

Full-Stack Web Development: Build 15 Projects

Master HTML, CSS, JavaScript, Node.js, and deploy real-world apps. Suited for beginners to advanced developers.

Programming Instructor: David Clarke • 5 sections • 54 lessons • 18.50h total

[View this course online](#) [Browse all courses](#)

About this course

Many courses teach web development theory without showing you how to build actual applications. This leaves learners stuck when facing real projects. This course fills that gap by focusing on practical skills from day one.

You'll start with how the internet works and set up your tools, then move through HTML, CSS, and JavaScript. We cover modern ES6+ features, back-end development with Node.js and databases, and full-stack deployment. Includes video lessons, code-along examples, and case studies from 15 different projects.

Course curriculum

Foundations of the Web

- **How the Internet Works: Browsers, Servers, and DNS (12m)**
We'll break down the client-server model, how DNS resolves domains, and the role of HTTP/HTTPS in data transfer.
- **Setting Up Your Development Environment (15m)**
Step-by-step guide to installing VS Code, Git, and configuring browser DevTools for efficient coding and debugging.
- **HTML5 Semantic Elements and Document Structure (18m)**
Learn to use tags like header, nav, and article to create well-structured, accessible web pages.
- **Creating Accessible Forms and Tables in HTML (20m)**
Build forms with proper labels, inputs, and validation, plus tables for displaying data clearly.
- **Introduction to Browser Developer Tools (10m)**
Explore DevTools to inspect elements, debug JavaScript, and monitor network requests in real time.
- **Project: Personal Portfolio Page with Semantic HTML (25m)**
Apply HTML skills to build a portfolio site showcasing your work, with a focus on clean, semantic markup.
- **Web Standards and Best Practices Review (8m)**
Discuss accessibility guidelines, SEO basics, and coding standards to ensure your sites are professional.

Styling and Responsive Design

- **CSS3 Fundamentals: Selectors, Properties, and the Box Model (22m)**
Master the core concepts of CSS, including how elements are sized and spaced with the box model.
- **Mastering Flexbox for Layout Alignment (18m)**
Use Flexbox to create flexible, aligned layouts for headers, footers, and card-based designs.
- **CSS Grid for Complex Page Layouts (20m)**
Build two-dimensional layouts with Grid, ideal for dashboards and multi-column designs.
- **Media Queries and Mobile-First Responsive Design (15m)**
Adapt your designs to different screen sizes using media queries and a mobile-first approach.
- **Custom Properties (CSS Variables) for Theming (12m)**
Create reusable color and font variables to easily switch themes and maintain consistency.
- **CSS Animations and Transitions for Interactive Elements (16m)**
Add motion to buttons, menus, and other elements with keyframes and transition properties.
- **Project: Responsive Landing Page (30m)**
Combine CSS skills to build a landing page that looks great on mobile, tablet, and desktop.
- **Advanced CSS Techniques: Filters, Blend Modes, and Shadows (14m)**
Explore visual effects like drop shadows, image filters, and blend modes for creative designs.
- **Project: Animated CSS Art Piece (25m)**
Create a piece of art using pure CSS, practicing animations and creative styling.

JavaScript and ES6+ Programming

- **JavaScript Basics: Variables, Data Types, and Operators (18m)**
Start with core JavaScript concepts, including let, const, strings, numbers, and basic operations.
- **Functions, Loops, and Conditional Statements (20m)**
Write reusable code with functions, loop through data, and control logic with if/else statements.
- **DOM Manipulation: Selecting and Modifying HTML Elements (22m)**
Use JavaScript to dynamically change page content, styles, and structure in response to user actions.
- **Event Handling and Interactive User Interfaces (15m)**
Attach event listeners to elements for clicks, inputs, and other interactions to build responsive UIs.
- **Introduction to ES6: Arrow Functions, Destructuring, and Template Literals (18m)**
Modern JavaScript features that make code cleaner and easier to read, with practical examples.
- **ES6 Modules and Async Programming with Promises (20m)**
Organize code into modules and handle asynchronous operations like API calls using Promises.
- **Working with Async/Await and Error Handling (16m)**
Simplify async code with async/await and implement try/catch blocks for robust error handling.
- **Local Storage for Client-Side Data Persistence (12m)**
Store user data in the browser to create persistent experiences without a back-end.
- **Project: Interactive Quiz Application (32m)**
Build a quiz app that scores answers, tracks progress, and provides feedback using JavaScript.
- **Project: Dynamic To-Do List with Local Storage (28m)**
Create a to-do app where tasks are added, edited, and saved to local storage for persistence.
- **Project: Real-Time Weather Dashboard with API (35m)**
Fetch data from a public weather API and display it dynamically, practicing async JavaScript.

Back-End Development with Node.js

- **Introduction to Node.js and the npm Ecosystem (15m)**
Learn what Node.js is, how to install it, and use npm to manage project dependencies.
- **Setting Up an Express.js Server for API Development (18m)**
Create a basic server with Express, define routes, and listen for incoming requests.
- **Building RESTful APIs: Routing and HTTP Methods (20m)**
Design API endpoints with GET, POST, PUT, and DELETE methods to handle data operations.
- **Middleware and the MVC Pattern in Express.js (16m)**
Use middleware for logging, auth, and error handling, and structure apps with Model-View-Controller.
- **Introduction to MongoDB and Data Modeling with Mongoose (22m)**
Set up MongoDB, define schemas with Mongoose, and understand NoSQL data structures.
- **Performing CRUD Operations on a MongoDB Database (18m)**
Implement create, read, update, and delete functions for your database collections.
- **User Authentication with JWT Tokens and Password Hashing (24m)**
Secure user accounts by hashing passwords with bcrypt and issuing JSON Web Tokens.
- **Environment Variables and Security Best Practices (12m)**
Store sensitive data like API keys in .env files and follow security guidelines to protect your app.

- **Handling CORS and API Error Responses (10m)**
Configure Cross-Origin Resource Sharing and send consistent error messages from your API.
- **Project: Blog API with Full CRUD Functionality (30m)**
Build a REST API for a blog, including routes for posts, comments, and user interactions.
- **Project: User Authentication System (28m)**
Implement signup, login, and protected routes using JWT and secure password storage.
- **Real-Time Communication with Socket.io (20m)**
Set up Socket.io to enable real-time, bidirectional communication between server and clients.
- **Project: Real-Time Chat Application with Socket.io (35m)**
Create a chat app where users can join rooms and send messages that appear instantly.

Full-Stack Integration and Deployment

- **Connecting Front-End Apps to Back-End APIs (18m)**
Use Fetch and Axios in your front-end code to send requests and handle responses from your API.
- **Full-Stack Architecture Patterns and Project Structure (15m)**
Organize a full-stack project with separate folders for client and server, and manage routing.
- **Modern Tooling: Vite for Fast Front-End Development (12m)**
Set up Vite as a build tool for quick hot-reloading and optimized production builds.
- **Testing JavaScript Applications with Jest (20m)**
Write unit and integration tests for your code to catch bugs early and ensure reliability.
- **Continuous Integration and Deployment (CI/CD) Basics (16m)**
Automate testing and deployment with GitHub Actions or similar tools for streamlined workflows.
- **Building Progressive Web Apps for Offline Access (22m)**
Add service workers and caching strategies to make your app work offline on mobile devices.
- **Mobile-Responsive Best Practices for 2026 Web Development (14m)**
Adapt layouts and interactions for the latest devices, considering touch and performance.
- **Containerizing Applications with Docker (25m)**
Create Dockerfiles and docker-compose files to package your app for consistent environments.
- **Deploying to Cloud Platforms (18m)**
Deploy your app to platforms like Vercel, Netlify, or AWS with step-by-step guidance.
- **Domain Setup and SSL Certificates for Production (10m)**
Connect a custom domain and enable HTTPS with SSL certificates for secure access.
- **Capstone Project: E-Commerce Store (40m)**
Build an online store with product listings, cart, checkout, and user accounts, integrating all skills.
- **Capstone Project: Social Media Dashboard (38m)**
Create a dashboard for managing posts, likes, and comments with real-time updates.
- **Capstone Project: Project Management Tool (42m)**
Develop a tool for task boards, team collaboration, and progress tracking with a database.
- **Capstone Project: Recipe-Sharing Platform (35m)**
Build a platform where users can upload, search, and save recipes, with user authentication.

Discover our blog

Learn more with free articles written by our professors and guest experts.

Latest article: **[Flight Attendant Preparation: What to Study Before You Apply](#)**

[Visit the blog](#) [All courses](#) [This course](#)