**Full Name** :

**Position** :

**Date** :

**User Appointment Management System**

**Overview**

Develop a user-friendly application for managing appointments while handling timezone conflicts and ensuring time readability. The system should allow users to create, view, and manage appointments while respecting the preferred timezones of participants.

**Key Features**

* **Authentication**: Simple login using a username with JWT or session-based authentication. Sessions expire after 1 hour.
* **User Management**: Manage user profiles with attributes like name, username, and preferred timezone.
* **Appointment System**: Create and view appointments, invite other users, and manage your schedule.
* **Timezone Handling**: Schedule appointments within working hours (09:00 - 17:00) of all participants and display times according to the logged-in user's preferred timezone.
* **User Interface**: Minimalistic design with a login page, appointment list page, and appointment creation form.

**Instructions**

* **Code Quality**: Ensure your code is clean, modular, and easy to run.
* **Version Control**: Use a GitHub/GitLab repository to store your work.
* **Documentation**: Include a README.md with setup and execution instructions, video, and photo for documentation.

**Project Implementation**

**1. Authentication**

* **Login**: Implement simple login using only the username.
* **Authentication Method**: Use JWT or session-based authentication.
* **Session Expiry**: Sessions should expire after 1 hour.

**2. User Management**

* **User Model**: Implement a User model with the following attributes:
  + id (UUID or auto-increment)
  + name (String)
  + username (String, unique)
  + preferred\_timezone (e.g., Asia/Jakarta, Pacific/Auckland)
* **Database**: Use a relational database (PostgreSQL/MySQL) or NoSQL (MongoDB).
* **API Endpoint**: Provide an API endpoint to fetch user data.

**3. Appointment System**

* **Appointment Model**: Implement an Appointment model with:
  + id (UUID or auto-increment)
  + title (String)
  + creator\_id (User relationship)
  + start (Datetime)
  + end (Datetime)
* **User Actions**: Users should be able to:
  + Create appointments and invite other users.
  + View a list of their upcoming appointments.

**4. Timezone Handling**

* **Scheduling**: Ensure the time falls within working hours (08:00 - 17:00) of all participants.
* **Display**: Adjust times according to the logged-in user's preferred timezone.

**5. User Interface**

* **Login Page**: Minimalistic design for user login.
* **Appointment List Page**: Display a list of upcoming appointments.
* **Appointment Creation Form**: Form for creating new appointments.

**Technical Questions**

Answer the following questions in markdown format inside the repository (answers.md):

1. **Timezone Conflicts**: How would you handle timezone conflicts between participants in an appointment?
2. **Database Optimization**: How can you optimize database queries to efficiently fetch user-specific appointments?
3. **Additional Features**: If this application were to become a real product, what additional features would you implement? Why?
4. **Session Management**: How would you manage user sessions securely while keeping them lightweight (e.g., avoiding large JWT payloads)?