Software Developer - Technical Assignment

Objective

This technical assignment evaluates your proficiency in web development with a focus on:

- Backend development fundamentals
- Database design and API development
- Role-based access control
- Application architecture and code structure
- Docker usage and deployment

Submission Guidelines:

1. Upload your completed projects to a **GitHub repository** and share the link.
2. **Deadline:** EOD of **12/02/2026**
3. **Partial completion** of the assignment will also be accepted.
4. Submission Mail - id Ferdeno A

Task: Role-Based Dashboard System

Description

Develop a web-based application that implements a **role-based dashboard system**.
The system should support user authentication, user management, and role-based permissions.

The goal is to evaluate your approach to backend logic, database design, and overall system implementation.

Technology Stack

You may use **any technology stack** you are comfortable with for the frontend and backend.

Mandatory Requirement:

- The database must be **PostgreSQL**

You are free to design the project structure, architecture, and implementation approach.

User Roles

Your system must support the following roles:

SuperAdmin

- Can create Admins and Users
- Can edit Admins and Users
- Can delete any user
- Full system access

Admin

- Can create users
- Can edit users
- Cannot delete users created by SuperAdmin or other Admins

User

- Can log in
- Can view the dashboard only
- No create/edit/delete permissions

Core Features

Your application must include:

- Login system (basic authentication is sufficient)
- Dashboard page after login
- User listing page
- Create user functionality
- Edit user functionality
- Delete user functionality with role-based restrictions

Important Requirements

- All role validation and permission checks must be implemented in the **backend**.
- Do not rely only on frontend restrictions.
- Proper error messages must be returned when unauthorized actions are attempted.

Example:

If an Admin tries to delete a user created by SuperAdmin →
Return: **"You do not have permission to perform this action."**

Docker Requirement

Use **Docker** to containerize and run the application locally.
You should handle configuration and setup decisions.

Bonus (Optional)

These are not mandatory but will strengthen your evaluation:

- Use environment variables for configuration
- Clean and structured UI
- Proper project documentation
- Live demo using **ngrok** to expose the running application and share public URL