WEB DEVELOPMENT

IN THIS SECTION I WILL BE COVERING THE TOPICS OF WEB DEVELOPMENT USING MERN STACK

TOPICS TO BE COVERED

- ✓ HTML
- ✓ CSS
- ✓ BOOTSTRAP
- ✓ JAVASCRIPT
- ✓ BACKEND
- ✓ DATABASE
- ✓ FRONTEND

ROADMAP TO FULL STACK DEVELOPMENT

TOPICS OF TODAY

- BASICS OF WEB DEVELOPMENT
- FRONTEND SKILLS
- FRONTEND FRAMEWORKS
- BACKEND SKILLS
- DATA AND DATABASE
- SERVER AND DEPLOYEMENT SKILLS

Topics

Web Development

Basics: Web Development

Websites are a bunch of files stored on computers called servers. The Servers are computers that are used to host websites and store the website files. These servers are connected to the giant network called the **World Wide Web**.

There are three main components of a website:-

i. HTML

ii. CSS

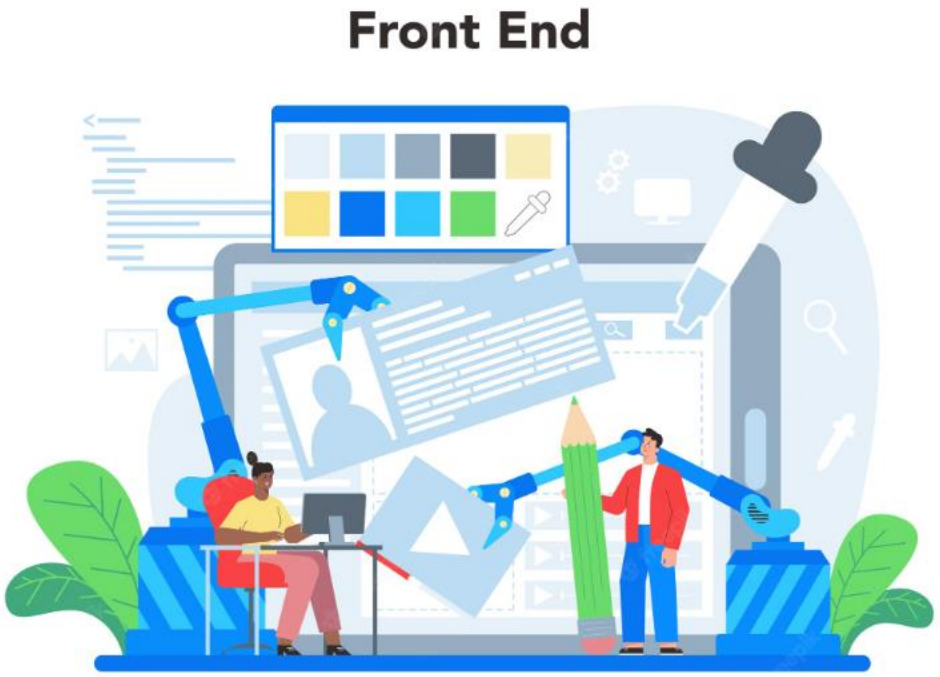
iii. JAVASCRIPT

A. **HTML(Hyper Text Markup Language)** is the foundation of all websites. It's the main file type that is loaded in your **browser** when you look at a website. This scripting language is used to **structure** the different parts of our content and define what their meaning or purpose is.

B. **CSS (Cascading Style Sheet)** is used for styling the HTML elements. It provides 1000s of styling functions which are used to **style** the **HTML elements** defined by us. It is the language that we use to style and layout our web content.

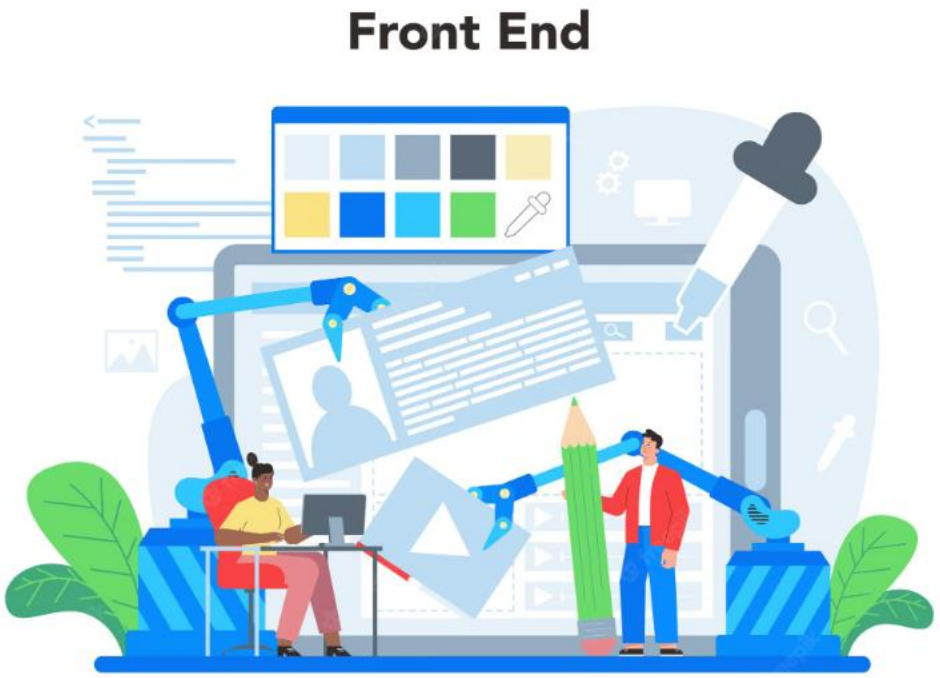
C. **JS(JavaScript)** This programming language allows you to interact with elements on the website and to manipulate them. While CSS adds style to HTML, Java Script adds interactivity and makes a **website** more **dynamic**.

Front End Skills



- It is important to make sure that web applications download fast and are responsive to user
- interaction, regardless of a user's bandwidth, screen size, network, or device capabilities.
- Front end skills include:-
- **Responsive Design** – We use different gadgets like computers, phones, and tablets to look at web pages. The web pages adjust themselves to the device you're using without any extra effort from your end. This is due to the responsive design. One major role of a front-end developer is to understand the responsive design principles and how to implement them on the coding side. It is an intrinsic part of **CSS frameworks** like the **Bootstrap**. These skills are all interconnected and so as you learn one you'll often be making progress in the others at the same time.

Front End Skills



- **Build Tools** – The modern web browsers come equipped with developer tools for testing and debugging. These tools allow you to test the web pages in the browser itself and finds out how the page is interpreting the code. **Browser developer tools** usually consist of an inspector and a **JavaScript console**. The inspector allows you to see what the runtime HTML on your page looks like, what CSS is associated with each element on the page. The JS console allows you to view any errors that occur as the browser tries to execute your JS code.
- **Version Control/Git** – Version Control is the process of tracking and controlling changes to your source code so that you don't have to start from the beginning if anything goes wrong. It is a tool that is used to track the changes made previously so that you can go back to a previous version of your work and find out what went wrong without tearing the whole thing down.

Front End Frameworks

- **CSS** and **JavaScript frameworks** are collections of CSS or JS files that perform different tasks by providing common functionality. Instead of starting with an empty text document, you start with a code file that has lots of JavaScript present already in it.
- Frameworks have their strengths and weaknesses which makes it important to choose the best framework for the type of website you're building. For example, some JS Frameworks are great for building complex user interfaces, while others excel at displaying all of your site's content.

- Some of the popular frameworks are:
- **React**
- **Angular**
- **Vue**
- **jQuery**

This was about the front-end skills that are required to build a website. Now, let's move on and know about the different back-end skills.

Backend Skills

- The back-end layer forms a dynamic connection between the front-end and the database. To get this layer working it's important to know at least one of the programming languages such as **Python, Java, PHP, Ruby**, etc. and knowledge of server-side frameworks such as **NodeJS** is mandatory.

- **Python** is an open-source, **object-oriented** programming language that was released in 1991 and ever since it has become one of the favorite languages of most software and **web developers**.
- **Java** is an open-source, high-level programming language which was released by Sun Microsystems in 1996. It follows the **Write Once Run Anywhere** (WORA) approach that makes it compatible to run on any platform.

Backend Skills

.

- **PHP** is an open-source, **server-side scripting language** used to develop the back-end logic of an application. It is a powerful tool for making dynamic and interactive websites.
- **NodeJS** is an open-source, **JavaScript framework** used specifically for creating the back end or the server-side of an application. Through NodeJS, JavaScript can now finally run on the server-side of the web.

Data and Database

- The data layer is a massive warehouse of information. It contains a **database repository** which captures and stores information from the front-end, through the back-end. A prerequisite is to have knowledge of how data is stored, edited, retrieved, etc. An understanding of Databases such as SQL(MySQL) or No-SQL(MongoDB) is a must.
  - **MySQL** is an open-source, Relational Database Management System that provides multi-user access and supports multi storage engines.
  - **MongoDB** is known for its ease of use and its quickness in handling a large amount of data. It is an open-source, object-oriented, NoSQL database which is highly scalable and is efficient in handling unstructured data.

Server and Deployment Skills

- Servers are basically computers that store website files and other resources like databases.

- **Server Setup**

- To make a website accessible publicly on the internet, it needs to be installed on a server. Once you have your domain name and server space, it's time to set up the site on the server. The first thing is to direct the domain name to the server's unique **IP address**. Then you need to set up website files and finally the **database** and other configurations.

- **Deployment Tools**

- To get files from your own computer up to your server, you need a **protocol**. This is basically a method of transporting files or other data to and from a server.

- The deployment tool stores your **FTP/SFTP** settings, and when a change is pushed in Git to the master branch, the tool will transfer the files for you. So, there is no need to remember which files you changed, reducing the number of mistakes you make.

THANK YOU

TO BE CONTINUED

NEXT TOPIC ->

HTML