

Node.js with Vue.js Essentials

3-days of Full-Stack development with MongoDB as the DB

FA-0007 · 3-days

Software Development

Introduction

This training course integrates Node.js and Vue.js, empowering developers to create robust, scalable backend services and dynamic, responsive front-end applications. Node.js, with its non-blocking, event-driven architecture, is ideal for building efficient server-side applications. Coupled with Vue.js, a progressive JavaScript framework for building user interfaces, developers can deliver seamless and engaging user experiences.

This course aims to provide a comprehensive understanding of both technologies, enabling participants to develop full-stack applications that are performant, maintainable, and scalable.

Learning outcomes

- Understand the fundamentals of Node.js and Vue.js.
- Set up development environments for Node.js and Vue.js on remote Ubuntu servers using VS Code.
- Develop and deploy server-side applications using Node.js.
- Build dynamic and responsive front-end applications with Vue.js.
- Implement asynchronous programming and handle asynchronous operations in Node.js.
- Create RESTful APIs with Express.js and integrate them with Vue.js.
- Connect and interact with databases (MongoDB) from a Node.js application.
- Handle real-time data with WebSockets.
- Implement error handling, logging, and debugging in Node.js applications.
- Deploy and scale full-stack applications using cloud platforms.

Prerequisites

- Basic knowledge of JavaScript.
- Familiarity with command-line interfaces.
- Basic understanding of web development concepts (HTTP, REST).
- Access to unrestricted internet
- Terminal / Console
- VS Code (with the SSH plugin installed)
- Access to a remote Ubuntu server and VS Code (will be provided)

Training outline

1. Introduction to Node.js and Vue.js

1. What is Node.js and why use it?
2. What is Vue.js and why use it?
3. History and evolution of Node.js and Vue.js
4. Advantages and use cases of Node.js and Vue.js in modern web development

2. Setting Up the Development Environment

1. Installing Node.js and npm on Ubuntu
2. Installing Vue CLI and creating a new Vue project
3. Configuring VS Code for remote development
4. Setting up SSH for remote server access
5. Basic terminal commands for managing Node.js and Vue.js

3. Node.js Fundamentals

1. Understanding the Node.js architecture
2. The event-driven, non-blocking I/O model
3. The Node.js process and the V8 JavaScript engine
4. Introduction to the Node.js REPL (Read-Eval-Print Loop)

4. Core Modules and npm

1. Overview of Node.js core modules

2. Working with the fs (File System) module
3. Using the http module to create a basic web server
4. Managing packages with npm
5. Creating and publishing a simple npm package

5. Building a Web Server with Express.js

1. Introduction to Express.js
2. Setting up an Express.js project
3. Routing and handling HTTP requests
4. Middleware in Express.js
5. Serving static files

6. Asynchronous Programming in Node.js

1. Callbacks, Promises, and Async/Await
2. Managing asynchronous operations
3. Error handling in asynchronous code
4. Using the util module for promisification

7. Interacting with Databases

1. Introduction to MongoDB and Mongoose
2. Setting up MongoDB on Ubuntu
3. Connecting a Node.js application to MongoDB
4. CRUD operations with Mongoose
5. Schema design and validation

8. Building RESTful APIs

1. Principles of RESTful API design
2. Creating RESTful routes in Express.js
3. Handling request data and responses
4. Implementing authentication and authorization
5. API versioning and documentation

9. Introduction to Vue.js

1. The Vue.js ecosystem and core concepts
2. Setting up a new Vue project with Vue CLI
3. Understanding the Vue instance
4. The Vue component system
5. Data binding and the Vue template syntax

10. Building User Interfaces with Vue.js

1. Creating and organizing Vue components
2. Handling user input and events
3. Using Vue directives for conditional rendering and list rendering
4. Managing state and props in Vue components
5. Using Vue Router for navigation and routing

11. State Management with Vuex

1. Introduction to Vuex
2. Setting up a Vuex store
3. State, getters, mutations, and actions
4. Using Vuex in Vue components
5. Advanced state management patterns

12. Integrating Node.js and Vue.js

1. Setting up communication between the front end and back end
2. Making API requests from Vue.js using Axios
3. Handling authentication and sessions
4. Implementing real-time updates with WebSockets and Socket.io
5. Building a full-stack application: a practical example

13. Real-Time Communication with WebSockets

1. Introduction to WebSockets
2. Setting up WebSocket communication in Node.js
3. Using Socket.io for real-time web applications

4. Handling events and broadcasting messages
5. Building a real-time chat application

14. Error Handling and Debugging

1. Common sources of errors in Node.js and Vue.js applications
2. Implementing robust error handling
3. Logging with winston and morgan
4. Debugging Node.js applications with VS Code
5. Debugging Vue.js applications with Vue Devtools

15. Deployment and Scalability

1. Preparing a Node.js and Vue.js application for production
2. Deploying full-stack applications on cloud platforms (e.g., Digital Ocean)
3. Using PM2 for process management and monitoring
4. Load balancing and scaling Node.js applications
5. Best practices for performance optimization

16. Project: Building a Complete Full-Stack Application

1. Planning and designing the application
2. Implementing the backend with Express.js and MongoDB
3. Developing the frontend with Vue.js
4. Integrating real-time features with WebSockets
5. Testing and debugging the application
6. Deploying the application to a remote server

This comprehensive training outline ensures that participants gain hands-on experience and a thorough understanding of both Node.js and Vue.js, preparing them to build robust, scalable, and efficient full-stack web applications.