A white letter d on a black background

Description automatically generated

Driven Auto Rental

A person standing next to a car

Description automatically generated

Lewis Innes-Miller & Miral Patel

FSD-08 – Web Development II – Fall 2023

Car Rental Booking Application

Proposal:

A booking wizard platform that enables users to register, book, and interact with their auto rental bookings. These user interactions include rating models previously booked, leaving reviews, and updating or deleting current bookings. The admins on the platform can perform CRUD operations on these entities, as well as access calendar-based views of bookings and related statistics.

Description:

Using React in the front-end on top of a nodeJS backend, we seek to build a versatile auto rental application. Users will be able to browse the car models available and create accounts to book them. Users cannot make a booking as a guest. When booking, users are able to use reactive calendar components to select their intended rental period (figure 4). Once authenticated, users can see past and current bookings, as well as update or cancel any upcoming rentals on their profile page. After a rental period is over, the user will be invited to leave a review and a rating of the model and the service. In most of these cases, flash messages inform and guide the user.

Admins on the platform have an even wider array of features available to them once they are authenticated. Not only can admins introduce new car models into the platform, and this includes photos added via file upload, but they can also perform CRUD operations on virtually every entity in the application, users included. A further perk is the ability to view the upcoming weeks and rentals in a calendar, as depicted in the mock-ups below. Lastly, the admins are also able to view charts which communicate valuable patterns about the business.

Added Challenge:

If conditions are permitting, a further user feature will be implemented: loyalty accounts. These accounts would pay a monthly subscription for access to special rates, deals, and the ability to accumulate reward points. Loyalty users would have their personal profile and views updated accordingly, e.g. they see discounted rates when browsing rental vehicles.

Main Technologies to be Deployed A blue and black circle with a circle in the middle

Description automatically generated

* React JavaScript Library (frontend)
* NodeJS JavaScript runtime with Express (backend)
* Sequelize ORM
* MySQL persistent storage database
* Bootstrap CSS
* Recharts data visualization library

# Additional Technologies that could be Deployed

* Sass for CSS
* JSON Web Tokens (JWT) for secure user authentication
* Azure VM hosting with cloud implementation of MySQL
* react-calendar package
* React Scheduler package

# Special Features

* Review and rating system for rentals
* Calendar and Chart views on the admin interface (figure 6)

Website URLs:  
/  (index)  
/register   
/login     
/profile  
/models    
/models/12    
/confirm

CRUD URLs (api calls):  
(User) GET, PUT, PATCH, POST, DELETE /api/user   
(CarModel) GET, PUT, PATCH, POST, DELETE /api/carmodel   
(CarBooking) GET, PUT, PATCH, POST, DELETE /api/carbooking    
(Review) GET, PUT, POST, PATCH, DELETE /api/review

ADMIN interface:  
/admin/users   
/admin/bookings   
/admin/models   
/admin/reviews

A screenshot of a computer

Description automatically generated

Figure 1 Driven Auto Rental Database Structure

A screenshot of a diagram

Description automatically generated

Figure 2-Driven Use Case Diagram

A screenshot of a car sales page

Description automatically generated

Figure 3-Visualization of the application's home page

A screenshot of a car sales page

Description automatically generated

Figure 4 - User view of a car model page

A screenshot of a computer

Description automatically generated

Figure 5-The Admin Page's List view of all bookings

A screenshot of a calendar

Description automatically generated

Figure 6- Visualization of the Admin's Calendar and Chart View

# Questions:

1. Are admin operations a part of user stories?
2. Should users be able to delete reviews?
3. File upload issue with cloud hosting?
4. What would implementation of a bilingual website entail?
5. Payment processing? How is this tested?
6. Why is S3 preferable to Azure Blob Storage?