Web Applications with Vue.js, Node.js, and MongoDB

Course Number: WEB-APPS

Duration: 5 days

Overview

Course Description

This web applications training gives learners a full introduction to modern full-stack development, including syntax, conventions, and best practices of modern JavaScript development. The course thoroughly covers Vue.js, Node.js, and MongoDB. Through a series of hands-on exercises and demonstrations, students learn how to develop and test complete Vue/Node/Mongo applications. After an optional overview of JavaScript, learners dive into the fundamentals of using Node and Express to build an API for the Vue frontend application with a MongoDB database. Learners discover how to connect the Vue frontend with the Node.js backend.

Skills Gained

After taking this course, students will be able to:

- Install, configure, and use modern web tooling
- Create test suites for Vue
- Understand what Vue.js is and what problem it solves

- Know the basic architecture of a Vue.js application
- Build a Single Page Application with Vue Router
- Use Pinia for maintaining state in a Vue.js application
- Apply Vue best practices
- Code web applications and RESTful APIs with Node.js
- Connect Node.js to MongoDB

Prerequisites

All students should have experience with HTML5, CSS3, and JavaScript.

Audience

Course Details

Basic JavaScript (optional)

- How JavaScript Works
- JavaScript Syntax
- JavaScript Data Types
- JavaScript Primitives
- Variables and Arrays
- Variable Scoping with const and let
- JavaScript Operators
- Template Literals
- Functions
- Arrow Functions
- JavaScript Objects
- Prototypal Inheritance

Advanced JavaScript (optional)

- Default Parameter Handling
- Rest Parameter
- Spread Operator
- Tagged Template Literals
- Enhanced Object Properties
- Property Shorthand
- Method notation
- Array Matching

- Object Matching
- Symbol Primitive
- For-Of Operator
- Creating and Consuming Generator Functions
- Class Definition
- Class Inheritance
- Understanding this
- Array.map()
- Array.filter()
- Array.reduce()
- Promises
- Async / Await

TypeScript Basics

Vue QuickStart

- What is Vue.js?
- Vue vs React and Angular
- Virtual DOM
- What's New in Vue 3
- Two ways to write Vue Components
- Code Editors and IDEs
- Volar extension
- Lab 1: Vue 3 Quick Start
- Lab 2: Your First Component
- Lab 3: Create More Components
- Lab 4: Testing Vue
- Lab 5: Manual In-Browser Testing and Debugging
- Getting Started with Vue.Js
- Basic Vue.Js Features
- Creating and Mounting a Vue App
- Configuring an App
- Which Style Should You Use?
- Vue Templates
- Vue.js Directives

- Loops and Lists
- Conditional Rendering
- Lab 6: Static Version
- Using Filters
- Binding HTML Classes
- Adding Styles Conditionally
- Binding Styles
- Lab 7: Styling Vue Components
- Computed Properties
- Vue State
- Vue with TypeScript
- Lab 09: Methods and State + TypeScript
- Event Handling
- Lab 10: Events
- Watchers
- Vue Instance Lifecycle
- Lab 11: Component Lifecycle
- Vue and Forms
- Lab 12: Forms
- Vue Components
- Content Distribution with Slots
- Lab 13: Slots
- Loading Your Components Asynchronously
- Lab 14: Composition API
- Introducing Pinia
- Lab 15: Pinia
- Single-Page Applications
- Lab 16: Routing
- Lab 17: AJAX with Pinia
- Transitions and Animations
- Custom Transition Classes
- Optional Lab: Transitions and Animation

Intro to Node.js

- What is Node.js?
- How Does Node.js Work?
- Blocking code
- Non-Blocking code
- V8 JavaScript Engine
- Node's REPL
- Running a Node.js program
- Lab 18: Basic Setup
- Callbacks
- Modules Overview
- CommonJS Example
- Using Modules
- Modules vs. Packages
- Sources of Modules
- Node's Core Modules
- Buffer Objects
- Modularizing Your Code
- Returning Values from Modules
- Using a Local Module
- Optional Lab: Creating Modules
- ES6 Modules
- events and Streams
- Non-blocking with Events
- Events
- Node Stream Objects
- Optional Lab: Working with Streams
- The pipe method
- Optional Lab: Piping Between Streams
- The process Object
- Command Line Arguments
- Optional Lab: The process object
- Understanding Callbacks
- Using Node's Error Convention

- Node on the Web
- Using Express
- Lab 19: Routing
- Lab 20: More Routing
- Lab 21: The req and res Objects
- Lab 22: Implementing the GET method
- Lab 23: Implementing the POST method
- Lab 24: Implementing the DELETE method
- Lab 25: Implementing the PUT method
- Lab 26: Middleware
- Database Access with Node.js
- Lab 27: Vue and Mongo

Conclusion