Web Design & Frontend Development with AI Integration

Comprehensive Course Plan (2.5 months, 1.5 hours daily)

Week 1: Fundamentals & Design

Day 1: Course Introduction & Designing with Figma & AI

- Course overview and expectations
- Introduction to web design principles
- Setting up Figma accounts
- Basic Figma interface navigation
- AI tools in design (Figma AI features)

Day 2: Designing with Figma & AI (continued)

- Creating wireframes and mockups
- Component-based design
- Using AI to generate design ideas
- Practice exercise: Create a simple landing page mockup

Day 3: Introduction to HTML & Web Page Structure

- HTML basics and document structure
- Essential HTML tags (headings, paragraphs, links)
- Setting up development environment (VS Code)
- Creating your first HTML page

Day 4: Introduction to HTML & Web Page Structure (continued)

- Working with images and media
- Lists and navigation elements
- HTML attributes
- Practice exercise: Building a personal profile page

Day 5: AI-Assisted HTML Forms & Tables

- Creating HTML forms (inputs, buttons, dropdowns)
- Form validation attributes
- Using AI tools to generate form structures

- Tables: structure and basic styling
- Practice: Create a contact form with AI assistance

Week 2: Advanced HTML & Git

Day 1: Semantic HTML & Best Practices

- Semantic HTML elements (header, footer, nav, etc.)
- Accessibility best practices
- SEO considerations
- Practice: Refactoring a non-semantic page

Day 2: Semantic HTML & Best Practices (continued)

- HTML5 features (audio, video, canvas)
- Validating HTML
- Performance considerations
- Practice: Building an accessible media page

Day 3: Version Control with Git

- Introduction to version control concepts
- Setting up Git and GitHub accounts
- Basic Git commands (init, add, commit)
- Creating repositories

Day 4: Version Control with Git (continued)

- Branching and merging
- Collaboration workflows
- Pull requests
- Practice: Initializing project repository and making commits

Day 5: Introduction to CSS & AI in Design

- CSS syntax and structure
- Selectors and specificity
- Inline, internal, and external CSS
- Using AI to generate CSS
- Practice: Styling your HTML profile page

Week 3: CSS Fundamentals

Day 1: Styling Text & AI-Driven Typography

- Web typography principles

- Font properties and web fonts
- AI tools for typography selection
- Practice: Creating a typography system with AI assistance

Day 2: CSS Box Model

- Understanding the box model (margin, padding, border)
- Width/height properties
- Box-sizing property
- Practice: Building content cards with proper spacing

Day 3: CSS Position Layout

- Static, relative, absolute positioning
- Fixed and sticky positioning
- Z-index property
- Practice: Creating a fixed header and positioned elements

Day 4: CSS Layouts & AI-Enhanced Design

- Flexbox fundamentals
- Flex container and flex item properties
- Using AI tools to optimize layouts
- Practice: Building a flexible card layout

Day 5: CSS Layouts & AI-Enhanced Design (continued)

- CSS Grid fundamentals
- Grid container and grid item properties
- Combining Flexbox and Grid
- AI-powered layout generation
- Practice: Creating a responsive grid gallery

Week 4: Responsive Design & Advanced CSS

Day 1: Responsive Design & AI Optimization

- Mobile-first approach
- Media queries
- Viewport meta tag
- Using AI to test responsiveness
- Practice: Converting a fixed layout to responsive

Day 2: Responsive Design & AI Optimization (continued)

- Responsive images and media

- Testing across devices
- AI tools for responsive optimization
- Practice: Building a fully responsive page

Day 3: CSS Animations, Transitions & AI Motion Design

- CSS transitions
- Transform properties
- Using AI to generate transition effects
- Practice: Adding hover effects and transitions

Day 4: CSS Animations, Transitions & AI Motion Design (continued)

- Keyframe animations
- Animation properties
- AI-generated animation patterns
- Practice: Creating a loading animation

Day 5: Introduction to SASS/SCSS & AI in Styling

- SASS/SCSS syntax and features
- Variables and mixins
- Nesting and partials
- AI-powered SCSS conversion
- Practice: Converting CSS to SCSS with enhancements

Week 5: CSS Frameworks & JavaScript Intro

Day 1: UI Frameworks

- Introduction to Bootstrap
- Grid system and components
- Customizing Bootstrap
- Practice: Building a Bootstrap-based landing page

Day 2: UI Frameworks (continued)

- Other popular frameworks (Tailwind CSS)
- Utility-first approach
- Framework selection considerations
- Practice: Converting a design to Tailwind CSS

Day 3: JavaScript ES6+ Features & DOM Manipulation

- JavaScript fundamentals
- Variables, data types, and operators

- Functions and scope
- Modern ES6+ features
- Practice: Basic JavaScript exercises

Day 4: JavaScript ES6+ Features & DOM Manipulation (continued)

- Arrays and objects
- Higher-order functions
- Template literals
- Arrow functions
- Practice: Manipulating data structures

Day 5: DOM Manipulation & Events

- Understanding the DOM
- Selecting and modifying elements
- Creating and appending elements
- Events and event listeners
- Practice: Building an interactive element

Week 6: JavaScript Advanced & jQuery

Day 1: DOM Manipulation & Events (continued)

- Form handling and validation
- Event delegation
- Local storage
- Practice: Creating a form with validation

Day 2: JQuery

- Introduction to jQuery
- jQuery selectors and methods
- DOM manipulation with jQuery
- jQuery events
- Practice: Converting JavaScript to jQuery

Day 3: JQuery (continued)

- jQuery effects and animations
- AJAX with jQuery
- jQuery plugins
- Practice: Building an interactive widget

Day 4: Asynchronous JavaScript

- Understanding asynchronous code
- Callbacks and callback hell
- Promises fundamentals
- Practice: Converting callbacks to promises

Day 5: Asynchronous JavaScript (continued)

- Fetch API
- Async/await syntax
- Error handling
- Practice: Building a data fetching application

Week 7: React Fundamentals

Day 1: Introduction to React

- React philosophy and virtual DOM
- Setting up React environment
- JSX syntax
- Your first React component
- Practice: Converting HTML to React components

Day 2: Introduction to React (continued)

- Functional components
- Props and state
- React hooks (useState)
- Practice: Building a stateful component

Day 3: React-Router-Dom

- Client-side routing concepts
- Setting up React Router
- Creating routes and navigation
- Practice: Building a multi-page React application

Day 4: React-Router-Dom (continued)

- Nested routes
- Route parameters
- Protected routes
- Practice: Enhancing application routing

Day 5: Form Handling

- Controlled components

- Form events and state
- Form validation in React
- Practice: Building a React form

Week 8: React Advanced & Next.js

Day 1: Introduction to APIs

- API concepts and RESTful principles
- Making API requests with React
- Displaying API data
- Practice: Building a data dashboard

Day 2: Introduction to Next.js

- Next.js features and benefits
- Setting up a Next.js project
- Pages and components structure
- Practice: Converting React app to Next.js

Day 3: Styling in Next.js

- CSS Modules
- Styled Components
- Global styles
- Practice: Implementing styling in Next.js

Day 4: Next.js Routing and Navigation

- File-based routing
- Dynamic routes
- Navigation with Link and Router
- Practice: Building a multi-page Next.js application

Day 5: Fetching and Managing Data

- getStaticProps and getServerSideProps
- Static Generation vs Server-side Rendering
- Data fetching strategies
- Practice: Implementing data fetching in Next.js

Week 9: Advanced Next.js & AI Integration

Day 1: State Management in Next.js

- Local state with useState

- Context API
- Introduction to Redux
- Practice: Implementing state management

Day 2: Performance Optimization

- Image optimization
- Code splitting
- Lazy loading
- Practice: Optimizing a Next.js application

Day 3: AI Integration in React/Next.js

- AI libraries for React
- Natural language processing integration
- Image recognition components
- Practice: Adding an AI feature to your application

Day 4: Deployment and Hosting in React/Next.js

- Build process
- Vercel deployment
- Environment variables
- Practice: Deploying your Next.js application

Day 5: Project Planning Session

- Project requirements discussion
- Planning architecture and features
- Setting up project repositories
- Initial scaffolding

Week 10: Projects

Days 1-5: Guided Project Development

- Students work on their projects
- Daily stand-ups and progress checks
- Instructor guidance and troubleshooting
- Code reviews and feedback

Teaching Methodologies:

1. **Hands-on Learning**: Each session includes practical exercises
2. **Project-Based Approach**: Multiple mini-projects building up to final projects

3. **AI Integration**: Demonstrate how AI tools can enhance development workflow
4. **Pair Programming**: Occasional pair programming sessions for collaborative learning
5. **Code Reviews**: Regular feedback on code quality and best practices
6. **Flipped Classroom**: Pre-reading materials with in-class application of concepts

Assessment Methods:

1. Daily quick exercises (30%)
2. Weekly mini-projects (30%)
3. Final projects (40%)

Resources Required:

- Computer lab with internet access
- Code editor (VS Code recommended)
- Figma accounts
- Git and GitHub accounts
- Access to AI design and development tools