Class 2

What is Full Stack Website Development?

What is “Web”?

“Web” is a small entity which belongs to bigger entity called “Internet”. There are many internet services/products. One of them is the web. Now, “Web” is a system where many documents, many data, many resources are interconnected. And we can access all these resources through internet.

What is “Website Development”?

“Development” means “Creating or Building”. Website Development is a process of creating “Websites or Web Application (app)”.

Now here the question arises that what is the basic difference between website and web app?

What is “Website”?

“Website” is a web page or multiple web pages where static web content is located. The term “static content” means “fixed content”. It is only Read-only data/blogs, where the user comes in, read the data and left. There is no user-interaction.

What is “Web App”?

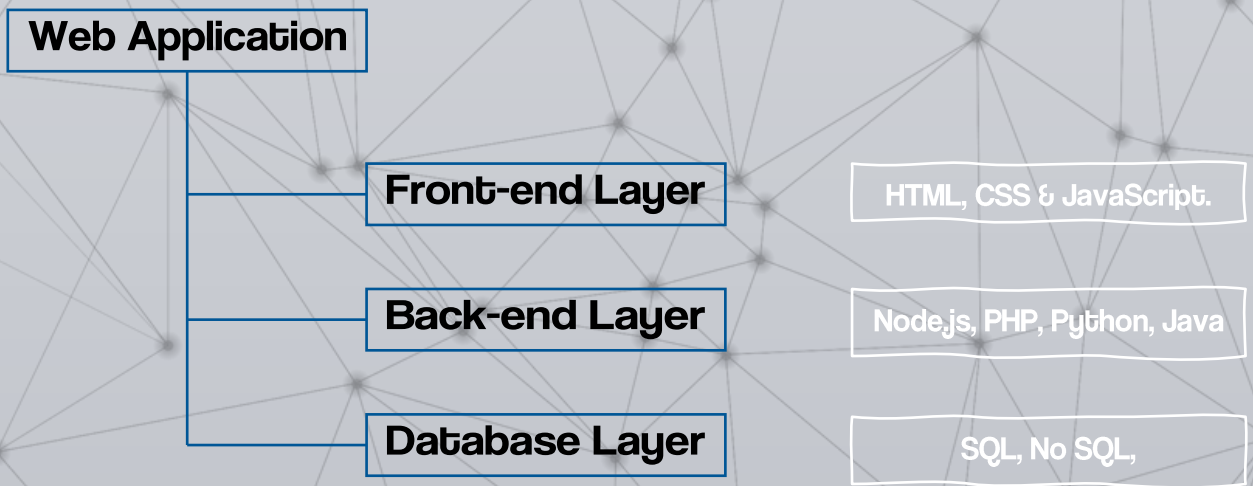
“Web App” is dynamic in nature where data changes if reloaded. So web app is a term that we create a product, web page with which user can interact. And things can change on your web page based on user interaction.

How do we decide whether to build a website or a web app?

It depends on a “use-case”. A “use-case” is a specific scenario or example of how a system, product, or technology can be applied to solve a particular problem or achieve a goal. It describes the situation, the users involved, and how the system addresses their needs.

Breaking a Web App

Web App is divided into three layers that are front-end layer, back-end layer and database layer.



with reference to above fig we discuss some basic terminologies below:

1. Front-end Layer

- "Responsible for the user interface and experience of the application."
- Technologies: HTML, CSS, and JavaScript.
- The person who works on front-end layer called "front-end engineer, front-end designer, front-end developer, UI/UX designer".
- Front-End Developer and Front-End Engineer are the most accurate terms for people who build the front-end using code.
- UI/UX Designer focuses on designing the user interface and experience, not coding it.
- Front-End Designer is less common but could be used to describe someone who handles design elements in the front-end layer.

2. Back-end Layer

- "Handles the application logic, server-side processing, and interactions with the database."
- Technologies: Node.js, PHP, Python, Java.
- The person who works on back-end layer called "back-end engineer, back-end developer, server-side developer".

3. Database Layer

- "Manages data storage, retrieval, and organization for the application."
- Technologies: SQL databases (like MySQL, PostgreSQL) and NoSQL databases (like MongoDB, Cassandra).
- The person who works on database layer called "database developer, database administrator (DBA)".

Each layer works together to deliver a complete, responsive, and functional web application.

Who is Full Stack Web Developer?

A Full Stack Web Developer is a developer who works on all three **front-end** (client-side) and **back-end** (server-side) and **database layer** of a website or web application. This means they have a broad range of skills that allow them to handle the entire web development process, from building the user interface to managing databases and server logic.

Web Development Stacks

There isn't a fixed number of web development stacks, as they can be combined in various ways based on project needs. Here are some of the most popular stacks:

- **LAMP** (Linux, Apache, MySQL, PHP) – Traditional stack for PHP applications.
- **LEMP** (Linux, Nginx, MySQL, PHP) – Similar to LAMP but uses Nginx for high-traffic sites.
- **MEAN** (MongoDB, Express.js, Angular, Node.js) – Full JavaScript stack for SPAs.
- **MERN** (MongoDB, Express.js, React, Node.js) – Like MEAN but uses React.
- **JAM** stack (JavaScript, APIs, Markup) – Static sites with serverless functions.
- **Django** Stack (Django, PostgreSQL/MySQL) – Python-based for data-driven apps.
- **Ruby** on Rails (Rails, PostgreSQL/MySQL) – Rapid development for startups.
- **.NET** (ASP.NET, SQL Server) – Microsoft stack for enterprise applications.
- **Spring Boot** (Spring Boot, Java, MySQL/PostgreSQL) – Java stack for large applications.
- **Serverless** Stack (AWS Lambda, DynamoDB) – For serverless microservices.