Department for Education

In partnership with HyperionDev

Full Stack Web Developer Bootcamp

BOOTCAMP OVERVIEW

Overview

You're here because you want to learn the skills needed to become a world-class web developer. Or perhaps you're coding already, and want to take your career to the next level. Either way, you've come to the right place. This bootcamp equips you with the skills needed to build dynamic, data-driven, and career-boosting web applications using the most cutting-edge and popular web development tools in the market today.

Learn to develop dynamic web applications using HTML, CSS, JavaScript, React, and Node.js, among many other industry-relevant web development technologies.

That sounds great, but is that all you're going to get from this web developer bootcamp? The answer is no! You'll learn far more than merely how to work with different technologies. Throughout the bootcamp, you learn how various technologies and devices interact to make full-stack web applications work like clockwork. You're taught professional approaches to product design, along with the best practice guidelines needed for implementing software development projects.

Expanding Your Web Development Horizons

During the bootcamp, you're introduced to different approaches to web development and more than one development framework. In addition to all of this, you're introduced to computer science algorithms, and how they're used in web development to produce world-class results.

Moreover, this bootcamp helps prepare you for interviews and job applications. As you work your way through the programme, you will complete a series of capstone projects to demonstrate your powerful development skills to future employers, or to build your own online business.

The Process

Step 1
Log onto your personalised dashboard

Step 2
Complete coding exercises online

Step 3
Your code reviewer reviews your work within 48 hours

Step 4
Perfect your coding over 16 weeks

Step 5
Graduate with a certificate of completion

Step 6
Begin your new career in tech with our support to help land your first interview

Outcomes of this Bootcamp

- Learn HTML, CSS, JavaScript, React, and Node.js.
- Develop and design dynamic web applications using various technology stacks.
- Apply product design principles to web development projects.
- Understand crucial software algorithms and their application to web development.
- Create data-driven web applications.
- Become job-ready with our career support team that guides and prepares you for the tech career you're aiming for.

Code reviewers Powered by HyperionDev

Bootcamp code reviewers are expertly trained to integrate code review into the lives and bootcamp curriculum of students. The on-demand code review method helps students to become fluent in the language of their choice.

Our 1-on-1 code-review-centric approach works

Code review enables you to learn to code and work with web development tools the right way, which is a prerequisite for a career in web development. We help you master the deeper aspects of industry-level coding skills to set the foundation for a lucrative career in web development.

Here's why learning through code review is smarter

DON'T MAKE THE SAME MISTAKES AS COMPUTERS

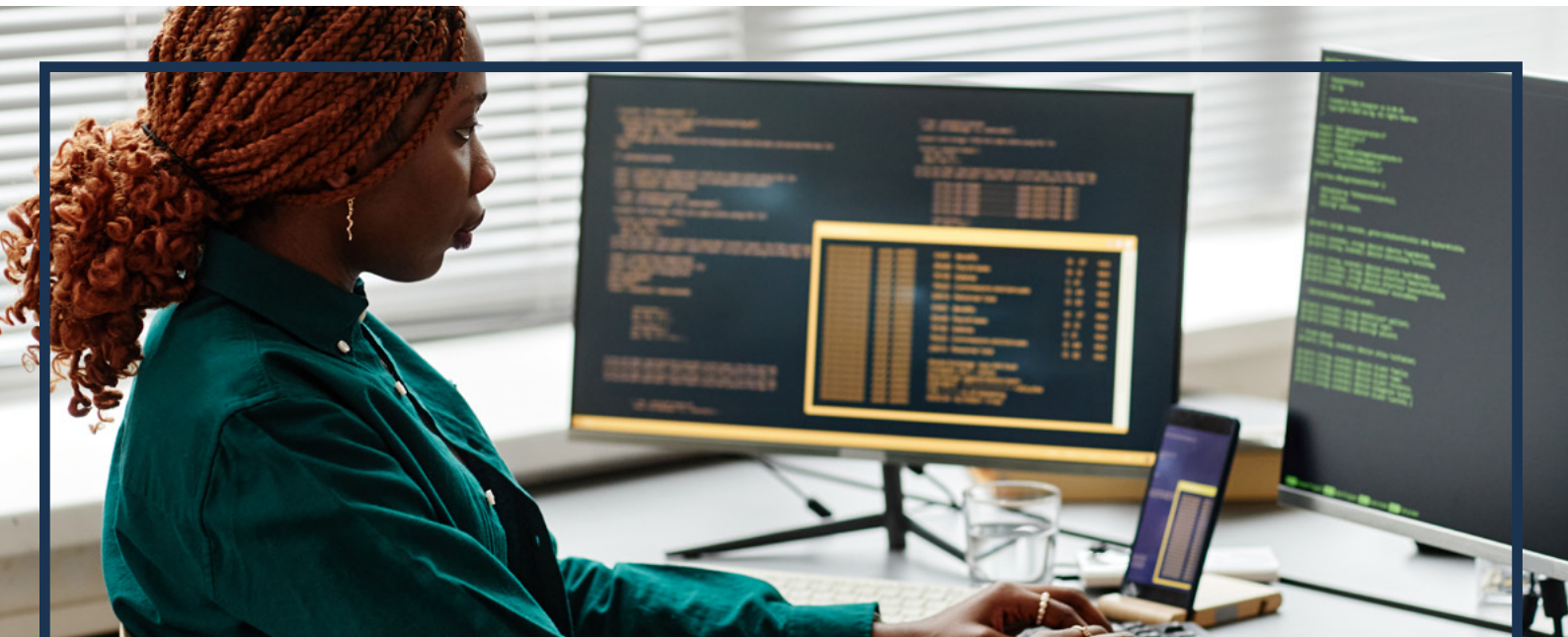
- Automated code checking is like spell check for computer programs. You can't write a world-class essay with just good spelling — you need the right tone, facts, grammar, and style. Only human-led code review can help you learn aspects of coding that are analogous to tone and style that will make you truly fluent as a developer — automated graders just can't help you learn this!

GET UNSTUCK WITH ON-DEMAND TECHNICAL HELP

- Our code reviewers will ensure you move at a steady pace by helping you debug your programs within 48 hours. This will help you to keep moving forward so that you never drop out.

BE EXPOSED TO THE INDUSTRY STANDARDS FROM DAY ONE

- Developers in the real world have their work assessed by a senior developer through the technique of code review. We're the only bootcamp in the world that exposes our students to this technique from day one so that you get an advantage in the job market.



We layer a proven, personalised approach to our code review

INDUSTRY EXPERTS TAILORED TO YOUR GOALS

- You'll work with experienced code reviewers who will guide you through 1-on-1 calls, career coaching, live chat, and email support.

JOIN A COMMUNITY OF CAREER CHANGERS

- Learn as part of a cohort of students all working towards ultimate career fulfilment. Join online group tutorials, community chats and meetups, and peer coaching.

FREE OF FEAR OF FAILURE

- Human-led code review builds trust with your educators and lets you progress at your own pace. Establish a safe space to discuss any roadblocks without fear of failure.

Why web development?

The World Wide Web has become ubiquitous. More than three billion people access it daily, with the list growing. Web developers remain highly sought-after, and their importance to the IT world shows no signs of waning. Aside from the social construct of the web, companies of all shapes and sizes are dependent on the internet. They rely on web developers to help them carve out a piece of digital real estate that keeps them relevant in today's digitised economy.

With the ever-increasing proliferation of online applications, the demand for skilled front-end and back-end web developers has grown greater than the number of people who possess those skills. This makes web development a competitive and lucrative career economy to break into. Even if you don't wish to become a web developer, familiarising yourself with the basic skills and languages used in web development is a valuable tool in any software developer's skillset.

According to Indeed, UK web developers can expect to make an average annual salary of £34,203, ranging from £18,000 to £39,000. Web developers can look forward to high-growth potential in this industry, with the U.S. Bureau of Labor Statistics estimating around 130,000 web developer jobs being added to the economy, with a 15% growth over the course of the next ten years.

How we get you hired

We're with you every step of your journey, and our support doesn't end when you graduate. Our career services are developed to help you stand out from the crowd, and grab the attention of top employers.

TECHNICAL CV AND PORTFOLIO

Receive technical assistance in getting your CV industry-ready according to accepted best-practice format.

BOOTCAMP CERTIFICATE

Walk away with a newly minted certificate as evidence of your skills and expertise in data science.

INTERVIEW PREPARATION

Know what to expect when getting ready for that big interview with expert interview preparation from professionals who have been where you are.

GUARANTEED INTERVIEW

We work with select hiring partners and will aim to help you land your first tech job interview after the completion of your bootcamp.

Career paths

JAVASCRIPT DEVELOPER:

JavaScript Developers are responsible for a website's programming, development, and implementation, and may find themselves juggling a variety of programming duties that go into the creation of websites. They may be responsible for a whole site or just specific aspects or pages of one or more websites.
Responsibilities include:

- Building sustainable coding that may be used in the future
- Ensuring the feasibility of UI/UX designs
- Enhancing performance of the main front-end website
- Modifying designs and specifications of complex applications
- Analysing code, requirements, system risks, and software reliability
- Collaborating with front-end and back-end web developers

According to Glassdoor.com, the average United Kingdom salary for a JavaScript Developer is £51,457/yr, with highs and lows ranging from £31,000 to £87,000/yr.

UX DESIGNER

UX Designers improve the accessibility and effectiveness of software and hardware from a user's perspective. They collaborate with developers, programmers, engineers, and project managers to determine product goals. UX designers build wireframes, test prototypes, and conduct focus groups. They make modifications to products as necessary. **Responsibilities include:**

- Planning and conducting user research and competitor analysis
- Interpreting data and qualitative feedback
- Creating user stories, personas, and storyboards
- Determining information architecture and creating sitemaps
- Creating prototypes and wireframes
- Conducting usability testing

According to Glassdoor.com, the average United Kingdom salary for a UX Designer is £48,755/yr, with highs and lows ranging from £38,000 to £63,000/yr.

FULL STACK DEVELOPER

Full Stack Developers are computer programmers who are proficient in both front-and back-end coding. Their primary responsibilities include designing user interactions on websites, developing servers, and databases for website functionality, and coding for mobile platforms. **Responsibilities include:**

- Developing front-end website architecture
- Designing user interactions on web pages
- Developing back-end website applications
- Creating servers and databases for functionality
- Ensuring cross-platform optimisation for mobile phones
- Ensuring responsiveness of applications

According to Glassdoor.com, the average United Kingdom salary for a Full Stack Developer is £47,520/yr, with highs and lows ranging from £30,000 to £75,000/yr.

COMPUTER PROGRAMMER

A Computer Programmer, or Systems Programmer, writes code to help software applications operate more efficiently. Their duties include designing and updating software solutions, writing and updating source-code, and managing various operating systems.

Responsibilities include:

- Reviewing operating systems and software frequently and making any adjustments necessary to keep them running well
- Writing code and implementing computer programs on multiple systems in the business
- Building and using computer-assisted software engineering tools to automate some coding
- Performing all requirements needed for the implementation of automated computer systems from start to finish
- Using code libraries to simplify the writing of code
- Collaborating with software developers in the creation of programs for their organisation

According to Glassdoor.com, the average United Kingdom salary for a Computer Programmer is £37,321/yr, with highs and lows ranging from £21,000 to £67,000/yr.

WEB DESIGNER

As a Web Designer, you'll plan, create and code web pages, using both technical and non-technical skills to produce websites that fit your customers' requirements. Being involved in the technical and graphical aspects of pages, you'll determine not only the look of the website but how it works as well. You may also be responsible for the maintenance of an existing site. **Responsibilities include:**

- Drawing up detailed website specifications
- Designing sample page layouts including text size and colours
- Designing graphics and animations, and manipulating digital photographs
- Registering web domain names and organising the hosting of the website
- Editing content, debugging code and re-designing web pages
- Working with other web specialists, including web developers and graphic designers
- Liaising with outside agencies
- Coding using a variety of languages
- Search engine optimisation (SEO)

According to Glassdoor.com, the average United Kingdom salary for a Web Designer is £27,614/yr, with highs and lows ranging from £17,000 to £44,000/yr.

TECHNICAL AUTHOR

As a Technical Author, you will be responsible for writing specialist information about products and services, and how they work. You will need to explain how things are used in a way that is easy to understand. The information may be presented in the form of user guides for software applications, reference and instruction manuals for appliances, training guides, instructional videos or online help incorporated into software and operating guides.
Responsibilities include:

- Collaborating with developers and managers to clarify any technical issues
- Using the product or service in question to understand the technology and applications for which documentation is being prepared
- Gathering and analysing the information needs of the user
- Organising information according to your user's needs

According to Glassdoor.com, the average United Kingdom salary for a Technical Author is £36,240/yr, with highs and lows ranging from £24,000 to £54,000/yr.

Breakdown of the syllabus

Our online coding bootcamp helps you progress from learning the basics of web development to becoming a full-stack web developer, with an exciting career. Advance from a beginner to a coding expert, and get started on the career path you want.

BOOTCAMP PREP

- Here is where you get to learn about the software development industry and how Hyperion supports you in achieving your development goals. Explore working with HTML and CSS, and get a taste of what web development involves, making you more prepared to commit to the Full-Stack Web Development Bootcamp.

WEB DEVELOPMENT ESSENTIALS

- An introduction to web programming using HTML, CSS, and JavaScript. Learn how to build websites that your user can interact with, such as online blogs and events applications. Develop an application that uses asynchronous JavaScript in your last capstone project.

FRONT-END DEVELOPMENT

- Master using React, a JavaScript library for building single-page front-end applications. Understand how to set up a repository for a new or existing project, and start using version control and common Git commands.

FULL STACK DEVELOPMENT

- Compile the web applications that you have developed in this bootcamp into an online portfolio that showcases your newly minted skills.

Structure of the Bootcamp

The bootcamp is structured to allow you to start coding as soon as possible.
Tasks are designed to:

- Teach you the theory needed to develop a skill.
- Enable you to practise implementing your knowledge by completing practical tasks.

Remember, with HyperionDev, you're never alone. Contact a code reviewer for support whenever you need help with a task. The code that you submit for each task is reviewed by an expert, ready to help improve the efficiency and quality of your code.

Web Development Syllabus

Tasks: 52 **Capstone projects: 7** **Build your Brand Tasks: 5**

1	**Thinking Like a Programmer - Pseudo code**	Learn how pseudo code can help you clarify your thoughts and properly plan your programs before writing any code.
2	**Your First Computer Program**	Get acquainted with JavaScript, an essential language for any web developer.
3	**Variables and Datatypes**	Learn how to store and interact with the data using variables.
4	**Beginner Control Structures - if, else, and else-if Statements**	Learn how to use conditional statements to make decisions in your program.
5	**Logical Programming - Operators and Switch Statements**	Learn how to tell the interpreter how to perform specific mathematical, relational, or logical operations using operators.

6	**Capstone Project I - Variables and Control Structures**	Put your knowledge of variables and control structures to the test.
7	**Build your Brand I**	Identify your top companies to work for and take your first steps towards securing an interview.
8	**Beginner Control Structures - While Loops**	Learn how to execute a block of code repeatedly using while loops.
9	**Beginner Control Structures - For Loop**	Learn how to use the for loop to execute a block of code for an explicit number of iterations.
10	**Towards Defensive Programming I - Error Handling**	Discover the different types of errors that might occur in your programs and how to handle them.
11	**Beginner Data Structures - Arrays and Maps**	Discover the most frequently used and versatile collection data types used in JavaScript: arrays and maps.
12	**JavaScript Functions - Built in, and Defining your Own Functions**	Learn how to use JavaScript's built-in functions as well as your own defined functions to provide better modularity for your programs and encourage code reuse.
13	**String Handling**	Learn how to process strings, an essential part of any developer's toolkit, with the use of built-in string methods.
14	**Capstone Project II - Arrays, Functions, and String Handling**	Use all the knowledge you have gained throughout this course to create a cipher.
15	**Build your Brand II**	Create a professional cover letter and CV.

16	**Data Structures - 2D Arrays**	Discover multidimensional arrays and how to traverse them in JavaScript.
17	**Applied Recursion**	Explore the concepts of recursive programming and how to “think recursively”.
18	**Towards Defensive Programming II**	Learn how to guard against errors you don’t expect.
19	**Hypothesis-driven Debugging with the Stack Trace**	Learn a methodical debugging process that reduces the reliance on changing code randomly to fix bugs.
20	**Introduction to OOP I: Objects and "this"**	Learn fundamental concepts about objects, OOP, and how the "this" keyword is used to limit variable scope to the current object.
21	**Introduction to OOP II - Classes**	Introduction to the principles of Object Oriented Programming.
22	**Introduction to OOP III - Inheritance**	Learn how you can improve the modularity and reuse of code using inheritance, and the critical role it plays in JavaScript's object system.
23	**Capstone Project III - OOP**	Apply the fundamentals of object orientation to solve a simple problem.
24	**Build your Brand III**	Create or update your LinkedIn profile to connect with a network of professionals and let people know about your skills.
25	**HTML & Semantic HTML**	Learn about semantic HTML and how it makes the meaning of an element clear to both the browser and the developer. This is critical for web accessibility as well.

26	**CSS I - Introduction to CSS**	Learn the fundamental concepts of basic CSS such as selectors.
27	**CSS II - The Box Model**	Learn about how the box model is used in CSS styling.
28	**Responsive Design**	Create websites that change structure according to the display size of the device viewing them, for an excellent device-independent user experience.
29	**Capstone Project IV - Create a Web Page using HTML and CSS**	Create an attractive web page using HTML and CSS.
30	**Closures and Arrow Functions**	Learn how to make use of closures in Javascript and get up to date with the latest arrow function syntax introduced in ES6.
31	**Higher-order Functions**	Learn about higher-order functions, i.e functions that take other functions as arguments or have a function as a return value, and how functions in JavaScript are first-class citizens.
32	**Programming with Callbacks**	Learn an essential part of JavaScript, the use of callback functions. These are functions passed as arguments to other functions.
33	**Promises**	Use promises to successfully apply asynchronous code in JavaScript, i.e. code that only executes when another piece of code has completed its execution.
34	**Async / Await**	Learn async-await for asynchronous code and the syntactical difference to promises.
35	**Capstone Project V - Concurrency with Asynchronous JavaScript**	Apply your new knowledge to create an application that uses asynchronous JavaScript.

36	**Build your Brand IV**	Identify and prioritise current job opportunities and submit applications to your top 5 advertised roles.
37	**DOM Manipulation**	Learn how to use DOM manipulation to dynamically change elements on your webpage.
38	**Event Handling**	Learn how to create JavaScript functions that handle events on your HTML pages.
39	**Introduction to Network Protocols and System Architecture: HTTP and Client-server**	Learn how computers communicate with each other over the internet using the HTTP protocol, and learn the commonly used client-server architecture for transferring information using HTTP.
40	**JSON**	Learn how JSON and the Web Storage API are used to facilitate communication between the client and the webserver.
41	**AJAX with Fetch**	Discover the AJAX technique for updating pages dynamically without reloading the whole page, and how we can use the built-in fetch function for retrieving data from external server APIs.
42	**Capstone Project VI - Build a complete website**	Create an interactive website using JavaScript.
43	**Git Basics**	Dive into using Git and discover how to set up a repository, use common Git commands, commit a modified file, view your project's history, and branch.
44	**Build your Brand V**	Use GitHub to start building a portfolio of work that you can share with others to showcase your skills.
45	**Node.js**	Install Node.js. Learn what Node.js is and the advantages of using it. Use existing Node.js modules (HTTP module and File modules) and create Node.js modules.

46	**Introduction to React.js**	Learn the fundamentals of ReactJS, set up your environment, and create React Components using JSX.
47	**Class Components and Props**	Discover the most important concept when using ReactJS: components.
48	**State Management and Component Lifecyle**	Apply the principles of OOP to managing the state of your class components.
49	**Function Components**	Learn to work in the simplest and most modern way to create React components.
50	**React Hooks**	Apply the principles of functional programming to managing the state of your function components.
51	**Redux and Global State Management**	Learn how to manage the state of larger applications by using a global state management system.
52	**Capstone Project VII - Create a React app**	Consolidate all that you have learned to showcase all the skills you have learned.