

WebSocket Chat (Client + Server) - Documentation

Description

This project is a simple WebSocket-based chat application using:

- Python (WebSocket server) for processing messages.
- React (Next.js client) for sending and displaying messages in real time.

The client allows connection to a local WebSocket server or a built-in mock server. Messages sent by the client are received by the server, which responds by capitalizing the first letter.

Python WebSocket Server

```
import asyncio
import websockets

async def handle_connection(websocket):
    async for message in websocket:
        if message:
            response = message[0].upper() + message[1:]
        else:
            response = "Empty message"
        await websocket.send(response)

async def main():
    async with websockets.serve(handle_connection, "localhost", 8765):
        print(" Server running at ws://localhost:8765")
        await asyncio.Future()  # run forever

asyncio.run(main())
```

React Client Summary

The React client (in page.tsx):

- Connects to the WebSocket server or uses a mock.
- Sends messages using a form.
- Displays message history in real time.

Workflow:

1. Connect to server.
2. Send message.
3. Receive response with capitalized first letter.
4. Display it in chat UI.

Message Flow

Real Server:

1. Client connects to `ws://localhost:8765`
2. Sends message
3. Server capitalizes first letter and returns it
4. Client displays response

Mock Server:

1. Same UI, browser-only logic
2. No Python server needed