**DOCUMENTATION OF PREVIOUS TOPICS**

**What is Front end?**

The visible part of the web pages to the clients is known as the front end. In FE we can view and collect the data. As a developer point of view the developer can view and collect the data that is Known as the UI(user interface). If the user enters the data into it then it is known as the user Experience.

The technologies used in Front end they are:

1.HTML

2.CSS

3.Java Script

**HTML:** Html is used to create Web pages and it is used to provide the user interface to the clients .

**CSS:** It is used to style the web pages and it is helpful in building the application in a attractive mode to the clients and the user.

**Java Script:**

The brain that makes it interactive .It responds to the user clicks and shows messages and alerts. Java Script is used to improve UI/UX makes user interactive.

**Frontend Frameworks/Libraries:** React,Angular,Vue.js.Svelte.

**Back-End Development (Server-side)**

This is the part of the application that manages and serves data. Back-end developers create the logic, databases, and server-side infrastructure that make the web application run.

**Technologies:**

Programming languages: Node.js (JavaScript), Python (Django, Flask), Ruby (Rails), PHP, Java (Spring), C# (.NET)

**Database**

A database is an organized collection of data that can be easily accessed, managed, and updated. It is designed to store data in a structured way so that it can be retrieved, updated, or manipulated efficiently.

**Relational Databases (RDBMS):**

Store data in structured tables with predefined relationships between them

MySQL

Oracle Database

NoSQL Databases.

**NOsql Database:**

**SDLC (SOFTWARE DEVELOPMENT LIFE CYCLE):**

The **Software Development Life Cycle (SDLC)** is a structured process used by development teams to design, develop, test, and deploy software. It provides a systematic framework for developing software applications and ensures that the software is of high quality, meets user requirements, and is delivered on time and within budget.

The SDLC consists of several stages, typically including:

1. **Planning**:
   * Identifying the purpose of the software, defining requirements, estimating costs, and setting goals and timelines.
   * Planning helps outline the scope, resources, and schedule.
2. **System Analysis**:
   * Assessing the technical, operational, and financial feasibility of the project.
   * Determining if the project is viable and worth pursuing.
3. **System Design**:
   * Defining system architecture and design.
   * Creating high-level and low-level design documents that include database design, software architecture, and user interfaces.
4. **Implementation (Coding/Development)**:
   * Writing the actual code based on the system design documents.
   * Developers implement the functionalities defined in the design phase.
5. **Testing**:
   * Testing the software to identify bugs and ensure it functions correctly.
   * Types of testing may include unit testing, integration testing, system testing, and user acceptance testing (UAT).
6. **Deployment**:
   * Deploying the software to a live environment for end-users.
   * In some cases, deployment might be done in stages (e.g., beta testing, phased rollout).
7. **Maintenance and Support**:
   * After deployment, the software may require updates, bug fixes, and ongoing support to ensure it remains functional.
   * Maintenance can involve patches, performance optimization, or adding new features.

**DevOps and Deployment**

Full-stack developers often work with DevOps tools and practices to deploy and maintain applications.

**Server**

A server is a computer, device, or software program that provides services, resources, or data to other computers, known as clients, over a network. Servers are typically more powerful and have greater storage and processing capacity than client devices

**Client**

A client is a computer, device, or software program that makes requests to a server for resources, data, or services. The client usually interacts directly with the end-user and displays data received from the server.

**What is an API?**

An API (Application Programming Interface) is a set of rules and protocols that allows one software application to interact with another. It defines how different software components should communicate and exchange data, often between a client and a server.

In simpler terms, an API is like a messenger that takes requests from the client (e.g., a web browser or mobile app) to the server, processes the request, and returns the response back to the client.

**What is DNS Lookup?**

DNS (Domain Name System) lookup is the process of translating a human-readable domain name (like [www.example.com](http://www.example.com/)) into its corresponding IP address (such as 192.0.2.1). The DNS system is crucial for navigating the internet, as it enables devices to locate and connect to web servers using domain names instead of numerical IP addresses.

**Browser**

A browser (or web browser) is a software application that allows users to access, retrieve, and view content on the World Wide Web (WWW). It interprets and displays the data found on websites, allowing users to interact with text, images, videos, and other resources.

**Local storage**

Local Storage is a web storage feature provided by web browsers that allows websites to store data locally on a user's device, within the browser, in a key-value pair format.

**What is Static web page and Dynamic Web page?**

**1.Static Web page:** Static web is a type of web page that delivers the content and once we provide the data then it is fixed we cannot change or we cannot add the data in that page is known as the static web page.

EX: Wikipedia.

**2.Dynamic Web page:** Dynamic web page is a type of page that delivers the content and we can add the data and manipulate and update the data day by day then it is known as the dynamic web page.

**EX:** Amazon, Flipkart, Zomato etc.

**How Full Stack Web application works?**

1.By using some technologies we can develop a full stack application such as E-commerce website. The Technologies are such as:

1.HTML  
2.CSS

3.JAVASCRIPT

4.NODE.JS

5.SQL

2.firstly by using Html we can create a UI (user interface) and we will add styling to that user interface.

3.After opening the application we will get one Register page if we click on the that we will get some credentials such as {username, password, DOB, etc} .

4.after we enter the credentials such as Username and password and DOB then the data will be sent to the backend through API

5.let us consider my API as Register API by using this it will sent the data to the backend.

6.In backend we will write the logics such as for the validation of the data.

7.For example the credentials what we have entered in UI it will comes to backend through API and if the Username or Password is already exist then it will send the message as username Already Exist. If not it will says that you have registered successfully.

8.The new data which came from the FE it will be stored in the database.

9.So this whole process runs as a request ->Response format.

10.So this Whole process is known as the Full stack application.