



# Full Stack Web Developer Online Bootcamp

**A 9 Months Full Stack Development  
Bootcamp**

Transform into a highly sought-after  
full-stack Software Developer in just  
one year through our innovative  
program

**Visit :** [www.mycareerxpert.com](http://www.mycareerxpert.com)

# Overview

Over the course of this blended bootcamp, you'll learn the principles of full stack web development and software engineering in detailed online modules, with frequent on-site check-ins to ensure that you're progressing consistently toward final certification.

**Learn web development and software engineering in a blended course that teaches:**

HTML

CSS

JavaScript

Git

PHP

SCSS

Node

SQL

React

Express

MongoDB

Java



## 1-on-1 mentorship

As with all HyperionDev bootcamps, you'll have access to 1-on-1 mentorship



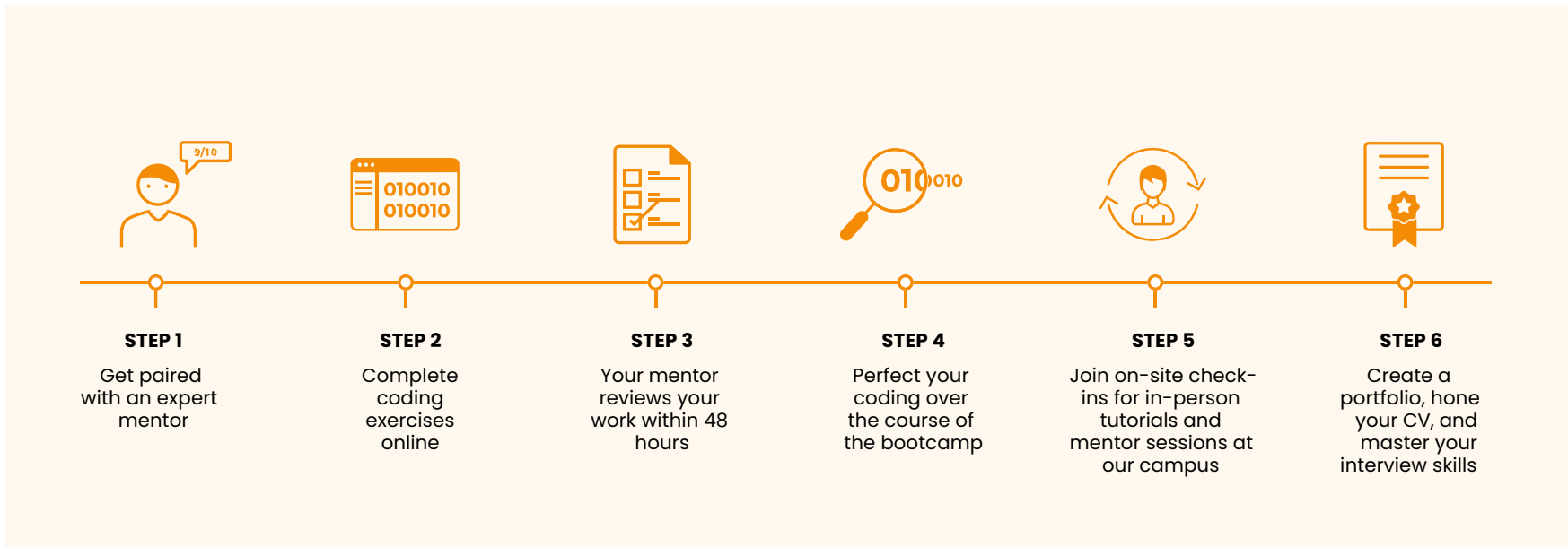
## Video chats & Prompt review

Regular video chats, and prompt review and feedback on all your coding submission

# Continuous Professional Development

Once you complete your bootcamp, you'll receive ongoing career development from our career support and placements team, and free access to our HyperionDev Careers online community for developers and recruiters, so that you can have the technical CV, interview skills, and industry connections you need to boost your career.

## The Process



# Outcomes of the Bootcamp

- ✓ Once you've completed the Certified Full Stack Web & Software Engineer Bootcamp, you will be able to:
- ✓ Understand essential programming concepts taught in JavaScript.
- ✓ Use basic web development concepts in HTML, CSS and SCSS.
- ✓ Make use of advanced JavaScript concepts like jQuery, JSON, and DOM manipulation.
- ✓ Employ in-depth version control with Git, GitHub, and Docker. Understand a wide range of software engineering concepts taught in Java, including Object- Oriented Programming, recursion, and algorithms.
- ✓ Use PHP to manage server-side functions, and employ SQL Server to access relational databases and create plugins for WordPress sites.
- ✓ Understand and use various advanced concepts in full stack web development using the MERN stack (MongoDB, Express, React and Node). Build your brand with a rich portfolio, detailed technical CV, and fully developed professional LinkedIn



# Our Mentored 1-on-1 Code Review Centric Approach Works

Code review enables you to learn to code the right way through mastery of deeper aspects of software development that are a prerequisite for a career in coding. We help you master the deeper aspects of industry-level development and set the foundation for a lucrative career in coding.

## Here's why learning through code review is smarter

- ✓ Don't make the same mistakes as computers
- ✓ Get unstuck with on-demand technical help
- ✓ Be exposed to the industry standards from day one

## We layer a proven 1-on-1 personalised mentorship approach over our code review

- ✓ Industry experts tailored to your goals
- ✓ Join a community of career-changers
- ✓ Receive grading and certification through on-site check-ins



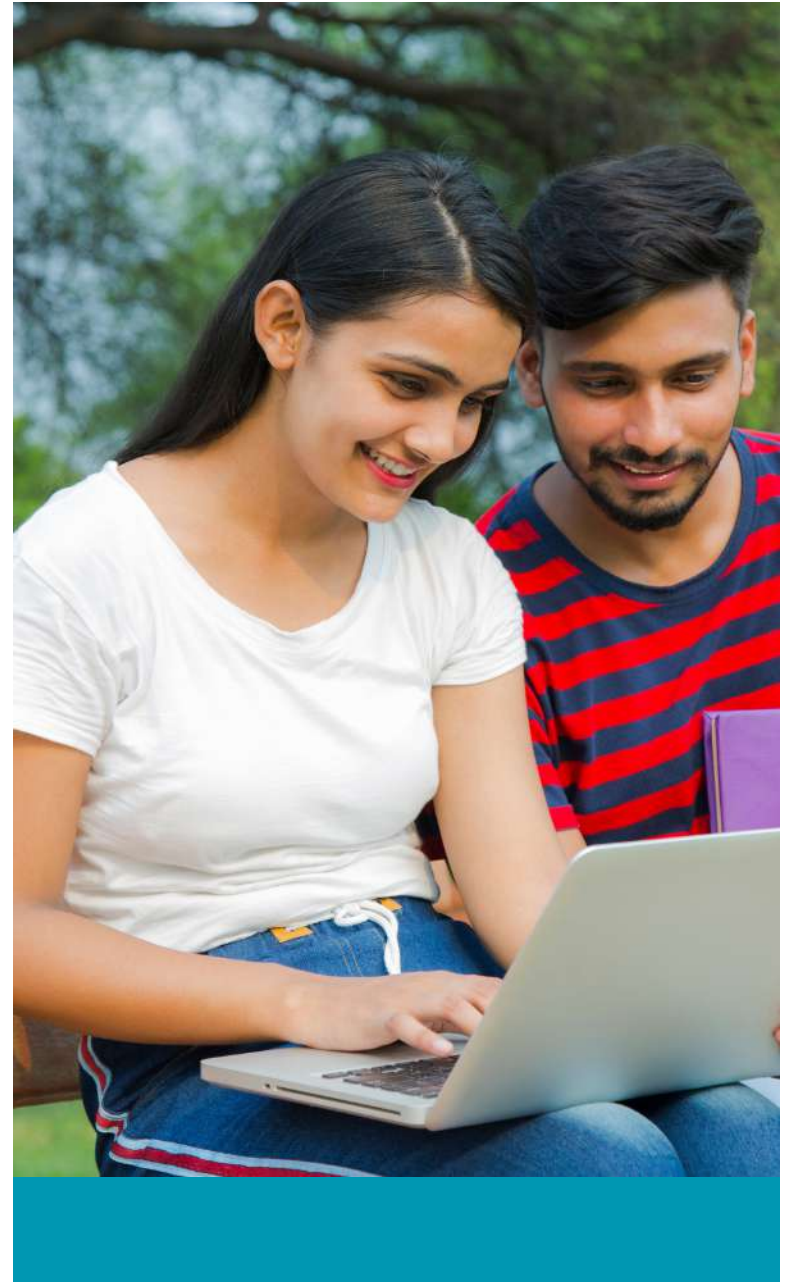


# Why you should choose a career in tech

There is great advantage in knowing how to code, as it brings new career opportunities, skills development, and personal growth. The last decade has seen massive growth in the use of digital technologies, and today, tech firms offer some of the highest-paying jobs in the world.

Knowing how to code increases your competitive advantage in the working world, even if you don't make the move to a new tech career. Companies are always looking for prospective hires who know how to solve problems and innovate, as well as use technology in addition to their core skill set – skills that can easily be highlighted with knowing how to code.

Finally, coding skills bring many personal rewards. Programming requires a mindset of thoughtfulness and innovation in assessing and solving problems, and as you develop these abilities, you build a skill that can be used in almost any area of your life.



# 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 software engineering.



## 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.



## SAQA Accredited

This course is credit-bearing, which will count towards any future studying you may wish to do.





# Career Paths



## Full stack web developer

Full stack developers are web developers who are experts in the wide range of technologies that a website needs to function.



## Back end web developer

Backend developers take care and maintain the back-end of a website, including databases, servers, and apps, and they control what you don't see.



## Software engineer

A software engineer is a person who applies the engineering design process to design, develop, test, maintain, and evaluate computer software.



# Full stack web developers

Full stack developers are web developers who are experts in the wide range of technologies that a website needs to function. These professionals deal with all parts of the web development cycle, from programming everything you see and interact with on the front end of a website, to creating the systems and databases on the back end that allow the website to complete complex tasks.

## The responsibilities of a full stack developer include:

- ✓ Conceptualising, planning and modelling the requirements for building websites
- ✓ Creating the front end of a website, including user interactions and the site's visual layout Uploading or updating the copy, images, and other content of the website
- ✓ Coding the back end of a website using an appropriate stack of technologies
- ✓ Creating the servers and databases that will be used by the back end's functions
- ✓ Designing and developing Application Programming Interfaces to be used on the website Optimising the speed and functionality of all web content across a range of operating systems, browsers, and platforms, as well as on mobile devices

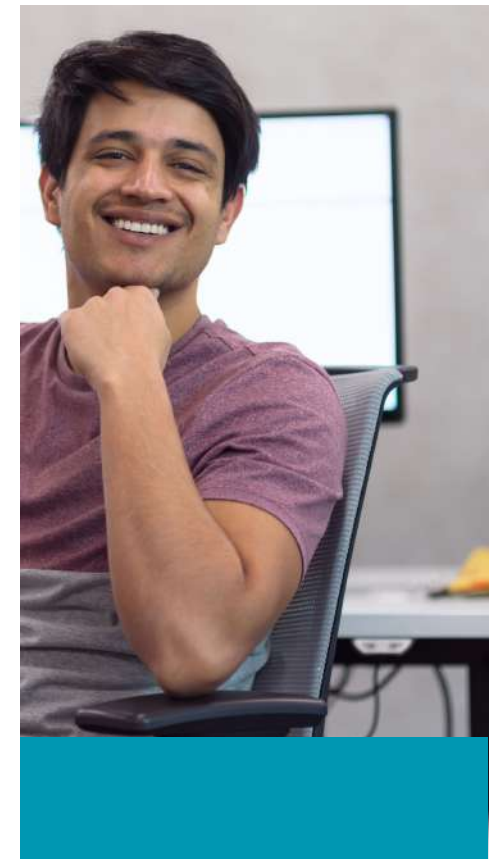


# Back end web developer

A website is always much deeper than what you see: beneath the buttons and banners lies a deep system of complex machinery that allows the website to do all kinds of things, from authenticating your activity, and ensuring your session is secure, to creating databases, interacting with other websites, and allowing deeper functionality. A back end web developer is tasked with writing out the code that makes these technologies and functions work.

## The responsibilities of a back end developer include:

- ✓ Designing and implementing methods of data storage, as well as managing these databases and storage methods
- ✓ Overseeing all server-side web application logic, and ensuring that the server is stable and continues to operate with zero downtime
- ✓ Coding the back end to ensure that front-end packages work as intended  
Documenting changes to code on the back end of a website, and recording detailed documentation of all changes
- ✓ Creating and implementing security systems to ensure the website is safe and stable
- ✓ Debugging code and resolving issues with back-end functionality  
Rewriting and improving code to improve the speed and efficiency of systems

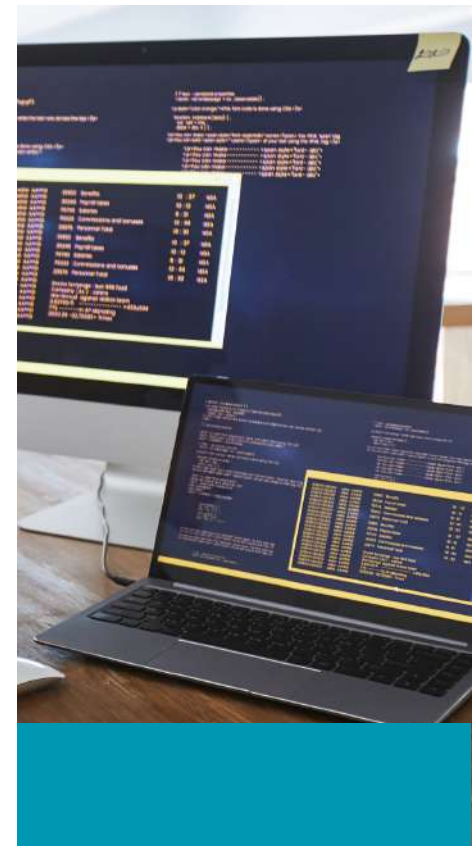


# Software engineer

A Large software systems require heavy high-level planning to ensure that code is written in a structured manner. Creating highly structured, meticulously documented code allows for the code to be easily understood and altered (if needed) – and software engineers are responsible for creating these highly ordered systems. Just like with civil engineers who plan, oversee and implement the construction of buildings, software engineers conceptualise, direct and create the software we all use on a daily basis.

## The responsibilities of a software engineer include:

- ✓ Conceptualising innovative ways to solve problems, based on a deep understanding of the problem and detailed knowledge of existing tools and methods
- ✓ Creating systems and software using up-to-date programming conventions and modern technology stacks Testing, validating and documenting all software and systems
- ✓ Directing the overall development and programming of software or systems
- ✓ Tracking and analysis of data, and using this information for systems optimisation
- ✓ Installing or implementing new frameworks, technology stacks, or software to improve performance



# Structure of the **bootcamp**

This immersive bootcamp begins with the foundational essentials of programming and web development. Master the basics of creating web pages and web programs with a variety of coding languages, frameworks and tech stacks. Then progress to more advanced concepts in programming and create highly advanced websites and software applications.

## **Bootcamp preparation and application**

- ✓ You'll start to learn the most essential concepts of programming that are required to begin your coding journey. You'll complete a short application that assesses your logical thinking and problem-solving skills, and submit a personal essay motivating your application for our immersive bootcamp program.



## **Level 1: Programming and Web Development Basics**

Work through the fundamentals of programming and web development, as you learn how to create basic programs and web pages using HTML, CSS, SCSS, and JavaScript.



## **Level 2: Advanced Programming Concepts**

Improve and refactor your code and extend the capability and efficiency of your programs as you learn more advanced programming techniques, including algorithms, hashing, and database creation and management.



## **Level 3: Advanced Web Development with the MERN Stack**

Build a powerful set of skills for both front end and back end web development, and learn how to incorporate the MERN stack into your web applications for increased functionality.

## **Post-bootcamp career support**

After you've graduated, you'll continue to receive career support and guidance, including interview preparation, CV review, and direct referral to our hiring partners.

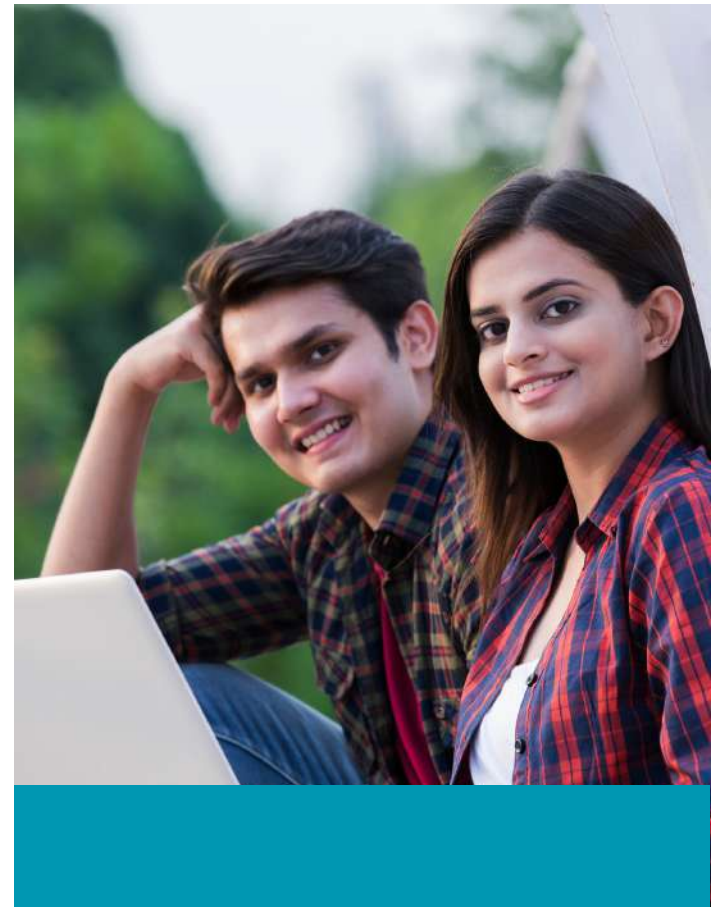


# Course Curriculum

This immersive bootcamp is structured to teach you the core tenets of front end and back end web development, and the essentials of software engineering, before progressing to more advanced concepts in both fields. Over the course of twelve months, you'll develop a deep and broad understanding of both web development and software engineering.

## These tasks are designed to:

- ✓ Introduce you to the fundamental concepts of programming and web development
- ✓ Expand your capabilities by teaching you how to use additional frameworks, technology stacks and programming languages
- ✓ Show you advanced functionalities and techniques in both software engineering and web development, so you're confident in handling a wide range of challenges
- ✓ Practise your skills by completing numerous capstone projects spread across all three levels of the bootcamp



# Our Instructors



**Pradeep Rawat**

Expert in **DevOps**

Currently working at  
Cisco



**Pranav Jaipurkar**

Expert in **Data Science**

Studied from  
University of Maryland



**Pankaj Kapoor**

Expert in **Full Stack Dev**

Studied from GLA  
University



**Gulshan Kumar**

Expert in **Data Science**

Studied from IIT  
Kharagpur



# Level 1: Programming and Web Development Basics

Tasks: 25 Capstone Projects: 4

	Task Name	Description
1	<b>Thinking like a programmer: Pseudocode</b>	Learn how pseudocode can help you clarify your thoughts and properly plan your programs before writing any code.
2	<b>Your first computer program</b>	Get acquainted with JavaScript and write your first program.
3	<b>Variables: String, numerical and boolean data types</b>	Learn how to store and interact with the data in your programs using variables.
4	<b>Beginner control structures: If, else and else-if statements</b>	Learn how to use conditional statements to make decisions in your program.
5	<b>Logical programming: Operators</b>	Learn how to tell the compiler how to perform specific mathematical, relational or logical operations using operators.
6	<b>Capstone Project I: Variables and control structures</b>	Put your knowledge of variables and control structures to the test.

	<b>Task Name</b>	<b>Description</b>
7	<b>Beginner control structures: While and for loops</b>	Learn how to execute a block of code repeatedly using loops.
8	<b>Towards defensive programming I: Error handling</b>	Discover the different types of errors that might occur in your programs and how to handle them.
9	<b>Beginner data structures: Arrays and maps</b>	Discover the most frequently used and versatile collection data types used in Javascript: arrays and maps.
10	<b>Beginner programming with functions: Using built-in functions and defining your own functions</b>	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.
11	<b>Capstone Project II: Arrays, functions and string handling</b>	Use all the knowledge you have gained throughout this course to create a cipher
12	<b>HTML</b>	Learn to use HTML to add content to a webpage.
13	<b>CSS</b>	Use CSS to improve the appearance of your webpage.
14	<b>SCSS</b>	Use SCSS, a more advanced version of CSS, to customise and improve your webpage.



	<b>Task Name</b>	<b>Description</b>
15	<b>Capstone Project III: Create an online store using HTML and SCSS</b>	Create an attractive website using HTML and SCSS.
16	<b>Programming in JavaScript I: DOM manipulation</b>	Programming in JavaScript III: JSON
17	<b>Programming in JavaScript II: Event handling</b>	Learn how to create JavaScript functions that handle events on your HTML pages.
18	<b>Programming in JavaScript III:JSON</b>	Learn how JSON and the Web Storage API are used to facilitate communication between the client and the webserver.
19	<b>Programming in JavaScript IV: jQuery</b>	A brief introduction to the popular "Write Less, Do More" JavaScript Library.
20	<b>Capstone Project IV: JavaScript fundamentals</b>	Add dynamic elements and functionality to the website created in the previous Capstone Project using JavaScript.
21	<b>The command line for web development</b>	Learn to use the command line for web development. Learn basic commands and functionality used with the command line.
22	<b>Version Control I: Introduction to version control and Git</b>	Explore the Git version control system and the GitHub collaboration platform.

	<b>Task Name</b>	<b>Description</b>
23	<b>Version Control II: Git basics</b>	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.
24	<b>Version Control III: Deployment of static websites</b>	Learn how to deploy your websites using GitHub Pages.
25	<b>Build your Brand I</b>	Use GitHub to start building a portfolio of work that you can share with others to showcase your skills.

## Level 2: Advanced Programming Concepts

Tasks: 21 Capstone Projects: 3

	Task Name	Description
1	<b>Introduction to Java programming I: Java basics</b>	Discover the fundamental concepts of Java, such as variables, data types and control structures.
2	<b>Introduction to Java programming II: Data structures</b>	Learn how one-dimensional and multidimensional arrays are declared, created, initialised and processed.
3	<b>Introduction to Java programming III: Methods</b>	Learn how to create and efficiently use Java methods.
4	<b>Java text I/O</b>	Learn how to read from and write to external storage mediums.
5	<b>Object-Oriented Programming</b>	Learn the fundamentals of object-oriented programming.
6	<b>Advanced OOP</b>	Learn about inheritance and how it can be utilised to make your code neater and more logical.

	<b>Task Name</b>	<b>Description</b>
7	<b>Capstone Project I: OOP</b>	Create an object-oriented program to solve a real-world problem.
8	<b>Recursion</b>	Explore the concepts of recursive programming and how to “think recursively”.
9	<b>Towards defensive programming II</b>	Learn how to guard against errors you don't expect.
10	<b>Refactoring</b>	Explore the concept of refactoring and the variety of tools provided by Eclipse which allow you to refactor your code quickly and easily.
11	<b>Capstone Project II: Putting it all together</b>	Extend and complete the programme you have been building in the previous two Capstone Projects.
12	<b>Java Collections Framework</b>	Discover the Java Collections Framework, a collection of interfaces and classes that helps to store and process data efficiently.
13	<b>Algorithms: Hashing</b>	Learn how to implement the popular computer science algorithms of hashing, shuffling and composition.
14	<b>Computer science fundamentals and Big O notation</b>	Learn what computer science is and discover how Big O notation is used to describe the performance of algorithms.



	<b>Task Name</b>	<b>Description</b>
15	<b>5 Build your Brand II</b>	Create or update your LinkedIn profile to connect with a network of professionals and let people know about your skills.
16	<b>Version Control IV: Pipelines</b>	Learn about more in-depth Git concepts like stashing, commits, pulling and pushing.
17	<b>Containers: Docker</b>	In this task, you will learn Docker: a service that is used to build and share applications regardless of the platform it is run on.
18	<b>Database Design, Implementation and Development</b>	Learn the fundamentals of SQL and create your own relational databases
19	<b>Java database programming: The JDBC</b>	Explore the JDBC: the Java API for accessing relational databases.
20	<b>Software documentation</b>	Learn about the various forms of software documentation and how they can improve the quality of your software.
21	<b>Capstone Project III: Databases</b>	Design a system that interacts with a database.
22	<b>Build your Brand III</b>	Join the HyperionDev Careers community to make yourself visible to Hyperion hiring partners.

# Level 3: Advanced Web Development with the MERN Stack

Tasks: 37 Capstone Projects: 5

	Task Name	Description
1	<b>Introduction to full stack web development</b>	Get to grips with some key concepts in full stack web development including server-side rendering (SSR), clientside rendering (CSR), single-page applications (SPA) and how a browser actually works.
2	<b>Introduction to PHP</b>	Learn about the benefits and uses of PHP as well as fundamental programming concepts like data types, conditional statements, loops and functions.
3	<b>HTML form handling in PHP</b>	Take a look at using superglobals to collect form-data.
4	<b>Form validation in PHP</b>	Explore how we go about validating forms using PHP.
5	<b>OOP in PHP</b>	Transfer your OOP knowledge to the PHP context.
6	<b>Cookies and session management</b>	Learn about how we store and manage information using cookies and session variables.

	<b>Task Name</b>	<b>Description</b>
7	<b>Error handling</b>	Practise using Exceptions and error handling in your PHP code.
8	<b>SQL Server in PHP</b>	Learn to use PHP to interact with a SQL Server database.
9	<b>Laravel</b>	Use the web development framework Laravel to build a web application.
10	<b>Capstone Project I: PHP and SQL Server</b>	Consolidate your knowledge of SQL Server, Laravel and PHP in this project.
11	<b>Introduction to WordPress</b>	Learn the basics of creating a WordPress site
12	<b>WordPress themes</b>	Learn to make a theme using your own CSS to customise an existing WordPress template.
13	<b>WordPress plugins</b>	Learn to write your own plugin to add to a WordPress site.
14	<b>Capstone Project II: PHP and WordPress</b>	Put your knowledge of PHP and WordPress in this project.
15	<b>JavaScript refresher</b>	Do a quick refresher of JavaScript to cement your understanding before moving onto more advanced concepts.

	<b>Task Name</b>	<b>Description</b>
16	<b>Advanced JavaScript</b>	Understand advanced concepts that make JavaScript different from other programming languages, including the Event Queue and Asynchronous Processing.
17	<b>ReactJS I: Introduction to ReactJS</b>	Learn the fundamentals of ReactJS, set up your environment and create React Elements using JSX.
18	<b>ReactJS II: Components</b>	Discover the most important concept when using ReactJS: components.
19	<b>ReactJS III: Events</b>	Learn how to create React applications that respond to events
20	<b>ReactJS IV: Managing states</b>	Learn about states and how to make React components more dynamic.
21	<b>Capstone Project III: Create a React app</b>	Create a game using React to showcase your mastery of the ReactJS concepts introduced so far
22	<b>Deploy a React app</b>	Deploy your app to Heroku to share your work with the world!
23	<b>ReactJS V: Fetching data</b>	Fetch data from a remote source using React.

	<b>Task Name</b>	<b>Description</b>
24	<b>ReactJS VI: Testing a React app</b>	Learn to write tests for your React application using Jest.
25	<b>Introduction to Node.js</b>	Install Node.js. Learn what Node.js is and the advantages of using it. Use existing Node.js modules and create Node.js modules.
26	<b>Express Web Framework I</b>	Introduction to Express: the fast, unopinionated, minimalist web framework for Node.js.
27	<b>Express Web Framework II</b>	Learn to use Express' routing and application-level middleware.
28	<b>Full stack with React and Express</b>	Learn how to get React to interface with your Express backend.
29	<b>Testing and refactoring Node. js</b>	Learn the importance of using test-driven development. Learn how to use popular testing frameworks.
30	<b>Capstone Project IV: ReactJS</b>	Create a full stack web application using Next.js and Express.
31	<b>Getting started with MongoDB</b>	Create a MongoDB database using Atlas: MongoDB's Database as a Service solution.

	<b>Task Name</b>	<b>Description</b>
32	<b>Mongo</b>	Learn how to use Mongo, MongoDB's administrative shell, to create databases and collections. You will also learn to create, read, update and delete documents from collections.
33	<b>Mongoose</b>	Use Mongoose, a library that sits on top of the MongoDB driver, to write your own CRUD operations.
34	<b>Authentication with JWT</b>	Learn how to introduce authentication with JSON Web Tokens into your webpages.
35	<b>Capstone Project V: Authentication</b>	Put your knowledge of authentication into practice in this Capstone task.
36	<b>Build your Brand IV</b>	Get some pointers for a successful technical interview and connect with your HyperionDev Careers team to arrange a mock interview.
37	<b>Build your Brand V</b>	Make your mark by ensuring that all components of your online presence related to your professional brand are finalised.



# Frequently Asked Questions

## What are the terms and conditions to be eligible for guaranteed job interviews?

To avail the guaranteed job interviews, you'll have to:

- Maintain attendance of at least 85% during the live classes for each phase of the program.
- Submit all your course-end projects and capstone projects within 7 days of program completion.  
Mandatorily submit all assignments, projects and case studies within the due timeline.
- Successfully complete all the course modules along with at least 80% of the selflearning videos.
- Avoid malpractices during appearing for the test or assignment submission else you will be permanently disqualified from this program.
- Candidates must clear the PRT ( Placement Readiness Test) after the course completion to get into the placement pool and get access to our job portal as well as the career mentoring program.

## What is expected from a candidate during the job assistance period?

The following is expected from the candidates during the job-assistance program:

- Should give their 100% to secure a good job.
- Attend all the career preparation sessions that are conducted.
- Remain active in job search and apply to at least 30 jobs per month.
- Once shortlisted for a job, the candidate should go through the entire selection process.
- Candidates should be open for relocation to the company's location.
- Failure to comply with any of the above will result in debarring from the placement process.

## What are the eligibility criteria for this program?

To be eligible for this program, you will need to meet the following criteria:

- Individuals who have pursued B.E., B.Tech, B.Sc., BCA, M.E., M.Tech, M.Sc., and MCA.
- Any working professional up to 28 years of age.
- Have a minimum of 50% throughout their academic journey (i.e. X, XII, Graduation, and Postgraduation)
- Have valid mark sheets and degree certificates for verification.
- Must be allowed to legally work in India.
- Have a valid Aadhar Card and PAN Card. Must pass the background check from previous employer/institute.
- **Note:** If you are in your final year of college, then you will be required to submit all mark sheets & certificates earned till the last semester (with at least 50% up to your last semester) and proof from your college that mentions the month and year of your graduation.

## What is the process of getting into the placement pool?

To be eligible for getting into the placement pool, the learner has to complete the course along with the submission of all projects and assignments. After this, he/she has to clear the PRT ( Placement Readiness Test) to get into the placement pool and get access to our job portal as well as the career mentoring program.



**Thank You**