

# Web and Mobile Application Development

Khaula Molapo

20240001

Week 2

## 1.

Building a back-end is an important part of web and mobile application development. For example, when creating a chat application using Node.js, the back-end is responsible for handling messages, users, and data storage. Node.js is used to create a server that listens for requests from users, such as sending or receiving messages. When a user sends a message, the server processes the request, saves the message to a database, and then sends it to the correct user. The back-end also manages user authentication, ensuring that only registered users can access the chat. Using Node.js allows the server to handle many users at the same time efficiently. This makes the chat app fast, reliable, and suitable for real-time communication. A well-built back-end ensures smooth interaction between users and keeps the application secure.

## 2.

Django is a server-side web framework written in Python. Its main role is to help developers build secure and organized web applications quickly. Django handles tasks such as database connections, user authentication, and URL routing. It follows a structured approach called the Model-View-Template pattern, which keeps code clean and easy to manage. Django is widely used because it includes many built-in features that reduce development time. It is suitable for large and small projects, including web and mobile back-end systems. Django also focuses strongly on security, helping protect applications from common attacks.

## 3.

Creating a simple API endpoint in Node.js helps understand how servers handle requests. An API endpoint receives a request from a client and sends back a response. For example, a basic endpoint can return a message such as 'Hello User' when accessed. This is done using Express, a Node.js framework that simplifies server creation. This practice showed how requests and responses work between the front-end and back-end. It also helped me understand routing, server setup, and basic debugging. Building API endpoints is important because they allow applications to share data and communicate effectively.

## 4.

A server-side application structure diagram shows how different parts of the system work together. The diagram usually includes the client, API, server logic, and database. The client sends requests to the API, which processes them using server-side logic. The server then

interacts with the database to store or retrieve data. This structure helps developers understand data flow and system design clearly. Using a diagram makes it easier to plan development and explain the system to others. It also helps ensure that the application is well-organized and easy to maintain.

