Kuvaka Tech - Real-Time Chat Application Documentation

1. Overview

This is a real-time chat application built using Node.js, Express, and Socket.IO. It allows users to join a chatroom by entering a username and exchange messages in real time. The server automatically serves both client and server files, and there's no need to run the client separately.

2. System Requirements

The following software is required to run this application:
- **Node.js**: v14 or higher
- **NPM** (Node Package Manager)
Ensure that you have Node.js installed on your machine to run the application.

3. Setup Instructions

3.1 Installation

Follow these steps to install and set up the application:
1. Clone the repository:
   ```
   git clone <repository-url>
   ```
2. Navigate to the server directory:
   ```
   cd <path-to-your-repo>/server
   ```
3. Install required dependencies:
   ```
   npm install
   ```

3.2 Start the Application

To start the application, run the following command:
```
node src/server.js

```
```

Once the server is started, you can access the chat application in your web browser.

4. Application Structure

The directory structure of the application is as follows:

```
/client        # Contains client-side files (HTML, CSS, JS)
 /public       # Static files served by Express
 /views        # EJS templates
/server        # Server-side files
 /src          # Contains server logic and socket events
 server.js     # Entry point for the server
 socket.js     # Custom socket event handler
```


5. Features

The application includes the following features:
- Real-time messaging using Socket.IO
- User authentication via a unique username
- Automatic client and server handling, no need for a separate client
- Messages are broadcast to all connected clients except the sender
- Simple and clean UI with the ability to send and receive messages

6. API & Socket Events

The following socket events are used in the application:
- **'set username'**: This event is emitted when a user sets their username.
- **'chat message'**: This event is emitted to send messages between users in real time.

7. Common Issues

Here are some common issues you might encounter while using this application:
- **Issue**: The app is not starting
  Solution: Ensure Node.js and npm are installed, and that all dependencies have been installed using `npm install`.
- **Issue**: Messages not displaying in real time
  Solution: Check the console for any socket connection errors. Ensure the client is connected to the server.

8. License

This project is licensed under the MIT License. See the LICENSE file for more details.

Made By AyuSh