Here's a comprehensive 6-month curriculum for mastering JavaScript and launching a live website. The plan assumes you'll spend around 10-15 hours per week on learning and building.

**Month 1: JavaScript Fundamentals**

1. **JavaScript Syntax & Data Types:**
   * Variables (let, const, var)
   * Data types (strings, numbers, booleans, arrays, objects)
   * Operators and expressions
   * Control flow (if-else, switch, loops)
   * Basic functions and scope
2. **DOM Manipulation & Event Handling:**
   * Querying DOM elements (getElementById, querySelector, etc.)
   * Modifying DOM elements (text, attributes, styles)
   * Event listeners (click, keyup, submit, etc.)
   * **Basic form handling and validation**
3. **Project 1: Interactive Quiz**
   * Build a simple interactive quiz to practice DOM manipulation and events.

**Month 2: Intermediate JavaScript**

1. **Advanced Functions & Objects:**
   * Higher-order functions
   * Closures and scopes
   * **this keyword and call/apply/bind**
   * **Object-oriented JavaScript (constructors, classes)**
2. **Asynchronous JavaScript:**
   * Promises, async/await
   * Fetch API for HTTP requests
   * Handling errors and response status codes
3. **Project 2: To-Do List with Fetch API**
   * Create a to-do list that fetches tasks from an external API.

**Month 3: JavaScript in the Browser**

1. **Introduction to Web APIs:**
   * Local storage, session storage
   * Geolocation API
   * Web workers (if needed)
2. **Version Control & Git Basics:**
   * Git fundamentals (commit, push, pull, branches)
   * Setting up GitHub repository
   * Collaborating with others (optional)
3. **Project 3: Build a Personal Portfolio**
   * Create a static portfolio site to showcase your projects using HTML, CSS, and JavaScript.

**Month 4: Frontend Frameworks and Tooling**

1. **Introduction to React.js (or an alternative like Vue.js):**
   * Components and JSX
   * State and Props
   * React lifecycle methods
   * Hooks (useState, useEffect)
   * Basic routing using React Router
2. **JavaScript Build Tools:**
   * Introduction to npm, Webpack, and Babel
   * Package management and versioning
3. **Project 4: Build a Single Page Application (SPA)**
   * Use React (or another framework) to build a simple app like a weather dashboard.

**Month 5: Backend with Node.js and Databases**

1. **Introduction to Node.js:**
   * Basic server setup with Express.js
   * Handling routes and middleware
   * Working with RESTful APIs
2. **Databases:**
   * Introduction to MongoDB (or SQL databases)
   * Basic CRUD operations
   * Connecting a database to your Node.js app
3. **Project 5: Full-Stack Application**
   * Create a basic CRUD application (e.g., a blog) with a Node.js backend, MongoDB, and a React frontend.

**Month 6: Deployment, Testing, and Optimization**

1. **Testing in JavaScript:**
   * Unit testing with Jest or Mocha
   * Integration and end-to-end testing
2. **Deployment & Hosting:**
   * Setting up a production server with Heroku, Netlify, or Vercel
   * Setting up a custom domain and HTTPS
   * Continuous deployment (using GitHub Actions or similar)
3. **Optimization & Security:**
   * Code minification and bundling
   * Lazy loading and performance optimization
   * Basic security (CORS, authentication with JWT)
4. **Final Project: Launch Your Website**
   * Polish your full-stack project and deploy it.
   * Ensure everything works (API, database, frontend).
   * Share your website with friends or on social media!