tripleten

Visit our website

# Software Engineering Program

## Syllabus

Online | Part-Time | 10 months

No tech background needed

# Contents

# Software Engineer

10 months

The Software Engineer Program by TripleTen is a 10-month course designed for people with little to no programming knowledge. The program aims to equip you with all the skills needed to land a job in the tech industry. You will learn the fundamentals of coding web applications on the front end and back end, i.e., the parts that happen on the user's computer and in the cloud. You'll use JavaScript and related technologies, such as HTML, CSS, React, Node.js, and more.

You'll also learn how to work with databases and APIs, while getting to grips with a collection of professional development tools, such as Git and GitHub. In addition to the technical skills you need to get started in the tech industry, the program aims to teach the soft skills required for a successful career. You'll learn broadly-applicable skills, such as time management, goal setting, teamwork, and more. The program also covers soft skills specific to the tech industry — such as how to work with documentation — and soft skills that will boost your career by building your online presence.

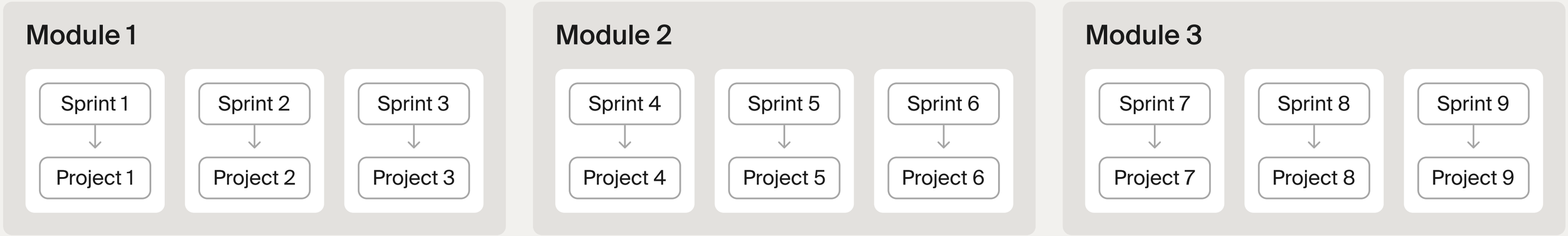
Our ultimate goal is to help you land your dream job. That's why this program also features a variety of career-focused lessons and sub-courses, focused entirely on enhancing your employment prospects.

# Course Structure

Your journey will be structured as a series of sprints, grouped into thematic modules. Each sprint will have a particular set of learning outcomes, reinforced through quizzes and tasks. At the end of the sprint, you will take the skills you've learned and combine them with your existing skills to work on a project that will be assessed by industry experts.

We provide some rough time estimates to help you plan and manage your time, and we recommend spending around 20 hours per week studying. However, we understand that everyone has different commitments and people learn at different speeds. We also understand you may need a break at times, so we have some suggested breaks scheduled in.

- Each sprint lasts 2-3 weeks
- One-week break recommended after each module



# Free Introductory Module

10 hours

This module will take you through the fundamentals of software engineering and cover the basics of HTML and CSS. You'll do this by working through a series of mini-projects on the interactive platform before creating your first independent project.

**Learning outcomes**

- Build a markup structure of websites with HTML
- Style website elements with CSS

# Module 1: Advanced HTML and CSS

7 weeks

In this module, you'll discover just how much HTML and CSS have to offer, as you go beyond the basics and start to take control of your page layout. As your codebase grows, you'll need to keep it organized — here, you'll learn how to do exactly that. Finally, you'll round off this module by learning how to make your pages adapt to a variety of devices.

* One-week break recommended after Sprint 3

### Tools & Technologies

HTML CSS Git Figma GitHub

BEM Chrome DevTools Bash Prettier

## Sprint 1

In your first sprint, you'll learn how to work with layouts using Flexbox and positioning elements. Here, you'll also be introduced to BEM and Git in preparation for later sprints and your projects.

### Learning outcomes

- Control a webpage layout with Flexbox
- Display and position elements using position properties
- Structure the code using the BEM methodology
- Navigate code changes using the version control system Git

**Project**

Landing Page 1 (HTML, CSS)

## Sprint 2

In this sprint, you'll bring your page to life using animations and embedded content. You'll learn how to use forms to gather information from users and you'll build on the BEM and Git skills you learned in Sprint 1. You'll also learn how to debug your web pages. Then, you'll put all of this knowledge to use as you finish off your landing page.

### Learning outcomes

- Manipulate elements visually using CSS transforms and keyframes
- Create markup for user input forms
- Organize file project structure according to BEM rules
- Navigate and use the command line for Git

**Project**

Landing Page 2 (HTML, CSS)

## Sprint 3

Software engineers have to be able to take design specifications and turn them into real products. In this sprint, you'll do just that, bringing Figma designs and spec sheets to life. Your Git skills will get an upgrade as we teach you some expert features. You will also learn how to adapt your interface to a variety of different devices and arrange your page using grid layout.

### Learning outcomes

- Write custom CSS for different screen sizes using media queries
- Build the layout of a webpage using grid columns and rows
- Collaborate with other developers using Git branches and pull requests

**Project**

Social Media Web App Part 1 (Adaptive Layout)

# Module 2: Basic JavaScript and Working with the Browser

6 weeks

This is where programming truly begins. Fully immerse yourself in JavaScript and put theory into practice to create a fully functioning interactive website.

* One-week break recommended after Sprint 6

**Tools & Technologies**

JavaScript HTML CSS Git Figma GitHub BEM Chrome DevTools OOP

## Sprint 4

This sprint kicks off with a crash course in one of the world's most popular programming languages, JavaScript. Often, software engineers will use JavaScript to manipulate the layout of a page, so here we take a closer look under the hood of your browser to give you a better idea about what's going on with the DOM. And once you've laid down your code, you'll no doubt have some bugs; you'll learn how to squash them in the debugging chapter.

### Learning outcomes

- Utilize basic programming concepts in JavaScript
- Manipulate HTML and CSS using JavaScript and the DOM
- Debug the JavaScript code using the developer tools and console
- Resolve Git merge conflicts with commands for managing code history

**Project**

Social Media Web App Part 2 (Vanilla JS)

## Sprint 5

In the fifth sprint, you will dive deeper into the fundamentals of programming, including primitives, conditions, loops, arrays, and functions. You'll build on your new knowledge of the DOM and learn how to use it to add new levels of interactivity to your webpages.

### Learning outcomes

- Manipulate different data structures in JavaScript using ES6 capabilities
- Create HTML content with JavaScript and the DOM methods
- Create event listeners that handle browser events

**Project**

Social Media Web App Part 3 (Vanilla JS)

## Sprint 6

In this sprint, you'll get to grips with more advanced programming concepts, such as objects and events. You'll also learn to work with forms and validate them using JavaScript. The sprint concludes with a bonus lesson for you about debugging in JavaScript.

### Learning outcomes

- Manipulate objects in JavaScript
- Control keyboard and mouse events
- Access form elements and validate user input

**Project**

Social Media Web App Part 4 (Vanilla JS)

# Module 3: Applied JavaScript

6 weeks

At this stage in the program, we'll dive even deeper into JavaScript and tackle the key concepts you'll need in order to become a software engineer.

* One-week break recommended after Sprint 9

## Tools & Technologies

JavaScript HTML CSS Git Figma GitHub

Webpack BEM Chrome DevTools

## Sprint 7

By this point, you'll be able to code some pretty impressive webpages. As with most projects, they'll quickly grow in size and you'll want to break them down into more manageable chunks. In this sprint, you'll dive into object-oriented programming as you learn about creating efficient reusable code.
We'll also show you how to separate your JavaScript into modules so you can reuse code across projects and only import what you need.

### Learning outcomes

- Implement object-oriented features in JavaScript using OOP principles
- Modify markup content inside classes
- Split JavaScript code into usable modules

**Project**

Social Media Web App Part 5 (Vanilla JS)

## Sprint 8

In this sprint, you'll learn how to unpack arrays and objects using destructuring syntax. You'll dive deeper into object-oriented programming as you learn more about interfaces. Then, you'll learn how to use webpack to bundle your code.

### Learning outcomes

- Manage objects and arrays using destructuring syntax
- Create HTML content with JavaScript and the DOM methods
- Configure webpack for automating build tasks

**Project**

Social Media Web App Part 6 (Vanilla JS)

## Sprint 9

By Sprint 9, you'll be a whiz at JavaScript. At this point, you'll be ready to learn another advanced technique: asynchronous programming. You'll then learn how to work with APIs to pull in data from sources all over the web. Finally, you'll start to prepare for interviews with our first lesson in the interview prep series, where you'll dive into object prototypes.

### Learning outcomes

- Write asynchronous JavaScript code with promises
- Make HTTP requests to a server using the Fetch API
- Update markup using fetched data

**Project**

Social Media Web App (Vanilla JS) Part 6

# Module 4: Creating an Interface with React

4 weeks

Get acquainted with the React library and its ecosystem. You'll start off by creating websites made up of components that can be reused in different places. You'll then be able to build on this to create more complex interfaces much faster and write code that's much easier to maintain.

* One-week break recommended after Sprint 11

### Tools & Technologies

React JSX Git GitHub

JavaScript HTML CSS Figma

## Sprint 10

Sprint 10 is where you'll get your first taste of React. You'll be introduced to the basic concepts of the React library, learn about the various tools you can use with React, and then take a look at React Hooks, a useful alternative to class components. You'll also dive into the "this" keyword in JavaScript as we take you through your second lesson on interview preparation.

### Learning outcomes

- Create user interface elements using JSX
- Create reusable React components
- Manage the state and lifecycle of components using React Hooks

**Project**

Web Application (React)

## Sprint 11

You'll continue with React in this sprint as we cover some more complex topics, such as React Router, how to work with data in React, and advanced React techniques. The interview preparation lesson for this sprint introduces you to whiteboard coding.

### Learning outcomes

- Add and access routes to a React application using React Router
- Pass data between React components using context
- Use controlled components to control form elements

**Project**

Web Application (React)

# Module 5: Back-End Basics for Software Engineers

8 weeks

There’s a limit to how much you can do using only the front end of a website. A modern website almost always uses a back end to store and retrieve user data and add additional functionality. The following 4 sprints teach you all about back-end development and will also introduce you to engineering concepts in the context of interviews.

* One-week break recommended after Sprint 15

## Tools & Technologies

Node.js Express.js MongoDB Postman Git

GitHub Webpack Jest Google Cloud ESLint

Nginx pm2 Certbot

## Sprint 12

To become a full-stack software engineer, you'll need to learn server-side development. In this sprint, you'll learn to use the Node.js runtime environment along with Express.js to build a back end for your webpages. In previous sprints, you worked with other people's APIs, but in this sprint, you'll create your own! We'll then round off this action-packed sprint by diving into databases and learning all about error handling.

### Learning outcomes

- Build a server and use it to serve data and responses to client requests
- Store, manipulate, and retrieve data on a server using MongoDB and Mongoose
- Determine different error types by name and class of the errors

**Project**

Web Application (Back End)

## Sprint 13

User data can be incredibly sensitive, so it's important to protect it. This sprint will show you how, with two chapters dedicated to authentication and authorization. Regular expressions are a handy tool for working with text, so we dedicate a whole chapter to them here. You'll also continue preparing for interviews, this time with a chapter on algorithmic complexity analysis.

### Learning outcomes

- Understand and implement back-end authentication and authorization
- Use regular expressions to validate and extract string data
- Have an understanding of complexity algorithms

**Project**

Web Application (Back End)

## Sprint 14

Building on your previous knowledge of authentication, this sprint shows you how to implement it on the front end using React. You'll cover other aspects of web application security too, before moving on to take a closer look at functions — this time with an emphasis on interviews.

### Learning outcomes

- Implement front-end authorization and authentication
- Protect routes in a React application using protected routes and user tokens
- Describe approaches to preventing web application security vulnerabilities

### Project

Web Application (Back-End and Front-End Authorization)

## Sprint 15

In this sprint, you'll learn about automated testing. And since you're almost ready for your final project, you'll also learn how to prepare and deploy your back-end code.

### Learning outcomes

- Write basic automated tests with Jest
- Deploy a web application to Google Cloud
- Have an understanding of complexity algorithms

### Project

Web Application (Back-End and Front-End Deployment)

# Final Project

Up to 4 weeks

This is where you put everything you've learned throughout the program into practice and prove that you have what it takes to be a software engineer. You won't have to complete any other assignments or work through any lessons on the interactive platform. It's just like real life: you'll have a task with a deadline and you'll use your skills, knowledge, and a search engine to get the job done.

**Tools & Technologies**

can vary depending on the project the student chooses to do

# Employment Preparation

At TripleTen, we know that learning the technical skills you need for a job is only one piece of the employment puzzle. That’s why we offer a range of courses to help you land your dream job. These are included as part of the course, but you are free to opt out of Career Acceleration and the Externships if you don’t need them (note: the Career Prep Course and Career Acceleration are necessary if you want to take advantage of the money-back guarantee).

10-15 hours in total, starting from Sprint 5

## Career Prep

If you want some guidance on landing your dream job, Career Prep has all the information you need. First, you'll cover some of the necessary groundwork before you can start applying for jobs. This includes creating a portfolio, building an online presence via LinkedIn, working on your job search strategy, and growing your professional network. Once that's done, you'll focus on the different stages of the job application process, perfecting your resumes and cover letters, acing interviews, and masterfully negotiating offers. Career Prep is included with the main program and is unlocked beginning from Sprint 5.

After graduation. Typically 3-4 months

## Career Acceleration*

Prepare for real-world interviews and gain experience through authentic practice. This program is designed to help you find a job and also provides extra work with technical skills. It can last anywhere from 3-6 months after graduation, but typically lasts 3-4 months.
You will attend mock interviews, get your career documents reviewed, and receive 1:1 career coaching that will take your job search to the next level.

** Only available to students eligible to work in the US.*

4 to 5 weeks, 30+ hours

## Externships

Gain confidence in solving work tasks by completing a project for a real company. Learn to communicate with clients, meet their expectations, exchange peer reviews with colleagues, and present results to a company. Externship projects become available for participants after Sprint 10. They are also available after graduation.

tripleten

Visit our website

# Learn  the job.

# ✨ Get the job. ✨