - Understanding when to use each approach.
- **API Routes**
  - Creating API routes in Next.js.
  - Handling HTTP requests.
  - Integrating with backend services.
- **Client-Side Data Fetching (SWR or Fetch)**
  - client side data fetching inside of nextjs.
  - using SWR or standard fetch.
- **Styling in Next.js**
  - CSS modules.
  - Styled components.
  - Tailwind CSS.
- **Deployment**
  - Deploying Next.js applications to platforms like Vercel or Netlify.

SQL

- **Introduction to SQL & Database Setup**
  - Relational databases and SQL basics.
  - Setting up a local SQL database (SQLite or PostgreSQL).
  - Creating tables and defining data types.
- **Basic SQL Queries (SELECT, WHERE, ORDER BY)**
  - Retrieving data using SELECT statements.
  - Filtering data using WHERE clauses.
  - Sorting data using ORDER BY.
- **Data Manipulation (INSERT, UPDATE, DELETE)**
  - Inserting new data into tables.
  - Updating existing data.
  - Deleting data.
- **Joins & Relationships**
  - Understanding table relationships (one-to-one, one-to-many, many-to-many).
  - Joining tables using INNER JOIN, LEFT JOIN, etc.

- **Connecting React/Next.js to SQL**
  - Using Node.js as a backend.
  - Connecting Node.js to a SQL database.
  - Creating API endpoints to interact with the database.
- **Project: Building a Full-Stack Application**
  - Putting it all together: building a simple application that uses React/Next.js for the frontend and SQL for the backend.

Spring Boot+MySQL

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

CO1: Explain the fundamental concepts of Spring Boot, dependency injection, and the architecture of RESTful APIs. **(K2)**

CO2: Describe how Spring Boot integrates with MySQL using Spring Data JPA for data persistence and database operations.. **(K2)**

CO3: Develop RESTful web services using Spring Boot, handling HTTP requests and performing CRUD operations with MySQL.. **(K3)**

CO4: Apply advanced features like entity relationships, custom queries, validation, and exception handling in a Spring Boot application.**K3)**

CO5: Apply a secure, tested, and documented RESTful API application by integrating Spring Security, Swagger, and testing frameworks like JUnit and Mockito. **(K3)**

- **Introduction to Spring Boot & Setup**
  - What is Spring Boot? Why use it?
  - Setting up a development environment (Java, Maven/Gradle, IntelliJ IDEA/Eclipse).
  - Creating a basic Spring Boot application.
- **Spring Boot Basics (Dependency Injection, Annotations)**
  - Understanding dependency injection.
  - Using Spring Boot annotations (@Component, @Service, @Controller, etc.).
  - Creating beans.
- **RESTful APIs (Controllers, Request/Response Handling)**
  - Building RESTful APIs using Spring MVC.
  - Handling HTTP requests (GET, POST, PUT, DELETE).
  - Serializing and deserializing JSON.
- **Data Persistence with Spring Data JPA**
  - Introduction to JPA and Hibernate.
  - Setting up Spring Data JPA.
  - Defining entities and repositories.
- **MySQL Database Setup & Integration**
  - Setting up a local MySQL database.
  - Connecting Spring Boot to MySQL.
  - Configuring database properties.
- **CRUD Operations with Spring Data JPA & MySQL**
  - Implementing CRUD operations (Create, Read, Update, Delete) using Spring Data JPA.
  - Testing API endpoints.
- **Advanced JPA (Relationships, Queries)**
  - Managing entity relationships (one-to-one, one-to-many, many-to-many).
  - Writing custom queries using JPQL or native SQL.
- **Exception Handling & Validation**
  - Handling exceptions in RESTful APIs.
  - Validating request data.
  - Global exception handling.

- **Security (Spring Security)**
  - Introduction to Spring Security.
  - Implementing basic authentication and authorization.
  - JWT Authentication.
- **Testing (JUnit, Mockito)**
  - Writing unit tests for Spring Boot applications.
  - Using Mockito for mocking dependencies.
  - Integration testing.
- **Logging & Monitoring**
  - Implementing logging in Spring Boot.
  - Basic monitoring.
- **API Documentation (Swagger/OpenAPI)**
  - Generating API documentation using Swagger/OpenAPI.
- **Project: Building a RESTful API Application**
  - Combining all skills learned into a complete project.

Aptitude & Verbal

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

- Use their Logical thinking and analytical abilities to solve quantitative aptitude questions form Company specific and other competitive tests. **[K3]**
- Solve questions relative to Time & distance and time & work etc. from company specific. **[K2]**
- Solve puzzle related questions for any type of competitive tests. [**K2]**
- Practice the Aptitude Round Clearing ability in interview Process. **[K3]**
- Analyze discourse markers to speak clearly on a specific topic in informal discussions. **[K3]**

Course Contents:

Aptitude:

- Number System
- Percentages
- Ration and proposition
- Averages
- Time and work
- Time and Distance
- Profit and loss
- Simple Interest and Compound Interest
- Permutations and combination
- Probability
- Data Analysis
- Series
- Analogy
- Directions and classifications
- Blood Relations
- Syllogism
- Seating
- Circular
- comparisons
- Coding and Decoding
- Clock and calendar

Verbal:

- Grammar, Tenses
- Preposition & Articles
- Speech & Voices
- Reading Comprehension
- Sentence Correction and Sentence Selection
- Sentence Arrangement
- Spotting Error
- Synonyms & Antonyms