

# Full-Stack Beginner Roadmap: From Zero to Job-Ready

**Subtitle:** Step-by-step guide to becoming a full-stack developer, from basics to real-world projects.

**Website Name:** haas.dev

**Website Link:** <https://dev-roast-app.vercel.app>

---

## Introduction

Becoming a full-stack developer can feel overwhelming, but a clear roadmap makes it achievable. This guide shows beginners exactly what to learn, in what order, and how to apply it in projects to become job-ready.

---

## Step 1: Learn the Basics

- **Programming Language:** JavaScript or Python
  - **HTML & CSS:** Build static webpages
  - **Git & GitHub:** Version control basics
  - **Mini Exercise:** Build a simple personal webpage with HTML, CSS, and GitHub deployment
- 

## Step 2: Frontend Development

- **JavaScript:** DOM manipulation, events
  - **Frontend Framework:** React or Vue basics
  - **Mini Project:** To-Do App or Portfolio Website
- 

## Step 3: Backend Development

- **Backend Language/Framework:** Node.js + Express or Python + Django/Flask
  - **Databases:** SQL (PostgreSQL/MySQL) & NoSQL (MongoDB) basics
  - **Mini Project:** Simple CRUD app (e.g., Blog platform)
- 

## Step 4: Full-Stack Integration

- Connect frontend to backend via APIs
  - Fetch and send data from frontend to backend
  - Mini Project: Full-Stack To-Do App with database persistence
- 

## Step 5: Deployment & Hosting

- **Frontend:** Netlify, Vercel
  - **Backend:** Render, Railway, or Heroku
  - Deploy full-stack projects online for portfolio visibility
  - Mini Exercise: Deploy your full-stack project and test live functionality
- 

## Step 6: Build Portfolio Projects

- Portfolio Website with live demos
  - CRUD-based apps (Blog, Task Manager)
  - Interactive apps using APIs (Weather, News, Movies)
  - Mini Exercise: Document each project with README.md, screenshots, and live link
- 

## Step 7: Interview Preparation

- Practice DSA: Arrays, Strings, Hash Maps, Stacks, Queues
  - Review full-stack concepts: HTTP methods, APIs, database operations
  - Prepare project explanations and problem-solving approach
- 

## Key Takeaways

- Start with **basics** → **frontend** → **backend** → **full-stack projects**
  - Apply knowledge in **real-world projects**
  - Deploy projects and maintain **GitHub portfolio**
  - Practice coding and project explanation for interviews
- 

Visit [haas.dev](https://haas.dev) for full-stack tutorials, project guides, and beginner-friendly resources.

---

---

Website Name: haas.dev

Website Link: <https://dev-roast-app.vercel.app>