

**LA GRANDEE INTERNATIONAL COLLEGE**

**Simalchaur, Pokhara Nepal**

A Final Report

On

**“Car Rental Management System”**

**Submitted to:**

Bachelor of Computer Application (BCA) Program

In partial fulfilment of the requirements for the degree of BCA under

Pokhara University

**Submitted by:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name:** | **Course** | **Semester** | **P.U. Registration Number** |
| R.A.Mohan Tiwari | BCA | 4th | 2022-1-53-0136 |
| Namrata Bastola | BCA | 4th | 2022-1-53-0129 |
| Rikita Gharti | BCA | 4th | 2022-1-53-0138 |
| Subash Gurung | BCA | 4th | 2022-1-53-0147 |

**Date: 28/06/2024**

# Acknowledgement

We are grateful to our project supervisor **Mr. Nabin Pandey** and our **BCA** coordinator **Mr. Ramesh Chalise** for the guidance, inspiration and constructive suggestions that helped us in the preparation of this project.

We are also appreciative among each other and understood the teamwork, the designation of the task per the skillset one represents constant synchronization and monitoring of progress and introducing new knowledge and skill is commanding for the success of any given work.

Sincerely,

Namrata Bastola

R.A Mohan Tiwari

Rikita Gharti

Subash Gurung

**Declaration for “Fleet-Ease”**

# Student’s Declaration

We hereby declare that we are the only authors of this work and that no sources other than the listed here have been using in this work. We assure you that the work we present here is unique to ourselves and likenesses to another similar project are purely coincidental.

With Regards

**Namrata Bastola (Roll No: 22530014)**

**R.A Mohan Tiwari (Roll No: 22530021)**

**Rikita Gharti (Roll No: 22530023)**

**Subash Gurung (Roll No: 22530032)**

Program: Bachelor of Computer Application

Semester: 4th Semester

Date: \_\_\_\_ / \_\_\_\_ / \_\_\_\_\_\_\_\_

# Supervisor’s Declaration

I hereby recommend that, **Ms. Namrata Bastola, Mr. R.A Mohan Tiwari**, **Ms. Rikita Gharti and Mr. Subash Gurung** did this project entitled “Fleet-Ease” under my supervisionduring their 4thSemester in partial fulfilment of the requirements for the degree of **BCA** under **Pokhara University** completed to my satisfaction and processed for final evaluation.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Mr. Nabin Panday**

**(Project Supervisor)**

Date: \_\_\_\_ / \_\_\_\_ / \_\_\_\_\_\_\_\_

# Letter of Approval

We certify that we have examined this report entitled “**Fleet-Ease**” and are satisfied with the project defense. It is satisfactory in the scope and qualify as project in partial fulfillment of the requirements for the degree of BCA under Pokhara University.

Date: \_\_\_\_ / \_\_\_\_ / \_\_\_\_\_\_\_\_

……………………

**Mr. Nabin Pandey**

Supervisor

…………………………

**Mr. Prithivi Raj Paneru**

External Examiner

………………..

**Er. Kiran K.C**

Principal

……………………

**Mr. Ramesh Chalise**

Department Coordinator

# Abstract

The Fleet-Ease is a innovative desktop application developed using VB.NET to automate and enhance car rental operations. This digital solution aims to streamline the process of vehicle reservations, bookings, and cancellations within a car rental agency.

The primary focus of this program is to provide a user-friendly interface for staff to efficiently manage vehicle inventory, handle customer bookings, and process rental transactions. The system enables staff members, even those with basic computer skills, to access and utilize its features effortlessly.

Customers benefit from the simplicity of booking vehicles through multiple channels, including in-person visits to the rental agency or by contacting them via phone. This accessibility ensures that customers can quickly secure their desired vehicle without undue delay.

The Fleet-Ease is designed with scalability in mind, allowing it to adapt seamlessly to evolving business requirements. It follows an Agile methodology, allowing for continuous enhancement and adaptation to changing business needs. This iterative approach ensures that the system evolves incrementally, incorporating feedback and updates efficiently throughout the development lifecycle.

Built on the VB.NET platform, known for its reliability and ease of development, this system ensures optimal performance and robust functionality. The use of VB.NET also facilitates seamless integration with existing systems and databases commonly used in car rental businesses.

Overall, the Car Rental system developed with VB.NET is a valuable tool for car rental agencies, enabling them to efficiently manage bookings, attract new customers, and optimize revenue streams. By utilizing this system, rental businesses can streamline operations, minimize administrative overhead, and deliver exceptional customer service, enhancing overall business productivity and customer satisfaction.

# TABLE OF CONTENTS

[Acknowledgement ii](#_Toc170507662)

[Student’s Declaration iii](#_Toc170507663)

[Supervisor’s Declaration iv](#_Toc170507664)

[Letter of Approval v](#_Toc170507665)

[Abstract vi](#_Toc170507666)

[TABLE OF CONTENTS vii](#_Toc170507667)

[Abbreviation 1](#_Toc170507668)

[1. Introduction 2](#_Toc170507669)

[2. Problem Statement 3](#_Toc170507670)

[3. Objectives 5](#_Toc170507671)

[4. Background Study 6](#_Toc170507672)

[5. System Design: 7](#_Toc170507673)

[5.1 Class Diagram 7](#_Toc170507674)

[5.2 E-R Diagram 8](#_Toc170507675)

[5.3 Use Case Diagram 10](#_Toc170507676)

[5.4 Flowchart 11](#_Toc170507677)

[5.5 Data Flow Diagram (DFD) 14](#_Toc170507678)

[6. Development 17](#_Toc170507679)

[7. Testing 19](#_Toc170507680)

[8. Requirements Document 27](#_Toc170507681)

[8.1 Functional Requirements 27](#_Toc170507682)

[8.2 Non-Functional Requirements 28](#_Toc170507683)

[8.3 Technologies Used 29](#_Toc170507684)

[9. Deliverables 30](#_Toc170507685)

[10. Project Result 31](#_Toc170507686)

[10.1 What are the function achieved by our project? 31](#_Toc170507687)

[10.2 What problems have been solved by the project? 32](#_Toc170507688)

[10.3 What requirements has been met? 33](#_Toc170507689)

[11. Future Enhancement 34](#_Toc170507690)

[12. Conclusion 35](#_Toc170507691)

[13. Annexures 36](#_Toc170507692)

[14. Project Gantt Chart 44](#_Toc170507693)

[15. References 45](#_Toc170507694)

# Abbreviation

|  |  |
| --- | --- |
| BCA | Bachelor of Computer Application |
| PU | Pokhara University |
| VB.NET | Visual Basic. NET |
| E-R Diagram | Entity-Relationship Diagram |
| DFD | Data Flow Diagram |
| UI/ UX | User Interface/ User Experience |

# Introduction

In today's business landscape, technology is a driving force across various industries, including transportation services like car rentals. Computers and desktop applications have become indispensable tools for managing business activities efficiently, especially within the car rental sector. As the demand for convenient and flexible transportation solutions continues to grow, the car rental industry is poised for significant expansion this year.

The Fleet-Ease is a desktop-based application developed to streamline vehicle rental processes for both personal and business use. The term ‘Fleet’ stands for number of vehicles and ‘Ease’ stands for simplicity. This system recognizes the increasing need for efficient digital solutions within the car rental market, offering customers a user-friendly platform to browse available vehicles, compare rental options, and make reservations effortlessly.

Similar to the demand for other booking services application, which cater to easiness and wellness needs, car rental services cater to the mobility and transportation needs of individuals and businesses alike. The Fleet-Ease aims to meet these needs by providing a seamless booking experience through a desktop interface.

Users of the Car Rental System can explore various car rental agencies, review vehicle options, select rental durations, and complete bookings with ease. This system focuses on simplifying the reservation process, allowing customers to check vehicle availability, understand rental terms, and view associated costs upfront.

By leveraging desktop technology, the Car Rental System empowers users to access and book rental vehicles efficiently, supporting the growing demand for flexible transportation solutions. This innovative application is designed to optimize operations for car rental businesses and deliver exceptional customer service in today's dynamic and competitive market.

# 2. Problem Statement

The existing Car Rental booking systems often face challenges related to manual processes, inefficiency and customer dissatisfaction, leading to revenue loss and operational issues. These challenges highlight the need for modernized and streamlined approach to managing vehicle reservations

The problems before the proposal of this system were:

* **Time consuming and manual appointment scheduling:**

Our current booking process relies heavily on manual tasks, making it time-consuming for customers to reserve vehicles. This often leads to long waiting times and inconvenience.

* **Difficulty in managing appointments:**

With manual bookings systems, it’s challenging to effectively manage vehicle reservations, especially during peak periods. This can result in errors, missed bookings, and customer dissatisfaction.

* **Lack of customer convenience:**

Traditional booking methods limit customer convenience, as customers may need to allocate specific time slots to visit rental agencies. This inconvenience can lead to customer frustration and deter repeat business.

* **Tension due to manual recording:**

Manual record-keeping for vehicle reservations poses risks such as lost or damaged records, making it difficult for staff to manage bookings effectively. This can result in administrative burdens and operational inefficiencies.

To address these challenges and enhance the customer service of the Car Rental system, we are proposing the development of a digitalized car rental system. This system will utilize modern technology to automate and simplify our booking processes, providing customers with a seamless and convenient experience while improving our operational efficiency and overall business performance.

# 3. Objectives

The objective of this project is to develop a modernized Car Rental System using advanced technology to enhance the efficiency and effectiveness of our vehicle booking process.

The proposed objectives of our Car Rental System are:

* **Security and Data Privacy:**

To implement robust security protocols to ensure that only authorized personnel have access to customer information and booking details safeguarding data privacy and confidentiality.

* **User registration and Profile Management:**

To enable staff to register new customers seamlessly by capturing essential details such as name, contact number, email address and other relevant information required for vehicle bookings.

* **Vehicle Booking Management:**

To provide staff with tools to efficiently manage vehicle reservations, including adding new bookings, cancelling existing reservations and accommodating rescheduling requests based on customer preference and availability.

* **Intuitive and User-Friendly interface:**

To develop a user-friendly interface that simplifies navigation and reduces the risk of errors, empowering staff to manage bookings efficiently and effectively.

* **Dynamic Price Management and Policy Implementation:**

To equip authorized personnel with the ability to adjust rental prices in alignment with business policies and market trends, ensuring competitiveness and profitability within the Car rental industry.

By achieving these objectives, our Car Rental System will optimize operational workflows, elevate customer service standards, and position Fleet-Ease as a leader in providing convenient and reliable vehicle rental service applications. This digital solution will enhance overall business performance and customer satisfaction, driving success and growth in the competitive car rental market.

# 4. Background Study

In the recent year, with the rise of digital technology and increasing customer expectations for convenience has change the way of renting or booking vehicles, offering features like real-time updates, improving customer satisfaction, etc. Traditionally, car rental companies faced numerous challenges such as manual booking errors, inefficient fleet management, and time-consuming renting processes. These issues often led to double bookings, vehicle downtime, and customer dissatisfaction. However, with the advent of desktop application, such as ours, renting has become more efficient, secure, and easier.

With the increasing demand for digital solutions and customer expectations for seamless experiences, a car rental management system becomes indispensable. Our system streamlines the entire rental process, from reservations and checkout and customer detail management and ensuring accuracy and efficiency. The system provides real-time availability updates and personalized services, significantly improving the overall customer experience. Additionally, it maintains compliance by accurately managing all necessary documentation and offers robust reporting and security. By implementing our car rental management system (Fleet-Ease), businesses can streamline their operations, reduce errors, and position themselves competitively in a rapidly evolving market.

# 5. System Design:

System Analysis and Design is a crucial phase in the development of Fleet-Ease. It

involves understanding the requirements of the application, defining its architecture,

In addition, creating a blueprint for its implementation. The process includes gathering user

Needs, analysing existing systems.

## 5.1 Class Diagram

A class diagram in software engineering is a type of structural diagram that depicts the static structure of a system by showing its classes, attributes, operations, and relationships between objects. It provides a high-level overview of the system's architecture and the interactions between its components.

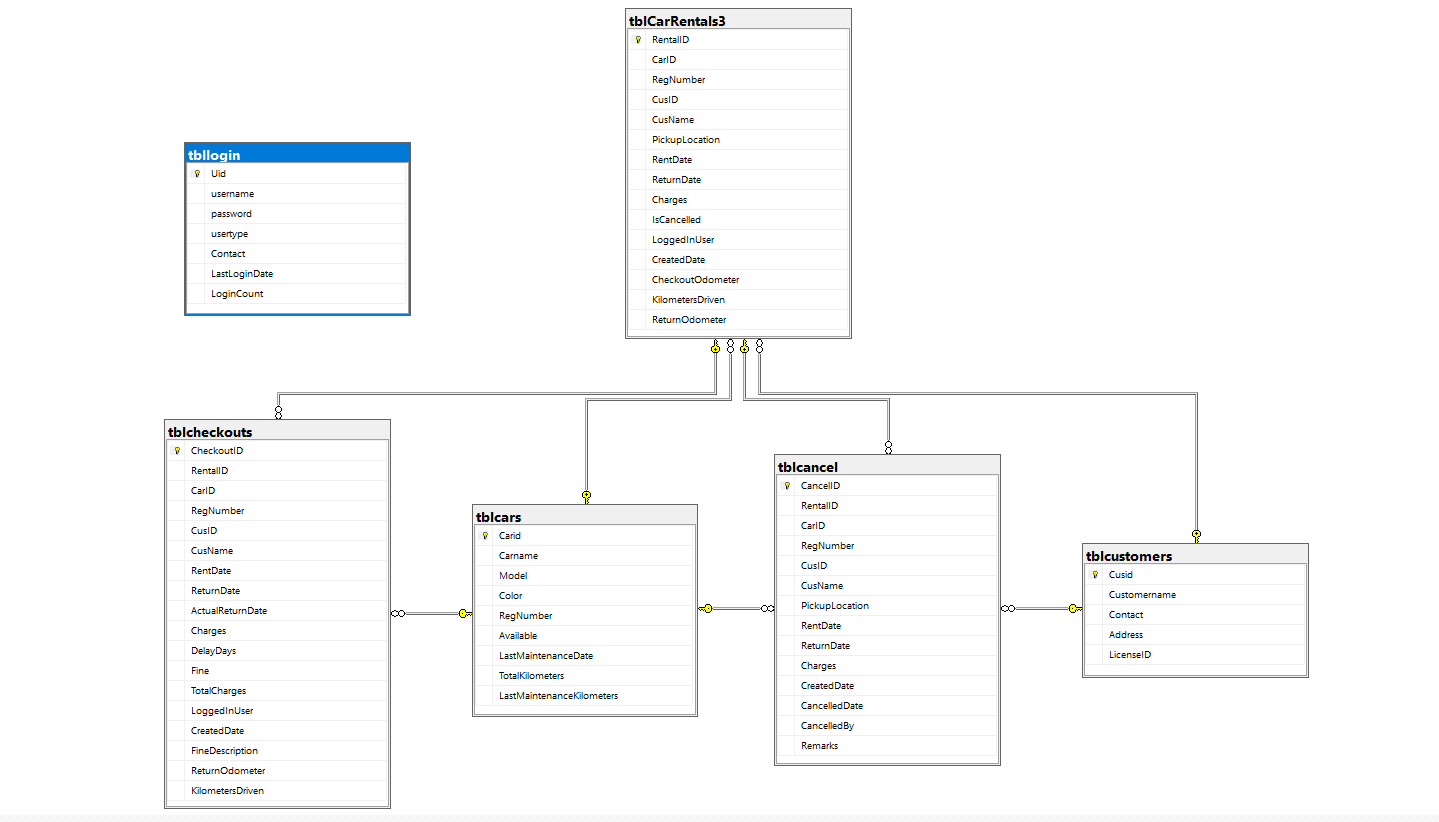


Fig 5.1.1: Class Diagram

## 5.2 E-R Diagram

An Entity-Relationship (ER) diagram is a type of structural diagram used in database design to visually represent the entities (tables) in a database and their relationships with each other. ER diagrams are crucial for planning and understanding database structures, including their attributes and the associations between different entities.

The Entity-Relationship (ER) Diagram below provides a detailed representation of the data structure and relationships of the Fleet-Ease application. This diagram is essential for understanding how data entities interact with each other in the database, forming the foundation of the application's data model.

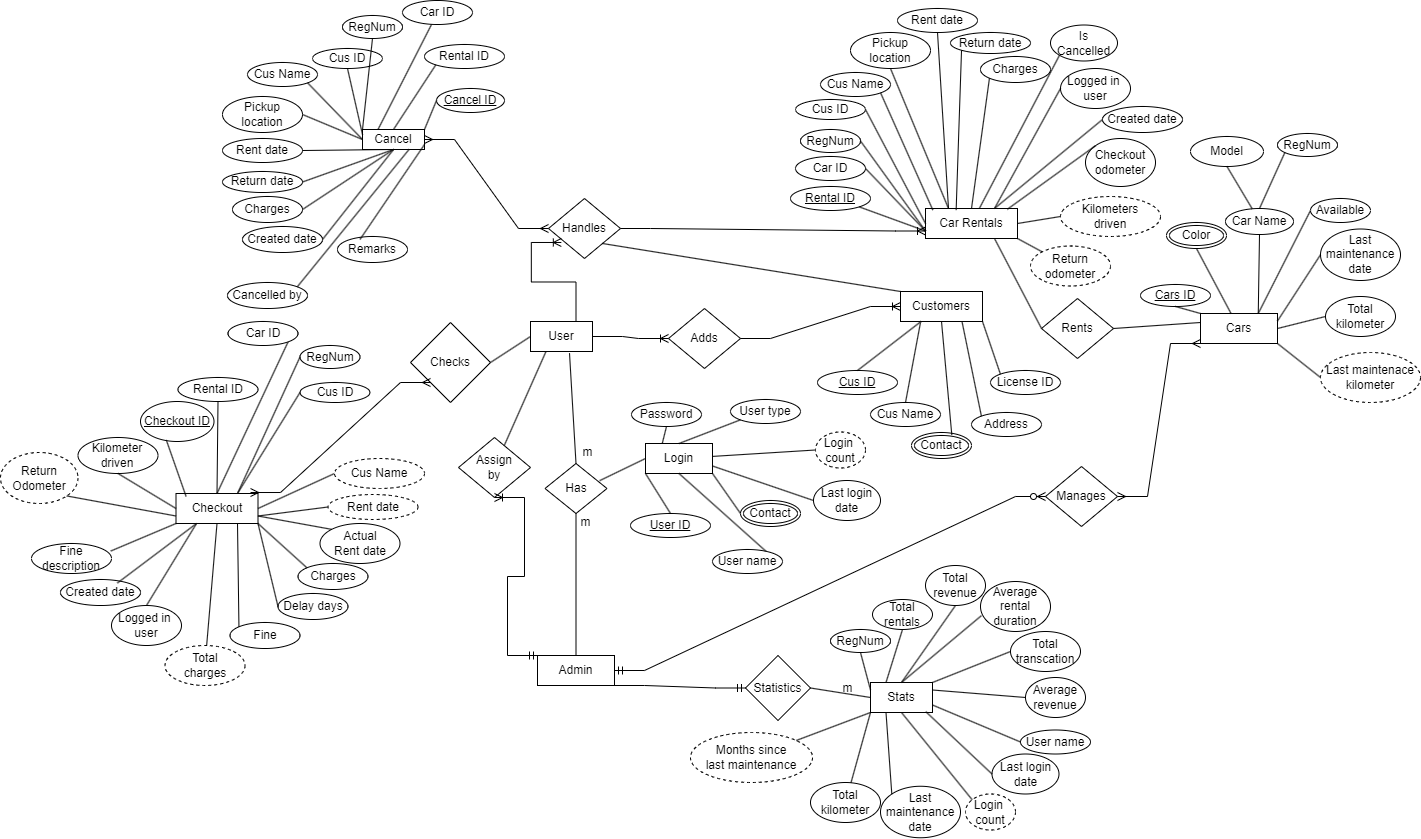


Fig 5.1.1: ER-Diagram

## 5.3 Use Case Diagram

A Use Case Diagram is a behavioural diagram in Unified Modelling Language (UML) that illustrates the interactions between actors (users or systems) and a system under various scenarios.

The Use Case Diagram below provides a visual representation of the interactions between various actors and the functionalities of the Fleet-Ease application. It illustrates how users and admin interact with the system to achieve their goals. This diagram serves as a foundational tool to understand the system’s functional requirements and the different use cases that the platform supports.

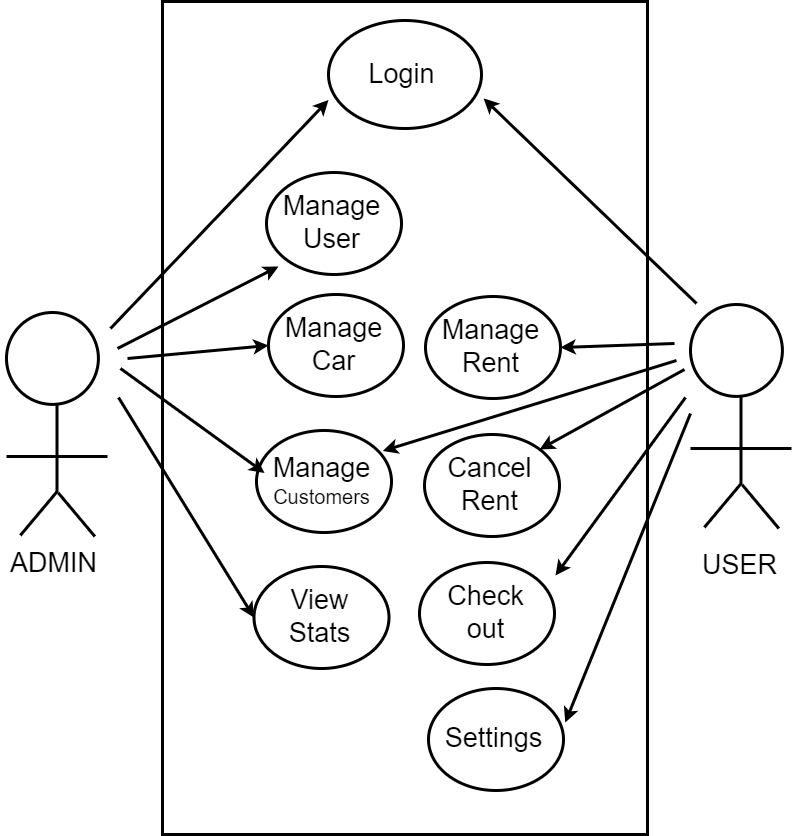


Fig 5.3.1: Use Case Diagram

## 5.4 Flowchart

The flowchart diagrams below illustrate the key processes and interactions within the Fleet-Ease application, made for user and admin. Each flowchart provides a visual representation of the distinct user journeys and the various actions that take place within their respective roles. These flowcharts serve as roadmaps for understanding the functional flow of the application, highlighting the main components and their interactions for each user type.

* Start Menu

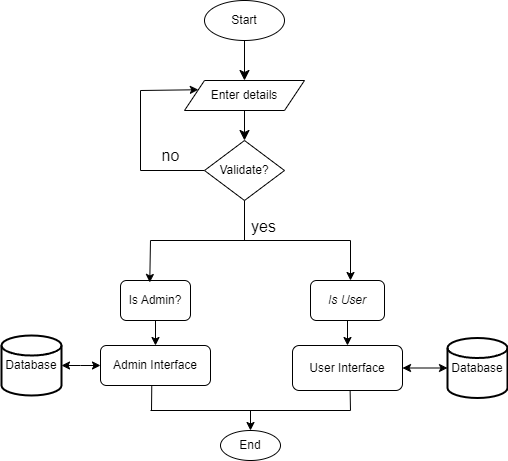


Fig 5.4.1: Start Menu

* Admin Menu

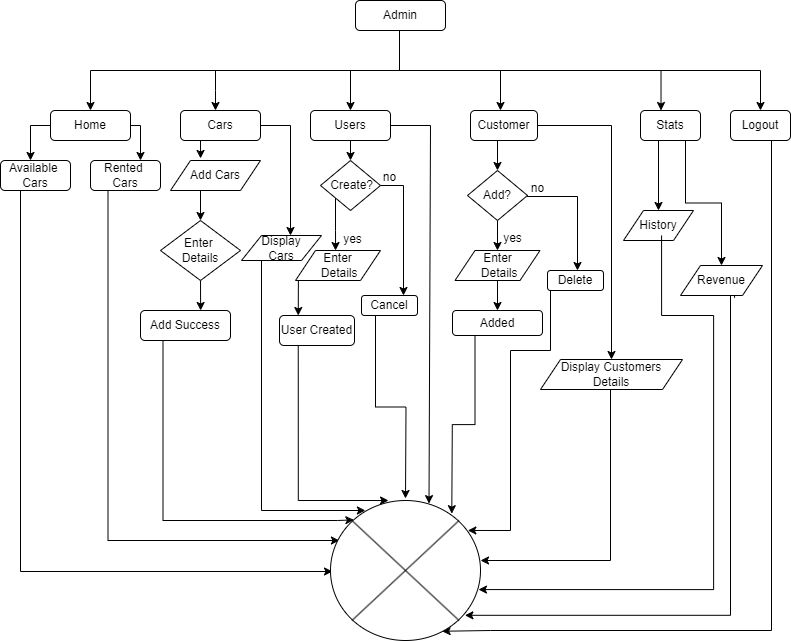


Fig 5.4.2: Admin Menu

* User Menu

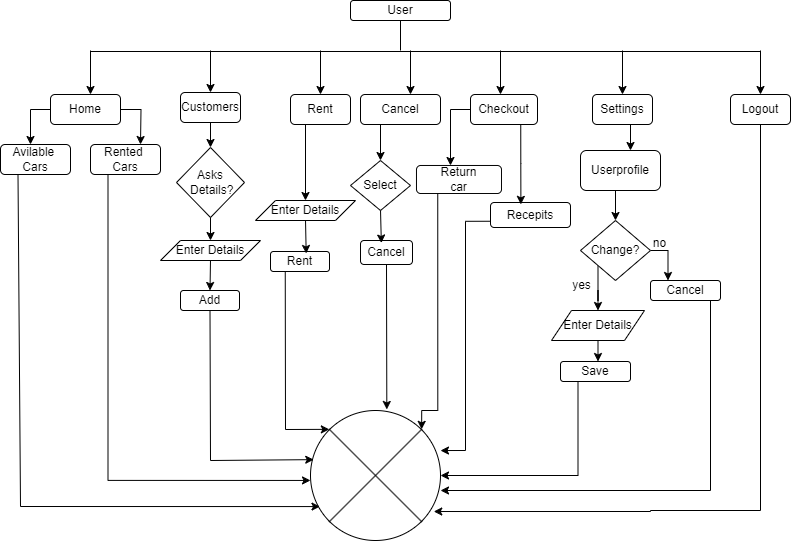


Fig 5.4.3: User Menu

## 5.5 Data Flow Diagram (DFD)

A Data Flow Diagram (DFD) is a graphical representation that illustrates the flow of data within a system. It is commonly used in system analysis and design to depict the interactions between processes, data stores, external entities, and data flows.

The Data Flow Diagram (DFD) below provides a detailed representation of how data moves through the Fleet-Ease application. It illustrates the interaction between different processes, data stores, and external entities within the system. This DFD serves as a visual tool to understand the data processing and flow within the application, ensuring a clear depiction of the system’s architecture and the data lifecycle.

The Level 0 Data Flow Diagram (DFD), also known as a Context Diagram, provides a high-level overview of the entire e-commerce system, illustrating the system’s boundaries and interactions with external entities. This diagram is crucial for understanding the main processes and the flow of information within the system.

* DFD 0

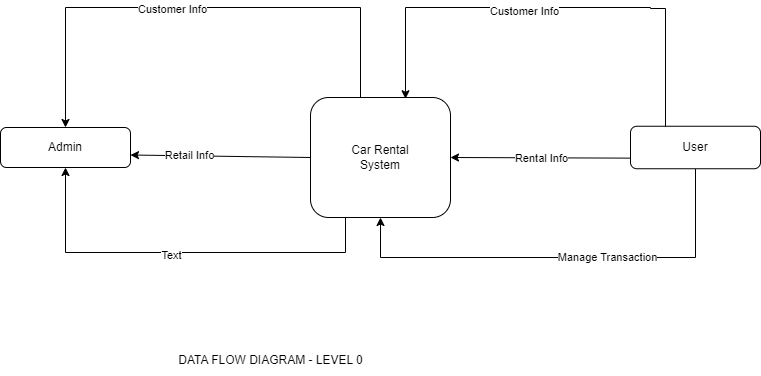


Fig 5.3.1: DFD 0

* DFD 1

A Data Flow Diagram (DFD) Level 1 provides a more detailed view of the system than the context diagram (Level 0). It breaks down the main process of the system into sub-processes and illustrates the flow of data between these sub-processes.

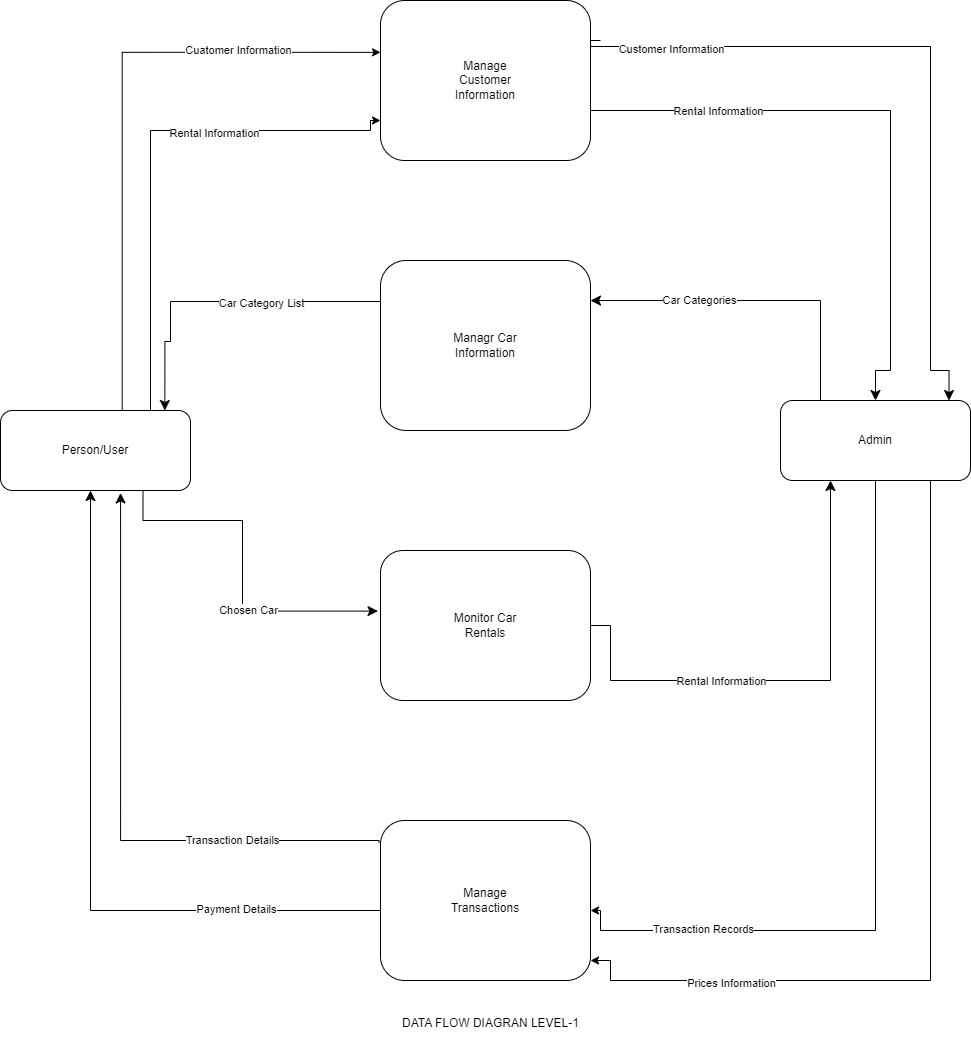


Fig 5.3.2: DFD 1

* DFD 2

A Data Flow Diagram (DFD) Level 2 further decomposes the processes identified in the Level 1 DFD into more detailed subprocesses. This level of DFD provides a finer granularity of the processes and data flows within the system, offering a deeper understanding of how the system functions.

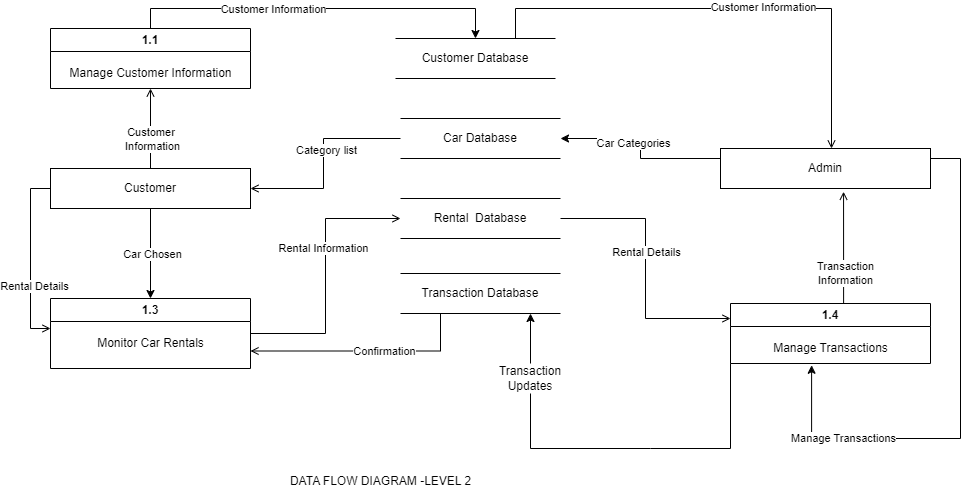


Fig 5.3.3: DFD 2

# 6. Development

During the development phase of our desktop application, Fleet-Ease, using the agile model, we embarked on an iterative and collaborative journey to bring our application idea to fruition. This phase was instrumental in transforming our initial project scope into a robust and user-friendly desktop application, catering to a diverse range of user needs.

The development process structured into two-week sprints, each with clear objectives aimed at delivering specific functionalities within set timelines. We conducted regular sprint planning sessions and retrospective meetings online to ensure effective progress tracking and continuous improvement. These virtual sessions helped us refine our goals, adjust priorities, and align team efforts to meet our development milestones.

Collaboration was pivotal throughout our development journey. Online meetings facilitated open communication among team members, providing a platform to share progress updates, discuss challenges via video conferencing, and synchronize our efforts effectively. This proactive approach enabled us to identify potential bottlenecks early on and maintain momentum throughout the development cycle.

Throughout the development phase, our focus was on implementing key features that enhance the desktop application experience for our users within Fleet-Ease. Core functionalities such as adding users and admin, renting cars, and maintaining their details meticulously designed to ensure seamless navigation.

Maintaining code quality and stability was a priority for us. We leveraged continuous integration practices using GitHub, which automated the process of code sharing and ensured that our app remained deployable at all times.

In conclusion, the development phase of our desktop application, Fleet-Ease using the agile methodology characterized by collaboration, adaptability, and a relentless focus on user satisfaction. By embracing agile principles and conducting online meetings, we successfully delivered a feature-rich application that meets the diverse needs of our user. Continuous integration and user-centric design enabled us to create an app that not only meets but also exceeds user expectations in the competitive desktop application landscape.

We group of four individual students confined to the development of the fourth semester project and here is the work division chart among us:

|  |  |  |  |
| --- | --- | --- | --- |
| S.N | Name of the member | Work assigned | Remark |
| 1. | Namrata Bastola | * Documentation * UI design * Functionality testing | Managed documentation, contributed to UI design, and tested application functionalities. |
| 2. | R.A Mohan Tiwari | * Coding * Database Management * Problem identification * Requirement document | Led the project, handled all coding, database management, and problem-solving. |
| 3. | Rikita Gharti | * Documentation * UI design * Functionality testing | Led documentation efforts, contributed to UI design, and conducted functional testing. |
| 4. | Subash Gurung | * System design * Documentation * UI design * Full functionality testing | Designed system architecture, created a significant portion of the UI, and performed comprehensive functionality testing. |

# 7. Testing

Our desktop application Fleet-Ease, will be using agile methodology, which includes a variety of techniques adapted to its changing needs. Projects with changing needs, like Fleet-Ease, where the features and functionalities the platform needs may change, are ideally suited for the agile methodology. The meaning of Agile is swift or versatile. “Agile process model” refers to a software development approach based on iterative development. It will break tasks into smaller iterations, or parts do not directly involve long term planning. Adopting adaptive planning gives Fleet-Ease, the flexibility to adjust to shifting market conditions and requirements, ensuring that the company continues to meet the needs of its customers.

The image below shows the process involved in Agile Methodology:



Fig 7.1: Agile Methodology

For the development of the Car Rental System, we will adopt the Agile methodology. Agile is a flexible and iterative approach to software development that emphasizes collaboration, adaptability, and continuous improvement throughout the project lifecycle.

Using the Agile methodology, our development process will involve:

* + Iterative Development:

We will break down the project into manageable iterations or sprints, with each iteration delivering working software that adds value to the system.

* + Continuous Feedback and Adaptation:

We will engage stakeholders, including internal teams and customers, to gather feedback regularly. This feedback will drive ongoing improvements and refinements to the system.

* + Cross-Functional Teams:

Our development teams will be cross-functional, comprising members with diverse skills (e.g., developers, testers, designers). This setup fosters collaboration and collective ownership of project goals.

* + Emphasis on Customer Collaboration:

We will prioritize customer collaboration throughout the development process to ensure that the Car Rental System meets their evolving needs and expectations.

* + Responding to Change:

Agile allows us to respond quickly to changing requirements and market dynamics by embracing flexibility and adaptability in our development approach.

By leveraging Agile principles and practices, we aim to deliver a high-quality Car Rental System that aligns closely with customer requirements, enhances user experience, and accelerates the delivery of value to our business and customers. This iterative and customer-centric approach will enable us to optimize project outcomes and drive success in the competitive car rental industry.

Admin testing

* Test Case 1 (Admin or User login)

Table 1 Test Case 1 (Admin or User login by providing valid user name and password)

|  |  |
| --- | --- |
| Objectives | To login Successfully. |
| Action | Entering valid and correct requirements. |
| Expected Result | User or admin successfully logged in and directed to their dashboard page. |
| Actual Result | Show the admin and user dashboard page. |
| Conclusion | The test was successful. |
| Test Executed by | Rikita Gharti |
| Test Reviewed by | R.A Mohan Tiwari |

* Test Case 2 (Admin or User login by providing invalid user name and password)

Table 2 Test Case 2

|  |  |
| --- | --- |
| Objectives | To check if it still login successfully or not. |
| Action | Entering invalid and incorrect requirements. |
| Expected Result | Display error message. |
| Actual Result | Displayed error message. |
| Conclusion | The test was successful. |
| Test Executed by | Rikita Gharti |
| Test Reviewed by | R.A Mohan Tiwari |

* Test Case 3 (Leave any text field empty after login to add details)

Table 3 Test Case 3

|  |  |
| --- | --- |
| Objectives | To check whether it runs successfully or not. |
| Action | Leaving any text field empty. |
| Expected Result | Display error message. |
| Actual Result | Displayed error message. |
| Conclusion | The test was successful. |
| Test Executed by | Namrata Bastola |
| Test Reviewed by | R.A Mohan Tiwari |

* Test Case 4 (Click on any icons)

Table 4 Test case 3

|  |  |
| --- | --- |
| Objectives | To ensure all icons works. |
| Action | Click on any icons. |
| Expected Result | Display icon’s features. |
| Actual Result | Icon’s features are opened. |
| Conclusion | The test was successful. |
| Test Executed by | Namrata Bastola |
| Test Reviewed by | Rikita Gharti |

* Test Case 5 (Entering valid (10 numbers) contact number)

Table 5 Test Case 5

|  |  |
| --- | --- |
| Objectives | To ensure contact number contain only 10 numbers. |
| Action | Entering 10 numbers in contact text. |
| Expected Result | Continue run the process/ task. |
| Actual Result | Program runs continuously and perform fuether task. |
| Conclusion | The test was successful. |
| Test Executed by | R.A Mohan Tiwari |
| Test Reviewed by | Subash Gurung |

* Test Case 6 (Entering more or less than 10 numbers in contact text)

Table 6 Test Case 6

|  |  |
| --- | --- |
| Objectives | To check whether program continue even after entering wrong contact number. |
| Action | Entering more or less than 10 numbers in contact text. |
| Expected Result | Display error message. |
| Actual Result | Displayed error message. |
| Conclusion | The test was successful. |
| Test Executed by | R.A Mohan Tiwari |
| Test Reviewed by | Subash Gurung |

* Test Case 7 (Click on checkout icon)

Table 7 Test Case 7

|  |  |
| --- | --- |
| Objectives | To ensure checkout function works. |
| Action | Click on checkout icon. |
| Expected Result | Display checkout functions. |
| Actual Result | Displayed checkout functions (like after entering car reg number all the necessary requirements are filled automatically). |
| Conclusion | The test was successful. |
| Test Executed by | Subash Gurung |
| Test Reviewed by | R.A Mohan Tiwari |

* Test Case 8 (Adding and updating the car)

Table 8

|  |  |
| --- | --- |
| Objectives | To ensure adding and updating function works. |
| Action | Filling the requirements of car to add and update. |
| Expected Result | Add new car and update previous car details. |
| Actual Result | New car added and updated previous car details. |
| Conclusion | The test was successful. |
| Test Executed by | Subash Gurung |
| Test Reviewed by | R.A Mohan Tiwari |

* Test Case 9 (Changing user’s password)

Table 9

|  |  |
| --- | --- |
| Objectives | To check whether user can change their password or not. |
| Action | Clicking on setting icon and fill the necessary requirements to change password (like old password, new password, etc.). |
| Expected Result | Old password get changed by new password. |
| Actual Result | Password was changed successfully. |
| Conclusion | The test was successful. |
| Test Executed by | Rikita Gharti |
| Test Reviewed by | Subash Gurung |

* Test Case 10 (logout)

Table 10

|  |  |
| --- | --- |
| Objectives | To determine if user can log out when clicking log out tab. |
| Action | Click log out icon. |
| Expected Result | Redirects to the Login page of the application |
| Actual Result | Display login screen. |
| Conclusion | The test was successful. |
| Test Executed by | Namrata Bastola |
| Test Reviewed by | Subash Gurung |

# 8. Requirements Document

Our application is a desktop-based car rental management system designed to streamline the entire car rental process. It aims to provide a seamless experience for users and admin by managing various aspects of the car rental service. The system will handle user registrations, car inventory management, and rental bookings. The requirement documentation outlines the functional and non-functional requirements of Fleet-Ease.

## 8.1 Functional Requirements

In Fleet-Ease, functional requirements refer to specific features and capabilities that the desktop application must possess to fulfil its intended purpose and meet the needs of its users and stakeholders. These requirements define the desired behaviour of the application and outline the functionalities that the system should offer to the end-users. Functional requirements describe what the system should do and how it should behave, focusing on the actions and operations that user can perform within the applications.

* User Registration and Authentication:

The application allows admin to create user and admin account by providing personal details. It ensures secure login for users and admin with their user name and password.

* User Management with User Types:

The app supports different user types (e.g., admin and user) with specific roles and permissions. Users can manage their accounts including changing password of their account.

* Car Inventory Management:

Admins can add new cars with details like model, registration number, color etc. Admins can modify car details as needed. Users and admin both can view available cars and their details.

* Rental Management:

User can book cars by selecting a vehicle and rental period as per customer’s needs. User can also cancel the rented car if needed.

## 8.2 Non-Functional Requirements

Non-functional requirements define system attributes like performance, usability, and security. Some of them are:

* Performance

The application has fast response times to ensure a smooth user experience. The app is able to handle multiple users without significant performance degradation.

* Reliability

It incorporates backup and recovery mechanisms to prevent data loss. It involves implementing robust data backup and recovery,

* Security

Provides robust access control mechanisms to prevent unauthorized access. It protect sensitive information and ensuring that all transactions are safe and secure.

## 8.3 Technologies Used

* Database: SQL

SQL (Structured Query Language) is a standardized programming language used for managing and manipulating relational databases. SQL used to query, insert, update, and modify data within a database. In our desktop application, SQL utilized to handle the storage, retrieval, and management of data such as user and admin information, car and customer details, and transaction records.

* Coding: Visual Studio 2022

Visual Studio 2022 is the integrated development environment (IDE) used for coding the application, offering tools for code editing, debugging, and performance optimization.

* Diagrams: Draw.io

Draw.io is used for creating system design diagrams, including flowcharts, ER diagrams, and other visual representations of the system architecture.

* Meetings: Discord

Discord is utilized for team communication and meetings, allowing for voice, video, and text collaboration.

* Data Sharing: GitHub

GitHub is used for version control and data sharing, enabling collaborative coding, tracking changes, and managing project files.

# 9. Deliverables

The Fleet-Ease is designed to streamline vehicle reservation processes and enhance customer service within our car rental business.

The deliverables of the proposed Car Rental System are as follows:

* Source Code and Application Development:

Develop the system using VB.NET, leveraging its functionalities and libraries to create a robust and efficient desktop application.

* Database Implementation:

Design and implement a database to store vehicle information, customer details, booking records, and other relevant data necessary for managing car rentals.

* Authentication and Access Control:

Implement an authentication system to ensure secure access, allowing only authorized users to perform booking operations and manage rental information.

* Vehicle Reservation Management System:

Develop a comprehensive system to facilitate vehicle reservations, modifications, cancellations, and viewing of rental bookings with intuitive user interfaces.

* Testing and Quality Assurance:

Conduct thorough testing and debugging of the system to identify and resolve any issues, ensuring the functionality and reliability of the Car Rental System.

* Comprehensive Documentation:

Prepare detailed documentation that outlines the usage and functionality of the Car Rental System, including installation instructions and configuration guidelines.

* Deployment and User Accessibility:

Deploy the finalized application for seamless access by end-users, ensuring it is ready for operational use within our car rental business.

By achieving these deliverables through the development of the Car Rental System using VB.NET, we aim to modernize our vehicle booking processes, improve operational efficiency, and deliver an exceptional experience for our customers and staff alike. This comprehensive solution will contribute to the growth and success of our car rental services in today's competitive market.

# 10. Project Result

Finally, after following the software development lifecycle, and going through testing module we were able to implement the project. Some of the features are still under development and kept under the future enhancement of the project. So, in this section we have the results based on the following:

## 10.1 What are the function achieved by our project?

The desktop app successfully fulfilled its intended function, providing users and admin with a reliable environment to rent the cars.

1. For User

* Through a secure authentication process, users can log in to their accounts to access features, and rent the car.
* Each users have their own username and password.
* Users can easily rent a car, cancel the rented car which are available at that time.
* Each car listing includes essential details such as descriptions, and their availability.
* Users can view the car details, customer details, rented car details and rent the car by fulfilling the given requirements.

1. For Admin

* Admin can log in through a secure authentication process to access their accounts and manage the system.
* Admin can easily add the new cars and their details, and is also able to add new users.
* Admin can view all the transaction, revenue of the individual car and overall.
* Admin receives a reminder when a car requires maintenance.

## 10.2 What problems have been solved by the project?

The application has addresses various inefficiencies and challenges in desktop application. This application has improved data accuracy, enhance overall customer satisfaction by solving issue. Some of them are:

* Elimination of paperwork:

By storing all rental information digitally, the application eliminates the need for handwritten rental agreements, booking forms, and maintenance logs, reducing errors and improving record-keeping accuracy.

* Real-Time update and notification:

The application provides real-time updates and notifications for renting, maintenance schedules, and ensuring timely responses and proactive management of rental operations.

* Efficiency in data entry:

Users can input and retrieve data quickly through digital forms and interfaces, minimizing the time and effort spent on manually writing, organizing, and searching through paper documents

* Accessibility and organization:

Digital storage ensures that all rental-related documents, customer records, and operational data are easily accessible and organized within the application, enhancing overall efficiency and workflow management.

* Human error:

Minimizes errors in data entry, calculations, and documentation that are common with manual processes, ensuring accuracy in rental agreements, pricing, and customer records.

* Security and privacy:

Ensure robust security and privacy measure by implementing individual user accounts with password protection. Only admin have privileged access right to view and manage sensitive data.

## 10.3 What requirements has been met?

The Android app met various essential requirements set at the beginning of the project. It provided a reliable and convenient solution for users seeking a seamless car rental experience.

* Reliable and convenient application for car rent:

The application successfully fulfilled the requirement of providing a reliable and convenient platform for users to rent the car. Users can easily browse a diverse selection of vehicles, make reservations and manage booking effortlessly.

* Vehicles inventory management:

Admin can add the new vehicles with their details. Users can view available vehicles with details (model, reg number, etc.).

* Renting system:

Users can view the available vehicles and can rent vehicles for specified dates. Users can also cancel the rented vehicles.

* Registration and authentication

Admin have capability to add new user and admin to the system and also can remove the existing user.

* User-friendly renting, checkout and cancel process:

Users can effortlessly rent cars o behalf of customers. They can view available vehicles, input rental details such as date, location, car details and finalize bookings seamlessly. Our system ensures a smooth process for renting, checkout and cancel the vehicles.

# 11. Future Enhancement

Future enhancements for the desktop application refer to potential improvements and additional features that can be incorporated to enhance its functionality, usability, and overall user experience. As technology evolves and user needs change, continuous development and improvement are essential to keep the platform relevant and competitive. Here are some possible future enhancements for the desktop application:

* Enhance desktop UI/UX:

Continuously improve the desktop user interface and user experience to ensure ease of use and efficiency.

* Advance reporting tools:

Develop advanced reporting features within the desktop application to analyze rental data, track performance metrics, and generate comprehensive reports.

* Fleet management features:

Expand fleet management capabilities to include maintenance scheduling, mileage tracking, and inventory management directly from the desktop interface.

* Data backup and security:

Enhance data backup procedures and security measures to safeguard sensitive customer and business information stored within the desktop application

* Driver verification and license management:

Introduce features for verifying driver’s licenses and managing driver information securely within the desktop application

# 12. Conclusion

In conclusion, our car rental management system (Fleet-Ease) brings substantial benefits such as streamlined operations, efficient fleet management, reliable customer service, and improved financial control through having feature of individual transaction and overall transaction. Features like real-time update and notification enhance operational efficiency.

To ensure the success of our system, it is crucial to prioritize proactive maintenance of our technological infrastructure. This includes regular updates and robust security measures to maintain system reliability and protect data integrity. Additionally, staying informed about industry developments and technological advancements allows us to continually enhance our system's capabilities and competitiveness in the car rental market.

By prioritizing infrastructure maintenance and staying ahead of industry trends, we can maximize the efficiency, reliability, and profitability of our car rental management system, driving sustained success in the market.

# 13. Annexures

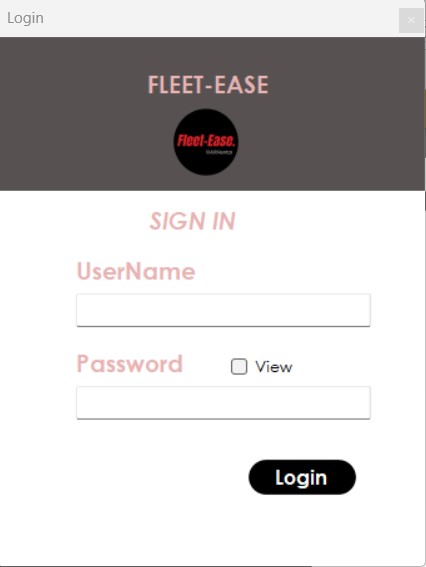


Fig 13.1: Login page

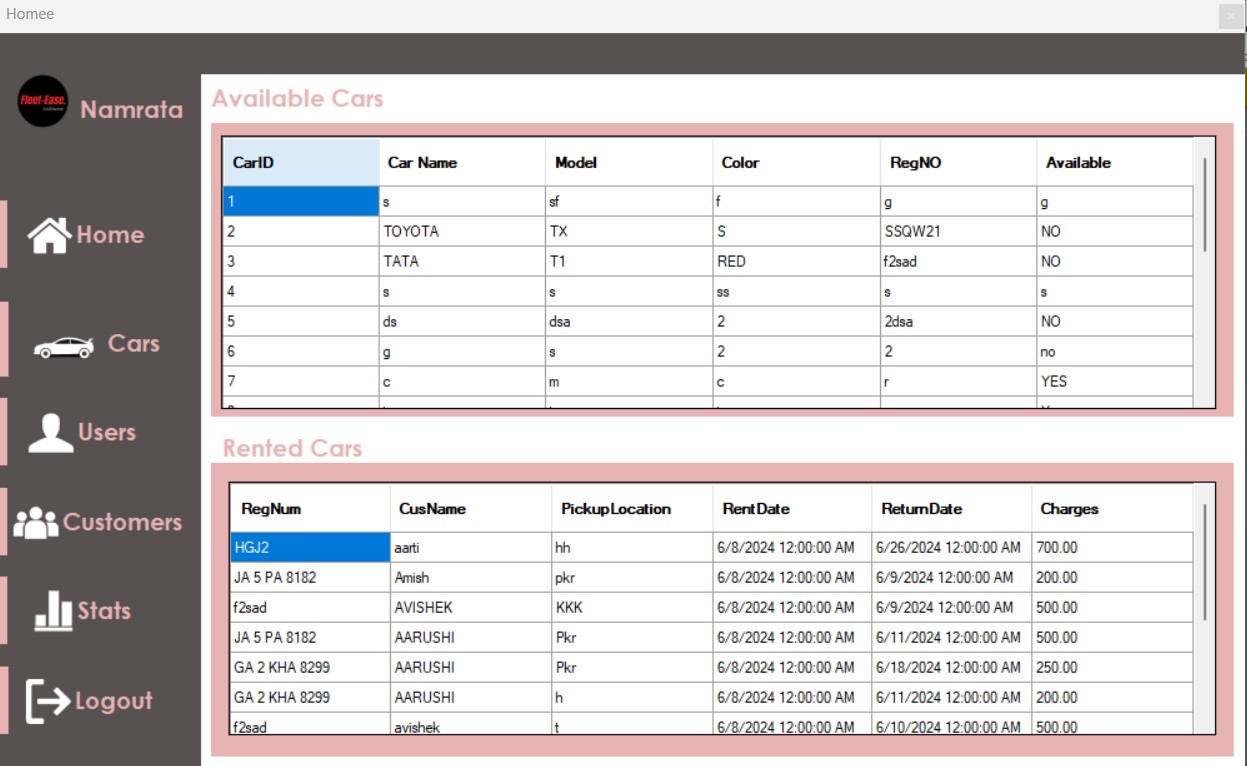


Fig 13.2: Admin’s dashboard

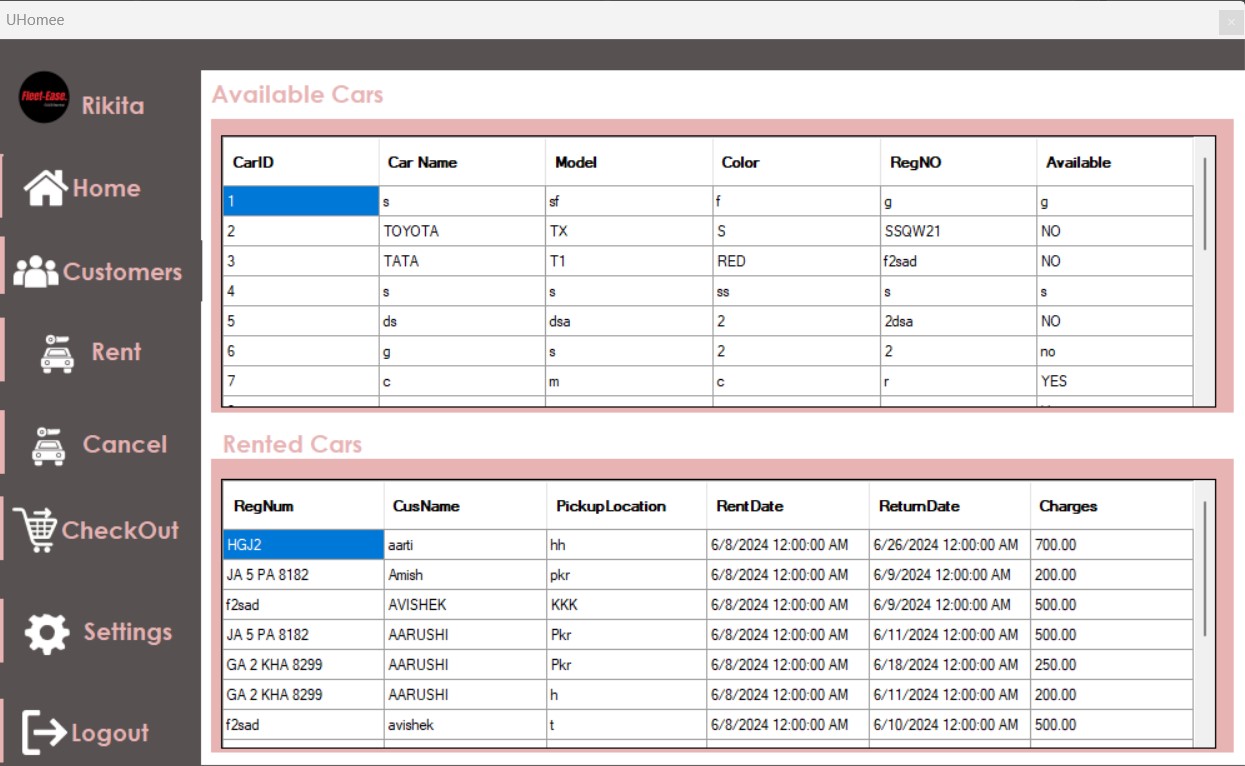


Fig 13.3: User’s dashboard

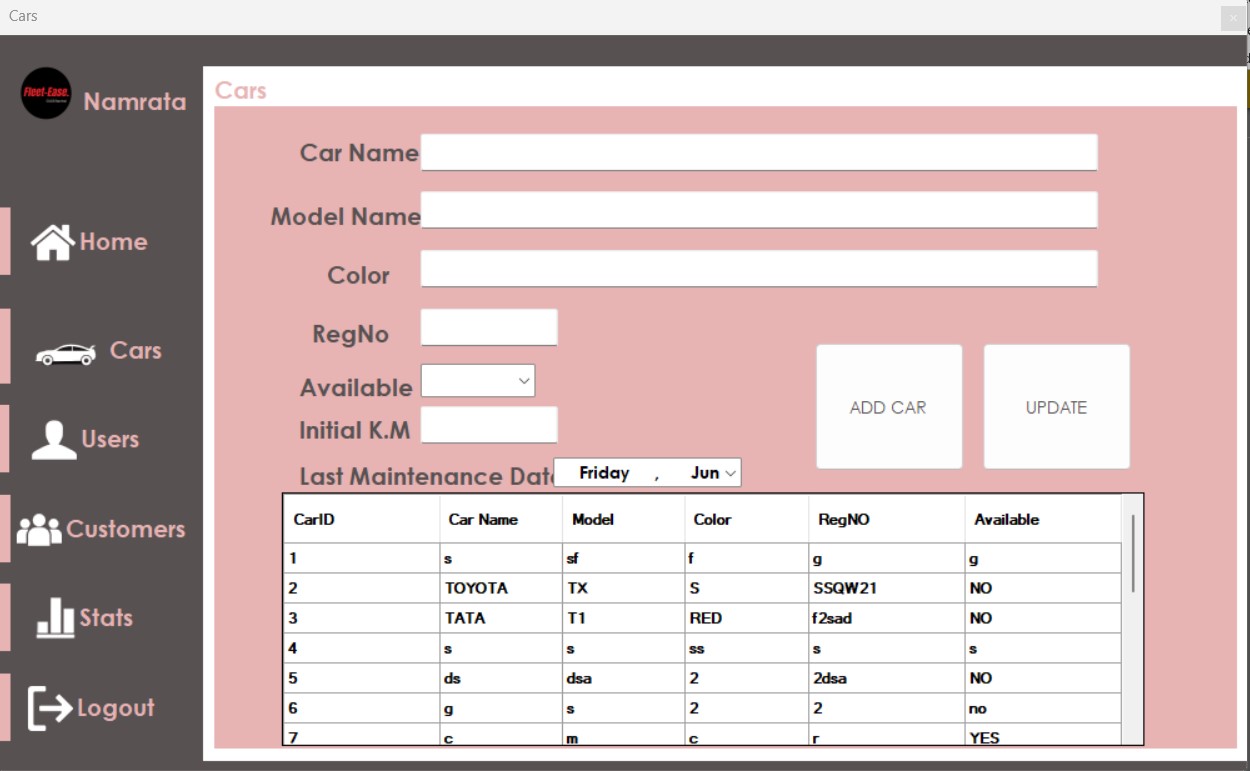


Fig 13.4: Admin category new car add by admin

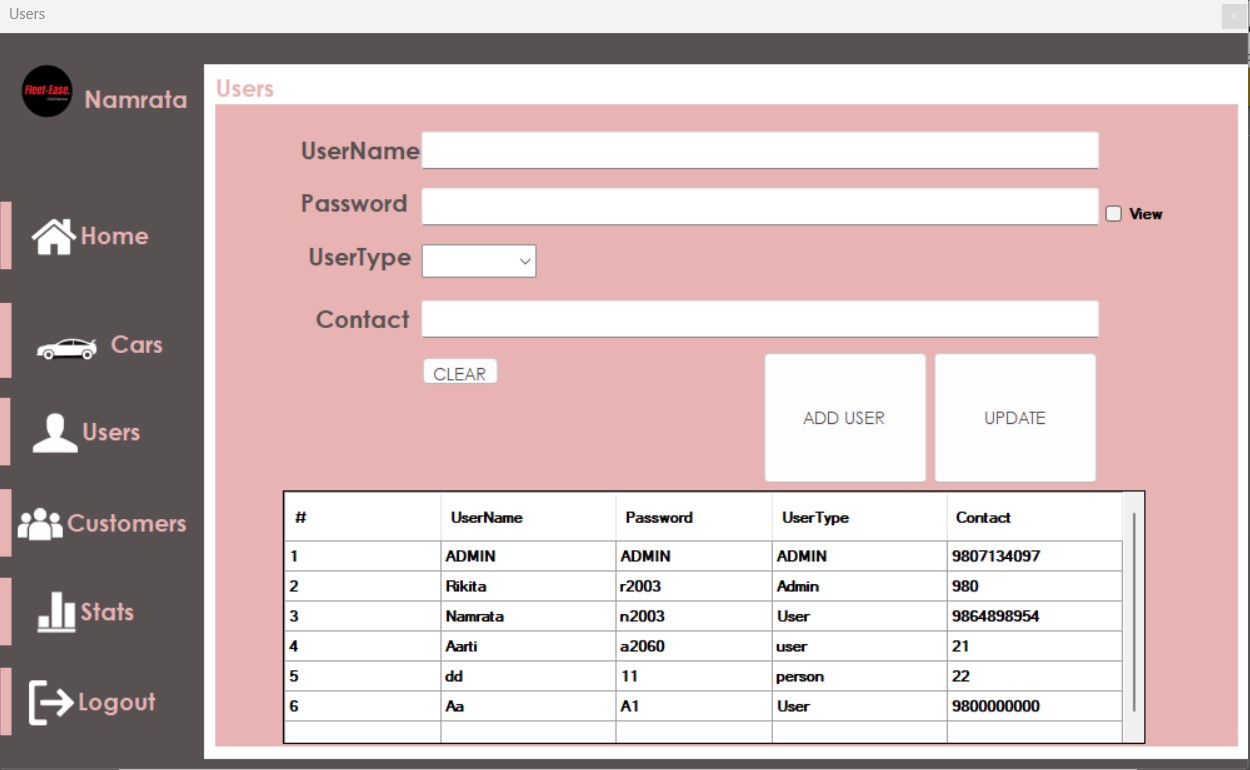


Fig 13.5: Admin category new user add by admin

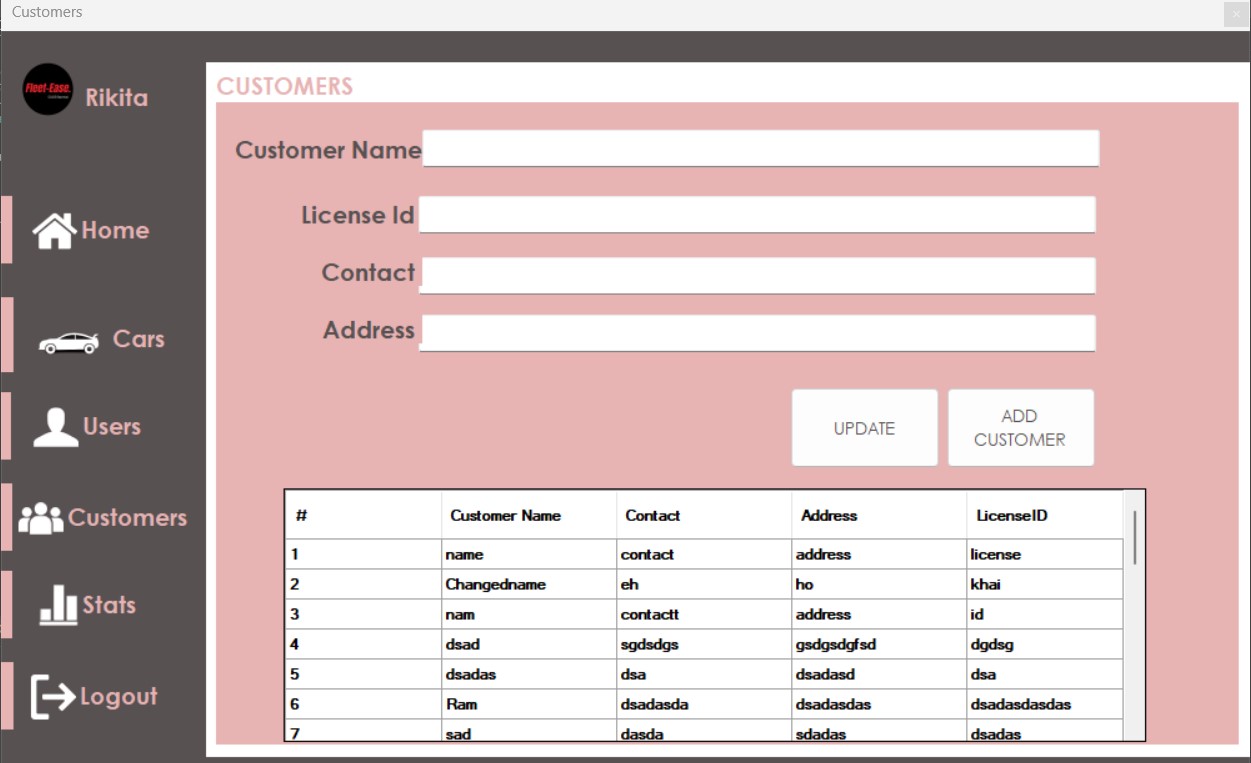


Fig 13.6: Adding new customer details

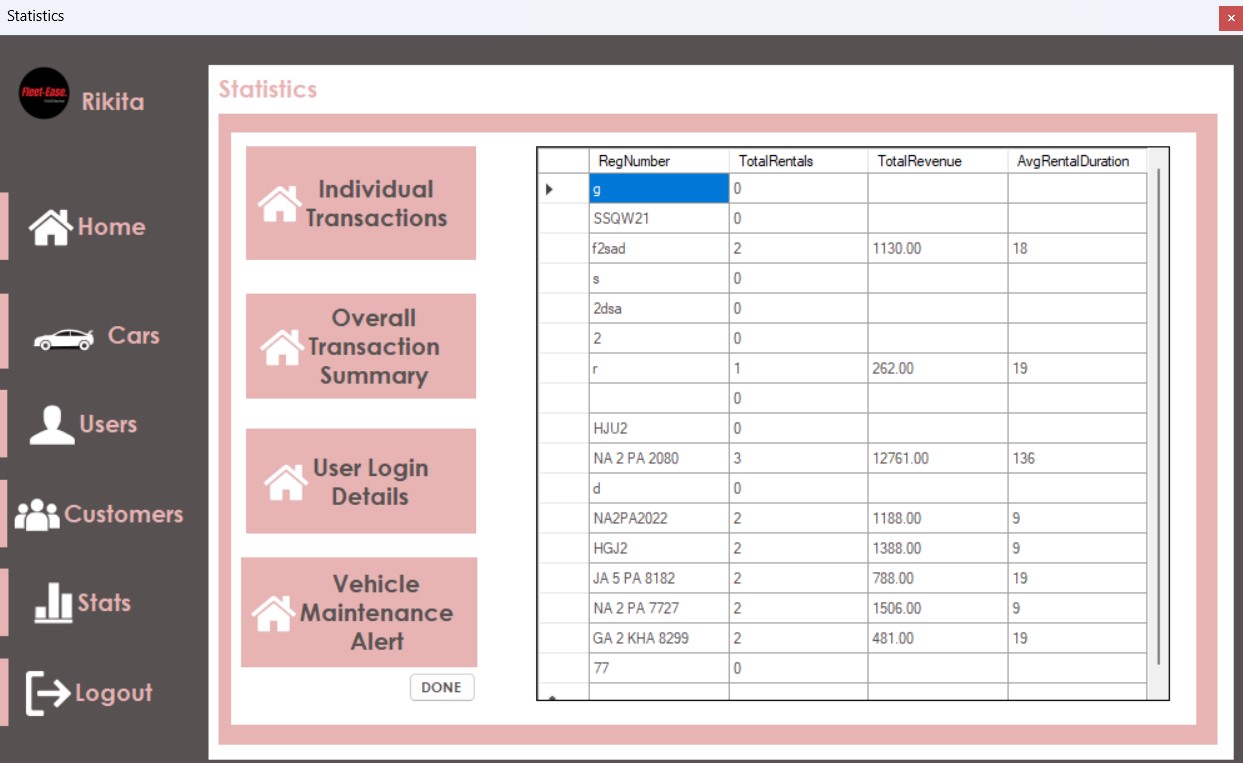


Fig 13.7: Admin category statistics (individual transaction)

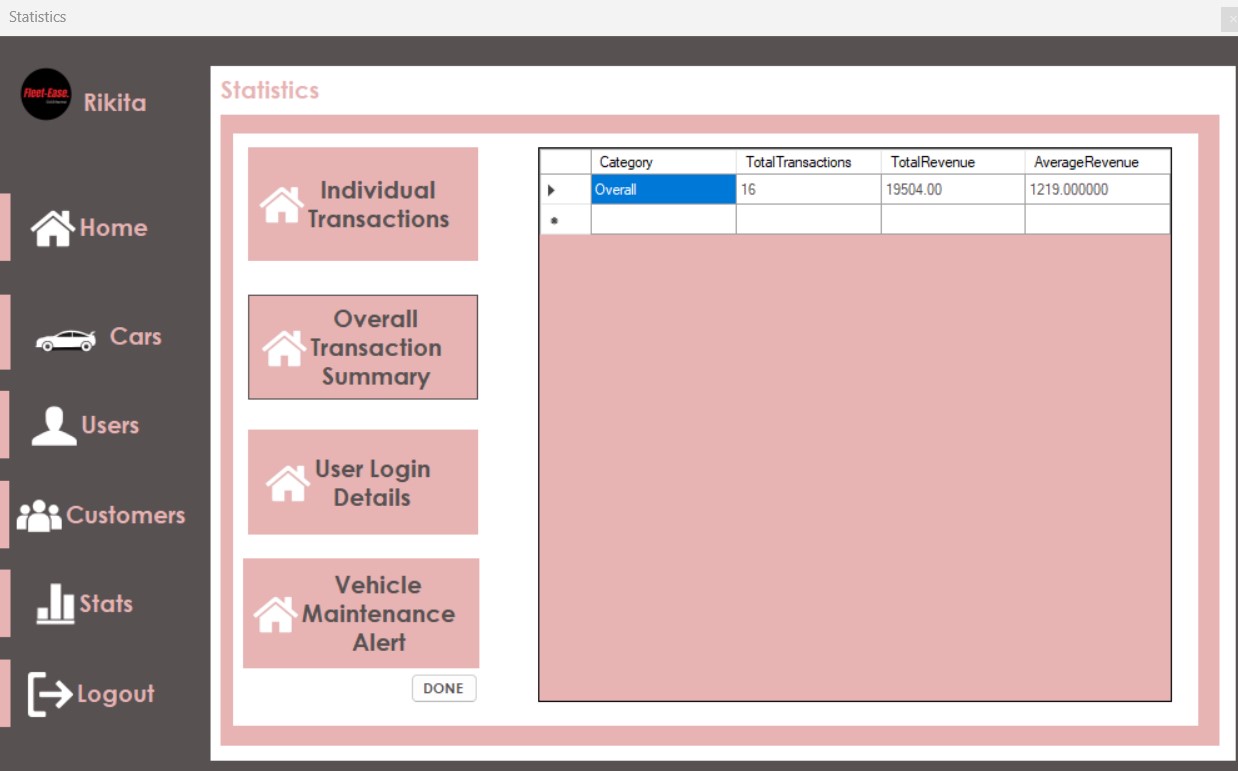


Fig 13.8: Admin category statistics (overall transaction summary)

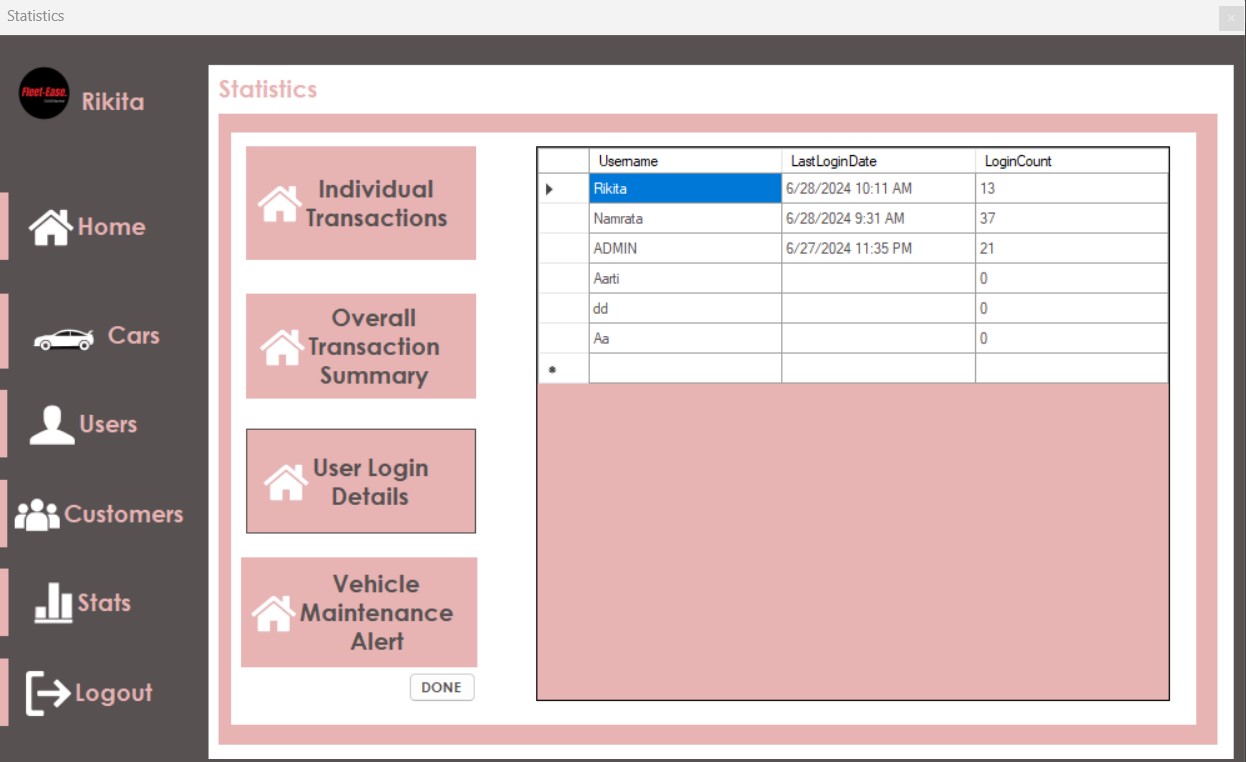


Fig 13.9: Admin category statistics (user login details)

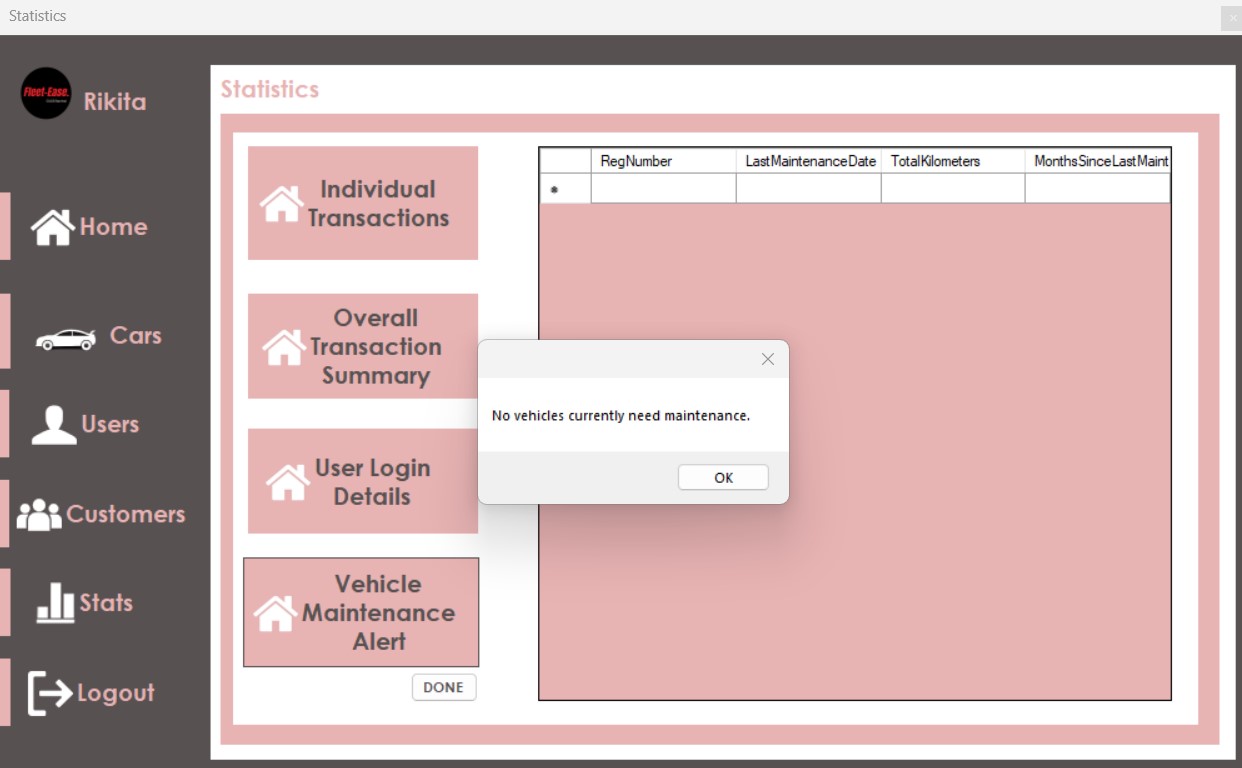


Fig 13.10: Admin category statistics (vehicle maintenance alert)

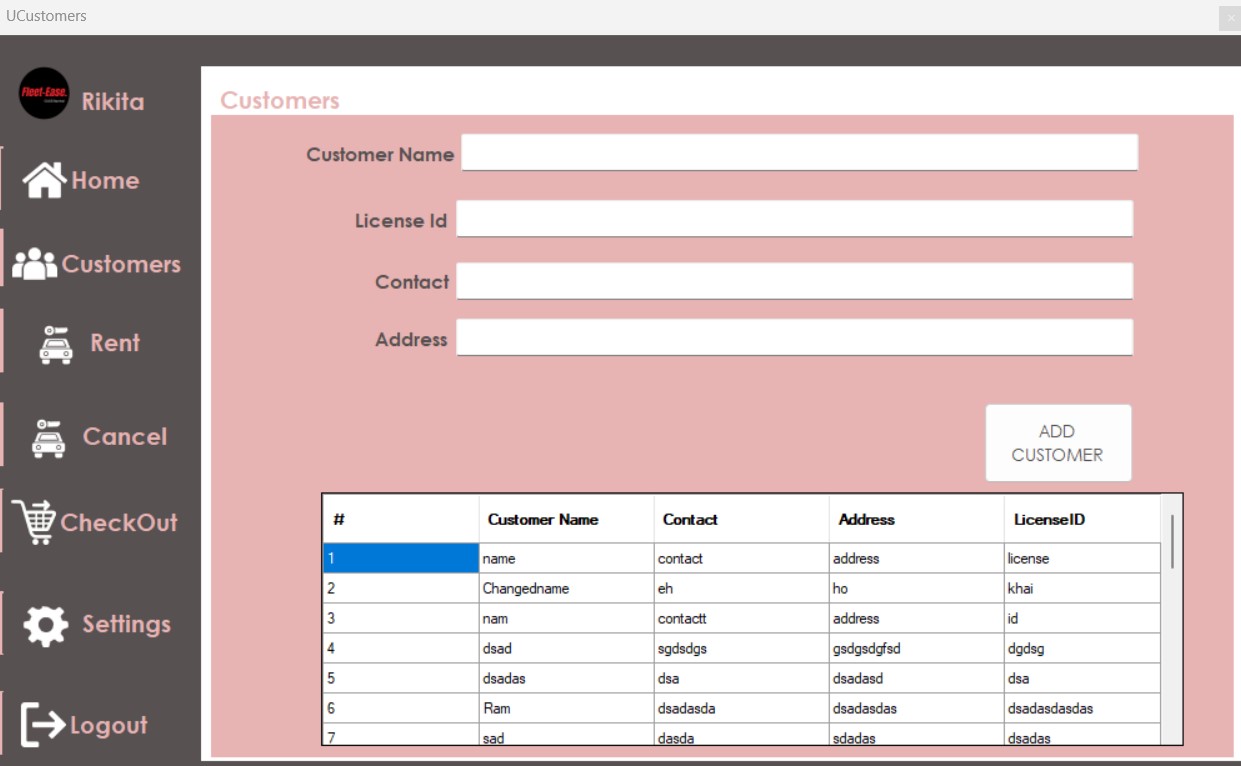


Fig 13.11: Customer category add new customer

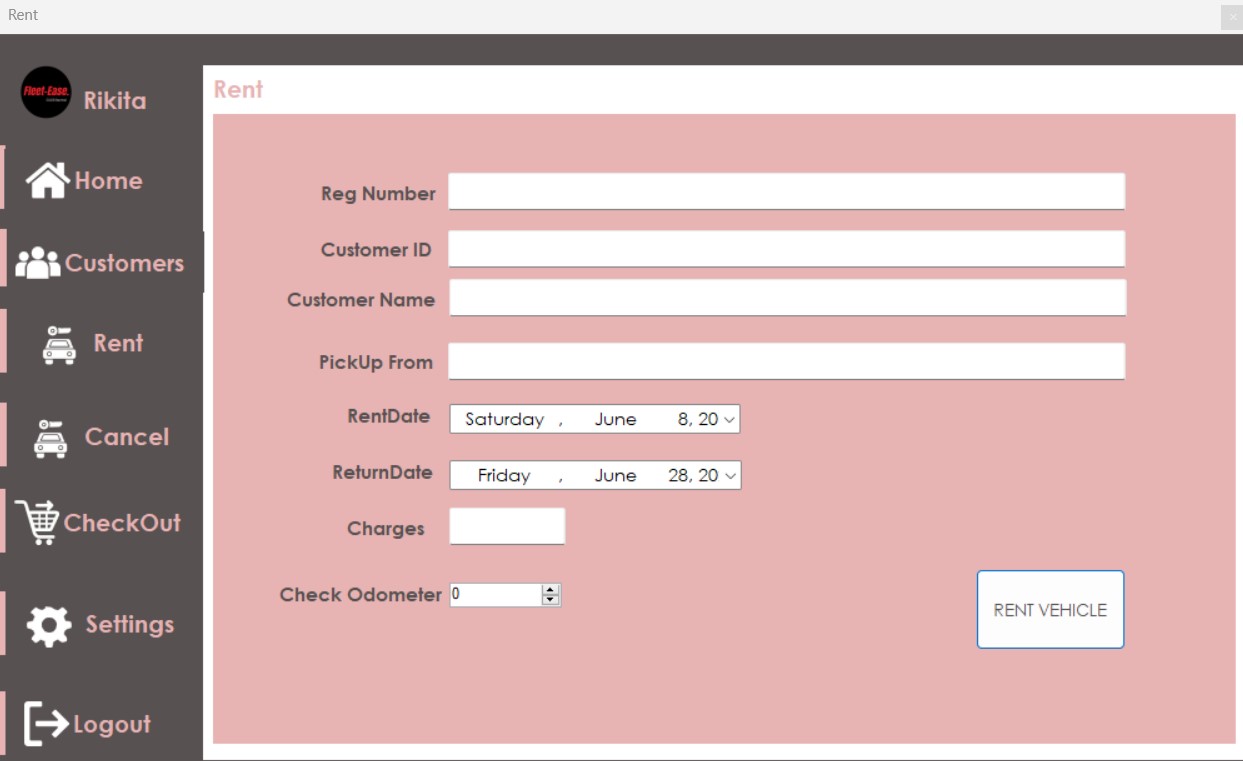


Fig 13.12: Customer category renting vehicles

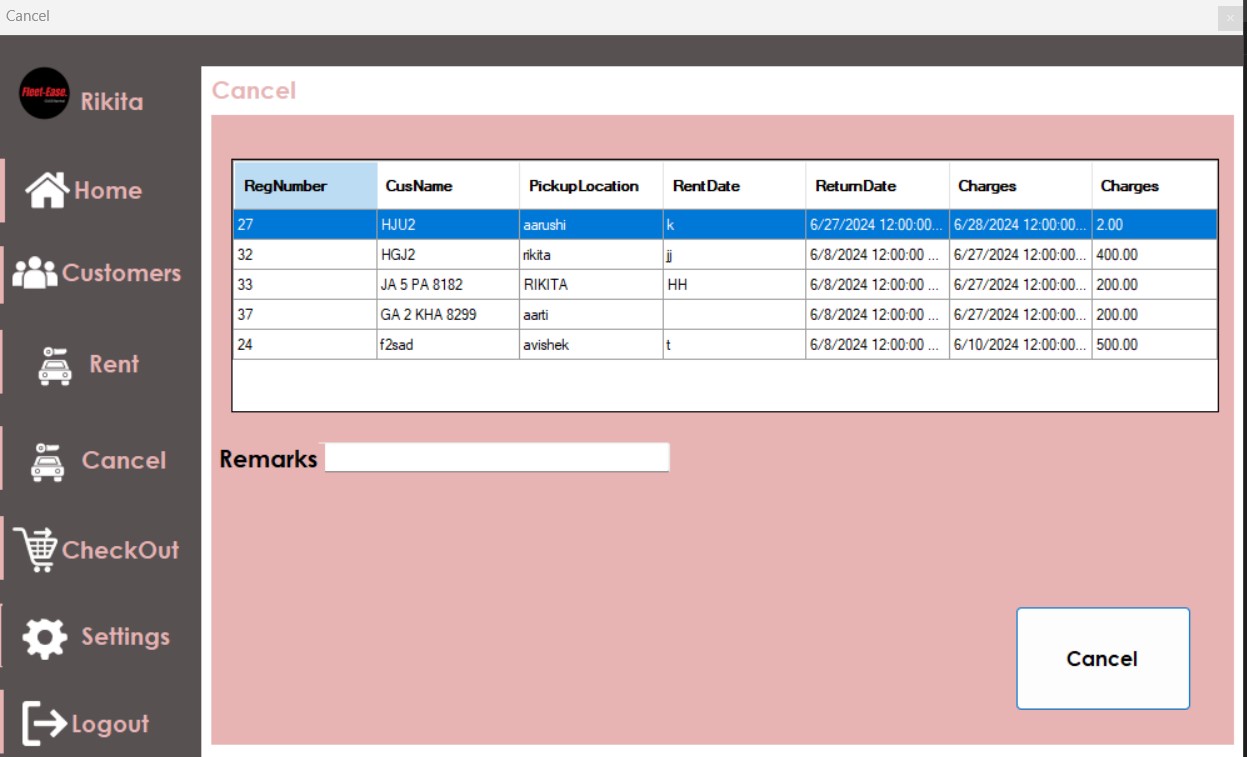


Fig 13.13: customer category cancel the rented vehicles

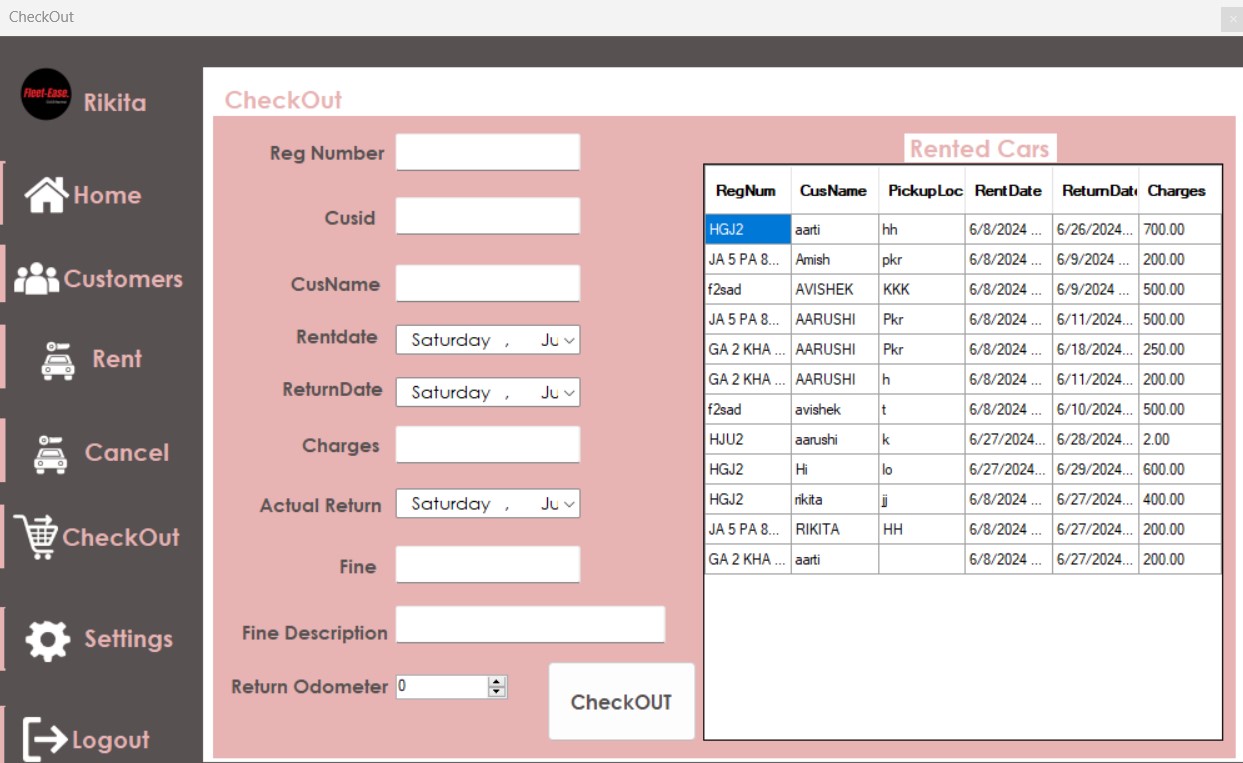


Fig 13.14: customer category checkout the vehicles

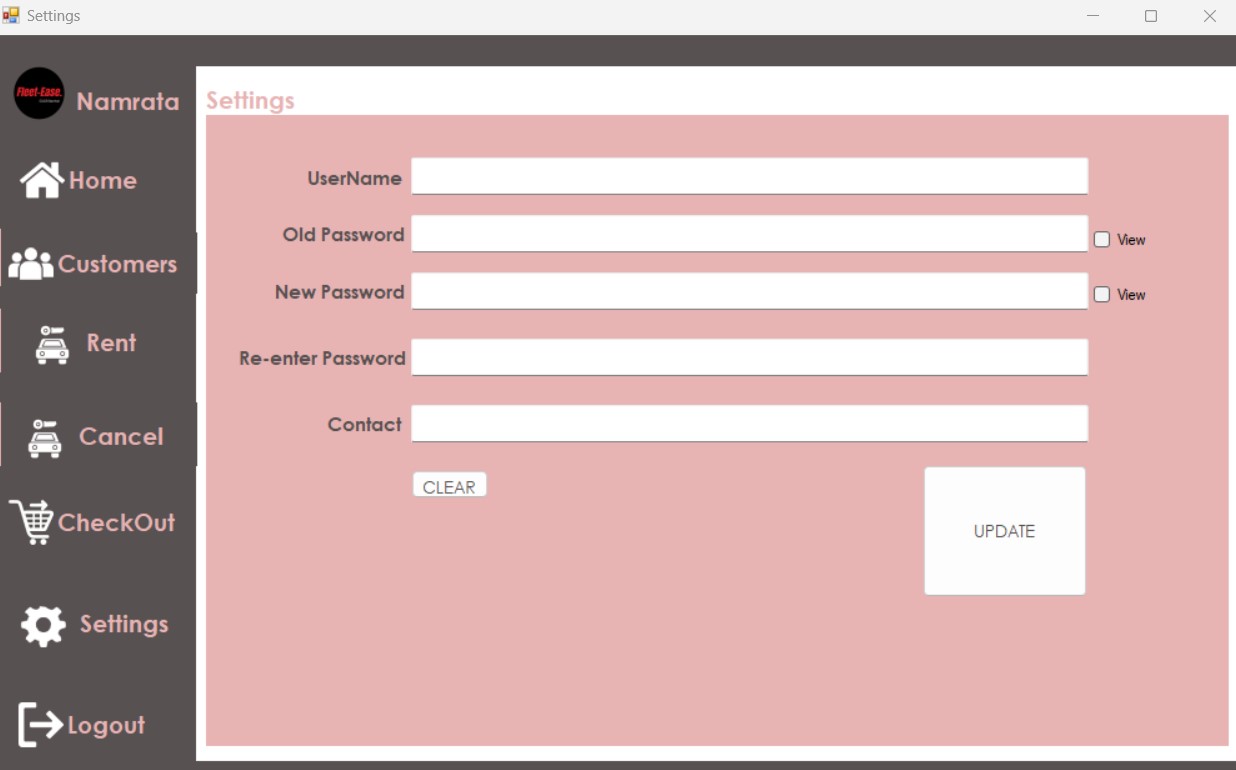


Fig 13.15: Customer category user’s setting

# 14. Project Gantt Chart

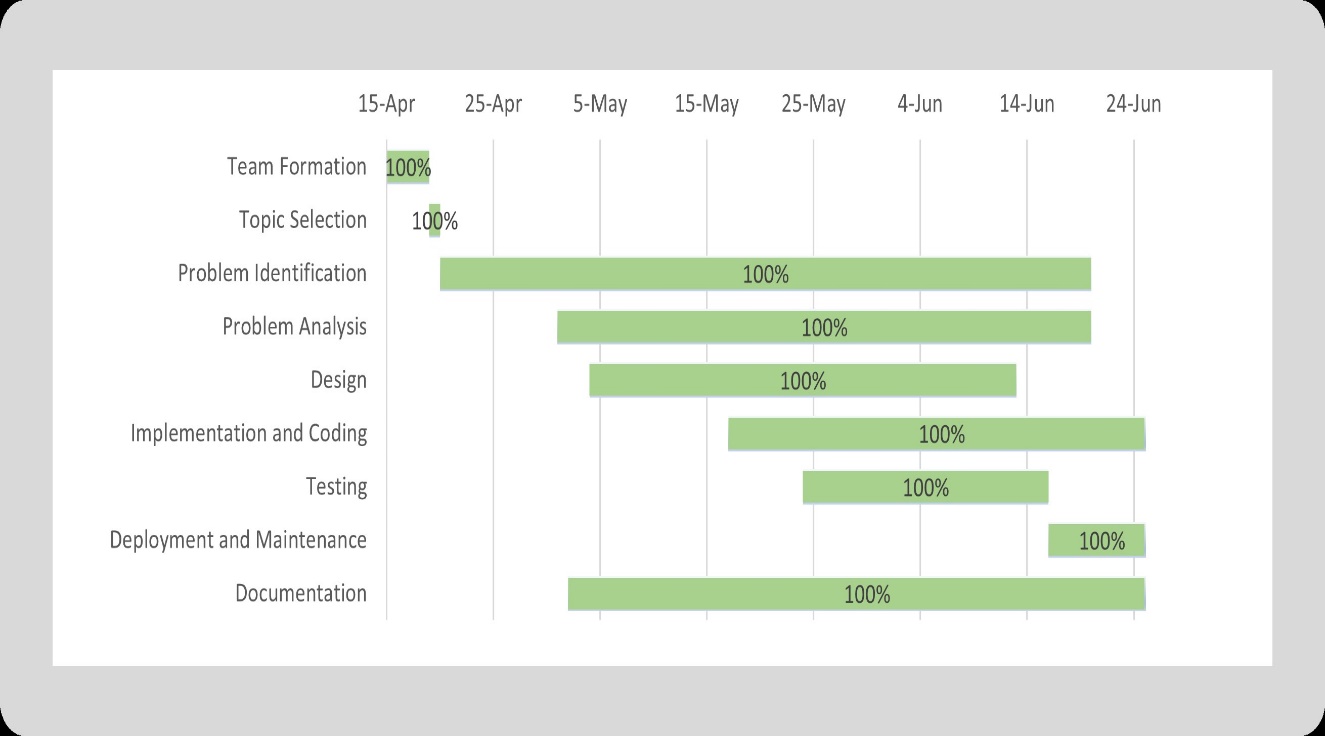


Fig 14.1: Gantt Chart

# 15. References

1. Kundan, C. (2020, November). Guidelines for writing a project report. Retrieved from <https://www.ckundan.com.np/2020/11/guidelines-for-writing-a-project-report.html>,

2. StackOverflow. (n.d.). StackOverflow. Retrieved March 20, 2021, from<https://stackoverflow.com/>

3. Car Hire Nepal. (2020, August 21). Home - Car Hire Nepal safe and trustworthy carrental service in Nepal . <https://carhirenepal.com/>

4. Smile Car Rental- Car rental agency in Kathmandu. <https://nepalvehiclerental.com/>