CAREERFOUNDRY

Full-Stack Web Development Program

CF

To code is to create!

From social media greats like Facebook to the phone alarm that wakes you up for your daily grind, everything we touch nowadays seems to have its weight measured in code. And they no longer take math whizzes to accomplish, either! Let CareerFoundry show you how easy it can be to go from coding-clueless to professional developer as you take a binary-laced journey of discovery through the world of web development. What were you made to create?

There's no other web development program on the market that combines a comprehensive curriculum, personalized mentorship, and student-focused learning quite like CareerFoundry, and our graduates speak for themselves! Read on to find out how it all works.

Contents

1.

Who is the program for and what are the requirements?

No need for math degrees here—if you can type, you can code!

Let CareerFoundry be your guide and safety net as you take that first step into the land of symbols and numbers. Our Web Development Program is the perfect solution for those brand-new to development. We've taken artists, secretaries, and even law enforcement personnel and helped them grow from creating small personal web pages to large ecommerce sites and apps. All you'll need to bring to the table is a computer, an Internet connection, and a healthy dose of enthusiasm—we provide the rest. Surprise yourself at how much you can accomplish!

The ZFU, or Staatliche Zentralstelle für Fernunterricht, is the state body for distance learning in Germany. In order to offer our Web Development Program publicly, it must undergo a rigorous quality assurance and certification process. On successful completion of this process, the program is assigned a unique approval number (7359420) which can be checked against a public register.

2.

What will you finish the program with?

The program takes you through the entire development process, from project planning and design basics to deploying a progressive web app. You'll begin by exploring the role of web developer using HTML and CSS, the building blocks of the web, to create a responsive portfolio website.

As you work through the program, you'll explore every aspect of full-stack web development, from client- and server-side JavaScript to frameworks (Node.js), test-driven development, and progressive web apps. The specific libraries you'll use with JavaScript include React, Angular, Vue, and React Native. The final leg of the program focuses on refining your portfolio and job skills—including Agile project management, teamwork and collaboration, and developer handoffs.

> By the end of the program, you'll know how and when to use industry-standard tools including Atom, GitHub, Node, React, Angular, Vue, and Bootstrap—just to name a few. You'll also have numerous technical skills under your belt such as HTML, CSS, JavaScript, API building, server-side development, test-driven development, progressive web apps, native app development, and more. To complement the technical aspects of the program, you'll explore Agile project management, UX design, teamwork and collaboration, developer documentation, code reviews, and how you can contribute to the tech community. These additions to your skillset will impress employers and clients alike.

By the end of the program, you'll be qualified for jobs in full-stack JavaScript development. Most of our graduates find jobs as junior web developers, but some find jobs in other areas such as web design, project management, product design, product management, frontend web development, software development, engineering, production and software analysis.

What you'll get

- **Regular calls and unlimited messaging** with your tutor and mentor
- **Portfolio reviews** As many mentor calls as you need
- **Job Guarantee** If you don't land a job within 180 days of graduation, we'll fully refund your tuition
- **14-day trial period** Get your money back if the program isn't right for you
- **Flexibly paced** Our 15-hour/week pace can be organised around any schedule
- **Access to our Job Preparation Course** Opt in to the Job Preparation Course for unlimited, lifetime access to a career specialist to guide your job search
- **Lifetime access to our curriculum** Written by expert web developers working in the field
- **Free read-only access to our other career-change programs** Learn more about UX Design, UI Design and Data Analytics for free
- **Active online student community** Find a study buddy through online and in-person meetups

3.

What kind of support is available?

At CareerFoundry, you're never alone! From the moment you start the program, your student advisor will be there to answer your questions, helping you get used to our platform and supporting you as a new CareerFoundry student. You'll also be assigned a personal mentor and tutor. These industry experts are located all over the world and will act as your teachers, cheerleaders, and confidants through every step of your web development journey.

Your tutor

is your program specialist, providing feedback on your daily submissions and answering any questions that come up about individual tasks or instructions. You'll be able to message back and forth with them via our platform every time you submit an assignment. exercise. They know the curriculum backward, forwards, and inside out—and they're ready to support you. As you progress throughout the program, your tutor will provide constructive feedback within 24 hours—highlighting where you can improve, and where you're on the right track.

Your mentor

Your mentor is a seasoned and influential professional that we've hand-picked to provide industry insights, conduct video reviews of your portfolio projects, and give advice to help you forge your new career. You can schedule up to 100 calls with your mentor over the duration of the program, making them your personal introduction to a career in tech. CareerFoundry mentors are also well-versed in how to educate. They've walked the path before you, grappled with imposter syndrome, and learned from their mistakes. Above all, they know first-hand how challenging a career change can be. As senior experts across a myriad of sectors, they provide you with invaluable insights into working in your chosen field and ensure that the projects you choose are aligned with your career goals. Together, you'll create a portfolio that will tell your story—and help you stand out in the job market.

Student Advisors

ensure you have the best possible experience throughout the program. You can message them from your dashboard (within the platform)—they're always happy to answer any questions you have about the administration of your program.

Fellow students

As a CareerFoundry student, you'll be part of an extensive community of fellow students both during the program and after graduation. You can reach out to your fellow students to discuss your program, organize meetups, or find a study buddy via Slack (instant messaging tool). As a part of our graduate community, you'll have lifetime access to our Slack workspace so that you can stay in touch with your fellow graduates.

We encourage students to work with a study buddy throughout the program—we'll even help you find one! Working with a peer will make your studies easier, more successful, and more enjoyable than working alone. Study buddies motivate each other and keep each other accountable, discuss course topics, help each other understand specific concepts, share knowledge, and give each other feedback.

Our students also regularly organize in-person meetups around the world, where they network and share experiences.

4.

How much time do I need to commit to the program?

Our Web Development Program takes you through two journeys—Intro to Frontend Development and Full-Stack Immersion—for a total of seven months of coursework (one month in Fundamentals, six months in Immersion).

The program requires an average of 15 hours per week. Since you're not required to be online at specific times to work on the material, you have a great deal of flexibility in which hours you set aside for study. It's possible to finish the program in less than nine months—some students put aside time specifically to complete their program, dedicating an average of 30-40 hours each week, for a total program duration of 4 months.

5.

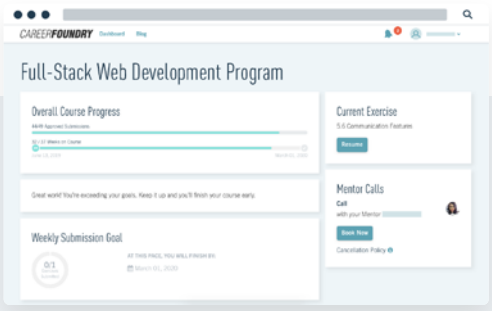
What is the structure and format of the program?

The program provides a 100% asynchronous, online learning experience that is flexibly-paced—so while there is an overall completion deadline, you get to decide how to fit your study hours around other obligations.

Our curriculum is built with the beginner in mind, but it is comprehensive and rigorous—created in-house by our team of curriculum designers, and authored in collaboration with industry experts. The curriculum is hands-on and project-based, meaning that everything you do during the program will contribute to the development of your professional portfolio, which you'll use to showcase your best work to potential employers.

Our learning platform will guide you through the program reading materials, supplementary videos, and assignments, but you'll have plenty of support along the way. Your dedicated tutor and mentor will be there to answer questions, offer guidance, and provide rapid and extensive feedback (as well as final approval) on all of your projects. There is no final exam: the program is counted as complete once all the exercises have been reviewed and approved by your tutor and mentor. You will be able to download your certificate of completion directly from the dashboard.

During the program, we'll send you gentle reminders to keep you on track if you start to fall behind. We want you to succeed, so don't worry if you've struggled to complete online courses in the past—at CareerFoundry, you'll get the support you need.



Get a tour of the learning dashboard on the CareerFoundry platform.

6.

Program Outline

Our Web Development Program is divided into two parts:

Intro to Frontend Development

lasts three weeks (full-time). You'll learn HTML and CSS, the building blocks of the web, as you work on creating your own responsive portfolio website from scratch.

1. **Getting Started with Web Development**
   Discuss the role of a web developer
   Practice project planning with Agile
2. **HTML and the Web**
   Explain the basic structure of the web and how the internet works and create an HTML document
3. **HTML Semantics**
   Practice commenting and code indentation

   Work with hyperlinks to access internal and external pages

   Implement HTML semantics as well as form and graphic elements
4. **Advanced HTML & Web Accessibility**
   Apply advanced HTML attributes for web accessibility

   Use ARIA to create dynamic content
5. **CSS & the Look of Your Website**
   Apply basic CSS styling to a web page
6. **Advanced CSS Layout**
   Create a CSS grid layout for a web page
7. **Advanced Topics in CSS**
   Apply advanced CSS styling such as transitions and animations to a webpage
8. **CSS Preprocessors & CSS Variables**
   Integrate CSS variables to ensure browser support

   Discuss how CSS pre- and post processors work
9. **Git & Version Control**
   Practice version control with Git
10. **Code Quality, Testing, and Web Hosting**
    Utilize code linting to prevent errors and ensure consistent code quality

    Conduct cross-browser testing

Full-Stack Immersion

is the second part of the program where you'll enter the world of functional programming as you take on your first programming language: JavaScript. It consists of six Achievements, based on several tasks each.

Achievement 1 - **Introduction to JavaScript**

You'll use JavaScript to build a simple client-side JavaScript application that can communicate with an external API.

1.1 **What is JavaScript?**
Describe the structure of JavaScript code and the DOM

Practice working with variables, mathematical expressions, and other basic programming language concepts

1.2 **JavaScript Basics Part 1**
Discuss JavaScript primitive and complex data types as well as how and why they're used

Practice creating and adding data to arrays

1.3 **JavaScript Basics Part 2**
Practice JavaScript basics with conditionals and loops

1.4 **JavaScript Functions Part 1**
Establish the fundamentals of functions in JavaScript

1.5 **Javascript Functions Part 2**
Discuss core principles of functional programminge

1.6 **DOM Interaction**
Create fluid user interfaces through DOM manipulation

Apply event handling for web interactivity and accessibility

1.7 **APIs, Ajax & Asynchronous Behavior**
Use Ajax to send and fetch data asynchronously from an external API

1.8 **Complex UI Elements with JS**
Implement UI patterns such as form validation, modals, or touch interactions in a JavaScript application

1.9 **Introduction to jQuery**
Evaluate JavaScript libraries, including jQuery, for app development

Differentiate libraries, plugins, and frameworks

1.10. **Bootstrap & UI Libraries**
Build responsive application layouts using Bootstrap

1.11. **Performance & Debugging**
Evaluate types of JavaScript performance

Practice debugging JavaScript code

Achievement 2 - **Server-Side Programming and Node.js**

> Discover how you can use JavaScript on the server-side using Node.js. Using your command line, you'll work with web server frameworks, REST architecture, and databases as you build an API from scratch.

2.1 **Intro to Server-Side Programming**

Explain the difference between server-side and client-side development

Navigate the terminal with basic commands

Prepare your developer environment for programming using Node.js

2.2 **Node.js Modules**

Use modules to complete tasks relevant to building an API, such as creating a web server and parsing a URL

2.3 **Packages & Package Managers**

Describe how to use package managers when working on server-side projects

Install necessary packages for building an API with Node.js

2.4 **Web Server Frameworks & Express**

Route HTTP requests using a web server framework

Describe how Express can simplify the process of building an API using Node.js

2.5 **REST & API Endpoints**

Discuss the key principles behind RESTful architecture and CRUD functionality

Define and document endpoints for a complete REST API

2.6 **Relational Databases & SQL**

Describe how a database interacts with a web application

Design a database using a database schema

Create a relational database using SQL

2.7 **Non-Relational Databases & MongoDB**

Create a non-relational database using MongoDB

Compare SQL and NoSQL databases

2.8 **The Business Logic Layer**

Integrate a database with an API using business logic

2.9 **Authentication & Authorization**

Apply principles of authentication to REST API

2.10. **Data Security, Validation, & Ethics**

Modify API to align with data security regulations

Achievement 3 - **Client-Side Programming & React**

> Learn how to use JavaScript to build powerful interfaces for server-side systems. You'll build the frontend for your very own API using cutting-edge tool: React.

3.1 **Intro to Frameworks & Libraries**

Define the principles of MVC architecture

Evaluate the benefits of using different frameworks and libraries

3.2 **Building with JavaScript**

Discuss the benefits of React that make it suitable for your application

Prepare a product for execution using build tools

3.3 **React Components & the Virtual DOM**

Distinguish between States and Props of React Components

Create low- and high-level components using component lifecycle methods

3.4 **Advanced React**

Compare how to use class components and function components throughout a React application

Apply the principles of good design to a React application using UI frameworks

3.5 **Client-Side App Routing**

Compare hash-based and state-based routing

Implement methods of authentication in React apps

3.6 **Introduction to Redux**

Describe the data lifecycle in a Redux application

Prepare your app for conversion to Redux

3.7 **React Redux**

Use React Redux to implement the Flux design pattern in a React app

Achievement 4 - **Testing in the Development Process**

> Explore test-driven development as you build a progressive web app with native-like functionality for your portfolio. You'll also learn complex JavaScript topics such as data visualization.

4.1 **Test-Driven Development & Test Scenarios**

Discuss the core principles and purposes of testing in the development process

Compare test-driven development with behavior-driven development

Generate application specifications to prepare for testing

Translate user stories into project outcomes

4.2 **Serverless Functions**

Analyze appropriate use cases for serverless development for different types of development projects

Create a serverless deployment package

4.3 **Unit Testing**

Analyze use cases for unit tests

Script unit tests for the meetup application

4.4 **Integration Testing**

Analyze use cases for integration tests

Integrate real data from an API into a web app

4.5 **User Acceptance & End-to-End Testing**

Evaluate the pros and cons of acceptance testing and end-to-end testing

Implement acceptance testing in an app

4.6 **Continuous Delivery**

Discuss the impact of continuous integration and delivery practices

Integrate an APM tool into the development of a web application

4.7 **Object-Oriented Programming**

Define the fundamentals of OOP and differentiate them from FP

Implement a feature in your web app using OOP methodology

4.8 **Progressive Web Applications**

Explain the principles behind progressive web applications

Implement progressive functionality into an existing app

4.9 **Data Visualization**

Supplement a web app with data visualization features in the UI

Achievement 5 - **Native App Development & React Native**

> Learn about native JavaScript applications while building an additional project for your portfolio: a native mobile app.

5.1 **Building Native Applications with JavaScript**

Articulate the distinction between native applications and other (web) application models

Establish a framework, layout, and UI elements for a native app

5.2 **Chat UIs & Accessibility**

Implement an accessible chat UI for a native application

5.3 **Real-Time Applications & Data Storage**

Compare different possibilities of implementing real-time applications

Use Firebase to set up a real-time database

5.4 **Storing Data on the Client Side**

Implement client-side data storage in React Native using asyncStorage

Describe possible solutions for storing data on the client-side

5.5 **Communication Features**

Implement communication features into a native application

Implement modern JavaScript APIs into an application

Achievement 6 - **Collaboration & Documentation**

> Master the key skills required to effectively collaborate with a wider product team, such as communicating requirements and unpacking design assets, all the while completing a project in Angular, a popular JavaScript framework.

6.1 **Collaboration & Project Management**

Discuss techniques for collaborating with designers

Discuss Agile project management techniques and common materials

Interpret user stories to determine project requirements and story point

6.2 **Introduction to Angular**

Create a basic app using Angular and TypeScript

6.3 **Advanced Angular Part 1**

Apply advanced Angular concepts to your app

6.4 **Advanced Angular Part 2**

Define the routes your Angular app requires

Wrap up your Angular app

6.5 **Documentation & Handoffs**

Explain the role of documentation in development projects and the hand-off process

Implement code linting and commenting when finalizing an app

Prepare all necessary documentation for an application

6.6 **Providing Constructive Feedback**

Communicate effectively and respectfully when collaborating on development projects

6.7 **Contributing to the Tech Community & Beyond**

Discuss best practices and ethical considerations for developers

Curate project deliverables for portfolio

7.

Which tools will you use on the program?

For the duration of the program, you'll have access to free trials and discounts for all the tools you'll need to complete your projects and get hands-on web development experience.

You'll mainly be working with a text editor. A text editor is often considered to be a programmer's most essential tool, and you'll use it to write and edit your code. We suggest using text editing programs such as Atom, VSCode, and Sublime Text—which are all free to use. Among other tools, you'll also use Google Chrome, GitHub, Repl.it, CodePen, Node.js, PostgreSQL, React, React Native, and Angular.

Google Chrome

A lot of testing, especially on the frontend side of things, can be done within Google Chrome's Web Inspector. This is an invaluable tool for previewing changes you make to your own sites and apps, as well as for taking a peek at how sites on the web are coded.

GitHub

GitHub is a platform for hosting code and collaborating on web development projects. You don't have to be an expert programmer in order to navigate and benefit from the many great features that GitHub offers. Although it isn't free, GitHub does offer free plans for open-source projects. And for the purpose and scope of this course, you can use a free version of GitHub.

Node.js

Node.js is a runtime environment that allows developers to write JavaScript code to interact with a server.

Repl.it

Repl.it offers a free, collaborative, in-browser integrated development environment (IDE) to code in more than 50 programming languages without having to spend time on setup. Our students will be able to use the repls that are integrated to our platform to practice coding right after learning a new concept.

CodePen

CodePen is a bit like a playground for developers. Anyone online can create and share pieces of code for other developers to read and play around with. This is especially useful for those just starting out in development, as it provides a place where you can test out code and immediately see the results.

React

React is a JavaScript library for building user interfaces. It allows you to create interactive UIs directly in JavaScript!

8.

What kind of projects will you be working on?

You'll have the opportunity to add eight new projects to your portfolio. In Intro to Frontend Development, you'll build a portfolio site using HTML and CSS; in Full-Stack Immersion, you'll build a total of six different apps, each with different functionalities, spread across seven projects:

- Build a small web application with HTML, CSS, and JavaScript that loads data from an external API and enables the viewing of data points in detail.
- For a mini-project, build an application using jQuery.
- Build the server-side component of a movie information web app using Node.js and Express. It should provide users with access to information about different movies, directors, and genres. Users will be able to sign up, update their personal information, and create a list of their favorite movies.
- Using React, build the client-side for a movie information web application based on its existing server-side code (REST API and database).
- Build a serverless, progressive web application (PWA) with React using a test-driven development (TDD) technique. The application uses the Google Calendar API to fetch upcoming events for a selected city.
- Build a chat app for mobile devices using React Native. The app will provide users with a chat interface and the possibility to share images and their locations.
- Build a simple web application with Angular.

9.

What will your portfolio look like?

You'll add your work to your own personal website (which you'll build during the program). If you decide to take our Job Preparation course, we'll provide guidance on your website like branding your portfolio to meet your career goals, site navigation, contact information, look and feel, and how projects are presented. We teach students how to use AWS Lambda to host their portfolio site, which is one of the most popular platforms used today (AWS Lamda is the Amazon Cloud for serverless functions).

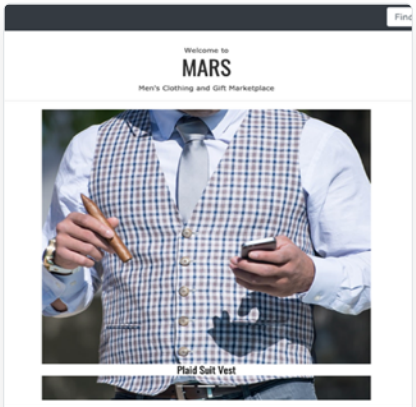
Here are some examples of projects created on the Web Development Program:

Tony Nicola
Web Developer

See project

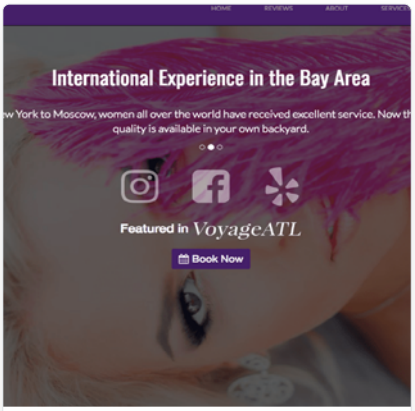
Eveline Kooijman
Web Developer

See project



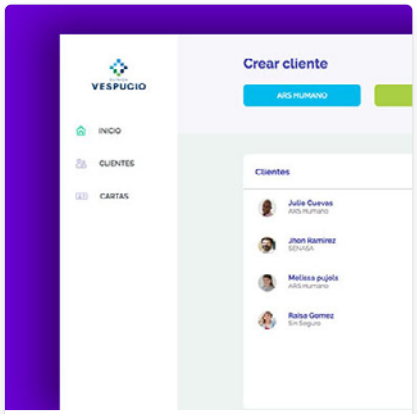
Matthew Primpas
Web Developer

See project



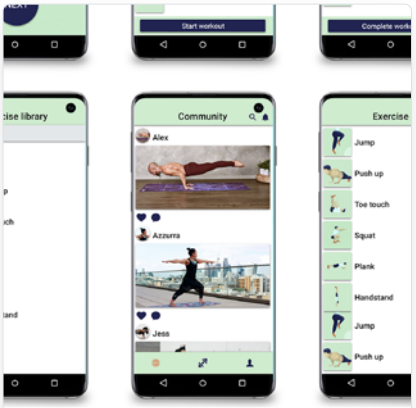
Aron Eidelman
Web Developer

See project



Felix Pujols
Web Developer

See project



Nadia Mohamed
Web Developer

See project

10.

What do you need to do to be eligible for the Job Guarantee?

This list includes what you'll need to be doing throughout the program towards your job search, as well as what you'll need to do or provide if, for some reason, you weren't able to find a job after the job search period.

For more information on the guarantee and job search process, check out our Career Services Brochure .

- ☐ You're at least 18 years of age, and have completed a high school diploma
- ☐ You're located within the list of metropolitan areas in selected countries ovr you're willing to relocate for work
- ☐ You're willing to accept both remote and non-remote positions
- ☐ You've completed 100% of your program, with all Achievements approved by your mentor
- ☐ You've completed the Job Preparation Course within one month of graduating from your main program
- ☐ You have no outstanding CareerFoundry program fees
- ☐ You've updated or created your online professional profiles (such as LinkedIn and Stack Overflow) and they have been approved by your career specialist
- ☐ You've update your resume (CV) with your new skills and it has been approved by your career specialist
- ☐ You've completed at least one call with your career specialist within three months of graduation
- ☐ You've demonstrated that you're actively using your new skills through showcasing a minimum of three completed portfolio projects
- ☐ You've taken the initiative to check in every other week with your career specialist and update them on your progress
- ☐ Proof of at least five job applications per week during the job search period
- ☐ You have a valid work permit for the country you want to work in, and you're able to speak the local language
- ☐ You are unable to find a job in the field within six months
- ☐ You don't have paid work in your new field exceeding 15 hours per week/60 hours per month

For a full list of the requirements that must be met in order to be eligible for a refund, take a look at our Terms and Conditions (under §15) .

11.

Tuition, payment, and scholarships

We offer two payment options:

- A one-time, upfront payment. This option comes with a 5% tuition reduction (i.e. $6,555 rather than $6,900). We also offer payments in GBP, EUR, CAD and AUD.
- An interest-free payment plan. This option requires an upfront deposit to secure your place on the program, followed by eight monthly installments (resulting in payoff before or by graduation).

For further information on tuition and payment plans, or if you'd like to learn about what kind of scholarships we offer, get in touch with us. You can also view our pricing and payment options on our website.

We offer you the chance to try out the program for a period of 14 days. This begins on the program start date. If, within that time frame, you decide that the program is not for you, we'll refund 100% of any tuition you've paid.

If the 14-day trial period has passed and you'd like to withdraw from the program before 60% of the course time has elapsed, you get a prorated refund based on remaining course time and your payment plan details. See the full terms and conditions on our website.

13.

Application and enrollment

We don't require you to go through any long-winded application process in order to participate in a CareerFoundry course or program. We know that our students can change careers successfully, regardless of their background.

> If you'd like to make sure you're familiar with all of the details before signing up, get in touch with us.

If you're ready to enroll, complete your enrollment now**. We can't wait to see you in the program!**