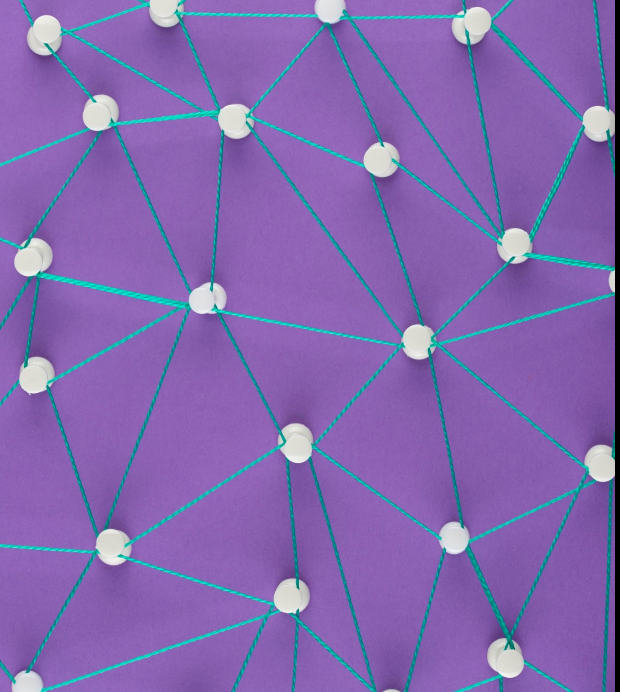
Designing a Chat Room in Operating Systems.

Saadallah Itani
Jad Ghamloush



Project Overview

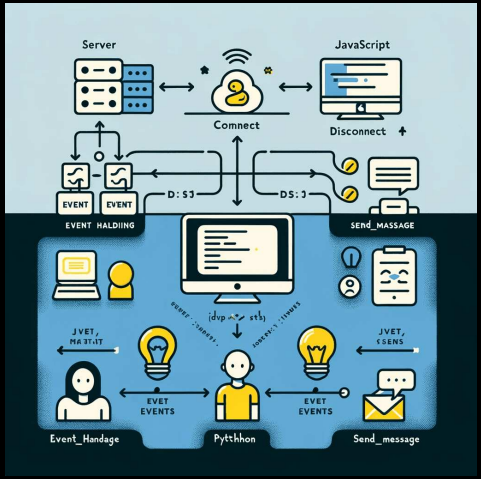
- Objective: To create a real-time communication platform
- Technologies used: Python, Flask, SocketIO, JavaScript, HTML/CSSKey
- Features: Real-time messaging, automatic user name generation

System Architecture

- Client-Server Model: Explains how the web server interacts with client browsers.
- Flask as a web framework: Manages incoming web requests and serves web pages.
- SocketIO: Handles real-time bi-directional communication between web clients and the server.

SocketIO Implementation

- Event Handling: connect, disconnect, send_message.
- Server-side: Python functions triggered by client events.
- Client-side: JavaScript listening and emitting events for real-time updates.



User Interface Design

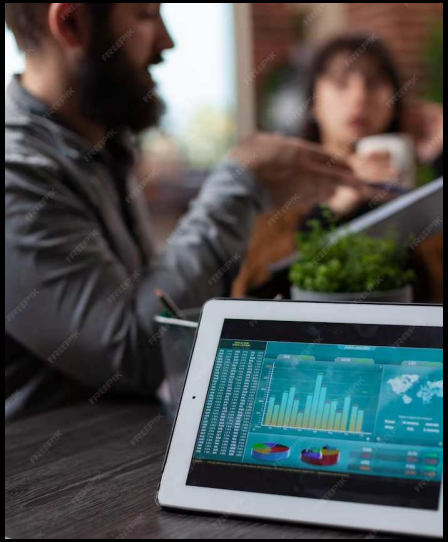
- Simple Chat Interface: Features a clean, user-friendly layout enabling easy reading and writing of messages.
- Dynamic User List: Automatically updates to show who's online, enhancing the sense of community.
- Responsive Design: Ensures full functionality across various devices and screen sizes for accessibility.

Scalability & Future Enhancements

Scalability: Implementing on a distributed server architecture.

Database: Integrate a scalable database for storing chat history.

User Experience Improvements: Develop features for private channels and direct messaging.



Thanks!