TR7 Admission Assignment

This assignment will be a part of your admission process to the TR7 Research and Development Team. Please read the following instructions carefully and deliver your work in at most 5 days in the requested format.

Introduction
In this assignment, you are tasked with developing a web application using Node.js and the Express framework. The goal is to create a simple web application that features user authentication and role-based access control.

Backend
The application should include three primary API endpoints:

1. "/login": This endpoint will handle user login, validating the provided username and password against a stored database. You might use JWT, cookie sessions or any other clear text indicator to manage user sessions.
2. "/admin": This endpoint should only be accessible to authenticated users with an "admin" role. It should return a simple welcome message to the request.
3. "/user": This endpoint should be accessible to both "admin" and "user" roles, but not to unauthenticated users. It should return a simple welcome message to the request.

Database
For this assignment, you are required to store user information, including roles and credentials, in a JSON file within your project directory. This means that instead of using external databases like MySQL, PostgreSQL, MongoDB, etc., you will define a JSON file that acts as a simple, local database for your application. This file will contain all necessary user data and will be used to validate login credentials and manage access to different API endpoints.

```
[
 {
   "username": "adminUser",
   "password": "adminPass",
   "role": "admin"
 },
 {
   "username": "regularUser",
   "password": "userPass",
   "role": "user"
 }
]
```

Frontend

Create a basic frontend setup using Vue.js 3 as a part of this assignment. This will integrate with the backend APIs you've set up using Express and Node.js. Below are the pages that should exist in your Vue.js 3 project.

1. Login Page: The Login Page allows users to enter their credentials and log in to the application. It should send a request to the “login” API to perform login functionality. Upon successful login, users will be redirected to either the Admin Page or the User Page based on their role.
2. Admin Page: The Admin Page is accessible to users with the "admin" role. It only shows a message specific to admin users. It should fetch data retrieved from the “admin” API.
3. User Page: The User Page is accessible to users with the "user" and "admin" role. It greets all users with a message. It should fetch data retrieved from the “user” API.

You are not required to write CSS for a better UI, but a better UI is always a bonus. Your priority is to show that you can integrate your backend design to a functional UI using Vue.js 3. An example is given below:

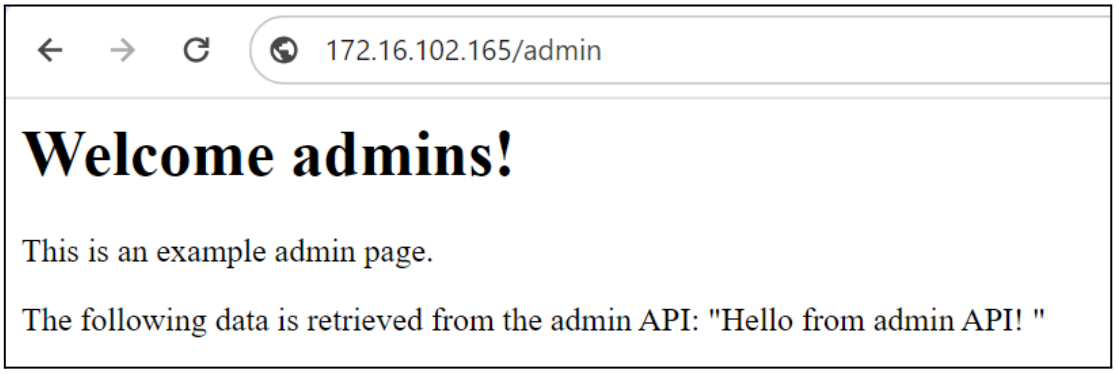

Figure 1: Example admin page.

Figure 2: Example user page.

Figure 3: Example login page.

Logistics

The best way to deliver your assignment is as a GitHub repository link. If you do not know how to do this, we strongly recommend that you learn how to do it and deliver your assignment as such. However, if you are unable to do so, you can also deliver your assignment as a zip file.

We need to be able to run your application by running the backend and frontend separately. Please include a README.md file to specify how to run your application. Both should run on localhost and on a port defined within the application source code. If your application cannot be run as described below, please include this information in your README.md file in a clear format.

If you are using Generative AI or any other external help, you need to cite it in your README.md file. Failure to cite your resources will result in your assignment being invalid, and your application to TR7 will be rejected.

Please note that this assignment is proprietary to TR7. Do not share or disclose it to anyone.

Good luck!