

Summer Web Development Camp – Progressive Frontend Development (3 Month)

Duration: 12 Weeks (3 Months)

Target Audience: Beginners with basic computer literacy and an interest in web development.

Goal: Equip participants with essential frontend skills to build and deploy responsive, interactive, and modern React applications.

Month 1 – The Foundations of the Web

Goal: Build strong HTML, CSS, and JavaScript basics for transitioning into React.

Week 1: HTML Fundamentals – Building Webpage Structure

Concepts Covered:

- What is the Internet? WWW, Browsers, Web Servers.
- Client-Server model (overview).
- HTML5 structure, tags, attributes.
- Headings, paragraphs, links, images, lists, div/span.
- Semantic HTML: header, footer, nav, main, article, section, aside.
- Forms: input types, textarea, select, buttons.

Learning Objectives:

- Understand how the web works.
- Build semantic, well-structured HTML pages.
- Create basic forms for user input.

Activities & Mini-Projects:

- Set up VS Code + Chrome DevTools.
- Create a personal 'About Me' webpage.
- Build a contact form.

Week 2: CSS & Responsive Design Basics

- What is CSS? How to link CSS to HTML.
- Selectors: element, class, ID, pseudo-classes.
- Box model (content, padding, border, margin).
- Common properties: colors, fonts, backgrounds.
- Display types: block, inline, inline-block.
- Basic layout with Flexbox.

Learning Objectives:

- Apply styles to HTML pages.

- Create layouts using Flexbox.
- Understand spacing and sizing.

Activities & Mini-Projects:

- Style the 'About Me' page.
- Create a two-column responsive layout.

Week 3: Bootstrap – Faster Responsive Design

- Why use a CSS framework?
- Bootstrap Grid system (containers, rows, columns, breakpoints).
- Common components: navbar, buttons, cards, forms, carousel.
- Responsive utilities.

Learning Objectives:

- Use Bootstrap to quickly build responsive layouts.
- Customize Bootstrap components for your needs.

Activities & Mini-Projects:

- Rebuild 'About Me' page in Bootstrap.
- Create a product landing page using Bootstrap components.

Week 4: JavaScript Essentials for React

- What is JavaScript? Where does it run?
- Variables (var, let, const), data types.
- Operators and expressions.
- Functions (regular & arrow).
- Arrays and objects.
- DOM basics (querySelector, event listeners).
- ES6+ features: destructuring, template literals, array methods (map, filter).

Learning Objectives:

- Understand JavaScript syntax and logic.
- Work with data and DOM events.
- Write reusable functions.

Activities & Mini-Projects:

- Build an image gallery with lightbox.
- Create a simple calculator.

Month 2 – React.js Foundations & Version Control

Goal: Begin React development earlier and master its basics before moving to intermediate features.

Week 5: Introduction to React & JSX

- What is React? Benefits of component-based design.
- Setting up React project with Create React App (CRA) or Vite.
- JSX syntax and rules.
- Functional components.
- Props for passing data.
- State management with useState.

Activities & Mini-Projects:

- Hello World component.
- Counter app with increment/decrement.
- Display a list of items from an array.

Week 6: React Events & Component Communication

- Handling events (onClick, onChange).
- Passing data via props.
- Passing functions from parent to child.
- Conditional rendering.
- Controlled vs uncontrolled components.

Activities & Mini-Projects:

- To-Do List app.
- Theme switcher (light/dark mode).

Week 7: React Hooks & Side Effects

- What are hooks? Rules of hooks.
- useEffect for side effects.
- Fetching data from public APIs.
- Loading and error states.

Activities & Mini-Projects:

- News feed app fetching articles from an API.
- Countdown timer using useEffect.

Week 8: Git & GitHub for React Projects

- Version control basics.
- Git commands: init, add, commit, status, log.
- Branching and merging.
- GitHub: repositories, pushing, pull requests.
- .gitignore basics.

Activities & Mini-Projects:

- Initialize Git in React project.

- Push portfolio to GitHub.
- Practice branching with group project.

Month 3 – Intermediate React & Final Project

Goal: Master more advanced React features, routing, and deploy a final project.

Week 9: React Router & Global State

- React Router: BrowserRouter, Routes, Link, useNavigate.
- Context API for global state.
- Managing shared data across components.

Activities & Mini-Projects:

- Multi-page portfolio site.
- Shopping cart UI with Context API.

Week 10: Advanced React UI Patterns

- Controlled forms and validation.
- Reusable components.
- Conditional rendering.
- PropTypes for type checking.

Activities & Mini-Projects:

- Product catalog with search & filters.
- Modal component reused across app.

Week 11: Final Project Development

- Project planning.
- UI/UX best practices.
- Debugging and optimization.

Activities:

- Students work on capstone project.

Week 12: Deployment & Showcase

- Deploying with Netlify or Vercel.
- Presenting projects.
- Career paths in frontend.

Activities:

- Deploy final project.
- Live presentation.

Final Course Outcome

By completing this camp, students will:

- Build responsive HTML/CSS layouts with Bootstrap.

- Write interactive JavaScript for the browser.
- Develop modern React applications.
- Fetch and display data from public APIs.
- Manage state locally and globally.
- Use Git & GitHub for version control.
- Deploy professional frontend projects online.
- Graduate with multiple React projects.