**What is Front End Development?**

Everything you see on a website, like buttons, links, animations, and more, were created by a front end web developer. It is the front end developer's job to take the vision and design concept from the client and implement it through code.



Everything on the page from the logo to the search bar, buttons, overall layout and how the user interacts with the page was created by a front end developer. Front end developers are in charge of the look and feel of the website.

Front end developers also have to make sure the website looks good on all devices (phones, tablets, and computer screens).

## What Skills Do You Need to Become a Front End Developer?

The three main languages you need to know well are HTML, CSS, and JavaScript. From there you can focus on frameworks, libraries, and other useful tools.

### HTML

HTML stands for HyperText Markup Language. HTML displays the content on the page like buttons, links, headings, paragraphs, and lists.

You should not use HTML for styling. That is what CSS is for.

### CSS

CSS stands for Cascading Style Sheets. CSS is responsible for the style of your web page including colors, layouts, and animations.

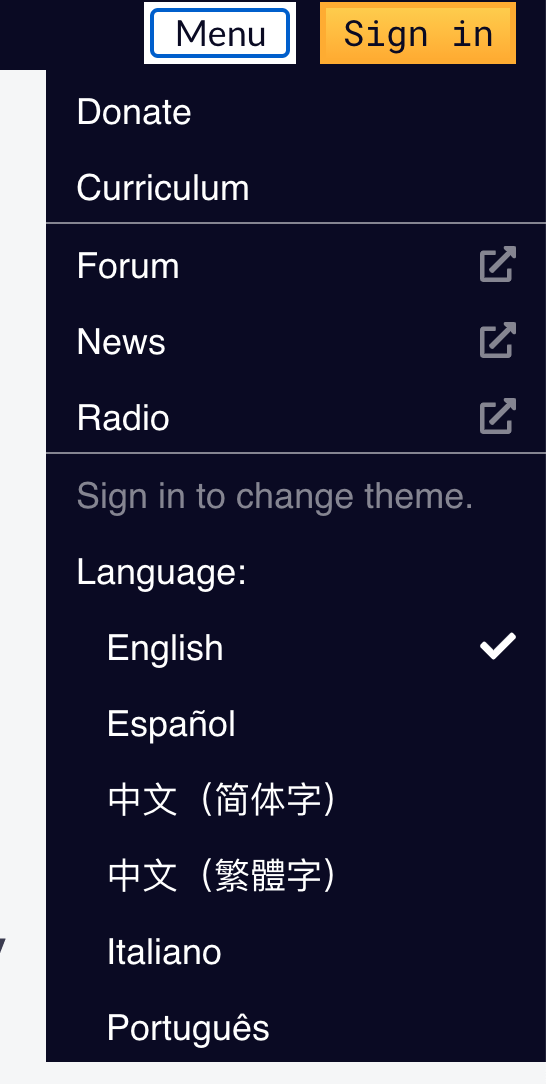
Responsive design is essential in creating websites that look good on all devices.

Accessibility is the practice of making sure that everyone can easily use your web sites. You do not want to create web sites that cannot be used by those with assistive technologies like screen readers.

### JavaScript

JavaScript allows users to interact with the web page. Examples of JavaScript can be found in virtually any web page including the freeCodeCamp homepage.

For example, when I click on the Menu button at the top of the page, it will open a dropdown list of options. Every time I click on that button, it will toggle back and forth between opening and closing the Menu.



JavaScript can be used in online games, web pages, mobile applications and more.

### CSS Frameworks, Libraries, and Preprocessors

Once you learn the basics of CSS, then you can start to work with different frameworks and libraries. These tools were created as a way to help speed up the development process.

Frameworks like [Bootstrap](https://getbootstrap.com/) and [Tailwind CSS](https://tailwindcss.com/) allow you to add the catalog of classes to your webpage. As a result, you end up with professional and mobile-friendly designs.

There are dozens of options on the market and you don't need to learn them all. It's often helpful to look at jobs in your area and see what technologies they're using. Then you can focus on the most common/in-demand skills.

Here is a list of a few options:

* [Bootstrap](https://getbootstrap.com/)
* [Tailwind CSS](https://tailwindcss.com/)
* [Bulma](https://bulma.io/)
* [Materialize](https://materializecss.com/)
* [Semantic UI](https://semantic-ui.com/)

CSS preprocessors like [Sass](https://sass-lang.com/) and [Less](https://lesscss.org/), allow you to add logic and functionality to your CSS. These tools make your CSS clean and easy to work with.

### JavaScript libraries and frameworks

Just like with the CSS libraries and frameworks, there are many options for JavaScript.

It is not necessary to learn them all. Same as above, research job postings in your area to see what libraries and frameworks are being used.

Here are some popular options:

* [React](https://reactjs.org/)
* [Angular](https://angular.io/)
* [Vue](https://vuejs.org/)

These frameworks and libraries allow you to save time and do more with less code. It is possible to get a job specializing in React, Vue, or Angular.

### Testing and Debugging skills

As you are developing your application, there will be errors in your code that need fixing. Debugging is the act of identifying those errors ("bugs") and fixing them.

Testing is another important skill to learn. Writing tests for your code is a way to ensure that your code is doing what it is supposed to do.

### Version control

Version control is a way to track and manage changes to the project's code. [Git](https://git-scm.com/) is a popular software that is used to track your code.

If you mess up a lot of things in your code, you can use Git to go back to a previous version of your code instead of manually rewriting everything.

Learning Git also allows you to collaborate with others on a team and make changes to the same code base from different locations.

I would suggest starting to [learn Git](https://www.youtube.com/watch?v=RGOj5yH7evk) and using a service like [GitHub](https://github.com/) to host your personal projects.

### Problem Solving

The most important skill for any developer is knowing how to problem solve. Companies and clients are looking for you to provide solutions.

It is important to learn how to tackle a problem, break it down into smaller manageable pieces, and troubleshoot the issue in these web applications.