15 JAN ‘23 BASE BUILDING

INTERNET AND HOW IT WORKS:

The internet is a network of computers and other devices that are connected together. These devices use a communication protocol called TCP/IP to send and receive data, such as websites, emails, and files. When you use the internet, your device sends a request for information to a server, which then sends the information back to your device. This process happens quickly, allowing you to access and share information with others all over the world. The most common way to access the internet is through a wired or wireless connection provided by an Internet Service Provider (ISP).

WHAT IS A WEB BROWSER:

A web browser is a software application that allows you to access, navigate, and view the content on the internet. The most popular web browsers include Google Chrome, Mozilla Firefox, Microsoft Edge, Safari and Opera. When you enter a website's address into a web browser, the browser sends a request to the server where the website is hosted. The server then sends the website's information back to the browser, which then displays the website on your device's screen. Web browsers also have features like bookmarks, history, and the ability to save passwords, making it easier for you to navigate and access your frequently visited websites.

WHAT IS BROWSER ENGINE:

A browser engine is the component of a web browser that is responsible for interpreting and displaying web content, such as HTML, CSS, and JavaScript. It acts as a bridge between the browser's interface (e.g. the address bar, back button) and the rendering engine which takes the web content and displays it on the browser.

A browser engine is typically made up of two main components: the layout engine and the JavaScript engine. The layout engine is responsible for interpreting and rendering HTML and CSS and determining the layout and formatting of the web page. The JavaScript engine is responsible for executing any JavaScript code on the page, which can interact with the page's content and layout.

Examples of browser engines are: -

WebKit (used by Safari, Chrome, and other browsers) -

Gecko (used by Firefox) -

Trident (used by Internet Explorer) -

Blink (used by chrome, opera) -

EdgeHTML (used by Edge)

Each browser uses a specific browser engine to display web pages, which can affect how the pages look and function across different browsers.

WHAT IS STATIC WEBSITE:

A static website is a type of website that contains fixed content that does not change, regardless of who views it or when it is viewed. The content of a static website is stored in files on a server and is served to the user's web browser as is, without any server-side processing.

Static websites are typically built using HTML, CSS, and JavaScript, and are often simpler in design and functionality compared to dynamic websites. They are also easier to create, host and maintain.

Examples of static websites include personal blogs, portfolios, brochure sites, and landing pages. They are also suitable for small business websites, informational websites, and simple e-commerce websites where the products or services offered do not change frequently.

Static websites can be hosted on a web server and can be served over the internet using a domain name. Users can access the website by typing the domain name into a web browser.

It is worth noting that as technology advanced some frameworks and tools have been developed for creating static sites that allow to add dynamic functionality like dynamic navigation, forms, and client-side rendering, so called JAMstack (JavaScript, APIs, Markup) approach.

Here are some Advantages and Disadvantages of static websites:

ADVANTAGES:

- Simplicity: Static websites are simple to create, host, and maintain, making them a cost-effective solution for small businesses and individuals.
- Fast loading time: Static websites are lightweight and load quickly, making them an excellent choice for users with slow internet connections or for websites with a lot of traffic.
- High performance: Static websites can handle many requests and high traffic loads with minimal server resources.
- Better security: Static websites are less vulnerable to hacking and other forms of cyber-attacks, as they do not have databases or other sensitive information stored on the server.
- Easy to host: Static websites can be hosted on a wide range of platforms, including web servers, content delivery networks (CDNs), and cloud storage services.
- Better SEO: Static websites can be optimized for search engines more easily, as they have a simple structure and do not require server-side processing.
- Easy to Backup: since the website is just a collection of files, it is quite easy to back up the whole website by copying the files on a different location.
- -Cost efficient: Static websites are cheaper to develop and maintain compared to dynamic websites, and often do not require a dedicated server or specialized hosting.
- Better control of the design: since the website is just a collection of files, developers have more control over the design, layout, and overall structure of the website.

DISADVANTAGES:

- Limited functionality: Static websites have limited functionality and cannot handle complex interactions or user input. They are not suitable for websites that require user registration, forms, or other dynamic features.
- No database interaction: Static websites do not have a database to store or retrieve information, so they cannot handle tasks such as user authentication, content management, or e-commerce.
- No server-side processing: Static websites cannot perform server-side processing, so they cannot handle tasks such as image manipulation, data validation, or email sending.
- Limited scalability: Static websites are not easily scalable, and adding new pages or features can be difficult and time-consuming.
- Lack of security: Static websites do not have built-in security features, making them vulnerable to hacking and other forms of cyber-attacks.
- Limited analytics: Static websites may lack analytics capabilities, which make it harder to track user behavior and gain insights about the website's performance.
- Harder for content management: Static websites can make it difficult for non-technical users to manage and update content.
- Limited caching: Static websites do not have caching capabilities that can speed up the loading time.

Note that with JAMstack approach and modern development tools these disadvantages can be mitigated.

WHAT IS DYNAMIC WEBSITE:

A dynamic website is a type of website that generates content on the fly, based on user input or other dynamic data. Unlike static websites, dynamic websites use server-side scripting to generate the HTML, CSS, and JavaScript that is sent to the user's web browser. This allows dynamic websites to provide a more interactive and personalized experience for the user.

Dynamic websites are typically built using server-side technologies such as PHP, Ruby on Rails, Python, or .NET, and are often more complex in design and functionality compared to static websites. They can handle tasks such as user registration, forms, e-commerce, and content management systems.

Dynamic websites are built using server-side technologies and a database to store the information and can be served over the internet using a domain name. They can handle tasks such as user registration, forms, e-commerce, and content management systems.

Examples of dynamic websites include e-commerce websites, social media websites, and content management systems (CMS) such as WordPress or Drupal.

Dynamic websites are more suitable for websites that require frequent updates, user interactions, or a large amount of data. However, they also require more resources and are more complex to develop, maintain and host.

Here are some Advantages and Disadvantages of dynamic websites:

ADVANTAGES:

- Interactive and personalized experience: Dynamic websites can provide a more interactive and personalized experience for the user, by generating content on the fly based on user input or other dynamic data.
- Database integration: Dynamic websites can interact with a database to store, retrieve and update information, which allows for tasks such as user authentication, content management, and e-commerce.
- Server-side processing: Dynamic websites can perform server-side processing, which allows for tasks such as image manipulation, data validation, and email sending.
- Scalability: Dynamic websites are easily scalable, and adding new pages or features is relatively simple.
- Advanced functionality: Dynamic websites can handle complex interactions and user input, and can provide advanced functionality such as forums, blogs, and e-commerce.
- Better analytics: Dynamic websites can have built-in analytics capabilities, which make it easier to track user behavior and gain insights about the website's performance.
- Better content management: Dynamic websites often have a content management system (CMS) which allows non-technical users to manage and update content.
- Caching: Dynamic websites can have caching capabilities that can speed up the loading time for the users.

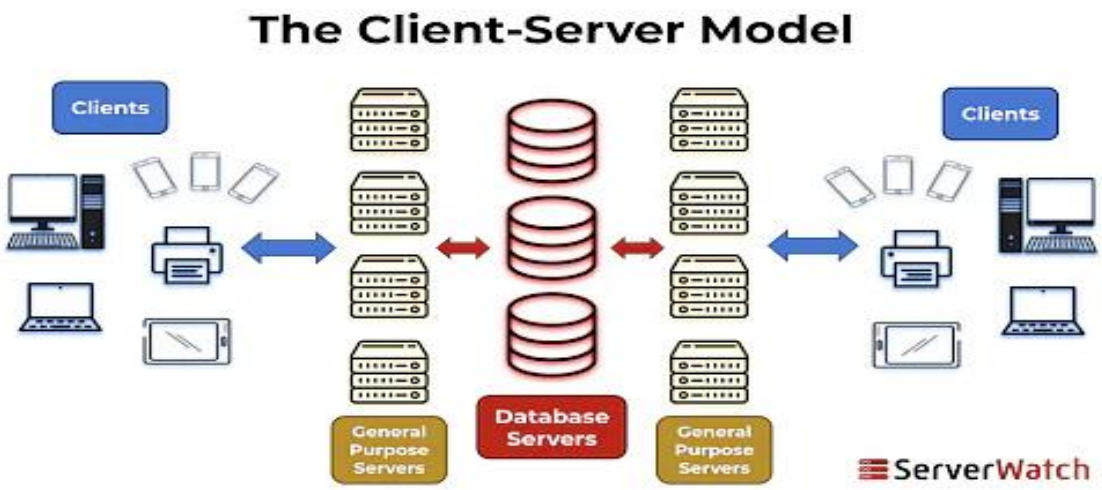
- Better security: Dynamic websites can have built-in security features, such as user authentication and access control, which can protect against hacking and other forms of cyber-attacks.

DISADVANTAGES:

- Complexity: Dynamic websites are typically more complex to create, host, and maintain, as they require server-side scripting and database integration.
- Higher cost: Dynamic websites are more expensive to develop and maintain compared to static websites, and often require a dedicated server or specialized hosting.
- Slow loading time: Dynamic websites can be slower to load than static websites, as they require server-side processing and database queries.
- Higher resource usage: Dynamic websites can consume more server resources, such as memory and processing power, which can lead to performance issues.
- Vulnerability to hacking: Dynamic websites are more vulnerable to hacking and other forms of cyber-attacks, as they have databases and other sensitive information stored on the server.
- Dependency on server-side technologies: Dynamic websites depend on server-side technologies, and if these technologies become obsolete or unsupported, it can cause issues with the website's functionality.
- Harder to backup: since dynamic websites rely on a database, it can be harder to back up the whole website and requires specific tools to back up the database and the files.
- Less control over design: since the website is generated on the fly, developers may have less control over the design, layout, and overall structure of the website.
- Security vulnerability: due to the complexity of the website, it can be harder to ensure the website is secure and is more vulnerable to hacking and other security threats.

CLIENT AND SERVER:

Client-Server architecture is a network architecture in which multiple client devices (such as computers, smartphones, tablets) connect to and access centralized server(s) to request and receive services or data. The client and server communicate with each other using a specific protocol, such as HTTP or FTP, and the server responds to the client's request by providing the requested service or data.



In this architecture, the client device (also known as the "front-end") is responsible for displaying the user interface and handling user input, while the server (also known as the "back-end") is responsible for processing requests, managing data, and providing services

In web-based applications, the client is typically a web browser, and the server is a web server. The client sends a request to the server using the HTTP protocol, and the server responds by sending the requested web page or data.

This architecture is widely used in many different types of applications, such as email, file transfer, and online databases, as it allows for the separation of the user interface and the data processing. It also allows for scalability, as multiple clients can connect to the same server simultaneously.

It's important to note that the client-server architecture has evolved with the advancements in technology, now we have new architectures like Microservices, Serverless and edge computing, however the basic concept of the client and server remains the same.