2022 Front End Developer Tools

11 Front End Developer Skills You Need to Get Hired in 2022



Think about one of your favorite websites. Maybe it has exceptional functionality, or perhaps you love it for its aesthetic characteristics. The joy and ease you experience on that website are largely thanks to the fruits of a front end developer’s labor.

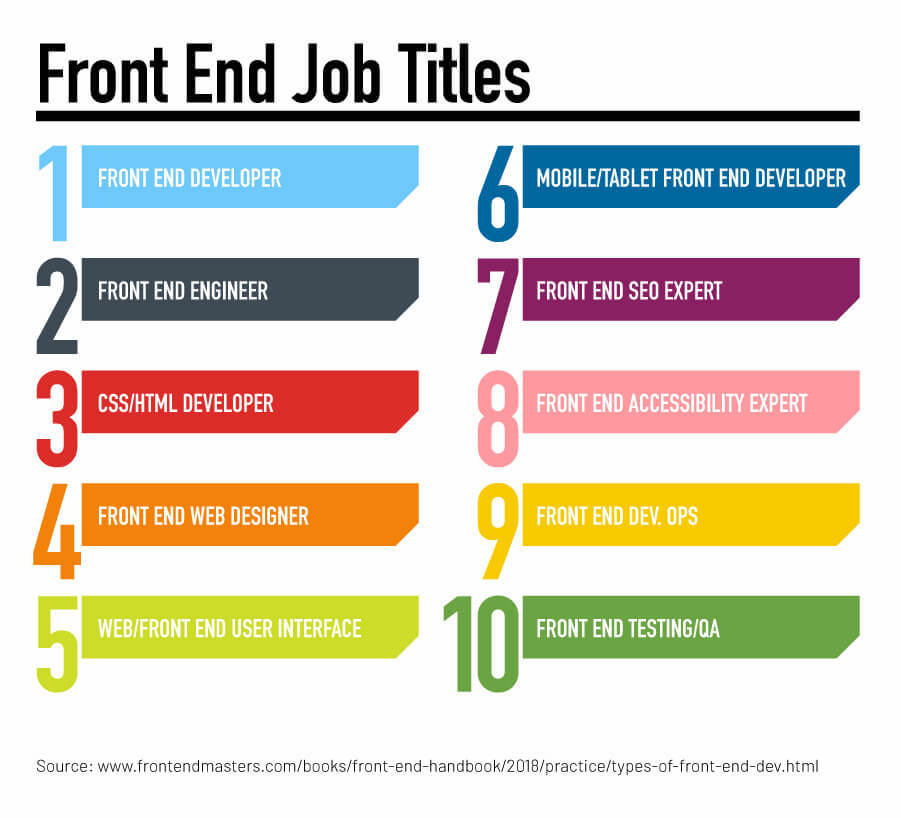
There are different types of web developers, all of whom participate in the coding, analysis, and maintenance of the websites and applications that drive businesses and organizations, from nonprofits to major e-commerce projects. While a back end developer might build the infrastructure that enables a website to handle queries, data requests, and innovative search structures, a front end developer makes all of that accessible and convenient for the site’s users. The front end developer skill set, in particular, focuses on the user environment and experience.

**11 Skills Front End Developers Need**

1. HTML
2. CSS
3. JavaScript
4. DOM Manipulation
5. JSON
6. AJAX
7. Node.js
8. MySQL
9. React
10. Java
11. Problem-Solving

However, front end developers aren’t just designers; they are programmers who have skills in HTML, CSS, and JavaScript, as well as some of the powerful frameworks and libraries that can bring a user-focused website to life. Technologies like Foundation, AngularJS, Bootstrap, Backbone, DOM, and EmberJS help to ensure that a site looks and works well on [any type of device](https://dzone.com/articles/comparison-of-js-frameworks-angularjs-vs-reactjs-v), and coding libraries like LESS and jQuery help the site’s users save time. Front end developers use these platforms creatively to build better, more beautiful experiences for website and application visitors everywhere. They also look for ways to improve and maintain the user experience — especially as technology shifts and users develop different demands and needs.

As companies have shifted more of their attention online, the demand for front end development skills in the job market has grown swiftly. The reason is simple: those skills are instrumental to a wide range of web-related jobs, including app development, web integration, and web design. Because customers want to access websites and apps that are fast, responsive, and easy to use, businesses in almost every industry — from healthcare to education to financial services — seek out great front end developers who can make their user experience a winning one.



While front end developers are often involved in site and app design, they are really programmers at heart. Practical experience matters, but some type of formal training is also essential to landing the programming chops and front end developer skill set necessary to create a great user experience.

A bachelor’s degree in computer science offers a strong foundation, but many programmers choose to build up their front end developer skills by attending a [coding boot camp](https://bootcamp.ce.arizona.edu/coding/). These boot camps provide [specialized programs of instruction](https://www.chronicle.com/article/Tech-Skills-Boot-Camps-Are-on/247314) that include programming skills and introductions to a wide range of languages and libraries, and they typically offer a faster track for acquiring the necessary skills to get career-ready.

In either case, there’s no avoiding the obvious: you will need a specific front end developer skill set to succeed in your new career. Below are some of the most important technical skills that you should learn if you plan to excel in front end development.

**1., 2., 3. HTML, CSS, and JavaScript**

Let’s discuss the fundamentals. [HTML, CSS, and JavaScript are the three foundational building](https://blog.hubspot.com/marketing/web-design-html-css-javascript)blocks that comprise just about every website available. As a front end developer, you’ll use programming languages, which instruct computers on how to display content, retrieve information, and respond to user input. Every website is built via an array of interlocking instructions and languages that direct the visual experience and every user interaction.

When the web first was developed, HTML, which stands for HyperText Markup Language, was the only language used on the internet. To this day, it still provides the [basic structure](https://developer.mozilla.org/en-US/docs/Web/HTML) for websites everywhere. [CSS](https://developer.mozilla.org/en-US/docs/Web/CSS), on the other hand, is used to construct a website’s layout, format, and design. HTML builds the structural underpinnings of a website, while CSS gives it its appearance and design. Technically speaking, HTML and CSS are markup instructions, rather than programming languages. JavaScript, however, is another story.

JavaScript, a dynamic scripting language, allows you to make websites interactive and build mobile apps. [JavaScript](https://developer.mozilla.org/en-US/docs/Web/JavaScript) is a lightweight programming language that interacts with Application Programming Interfaces (APIs). This capability helps front end developers create complex interactions quickly and simply on their websites, thereby providing an ideal user experience.

JavaScript was also built to enable DOM manipulation, which changes the way that a user views a website based on their interactions. Today, JavaScript lies at the core of a wide range of scripting technologies and runtime environments that drive powerful applications and fast, responsive sites.

HTML, CSS, and JavaScript are different languages, but they are often linked together as the core front end development skills you need to begin creating great sites.

**4. DOM Manipulation**

The [Document Object Model (DOM)](https://developer.mozilla.org/en-US/docs/Web/API/Document_Object_Model/Introduction) provides a programming interface for any document on the web. It represents the structure and the content of any web document, including its styles and HTML sections.

Through manipulating and interacting with the DOM, you can change the way the content on any page displays for the user. The DOM turns each part of the website into nodes and objects; [this transition allows developers to interact with the model](https://developer.mozilla.org/en-US/docs/Learn/JavaScript/Client-side_web_APIs/Manipulating_documents) using JavaScript or any other object-oriented programming language. The DOM contains several APIs that will enable front end developers to change any user’s experience, from displaying content in their preferred language to allowing them to interact with a site via a webcam. If you want to build interactive websites where users can store their information or direct their digital experience, DOM manipulation is an essential front end developer skill to add to your arsenal.

Many technologies used in front end web development build on the basics of DOM manipulation to change what users see to reflect their preferences and needs. Your users aren’t one-size-fits-all, so DOM manipulation allows you to create web experiences that are personalized and responsive.

**5. JSON**

Derived from JavaScript, JSON is a data format known as JavaScript Object Notation. It provides a text-based representation of web data and has become almost inescapable on the web over the past 15 years. Many of the databases that underpin websites — think MySQL and PostgreSQL, for example — have core native support for JSON queries and data.

This data format is designed to be used by a wide array of programming languages and to pass information easily between users and your website, whether that data will be stored or further processed. [JSON is essential](https://www.infoworld.com/article/3222851/what-is-json-a-better-format-for-data-exchange.html) for any site where you need to transfer information between your users and the website’s back end.

<https://www.json.org/json-en.html>

**6. AJAX**

AJAX, which stands for Asynchronous JavaScript and XML, brings together all the core front end developer technologies to communicate information between a client and a web server.

AJAX helps a client — the web browser a customer is using, for example — make a request to the server. Then, it allows the server to convey a response without forcing a reload. Because AJAX is asynchronous, the site doesn’t freeze up until the server replies; instead, it works behind the scenes to process a JavaScript request and update the DOM to reflect the server’s response to the user’s requests.

Many of the most popular [social media sites](https://webdesign.tutsplus.com/tutorials/an-introduction-to-ajax-for-front-end-designers--cms-25099) that drive web traffic today, like Facebook and Twitter, rely on simple AJAX-based interactions to show more content and updated information as users scroll up and down through their timelines. In most cases, AJAX is so firmly woven into the framework of popular websites that it illustrates how most users expect their sites to work: proactively and without requiring extra effort.

**7. Node.js**

As a front end developer, you want to give users the best experience possible. That means making it easy for customers to use your website without subjecting them to unnecessary waits and delays — and ensuring that your site can scale up to be used simultaneously by thousands or even millions of people. By [using Node.js as a runtime environment for JavaScript](https://www.infoworld.com/article/3210589/what-is-nodejs-javascript-runtime-explained.html), front end developers can minimize delays and process a high quantity of requests, thus providing a smooth experience for every user.

Node.js is one of the most popular technologies used in back end web development, but its uses aren’t limited to back end developers. While Node.js provides an excellent framework for writing applications, it also offers useful minimalism and speed that can empower an exceptional user experience.

As a front end developer, you may [write some front end apps](https://www.netguru.com/blog/node.js-in-front-end-development-youre-already-using-it-every-day) or use bundlers and task runners to provide a speedy, seamless experience for your users. All of these technologies are typically built on JavaScript and then executed using Node.js. Even alternative scripting languages and CSS transformation tools like PostCSS are based on JavaScript. Node.js gives you the power to create a speedy, JavaScript-based framework that minimizes server load and user waiting time.

**8. MySQL**

MySQL is a popular system that manages the database structures that websites use to organize their information. Content management systems like WordPress, for example, often rely on a MySQL database to [store and relay information](https://kinsta.com/knowledgebase/what-is-mysql/) to users, including all of the posts, articles, and user data on the website.

Though there are several database options available to front end developers, MySQL — developed and supported by Oracle — is one of the most popular open-source relational database management systems used across the web. As a relational database, MySQL uses an array of tables to define and organize information while retaining a clear record of its connections and relationships.

MySQL is primarily a back end developer tool because it is often part of the foundation on which back end developers construct the user experience. However, interaction with the back end database is one of the most important responsibilities for any front end developer. When you write an app for use on the front end, it is built on the structure that stores, retrieves, edits, and saves data. Every search and user login on the front end of your website may well involve interacting with the MySQL database, so you need to know and understand how the database works and how best to interact with it to deliver a smooth, seamless user experience.

**9. React**

React is a JavaScript library that enables front end developers to build user interfaces. Front end developers can use React to describe or declare how they want the DOM to be changed and the user interface to appear. It is similar to HTML in that sense, but the React library also gives you the power to display dynamic, changing data.

Essentially, React’s value as a JavaScript library is described in its name; it allows fast, dynamic, and user-friendly adaptations, or *reactions*, based on how a user interacts with the website. React automatically updates the DOM and responds to the user’s requests. There are several JavaScript technologies that you may use as a front end developer, but React is highly popular because it can use JavaScript to create HTML, rather than enhancing and refining HTML that already exists.

React is particularly helpful for developing [applications that live on the web](https://flaviocopes.com/single-page-application/) but work like full-featured desktop programs. Consider advanced webmail programs like Gmail or mapping software like Google Maps. These load within one single webpage but are continually changing for instant interaction and feedback. Websites that offer single-page applications often rely on React for a smooth user experience and fast, seamless responses.

**10. Java**

Java may sound like JavaScript, but they are two completely separate programming languages used for very different tasks. While JavaScript typically makes websites more interactive and responsive, providing the excellent user experience that defines front end development, Java is a programming language more frequently used by back end developers.

Java is popular for several reasons. First, it’s ubiquitous; a Java-based program will run on any computer that has Java Runtime Environment (JRE) installed — and most computers do. Windows, Macintosh, Linux, and even most mobile operations all run JRE, making it easy for the vast majority of clients to execute a JavaScript program. This makes it the ideal language to develop mobile apps, financial instruments, and web enterprise services.

The second reason for Java’s popularity is its design. Java is an object-oriented programming language, which means that it uses programming “objects” to represent and define behavior for both real-world objects and abstract concepts. Doug Lowe and Barry Burd provide an example [in “Java for Dummies”](https://books.google.com/books?id=QEj6CFFdN4sC&pg=PA12&lpg=PA12&dq=Suppose+you%E2%80%99re+writing+a+payroll+program.+This+program+needs+objects+to+represent+the+company%E2%80%99s+employees&source=bl&ots=bn2kbXyqHA&sig=ACfU3U2jr87OC_nUG0VY5E5vhNQsekaXFw&hl=en&sa=X&ved=2ahUKEwjk9ei245zpAhU7l3IEHaICD0YQ6AEwAHoECAgQAQ#v=onepage&q=Suppose%20you%E2%80%99re%20writing%20a%20payroll%20program.%20This%20program%20needs%20objects%20to%20represent%20the%20company%E2%80%99s%20employees&f=false): “Suppose you’re writing a payroll program. This program needs objects to represent the company’s employees. So, the program includes a class (probably named Employee) that defines the data and methods for each Employee object. Then, when your program runs, it uses this class to create an object for each of your company’s employees.” By creating an “object”, Java developers can thus define payroll activity for each employee.

It’s an undeniably useful skill for both back end and front end developers alike.

**11. Problem-Solving**

Of course, problem-solving isn’t a technical skill like HTML, CSS, JavaScript, or their many implementations and libraries — but it is essential to front end development nonetheless. As the experts say, “Being good at solving non-developer problems often means you’re good at [solving code problems](https://gomakethings.com/what-makes-someone-a-good-front-end-developer/), too. So much of good problem solving is really just breaking a big thing into smaller parts and tackling them one at a time.”

Luckily, you already have practice solving problems, whether at home, school, or work. With a little effort, these experiences in other arenas can translate over into your efforts as a front end developer. Whether you’re getting your bachelor’s in computer science or attending a [coding boot camp](https://bootcamp.ce.arizona.edu/coding/), your ingenuity and willingness to tackle problems will help you in your explorations as a front end developer.

You’ll need the technical skills mentioned above for your career, but the foundations of front end development are much more universal: endless curiosity, a willingness to experiment, and critical thinking. Whether you’re just starting or want to expand your career options, front end development holds boundless possibilities for aspiring programmers.

It’s almost impossible to imagine today’s world without the web, which indicates just how vital front end developers are to the business and social worlds we interact with every day — from shopping on Amazon to catching up with friends on Facebook, Twitter, or Instagram. Explore your educational options, and you, too, can open the door to this growing, challenging field.

1. HTTP - You absolutely need to understand the client and server paradigm via HTTP. You won't be able to program in any server side technology stack if you don't understand the basics of it. I'm not asking you to understand the entire spec but just the verbs like GET, POST, PUT, DELETE and REST based mechanism to invoke code on the server side.

2. Knowledge of javascript - This is an absolute must otherwise you can't even program in Node.js :) It's server side javascript thus you need to be familiar with the nuances of the programming language and its model.

3. Event Loop - What is it? Why should it be used? What's the benefit of it? One you understand the semantics of it then you would easily be able to write Node.js programs

4. Async programming - It's a very different programming model. You need to understand the concept of delegates which would be invoked once a routine has been successfully executed or failed to execute. Without understanding the async model, you wouldn't be able to write efficient code.

That's all :)

I would also like to share the online book that helped me tremendously in learning Node.js. Hope you find it useful as well - [The Node Beginner Book](http://www.nodebeginner.org/)