**Web and Software Development 3 Months Course Outline**

|  |  |  |  |
| --- | --- | --- | --- |
| **TIME** | **DETAILS** | **OUTCOMES** | **SIGNATURE** |
| **Week 1** | Introduction to Web Development: Basics of HTML, CSS, and JavaScript. Setting up environment. | Understand the structure of a web page and basic web technologies. |  |
|  | Install Node.js, npm, and a code editor (e.g., VSCode). | Ability to set up and navigate a development environment. |  |
| **Week 2** | JavaScript Deep Dive: Variables, Data Types, Functions, and Loops. | Strong foundation in JavaScript syntax and programming concepts. |  |
|  | Introduction to ES6+ features like let/const, arrow functions, template literals, etc. | Familiarity with modern JavaScript features. |  |
| **Week 3** | DOM Manipulation: Working with DOM, Event Listeners, and Basic Interactions. | Ability to create dynamic web pages using JavaScript. |  |
|  | Introduction to Git and GitHub: Creating repositories and version control basics. | Basic version control knowledge for collaborative development. |  |
| **Week 4** | Basics of HTTP and REST APIs: Understanding requests, responses, and status codes. | Understanding of how web applications communicate over the internet. |  |
|  | Fetch API and basic data handling in JavaScript. | Ability to make API calls and process responses. |  |
| **Week 5** | Introduction to React: Functional components, JSX, and Props. | Build basic React components and structure simple applications. |  |
|  | Setting up a React project using Create React App (CRA) and Vite. | Hands-on experience setting up and running React applications. |  |
| **Week 6** | State Management in React: useState and useEffect hooks. | Manage component state and handle side effects in React applications. |  |
|  | Styling in React: CSS Modules, Styled Components, and inline styles. | Create visually appealing and responsive React applications. |  |
| **Week 7** | Introduction to Node.js and Express.js: Building RESTful APIs. | Develop and deploy basic backend services. |  |
|  | Working with Middleware, Routes, and Controllers. | Structure Express applications using best practices. |  |
| **Week 8** | MongoDB Basics: Introduction to NoSQL databases, CRUD operations. | Understand and work with MongoDB for data storage and retrieval. |  |
|  | Mongoose ODM: Schema creation and validations. | Effectively manage MongoDB collections using Mongoose. |  |
| **Week 9** | MERN Stack Integration: Connecting React frontend with Express and MongoDB backend. | Build a full-stack application with a functional backend and frontend. |  |
|  | Authentication: Implement JWT-based authentication. | Secure applications using authentication mechanisms. |  |
| **Week 10** | Advanced React: useContext, useReducer, and React Router for state and navigation. | Handle complex state management and navigation in React apps. |  |
|  | Testing React Applications: Jest and React Testing Library basics. | Write test cases to ensure the reliability of React applications. |  |
| **Week 11** | Advanced Backend: File uploads, error handling, and optimization techniques in Express. | Create robust and efficient server-side applications. |  |
|  | WebSockets and Real-time Communication with Socket.io. | Add real-time features like chat or live updates to applications. |  |
| **Week 12** | Deployment: Deploying applications using services like Heroku, Netlify, or Vercel. | Successfully deploy and maintain MERN stack applications. |  |
|  | Capstone Project: Build a complete MERN stack application. | Demonstrate the ability to create and deploy a full-stack application from scratch. |  |