project

CAR RENTAL SYSTEM

proposal

Submitted By:

script squad

Submitted To:

depi

script squad

Project Overview:

The goal of this project is to develop a comprehensive online car rental platform catering to both customers and administrators.
The system will enable customers to search, filter, and rent cars based on their preferences,
while the admin side will offer a full-fledged management dashboard.
The project will be built using the MERN stack, ensuring scalability and efficiency.
We will also integrate Figma for designing an intuitive user interface for both client-side and admin-side operations.

Development Phases

Develop a static prototype using Vanilla JavaScript.

Implement the React frontend with Redux for state management.

Integrate the backend using Node.js and MongoDB

Client-Side Features:

visually appealing user interface including

functional features like

car search & filtering:

users can search for cars based on filters like

price, brand, luggage space and color.

car selection & marking:

customers can view detailed car descriptions,

availability, and pricing information.

reservation & rental process:

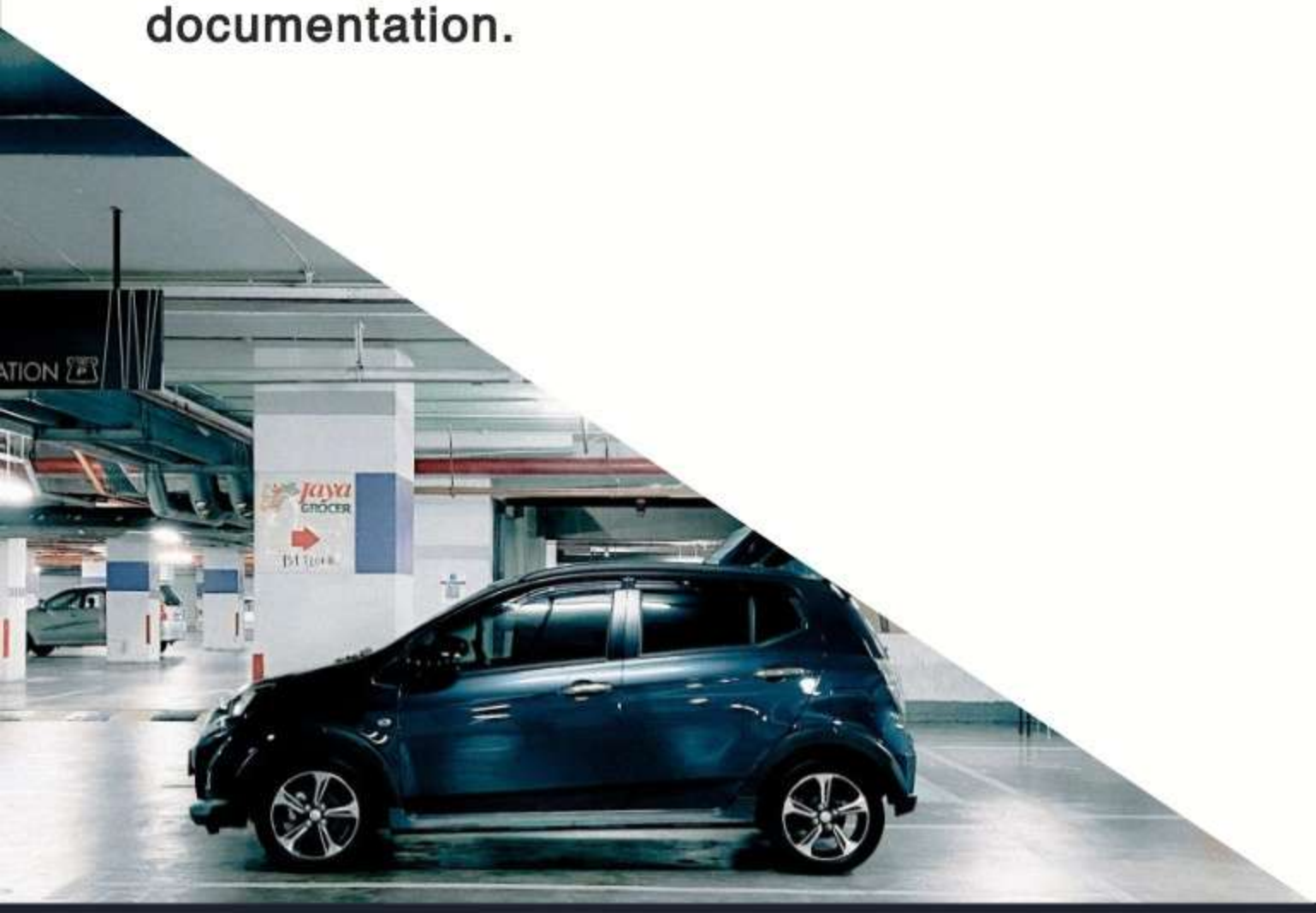
users can book rental cars for a specific period .

customer account:

users can create and manage their accounts.

view rental history, notifications, and

documentation.



Admin-Side Features:
(Management Dashboard):

Car Management:

Add, edit, or delete car listings.

Track car maintenance schedules (e.g., oil changes, inspections).

Rental Contracts:

Manage rental contracts, including customer details, car information, and rental period. Track ongoing rentals and access full rental history (V2).

Customer Data Management:

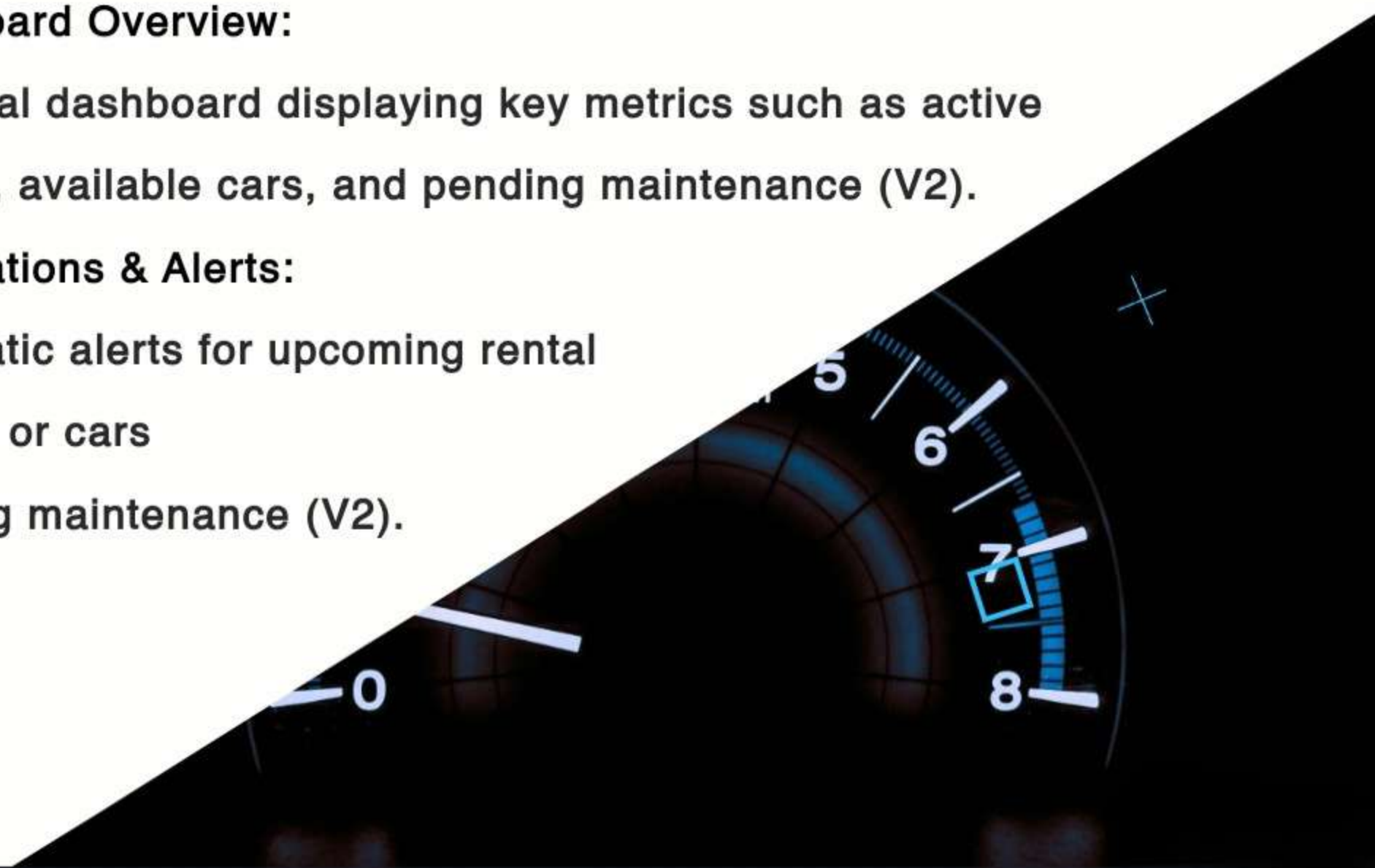
Admins can manage customer profiles (add, edit, or remove).

Dashboard Overview:

A central dashboard displaying key metrics such as active rentals, available cars, and pending maintenance (V2).

Notifications & Alerts:

Automatic alerts for upcoming rental returns or cars needing maintenance (V2).



Redux Integration:

We will integrate Redux to manage the application state across both client and admin sides. Redux will handle the following:

1. **Global State Management:**

Centralized state for user data, car availability, and bookings.

2. **Real-Time Updates:** Ensuring that changes in car availability, user authentication, and bookings are reflected across the application in real-time. For example, when a user books a car, Redux will instantly update the availability for all users.

3. **Data Persistence:** Persistent state for user sessions and rentals, even on page reloads. If a user reloads the page during booking, their selected car and details will remain.

4. **Middleware:** Using Redux Thunk or Redux Saga for asynchronous actions like API calls for fetching car data or processing bookings.

Technologies & Tools:

JS

HTML

CSS

Mongoose {

Figma

AXIOS

node JS

Redux

React JS

mongoDB.

EXPRESS Js

Functional Requirements:

User Authentication:
Secure login for customers and admins.
Real-Time Updates:
Car availability and rental status updates dynamically for both customers and admins, managed by Redux.
Data Validation & Security:
Ensure input validation for car details and customer information to prevent errors.
State Management:
Using Redux to handle the application's state globally across all components, ensuring consistency.

Non-Functional Requirements:

Performance:
The platform will handle concurrent users efficiently without performance degradation.
Maintainability:
Modular code structure for easy maintenance and future feature expansion.
Scalability:
The application is designed to accommodate an expanding fleet of cars and users.
Usability:
A user-friendly system requiring minimal learning for both customers and administrators.

Expected Timeline

submit full Vanilla project responsive and including documentations and wire frames

submit react project (part 2) integrated with backend (redux , authentication , api)

final touches , testing and improving before submitting project

script squad

expected outcome:

A fully functional car rental system with user and admin access.

Secured handling of user data and rental transactions.

An engaging user experience with the car quiz and gallery and booking system

a functional dashboard that provide functional features

conclusion:

This project will provide valuable real-world experience in full-stack web development, using the MERN stack. Where The system's modular design ensures scalability and significant user experience