

# Ultimate Front-End Web Developer Career Guide

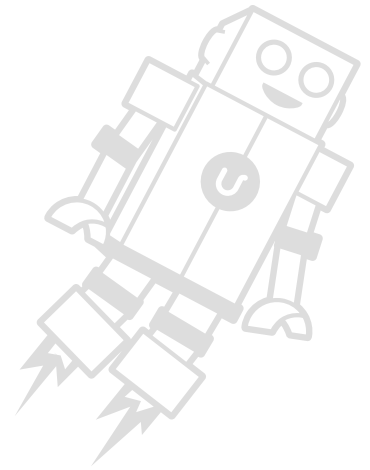


# Introduction

So you want to be a front-end web developer? Congrats! You've chosen a fantastic field. Soaring demand, strong salaries, and geographical flexibility make web development a super appealing career path right now.

But that path can seem complex and bewildering when you're just starting out. The purpose of this guide is to get you ready to navigate entry into the field of front-end web development. It'll explain why front-end development is such a solid career choice, offer insight into what typical front-end devs do, teach you what skills you'll need to get down pat, and demonstrate how to stand out from the rest of the developer pack.

In short, consider this your ultimate guide to getting started as a front-end web developer.



# Front-End Web Developer Career Guide: What We'll Cover

<b>Get Inspired: What Is Front-End Web Development and Why Is It a Great Career?</b>	<b>04</b>
<b>Skill Up: What You Need to Learn to Be a Front-End Developer</b>	
<b>Hard Skills</b>	<b>06</b>
<b>Soft Skills</b>	<b>07</b>
<b>Navigating Failure</b>	<b>08</b>
<b>Stand Out: Front-End Development Career Prospects and Growth</b>	<b>09</b>
<b>Resume Building</b>	<b>10</b>
<b>Personal Branding</b>	<b>11</b>
<b>What Next? Learning Resources</b>	
<b>Udacity Nanodegrees</b>	<b>12</b>
<b>Udacity Courses</b>	<b>13</b>
<b>Other Resources</b>	<b>14</b>
<b>5 Steps to Get Started Today</b>	<b>15</b>



# Get Inspired: What Is Front-End Web Development and Why Is It a Great Career?

Front-end developers create the user-facing side of websites, also known as the “front end.” That’s everything that you see when you’re navigating around the Internet, from fonts and colors to dropdown menus and sliders. Front-end developers construct static pages like an Amazon search results list, as well as dynamic web applications like Twitter, Facebook, and Google Docs, using three main languages: HTML, CSS, and JavaScript.

So, front-end devs are responsible for the development of user-facing code, yes; but that’s far from all. They also are charged with optimizing web applications to maximize speed and scalability, creating fast, easy-to-use applications and developing prototypes for those applications quickly, and generally building amazing, immersive user experiences.

It’s a powerful feeling to directly affect an end user’s experience. Mikey Ilagan, a front-end developer with eight years of experience, said, “I love being able to make an impact on the user interface, the aspects of an app or website that the user interacts with and sees.”

Moreover, front-end development is hyper-creative work. As Greg Matranga, Director of Product Marketing at Apptix, said of the team of developers he oversees, “The developers that work on the front end are... excited about what they do because they’re really able to leverage their creativity.”



Front-end devs work closely with web designers and other digital creators to craft the sublime interactive experiences that make the web the addictive environment you know it as. On a typical website, a designer crafted the logo and graphics, a photographer took the pictures, and a copywriter wrote the text. But a front-end developer assembled all of those pieces, translated them into web-speak, and built the experience you have with each page.

Front-end developers also work closely with user experience analysts to bring mockups, or wireframes, from development to delivery. In addition, it's important for developers to be able to fluidly partner with other teams across the business to understand specific goals, needs, and opportunities, and then to execute on those directives.

Front-end devs' counterparts are back-end developers. The back end of a website consists of a server, an application, and a database. Back-end devs are responsible for building and maintaining the technology that powers the websites and apps a front-end developer makes.

Looking to the future of the industry, as the tech that powers websites and web applications grows increasingly complex, the best front-end devs will need to understand exactly what the back-end developers are doing, and why, and to be able to work ever more closely with them so that, for example, their applications don't generate requests that drive the database bonkers.

A role increasingly prevalent in the industry is the full stack developer, a title popularized four years ago by Facebook's engineering department. You've probably heard of it, or read about it, but may not be quite sure what full stack devs do compared to front-end and back-end pros. The idea is that a full stack developer can work cross-functionally on the full "stack" of technology, i.e., both the front end and the back end. That makes full stack developers responsible for the entire flow of user experience, from load time and layout to interactivity and structural underpinnings.

# Skill Up: What You Need to Learn to Be a Front-End Developer

## Hard Skills

Front-end developers must be adept at three main languages: HTML, CSS, and JavaScript. In addition to fluency in these languages, front-end devs should be familiar with frameworks like Bootstrap, Foundation, Backbone, AngularJS, and EmberJS, which ensure great-looking content no matter the device, and libraries like jQuery and LESS, which package code into a more useful, time-saving form.

Front-end developers also use responsive design principles to make websites that scale and adjust themselves to offer compelling experiences regardless of device, and they use principles of browser rendering like critical rendering path, image optimization, and JavaScript minification to ensure fast, efficient websites. There's nothing more annoying than waiting for an app or site to load, not to mention if it renders funky, right? A lot of the responsibility of making sure users don't lose patience and navigate away falls on the shoulders of front-end devs.

As far as other skills and tools, browser developer tools like application profiling and element inspecting help front-end developers troubleshoot bugs in their code. Version control software like Git helps save and maintain that code. Lastly, a lot of front-end developer job listings call for experience with AJAX, a widely used technique for using JavaScript that lets pages dynamically load by downloading server data in the background.

For a comprehensive list of hard skills crucial to success as a front-end developer, check out Udacity's definitive [Front-End Web Developer Skills Checklist](#), which includes 17 pages of essential skills and tools to master and development concepts to understand.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Awesome Site</title>
    <link rel="stylesheet" href="css/style.css">
  </head>
  <body>
    <div class="top">
      <div class="container">
        <div class="three columns info-top">
          <p class="font-icon">q</p>
          <p>(123)-456 -789-00</p>
        </div>
        <div class="four columns info-top">
          <p class="font-icon">w</p>
          <p>yourmail@company.com</p>
        </div>
      </div>
    </div>
  </body>
</html>
```

## Soft Skills

At the same time, as you sharpen your technical abilities, it's just as important to nurture your soft skills. Often, the differentiating factor in landing a job isn't how clean your code is—it's your interpersonal competence. In fact, in a recent [CareerBuilder survey](#), 77 percent of employers said soft skills are as important as hard skills in getting the job offer. One in five said soft skills are actually more important.

Be assured, soft skills like attention to detail and deft communication will stand you in great stead on top of your technical skills in your search for work as a developer.

**Strong Communication:** You may communicate handily in HTML, but if you don't speak the same social language as your manager, your peers, and your clients, you'll be at a loss. You're going to want to be diligent about delivering progress updates to your boss, to be open to feedback, and to be industrious in applying it. Always work to provide your peers and direct reports with the right information, in the right way, at the right time, just as you'll do with your end users. And never leave clients in the dark: keep them apprised of your progress at preestablished intervals, and don't forget to translate industry jargon into English.

**Agile Problem-Solving:** The most technically capable developers wouldn't be where they are in their careers today without powerful problem-solving skills. That means troubleshooting by breaking down larger problems into smaller, more easily solved pieces, as well as figuring out the question behind the question, or working with your client or colleague to detect exactly what their ask is. Lastly, consider the larger picture in every move you make on a project. Avoid the tunnel vision of zeroing in only on your individual role, which can blind you to pain points or potential gems that are peripherally related to your work, but still crucial.

**Healthy Passion:** If you're not obsessed with it, why are you doing it? As you look toward your future in the industry, aim to stay plugged-in, knowledgeable, and curious: there's never an end to learning. Don't ever stop investigating, exploring, and asking questions. Use your free time to fiddle around with side work. It's the best way to perfect your skills without the perils of screwing up a high-profile project. Proud of something you've been tinkering with? Share it on GitHub.

**Self-Starting Motivation:** The top tier in any field is composed of self-starters. Take the initiative to ask questions, then digest the answers and act. But don't be discouraged when you hit roadblocks. Spend time probing the issue, take a breather to get some perspective, or turn to a trusted colleague or mentor.

These soft skills are just as important when working on a project for an online course or a freebie for a friend as when you're staffed on a high-profile assignment at a full-time job. You know that Gandhi quote, "Be the change that you wish to see in the world"? Be the developer you'd wish to work with in the industry!

## Navigating Failure

Once you've got the hard skills and the soft skills down, you're good to go, right? Well, there's one more ability you'll want to cultivate as you navigate the world of web development: how to handle failure. As you've no doubt gathered, there's a ton to learn, even more to practice, and a plethora of resources to wade through when it comes to learning how to be an excellent developer. The job requires ample time, patience, and fortitude. You could spend hours looking for one errant semicolon in your code, or slogging away conducting endless Google or GitHub searches to find what you're looking for. Your months-long project could end up getting rerouted to a different team by your boss or client, or an idea you had for a cool feature could be squashed before you feel it's even had a fair hearing. Being able to handle the grind without getting discouraged when things look tough is paramount to becoming successful as a front-end dev.

Cultivate patience and perseverance by keeping the end goal in mind on a project, rather than letting yourself feel bogged down by the nitty gritty, and by swapping stories and ideas with fellow developers. Take breaks when you need to come up for air. Log your achievements at the end of each day, so you can see a visual gauge of your progress.

In the end, it's all absolutely worth it. And here's why.



# Stand Out: Front-End Development Career Prospects and Growth

There's never been a better time to pursue a career in web development. Employment of web developers is [projected to grow](#) 20 percent over the 10-year span from 2012 to 2022, faster than the average for all occupations.

And front-end web developers have one of the most bankable skill sets on the job market today. According to [Glassdoor](#), the national median salary for front-end developers tops more than \$75,000. In markets where the competition for top tech talent is fiercest, the median salary can climb above \$100,000.

But not all job markets are created equal. Location plays a major role in front-end developer salaries. In fact, according to data from [Payscale](#), geographic location is the biggest factor affecting pay for front-end web developers, even more than experience.

The seven cities with the highest average salary for developers are, from most to least dollars above average: San Francisco, Dallas, New York, Los Angeles, Washington, D.C., Atlanta, and Chicago. The lowest: Charlotte, St. Louis, Minneapolis, and Denver.

That said, if your desired hometown isn't brimming with front-end developer jobs, know that there are more opportunities for contract, freelance, and remote work than ever before. You've got the freedom to determine what works best for your goals, lifestyle, and experience.



## Resume Building

So, the jobs are ample and the payoff is legitimate. But how do you go about attracting work?

First off, keep building your resume and portfolio as you learn. You'll want to use your collection of work as an opportunity to reflect your personal brand and to cleanly showcase your best stuff. If you're not sure exactly what to include, remember that anyone viewing your portfolio will likely make a snap judgment, based on first impressions—you'll want to give them a feel for the full spectrum of projects you've done, so feature a curated but wide range of work. GitHub, Behance, or a beautiful personal website are all strong options for housing your portfolio.

No professional projects to your name yet? It's OK. You can include something stellar from a course or workshop. It can also be helpful, once you've got the skills down, to offer to complete some work for free, or for cheap, for friends and acquaintances, in order to secure some strong work examples.

Another solid tactic to start developing your career is to embark on informational interviews and network your tail off. Your Rolodex, after all, is your most powerful tool in your hunt for good work. Sometimes it really is who you know, rather than what you know, that can land you the dream job. And having the right professional network at your fingertips can expose you to more job opportunities than if you were trying to land a gig alone. Just remember that professional relationships are a two-way street. Sometimes the best way to ensure you'll get the most out of one is to do the assistance yourself, first: doing someone a favor by making an introduction or offering to review their e-book on Amazon, for example, is money in the networking bank for the future.

You can also rub virtual elbows with others in the industry on Twitter and LinkedIn. For instance, put Twitter's search engine to work, regularly searching for the latest conversations around industry topics, companies you've your eye on, and developers you respect. Join industry Twitter chats. And don't feel creepy about browsing your followers' followers, and the followers of people you're following (say that ten times fast). The best way to find the most helpful people to follow on Twitter is to explore who's following, and being followed by, users you already admire.

## Personal Branding

Speaking of online networking, one of the first things recruiters or hiring managers are likely to do when considering you as a job candidate is Google you to see what you're all about. Investing the time and energy into building your brand online is essential in landing a good job.

Here are three handy quick tips to cultivate a strong personal online brand.

**Google Yourself:** Since recruiters will be Googling you, Google yourself to see what comes up and where. Consider having a trusted friend or colleague offer an external set of eyes for the same search. Put your first and last name together in quotes and make sure you're logged out of your Google Apps account and using private browsing so the search engine doesn't skew results toward links you've clicked regularly. Pinpoint any search results that wouldn't do you any favors in the job recruiting process, and any glaring omissions you think you should fill (for example, evidence that you network online with others in the industry, proof that you're a competent writer, and documentation of projects you've worked on). Going forward, only publish things to the Internet that you wouldn't mind a recruiter seeing.

**Streamline Your Digital Identity:** Ensure that your usernames, profile photos, and bios are polished and as uniform as possible across every social network or online community. Be certain that the information included is correct and that it does justice to your experience, background, and skills.

**Optimize Your Posts:** When's the last time you contributed on GitHub, posted on your blog, added new content to your portfolio, recommended an article, or posted in a group on LinkedIn? Strengthen your online presence with regular updates on industry-relevant sources. When it comes to social sites that you may use for personal rather than professional networking, like Facebook, Instagram, and Pinterest, be cognizant of the fact that much of the content you post there can still be surfaced online to others. Also consider filtering in professional content when it makes sense. That might mean building a Pinterest board of impressive web design examples, participating in a Twitter chat with other developers, or posting articles on Facebook about the latest trends in web development.

# What Next?

## Learning Resources

### Udacity Nanodegrees

Check out Udacity's [Front-End Web Developer Nanodegree](#) for a structured program to help you learn all the essential technical skills of the trade with the support of coaches and fellow students. Nanodegrees are a new type of credential, designed to prepare you for a job. Each of the programs varies in length from 6 to 12 months, based on a time commitment of 10 hours per week.

In the Front-End Web Developer Nanodegree, you'll work your way through six projects designed to teach you the web development fundamentals as you build a portfolio that will demonstrate your new skills to leading employers. You'll also seriously advance your career prospects by learning how to ace a technical job interview and by participating in code reviews. Think of it as your action plan for embarking on a full-fledged career as a front-end developer.

Want more? The brand new [Full Stack Web Developer Nanodegree](#) is the most efficient curriculum to prepare you for a job as a full stack dev. You'll learn to build complex server-side web applications that make use of powerful relational databases to persistently store data. You'll then learn how to secure and configure your very own Linux server to host your applications. Finally, you'll explore the challenges in horizontally scaling an application to support thousands of users with a cloud-based application hosting provider.

You can also check out the [iOS Developer Nanodegree](#) if you're intrigued by mobile development. You'll learn about MVC architecture and why it's important when developing apps, how to build sophisticated user interfaces with UIKit components, how to implement Core Data for local persistence of an object structure, and how to use external APIs to make use of data from other companies.



## Udacity Courses

If you're in search of a learning plan with less of a time commitment, or if you're looking to fill a specific gap in your skill set, check out our individual web development-related courses:

[Intro to HTML and CSS](#): Learn how to convert digital design mockups into static web pages and how to build a responsive portfolio site to showcase your work.

[JavaScript Basics](#): Learn JavaScript syntax and coding conventions that web developers use to create interactive and dynamic websites while you create an online resume.

[Intro to jQuery](#): Learn how to access and modify the DOM with ease using jQuery's core features—DOM element selections, traversal, and manipulation.

[Object-Oriented JavaScript](#): Build a variety of JavaScript objects and explore how their inheritance models affect your app's in-memory model. Gain simplicity and modularity in your own code.

[HTML5 Canvas](#): Learn how to use HTML5 Canvas to create and modify images and interactive animations.

[Website Performance Optimization](#): Learn how browsers convert HTML, CSS, and JavaScript into websites, while you experiment with Chrome Developer Tools to measure and optimize website speed.

[How to Use Git and GitHub](#): Learn how to create a repository and save versions of your project, and get practice using GitHub or other remote repositories to share your changes with others and collaborate on multi-developer projects.

[Intro to AJAX](#): Learn how to make asynchronous requests with JavaScript using jQuery's AJAX functionality, and how to perform asynchronous requests as you build a web app with data from Google Street View, *The New York Times*, and Wikipedia.

## Other Resources

### Friends of Udacity

[Ilya Grigorik Paul](#)

[Lewis](#)

[Paul Kinlan](#)

[Pete LePage](#)

### Books

[Dive Into HTML5 Eloquent](#)

[JavaScript JavaScript Design](#)

[Patterns](#)

### Newsletters

[HTML5 Weekly](#)

[Tales from the Front End](#)

[JavaScript Weekly Hacker](#)

[Newsletter](#)

### Blogs/Websites

[HTML5 Doctor HTML5](#)

[Hub HTML5 Rocks](#)

[Smashing Magazine A](#)

[List Apart](#)

[CSS Tricks DailyJS](#)

[Hacker News](#)

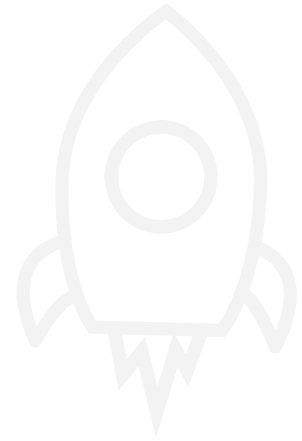
[Slashdot](#)

[Tuts+](#)

[Geeks for Geeks](#)

# 5 Steps to Get Started Today

You made it to the end of the guide—congrats! If you want to further explore front-end development as a career, here are the best five steps to take next.



1. Download Udacity's [Front-End Web Developer Skills Checklist](#) and evaluate what skills you need to learn.
2. Bookmark and start reading these [24 sites](#) to familiarize yourself with, and keep your finger on the pulse of, the web development industry.
3. Talk to your network and make it a goal to land one informational interview within the next two weeks.
4. Plot out a learning plan. The resources section of this guide offers different options for you to explore, including Udacity Nanodegrees, courses, and other resources online and off.
5. Get started! What will your first side project be?

