

Outlines



WELCOME TO THE TEAM!

Outlines

- **What is the web?**
- **What are web servers?**
- **How does a web application work?**
- **What is the Internet and intranet?**
- **Front-end and back-end development.**
- **How data is accessed and shared via browsers.**
- **Server computers, client computers, and their communication.**
- **Browsers and their functionalities.**
- **Request-Response Model and how it works.**
- **Databases and database servers.**
- **Static vs. dynamic websites**
- **UI/UX in web development.**
- **Full-stack web development and its components.**



Outlines



Full Stack Development

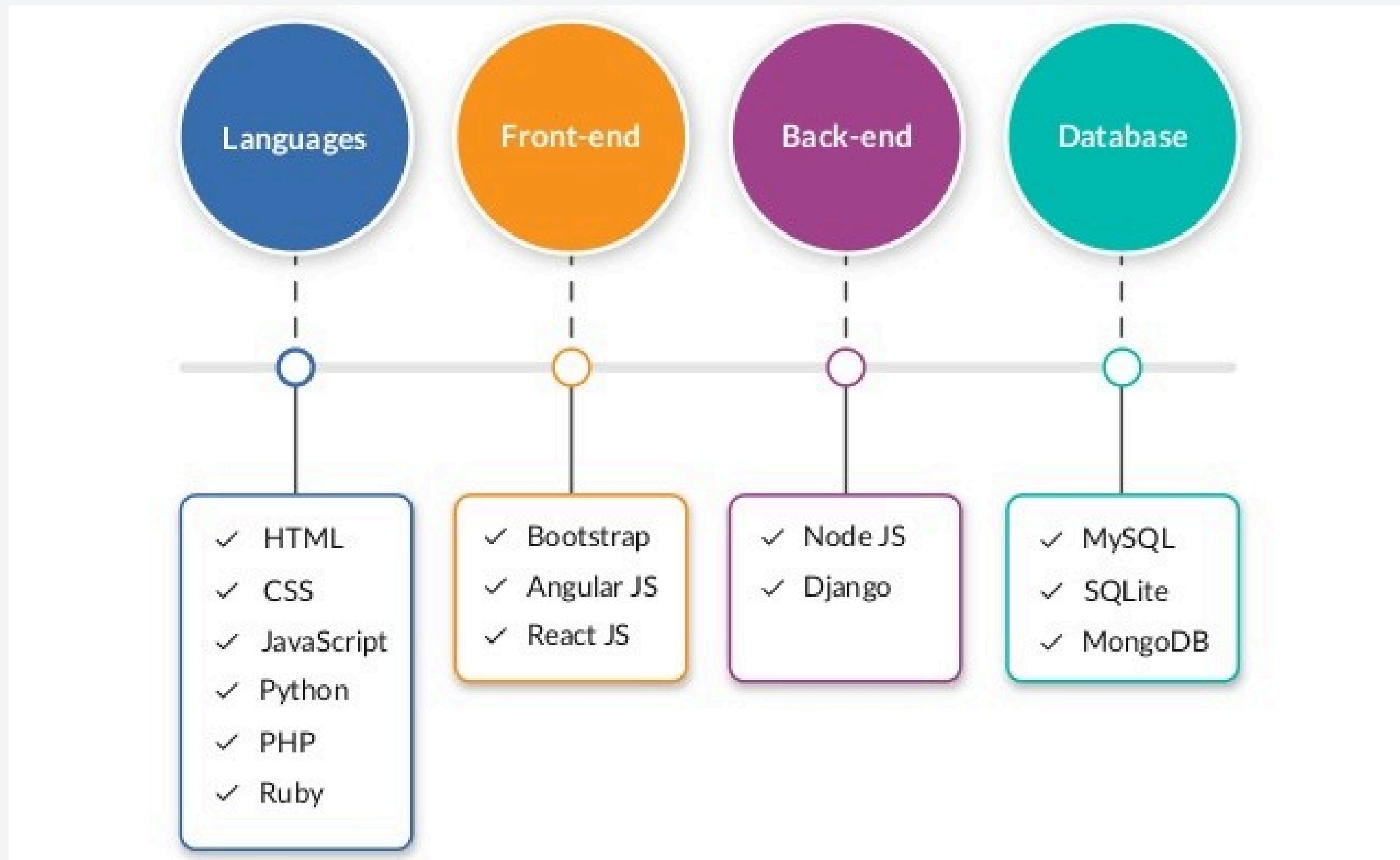
Front-end: User interface (ui) using HTML, CSS, and JavaScript.

Back-end: Server-side logic, database management, and APIs.

Key tools: React, Angular, Node.js, Python.



Full Stack Development



What is WEB

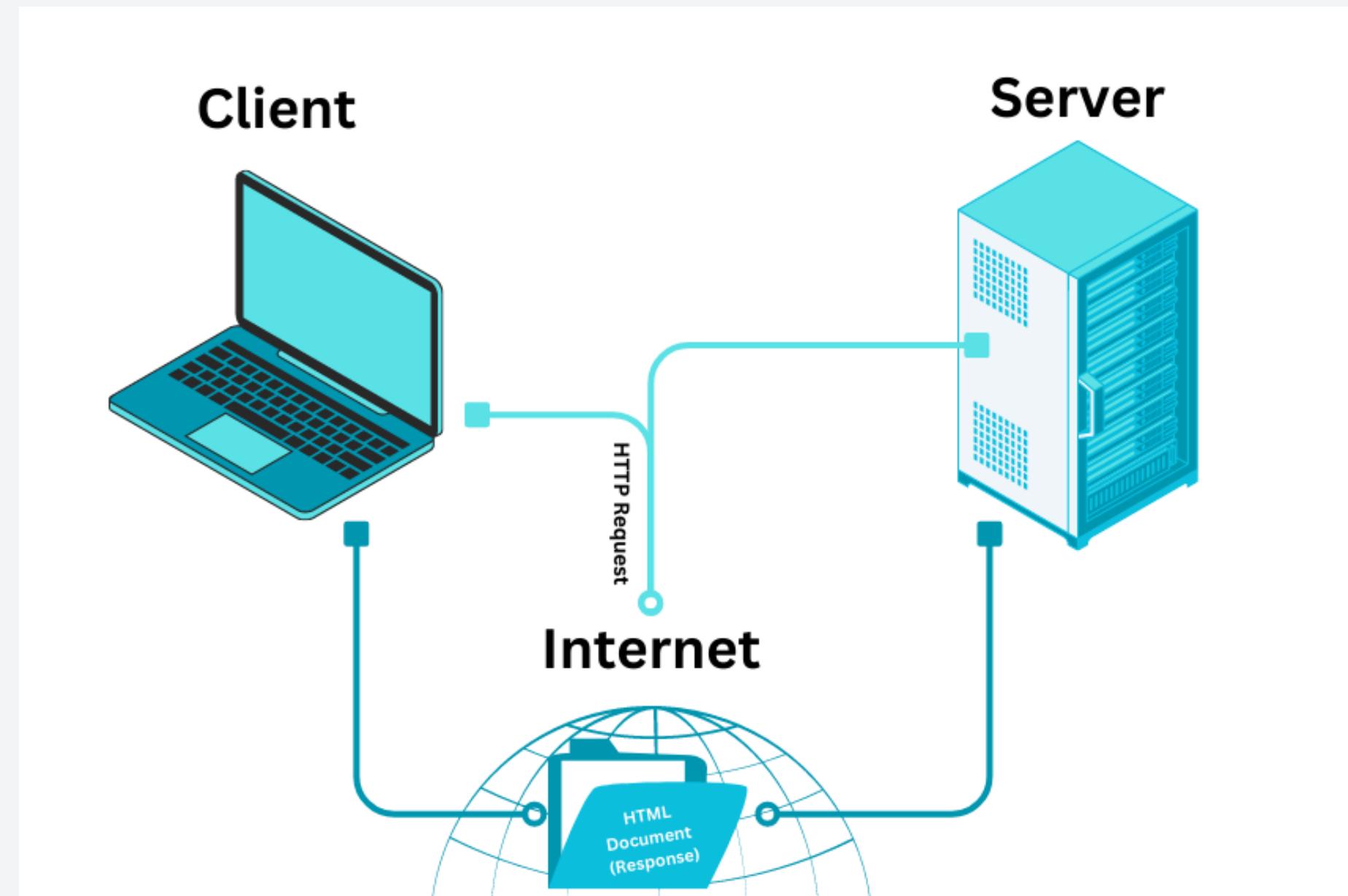
The web, or World Wide Web (www), is a collection of **interconnected web pages** that are accessible through the **internet**. It's a system that allows users to share and access information, such as text, images, videos, and audio.

It relies on protocols like HTTP and technologies like HTML, CSS, and JavaScript.



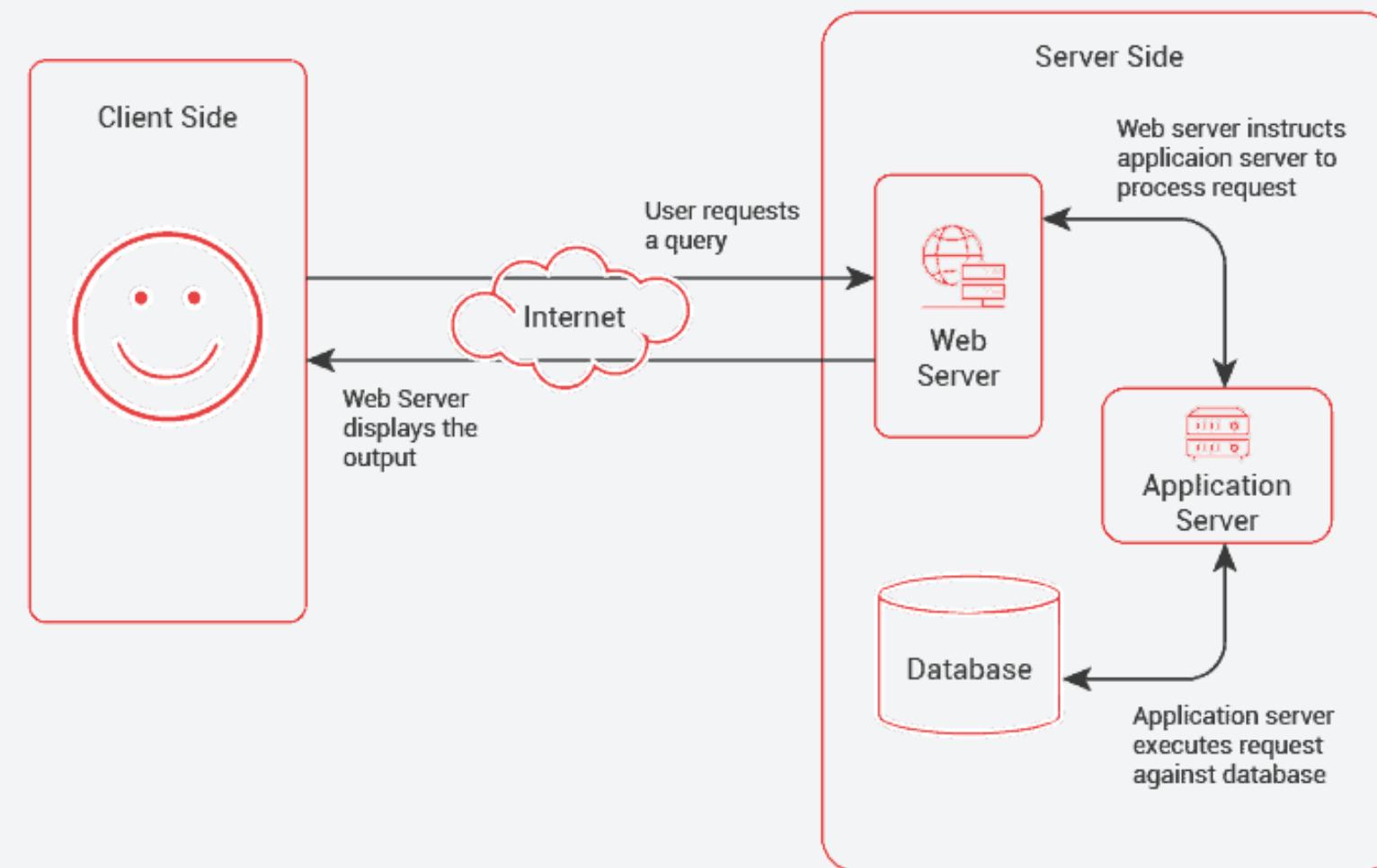
Web Server

- A web server is a **computer system** that stores and delivers web content to users **over the internet**. Web servers are made up of hardware and software, and are always connected to the internet.

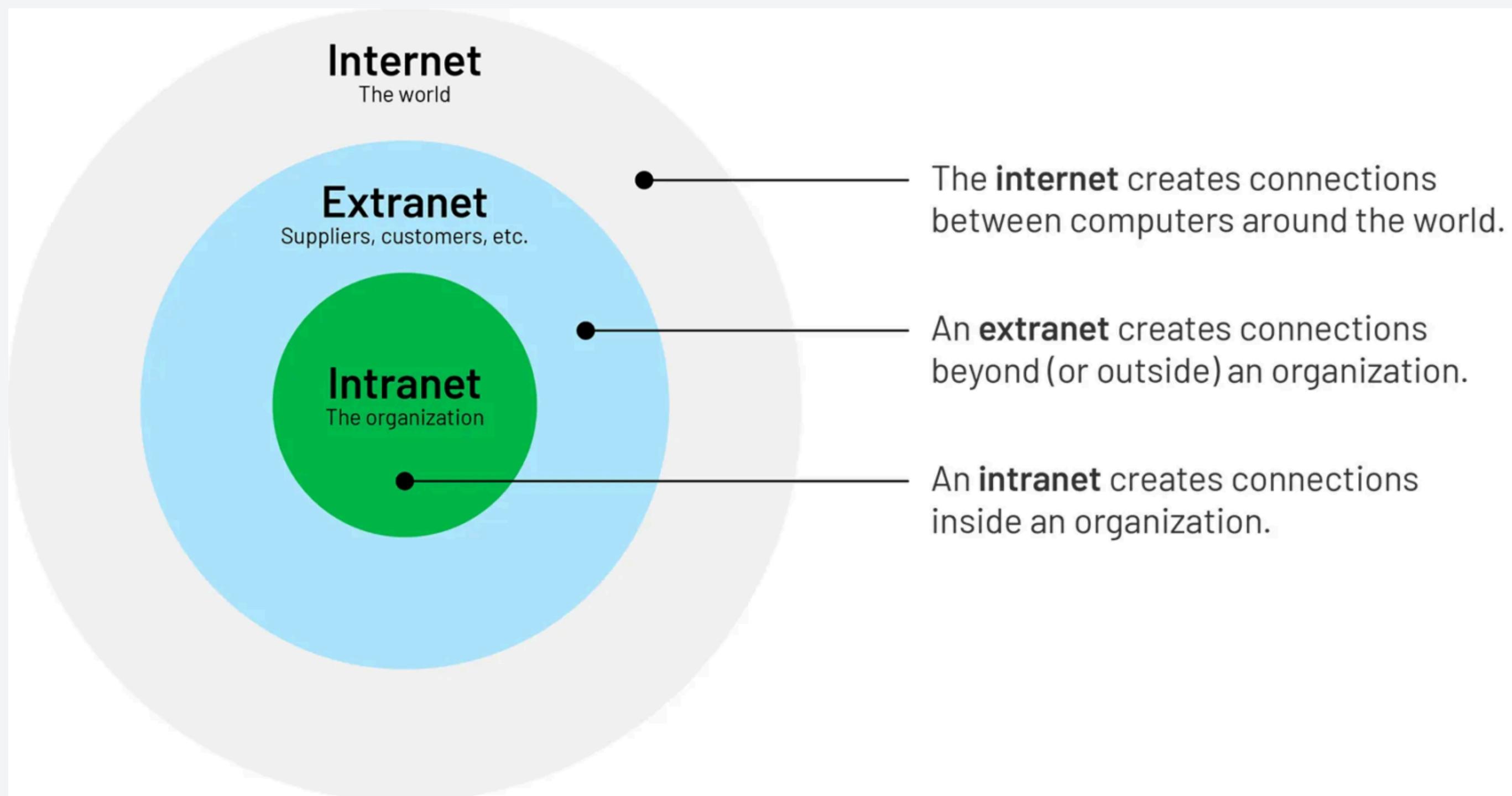


How Does a Web Application Work?

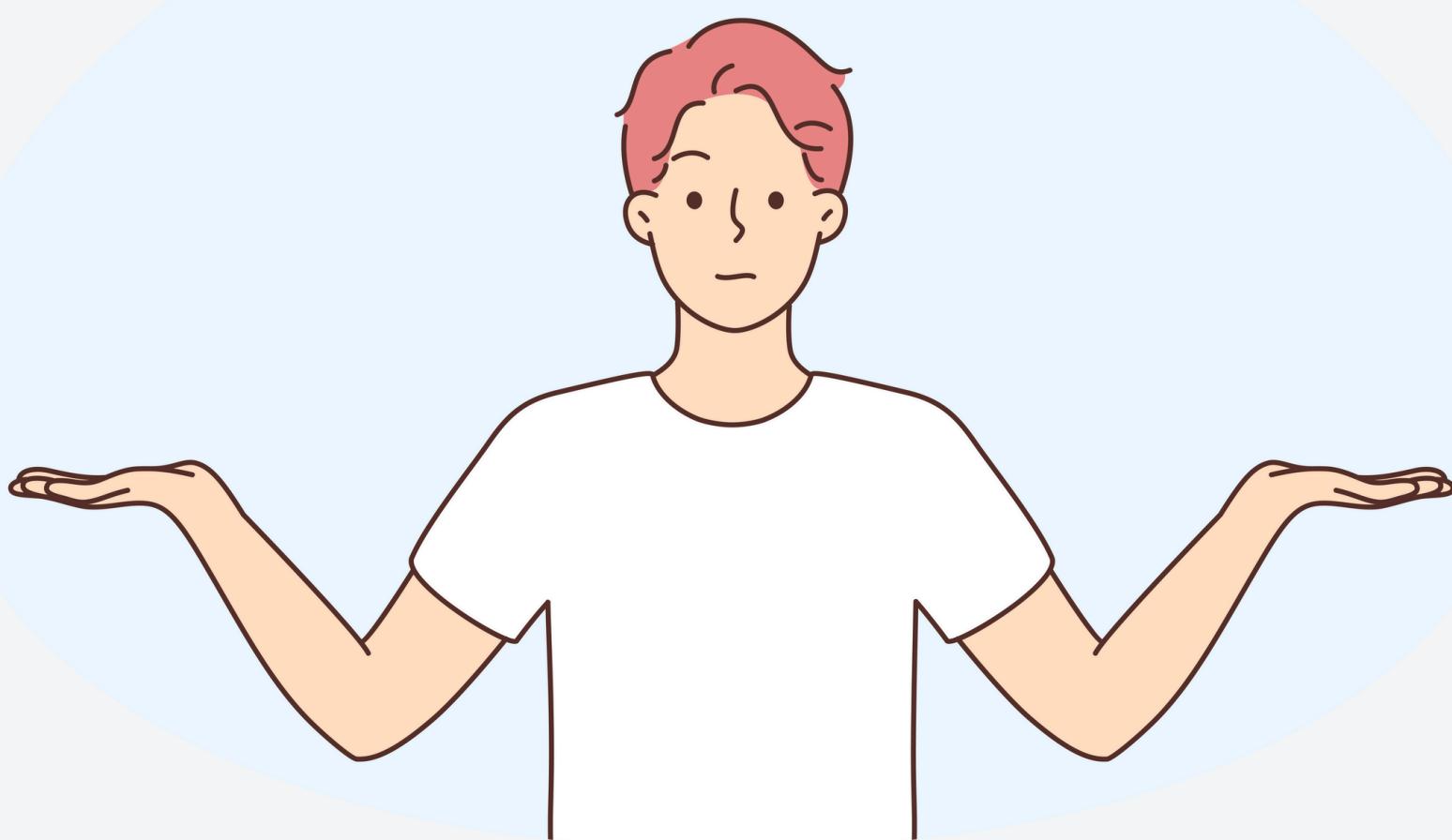
- User interacts with the front-end via a browser.
- Requests are sent to the back-end server, which processes and retrieves data.
- Server responds with the requested information



Internet VS Intranet VS Extranet



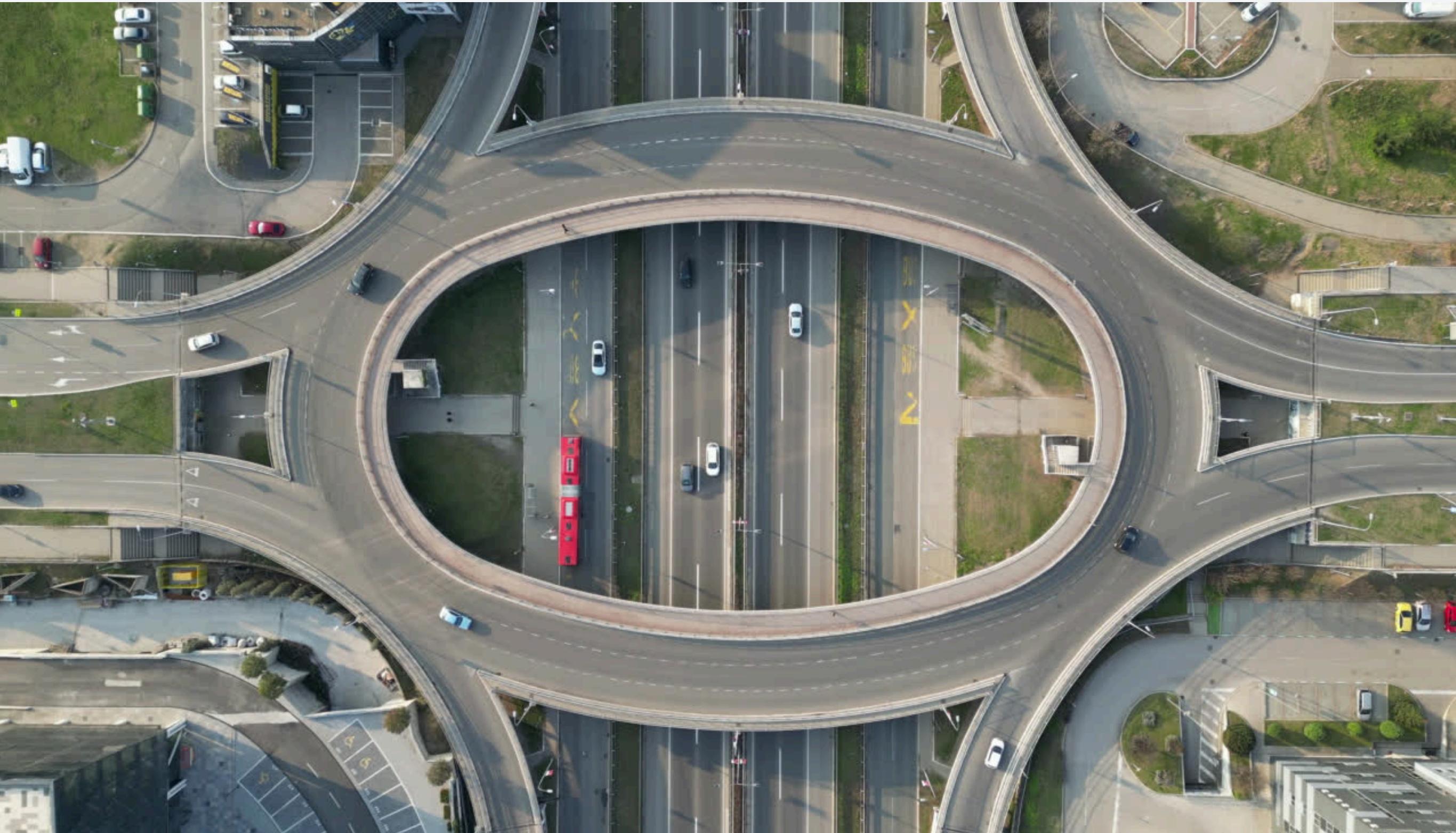
**Web VS
Internet**



Web VS Internet

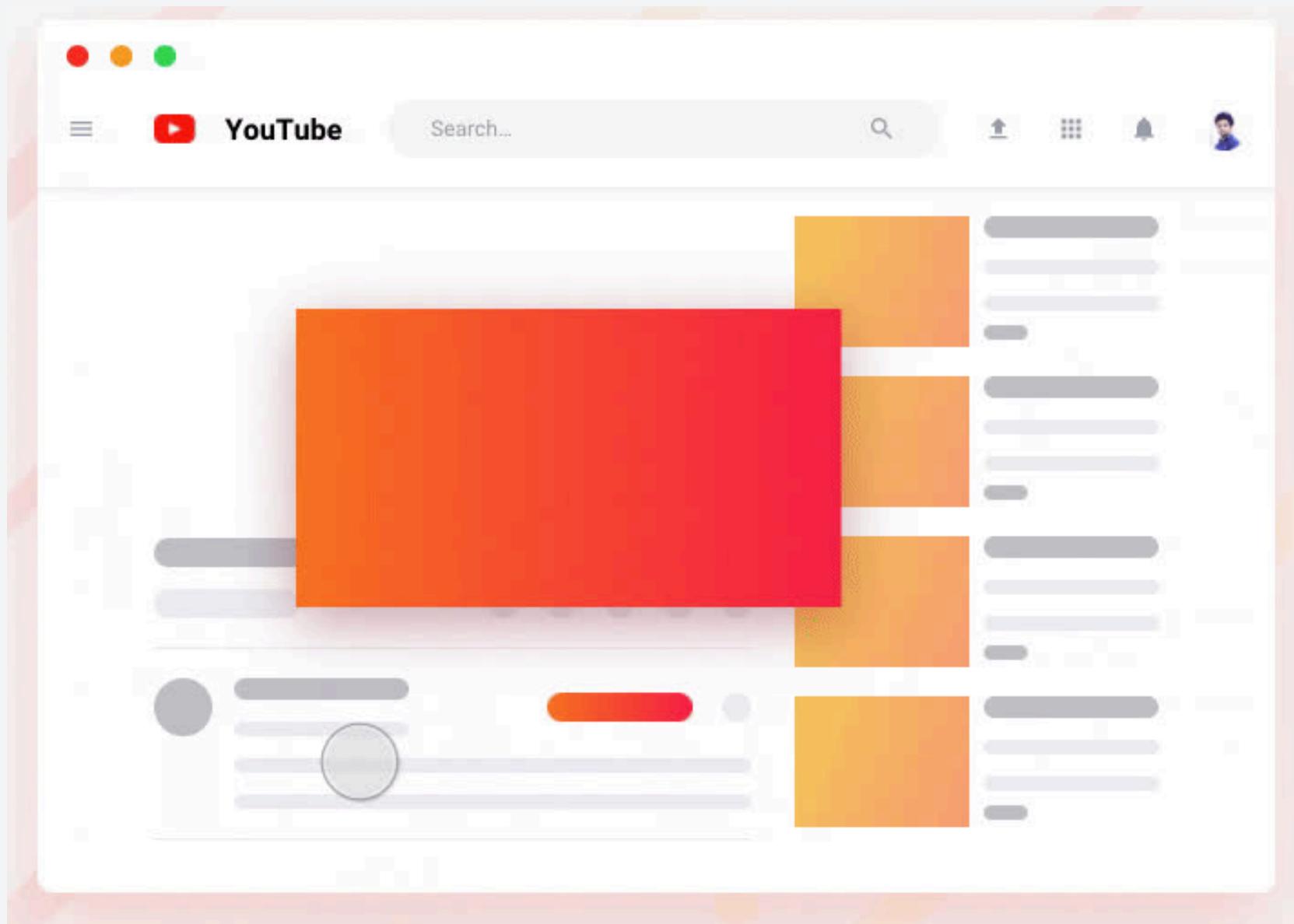
Internet = Highways and Roads

Web = Specific Vehicle on the Road



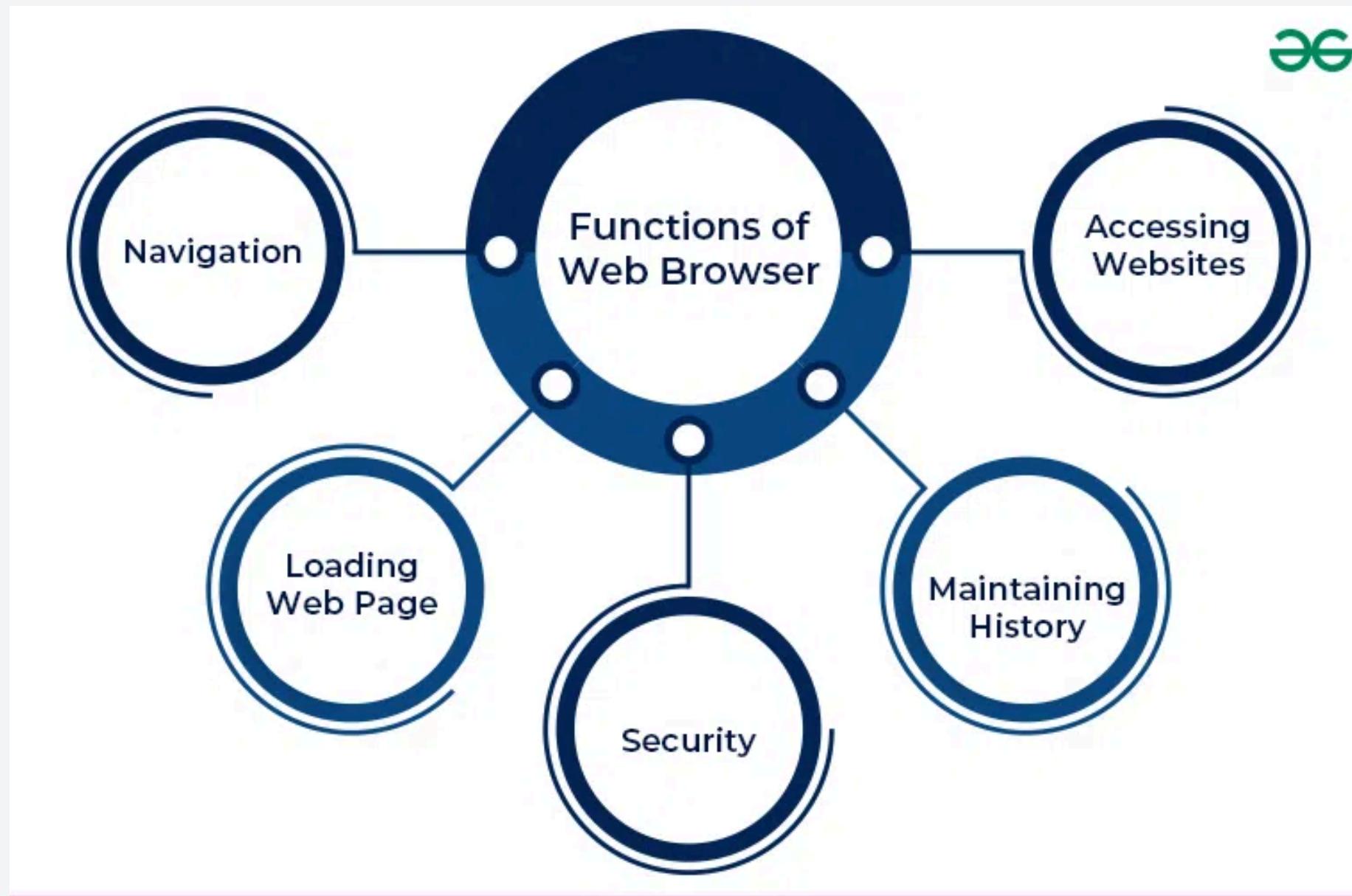
Web VS Internet

- Internet: Provides the connection to transfer video data from YouTube servers to your computer or phone.
- Web: You visit the YouTube website in your browser to search and watch videos.



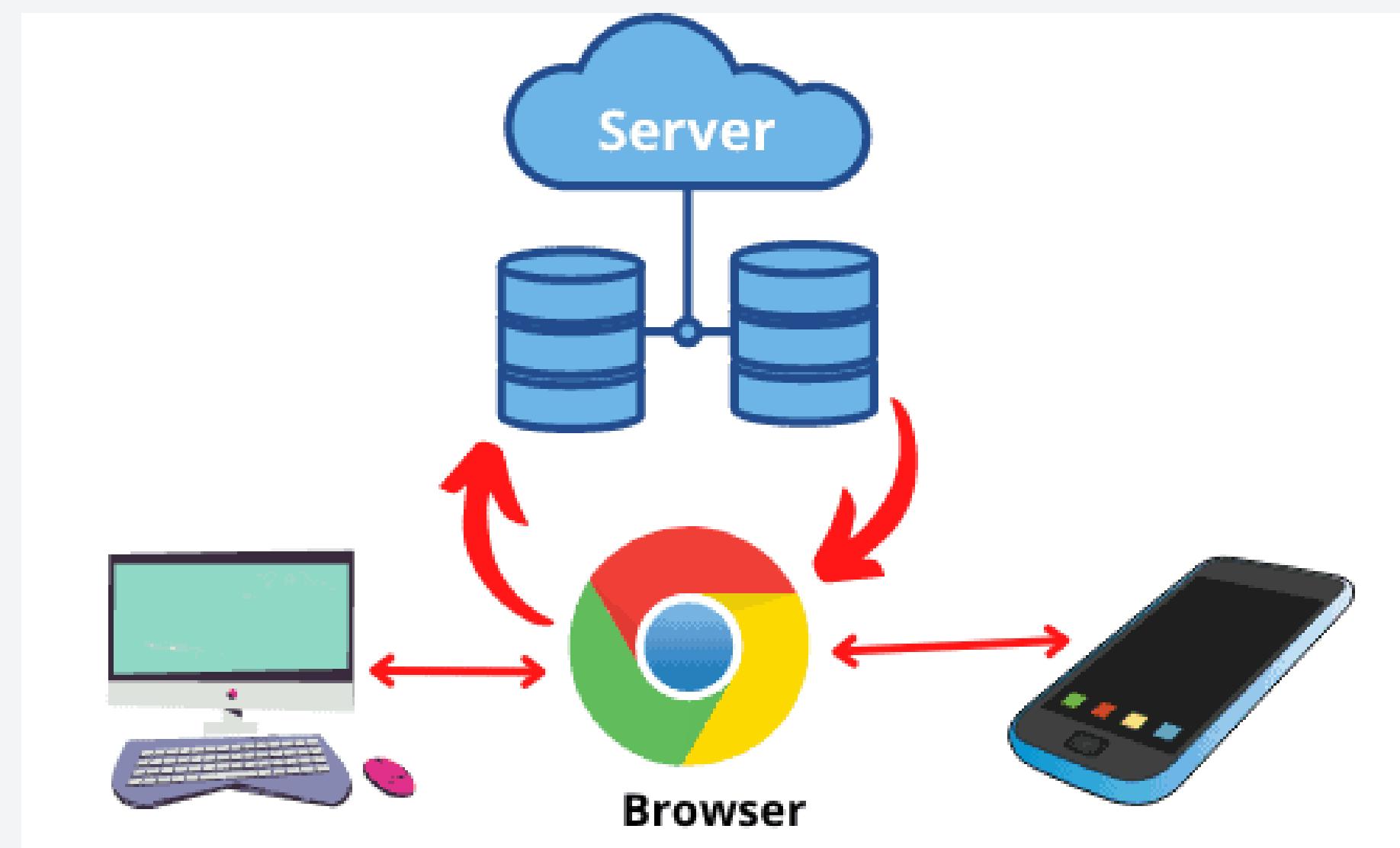
How Data is Accessed and Shared via Browsers

a web browser is an application through which we can access websites on the World Wide Web (www)



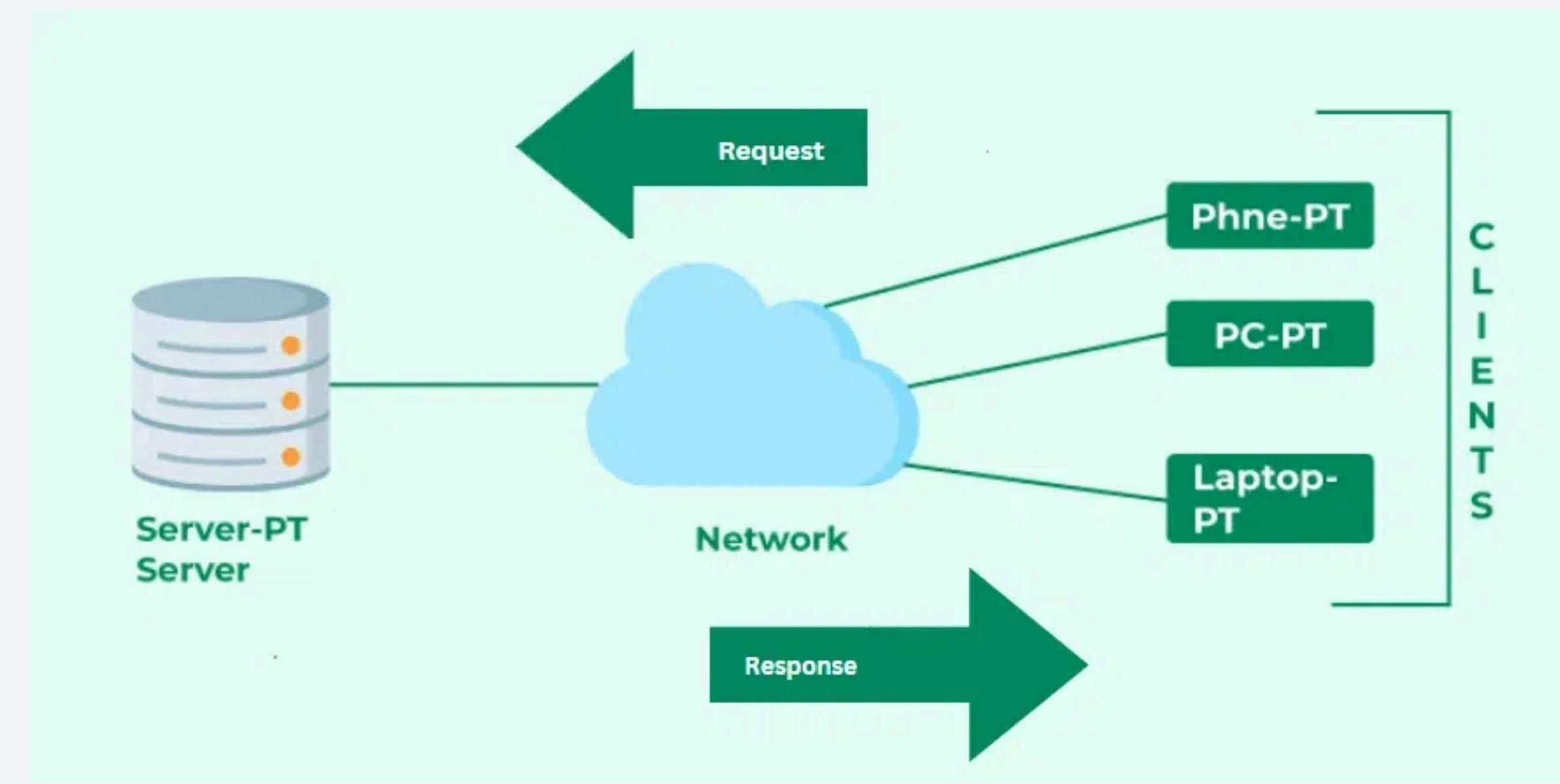
How Data is Accessed and Shared via Browsers

- Browsers send HTTP/HTTPS requests to servers.
- Servers fetch data from databases and return it in HTML, JSON, or XML.
- Browsers render the data into a usable interface.



Server Computers, Client Computers, and Their Communication

- Servers host resources, and clients request them.
- Communication happens through protocols like HTTP, FTP, and WebSocket.



What is a Protocol

- A protocol is a set of rules or standards that define how devices communicate with each other in a network.

Here are a few important protocols and their purposes:

1. HTTP (HyperText Transfer Protocol):

- Used to transfer web pages and resources from servers to browsers.
- Example: When you browse a website, your browser uses HTTP to request the page from a server.

2. FTP (File Transfer Protocol):

- Used for transferring files between a client and a server.
- Example: Uploading a website's files to a hosting server.

3. TCP/IP (Transmission Control Protocol/Internet Protocol):

- The backbone of the Internet, responsible for routing and transmitting data packets across networks.
- Example: When you send an email or visit a website, TCP/IP breaks your data into packets and ensures they reach the correct destination.



What is a Protocol

4. WebSocket:

- Enables full-duplex communication (both client and server can send data simultaneously) for real-time applications.
- Example: Live chat apps or online gaming use WebSocket for fast, real-time communication.

5. SMTP (Simple Mail Transfer Protocol):

- Used to send emails between mail servers.
- Example: When you send an email using Gmail, SMTP ensures it gets delivered to the recipient's server.

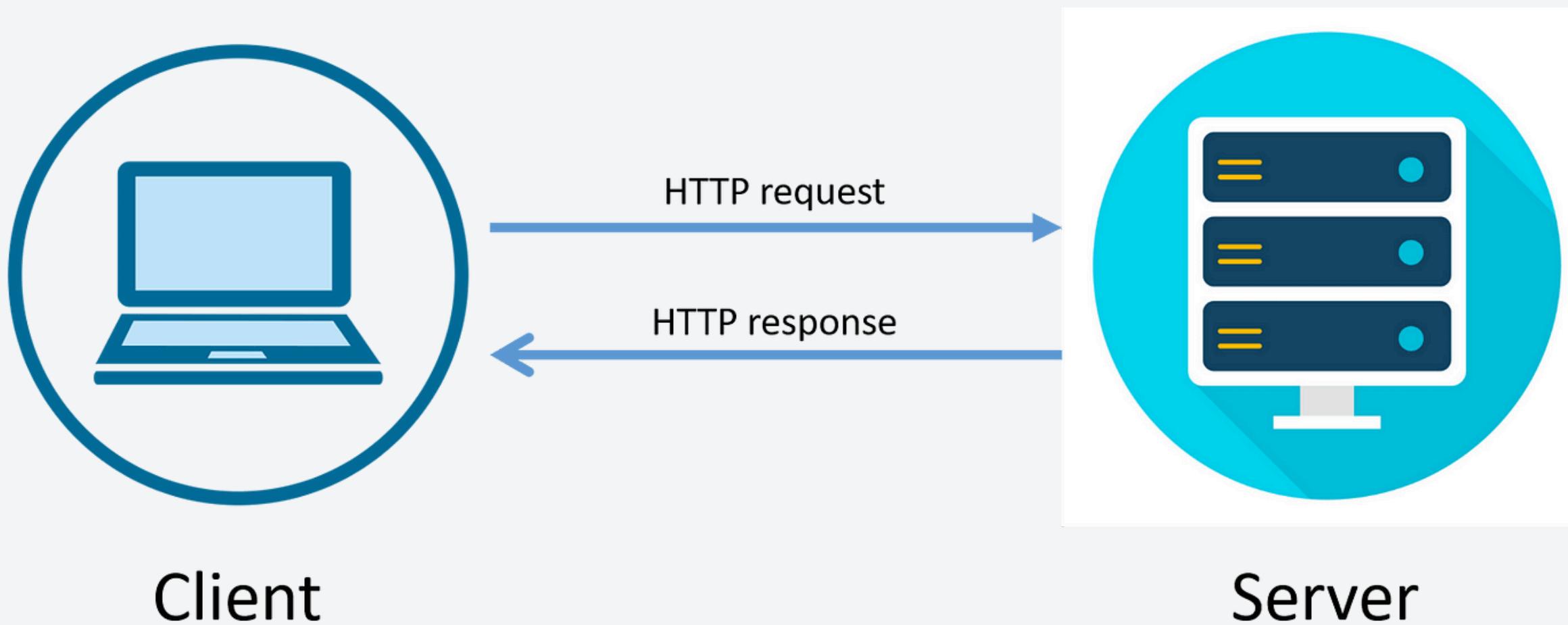
6. DNS (Domain Name System):

- Translates human-readable domain names (e.g., www.google.com) into IP addresses (e.g., 8.8.8.8) that computers use to locate servers.
- Example: Typing “google.com” in your browser triggers a DNS query to find Google’s server IP.

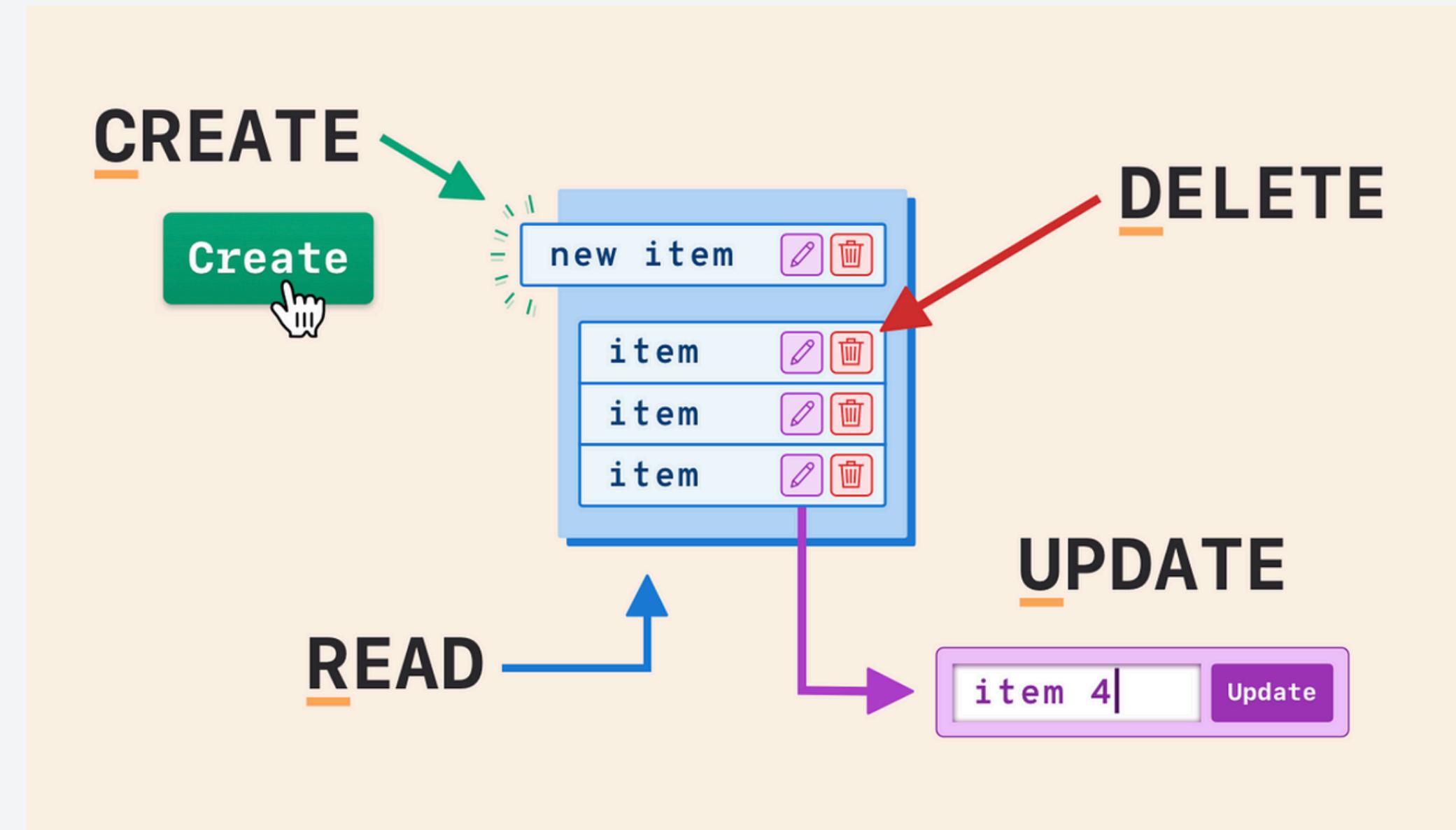


Request-Response Model and How It Works

- Client sends a request (e.g., GET, POST) to the server.
- Server processes the request and sends back a response.
- Example: Loading a webpage or submitting a form.



CRUD Operations



Databases and Database Servers

- Databases store structured data.
- Examples: MySQL, PostgreSQL, MongoDB.
- Database servers manage access to the data for applications.

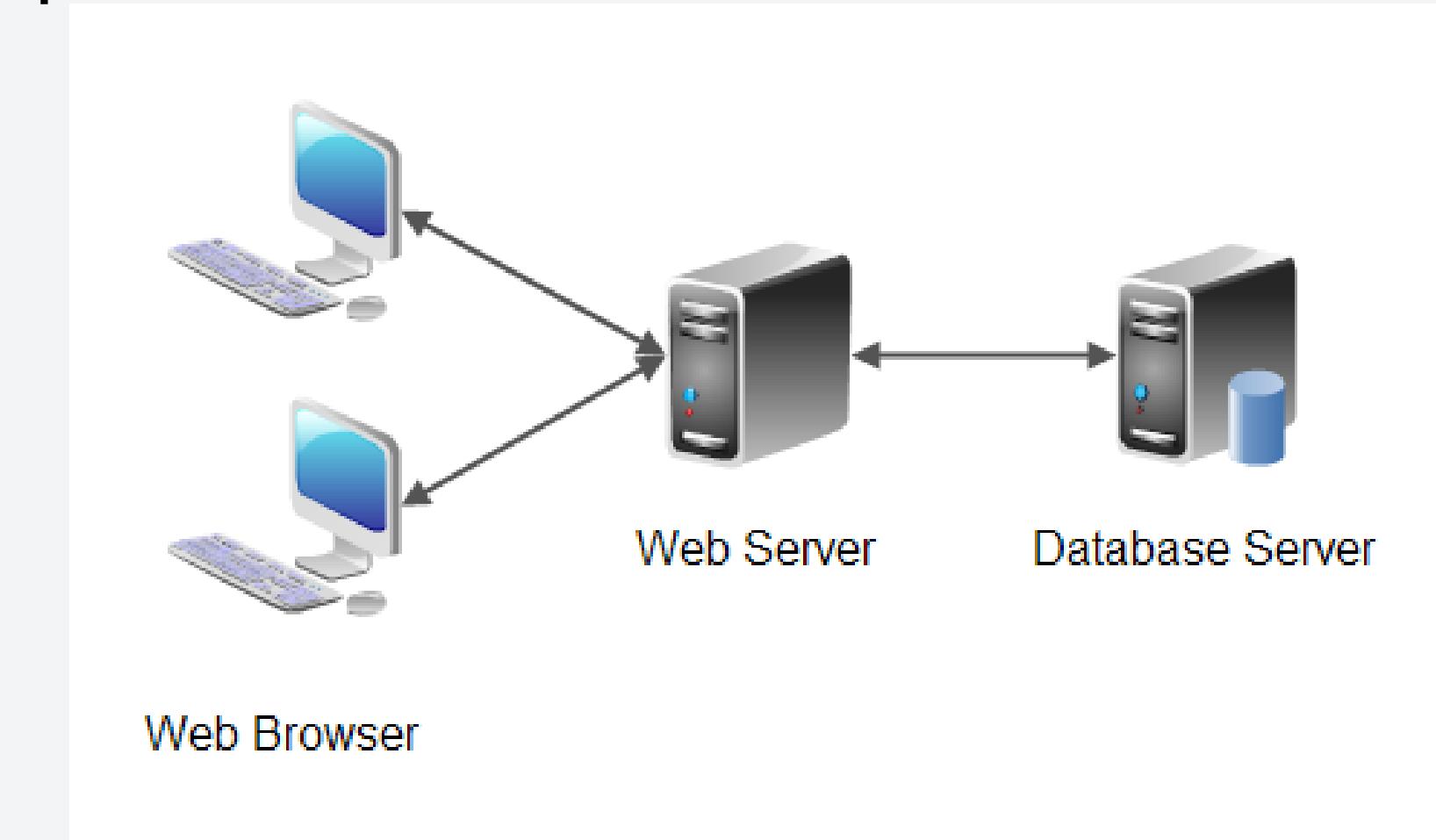


Web vs Database Servers



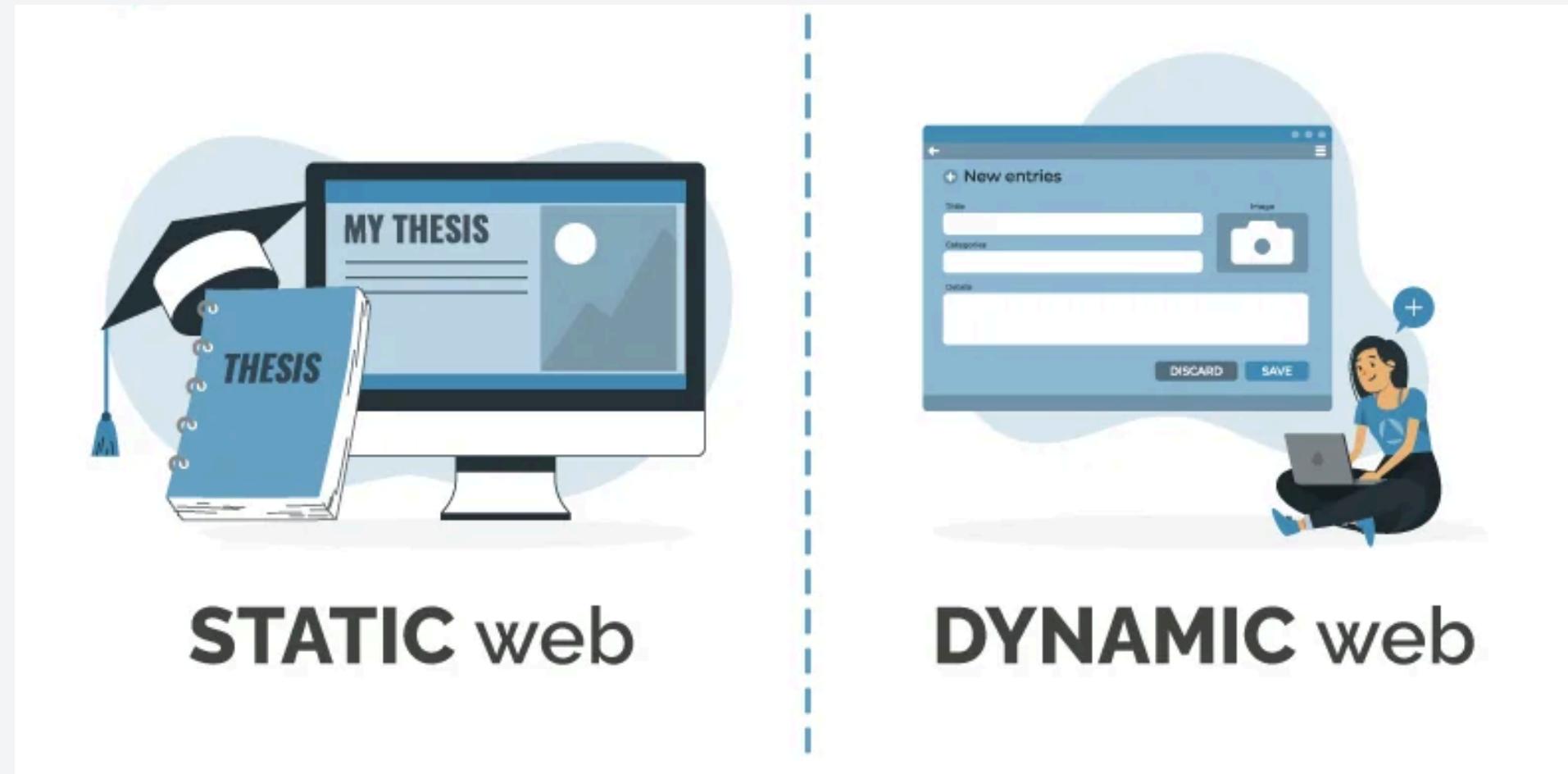
Web vs Database Servers

- A web server primarily stores and delivers website content like HTML pages and images to users through HTTP requests,
- while a database server focuses on storing, organizing, and managing large amounts of data and provides access to that data through queries from applications or other servers



Static vs. Dynamic Websites

- Static: Fixed content, no server-side processing (e.g., basic HTML sites).
- Dynamic: Content generated on-the-fly based on user interactions.



UI/UX in Web Development

- UI (User Interface): Design and layout of the application.
- UX (User Experience): Ease of use and overall user satisfaction.

