

# 6-Month Software Development Bootcamp

A structured Notion template for planning, running, and tracking a 24-week software development bootcamp (2 sessions per week).

---

## Bootcamp Overview

**Duration:** 6 Months (24 Weeks)

**Frequency:** 2 sessions per week

**Total Sessions:** 48

**Format:** Live, hands-on training

**Outcome:** Frontend + Backend skills, a deployed full-stack project, and a clear strategy for getting paid as a software developer.

---

## Learning Outcomes

By the end of this bootcamp, students will be able to:

- Build responsive frontend applications
  - Create and consume backend APIs
  - Work with databases and authentication
  - Deploy full-stack applications
  - Build a developer portfolio
  - Understand job, freelance, and income paths in software development
- 



## MONTH 1 — FRONTEND DEVELOPMENT FOUNDATIONS

**Goal:** Build strong frontend fundamentals and confidence in building user interfaces.

---

### Week 1 – Web & Programming Fundamentals

#### Session 1: Introduction to Software Development

- What frontend, backend, and full-stack development mean
- How the web works (browser, server, HTTP)
- Development tools setup
- VS Code

- Browser DevTools
- Git & GitHub (intro)

**Assignment:** Install tools and push first GitHub repository

---

## Session 2: HTML Fundamentals

- HTML structure and semantics
- Common tags (div, section, header, footer)
- Forms and inputs

**Mini Project:** Personal profile webpage

---

## Week 2 – CSS & Responsive Design

### Session 3: CSS Basics

- CSS selectors
- Box model
- Fonts, colors, spacing
- Flexbox fundamentals

**Assignment:** Style the profile page

---

### Session 4: Responsive Design

- Mobile-first design
- Media queries
- Flexbox & Grid layouts

**Mini Project:** Responsive landing page

---

## Week 3 – JavaScript Fundamentals

### Session 5: JavaScript Basics

- Variables and data types
- Functions
- Conditions and loops
- Introduction to the DOM

**Assignment:** Simple JavaScript exercises

---

## Session 6: Interactivity & Events

- DOM manipulation
- Event handling
- Form validation
- Debugging basics

**Mini Project:** Interactive to-do list

---

## Week 4 – Modern Frontend Workflow

### Session 7: React Fundamentals

- What React is and why it's used
  - Components
  - Props and state
  - Component-based thinking
- 

### Session 8: Data Fetching & Frontend Capstone

- Fetching data from APIs
- Project structure
- State management basics

**Frontend Capstone Project:**

- Simple dashboard / blog / expense tracker UI
- 



## MONTH 2 — BACKEND DEVELOPMENT FOUNDATIONS

**Goal:** Build backend services and connect them to frontend applications.

---

## Week 5 – Backend Basics

### Session 9: Introduction to Backend Development

- What happens on the server
- REST APIs explained
- Node.js and Express setup

---

## Session 10: API Development

- HTTP methods (GET, POST, PUT, DELETE)
- API routing
- Testing APIs with Postman

**Mini Project:** Simple REST API

---

## Week 6 – Databases & Data Modeling

### Session 11: Databases Explained

- SQL vs NoSQL
  - Database setup (PostgreSQL or MongoDB)
  - CRUD operations
- 

### Session 12: Data Relationships

- One-to-one and one-to-many relationships
- Backend validation

**Mini Project:** User + data storage API

---

## Week 7 – Authentication & Security

### Session 13: Authentication Basics

- Signup and login flows
  - Password hashing
  - JWT authentication
- 

### Session 14: Authorization & Security Patterns

- Roles and permissions
- Error handling
- Securing APIs

**Assignment:** Add auth to backend project

---

## Week 8 – Full-Stack Integration & Deployment

### Session 15: Deployment Basics

- Environment variables
  - Frontend deployment
  - Backend deployment
- 

### Session 16: Full-Stack Capstone

- Connect frontend to backend
- Final deployment

#### Full-Stack Capstone Project:

- Task manager/booking app
- 



## MONTH 3 — FULL-STACK CONSOLIDATION & REAL-WORLD PRACTICE

**Goal:** Deepen full-stack confidence, improve code quality, and prepare students for advanced topics.

---

## Week 9 – Full-Stack Patterns & Best Practices

### Session 17: Code Quality & Project Structure

- Writing readable and maintainable code
  - Folder structures and naming conventions
  - Refactoring basics
- 

### Session 18: Working With Existing Codebases

- Reading unfamiliar code
  - Debugging strategies
  - Incremental improvements
-

## **Week 10 – APIs, Data & Edge Cases**

### **Session 19: Advanced API Usage**

- Error states and edge cases
  - Defensive programming
- 

### **Session 20: Data Consistency & Validation**

- Frontend vs backend validation
  - Handling failures gracefully
- 

## **Week 11 – Collaboration & Team Workflow**

### **Session 21: Git Collaboration**

- Branching strategies
  - Pull requests and reviews
- 

### **Session 22: Team Project Sprint**

- Small group feature implementation
  - Code reviews
- 

## **Week 12 – Pre-Advanced Assessment**

### **Session 23: Technical Assessment**

- Timed feature build
  - Debugging challenge
- 

### **Session 24: Review & Readiness Check**

- Strengths and gaps
  - Preparation for advanced track
-



# MONTH 4 — ADVANCED FRONTEND & APPLICATION ARCHITECTURE

**Goal:** Build production-quality frontend applications with better structure, performance, and UX.

---

## Week 13 – Advanced React Patterns

### Session 25: Advanced State Management

- Lifting state and composition
  - Context API
  - Introduction to state management libraries
- 

### Session 26: Forms, Validation & UX

- Advanced form handling
- Client-side validation
- Accessibility basics (a11y)

**Mini Project:** Multi-step form application

---

## Week 14 – Performance & Testing

### Session 27: Performance Optimization

- Component re-renders
  - Memoization concepts
  - Code splitting
- 

### Session 28: Frontend Testing Basics

- Why testing matters
  - Unit testing fundamentals
  - Testing user flows
-

## Week 15 – UI Systems & Design Collaboration

### Session 29: Design Systems

- Working with design files
  - Component reuse
  - Styling consistency
- 

### Session 30: Animations & Polished UI

- Animations and transitions
  - Micro-interactions
  - Improving perceived performance
- 

## Week 16 – Frontend Capstone Sprint

### Session 31: Capstone Planning & Refactor

- UI/UX audit
  - Accessibility fixes
  - Performance improvements
- 

### Session 32: Frontend Capstone Review

- Code review
- UI polish
- Production readiness checklist

#### Frontend Capstone Upgrade:

- Production-ready frontend application
- 



## MONTH 5 — ADVANCED BACKEND & SYSTEM DESIGN

**Goal:** Build scalable, secure backend systems and understand real-world architecture.

---

## Week 17 – Backend Architecture

### Session 33: Application Architecture

- MVC and service-based patterns
  - Separation of concerns
- 

### Session 34: Advanced API Design

- Pagination and filtering
  - API versioning
  - Documentation basics
- 

## Week 18 – Security & Reliability

### Session 35: Backend Security

- Common vulnerabilities
  - Rate limiting
  - Secure authentication patterns
- 

### Session 36: Logging & Monitoring

- Error logging
  - Monitoring concepts
  - Debugging production issues
- 

## Week 19 – Background Jobs & Integrations

### Session 37: Background Processing

- Cron jobs
  - Queues and async tasks
- 

### Session 38: Third-Party Integrations

- Payment gateways
  - Email services
  - External APIs
-

## Week 20 – Backend Capstone Sprint

### Session 39: Backend Capstone Build

- Scalable API implementation
  - Security hardening
- 

### Session 40: Backend Capstone Review

- Load testing basics
- Production readiness

#### Backend Capstone Project:

- Production-ready backend service
- 



## MONTH 6 — BUSINESS OF SOFTWARE, CAREER & SCALE

**Goal:** Teach students how to position themselves, earn, and grow long-term as software professionals.

---

## Week 19 – How Developers Make Money

### Session 37: Income Paths in Software

- Employment, freelancing, contracting
  - Startups, products, and SaaS
  - What clients and companies actually pay for
- 

### Session 38: Positioning & Niches

- Choosing a niche
  - Market-driven skill positioning
  - Junior vs mid-level expectations
-

## **Week 20 – Portfolio, CV & Personal Brand**

### **Session 39: Portfolio That Converts**

- Selecting the right projects
  - Writing compelling project stories
- 

### **Session 40: CV, LinkedIn & Online Presence**

- CV structure for developers
  - LinkedIn optimization
  - Personal branding basics
- 

## **Week 21 – Freelancing, Consulting & Products**

### **Session 41: Freelancing & Consulting**

- Platforms vs direct clients
  - Pricing, proposals, retainers
- 

### **Session 42: Building Products & SaaS**

- Idea validation
  - MVP thinking
  - Monetization strategies
- 

## **Week 22 – Career Acceleration**

### **Session 43: Advanced Interview Preparation**

- System design interviews
  - Take-home assessments
- 

### **Session 44: Personal Brand & Authority**

- Writing technical content
  - Open-source contribution
  - Networking strategies
-

## Week 23 – Final Capstone

### Session 45: Capstone Planning

- Define problem and scope
  - Architecture decisions
- 

### Session 46: Capstone Execution

- Build and iterate
  - Code reviews
- 

## Week 24 – Demo Day & Transition

### Session 47: Final Presentations

- Live demos
  - Feedback from instructors
- 

### Session 48: Transition to Industry

- Job search plan
  - Freelance pipeline setup
  - 6-12 month growth roadmap
- 

## Final Deliverables (6-Month Program)

- Production-ready full-stack application
- Advanced frontend or backend specialization project
- Portfolio with multiple real-world projects
- Interview-ready CV and LinkedIn
- Clear job, freelance, or product strategy