



PROGRAM OVERVIEW

This intensive program transforms you from frontend basics to full-stack contributions in our live NHS-compliant healthcare platforms and AI-powered client projects. You'll master modern JavaScript, React, Next.js, Node.js, and NestJS while contributing real code to Coredge's production systems.

Learning Philosophy

Build → Break → Fix → Ship

Every week includes hands-on coding, peer reviews, and incremental company contributions.

INTERNAL TRAINING PROGRAM

4-MONTH FULL-STACK DEVELOPMENT INTERNSHIP PROGRAM

Engineering Lead: Babar

Program Duration: 16 Weeks

Time Commitment: 20-25 hours/week

COREDGE.CO.UK

Transform from Frontend Basics to Full-Stack Contributions



SUCCESS METRICS & EXPECTATIONS

Metric	Target	How We Measure
Weekly Code Commits	15+ commits	GitHub activity
Mini-Projects Completed	8 total	Deployed links + code review
Company PRs Merged	6+ by Month 4	Pull request approvals
Technical Blog Posts	2 posts	Published on Coredge blog
Weekly 1:1 Attendance	100%	Calendar tracking
Final Capstone Feature	1 complete feature	Production deployment

JUL
17

MONTH 1

Frontend Foundations

Goal: Master JavaScript fundamentals and React basics to build interactive UIs.

Week 1: HTML, CSS & JavaScript Core

Learning Objectives

- Understand semantic HTML5 and accessibility basics
- Master CSS Flexbox and Grid for modern layouts
- Write clean JavaScript (ES6+): variables, functions, DOM manipulation

Daily Breakdown

Monday - HTML/CSS Foundations

- Study (2 hrs): freeCodeCamp Responsive Web Design (Sections 1-3)
- Practice (1 hr): Build a semantic article page with header, nav, main, footer
- Evening Task: Read MDN docs on CSS Flexbox

Tuesday - CSS Layouts

- Study (2 hrs): CSS Grid Garden game + Flexbox Froggy
- Practice (2 hrs): Create a responsive card layout (3 cols → 1 col mobile)
- Deliverable: CodePen link with your layout

Wednesday - JavaScript Basics

- Study (2 hrs): JavaScript.info (Part 1: Chapters 1-6)
- Practice (2 hrs): Console exercises: variables, functions, arrays
- Code Review: Share your functions in 1:1

Thursday - DOM Manipulation

- Study (1.5 hrs): Traversy Media - "JavaScript DOM Crash Course"
- Practice (2.5 hrs): Build an interactive FAQ accordion (click to expand)
- Test: Make it keyboard-accessible (Enter key works)

Friday - Mini-Project Start

- Build (4 hrs): Landing page with hero section, feature cards, contact form, responsive nav
- Demo: Show progress in Friday call

SETUP & TOOLS

(Complete in Week 0)

Required Software

- ```
Install these before Day 1
- Node.js v20+ (https://nodejs.org)
- VS Code + Extensions: ESLint, Prettier, GitLens
- Git + GitHub account
- Postman (API testing)
- Figma account (UI references)
```

## Development Environment

```
Terminal setup
git config --global user.name "Your Name"
git config --global user.email "your.email@coredge.co.uk"

Create internship repo
mkdir coredge-internship
cd coredge-internship
git init
echo "# Coredge Internship Journey" > README.md
```

## Weekly Routine Structure

**Monday 10:00 AM:** Week planning call (30 mins)

**Wednesday 3:00 PM:** Code review session (45 mins)

**Friday 4:00 PM:** Demo & feedback (30 mins)

**Daily:** 15-min standup message in Slack (what you did, blockers, next steps)

## Weekend Challenge

Complete the landing page and add JavaScript interactions:

- Smooth scroll to sections
- Mobile hamburger menu toggle
- Form field focus effects

## Week 1 Deliverable

Deployed Landing Page (Netlify/Vercel)

- ✓ GitHub repo with clean commits
- ✓ README with screenshots
- ✓ Mobile + desktop responsive

# Week 2: Advanced JavaScript & Async

## Learning Objectives

- Master array methods (map, filter, reduce)
- Understand promises and async/await
- Fetch data from REST APIs
- Handle errors gracefully

### Monday - Array Methods Deep Dive

- Study (2 hrs): JavaScript array methods (map, filter, reduce, find)
- Practice (2 hrs): Codewars problems (6-7 kyu difficulty)

### Tuesday - Asynchronous JavaScript

- Study (2 hrs): Promises, async/await, error handling
- Practice (2 hrs): Fetch data from JSONPlaceholder API

```
// Practice: Fetch users, posts, and comments
async function getUsers() {
 try {
 const res = await fetch('https://jsonplaceholder.typicode.com/users');
 const data = await res.json();
 console.log(data);
 } catch (error) {
 console.error('Error:', error);
 }
}
```

### Wednesday - API Integration

- Study (1 hr): REST API concepts, HTTP methods
- Practice (3 hrs): Build a simple user directory
- Features: Fetch users, display in cards, add search filter

### Thursday - Error Handling & UX

- Study (1.5 hrs): Error boundaries, loading states, empty states
- Practice (2.5 hrs): Improve user directory with loading spinner, error messages, empty state

### Friday - Mini-Project Build

- Build (4 hrs): Weather Dashboard using OpenWeather API
- Features: Search by city, display current + 5-day forecast, handle errors

## Week 2 Deliverable

Weather Dashboard App

- ✓ Deployed link + GitHub repo
- ✓ Clean code with comments
- ✓ Error handling implemented
- ✓ Screenshots in README

# Week 3: React Introduction

---

## Learning Objectives

- Understand component-based architecture
- Create reusable components with props
- Manage state with useState hook
- Handle events in React

## Monday - React Setup & JSX

- Study (2 hrs): React.dev "Learn React" (Quick Start)
- Practice (2 hrs): Create React app, explore JSX syntax

```
npx create-react-app my-first-react-app
cd my-first-react-app
npm start

Build: Header, Footer, Button components
```

## Tuesday - Props & Component Composition

- Study (1.5 hrs): Props, children, component composition
- Practice (2.5 hrs): Build reusable UI library
- Components: Button (variants), Card, Badge
- Create demo page showcasing all components

## Wednesday - State Management (useState)

- Study (2 hrs): useState hook deep dive
- Practice (2 hrs): Interactive components
- Build: Counter with increment/decrement, Toggle switch, Accordion

## Thursday - Event Handling & Forms

- Study (1.5 hrs): Event handling in React, controlled inputs
- Practice (2.5 hrs): Build contact form
- Features: Name/email/message fields, controlled inputs, validation, success message

## Friday - Mini-Project Start

- Build (4 hrs): Counter App + Theme Toggler
- Features: Multiple counters (add/remove), light/dark theme toggle, persist in localStorage

## Week 3 Deliverable

Interactive React Components Collection

- ✓ Deployed Storybook or demo page
- ✓ 5+ reusable components
- ✓ Clean component structure
- ✓ Props documentation

# Week 4: React Lists, Forms & Conditional Rendering

## Learning Objectives

- Render lists with keys
- Implement conditional rendering patterns
- Build complex forms with validation
- Understand React best practices

## Monday - Lists & Keys

- Study (1.5 hrs): Rendering lists, importance of keys
- Practice (2.5 hrs): Build product catalog
- Features: Map over products, display with cards, filter by category

## Tuesday - Conditional Rendering

- Study (1 hr): Ternary operators, logical &&, early returns
- Practice (3 hrs): Build authentication UI (no real auth)
- Features: Login form, conditional render for logged in/out views, protected content

## Wednesday - Form Validation

- Study (2 hrs): Form validation patterns in React
- Practice (2 hrs): Registration form
- Features: Email validation (regex), password strength indicator, confirm password match, real-time feedback

## Thursday - Complex State Management

- Study (1.5 hrs): Managing complex state objects
- Practice (2.5 hrs): Multi-step form
- Steps: Personal info → Address → Review, with Previous/Next navigation

## Friday - Month 1 Capstone Start

- Build (4 hrs): Todo List App
- Features: Add/edit/delete todos, mark complete, filter (All/Active/Completed), localStorage persistence

## Weekend: Complete Todo App

Add advanced features:

- Due dates
- Priority levels (high, medium, low)
- Search functionality
- Statistics (total, completed, pending)

## Month 1 Final Deliverable

Full-Featured Todo Application

- ✓ Deployed to Vercel/Netlify
- ✓ GitHub repo with clean commit history
- ✓ README with features list, screenshots, setup instructions, technologies used
- ✓ Code reviewed and approved by mentor

# Month 1 Checkpoint Review

Friday Week 4

## Discussion Topics

- What was the biggest challenge?
- Which concepts need reinforcement?
- Feedback on mentorship approach
- Readiness for Next.js and company contributions

### Month 1 Success Indicators

- All 4 weekly projects completed and deployed
- Todo app has all required + bonus features
- 50+ GitHub commits in Month 1
- Consistent attendance in all 1:1s
- Code quality improving week over week

### Next Month Preview

- Introduction to Next.js and SSR
- Advanced React patterns (custom hooks, context)
- Working with real Coredge codebases
- First company PR contributions
- TypeScript fundamentals



# PROGRAM NOTES

## Learning Path Philosophy

This program follows a deliberate progression: master fundamentals before frameworks, build confidence through repetition, and contribute to real projects as skills solidify. Each week builds directly on the previous, so consistency is critical.

## ⚠️ Falling Behind

If you're struggling with any week's material:

- Flag it immediately in your daily standup
- Schedule an extra 1:1 with your mentor
- We'll adjust the pace or provide additional resources
- Missing 2+ deliverables triggers a formal check-in

## Code Quality Standards

All submitted code must meet these minimum standards:

- ✓ Meaningful variable and function names
- ✓ Consistent indentation (2 or 4 spaces, not mixed)
- ✓ Comments explaining "why", not "what"
- ✓ No console errors in production builds
- ✓ Responsive design (mobile, tablet, desktop)
- ✓ README with setup instructions

## Resources & Support

### Mentor Support:

- Scheduled 1:1s (Mon, Wed, Fri)
- Slack response within 4 hours (Mon-Fri)
- Emergency pairing sessions available

### Learning Resources:

- Coredge internal docs & style guides
- Recommended course access (paid by company)
- Peer programming sessions

# **Ready to Begin?**

16 weeks of hands-on learning awaits

## **Month 1 Starts Now**

Complete your Week 0 setup and be ready for your first Monday planning call

---

**BUILD. LEARN. SHIP.**

COREDGE.CO.UK

4-Month Full-Stack Development Internship Program  
Internal Training Document | Confidential & Proprietary