THE COMPLETE FRONT-END DEVELOPER ROADMAP

Go from zero to a front-end developer in 12 months

Mosh Hamedani

Hi! I am Mosh Hamedani, a software engineer with over 20 years of experience.

Over the past 10 years, I've had the privilege of teaching millions of people how to code and become professional software engineers through my YouTube channel and online courses.

It's my mission to make software engineering accessible to everyone. Join me on this journey and unlock your potential in the world of coding!

https://codewithmosh.com

Copyright 2024 Code with Mosh

Table of Content

Copyright 2024 Code with Mosh

Introduction

This guide is designed to help you navigate the essential skills needed to become a successful frontend developer. Whether you're just starting out or looking to enhance your existing skills, this roadmap will provide a clear and structured path.

Target Audience

This guide is for:

- **Beginners** who want to know what they need to learn to land a front-end developer job.
- **Experienced individuals** looking to level up their skills and fill in the gaps in their knowledge.

Resources

For detailed tutorials and full courses, check out the following resources:

- **YouTube Channel**: https://www.youtube.com/c/programmingwithmosh
- **Full Courses**: https://codewithmosh.com

Copyright 2024 Code with Mosh

Essential Skills and Learning Timeline

Front-end development has many tools and technologies. Trying to learn them all is impossible and not practical. This guide focuses on the most important and widely used skills and tools to help you get the best job opportunities.

I've selected these skills because they are in high demand. Mastering them will give you a strong foundation and make you a competitive job candidate.

For the first 12 months, focus only on the tools and technologies listed in this document. Instead of trying to learn too many things at once, build a strong foundation with these essential skills. You can always learn other tools and technologies on the job as you go.

Skill	Time required	Learning Phase
HTML	2 weeks	Beginner
CSS	1 month	Beginner
JavaScript	2 months	Beginner
Git	2 weeks	Beginner
TypeScript	3 weeks	Intermediate
React	2 months	Intermediate
SASS	2 weeks	Intermediate
Tailwind	3 weeks	Intermediate
Automated Testing	1 month	Advanced
Next.js	1.5 month	Advanced
React Native (Optional)	2 months	Advanced
Total	**1 Year**	

Copyright 2024 Code with Mosh

Top Tips Every Beginner Should Know

1. **Start Small and Build Up:** Begin with the basics and gradually move to more complex topics. Don't rush; build a strong foundation.

2. **Practice Consistently:** Set aside time each day or week to practice coding. Consistency is key to retaining knowledge and improving skills.

3. **Work on Projects:** Apply what you learn by working on real projects. This helps reinforce your knowledge and gives you practical experience.

4. **Ask for Help:** Ask ChatGPT for help or post your questions on StackOverflow to get help from the community. Additionally, participate in answering other people's questions. This is a great way to learn and reinforce your knowledge.

5. **Join a Community:** Engage with other learners and professionals. Join online forums, attend meetups, and participate in coding challenges. This can provide support, motivation, and valuable insights.

6. **Learn to Debug:** Debugging is a crucial skill. Practice finding and fixing errors in your code. It improves problem-solving skills and helps you understand how code works.

7. **Try Rubber Duck Debugging:** Explain your code and the problem you're facing to an inanimate object like a rubber duck. This method, known as rubber duck debugging, can help you think more clearly and often leads to discovering the solution on your own.

8. **Read Documentation:** Familiarize yourself with official documentation. It's an essential skill for understanding and using new tools and technologies effectively.

Copyright 2024 Code with Mosh

9. **Stay Updated:** Front-end development is always evolving. Follow industry blogs, news, and social media to stay informed about the latest trends and updates.

10. **Be Patient and Persistent:** Learning to code is a journey. Be patient with yourself and stay persistent, even when things get tough. Progress takes time and effort.

11. **Embrace Change:** The front-end industry is always evolving with new versions of languages and tools being released frequently. Dealing with breaking changes is a common challenge, and many courses and books can become outdated quickly. Be patient and embrace these changes. Practicing how to handle them will prepare you for real-world scenarios, as this is a regular part of a front-end developer's job.

12. **Take Regular Breaks:** Stepping away from your desk and taking regular breaks can refresh your mind and help you find solutions to problems. Sometimes, a short walk or a change of scenery is all you need to spark new ideas and improve your productivity.

Good luck, and happy coding!

Mosh

Copyright 2024 Code with Mosh

HTML

HTML, or Hypertext Markup Language, is the foundation of web development used for structuring web pages. It defines the structure and content of web documents through elements like headings, paragraphs, links, images, and lists.

Time required: 1-2 weeks

Learning resources: YouTube Tutorial | Full Course

Essential Concepts

- **Basic Tags**: <html>, <head>, <body>, <title>
- **Text Formatting:** <h1> to <h6>, <p>,
, <hr>, ,
- **Lists:** , <ol>,
- **Links:** <a>, href, target
- **Images:** <img>, src, alt, width, height
- **Tables:** <table>, <tr>, <td>, <th>, colspan, rowspan
- **Forms:** <form>, <input>, <textarea>, <button>, <select>, <option>, <label>
- **Semantic Elements:** <header>, <nav>, <main>, <section>, <article>, <footer>
- **Meta Tags:** <meta>, charset, name, content, viewport
- **Multimedia:** <audio>, <video>, controls, <source>

Copyright 2024 Code with Mosh

CSS

CSS, or Cascading Style Sheets, is used to style and layout web pages. It allows you to control the visual presentation of HTML elements, including colors, fonts, spacing, and positioning, creating responsive designs that adapt to various screen sizes.

Time required: 2-4 weeks

Learning resources: YouTube Tutorial | Full Course

Essential Concepts

- **Selectors:** element, class, id, attribute, pseudo-class, pseudo-element
- **Box Model:** margin, border, padding, content
- **Positioning:** static, relative, absolute, fixed, sticky
- **Display:** block, inline, inline-block, none, flex, grid
- **Flexbox:** justify-content, align-items, flex-direction, flex-wrap
- **Grid:** grid-template-columns, grid-template-rows, gap, grid-area
- **Typography:** font-family, font-size, font-weight, line-height, text-align, text-decoration
- **Colors:** color, background-color, opacity, rgba, hex, hsl
- **Units:** px, em, rem, %, vh, vw
- **Transitions and Animations:** transition, transform, animation
- **Responsive Design:** media queries, @media, max-width, min-width

Copyright 2024 Code with Mosh

HTML/CSS Project Ideas

Personal Portfolio Website

Create a personal portfolio website that showcases your projects, skills, and contact information.

- Home page with a welcome message and navigation menu
- About page with a brief bio and photo
- Projects page with thumbnails and descriptions of your work
- Contact page with a contact form and social media links
- Get design inspirations from dribbble.com

Responsive Blog Layout

Design a responsive blog layout with a header, footer, sidebar, and main content area.

- Header with a logo and navigation menu
- Sidebar with recent posts and categories
- Main content area with blog posts formatted with headings, images, and paragraphs
- Footer with social media links and copyright information

Copyright 2024 Code with Mosh

Landing Page for a Product

Create a landing page for a fictional product, including a call-to-action (CTA) button and an email subscription form.

- Hero section with a large background image, product tagline, and CTA button
- Features section with icons and descriptions of the product's benefits
- Testimonials section with customer reviews
- Email subscription form

Copyright 2024 Code with Mosh

JavaScript

JavaScript is a programming language that adds interactivity and dynamic behavior to web pages. It handles tasks like user interactions, form validation, animations, and fetching data from servers, making web pages more engaging and functional.

Time required: 6-8 weeks

Learning resources: YouTube Tutorial | Full Course

Essential Concepts

- **Variables:** declarations (var, let, const), scope (block, functional, global), hoisting
- **Data Types:** primitive types (strings, number, boolean, undefined, null, Symbol), Object, typeof operator
- **Type Casting:** explicit casting, implicit casting, type conversion vs coercion
- **Operators:** assignment, comparison, arithmetic, bitwise, logical, conditional
- **Equality Comparisons:** ==, ===, Object.is
- **Control Flow:** if, else, switch
- **Loops:** for, for...in, for...of, while, do...while, break, continue
- **Functions:** function declaration, function expression, arrow functions, parameters, return values
- **Arrays:** creation, methods (push, pop, shift, unshift, map, filter, reduce)
- **Objects:** creation, properties, methods, this keyword
- **Classes**

Copyright 2024 Code with Mosh

- **Data Structures:** Map, WeakMap, Set, WeakSet, JSON
- **Error Handling:** try, catch, finally, throw, Error objects
- **Asynchronous JavaScript:** Promises, async/await, callbacks, callback hell
- **DOM Manipulation:** document.getElementById, document.querySelector, addEventListener, innerHTML, style
- **Events:** click, submit, load, change, focus, blur, event propagation (bubbling and capturing)
- **Working with APIs:** fetch
- **Browser Storage:** local storage, web storage
- **Modules:** CommonJS, ECMAScript Modules

Copyright 2024 Code with Mosh

JavaScript Project Ideas

Todo List

Build an interactive to-do list application where users can add, remove, and mark tasks as completed.

- Input field for new tasks
- List of tasks with checkboxes to mark completion
- Delete button to remove tasks
- Save tasks in local storage

Weather App

Create a weather application that fetches and displays weather data based on user input.

- Input field for city name
- Display current weather information (temperature, description, icon)
- Fetch data from a weather API
- Error handling for invalid inputs

Copyright 2024 Code with Mosh

Image Carousel

Develop an image carousel that automatically transitions between images and allows manual navigation.

- Automatic image sliding with a timer
- Previous and next buttons for manual navigation
- Indicators to show the current image
- Responsive design

Quiz App

Build a quiz application that presents multiple-choice questions to the user and displays their score at the end.

- Display one question at a time with multiple-choice answers
- Highlight correct and incorrect answers
- Track and display the user's score
- Option to restart the quiz

Copyright 2024 Code with Mosh

Git

Git is a version control system that tracks changes in code, allowing multiple developers to collaborate efficiently. It helps manage and maintain different versions of code, facilitates branching and merging, and stores the project history.

Time required: 1-2 weeks

Learning resources: YouTube Tutorial | Full Course

Essential Concepts

- **Setup and Configuration**: init, clone, config
- **Staging**: status, add, rm, mv, commit, reset
- **Inspect and Compare**: log, diff, show
- **Branching**: branch, checkout, merge
- **Remote Repositories**: remote, fetch, pull, push
- **Temporary Commits**: stash
- **GitHub**: fork, pull request, code review

Copyright 2024 Code with Mosh

TypeScript

TypeScript is a superset of JavaScript that adds static typing and other features, making code more robust and maintainable. It helps catch errors early during development and is widely used in large-scale applications.

Time required: 2-3 weeks

Learning resources: YouTube Tutorial | Full Course

Essential Concepts

- **Basics Types:** string, number, boolean, array, tuple, enum, any, void, null, undefined, never, unknown
- **Type Assertion:** as keyword, <> syntax
- **Interfaces:** defining, extending, optional properties, readonly properties, dynamic keys
- **Classes:** properties, methods, constructors, inheritance, access modifiers (public, private, protected)
- **Functions:** type annotations, optional and default parameters, rest parameters
- **Generics:** generic functions, generic classes
- **Modules:** import, export, namespaces
- **Utility types:** Partial, Pick, Omit, Readonly, Record, Exclude, etc

Copyright 2024 Code with Mosh

React

React is a popular JavaScript library for building user interfaces, particularly single-page applications. It allows developers to create reusable UI components, manage application state efficiently, and handle dynamic data changes.

Time required: 6-8 weeks

Learning resources: YouTube Tutorial | Full Course

Essential Concepts

- **Basics:** components, props, state, JSX
- **Rendering:** conditional rendering, rendering lists
- **Hooks:** useState, useEffect, useReducer, useRef, custom hooks
- **Styling:** using vanilla CSS, CSS modules, CSS-in-JS
- **Forms:** react-hook-forms, zod
- **Data Fetching:** fetch API, axios
- **State Management:** lifting state up, Context API, React Query, Redux
- **Routing:** React Router

Copyright 2024 Code with Mosh

React Project Ideas

Simple Todo List App

Create a simple to-do list application where users can add, delete, and mark tasks as complete.

- Add new tasks with a form input
- Display a list of tasks with checkboxes to mark completion
- Delete tasks from the list
- Filter tasks by completed and pending status

Weather App

Build a weather application that fetches and displays weather data based on user input.

- Input field for entering a city name
- Fetch current weather data from a weather API
- Display weather information such as temperature, humidity, and weather conditions
- Handle loading states and errors

Copyright 2024 Code with Mosh

Recipe Finder

Create an application that allows users to search for recipes and view details.

- Search bar for entering ingredients or recipe names
- Display a list of matching recipes with images
- Click on a recipe to view detailed information including ingredients and steps

E-commerce Storefront

Build a simple e-commerce storefront with product listings and a shopping cart.

- Display a list of products with images, prices, and descriptions
- Add products to a shopping cart
- View the shopping cart with a list of selected products and total price
- Remove items from the cart and update quantities

Expense Tracker

Build an expense tracker application to manage personal finances.

- Add new expenses with details such as amount, category, and date
- Display a list of expenses with filtering options
- Visualize expenses with charts (e.g., pie chart for categories)
- Calculate total expenses and display summary statistics

Copyright 2024 Code with Mosh

SASS

SASS (Syntactically Awesome Stylesheets) is a CSS preprocessor that extends CSS with features like variables, nested rules, and mixins. It simplifies writing and managing CSS for large projects, improving efficiency and maintainability.

Time required: 1-2 weeks

Essential Concepts

- **Variables:** defining, using, scope
- **Loops:** for loops, each loops, while loops
- **Nesting:** rules, selectors
- **Mixins:** creating, including, parameters, default values
- **Inheritance:** @extend
- **Functions:** built-in functions, custom functions
- **Feature checks:** feature-exists
- **Other features:** conditionals, lists, maps, interpolation

Copyright 2024 Code with Mosh

Tailwind CSS

Tailwind is a utility-first CSS framework that provides a set of predefined classes for rapid UI development. It enables developers to build custom designs directly in the HTML by applying utility classes, ensuring consistency and speed.

Time Required: 2-3 weeks

Essential Concepts

- **Utility-first CSS:** principles, benefits
- **Configuration:** tailwind.config.js, customizing themes
- **Applying Styles:** utility classes for layout, flexbox, grid, sizing, spacing, borders, typography, colors, backgrounds, transitions, animations, transforms
- **Responsive Design:** responsive utilities, breakpoints
- **Plugins:** adding and configuring plugins

Copyright 2024 Code with Mosh

Automated Testing

Jest and Vitest are testing frameworks for JavaScript applications that enable developers to write tests for their code. They help ensure code reliability and correctness by automating the testing process, identifying bugs, and verifying functionality.

Time required: 3-4 weeks

Learning resources: YouTube Tutorial | Full Course

Essential Concepts

- **Basics:** setting up, writing test cases
- **Matchers:** common matchers (toBe, toEqual, toContain, toBeTruthy, toBeFalsy, toBeNull, toBeUndefined, toBeDefined, toMatch, toMatchObject, toHaveProperty, toHaveLength)
- **Mocks:** mocking functions, modules, timers
- **Testing Asynchronous Code:** async/await, promises
- **Code Coverage:** collecting and reporting coverage
- **Testing React Components with React Testing Library:** queries (get, query, find), matchers (toBeChecked, toBeDisabled, toBeInTheDocument, toHaveAttribute, toHaveTextContent), firing events with user-event
- **Mocking APIs** with Mock Service Worker (MSW)

Copyright 2024 Code with Mosh

Next.js

Next.js is a meta framework built on top of React, enhancing its capabilities with features like server-side rendering (SSR) and static site generation (SSG). It simplifies building and optimizing modern web applications with improved performance and SEO.

Time required: 4-6 weeks

Learning resources: YouTube Tutorial | Full Course

Essential Concepts

- **Basics:** client and server components, client and server rendering, static and dynamic rendering, static generation (SSG)
- **Styling:** global styles, CSS modules, Tailwind
- **Routing:** pages, layouts, dynamic routes, linking and navigation, error handling, loading UI and streaming
- **Data Fetching:** fetch API, caching
- **Building APIs:** route handlers
- **Database Integration:** Prisma
- **Authentication:** NextAuth.js
- **Optimizations:** image optimization, lazy loading, automatic code splitting

Copyright 2024 Code with Mosh

Next.js Project Ideas

Personal Blog

Create a personal blog where you can write and publish articles.

- Static generation for blog posts using Markdown or a CMS
- Dynamic routing for individual post pages
- A homepage with a list of recent posts
- SEO optimization with metadata and Open Graph tags

E-commerce Store

Build a fully-functional e-commerce store with product listings and a shopping cart.

- Product pages generated statically
- Shopping cart with add/remove items functionality
- Checkout page with order summary
- Fetch product data from an API or CMS

Copyright 2024 Code with Mosh

Event Booking Platform

Build an event booking platform where users can browse and book events.

- Event listings with details
- Booking form with validation
- User authentication for managing bookings
- Admin dashboard for creating and managing events

Recipe Sharing Platform

Build a platform where users can share and discover recipes.

- User authentication for submitting recipes
- Recipe pages with ingredients, steps, and images
- Search functionality to find recipes by ingredients or name
- Static generation for recipe pages

Copyright 2024 Code with Mosh

React Native

React Native is a framework for building cross-platform mobile applications using React. It allows developers to write code once and deploy it to both iOS and Android platforms, leveraging native components for a seamless user experience.

Time required: 6-8 weeks

Learning resources: YouTube Tutorial | Full Course

Essential Concepts

- **Basics:** View, Text, Image, Touchables, Button, Alert, platform-specific code
- **Layouts:** dimensions, detecting orientation, Flexbox, absolute and relative
- **Styling:** StyleSheet, borders, shadows, padding, styling text, icons, platform-specific styles
- **Input Components:** TextInput, Switch, Picker, custom pickers
- **Navigation:** React Navigation, stack navigator, tab navigator, drawer navigator
- **Native Modules:** linking native code, using third-party libraries
- **Offline Support:** detecting network status, caching, AsyncStorage
- **Authentication:** auth providers
- **Notifications:** push notification services
- **Distribution:** optimizing assets, building, error reporting, environment management

Copyright 2024 Code with Mosh

React Native Project Ideas

Fitness Tracker

Create a fitness tracker application to log workouts and track progress.

- Log workouts with details like type, duration, and calories burned
- Display a summary of daily/weekly workouts
- Track progress with charts and statistics
- Set and track fitness goals

Expense Tracker

Develop an expense tracker application to manage personal finances.

- Add new expenses with details such as amount, category, and date
- Display a list of expenses with filtering options
- Visualize expenses with charts (e.g., pie chart for categories)
- Calculate total expenses and display summary statistics

News App

Create a news application that fetches and displays the latest news articles.

- Fetch news articles from a news API
- Display a list of news articles with headlines and images
- View detailed news articles
- Filter news by categories (e.g., sports, technology, politics)

Copyright 2024 Code with Mosh

E-commerce App

Build an e-commerce application with product listings, a shopping cart, and checkout functionality.

- Display a list of products with images, prices, and descriptions
- Add products to a shopping cart
- View the shopping cart with a list of selected products and total price
- Checkout process with order summary

Copyright 2024 Code with Mosh

Learning to code is a journey. Be patient with yourself and stay persistent, even when things get tough.

- Mosh