WebSocket Server Documentation

Overview

This WebSocket server is designed to provide real-time notifications to clients based on the topics they have subscribed to. The server supports two topics: **TIME** and **CPU**. Clients can connect to the server, subscribe to one or more topics, and receive periodic updates for the subscribed topics.

Server Details

- **Port**: `5000`
- **Address**: `ws://localhost:5000/`
- **Purpose**: To send real-time updates to clients based on their subscriptions. The server broadcasts messages every 5 seconds for the **TIME** topic and every 5 seconds for the **CPU** topic.

How It Works

1. **Client Connection**: Clients connect to the server using WebSocket.
2. **Subscription**: After connecting, clients send a subscription message to the server to specify which topics they are interested in.
3. **Notification**: The server periodically sends messages to all connected clients who have subscribed to a particular topic.

Message Payloads

1. Server-to-Client Message Payload

- **Description**: Messages sent from the server to the client based on the subscribed topics.
- **Format**: Plain text.
- **Examples**:
    - **TIME Topic**: `"TIME: 2024-09-04T14:35:21.345Z"`
        - Contains the current ISO 8601 formatted UTC datetime.
    - **CPU Topic**: `"CPU: 23.45%"`
        - Represents the current CPU usage percentage.

2. Client-to-Server Subscription Message Payload

- **Description**: Messages sent from the client to the server to subscribe to a topic.
- **Format**: `SUBSCRIBE: [TOPIC]`
- **Examples**:
  - `"SUBSCRIBE: TIME"`: Subscribes the client to the **TIME** topic.
  - `"SUBSCRIBE: CPU"`: Subscribes the client to the **CPU** topic.

Supported Topics

1. **TIME**
   - **Description**: Provides the current UTC datetime in ISO 8601 format.
   - **Frequency**: Every 5 seconds.
   - **Example Message**: `"TIME: 2024-09-04T14:35:21.345Z"`
2. **CPU**
   - **Description**: Provides the current CPU usage percentage of the server.
   - **Frequency**: Every 5 seconds.
   - **Example Message**: `"CPU: 23.45%"`

Usage

1. **Connect** to the server using a WebSocket client at `ws://localhost:5000/`.
2. **Send a subscription message** using the format `SUBSCRIBE: [TOPIC]` to receive updates for the desired topic.
3. **Receive updates** from the server at the specified frequency for the topics you have subscribed to.

Example Usage

1. **Connect**: `ws = new WebSocket("ws://localhost:5000/");`
2. **Subscribe**: `ws.send("SUBSCRIBE: TIME");`
3. **Receive**: Listen for incoming messages: `ws.onmessage = (event) => console.log(event.data);`

By following these steps, clients can efficiently receive real-time notifications for their desired topics.