



TASK

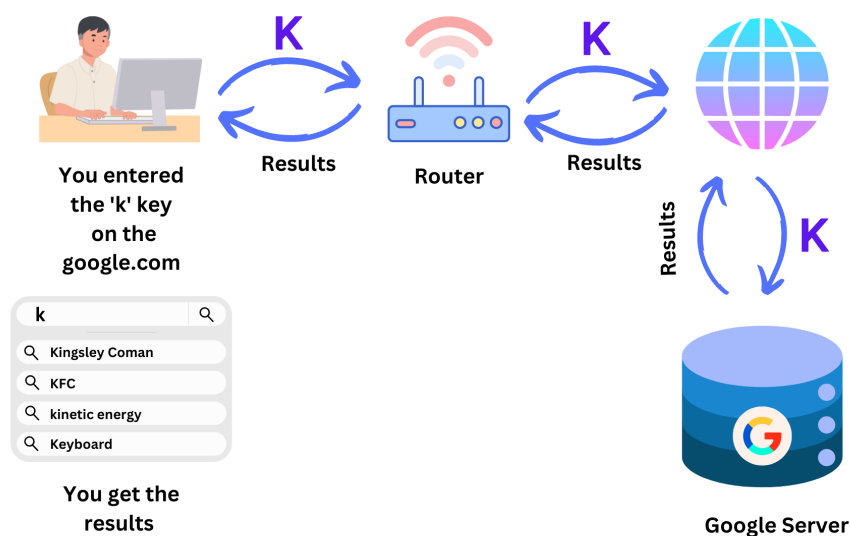
Introduction to Web Development

Visit our website

Introduction

WELCOME TO THE INTRODUCTION TO WEB DEVELOPMENT TASK!

Think about it - every single keystroke you make on your computer sets off a mesmerising chain of events. It's an intricate dance of technology that begins at your fingertips and, with the magic of web development, could end up fetching data from a server halfway around the world, as shown in the “Google Query example” in the figure below. This captivating process underlies the vast potential of web development.



Our digital world functions due to the intricate dynamics of web development. It handles data transmission, processing, and display, powering every online interaction we make. Whether we are perusing the latest headlines, scrolling through our social media feeds, or managing our finances via online banking, web development plays a pivotal role. It's the foundation of the World Wide Web (www) - the vast global information system we depend on daily.

THE WORLD-WIDE WEB

What is the World Wide Web really? Well, it is a global information system consisting of web pages linked to each other using **hyperlinks**. These links allow us to navigate from one page on a website to another. They also allow us to navigate to pages from other websites from around the world. This linking technology creates the effect of an infinite web of information that we navigate daily.

Have you ever wondered who makes all these pages and ensures they're all linked together? The answer is web developers. They're a bit like the authors and

illustrators of a giant online book, creating and drawing each page and then connecting them so we can explore.



A note from the
HyperionDev Team

Even though we can't imagine our lives without it, the World Wide Web is a rather recent invention. It was invented by Tim Berners-Lee, an English computer scientist, in 1989. Since its invention, it has expanded exponentially until it has become intricately interwoven into every part of our lives. Our work, entertainment, communication, and even our culture are strongly influenced by this powerful technology.

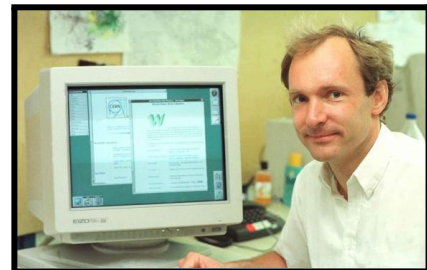


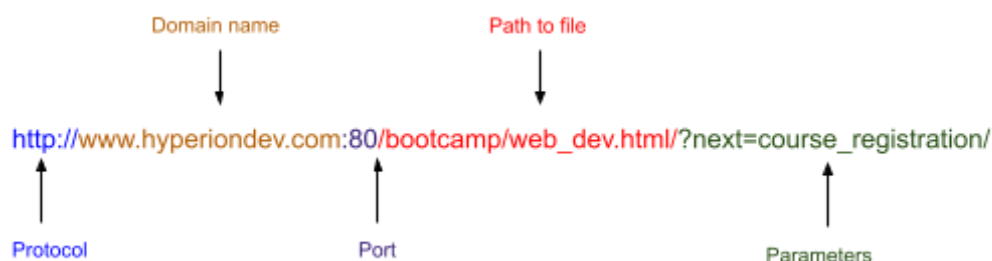
Figure 1: Tim Berners-Lee ¹

CORE COMPONENTS OF THE WORLD WIDE WEB

The World Wide Web consists of three key components.

The first is called a **Universal Resource Identifier**, or URI. This is a unique identifier assigned to each page and resource on the web. It allows us to identify a specific page, or video/audio/etc. file that we want over the Internet. Now, a subset of URIs exists, known as **Uniform Resource Locators**, or URLs. These not only identify but also locate Internet resources.

Consider the following fictional URL:



¹ "History of the Web – World Wide Web Foundation."
<http://webfoundation.org/about/vision/history-of-the-web/>. Accessed 13 July. 2023.

As you can see, the URL contains a lot of information:

1. It identifies the protocol being used to send information. In the example above, the protocol being used is HTTP.
2. It identifies the domain name of the web server on which the resource can be found, e.g. `www.hyperiondev.com`.
3. It identifies the port on the server. In this example, the port number is given as port 80. In reality, if the default HTTP ports are used (port 80 is the default for HTTP, port 443 for HTTPS), they don't have to be given in the URL.
4. It gives the path to the resource on the web server, e.g. `/bootcamp/web_dev.html`
5. Parameters can be passed using the URL. Parameters are passed as key-value pairs (`?key=value&key2=value2`), e.g. `?next=course_registration`

The second technology is the language called **HyperText Markup Language (HTML)**, which is used to create web pages. Unlike other programming languages that allow us to create programs that actually perform tasks, this language is used to create the format of the page - what the contents are and how they are placed on the page.

The third technology is the protocol used to request and transfer web pages from one location to another. The Internet is a frantically busy and complex medium of communication and, in order for us to ensure that we transfer things successfully from one location to another, we must have rules of transfer - a protocol - for devices to adhere to. This protocol is called **HyperText Transfer Protocol**, better known as **HTTP**.

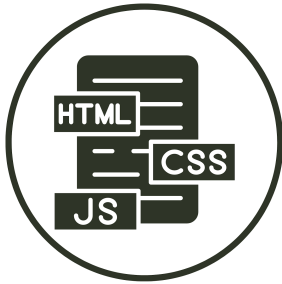
FRONT-END AND BACK-END WEB DEVELOPMENT

Within the realm of web development, two complementary engines drive our digital experiences: **front-end** and **back-end** development. Front-end development concentrates on crafting the user-facing elements and user experience, ensuring the visual aesthetics and interactivity align with user expectations. By contrast, back-end development focuses on behind-the-scenes functionality, facilitating data management, and handling server operations to deliver dynamic content to the user.

Front-End Development

Everything that you physically see, hear, and interact with on your screen when using the Internet is maintained by this component. When you type a query into

Google, for instance, and hit enter, your request is whisked away to servers far removed from your immediate vicinity. A response is soon sent back to your computer, where your web browser interprets the data and renders it on your screen.



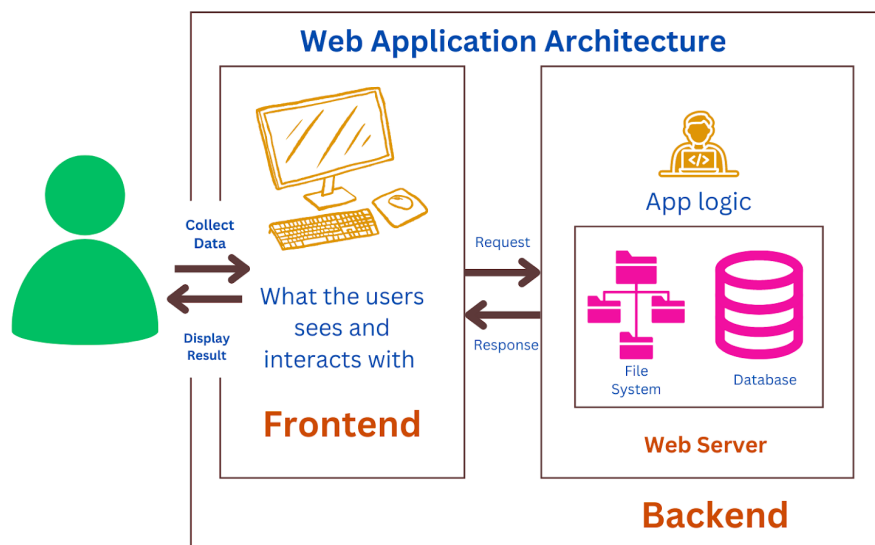
A trinity of powerful tools - [HTML](#), [CSS](#), and [JavaScript](#) - play an essential role in this process. HTML (HyperText Markup Language) provides the foundational structure of a webpage. CSS (Cascading Style Sheets) applies styling, adjusting the layout and adding colours to make the site visually appealing. Finally, JavaScript introduces interactivity, turning a static page into a dynamic experience.

Back-End Development

Back-end development refers to the processes and procedures occurring behind the scenes on the server side. These unseen elements work together to ensure the seamless operation of a website or web application.

To illustrate, let's revisit our “Google query example”. When you type a query and press enter, the front-end part is the interface you see and interact with. However, the heavy lifting, including fetching the most relevant results according to your location, predicting your typing, and handling thousands of similar requests simultaneously, occurs in the back-end.

Back-end developers primarily deal with servers, databases, and application logic. Applications are built using backend languages such as PHP, Python, Java, Ruby, etc. To manage, modify, and deliver data to the user interface, databases like MySQL, Oracle, and SQL Server are utilised.



FULL-STACK WEB DEVELOPMENT

A web developer should know how to build a website from the ground up, meaning that they should be able to create custom code to accommodate a client's unique needs and develop everything on the webpage, from the site layout to features and functions.

Web Development can be divided into three parts:

- **Client-side scripting:** This is code that executes in a web browser. It is what people see when they open a website.
- **Server-side scripting:** This is code that executes on a web server. It is everything behind the scenes that makes a website work.
- **Database technology:** This stores and manages all the data needed for a website.



A note from the
HyperionDev Team

Check out this HyperionDev blog
on a detailed comparison of
[Front-End vs Back-End Web Dev.](#)

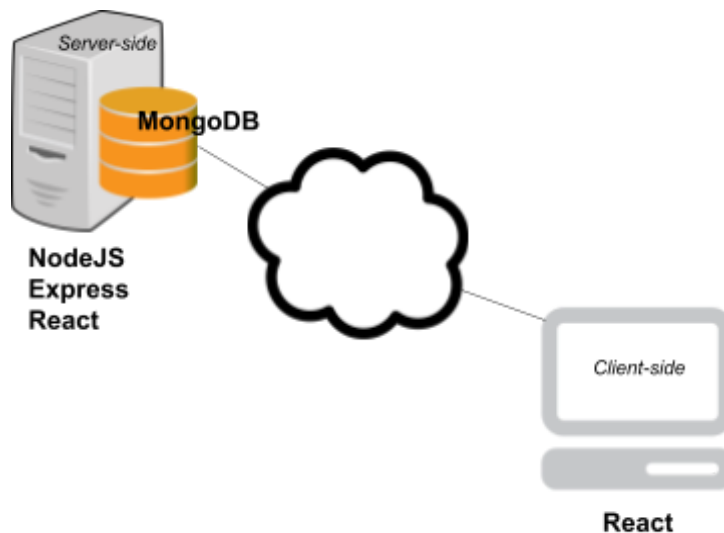
THE MERN STACK

A **stack** is a term used to describe a collection of technologies that are used together to create a web application. There are a number of web development stacks including the LAMP stack, the MEAN stack, and the MERN stack.

One of the most popular approaches to full-stack web development using JavaScript is using the MERN stack. The MERN stack includes the following technologies:

- **MongoDB:** A non-relational database that stores data as documents or objects. It provides a JSON-based data-storage approach as an alternative to traditional SQL-based databases.
- **Express:** An un-opinionated web framework that simplifies server-side web development using Node.js. It streamlines the process of creating server-side applications with Node.js.

- **React:** A JavaScript library by Facebook for creating user views (components rendered in HTML). It offers a powerful toolset for building dynamic user interfaces on both the client and server side.
- **Node.js:** A runtime environment allowing JavaScript to be used not only in browsers but also on web servers. It enables server-side execution of JavaScript, expanding its capabilities beyond the browser environment.



Extra resource

The MERN stack is becoming increasingly popular and is a powerful stack in which to work. If you're able to build and deploy good MERN applications, it will greatly help your career prospects as a developer. Read more about the benefits of using the MERN stack [here](#).

SETTING UP YOUR DEVELOPMENT ENVIRONMENT

If you did not set up your development environment during your bootcamp onboarding, please visit the **HyperionDev Web Development Environment Setup** on [GitHub](#) and follow the instructions to install Visual Studio Code and Node JS.

If you run into any trouble, submit a query via your dashboard for assistance.

Once you have set up your development environment, you are ready to start programming! You will use Visual Studio Code (VS Code) as your integrated

development environment (IDE) to open all text files (.txt), HTML files (.html), CSS files (.css) and JavaScript files (.js).

You can visit the [Javascript](#), [HTML](#), and [CSS](#) overview pages to learn how to use VS Code with these web development languages. If you've never programmed before, we strongly recommend that you watch the [introductory videos](#) and browse through the user guide to familiarise yourself with some [basic editing](#) and the [extension marketplace](#). If you encounter problems, [Stack Overflow](#) has a number of posts about VS Code.



Take note:

Opt-out telemetry in VS Code refers to the practice of automatically collecting usage data from users of the software, and sharing it with the VS Code developers to help them identify bugs, improve performance and make other changes that will benefit users.

If you have concerns about your privacy, data, and usage habits, please turn it off by using the instructions from:

https://code.visualstudio.com/docs/getstarted/telemetry#_disable-telemetry-reporting.

CONCLUSION

In conclusion, web development is a critical function that powers our online world, using complex technologies and programming languages to handle data transmission, processing, and display. The World Wide Web, an intricate system of interconnected web pages, is made possible by tools like URLs, HTML, and HTTP. Web development is divided into front-end and back-end aspects, each responsible for unique functions - from visual rendering to server-side operations.

Instructions

Read through the accompanying Additional Readings provided before doing the Compulsory Task to familiarise yourself with HyperionDev's plagiarism policy and how to reference your work.

Compulsory Task

Submit a pdf document entitled **WebFundamentals** in which you briefly answer the following questions:

1. What is the World Wide Web?
2. Explain the functional differences between a web application's front-end and back-end. See [here](#) for more information.
3. Describe what occurs on the back-end during a web interaction using the "Google query example" from the article or create your own example.
4. What is the MERN stack?



Rate us
Share your thoughts

HyperionDev strives to provide internationally-excellent course content that helps you achieve your learning outcomes.

Think that the content of this task, or this course as a whole, can be improved or think we've done a good job?

[Click here](#) to share your thoughts anonymously.

