Car Rental Service Agency

[1] Niharika Santhoshini Karri, [2] Sraddha Pedda Gangireddy Gari, [3] Sahithi Reddy Marapadaga

[1] Information Systems, Northeastern University
[2] Information Systems, Northeastern University
[3] Information Systems, Northeastern University

{karri.n , peddagangireddygar.s ,marapadagavenkata.s }@northeastern.edu

Abstract -

The Car Rental Service Agency serves as a versatile and efficient software solution designed to streamline the operations of car rental companies. It is built utilizing JavaFX, JDBC, and adheres to the Model-View-Controller (MVC) architectural pattern, ensuring a flexible and easily manageable codebase**.**

For contemporary and responsive graphical interfaces, the user interface is developed using JavaFX along with SceneBuilder. JavaFX offers a wide array of features, including support for CSS styling, multimedia, significantly enhancing the overall user experience for both customers and admin. Data management within the system is facilitated through JDBC, enabling seamless interaction with the relational database. This facilitates the efficient execution of SQL queries for tasks such as vehicle addition, rental record updates, and customer management.

The Car Rental Service Agency is structured according to the MVC architecture, consisting of three core components: the model, the view, and the controller. The model component manages and stores data concerning vehicles, rentals, and customers, integrating essential business logic for rental calculations and generating reports.

The view component acts as the user interface, enabling customers to search for available cars, make reservations, and view their rental history. Concurrently, admin can access features like car availability status, rental records, and customer details.

Serving as a mediator between the model and the view, the controller component handles user input, processes requests, and updates both the model and view accordingly. This clear separation of responsibilities simplifies maintenance and facilitates the addition of new features without disrupting the overall system functionality. The Car Rental Service Agency offers a comprehensive set of functionalities, encompassing car searches, rental reservations, customer management, rental record updates, and various report generation options. Its design ensures scalability and customization, making it suitable for meeting the specific requirements and preferences of diverse car rental companies.

In a nutshell, the Car Rental Service Agency provides a comprehensive and user-friendly solution for efficiently managing car rental operations. Its utilization of JavaFX**,** JDBC, and the MVC architecture ensures a modular**,** scalable, and easily maintainable system, making it an ideal choice for modern car rental businesses.

Keywords* – *JavaFX, SceneBuilder, JDBC, MVC Architecture, User-Friendly Interface

I. Problem Description

In the fast-paced world of car rental services, the need for an efficient and streamlined management system has become increasingly vital. The existing manual processes for resource management, customer records, and transaction handling are not only time-consuming but also prone to errors. To address these challenges and usher in a new era of digital car rental management, we propose the development of a comprehensive Car Rental Service Agency.
The primary objective of the Car Rental Service Agency is to revolutionize how car rental companies operate by offering an integrated solution that effectively manages vehicles, clients, and transactions. By leveraging the power of JavaFX, Scene Builder, and MySQL, our system aims to eliminate the drawbacks of traditional manual approaches, providing a sophisticated yet user-friendly platform.

Our proposed solution presents an innovative way for car rental firms to enhance their operational efficiency and customer service. With the Car Rental Service Agency, rental companies will be empowered to keep track of available cars, their specifications, and maintenance history. This ensures a well-maintained and up-to-date fleet that meets customer demands. It will store and manage customer information securely, allowing for quick retrieval and analysis. This leads to personalized customer experiences and effective customer relationship management. The system facilitates seamless and accurate rental transactions, including booking, check-in, and check-out. Automating these processes reduces errors and enhances customer satisfaction. It ensures data security and proper access control by implementing user authentication and authorization mechanisms. This prevents unauthorized access and protects sensitive information. It would gain valuable insights into rental trends, demand patterns, and popular vehicle choices through comprehensive data analysis. This enables informed decision-making for optimizing the fleet and enhancing profitability. The system provides a user-friendly interface for both rental company managers and customers. Managers can efficiently oversee operations, while customers can easily browse and select vehicles from the inventory.

By adopting our Car Rental Service Agency system, rental companies can enjoy numerous benefits, includes automating manual processes reduces administrative burden and allows staff to focus on more strategic tasks, efficient resource management and streamlined transactions lead to improved customer experiences and satisfaction. In-depth analytics empower rental companies to make informed decisions, optimizing their fleet composition and overall business strategies. The system's intuitive design and features boost overall productivity, making tasks such as client management and transaction handling more efficient. With accurate insights and enhanced operations, car rental companies can expect better financial performance and a competitive edge in the market.

II. ANALYSIS (RELATED WORK)

The first thing we need when we land in a new place, or we look for, is a car for rent for better and easier transportation. Technology involvement has increased a lot for car rental management. Lately people used to go manually checkout the cars and if they are available then they would rent a car. This situation creates uncertainty in times when we really need a car.

A management system where it handles car rental is a useful option when compared to manual renting. This Management system helps a customer to see the availability of cars and plan, increased accuracy. Such management systems will also be useful for increasing the business profits because they increase the reach among customers.

Certain things that should be taken care of for successful implementation of this system. The data of the car should be managed properly and there should be proper communication between the Car Agency and the professionals handling this app. The other challenge is this system should be able to work on various platforms and devices.

This system will help in user management with the help of distinct roles, managing billings and charges, security for vehicles and data and it will create various formats for data storage. Additional features like oil change, vehicle maintenance and generating of reports can be further implemented.

The use of technology in management systems is rapidly increasing, but the implementation challenges can be eliminated with proper analysis. Ther are also many websites available on this day with various features. All these websites and systems have their advantages and disadvantages, so it is the agencies job to choose which is the best way possible for them

III. SYSTEM DESIGN

System Design:

The Car Rental Service Agency will be constructed using a combination of JavaFX and SceneBuilder for the user interface, JDBC for database connectivity, and the Model-View-Controller (MVC) framework for a structured architecture. The platform will feature a JavaFX-based graphical user interface created with SceneBuilder, a MySQL-powered backend database, and a Java-based controller layer facilitating seamless communication between the GUI and the database via JDBC.

System Architecture:

The architecture of the system will adhere to the Model-View-Controller (MVC) pattern, a structured design approach commonly used in the car rental service industry. In this pattern, the model component will manage the data and handle the essential business logic of the system. The view component will handle the graphical user interface (GUI), responsible for rendering the system's visual elements. Lastly, the controller component will serve as a bridge between the model and view, effectively managing user input, and ensuring that the model and view stay synchronized, updating them when necessary.

User Interface Design:

The user interface (UI) of the Car Rental Agency system will feature a sleek and intuitive design, ensuring a seamless experience for the customers. The primary dashboard will prominently feature a search bar, enabling customers to swiftly find available cars based on diverse criteria, such as car Id, model, owner and the availability. Search results will be elegantly presented below in a tabular layout, offering essential specifics about each vehicle, including make, model, and real-time availability status.

UML Class Diagram:

The UML class diagram for the Car Rental Service Agency portrays several classes and their relationships. The Cars class contains attributes such as Model Owner, ID, and availability status, and represents a Car in the Agency.

In addition to this, the system also includes two user classes: Admin and Customer. The Create User class is responsible for creating and managing user accounts. The Login class handles the authentication process for both Admin and Student users.

Furthermore, the system includes two additional classes to manage car circulation: Car Issue and Car Return. The Car Issue class handles the process of issuing cars to users, whereas the Car Return class manages the return of cars to the library and collecting penalties for overdue car returns. Finally, the Manage User class allows Admin users to manage user accounts, including modifying user details and resetting passwords.

Controller acts as a go-between for view and model. View displays the GUI (Graphical User Interface) of the project. The database is responsible for connection with MySQL and to add, delete, update status, and manage cars data.

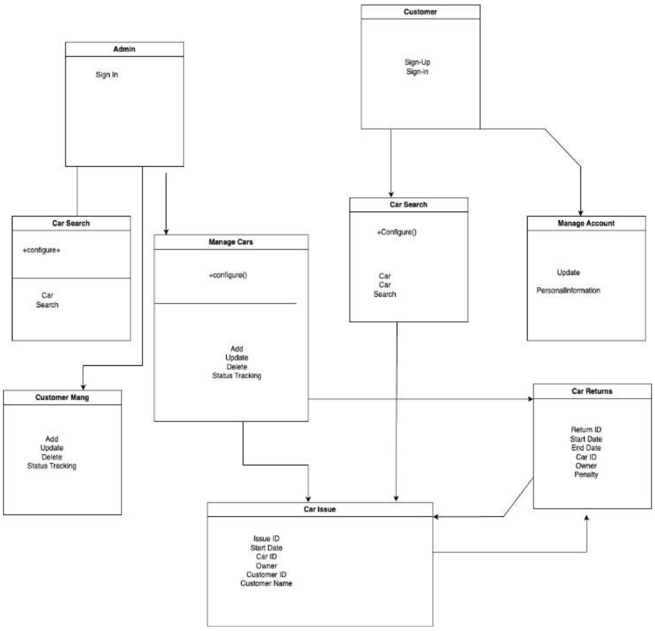


Figure 3. UML Diagram

IV. IMPLEMENTATION

A. JavaFX GUI Implementation

Developing the JavaFX GUI for the Car Rental Service Agency was a pivotal aspect of our development journey. Leveraging the robust JavaFX GUI library alongside SceneBuilder, we crafted an interface that not only boasts a contemporary design but is also incredibly intuitive for users. In crafting the layout of the application windows, we extensively harnessed the capabilities of the Anchorpane layout. This layout facilitated effortless arrangement and resizing of diverse elements within the interface, ranging from TextFields, Buttons, DatePickers, ComboBoxes, to TableView components. Our adoption of the Anchorpane layout ensures that the interface maintains its responsiveness and visual appeal, regardless of varying screen sizes or resolutions.

Our interface design further employed a variety of child elements. For example, the ImageView element played a pivotal role in presenting Cars and other images within the interface. To display various textual elements such as titles, labels, and messages, we employed the Text element. Additionally, the Font element allowed us to harmonize text styles, establishing a cohesive visual theme across the entire interface.

Incorporating the TextField element facilitated seamless user input of diverse data, encompassing details like names, IDs, or Car models. Buttons served as a means for users to interact with various functions, be it searching for a Car or adding a user. The DatePicker element enriched user experience by enabling straightforward date selections, such as the issuance or return dates of Cars. For presenting users with diverse choices like Car categories or user roles, we turned to the ComboBox element. To display comprehensive data lists—be it available Cars or user accounts—the TableView component proved invaluable. We harnessed an array of TableView functions, including sorting, filtering, and pagination, enhancing both the practicality and user-friendliness of the interface.

All in all, the JavaFX and SceneBuilder libraries furnished us with an adaptable and potent framework, enabling us to craft a contemporary and user-centric interface for the Car Rental Service Agency System

B. JDBC Implementation

To establish a seamless connection between the Java application and the MySQL database housing the Car Rental Service Agency system data, we employed the JDBC API. Specifically, we leveraged the Connector driver to facilitate this connection, enabling us to execute essential SQL queries for data retrieval, insertion, updating, and deletion from the database. Within the database, we structured various tables to accommodate distinct types of data pertinent to the Car Rental Service Agency system. For instance, the CarTM table was designated for Car-related information, housing columns such as Car ID, model, owner, and status. Meanwhile, the MemberTM table stored user-related details, encompassing fields like user ID, email, name, address, contact information, and password.

To effectively manage Car circulation, we employed two additional tables: CarIssueTM and CarReturnTM. The CarIssueTM table was tasked with recording Car issuance details, encompassing attributes such as issue ID, issue date, member ID, and Car ID. Similarly, the CarReturnTM table captured Car return specifics, including return ID, return date, issue date, and associated issue ID. Leveraging the JDBC API, we executed a range of SQL queries pertinent to these tables, spanning insertion, updating, and deletion operations. For example, when a user initiates a Car issuance, an insert query is executed to populate the CarIssueTM table with relevant particulars. Correspondingly, when a Car is returned, an update query is employed to refresh the CarReturnTM table with pertinent return date data. These SQL queries seamlessly facilitated efficient management and manipulation of data stored within the MySQL database.

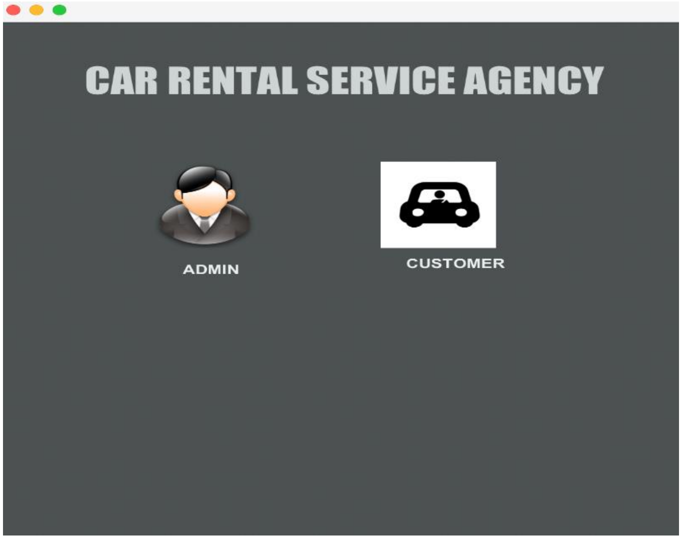
C. Controller Implementation

The Car Rental Service Agency was developed using the Model-View-Controller (MVC) architectural pattern within the Java programming framework. In this structure, the Controller played a pivotal role as an intermediary linking the Model and the View components. It managed user inputs, synchronizing them with the Model's state and facilitating updates to the View. This architectural approach facilitated the separation of concerns, enhancing codebase maintainability and adaptability. In parallel, we harnessed fundamental Java principles like Inheritance and Polymorphism to actualize the Car Rental Service Agency. These principles enabled us to establish a structured hierarchy of classes and leverage versatile behavior across different instances. By embracing interfaces and capitalizing on advanced concepts like Iterators, Generics, and Collections, we streamlined data handling, fostering efficiency and effectiveness. This amalgamation of architectural patterns and Java concepts yielded a resilient and scalable application primed to cater comprehensively to the requisites of the Car Rental Service Agency system

V. EVALUATION

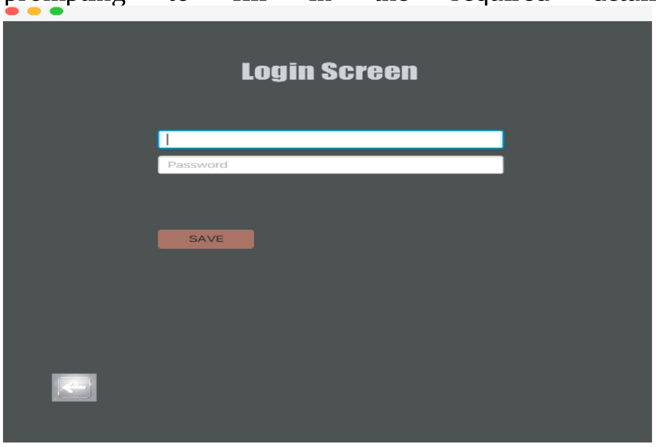
Home Screen:

This is the first screen we see when launching our car rental service agency. It offers two clickable images: "Admin" for administrative access, and "Customers" for customer access. Clicking on "Admin" will redirect us to the Admin login page, while selecting the "Customers" will take us to the Customer Sign-In/Sign-Up screen.


CAR RENTAL SERVICE AGENCY

Login Screen:

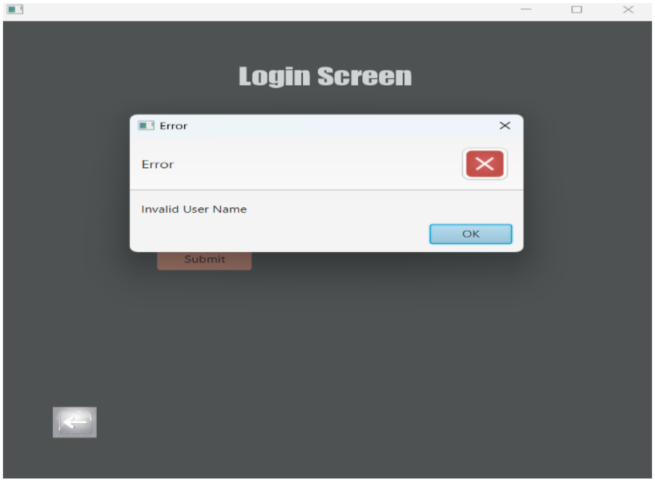
The admin login page contains fields where the admin can input their username and password. The credentials of the admin are predefined and we've implemented validation, so if any field is left blank, an error message will appear prompting to fill in the required details.


Login Screen

Password

If the provided credentials are accurate, the system will navigate to the Admin Operation Screen. However, if the details are incorrect, an error message will display indicating that the entered information is invalid.

The login screen for the customer follows the same procedure where they can input their username and password. But however, the credentials of the customers are not predefined. We've implemented a validation feature that triggers an error message if any of the fields are left empty, urging the user to complete all required details. When the provided credentials are accurate, the system will navigate to the Customer Operation Screen. Conversely, if the details entered are incorrect, an error message will be displayed, indicating that the provided information is invalid.


Login Screen

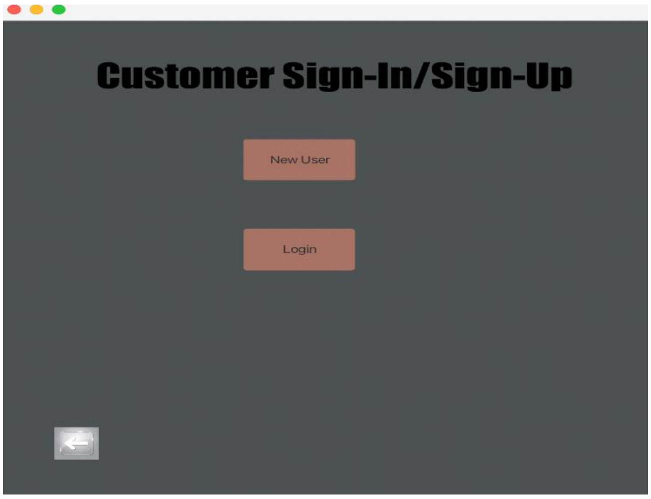
Error

Error

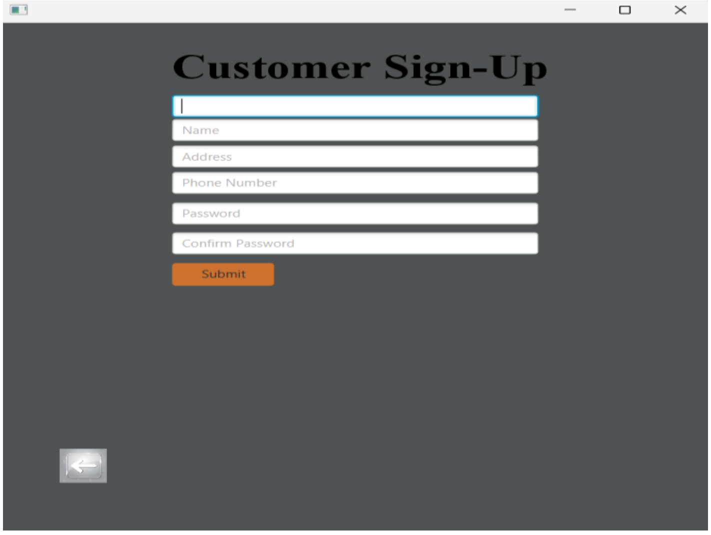
Invalid User Name

Customer Sign up/ Sign in page:

The Customer Sign-In/Sign-Up screen is a component of the project's login page, specifically tailored for customers. Its purpose is to offer customers the choice of either logging into their existing accounts or creating new ones. This screen features input fields where customers can enter their email addresses and passwords, along with buttons for initiating the sign-in or sign-up processes.


Customer Sign-In/Sign-Up

The Customer Sign-Up page is intended for new customers who want to create an account within the application. This page typically displays a form that the customer is required to complete, in which the customer is asked for various details such as name, address, phone number, and email id


Customer Sign-Up

Name

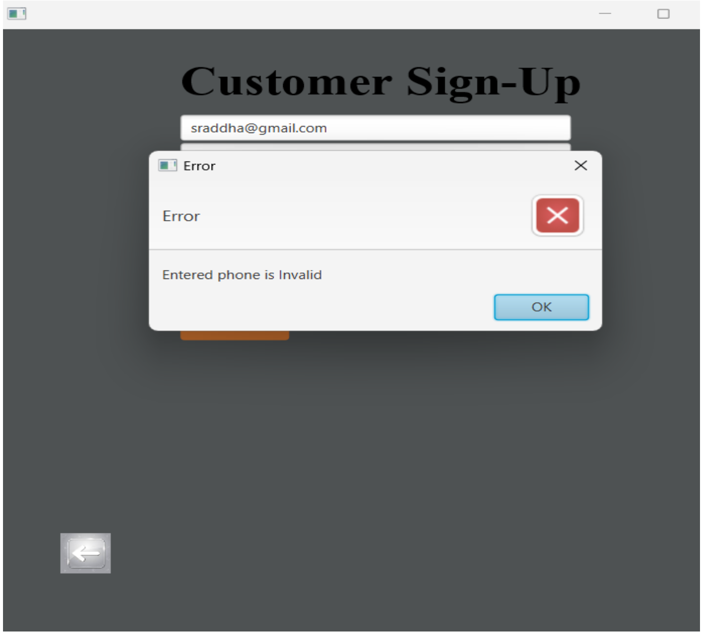
Address

Phone Number

Password

Confirm Password

If the customer misses out on any of the information, as we added regex to all of the fields, if the entered information is invalid, it throws an error. When the customer completes and submits this form, the system will proceed to create a new account for the customer and redirect accordingly.



Customer Sign-Up

sraddha@gmail.com

Error

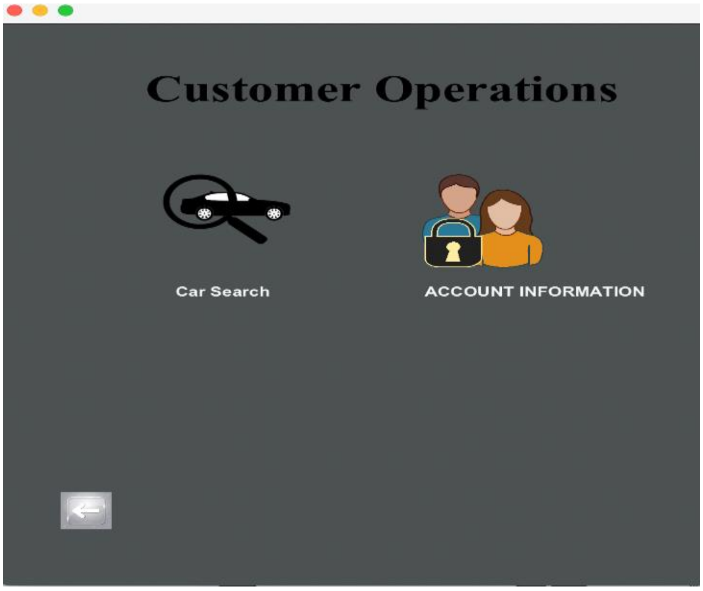
Error

Entered phone is Invalid

When a customer logs into the system, they'll be prompted to provide their email address and password. This information will be checked against the database of registered users to ensure its accuracy. If the login details are valid, the customer will be directed to the application's main home screen.

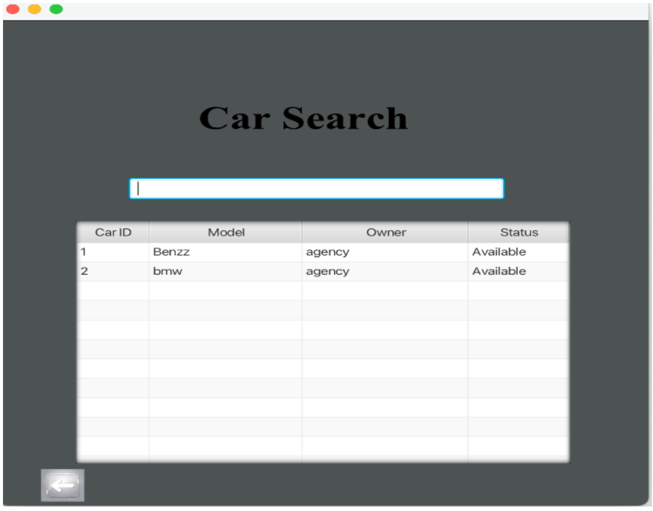
Customers Operations Page:

In this Customer operations screen we have car search and account information



Customer Operations

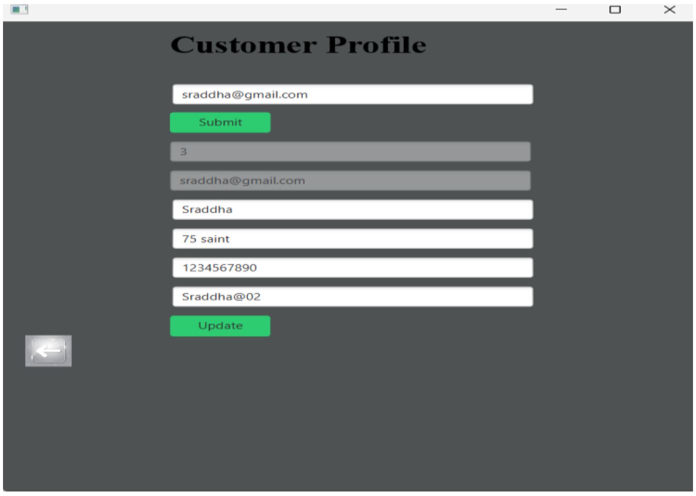
if the customer clicks on car search the system will redirect to the car search page where the customer can browse through a list of available cars and search the cars based on the car ID, car Model, car owner and availability.



Car Search

Car ID	Model	Owner	Status
1	Benzz	agency	Available
2	bmw	agency	Available

If, on the other hand, the customer clicks on Account information, it will be redirected to the customer account information page, where the customer must first enter their email address and then click on submit, after which the remaining details will be auto generated.



Customer Profile

sraddha@gmail.com

3

sraddha@gmail.com

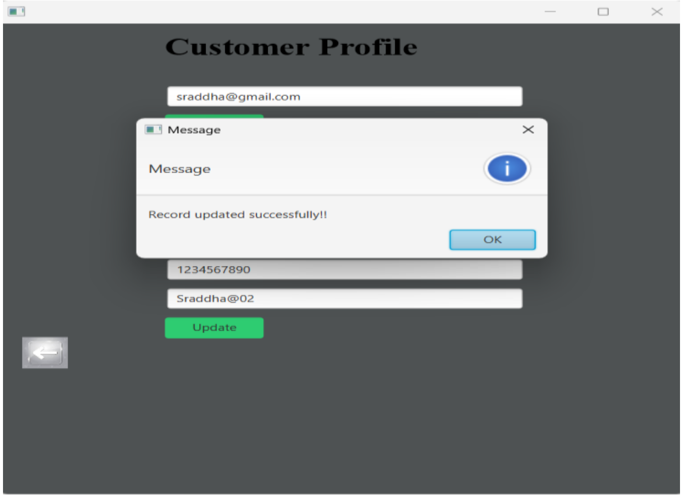
Sraddha

75 saint

1234567890

Sraddha@02

The customer will be able to update their information here. However, even on this page we have added the validations it throws an error message if any of the details doesnot satify the requirements.



Customer Profile

sraddha@gmail.com

Message

Message

Record updated successfully!!

OK

1234567890

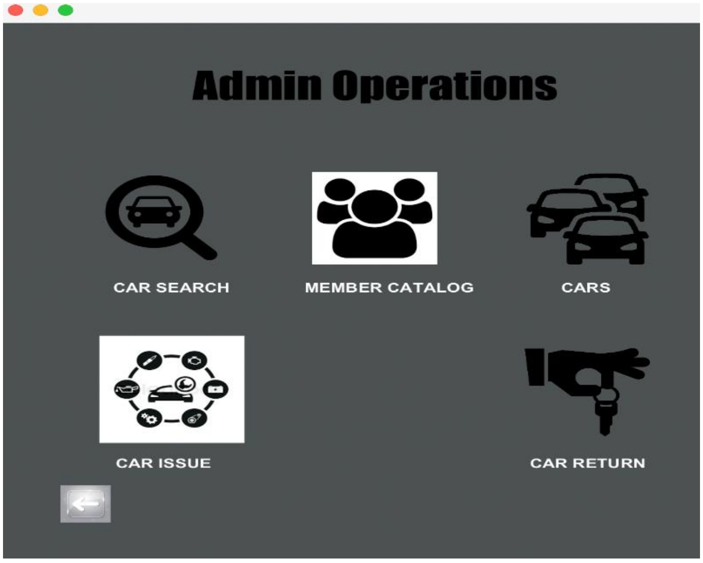
Sraddha@02

Update

Admin Operations Page:

The Admin plays a very crucial role in our car rental service agency. It empowers the admin to oversee and regulate the fleet, customer accounts, and handle other administrative responsibilities. Access to this screen is exclusive to authorized admin users with valid login credentials which are already predefined.

Once the admin has successfully logged in, the system will redirect to the admin operations page. The main essential operations of the admin are

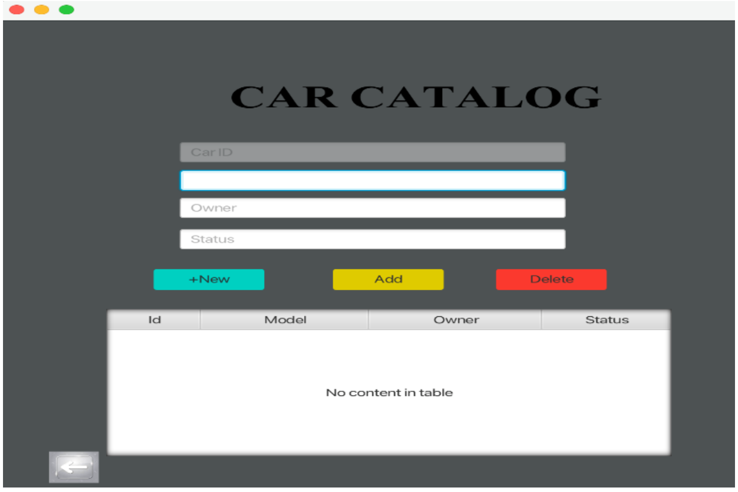


Admin Operations

Car Catalog:

The Car Catalog screen is tailored for the admin to handle the cars within the system. It provides capabilities such as adding new cars, removing existing ones, modifying car details, and displaying the list of cars available in the agency

Once if the admin clicks on the New button the Issue ID of the car will be generated and then the admin will be able to input the details of the car such model name, owner and then by clicking on the add button the car will be added to the car rental system. The admin also has the privilege of deleting the cars.



CAR CATALOG

Car ID

Owner

Status

+New Add Delete

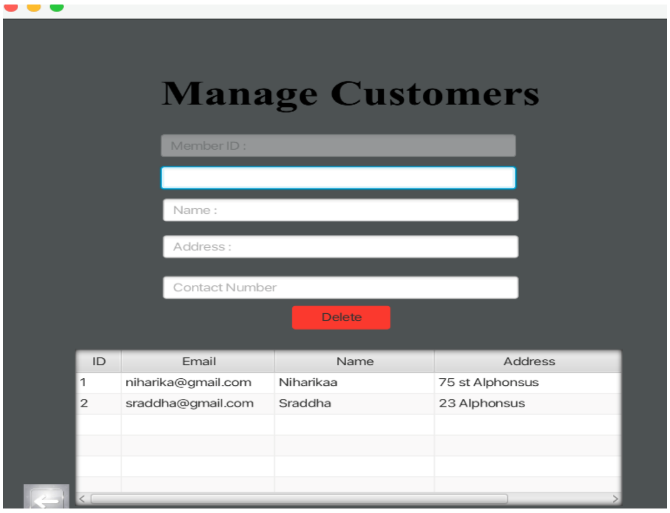
Id	Model	Owner	Status

No content in table

Manage Customers:

The Manage Customer Dashboard is a core feature of our Car Rental Service System, giving the admin the ability to oversee customer-related tasks. This dashboard empowers admin to manage customer profiles, including essential information like names, addresses, contact numbers, and email addresses. Additionally, the dashboard provides a valuable metric by showing the total count of registered customers, helping the admin understand the size of the customer base.

The Customer Dashboard boasts a user-friendly interface, simplifying customer data management. This functionality enhances the admin's ability to maintain accurate customer records, ultimately contributing to the efficient management of our car rental services.



Manage Customers

Member ID :

Name :

Address :

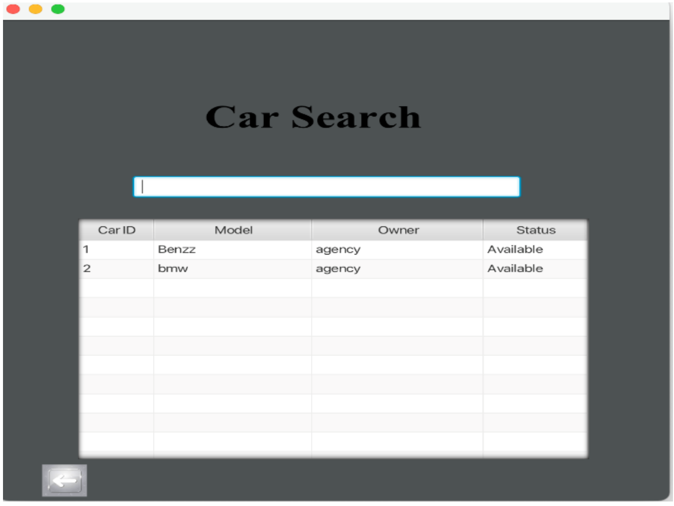
Contact Number

Delete

ID	Email	Name	Address
1	niharika@gmail.com	Niharikaa	75 st Alphonsus
2	sraddha@gmail.com	Sraddha	23 Alphonsus

Car Search Screen

Once the admin adds the cars into the car rental database. In this screen a search bar has been provided where both the customers and admin will be able search the cars based on Car ID, Model, owner and availability of the cars. This screen makes it is easy for the customers to navigate through the cars available and also the information about the cars and also for the admin to manage the cars easily by just browsing them



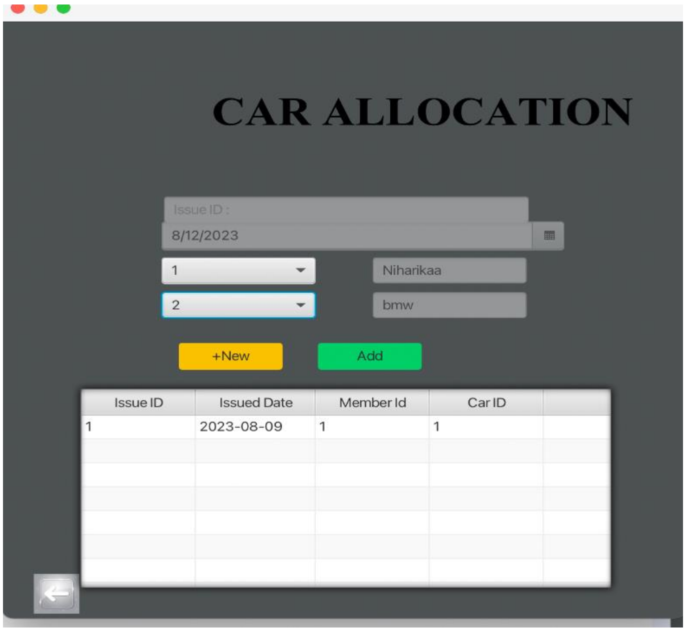
Car Search

Car ID	Model	Owner	Status
1	Benzz	agency	Available
2	bmw	agency	Available

Car Issue/Allocation Screen:

The Vehicle Rental Screen enables the Admin to rent vehicles to customers using our car rental service.

When the admin opens the car issue /allocation screen, once after clicking on the new button, an issue Id will be generated and the date when the vehicle is rented to the customer, and it automatically captures the current date. On selecting the car Id and member Id which was generated during adding of the cars and customers previously it automatically generates the customer name and model. If the admin now clicks on add button the car will be successfully allocated. Now we can go back to car search, we can see the status will be updated to "unavailable"



CAR ALLOCATION

Issue ID :

8/12/2023

1 Niharikaa

2 bmw

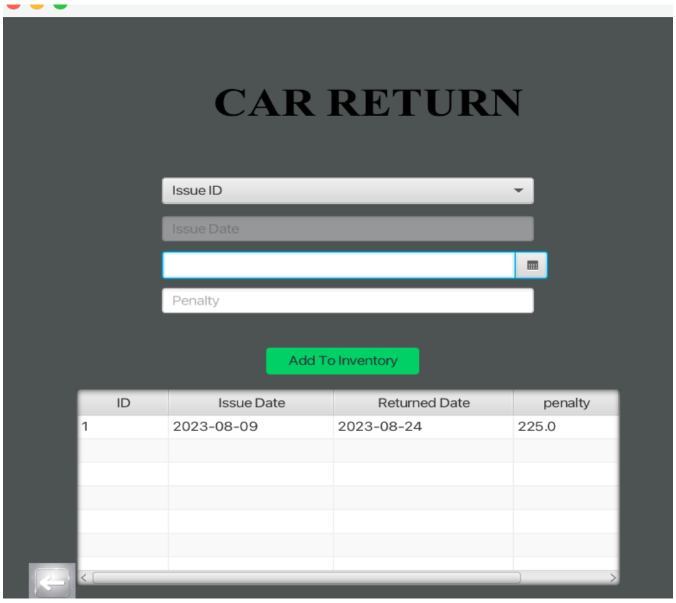
Issue ID	Issued Date	Member Id	Car ID
1	2023-08-09	1	1

Car Return Screen:

After the admin clicks the "car return" option,the system redirects to car return page where an Issue ID is retrieved from the car Issue table, which is generated automatically for each new record. The issue date is also automatically fetched from the car issue table, representing the date when the customer initially rented the car. In this process, the admin will choose the current date as the returned date of the car

If the customer returns the car on time, no fine is generated. However, if the car is returned after the initial two weeks from the issue date, a fine of fifteen dollars per day is automatically calculated based on the issue date and the returned date as we coded it in that way

Upon car return, the system updates the status of the car to "available." Consequently, when we navigate back to the car search function and check the car's status, it will reflect this automatic update to "available."



ID	Issue Date	Returned Date	penalty
1	2023-08-09	2023-08-24	225.0

Comparison with Other Teams:
In comparing our team's project to others, we analyzed key factors, such as user experience, features, speed, scalability, and data protection. Our application stood out in all these aspects, reflecting a comprehensive approach to delivering a high-quality application.

User Study with the real users:

We were unable to conduct a user study with real users due to the availability of resources and time.
Nonetheless, we thoroughly tested our application to ensure that it was user-friendly and simple to use. While building the application, we took feedback from our classmates and teaching assistants and incorporated their suggestions into our final product. We're confident our application is user-friendly, intuitive and effective, and we hope it meets the needs of potential users.

VII. Discussion (Reflection)

With the implementation of advanced analytics capabilities, the system empowers car rental companies to make data-driven decisions. By analyzing trends in vehicle demand, peak rental periods, and popular vehicle models, managers can optimize their fleet composition and pricing strategies. These insights have led to better-informed business strategies that maximize revenue and occupancy rates. Additionally, predictive maintenance analytics have helped reduce vehicle downtime and minimize maintenance costs.

One of the most significant outcomes of the project is the evident improvement in operational efficiency for car rental companies. By automating processes such as vehicle management, customer records, and rental transactions, the system has streamlined daily operations. Managers now have real-time visibility into the availability of vehicles, making it easier to allocate resources and meet customer demands promptly. The reduction in manual paperwork and administrative tasks has led to a notable increase in staff productivity.

The project's outcomes have facilitated business growth and expansion opportunities for car rental companies. The ability to efficiently manage operations, allocate resources, and maintain accurate records has provided a solid foundation for scaling up the business. Some companies have reported an increase in the number of transactions and repeat customers since the system's implementation. The streamlined processes have also allowed rental companies to explore new markets and diversify their offerings, such as luxury or eco-friendly vehicle options.

By optimizing fleet management, reducing paperwork, and automating tasks, the Car Rental Service Agency system has led to significant cost savings. These savings stem from reduced administrative overhead, minimized vehicle downtime, and improved resource allocation. Companies have reported notable reductions in operational costs and a healthier bottom line. The ability to identify underutilized vehicles and replace them with more in-demand models has further contributed to resource optimization.

VIII. Conclusions and Future Work

Conclusion:

In a nutshell, our project demonstrates that the car rental service agency's newly developed management system is both efficient and user-friendly for handling the agency's fleet of cars. The system's core features, such as car search, inventory management, car rental, car return processing, and fine calculation, have proven highly effective in efficiently managing the agency's car inventory and significantly enhancing the user experience.

Findings:

The deployment of the JAVAFX-JDBC-JAVA Car Rental Service Agency yields numerous advantages and benefits, such as

Stream Lining and automation: The car rental system automated tasks like checking car availability, searching, renting, and returns, saving time for both customers and staff, leading to improved operational efficiency.

Intuitive and User-Friendly Interface:The system focused on a user-friendly interface that improved the overall experience helping them easily navigate and , making it easy for customers to find and rent cars.

Efficiency and customer satisfaction:The car rental system proved to be more efficient, by automating tasks, making the rental process faster, reducing wait times, and lowering the chances of errors.

Problems encountered

- Ensuring the security of sensitive customer data, transaction details, and access control mechanisms is paramount. Implementing strong authentication, encryption, and protection against vulnerabilities like SQL injection and cross-site scripting is crucial.
- As the system grows and handles more data, ensuring optimal performance becomes vital. Optimizing database queries, managing system resources, and implementing caching strategies are important considerations.
- Integrating the JavaFX frontend with the MySQL backend and ensuring seamless communication between different components can be challenging. Handling data synchronization and maintaining consistency between different modules may require careful planning.
- Thoroughly testing the system to identify and rectify bugs, errors, and edge cases is time-consuming but necessary. Creating comprehensive test cases and conducting usability testing can help ensure a reliable and robust application.

Future Work

- The Car Rental Management System project holds significant potential for future expansion and enhancement. As technology continues to evolve and customer expectations evolve, there are several avenues to explore in terms of future scope. With the rise of ridesharing and on-demand transportation services, integrating the Car Rental Management System with mobility solutions could allow users to seamlessly switch between rental cars and other transportation modes, providing a more comprehensive travel experience.
- Incorporating advanced analytics and artificial intelligence capabilities could provide even deeper insights into customer preferences, demand patterns, and fleet optimization. AI-driven algorithms could recommend pricing strategies, predict vehicle maintenance needs, and enhance customer recommendations. Integrating with other relevant systems and services, such as insurance providers, payment gateways, and travel agencies, could create a seamless ecosystem for customers and streamline various aspects of the rental process.
- The system could explore partnerships with other stakeholders in the travel industry, such as hotels, airlines, or travel agencies, to offer bundled services and create a more holistic travel experience. Beyond managers and customers, additional user roles like maintenance staff, call center operators, and regional managers could be incorporated to streamline various operational aspects. Developing more comprehensive reporting and customizable dashboard features could provide managers with deeper insights into business performance and help them make data-driven decisions.

IX. Job Assignment

Niharika Santhoshini Karri

- Set up the development environment by installing the necessary tools (JavaFX and Scene Builder) configuring the project. Next, create a user-friendly UI with screens and components for adding, editing, and deleting cars from database. Additionally, implement the search and filter functionality so that users can easily find specific cars based on make, model, or availability. The goal is to create a streamlined system for managing car rentals.
- Implement the backend functionality for the car management module, enabling the ability to add, edit, and delete cars from the database. This involves creating the necessary functions and methods to handle these operations, such as adding new cars to the database, updating existing car details, and removing cars from the database. The goal is to ensure a robust backend system that effectively manages the car database in the car rental management system.
- Integrate the car management module with the database, ensuring synchronization between the module and the database. Perform thorough testing of the module to ensure its functionality and usability. This involves testing various features, such as adding, editing, and deleting cars, and documenting the bugs found during

testing and to ensure they work correctly and provide a user-friendly experience.

Sraddha Pedda Gangireddy gari

- Design the user interface (UI) for the client management module in the car rental system, enabling the addition, editing, and deletion of clients' information from the database. This includes creating screens and easily input and update borrower details such as name, contact information, and identification. Additionally, implement functionality to delete clients when necessary. The goal is to provide a user-friendly interface for managing client information in the car rental system.
- Implement the backend functionality for the client management module in the car rental system. This involves creating the necessary functions and methods to handle these operations, such as adding new client information to the database, updating existing client details, and removing clients from the database.
- Integrate the client management module with the database, ensuring synchronization between module and the database. Perform thorough testing of the module to ensure its functionality and usability. This involves testing various features, such as adding, editing, and deleting the clients, and documenting the bugs found during testing and to ensure they work correctly and provide a user-friendly experience.

Sahithi Reddy Marpadaga

- Create the UI for the rental transaction management module in the car rental system, enabling car borrowing and returning. Implement due date reminders and overdue notifications to keep clients informed about rental status.
- Implement the backend functionality for the client management module in the car rental system. This involves creating the necessary functions and methods to handle these operations, such as adding new client information to the database, updating existing client details, and removing clients from the database.
- Integrate the rental transaction management module with the database, ensuring synchronization between the module and the database. Perform thorough testing of the module to ensure its functionality and usability. This involves testing various features, due date reminders and overdue notifications, and documenting the bugs found during testing and to ensure they work correctly and provide a user-friendly experience.

REFERENCES

- Car Rental From Wikipedia : https://en.wikipedia.org/wiki/Car_rental
- Car Rental System from Research gate : https://www.researchgate.net/publication/353174644_Car_Rental_System
- What are the benefits of online car rental software?https://rentrax.com/what-are-the-benefits-of- online-car-rental-software/