Backend for Dummies

What is Backend Development?

The backend is like the **kitchen of a restaurant**. When you order food (make a request), the chef (backend) prepares it, and the waiter (frontend) brings it to your table. The backend does the heavy lifting behind the scenes—storing data, handling logic, and ensuring everything works smoothly.

Why Do We Use Node.js?

Node.js is a tool that helps us build the backend. It's like hiring a fast, multitasking chef who can cook many meals at once without getting tired.

- **Fast:** Node.js is very fast because it uses JavaScript, the same language that runs in the browser.
- **Real-life Example:** Imagine an app like Jumia. Every time you search for products or place an order, Node.js handles the request quickly and efficiently.

What is a Server?

A **server** is like a **receptionist** in an office. It listens to requests (e.g., asking for directions or submitting a form) and sends the correct response.

- **Real-life Example:** When you open YouTube, the server sends the videos and data you see.

What is a Database?

A **database** is like a **giant notebook** where all the app's data is stored. For example:

- Instagram uses a database to store your photos, comments, and likes.
- WhatsApp uses a database to keep track of your chats and contacts.

In this class, we'll use **MongoDB**, a database that stores information like a digital notebook using JSON (like a list of objects).

What are APIs?

An **API** (Application Programming Interface) is like a **menu in a restaurant**. It tells you what you can order (e.g., GET, POST, PUT, DELETE) and gives you the food (response) you asked for.

- **GET:** Ask for information (e.g., “Show me all the products on Jumia”).
- **POST:** Send new data (e.g., “Add a new product to the store”).
- **PUT:** Update data (e.g., “Edit the price of a product”).
- **DELETE:** Remove data (e.g., “Delete a product”).

Real-life Use Case: Jollof Rice Delivery App

Let’s say you’re building a **Jollof Rice Delivery App**. Here’s how backend concepts apply:

1. **Node.js:** Ensures your app is fast and can handle many users ordering jollof rice at the same time.
2. **Server:** Receives requests like “Show me all restaurants near me” and sends the list.
3. **Database:** Stores details of restaurants, their menus, and customer orders.
4. **API:** Allows your app to talk to the server. For example:
    - GET: “Show me all available jollof rice recipes.”
    - POST: “Place an order for spicy jollof rice with chicken.”

In simple terms, the backend makes sure your app works the way it should, even when thousands of people are using it at the same time.