



# E-HOOYIA

## Formation sur le développement Front End

**Presented by:**  
Hooyia

 [e-hooyia.com](http://e-hooyia.com)

 Bafoussam, Cameroon



# THE FRONTEND ROADMAP

## Table of contents

Training program Overview: .....	1
Session 1: HTML & CSS Core Concepts (Month 1) .....	1
Week 1: HTML Basics .....	1
Week 2: CSS Fundamentals .....	1
Week 3: Flexbox & CSS Grid .....	1
Week 4: Advanced CSS Concepts .....	1
Session 2: JavaScript & Introduction to React.js (Month 2) .....	2
Week 1: JavaScript Basics .....	2
Week 2: DOM Manipulation & Events .....	2
Week 3: JavaScript ES6+ Features .....	2
Week 4: Introduction to React.js .....	2
Session 3: React.js Core Concepts (Month 3) .....	3
Week 1: React Hooks (useState, useEffect) .....	3
Week 2: React Router & Component Styling .....	3
Week 3: Advanced React (Hooks, Context API) .....	3
Week 4: Final Project .....	3
Deliverables: .....	4

## Training program Overview:

- Session 1(month 1) : HTML & CSS Core Concepts
- Session 2(month2): JavaScript & Introduction to React.js
- Session 3(month 3) : React Js core concepts

Each week, students are required to submit a project based on the topics learned.

---

## Session 1: HTML & CSS Core Concepts (Month 1)

### Week 1: HTML Basics

- **Topics:** Basic HTML structure, elements, attributes, forms, and semantic HTML.
- **Project:** Create a simple personal webpage with a header, navigation menu, content sections (about me, skills), and a footer.

### Week 2: CSS Fundamentals

- **Topics:** CSS selectors, the box model, positioning, and display properties.
- **Project:** Style the personal webpage from Week 1 using CSS for layout and design, implementing the box model and proper positioning of elements.

### Week 3: Flexbox & CSS Grid

- **Topics:** Flexbox layout, CSS Grid, media queries for responsive design.
- **Project:** Redesign the personal webpage to be responsive, using Flexbox and CSS Grid. The layout should adapt smoothly across mobile, tablet, and desktop.

### Week 4: Advanced CSS Concepts

- **Topics:** Animations, pseudo-classes, pseudo-elements, and CSS transitions, keyframes.
- **Project:** Add hover effects, animations, and transitions to the personal webpage. Implement smooth scrolling and animated navigation.

---

## Session 2: JavaScript & Introduction to React.js (Month 2)

### Week 1: JavaScript Basics

- **Topics:** Variables, data types, functions, and event listeners.
- **Project:** Create a basic calculator web app with HTML, CSS, and JavaScript. The calculator should perform simple arithmetic operations and update results dynamically.

### Week 2: DOM Manipulation & Events

- **Topics:** DOM elements, query selectors, manipulating styles, and handling events.
- **Project:** Add interactivity to a to-do list web app. Users should be able to add, edit, and delete tasks. Use DOM manipulation to update the UI in real-time.

### Week 3: JavaScript ES6+ Features

- **Topics:** Let/const, arrow functions, template literals, classes, and array methods.
- **Project:** Refactor the to-do list app using ES6+ features like arrow functions, classes, and array methods (map, filter, reduce).

### Week 4: Introduction to React.js

- **Topics:** React components, JSX, props, and state management.
- **Project:** Build a basic weather app using React.js. The app should display weather information for a city by fetching data from a public API (e.g., OpenWeather API).

## Session 3: React.js Core Concepts (Month 3)

### Week 1: React Hooks (useState, useEffect)

- **Topics:** Managing state and side effects with useState and useEffect.
- **Project:** Extend the weather app to include multiple cities and update weather data automatically based on a timer using useEffect.

### Week 2: React Router & Component Styling

- **Topics:** Setting up routes with React Router and styling components with CSS Modules or styled-components.
- **Project:** Convert the weather app into a multi-page application with a homepage, city details page, and an about page using React Router.

### Week 3: Advanced React (Hooks, Context API)

- **Topics:** Advanced hooks (useContext, useReducer), Context API for global state management.
- **Project:** Implement global state management for the weather app, allowing users to save and view their favorite cities. Use Context API or useReducer for managing the state across components.

### Week 4: Final Project

- **Topics:** Full-stack concepts (optional), component optimization, code splitting, and best practices.
  - **Project:** Students will build a portfolio website using React.js to showcase the projects they've created during the course. This final project should include smooth animations, responsive design, routing, and an interactive UI.
-

**Deliverables:**

- Each student must submit their projects weekly. All code must be hosted on GitHub, with detailed documentation explaining their approach, challenges, and lessons learned.