Course Title: Web Development using MERN Stack

Course Overview:

The **MERN Stack** (MongoDB, Express.js, React.js, Node.js) is a popular technology stack for building full-stack web applications. This course will teach you how to develop modern, scalable, and interactive web applications using these technologies. You will learn how to connect MongoDB with Node.js, build RESTful APIs using Express.js, and create rich, interactive user interfaces with React.js. By the end of the course, you will be able to develop full-stack applications with seamless front-end and back-end integration.

Course Duration:

- **Duration:** 3-4 months
- **Mode of Delivery:** Online/Offline

Prerequisites:

- Basic understanding of JavaScript, HTML, and CSS.
- Familiarity with web development concepts.
- Experience with Node.js and React.js is a plus but not mandatory.

Course Modules:

Module 1: Introduction to Web Development with MERN Stack

- What is the MERN Stack and Why Use It?
- Overview of MongoDB, Express.js, React.js, and Node.js
- Setting Up the MERN Development Environment
- Introduction to JavaScript ES6 and modern JavaScript features
- Basic Web Development Concepts: HTTP, APIs, and Web Servers

Module 2: Setting Up the Development Environment

- Installing Node.js and npm
- Installing MongoDB and Setting Up the Database
- Introduction to Visual Studio Code (VS Code) as the IDE
- Installing Express.js and React.js

- Setting Up and Using Git for Version Control

Module 3: Introduction to MongoDB

- What is NoSQL and MongoDB?
- Setting Up and Configuring MongoDB
- Basic MongoDB Commands (CRUD Operations)
    - Create, Read, Update, Delete Documents
- Understanding Collections and Documents in MongoDB
- MongoDB Data Modeling and Schema Design

Module 4: Node.js and Express.js Basics

- Introduction to Node.js and its Advantages
- Setting Up a Simple Node.js Server
- Introduction to Express.js and Routing
- Handling HTTP Requests (GET, POST, PUT, DELETE)
- Middleware in Express.js
- Building RESTful APIs with Express.js
- Error Handling in Express.js

Module 5: Connecting MongoDB with Node.js

- Introduction to Mongoose (ODM for MongoDB)
- Defining Schemas and Models in Mongoose
- Connecting Node.js with MongoDB using Mongoose
- Performing CRUD Operations with MongoDB and Mongoose
- Handling Relationships and Data Validations in MongoDB
- Implementing Pagination and Sorting in MongoDB Queries

Module 6: React.js Fundamentals

- Introduction to React.js and its Role in Full-Stack Development
- Setting Up a React Project using Create React App
- Understanding Components, Props, and State in React
- React Lifecycle Methods and Hooks (useState, useEffect)

- Handling Events and Forms in React
- Managing State in React using Context API
- Introduction to React Router for Navigation

Module 7: Building a RESTful API with Express.js and Node.js

- Designing RESTful APIs (GET, POST, PUT, DELETE)
- Connecting React.js with the Back-End via Fetch API and Axios
- Implementing Authentication and Authorization with JWT (JSON Web Tokens)
- Secure Password Storage with bcrypt
- Error Handling in REST APIs
- Debugging and Logging with Node.js and Express.js

Module 8: React.js and State Management

- State Management with Redux
    - Setting up Redux in a React application
    - Actions, Reducers, and Store
    - Handling Asynchronous Actions with Redux Thunk
- Component Design and Reusability in React
- Component Styling with CSS Modules and Styled-Components

Module 9: Connecting Front-End (React) with Back-End (Node.js)

- Sending Data from React to Node.js and MongoDB
- Handling Data Display in React from MongoDB
- Managing Authentication in MERN Applications (Login, Logout)
- User Registration and Profile Management
- Session Management and JWT Token Storage (LocalStorage, SessionStorage)

Module 10: Advanced MERN Concepts and Optimization

- Performance Optimization in React
    - Lazy Loading of Components
    - Code Splitting and Dynamic Imports
    - Optimizing Rendering Performance with React.memo

- Handling File Uploads in Node.js and React
- Pagination and Infinite Scrolling in MERN Applications
- Testing MERN Applications (Jest, Mocha, Chai)
- Deploying MERN Applications to Production (Heroku, DigitalOcean, AWS)

Module 11: Final Project

- **Project 1: Task Management Application**
  Build a task management application where users can sign up, create tasks, update them, and mark them as completed.
- **Project 2: E-commerce Website**
  Develop an e-commerce website where users can browse products, add them to the cart, and make purchases using a payment gateway.
- **Project 3: Social Media Platform**
  Build a basic social media platform with user authentication, profiles, posts, and likes.

In these projects, you will apply your MERN Stack knowledge to build a fully functional web application that integrates the front-end and back-end seamlessly.

Course Resources:

1. **MongoDB Documentation** – The official MongoDB documentation for database queries, operations, and optimization.
2. **Express.js Documentation** – Official documentation for creating APIs and handling server-side routing in Node.js.
3. **React.js Documentation** – The official React documentation for building components and managing state.
4. **Node.js Documentation** – Official documentation for working with Node.js and building servers.
5. **Redux Documentation** – Official documentation for state management using Redux.
6. **Visual Studio Code** – Code editor with excellent support for JavaScript and MERN Stack development.
7. **MongoDB Compass** – A graphical tool to manage MongoDB databases.
8. **Postman** – API testing tool to test RESTful APIs created with Express.js.
9. **GitHub** – To manage and share your code repository.

Evaluation:

- **Assignments:** Weekly assignments on MERN Stack technologies and project work.
- **Midterm Project:** Build a small web application using Node.js, Express.js, MongoDB, and React.js.
- **Final Project:** Develop a complete full-stack MERN application that is functional and optimized.

Certification:

Upon successful completion of the course, students will receive a **Certificate of Completion**