RentHubDB

Project Summary

RentHubDB provides a seamless solution for car rental companies to oversee their vehicle inventory, manage reservations, and analyze customer preferences. There are several features that enable businesses to make data-driven decisions when monitoring vehicle usage, identifying rental trends, and adjusting inventory across location as well as pricing based on demand. For example, an interactive heat map highlights popular rental locations, which is useful information for the business to store cars in their garages and optimize pricing.

In addition to rental car administrators, customers can also benefit from using the RentHubDB platform. For customers, this application offers a smooth and easy experience when searching for and booking rental cars. Whether they're looking for a specific model, price range, or location, they can easily find options that match their needs. The system updates availability in real time, so customers always have up-to-date information. By focusing on small and mid-sized rental businesses that don't have access to advanced tracking tools, RentHubDB fills a major gap in the industry, offering a practical and efficient way to manage both inventory and customer experience.

Description

Our rental car tracking application is a dual-user digital platform for car rental business administrators and customers. Car rental companies can monitor vehicle availability in real-time, track rental history, and distribute vehicles across multiple garage locations. On the other hand, customers can use this system to tailor their car rental experience by choosing cars that match their budget, model, and location preferences. In addition, customers can provide satisfaction ratings and comments, which allows the business to improve their services based on feedback. Our application is targeted towards small and mid-size car rental businesses as they typically do not have advanced tracking and inventory management tools. Moreover, the features and functionality create a comprehensive platform by integrating accurate vehicle tracking, inventory analysis, and customer feedback.

By joining rental history data, which includes rental period, miles, and purpose of trip, with customer satisfaction data, the business can determine rental trends and identify their best-selling models to enhance their marketing strategy. Since our database also stores car and garage data, businesses can replace high-mileage cars, add new models, and distribute cars across garages to optimize availability at popular locations.

Creative Component

The creative component of our car rental system is a heat map that depicts high-demand rental locations based on real-time booking data. This interactive visualization feature can aid businesses to identify peak demand conditions, such as in large cities during holidays, and adjust rental rates to make more profit.

Usefulness

Our rental car tracking system is designed to help car rental businesses efficiently manage their vehicles, track customer reservations, and analyze customer satisfaction. This application is useful because it allows for customers to browse available vehicles, make any

reservations, and also leave feedback on their rental experience. In addition to customers, administrators at the rental car company can track vehicle availability, monitor mileage, and optimize the distribution of cars throughout multiple locations. Because of the customer satisfaction system, the rental car company can identify improvements to their service and enhance customer experience.

There are many similar websites and applications that accomplish a similar goal. However, unlike existing rental car systems used by companies like Budget, Avis, and Enterprise, our system offers a more detailed and flexible way to manage rentals. Specifically, our application is useful for small businesses that may not have access to advanced tracking systems. Since we are integrating garage tracking into our application, we can ensure better distribution of vehicles across locations. By focusing on customer insights and detailed tracking of vehicles, our application offers a unique approach that can be tailored to the needs of small rental car businesses.

Realness

The primary dataset for the vehicle information is a .csv file containing 5,851 rows of data and 15 columns and includes key attributes such as fuel type, rating, renter trips taken, review count, rental price, and vehicle specifications (make, model, type, year). Additionally, the dataset has the preliminary location details of where cars are available, including the city, state, latitude, and longitude of their location data. We obtained this data from Kaggle from two different datasets named Cornell Car Rental Dataset and My Uber Drives, respectively. We will likely expand on this data with generated data so there is more usefulness to the Garages table. Since this dataset will be split into different tables, we will structure the data such that each car is uniquely identified by a Car ID, while the garages storing the vehicles will be linked by Garage ID. The Owner ID allows for a further categorization of the vehicles under specific business owners and grouping to understand a renter's history. By normalizing the data, we can ensure improved organization, avoid any redundancy, and push for more efficiency in our queries when we retrieve vehicle details and their locations.

The rental history dataset consists of 1,156 rows and 7 columns that detail each transaction, including its start and end dates, rental category, pickup and drop-off locations, miles driven, and the purpose of the trip. This dataset will be connected to both the Customers and Car tables using Customer ID and Car ID. Each rental data instance will have a unique Rental ID, which will also be the primary key for the Customer Satisfaction table, which will link customer ratings and reviews to the specific rentals. The Garage ID will allow for the tracking of vehicle distribution across many different locations, making sure that businesses can optimize inventory based on demand. Since we are generating the customer data, we will assign each user a Customer ID, which will be referenced back in the rental history and customer feedback records.

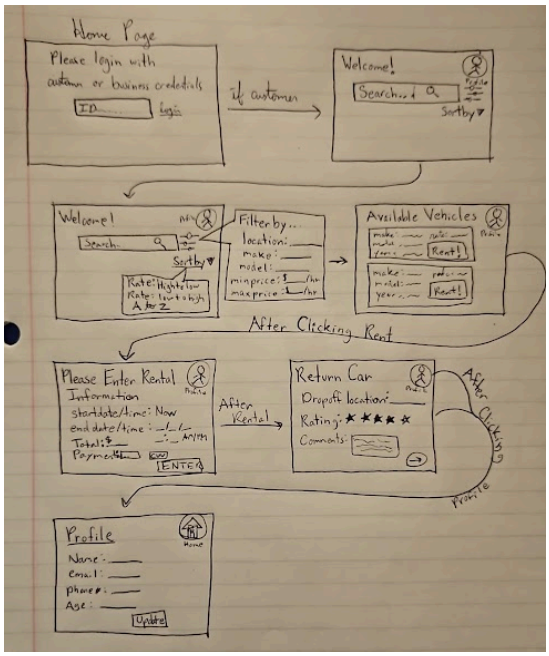
Functionality

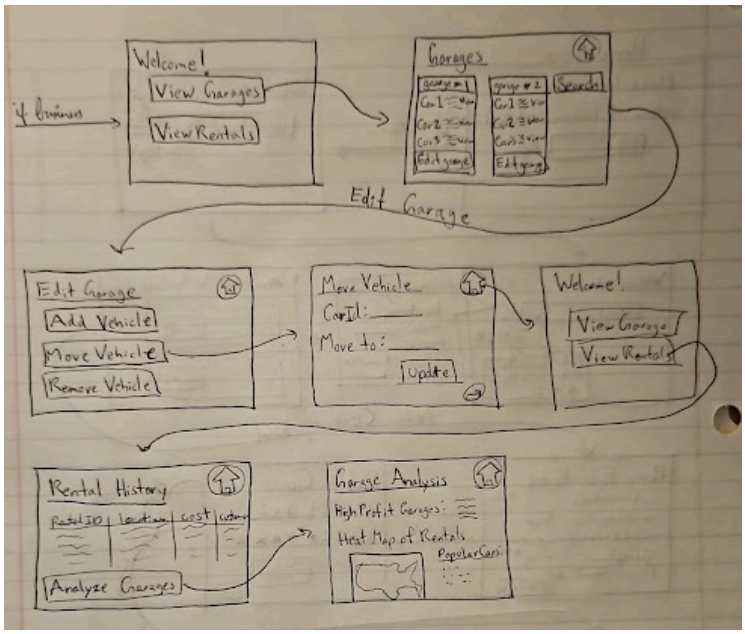
Our application has many functions for both customers and businesses. The home page would feature a login screen where users input either their customer or business login information. As a customer, you will be able to browse available vehicles within a specified city by using the search function. In addition to searching by location, users can find vehicles based on model or hourly rate. After finding a vehicle to their liking, a user can create a rental, which

will temporarily update the vehicle's availability. Once the vehicle is returned, the availability will once again be updated and the rental information will be added to the car's rental history. Users can also leave a 1 to 5 star rating based on their rental experience which will update the customer satisfaction table.

On the other hand, if a business login is recognized, a different set of functionalities will be available. Rental business owners can monitor vehicle demand by searching the rental history and tracking the number of times a vehicle has been rented. Based on demand, businesses can add (or insert) vehicles to their garage. Additionally, businesses can remove (or delete) vehicles that have a high mileage. They can also analyze which vehicles are being rented for different purposes such as business or personal. Another feature is that companies can improve their service or retire vehicles based on customer ratings. Finally, business owners can create new garages or move vehicles between garages based on demand.

Low-Fidelity UI Mockup





Project Work Distribution

We all agree to contribute equally to this project and the creation of this vehicle rental application. Every team member will actively participate in developing the final web application. Sharon will take the lead on UI design of the application, focusing on creating a user-friendly interface and developing features that ensure the website is easy to interact with. Sharon and Rishab will be responsible for managing the database and ensuring proper relationships between the five entities. Divya and Soundarya will focus on comprehensive testing of the application. They will collaborate to conduct tests that ensure data is correctly returned from all queries and that all components of the system function as expected. This includes handling edge cases and debugging any errors to preserve data integrity.