

When we hit <https://www.techtonicgroup.com/> what happens? Don't focus too much on architecture (Monolithic, SOA, Microservices, etc.). Try to focus more on how the web functions.

On the highest level, the internet is basically a large network of computers all connected to each other exchanging information. When an end user types the url "www.techtonicgroup.com" into a web browser, a connection between two computers is established to pass the information from the server to the client. Breaking it down even further - the url(uniform resource locator) or IP address is a unique identifier that when typed into a web browser connects to the dedicated server that houses that information associated with the url. The url is composed of three different items: the protocol - https (browsing the internet and html pages), the domain name (techtonicgroup.com) The server's sole purpose is to deliver content.

From start to finish how does that data reach you to be rendered in the browser?

The client computer - or the computer making the request, asks the server - where the information is stored to display the html document. In short, it is a series of communications between the client and the server in requesting information. The server then sends the information to the client computer. The client then displays the information.

A good analogy for fetching and loading a web page is similar to the browser making a phone call. When you make a phone call, you dial the number and wait for someone to pick up and say hello. And wait for a proper "response" before you start the conversation. If there are problems connecting, then you have to redial the number. In web applications, this process all takes less than a split second.

What code is rendered in the browser?

The webpage document is written in html, and are able to be opened in web browsers. Starting with the html skeleton, followed by the html tags, then sends out get requests for additional elements - images, css, javascript files. The static files are cached by the browser so it doesn't have to be fetched again.

What is the server-side code's main function?

The server side code is involved with choosing which content is returned to the browser in response to requests. The server side code handles tasks like validating data and requests, using databases to store and retrieve data and sending the correct data to the client. Server-side code can be written in any number of programming languages, including PHP, python, ruby, C#, NodeJS, etc. Server side code has access to the server operating system. As with client side code, this can be developed using a framework - or a set of functions, rules, objects, or any number of code constructs designed to solve common problems. This speeds up development, and it's important to note that server-side code is rarely developed without a framework.

What is the client-side code's main function?

The code within the browser primarily is concerned with the appearance and behaviour of the rendered webpage. This can range from anything within UI components, creation of layouts, navigation, form validation, etc. The type of code written is usually in HTML, CSS, and JavaScript. It is run inside the browser and has little or no access to the underlying operating system. It's important to note that HTML is a markup language(annotated with keywords), it provides meaning to the text through a variety of

tags or elements.

How many instances of the client-side assets (HTML, CSS, JS, Images, etc.) are created?

For each request made, the server will return one instance. However, sessions can be created - a mechanism that allows a server to store information on the site and will tailor responses based on that information.

How many instances of the server-side code are available at any given time?

The server returns content by responding to requests. The server handles validating submitted data and requests and using databases to store and retrieve data and sending it to the client as required. There should only be one instance or version of server side code at any given time.

What is runtime?

Runtime is the instructions or software that is executed while the program is running. In short it is part of the language implementation which executes code. The time when a program is run. It starts or is executed when the program is opened, and ends when the application is closed/quit.

How many instances of the the databases connected to the server application are created?

Each application can have one connection to the database done through one or more server side scripts. SQL is used to retrieve certain sets of data. These queries can be executed within the server side code.

References:

https://developer.mozilla.org/en-US/docs/Learn/Server-side/First_steps/Introduction

<https://medium.com/@maneesha.wijasinghe1/what-happens-when-you-type-an-url-in-the-browser-and-press-enter-bb0aa2449c1a>

<https://marksheet.io/web.html>

<http://www.20thingsilearned.com/en-US/credits>

Reflection:

I enjoyed the reading, it really helped to put into context information that I would otherwise overlook on a daily basis. Both resources provided a nice higher level overview of some terms and concepts. Would like to play more with git/sourcetree. I have no experience with either of those beyond a few youtube videos, and an additional course on Codecademy.

HTML and CSS is originally how my interest in coding came about. I loved the idea of building things with code. The exercises were a good refresher for me. I've built primitive websites for friends and myself prior. The area where I could use more instruction and practice is grids. I found the JS exercises to be of more interest, because I prefer OOP methodologies and logical reasoning. I enjoyed the course, but i need to go through modules, browser compatibility and requests again.

To work on:

- Code structure/organization and file structure/organization. In retrospect, I should have developed a better file system for organizing the exercises
- Curious about how professional organizations develop projects, how do agile methods work within a team environment
- github/git - more practice with terminal commands
- Using sourcetree - in retrospect, I should have developed a better schema for my commits, how does using sourcetree work within a development team
- More practice with JS. I will be going through codecademy a second time for some modules.

I would be extremely thrilled to take part in this program. As someone without a traditional CS/STEM background, it's impossible to not feel the stereotypical imposter syndrome. Techtronic Group would help me to become an exceptional, well-rounded developer by placing me in an environment where I can learn with a fellow body of peers who can offer guidance. I am excited that more tech companies are developing better avenues to help people get tech jobs. Techtronic is doing their part for making these positions more accessible by developing an apprenticeship program. In five years, I see myself as a successful software engineer within the Boulder/Denver markets. I do have an avid interest in security and would like to explore topics related to cybersecurity development and application engineering. The apprenticeship would help me to start down this path for future success. There is no end to the new, and ever evolving world of technology, therefore I am dedicated to a lifetime of learning.