Real-Time Comments System with User Authentication

This project is a real-time comments system that allows users to log in with a username, post comments, and view new comments as they are added, all in real-time. The application is built using Next.js for the front end and Node.js for the back end, with comments stored in a MySQL database.

Table of Contents

Features

- User authentication via a simple username (no password).
- Real-time comments posting and viewing using Socket.IO.
- Responsive and mobile-friendly UI built with Material UI (MUI).
- Comments stored persistently in a MySQL database.

Technologies Used

- **Frontend**: Next.js 15 (App router), React 19, Material UI (MUI), TypeScript, Axios
- **Backend**: Node.js, Express, MySQL (using `mysql2`), Socket.IO, TypeScript
- **Database**: MySQL
- **Deployment**: [Vercel] (in progress)

Getting Started

To get a local copy of this project up and running, follow these steps:

Prerequisites

- Node.js (version ^20) "packageManager":
- npm or Yarn
- MySQL server running

Package Manager

- yarn@4.5.0

Installation

1. Clone the repository:

   ```
   git clone https://github.com/Kunwar-Pratap/real-time-comments-system.git
   ```

2. Navigate to the project directory:

   ```
   cd detrator-isp
   ```

3. Install the necessary dependencies for both the front end and back end:

```
npm install
```

or

```
yarn install  //recommended
```

4. Create a .env file in the backend or root directory and add your database credentials:

```
DB_HOST=localhost
DB_USER=your_db_user
DB_PASSWORD=your_db_password
DB_NAME=your_db_name
PORT=5000
```

5. Start the development server for both front end and back end:

- Frontend

```
npm run dev
```

or

```
yarn run dev //recommended
```

- Backend

```
npx tsx server.ts
```

Usage

- Open your browser and navigate to http://localhost:3000 to view the application.
- Users can log in with a username and post comments.
- Comments will be displayed in real-time as they are posted.

Database Setup

To set up the MySQL database, execute the following SQL commands in your MySQL console:

```
CREATE DATABASE your_db_name;
USE your_db_name;

CREATE TABLE comments (
  id INT AUTO_INCREMENT PRIMARY KEY,
```

```
  username VARCHAR(255),
  comment TEXT,
  timestamp DATETIME DEFAULT CURRENT_TIMESTAMP
);
```

APIs

Backend APIs

- POST /api/login: Accepts a username and returns a session ID.
- GET /api/comments: Fetches the list of comments from the MySQL database.
- POST /api/comments: Post a comment with the associated username and stores it in the MySQL database.

Real-Time Feature

When a new comment is posted, the server broadcasts the comment to all connected clients using Socket.IO, allowing real-time updates without needing to refresh the page.

Contributing

Contributions are welcome! Please follow these steps:

1. Fork the project
2. Create your feature branch (git checkout -b feature/YourFeature)
3. Commit your changes (git commit -m 'Add some feature')
4. Push to the branch (git push origin feature/YourFeature)
5. Open a pull request

License

This project is licensed under the MIT License - see the `LICENSE` file for details.

Acknowledgments

- Thanks to the developers of Socket.IO for their fantastic real-time library.
- Special thanks to the MySQL team for providing a robust database solution.
- Inspiration from various online resources and tutorials.