Web Programming

Igor Dimitrov

2024-05-17

Table of contents

Preface

This is a Quarto book.

To learn more about Quarto books visit https://quarto.org/docs/books.

1 Reading List

Phase 1: Fundamentals

1. Core HTML, CSS, and Intro JS Book

- *Fundamentals of Web Development* – Randy Connolly & Ricardo Hoar Covers HTML5, CSS3, JavaScript basics, HTTP, forms, and more. Structured, modern, and comprehensive.

2. Core JavaScript Book

- *Modern JavaScript for the Impatient* – Cay Horstmann Modern ES6+ focused guide with practical examples. Best suited to your background and learning style.

3. Supplementary JavaScript References (optional)

- *Eloquent JavaScript* – Marijn Haverbeke Functional and example-driven alternate style with exercises.
- *JavaScript Cookbook* – Adam Scott Recipe-style reference for real-world JS problems. Useful during practice or projects.
- *Professional JavaScript for Web Developers* – Matt Frisbie Massive, modern reference. Suitable for deep dives and architectural clarity.
- *JavaScript: The Definitive Guide* – David Flanagan Authoritative reference text. Best used for looking up complex behavior, language features, or browser APIs.

4. Optional CSS Enrichment

- *CSS in Depth* – Keith J. Grant Excellent modern CSS resource. Goes beyond basic selectors to explore layout systems (Flexbox, Grid), cascade/specificity, and responsive design patterns.

5. Optional JavaScript Enrichment (Phase 1.5)

- *The Joy of JavaScript* – Luis Atencio Read after Horstmann. Helps refine expressive, idiomatic, and functional JavaScript style. Encourages declarative thinking and better code practices.

Phase 2: Frontend Frameworks (React Track)

Core Books

1. **Learning React** – Alex Banks & Eve Porcello Read fully. Covers components, JSX, props/state, hooks, effects, and modern React design patterns. Best read steadily over 2–3 weeks with small examples.
2. **React for Real** – Ludovico Fischer A more hands-on, project-focused book. Builds an actual app while explaining key design decisions. Use it to reinforce what you've learned and make React "real."

Optional Follow-up (later phases or parallel)

- **Vue.js: Up and Running** – Callum Macrae If you want to try a second frontend paradigm after React.

Phase 3: Backend with Node.js and Express

1. Core Backend Books (in recommended order)

- *Web Development with Node and Express* – Ethan Brown Primary backend learning resource. Covers Express fundamentals, routing, middleware, APIs, authentication, templating, and database integration.
- *Node.js Design Patterns* – Mario Casciaro, Luciano Mammino Intermediate-to-advanced guide to designing scalable Node.js applications. Focuses on asynchronous control flow, modular design, architecture patterns, and production-grade backend structure.

2. Optional Reading and Reference

- *Express in Action* – Evan Hahn Focused guide to Express. Useful for alternate explanations and practical reinforcement of core concepts from Ethan Brown.
- *Node.js in Action* – Mike Cantelon et al. Broader overview of Node.js development. Covers more than just Express. Use for general Node exploration and extended examples.

- *Node.js 8 the Right Way* – Jim Wilson Low-level Node programming using streams, buffers, and child processes. Systems-oriented; not necessary unless you're exploring Node beyond web development.

Phase 4: Full-Stack Integration

Core Learning Resource

- *MiniPost* — a custom full-stack project A minimal blog application with authentication, built using your chosen stack:
  - **Frontend**: React + Vite + Tailwind CSS
  - **Backend**: Node.js + Express + Apollo Server (GraphQL)
  - **Database**: PostgreSQL (via Docker) accessed through raw SQL (`pg`)
  - **Auth**: JWT (jsonwebtoken) and bcrypt for password hashing

Project Structure (by phases)

1. **Phase 1: Read-Only Public Blog**
   - Set up PostgreSQL with Docker
   - Create `posts` table and seed sample data
   - Build GraphQL backend (`Post` type + `posts` query)
   - Create frontend with Apollo Client to display posts
2. **Phase 2: Authentication System**
   - Add `users` table and mutations for `signup` and `login`
   - Hash passwords with `bcrypt`, issue JWTs
   - Store token client-side and attach to Apollo requests
3. **Phase 3: Authenticated Posting**
   - Add `author_id` to posts, restrict post creation to logged-in users
   - Implement `createPost` mutation and frontend form
   - Handle user-based permissions
4. **Phase 4: Polish and Enhancements**
   - Add timestamps, `editPost`, `deletePost` (author-only)
   - Add logout, validation, UI polish with Tailwind
   - Explore Apollo Client cache and optimistic updates

Optional Future Phases

- **User Profiles** (avatars, bio, personal pages)
- **Comments System**
- **Admin Panel**
- **Production Deployment** with Docker, Fly.io, Railway, or VPS

Phase 5: Deployment and Tooling

1. Recommended Books

- *Docker for Rails Developers* – Jack Moffitt Despite the Rails context, this book provides a clear and practical introduction to Docker for web applications. Covers Dockerfiles, `docker-compose`, volume management, environment separation, and deployment strategies. Recommended as a general-purpose Docker learning resource.
- *Web Security for Developers* – Malcolm McDonald Practical guide to securing web applications. Covers topics like XSS, CSRF, SQL injection, HTTPS, and secure token-based authentication. Useful reference while preparing your app for production.

2. Optional Advanced Reading

- *The Book of Kubernetes* – Alan Hohn In-depth guide to Kubernetes for container orchestration and scalable deployment. Best reserved for later exploration if you move toward DevOps or distributed systems. Not necessary for basic app deployment.

3. Archived / Not Applicable

- *Build Websites with Hugo* – Brian P. Hogan Focuses on static site generation with Hugo. Not applicable to your current full-stack app (React + Node). Archived for possible future use in static site or documentation projects.

Perfect. Here's your updated list, now integrated as **Phase 7** of your full web development roadmap:

Phase 6 – Deployment & Infrastructure (DevOps Essentials)

Goal: Learn how to confidently deploy, operate, and manage modern web applications using Linux, Docker, Kubernetes, and infrastructure as code (IaC).

Focus: Practical DevOps skills — from Linux troubleshooting to containerization and orchestration — tailored to support your full-stack projects (e.g. MiniPost).

DevOps Mindset & Principles

- *Grokking Continuous Delivery* – Christie Wilson Clear and modern guide to DevOps thinking, continuous delivery, and deployment pipelines.

  Continuous Delivery – Jez Humble & David Farley *(Reference only)* Foundational theory book on DevOps culture and release practices. Optional background reading for later.

Linux & Server Operations

- *DevOps Troubleshooting: Linux Server Best Practices* – Kyle Rankin Hands-on guide for diagnosing and fixing Linux server issues. Still relevant and reliable for learning system-level debugging.
- *Hands-On DevOps with Linux* – Alisson Machado de Menezes Practical walkthrough of modern Linux DevOps tools (systemd, ufw, Docker basics, SSH). Complements the above with updated commands.

  DevOps for the Desperate – Bradley Smith *(Optional)* Lightweight crash course for new sysadmins. Good for warm-up or quick reference.

Containers & Docker

- *Docker in Action* – Ian Miell & Aiden Hobson Sayers Deep and practical introduction to Docker, containerization, image building, and orchestration basics.

Orchestration & Kubernetes

- *Kubernetes: Up and Running* – Kelsey Hightower, Brendan Burns, Joe Beda Best introductory book to Kubernetes. Written by creators, well-balanced between concept and action.

Infrastructure as Code

- *Infrastructure as Code* – Kief Morris Strategic guide to automating infrastructure in a scalable and maintainable way. Focuses on principles and practices.
- *Terraform in Action* – Scott Winkler Hands-on guide to writing and managing Terraform configurations. Pairs well with the above for practical IaC experience.

Alternative Frameworks (Rails, Phoenix, Flask, Django)

Elixir + Phoenix

- *Introducing Elixir* – Simon St. Laurent A beginner-friendly guide to the Elixir language and functional programming fundamentals.
- *Programming Phoenix 1.4* – Chris McCord, Bruce Tate, José Valim Full-stack web development with Phoenix, including real-time apps via channels and LiveView.

Ruby on Rails

- *Ruby on Rails Tutorial (3rd Edition)* – Michael Hartl Beginner-friendly, project-based guide to building a Twitter-style app with Rails.
- *Agile Web Development with Rails 6* – Sam Ruby, Dave Thomas Classic full-stack Rails book. Covers the entire web dev cycle using Rails conventions.
- *The Rails 5 Way* – Obie Fernandez Comprehensive reference on Rails idioms, patterns, and best practices.
- *Modern Front-End Development for Rails* – Noel Rappin Focuses on integrating Rails with modern JS tooling (Webpack, Stimulus, React).
- *Crafting Rails Applications* – José Valim Internals of the Rails framework — for advanced readers who want to understand its architecture.
- *Rails AntiPatterns* – Chad Pytel & Tammer Saleh Practical guide to avoiding common mistakes in large Rails apps.

Python Web Frameworks

- *Flask Web Development (2nd Edition)* – Miguel Grinberg Definitive book on Flask. Builds a full-featured web app using SQLAlchemy, WTForms, authentication, REST APIs, and deployment.
- *Lightweight Django* – Julia Elman & Mark Lavin Teaches how to use Django in a modular fashion, often as a backend for JS-heavy frontends.

Creative Coding & Visual JavaScript (Optional Enrichment Track)

A focused, joy-driven track for exploring p5.js, interactive media, generative design, and canvas-based game development. This complements your main web development roadmap with creativity and experimentation.

1. Core Creative Coding with p5.js

- *Getting Started with p5.js* – Lauren McCarthy, Casey Reas, Ben Fry A hands-on, visual introduction to creative coding in JavaScript using p5.js. Ideal entry point.
- *The Nature of Code* – Daniel Shiffman Builds on p5.js to simulate motion, forces, particle systems, and autonomous agents. Fun and exploratory.
- *Generative Design with p5.js* – Benedikt Groß, Hartmut Bohnacker, Julia Laub, Claudius Lazzeroni Visual and expressive design principles implemented in p5.js. Best used after the basics.

2. JavaScript Game Development & Canvas

- *Build an HTML5 Game* – Karl Bunyan Practical guide to building a complete game in JavaScript. Covers sprites, collisions, and game loops.
- *HTML5 Canvas Cookbook* – Eric Rowell Task-based reference for drawing, animation, image manipulation, and interactivity using the canvas API.

3. Optional Supplements and References

- *HTML5 Game Development by Example* – Makzan Alternative project-oriented approach to HTML5 games.
- *2D Game Development: From Zero to Hero* – Daniele Penazzo Compilation of practical game dev techniques and concepts.
- *3D Game Programming for Kids* – Chris Strom Entry-level guide to 3D with playful examples. Optional if you're interested in 3D basics.

- *Supercharged JavaScript Graphics* – Raffaele Cecco Advanced techniques for canvas performance and visual effects. Optional deep dive.
- *JavaScript for Sound Artists* – Leonard & Turner Introduction to interactive sound using the Web Audio API.
- *An Artist's Guide to Programming* – Jim Parker Cross-disciplinary, conceptual approach to programming as an artistic medium.
- *Quests: Theory and Design in Games* – Jeff Howard Game design theory with a narrative focus. No programming, but inspiring if you're curious about story design.
- *The Art of WebAssembly* – Rick Battagline Intro to compiling to the web (via WASM). Optional advanced topic for performance-focused experimentation.