Car Management System

This Car Management System allows users to register, log in, and manage a catalog of cars with details like images, titles, descriptions, and tags. The system is divided into two primary components: frontend requirements (user interface) and backend APIs to handle car management operations.

Required Functionalities

1. **User Authentication (Signup/Login):** Users can create an account and log in to access and manage their cars.
2. **Add Car Details:** Users can add a car with up to 10 images, including a title, description, and tags.
3. **List User's Cars:** Users can view a list of all their cars.
4. **Global Search:** Users can search through all cars by a keyword, which filters the cars based on a match in the title, description, or tags.
5. **View Car Details:** Users can click on a car to view its detailed information.
6. **Edit Car Details:** Users can update a car's title, description, tags, or image.
7. **Delete Car:** Users can delete a car from their list.

Frontend Requirements

The frontend is structured to allow seamless navigation and interaction with the Car Management System. Each page serves a specific function based on user actions:

1. **Sign Up / Login Page:** This page allows new users to sign up and existing users to log in to access their cars. Once logged in, users are directed to the Product List page, where they can view and manage their cars.
2. **Product List Page (Dashboard):** This page displays a list of all cars created by the logged-in user. It includes a search bar where users can enter keywords to filter cars based on title, description, or tags.
3. **Modals:** provides a modal for creating a new car entry. Users can upload up to 10 images, set a title, provide a description, and add tags to categorize the car, editing the products and deleting
4. **Product Detail Page:** This page displays detailed information about a selected car, including its images, title, description, and tags.

Backend API Documentation

The backend exposes RESTful APIs to handle user authentication, car management, and search functionality.

All routes are prefixed with /api/

https://documenter.getpostman.com/view/29490334/2sAYBPmECe

[Note: Ids used in postman may not be functioning as they subject to change during development]