CAR RENTAL USING MOBILE APPLICATION DEVELOPMENT

BY

MAH GUO XIANG

A REPORT

SUBMITTED TO

Universiti Tunku Abdul Rahman

in partial fulfillment of the requirements

for the degree of

BACHELOR OF INFORMATION TECHNOLOGY (HONOURS)

COMMUNICATIONS AND NETWORKING

Faculty of Information and Communication Technology

(Kampar Campus)

FEBRUARY 2025

COPYRIGHT STATEMENT

This Final Year Project report is submitted in partial fulfillment of the requirements for the degree of Bachelor of Information Technology (Honours) Communications and Networking at Universiti Tunku Abdul Rahman (UTAR). This Final Year Project report represents the work of the author, except where due acknowledgment has been made in the text. No part of this Final Year Project report may be reproduced, stored, or transmitted in any form or by any means, whether electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of the author or UTAR, in accordance with UTAR's Intellectual Property Policy.

ACKNOWLEDGEMENTS

I would like to sincerely thank Dr. Farina Saffa Binti Mohamad Samsamnun for her guidance and support throughout this project. Her advice and encouragement were very helpful in developing this car rental booking mobile app. Thank you for always being there to help and for your valuable feedback. This project would not have been possible without your support.

A special thanks to my friends, who are always there for me when I'm stressed or worried, and who have given me great advice as I worked on my project. Finally, I want to thank my parents and family for their love, support, and constant encouragement throughout my course.

ABSTRACT

This project aims to develop an innovative mobile car rental system that focuses on modernizing the traditional car rental process by integrating real-time tracking features, comprehensive customer support, and an intuitive user interface (UI). The transformation brought by mobile technology to the car rental industry has completely changed the way users interact with rental services. By analyzing existing car rental mobile applications such as GoCar Malaysia, TREVO, and Moovby, this study identifies strengths and weaknesses to inform the development of the proposed system. To address the problems found in these existing systems, the project objectives for the proposed system are discussed. The advantages of the existing systems have been adopted and integrated into the proposed system to enhance its functionality. In the proposed car rental mobile application system, it provides the modules included profile module, booking module, history module, and tracking module. The project uses the Android Studio and Flutter programming languages to develop a mobile-based car rental mobile application system. The choice of a mobile application is driven by the fact that nearly everyone uses mobile devices and to transform the traditional car rental booking process into a modern, more convenient experience. Through this project, our goal is to transform the car rental experience and provide users with a seamless, efficient, and modern rental service platform. This effort not only solves current industry challenges but also sets new standards for convenience, reliability, and customer satisfaction in the car rental sector.

Area of Study : Mobile Application Development, Car Rental System

Keywords : Mobile Car Rental System, User Interface, Android Studio, Real time tracking, Mobile Application.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1
Introduction

1.1 Background Information

The quick growth of connectivity in remote areas through the internet and mobile phone is changing people lives in unexpected ways, bringing a lot of important services such as job skills, economic empowerment, and health care for urban and rural people. The increasing of information and communications technology (ICT) brings more than just economic benefits to us in our daily life. This change has made some traditional companies or individuals to change their business to modern business models by opening online stores (Lazada and Shopee), mobile applications, or any technology that can help to make changing. This digital transformation not only extends the market coverage of these businesses but also meet consumers' growing desire to do business online, marking a major shift in how they make money and interact with global and local consumer.

The develop rapidly of mobile technology has completely changed various industries, such as travel and car rental. A mobile application specifically designs for car rentals that provides unexpected convenience and efficiency, fitting seamlessly into users' daily lives while using the mobile application. The project focuses on utilizing this technological advancement by developing and deploying a mobile application to optimize the car rental process, providing ease of use, convenience, and user friendliness to customers. By providing a mobile application platform where users can easily search available cars, register, manage account and booking vehicles, the application match to the different needs of local and international travellers, recognizing the increasing of user reliance on smartphone devices to complete daily tasks and financial needs.

Furthermore, the success of a mobile car rental system is deeply connected to the satisfaction of its users. A good user experience is built on convenience, simplicity, and efficiency. It should be simple and smooth to allow user to handling complex booking process and using the system interface, no matter their technical skill. To

achieve this, user-centred design plays an important role in creating an interface that is safe, simple, and satisfying for users to use.

1.2 Problem Statement

The existing car rental system faces some important challenges that affect user experience and operational efficiency when users use the system. Without a proper and simple car rental booking system, even if it provides a variety of tools to allow users to book the car or even rent the car, it will be useless. Traditional rental car methods often discover various challenges, such as many processes, limited payment options, and a lack of real-time visibility into the states of the rental car.

1.2.1 Simplifying the Car Rental Booking Process

One of the major challenges is the inherent complexity of the booking process in the system when the customer rents a car. Existing car rental platforms often face complex interfaces that need users to navigate through several pages, fill in extensive personal information, and follow strict booking procedures. This complexity not only causes user impatience but also leads to decreased conversion rate and lost opportunities for car rental companies.

1.2.2 Addressing Limited Payment Options in Car Rental Applications

Many car rental applications offer limited payment options, which negatively impacts the user experience. Traditional payment methods such as cash, credit/debit cards, and bank transfer have been overshadowed by emerging trends in digital payment such as Touch and Go, Grab Pay and Boost. The lack the flexible payment methods fail to meet the diverse preferences of modern consumers, ultimately resulting in missed opportunities for revenue growth and customer satisfaction.

1.2.3 Improving Customer Support Channels in Car Rental Services

Insufficient customer support channels further compound the challenges encountered during the rental process. Users often find it difficult to receive timely assistance when they encounter problems or have questions, resulting in dissatisfaction and negative feedback from users. Poor customer support not only damages a rental

company's reputation, but also damages trust and loyalty among users, ultimately impacting business performance and growth.

1.3 Motivation

The motivation behind developing a car rental booking system is to enhance users' ability to effectively book the car rental. This program is expected to significantly improve user experience and operational efficiency and it is designed to simplify and modernize the car rental process. Through the car rental mobile application, users will have a seamless booking and management platform at their fingertips to rent a car. Additionally, the application will offer alternative payment methods and real-time tracking systems to further improve the car rental experience.

1.4 Research Objectives

1.4.1 To develop a streamlined and user-friendly car rental booking mobile application

The goal is to develop a modern, efficient, and user centric car rental mobile application that completely transforms the booking process and enhances the overall user experience. The first objective is to simplify and modernize the car rental booking process, transforming the traditional booking process to a mobile-first approach. With an intuitive interface and seamless functionality, the application will allow users to easily browse, book, and manage rental vehicles directly from their mobile devices.

1.4.2 To develop timely assistance customer support features for immediate user assistance

To ensure users receive timely assistance, it is essential to enhance the customer support features within the car rental mobile application. Effective customer support enables users to get help promptly when facing problems during the booking process. This includes implementing features such as FAQs, live chat, or help guides within the app for easy user reference. By providing these features, the application enhances the

overall user experience, builds trust with users, and promptly resolves any issues or queries during the car rental booking process.

1.4.3 To develop a multiple choice of payment system in car rental booking system

By offering alternative payment methods in the car rental application such as digital wallets (Touch and Go, Boost, Grab Pay and GX bank.), we aim to promote financial inclusion and meet the changing needs of modern consumers. Additionally, users can choose to continue using traditional payment methods such as cash or bank transfers. However, digital payments have become an integral part of the financial sector, reflecting the ongoing trend of digitizing and facilitating personal and business transactions. This feature not only caters to different user preferences but also accommodates traditional customers.

1.5 Project Scope and Direction

In the car rental booking application, it will provide the features of booking management, account management, history management, user authentication, car management and firebase integration.

I. Booking management

In this module, users can select the booking date, time and location to look for the available car that is available at that time. After the users fill in the booking details, users can start to choose the car that is available at that time.

II. Account management

In this module, users can complete their personal details such as their car license picture (front and back), identity card picture (front and back), and password. Besides, there are some functions that are available in the account module, such as notification setting, display setting, sign out, and delete account.

III. User authentication

In this module, when the user is first time using this application, he or she needs to register by filling their personal information, password and email to register as a user for the app. Furthermore, for those users who have forgotten their password can click

on the forgot password button to reset their password by entering the email address to change the password.

IV. Car management

In this module, as car rental owner users are able to add and remove the car that uploads to the database. Besides, users can store the details about each car such as model, year, registration number, colour, rental price and other information. As a rental, users can check the car information, price, available or not at the time of booking, and the location.

V. Firebase integration

After a user has signed up for an account, their information will update automatically at that time. When the car information is added or removed by the users, the data will automatically update as well. A record of the booking will also be stored in the database as well.

1.6 Contributions

This project's contribution to the development of mobile applications for car rental service is crucial to change how the consumers interact with car rental companies and the way how the companies manage their operations using the technology. By studying the details, the different aspects of the project become more visible, covering technological advancements, user experience enhancements, operational efficiencies, and wider industry impacts. The aim of the project is to use the innovation of technology to solve the traditional problem in the car rental company. By implementation of the real-time car tracking technology in the application it offers many benefits. With this feature, users can monitor the location and status of rental cars directly through their smartphone. Not only giving benefit to users but also to rental companies also, this technology provides real-time vehicle management capabilities that enable the distribution and utilization of cars more efficiently, thereby improving operational efficiency and customer satisfaction. User experience is the major content contributed to this project. With the traditional car rental process user recognizing the trouble associated, the car rental mobile application is designed to be user-friendly with a direct interface that simplifies booking and management of rental cars. This ease of use is expected to make the users easier to rent a car and make car rental services more

accepted by a wider audience. Moreover, by implementing payment options by including digital wallets (Touch and go, Grab Pay and Boost) and other modern payment methods. This feature not only caters to the different preferences of users but also catering to traditional customers. Lastly, with this application car rental companies can collect the data easier from the user feedback to vehicle usage patterns, it provides valuable insights that can inform business decisions from fleet management to targeted marketing strategies.

1.7 Report Organization

In Chapter 2, it provides a literature review that relates to car rental booking applications, reviewing the current existing application and comparing existing applications with the proposed improvements. Chapter 3 discusses the system design, including the system architecture, requirements, project timeline use case diagrams, and activity diagrams for different system modules. In Chapter 4, it focuses on the system design by showing the system block diagrams, provide the system flowcharts for each module in the proposed work, and the entity relationship diagram. In Chapter 5, it explores the initial development stages, including the software setup (Android studio, firebase and flutter), configuration and the development of the application activities such as login, registration, car booking, profile management, and notification. In Chapter 6, it evaluates the proposed system performance, which presents the testing results of the system, discusses the project challenges encountered during the develop stage, and evaluate the achievement of project objectives. In the last chapter 7 it covers the conclusion and future direction.

Chapter 2

Literature Review

2.1 Introduction

The development of mobile technology has had a significant impact on the car rental industry, launching platforms that enhance user experience through convenience, efficiency, and personalization. In this literature review chapter, we will review and analyse the existing car rental mobile applications which with the similar functionalities. The aim of the objective in this chapter is to identify the strengths and weakness in each system and compare then to understand the development of the proposed system. Furthermore, we will also evaluate the functionality and features of each component of the current system. Extracting valuable insights from these evaluations will help improve the proposed system.

2.1 Review on car rental using mobile application development

2.1.1 GoCar Malaysia [1]

Figure 2.1 Home module

Figure 2.2 Subscription module

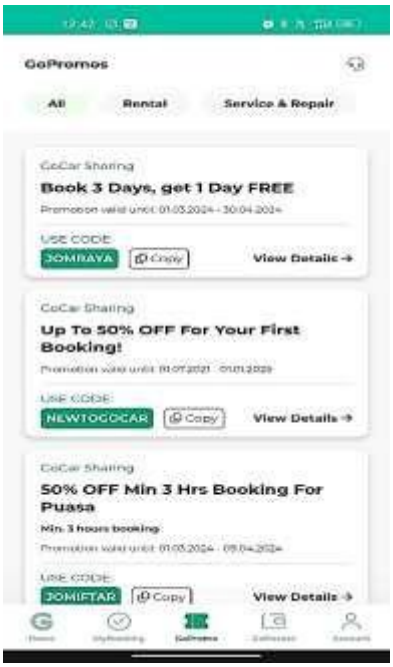


Figure 2.3 Promotion module

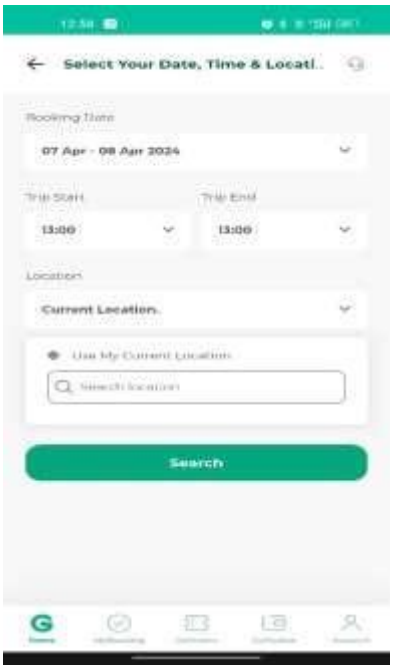


Figure 2.4 Booking module

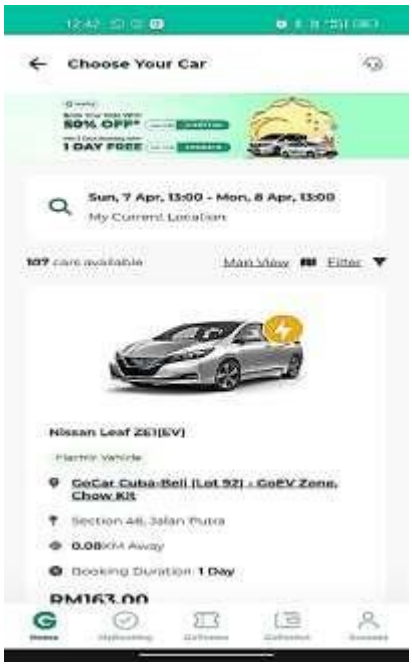


Figure 2.5 Available car module

GoCar Malaysia is developed by GOCAR MOBILITY SDN BHD, is a popular car rental mobile application that offers users a convenient and flexible way to rent vehicles in Malaysia. It available on both Android's Play Store and Apple's App Store. With over 500k downloads and a remarkable rating of 4.3 out of 5 in both platforms. It is clear that many users express high satisfaction with the services and features offered by the GoCar Malaysia mobile application.

GoCar Malaysia provides a wide range of vehicles for rental, which include sedan car, SUVs, and MPVs. This unique feature allows users to make booking, tracking, and payment within a single application. GoCar Malaysia created with a user-friendly interface, this app simplifies the traditional booking process allow users to easily to rental a vehicle by the hour, day or monthly. The app also offers other features such as real-time vehicle tracking, allowing users to monitor the location and status of their rental vehicles. All features and choices are seamlessly merged into one interface, ensuring users enjoy a smooth and effective experience. The app utilizes a local storage-based database, ensuring both data security and accessibility.

GoCar Malaysia provide the wide range of vehicles module and flexible booking option module in the single interface. Users can effortlessly to select any function by tapping on the icons at the middle of the interface. Not only that, user also can scroll down to tapping on the subscription to subscribe the car for long terms purpose. Besides at the bottom part provides a clear navigation bar to allow users to choose for next interface. At the bottom right provides a help icon for users to ask for help when users facing any problem.

In the booking interface, Users can effortlessly to fill in the booking date, time, and location details by just tapping and select the date, time, and location. After the users is fill in the details, it will provide the car that available at that time. In this interface it provides wide range of car to let users to choose. At the middle right of the interface users are allowed to filter the types of vehicles, the seats, and the model of the vehicle. Users are allowed to at the promotion code when in the payment interface. In the payment interface provide multiple payment method to allow user to choose for making the payment.

Furthermore, in the bottom of the interface users are allowed to view their booking list. In this interface users are allowed to switch between rental, and subscription to check the current history and booked history. However, it's worth noting that users need to submit and complete the verify account by submitting the valid document to unlock the rental car function. The exceptional features and user-centric design of GoCar Malaysia have played a significant role in its remarkable user satisfaction and popularity within the mobile application market.

2.1.2 TREVO - Car Sharing Done Right [2]

Figure 2.6 Home module

Figure 2.7 Host module

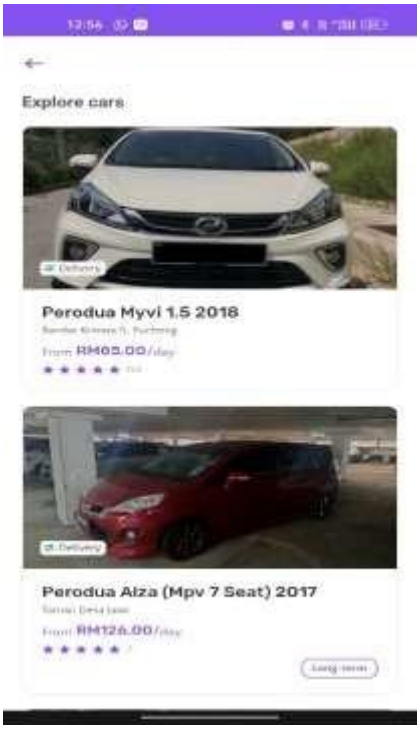


Figure 2.8 Booking module

Figure 2.9 Vehicle Details module

TREVO is developed by Future Mobility Solution SDN BHD. TREVO distinguishes itself in the competitive realm of mobile vehicle rental services by pioneering a community-driven approach to car rental. TREVO is available on both platform which is Android's Play Store and Apple's App Store. TREVO attract over a million downloads and a remarkable rating of 3.8 out of 5 on Play Store and 4.1 out of 5 on App Store. Its focus on flexibility, different kinds of vehicle options, user-friendly interface, and robust support systems position it as a standout player in the field of car sharing.

The main function is conveniently located at the bottom of the interface as a navigation bar to allows user to select, including Home, Booking, Host, and Account. Additionally, the other functions such as TREVO Guard (Renew Insurance) and Buddy Driver (Book a Driver) that less frequently used by users are listed in the top of the interface. This design enhances user experience by sorting icons based on their frequency of use, with commonly used ones prioritized at the front of the list.

In the Booking module, TREVO utilizes a familiar design similar to many other car rental mobile applications that can found in the App Store. Users needed to select the rental date and time before choosing the type of vehicles. Users are allowed to view the vehicles information after choosing the vehicle. In the vehicle information module, user can know the seats available, petrol type, transmission types and other information about the vehicle. This function generally receives positive feedback from users because of the one-stop interface.

In the host module, TREVO allow host to share their car details on the platform to allow the renter for reference and choose. Furthermore, TREVO also allow the host to tracking and monitor the location and status of their rental vehicle directly from their smartphones. Not only that, host also can set their preferred pricing by just tapping on it and select their car features. This thoughtful design allows TREVO users to gain deeper insights into their rental spending patterns and car-sharing habits.

TREVO's combination of a user-friendly interface and insightful car-sharing analysis tools makes it a valuable app for all the users seeking to manage their car rental and sharing activities effectively.

2.1.3 Moovby - Car Sharing [3]

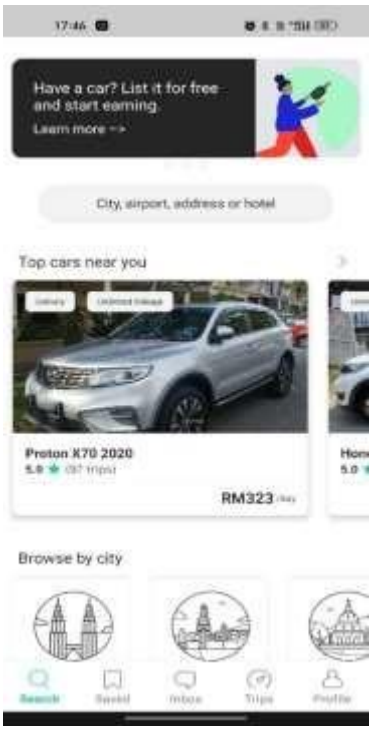


Figure 2.10 Home module

Figure 2.11 Booking module

Figure 2.12 Vehicle module

Moovby is developed by Urban Mobility Asia Sdn Bhd. Moovby's car-sharing platform offers flexible, affordable, and eco-friendly transportation options, showcasing how technology can benefit both users and the community. It available on both Android's Play Store and Apple's App Store. Despite achieving over 100k downloads, it holds a commendable rating of 4.4 out of 5 in App store and 3.9 out of 5 in Play Store, showcasing strong user satisfaction.

The main function is conveniently located at the bottom of the interface as a navigation bar to allows user to select, including Search, Saved, Inbox, Trips, and Profile. Users can directly choose the cars that listed at the Top cars near you when the users are giving the permission to provide the accuracy of their location. With the Browse by city function users are allowed to swap from right to left to browse the city to choose their located city. The function aims to make car rental more accessible, affordable, and convenient through the sharing economy model.

The booking module and vehicle information module in the app are integrated into one interface. Users can choose the start and end for both date and time and location they want to pick up and return the car by just tapping on the interface. After choosing the location, users are allowed to select the available car that list in below. Usera can the amount of one day and the total amount for the selected time period. In the bottom of the interface users can filter the car type, car brand, price range, and transmission type (Auto or Manual). Beside the filters users can open the map for locate another place to check the available car. While it meets the essential needs of a car-sharing system, users have expressed concerns about Moovby's user interface (UI) design, describing it as unexciting and of low quality.

In the bottom of the navigation bar users are allowed to tap on the Saved module to check the saved vehicles that save before, and users can direct click on it to make next rental. In the Inbox module users can see the notification and messages that sent or receive. Besides, users can check their currently and history of rented by just tapping on the trips button at the navigation bar. While Moovby covers essential car-sharing functions, feedback on its UI design indicates there's space for enhancement in crafting a more visually attractive and user-friendly interface.

2.2 Comparison between previous works and proposed works

In below is the existing works (GoCar, Trevo, and Moovby) have been evaluated based on their features. The comparison is made using general functionalities typically sought in car-sharing and rental services. Each system evaluated has its advantages and disadvantages, which are compared with the proposed system in Table 2.1.

Table 2.1 Comparison between previous works and proposed works

	GoCar	Trevo	Moovby	Proposed work
Design of UI	Attractive design	Attractive design	Simple and unattractive design	Simple and attractive design
Booking Process	Online via app or website	Online via app	Online via app	Online via app
User Requirements	Driver's license	Driver's license, registration	Driver's license, registration	Driver's license, registration
Additional Service	Car delivery for some packages	Car delivery options	Not available	Car delivery options
Customer Support	24/7 Support	In-app Support, FAQ	In-app support, direct contact with owners	In-app support, direct contact with owners
Supported Storage	Local Storage	Local Storage	Local Storage	Local Storage / Cloud
Payment Method	E-wallet, Online Banking, Credit/debit Card	Online Banking, Bank in, Credit/debit Card	Online Banking, Bank in, Credit/debit Card	E-wallet, Online Banking, Bank in, Credit/debit Card

Chapter 3

System Methodology

3.1 Scrum – Agile Methodology

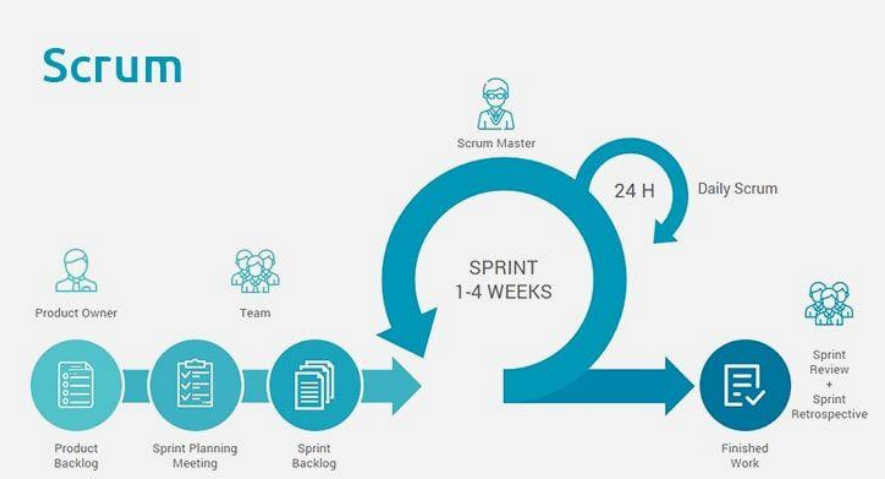


Figure 3.1 Agile methodology (Scrum) (Aziz, 2023)

Scrum, an agile framework main employment in software development. It provides a structured approach to product development that emphasizes on iterative progress, collaboration, and adaptability. By breaking down the complex project into smaller and manageable tasks, it enables to deliver value to customers more quickly and efficiently. Developing a car rental mobile application by using the Scrum framework involves a series of structured steps that are consistent with Scrum's iterative development and feedback-oriented approach. This ensures effective project management through regular progress assessment and adjustments. Below are the stages for implementing Scrum in the development of a car rental mobile application:

1. Product Backlog phase:

- In the initial step it involves gathering all the features, improvement, fixes, and basic requirements for the car rental mobile applications. This may include the functionalities such as user registration, vehicle selection, boking features, payment integration, user feedback, and administrative features for managing the rentals.

2. Sprint Planning phase:

- This event marks the beginning of each sprint by determining what can be delivered during the sprint and how that work will be accomplished. We will select the items from the Product Backlog that can complete during the next sprint, focus on the top priorities outlined. Besides, we discuss and plans the technical approach and break down the work into manageable tasks, and estimates the effort needed for each task, such as creating a car listing or implementing a payment gateway.

3. Daily Scrum Meeting phase:

- In this phase, having a short meeting for promote quick decision making and identify the potential problem that may delay the project. We should provide updates on the progress since the last meeting, what to do for next step and identify any problem facing. This includes addressing technical challenges encountered in implementing features such as GPS integration for car rental mobile apps.

4. Sprint Development phase:

- This is the phase where involves the execution of tasks outlined in the sprint backlog with the goal of building a potentially shippable product increment at the end of the sprint. This includes designing the user interface, coding, performing unit tests, integrating various components (such as linking the user interface with backend services for booking management), and ensuring every module works as expected.

5. Sprint Review phase:

- This occurs at the end of the sprint to evaluate the increment and make necessary adjustments to the product backlog. Showcases the new features created during the sprint, which involve demonstrating how users can select a vehicle, making a booking, and receive confirmation.

6. Product Backlog Refinement phase:

- This process also referred as backlog grooming, this activity is conducting to add details, estimates, and order to items in the Product Backlog. This includes reviewing backlog items to ensure clarity and completeness, breaking down the larger tasks into smaller and more manageable, and adjusting item priorities based on recent feedback and project insights.

3.2 System Requirement

3.2.1 Hardware

Below are the hardware specifications, involving a laptop and a mobile device. The laptop is utilized for coding the system, while the final testing will be conducted on the mobile device.

Table 3.1 Specifications of laptop

Description	Specifications
Model	HP Pavilion - 15-cs3134tx
Processor	Intel(R) Core(TM) i5-1035G1
Operating System	Window 10
Graphic	NVIDIA GeForce MX250
Memory	12GB DDR4-2666 SDRAM
Storage	512 GB PCIe® NVMe™ M.2 SSD

Table 3.2 Specifications of mobile device

Description	Specifications
Model	OPPO RENO 4 PRO
Processor	Qualcomm SM7125 Snapdragon 720G (8 nm)
Operating System	Android 12
UI version	ColorOS 12.1
Memory	8GB RAM
Storage	256GB ROM

3.2.2 Software

Android studio:

Android Studio is the free Integrated Development Environment (IDE) for developing android apps that allows users to build, test, and debug mobile applications. It provides a smart code editor with code completion and refactor; it allows developers to design the UI based on the visual layout editor. Android studio can offer various features that enhance users' productivity when users develop a mobile application due to the powerful code editor and developer tools from the IntelliJ IDEA. Besides, android studio provides a built-in for Google Cloud Platform that allows users to easily interact with Google Cloud Messaging and App Engine. In the proposed system, Android Studio is chosen because of the need to merge multiple Framework and used to run on the Android phone emulators.

Firebase:

Firebase is not a database, but it is backend-as-a-service (Baas) that include various type tools and services that allow users to save data such as real-time Database and Cloud Firestone. Both of real time and cloud are NoSQL databases allow to apply into different applications needs and uses. Realtime Database storing the data as JSON objects, allowing users synchronization of the data across connected devices via real-time. On the other hand, Cloud Firestone uses the document-based approach to store data. Because of this feature, it had been selected to be used in developing the car rental booking mobile application.

3.3 Project Milestone

3.3.1 Project timeline (FYP 1)

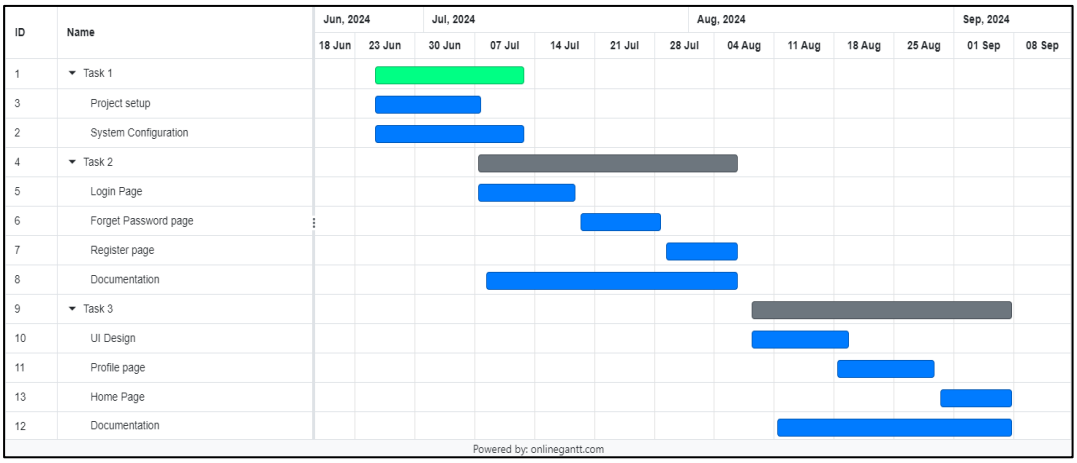


Figure 3.2 Project timeline FYP 1

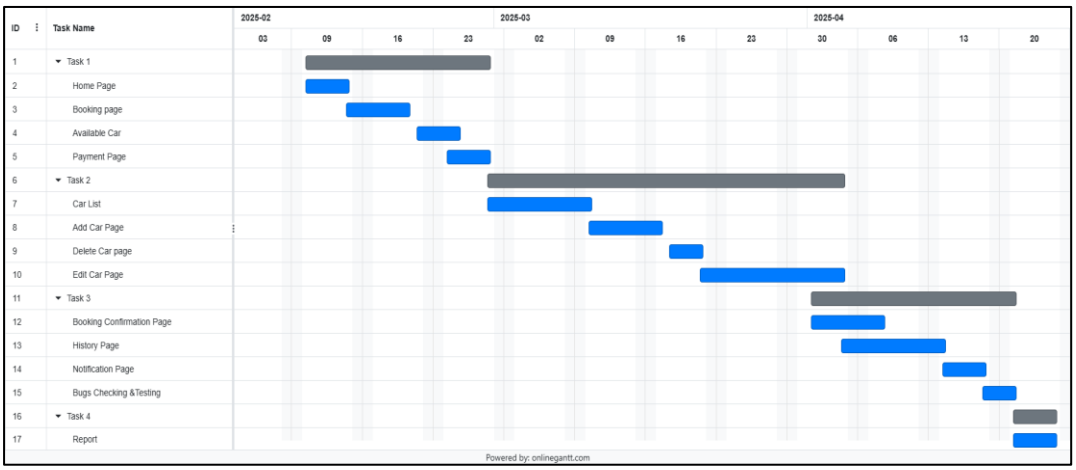


Figure 3.3 Project timeline FYP 2

3.4 System Architecture Diagram

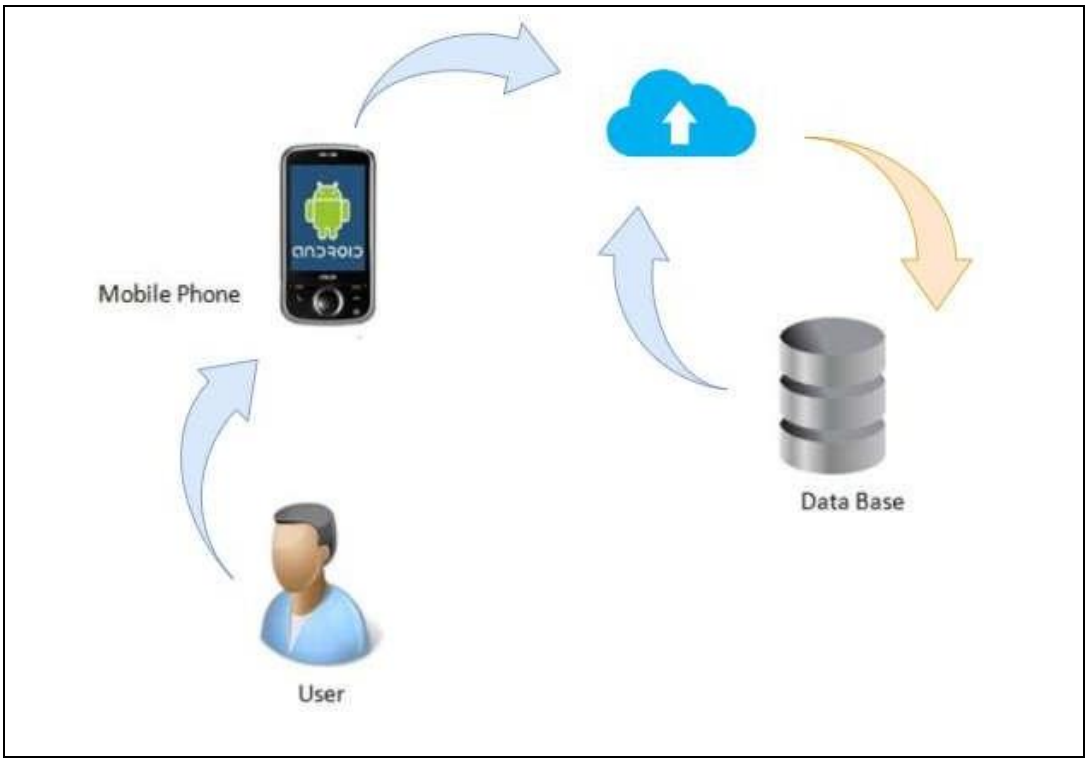


Figure 3.4 System Architecture Diagram

The flow of the user input date in the mobile applications, when the user accesses the app on an Android mobile device. The user selects the booking details, such as the date, time, locations and confirms the booking. The data is then transmitted from the mobile device to the Firebase database via the internet. Once the data is uploaded to the database it will filter and update the available car via the internet to the mobile device. Once the user confirms the booking and uploaded and stored in Firestone, the database will be updated with the new booking transaction. The firestone will send back a confirmation notification to the user's mobile device. In the mobile app it will update the transaction history with the latest booking details, allowing the user to view and manage their bookings directly from the mobile device. This process ensures that the booking data is synchronized between the user's mobile device and the cloud (Firebase database), provides real-time updates and access to the most current information in the mobile app.

3.5 Use Case Diagram and Description

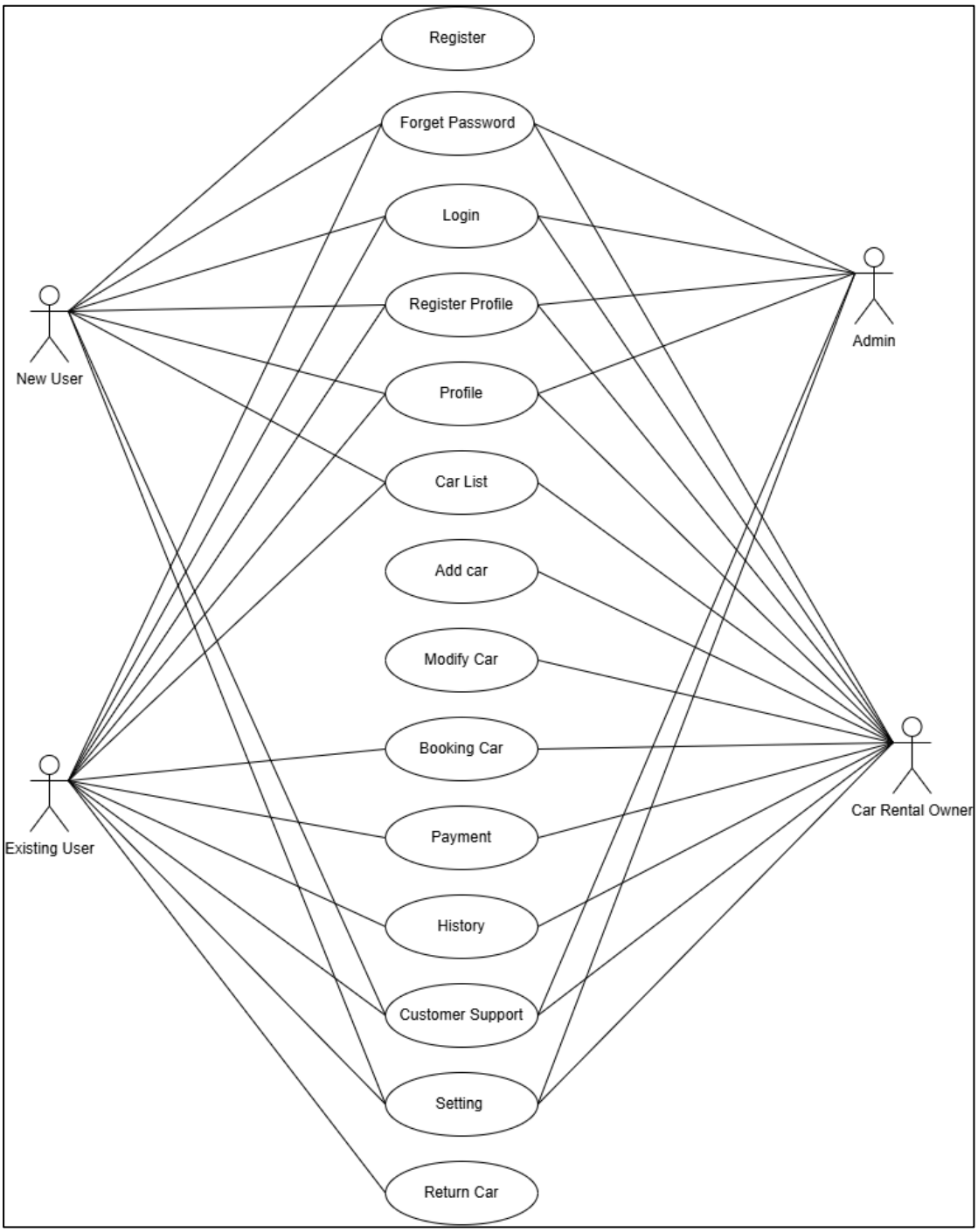


Figure 3.5 Use Case Diagram

Table 3.3 Register Use Case Description

Use Case	Register
Aim	Register an account if the user is new user and did not have an account.
Actor	New User
Trigger	When user click on the Sign-Up button.
Main Flow	Users submit the valid Email & Password. The Email & Password store in database. Browse user back to Login page when successfully registered. Use case end.
Alternative Flow	System prompt error message if the account had existed. System prompt error message if the password length is not including the at least one Upper case, Lower case, Number, and Special character and no longer than 6 lengths. System prompt error message if the password and confirm password is not same.

Table 3.4 Login Use Case Description

Use Case	Login
Aim	Enter the registered email and password to login into the system
Actor	Existing User
Trigger	When user click Login button at main page.
Main Flow	User submits the correct registered email and password. Browse user to home page once authenticated. Use case end.
Alternative Flow	System will prompt error message if email and password is invalid.

Table 3.5 Forget Password Use Case Description

Use Case	Forget Password
Aim	Reset password when forgotten
Actor	Existing User
Trigger	When user click the forget password button.
Main Flow	Users enter the registered email. System sends the reset password link to email. User reset the password. Use case end.
Alternative Flow	System prompt error message if the email is wrong and not registered. System prompt error message if the email field is empty.

Table 3.6 Profile Use Case Description

Use Case	Profile
Aim	Able to review user details and register profile.
Actor	Existing user
Trigger	When user click Profile button on navigation menu.
Main Flow	User able to check their information details.
Alternative Flow	Error message of cannot load the details if the user had not registered their profile.

Table 3.7 Register Profile Use Case Description

Use Case	Register Profile
Aim	Able to update user's full profile details in to database.
Actor	Existing user
Trigger	When user click update profile button.
Main Flow	User able to update their personal information. User able to upload their license and IC image. User able to update to become car rental owner.
Alternative Flow	Error message if email is not same as the registered email. Error message of invalid Name if the name includes except character. Error message of invalid Phone number if the phone number includes except number. Error message of invalid Email if the email is not same as registered email. Error message of invalid IC number if the IC number does not follow the format xxxxxx-xx-xxxx.

Table 3.8 Car List Use Case Description

Use Case	Car List
Aim	Able to review existing car list and car details.
Actor	Existing User
Trigger	When user click Car List button on navigation menu.
Main Flow	User able to review the car. User able to review the car details. User (car rental owner) able to upload the car for rent.
Alternative Flow	Error message prompt if cannot load the car from database. Error message prompt if cannot load the car details from database.

Table 3.9 Add Car Use Case Description

Use Case	Add Car
Aim	To add a new car into database.
Actor	Car rental owner user
Trigger	When user click upload button.
Main Flow	User able to upload the car image. User able to insert car details. User able to insert the amount of car. User able to select the type of car. User able to select the extra feature of car. Use case end.
Alternative Flow	Error message prompt if did not fill in correctly. Error message prompt if did not upload an image. Error message prompt if the insert field is empty.

Table 3.10 Modify Car Use Case Description

Use Case	Modify Car
Aim	To update the car details.
Actor	Car rental owner user
Trigger	When user click modify button.
Main Flow	User able to update the car details. User able to update the car image.
Alternative Flow	None

Table 3.11 Booking Car Use Case Description

Use Case	Booking Car
Aim	Allow user to booking an available car.
Actor	Existing User
Trigger	When user click search button.
Main Flow	User able to select the start date of booking. User able to select the end date of booking. User able to select the Pick-up time of booking. User able to select the Return time of booking. User able to select the location of booking. Use case end.
Alternative Flow	Error message of did not select the start date of booking. Error message of did not select the end date of booking. Error message of did not select the pick-up time of booking. Error message of did not select the return time of booking. Error message of did not select the location of booking.

Table 3.12 Payment Use Case Description

Use Case	Payment
Aim	Allow user to pay for the booking.
Actor	Existing User
Trigger	When user selected the car.
Main Flow	User able to select payment method to pay. User able to see the booking details.
Alternative Flow	

Table 3.13 History Use Case Description

Use Case	History
Aim	Allow user to see the booking details.
Actor	Existing User
Trigger	When user select history button on navigation menu.
Main Flow	User able to check the booking details. User able to check the booking status. User able to click the pickup button to pick up the car. User able to return the car by click the return car button to fill in the return information. User able to upload the payment slip, if the user is using the bank in method.
Alternative Flow	Error message prompt if user did not upload the payment slip on time. Error message prompt if user did not upload the car condition image. Error message prompt if user did not fill in the return mileage.

Table 3.14 Customer Support Use Case Description

Use Case	Customer Support
Aim	Allow user to get basic support.
Actor	Existing User
Trigger	When select support button.
Main Flow	User able to call to get customer support on time by click the hotline number. User able to give feedback by email after clicking the email address.
Alternative Flow	

Table 3.15 Return Car Use Case Description

Use Case	Return Car
Aim	Allow user to update the car condition when return the car.
Actor	Existing User
Trigger	When user click return car on history module.
Main Flow	User able to update the return car mileage. User able to send the car condition image when return.
Alternative Flow	Error message prompt when user did not upload the car condition image. Error message prompt when user did not key in the return mileage.

Table 3.16 Setting Use Case Description

Use Case	Setting
Aim	Allow users to control the system.
Actor	Existing User
Trigger	When users click setting in navigation menu on home page.
Main Flow	Users are able to delete the account by clicking the “Delete Account” button on the setting page. Users are able to reset the account password by clicking the “Forget Password” button on the setting page. Users are able to logout the account by clicking the “Logout” button on the setting page. Users are able to get the customer support by clicking the “Support” button on the setting page. Users are able to Edit the Car List by clicking the “Edit Car List” button on the setting page if they are car owner.
Alternative Flow	

3.6 Activity Diagram

3.6.1 User Authentication

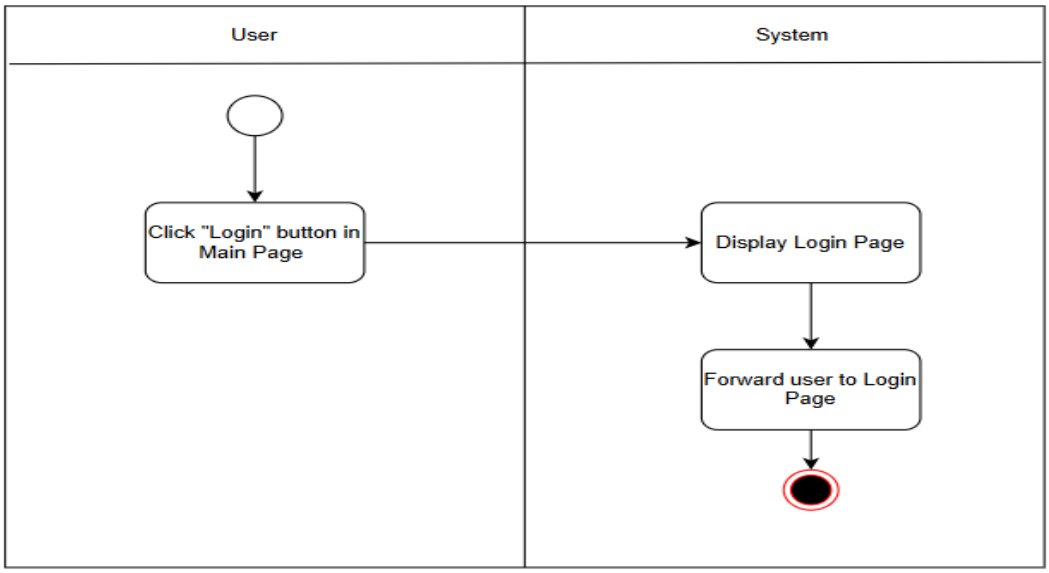


Figure 3.6 Main Page Activity Diagram

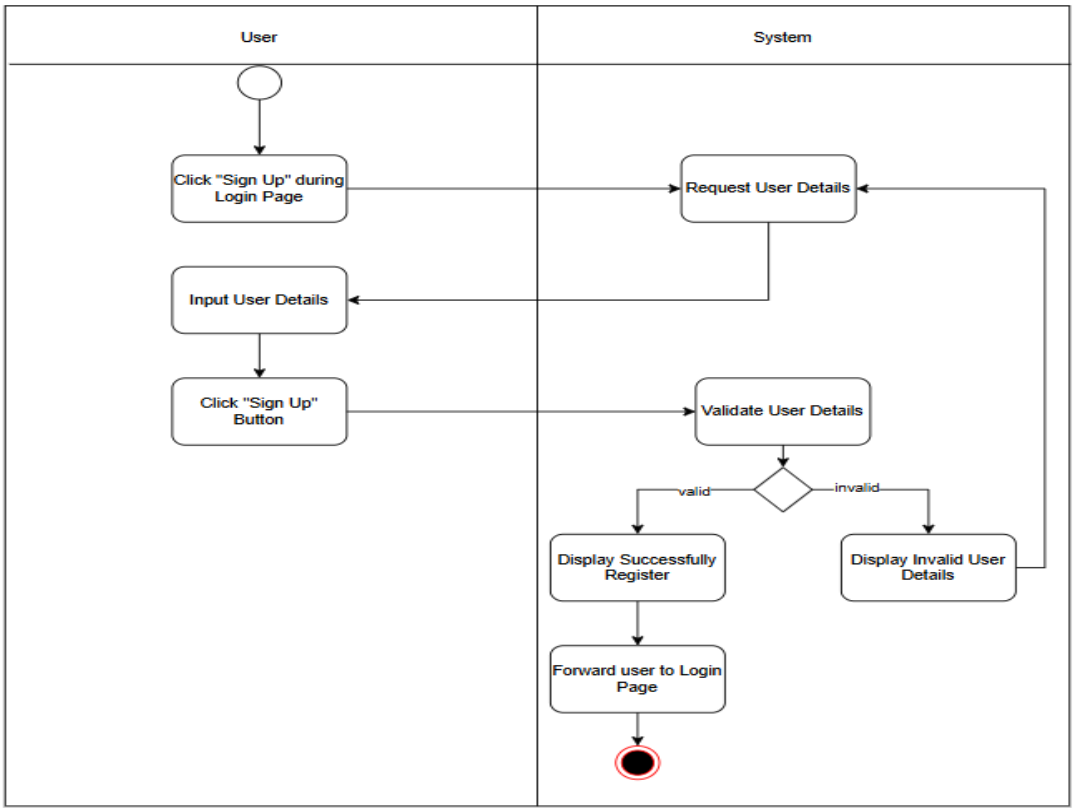


Figure 3.7 Sign-Up Activity Diagram

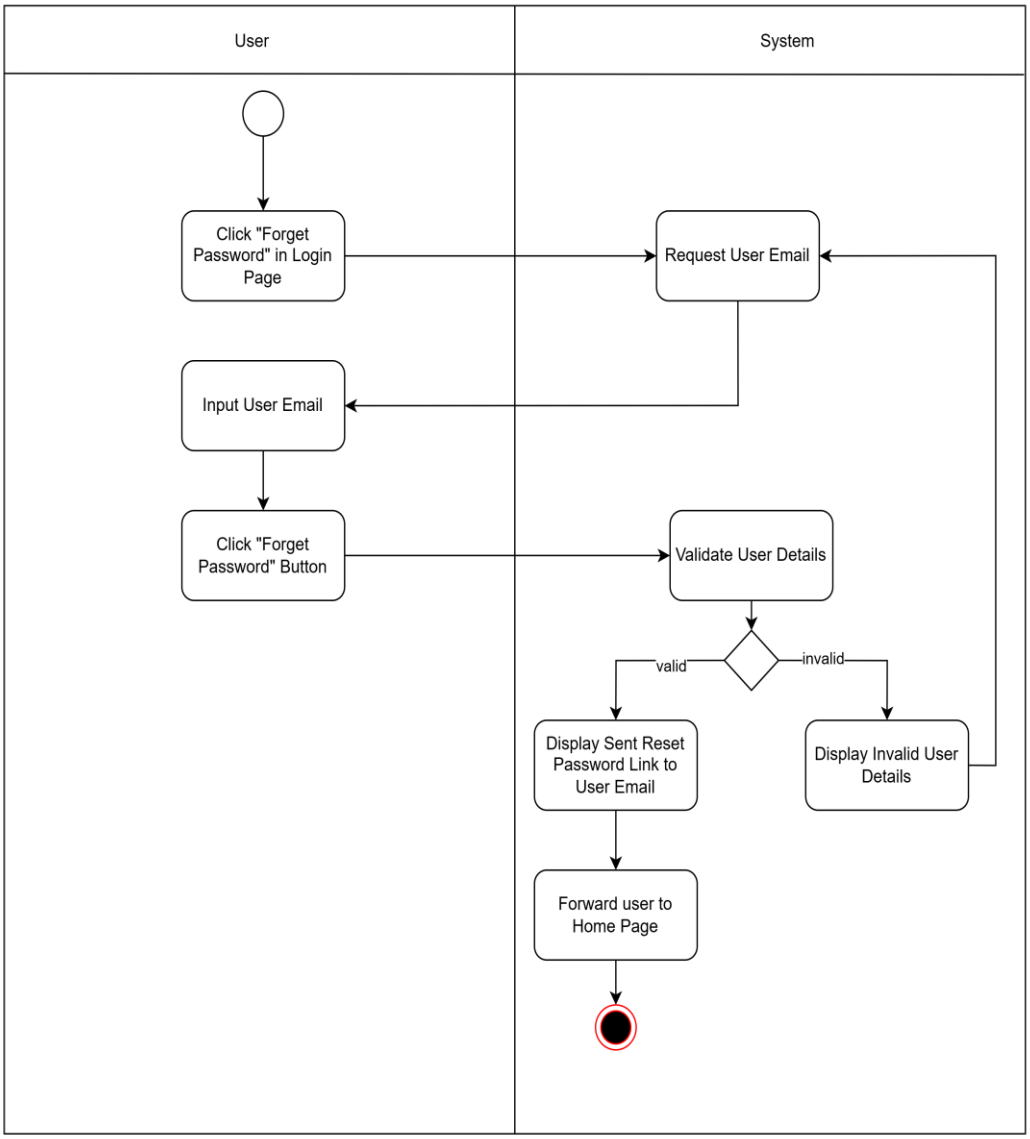


Figure 3.8 Forgot Password Activity Diagram

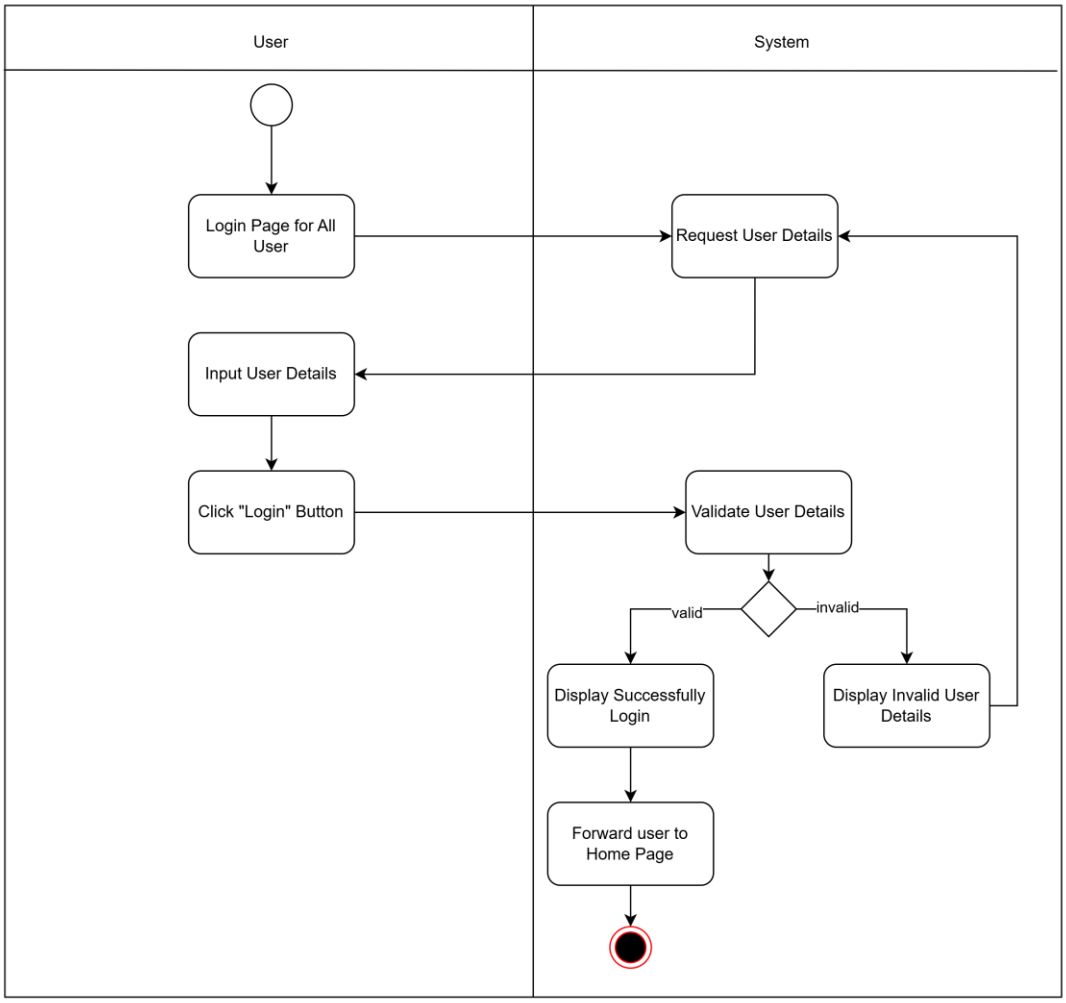


Figure 3.9 Login Activity Diagram

3.6.2 Manage Booking

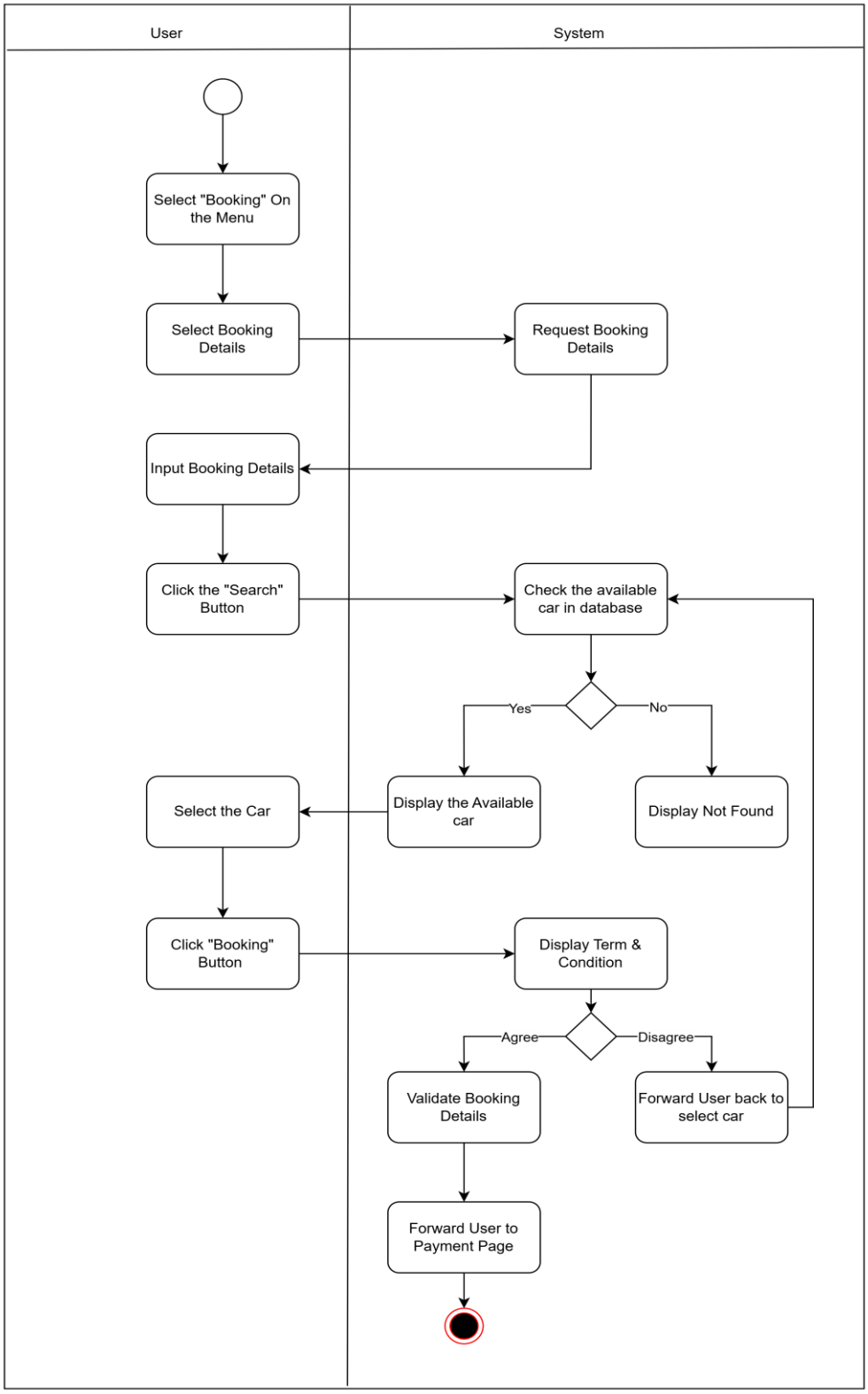


Figure 3.10 Booking Activity Diagram

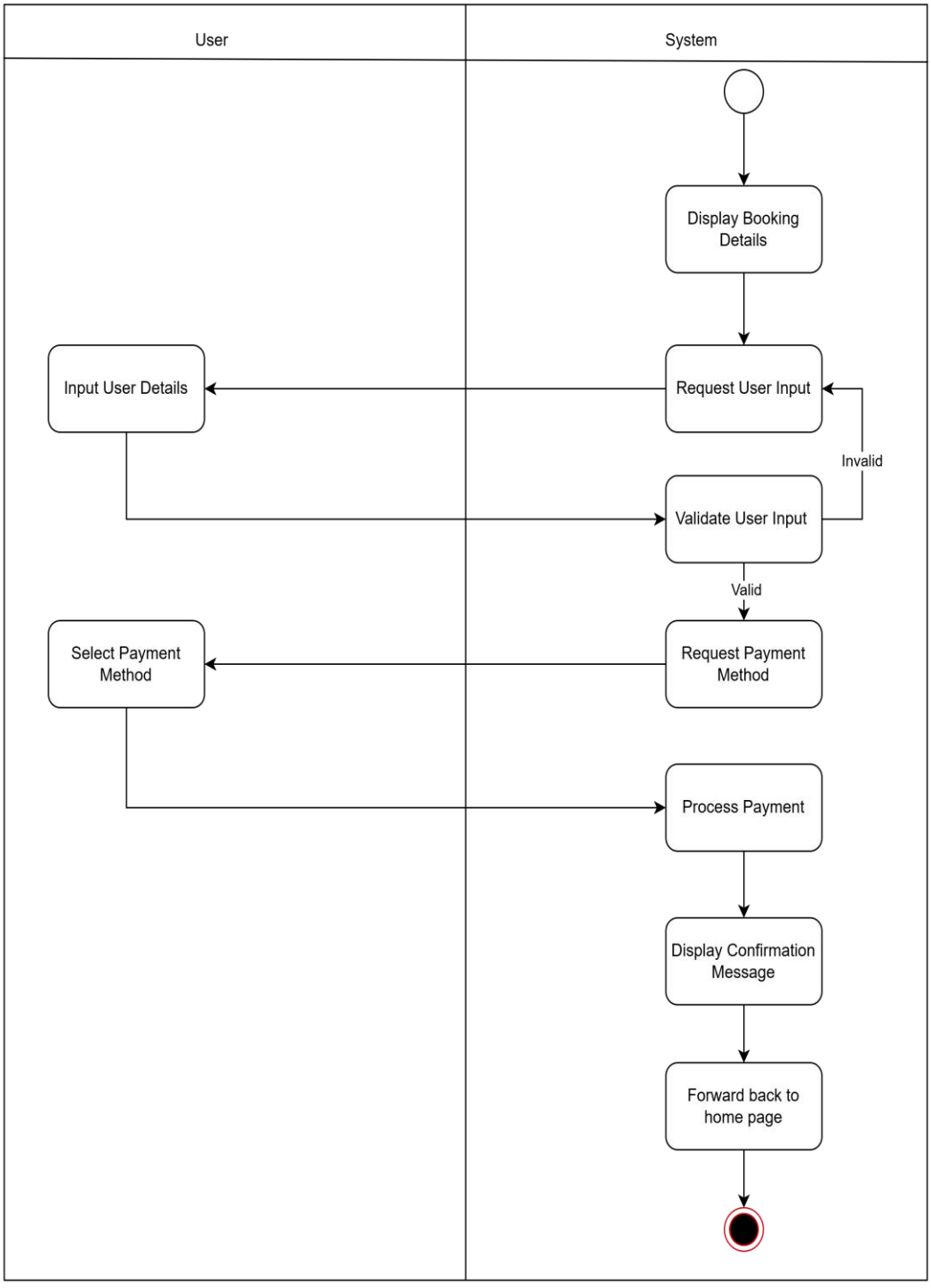


Figure 3.11 Payment Activity Diagram

3.6.3 Manage Profile

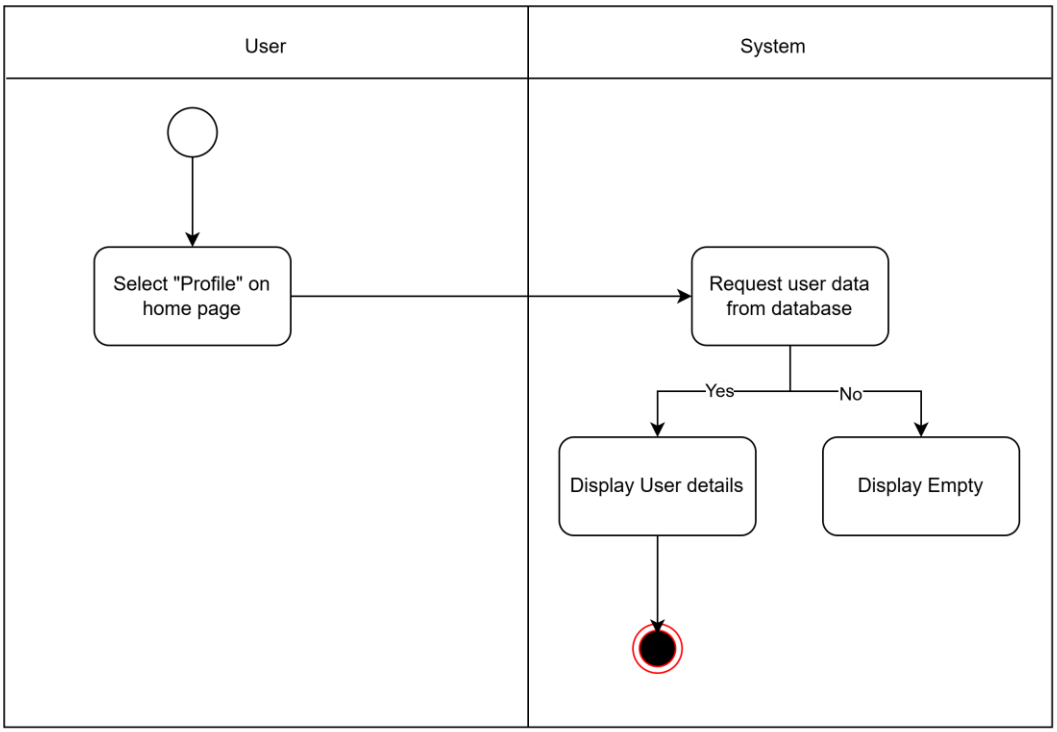


Figure 3.12 Profile Activity Diagram

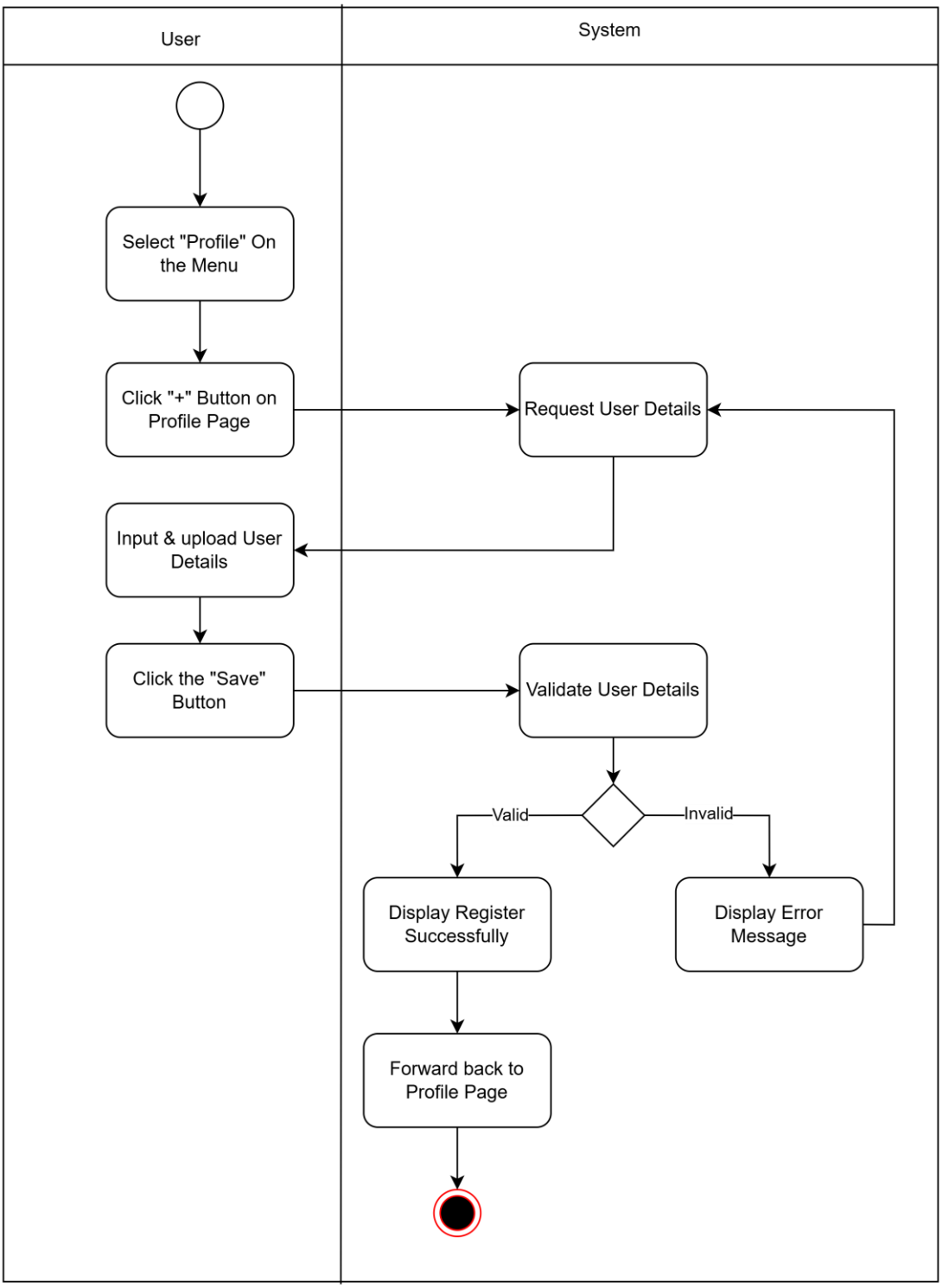


Figure 3.13 Registration Profile Activity Diagram

3.6.4 Manage Car

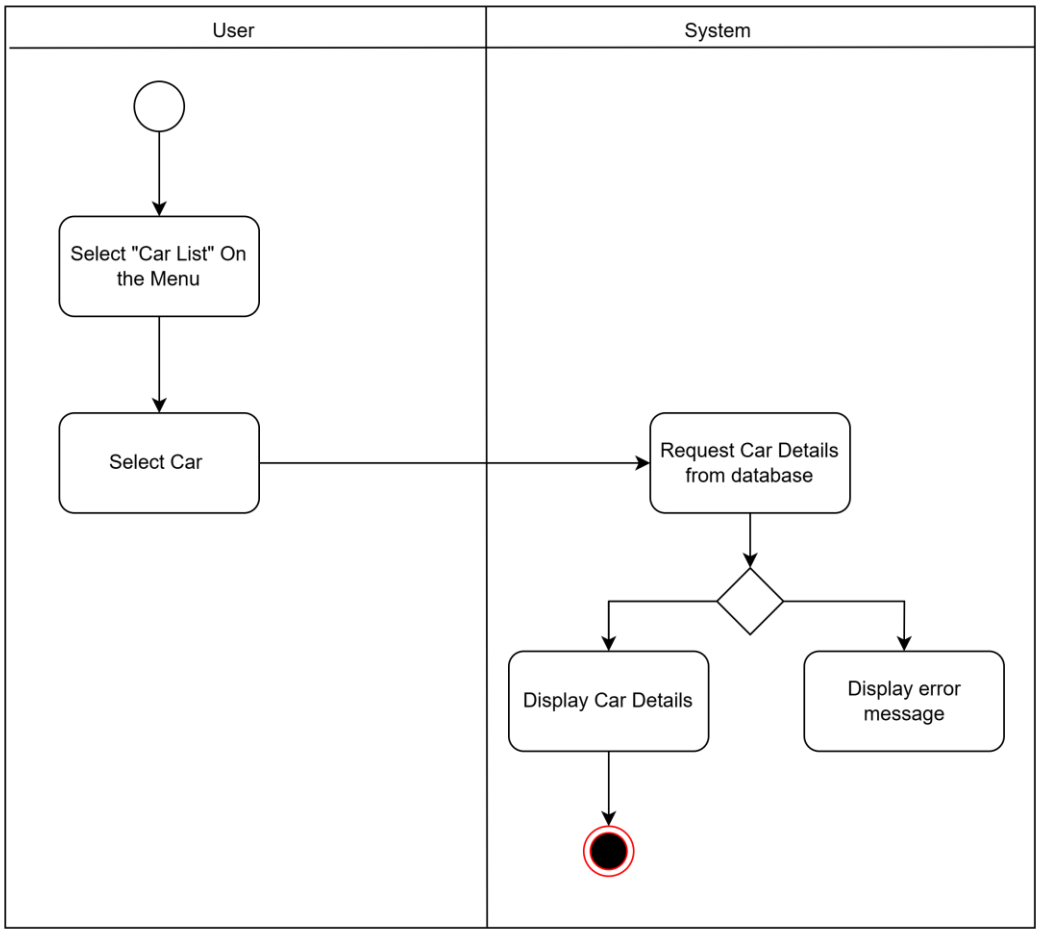


Figure 3.14 Car List Activity Diagram

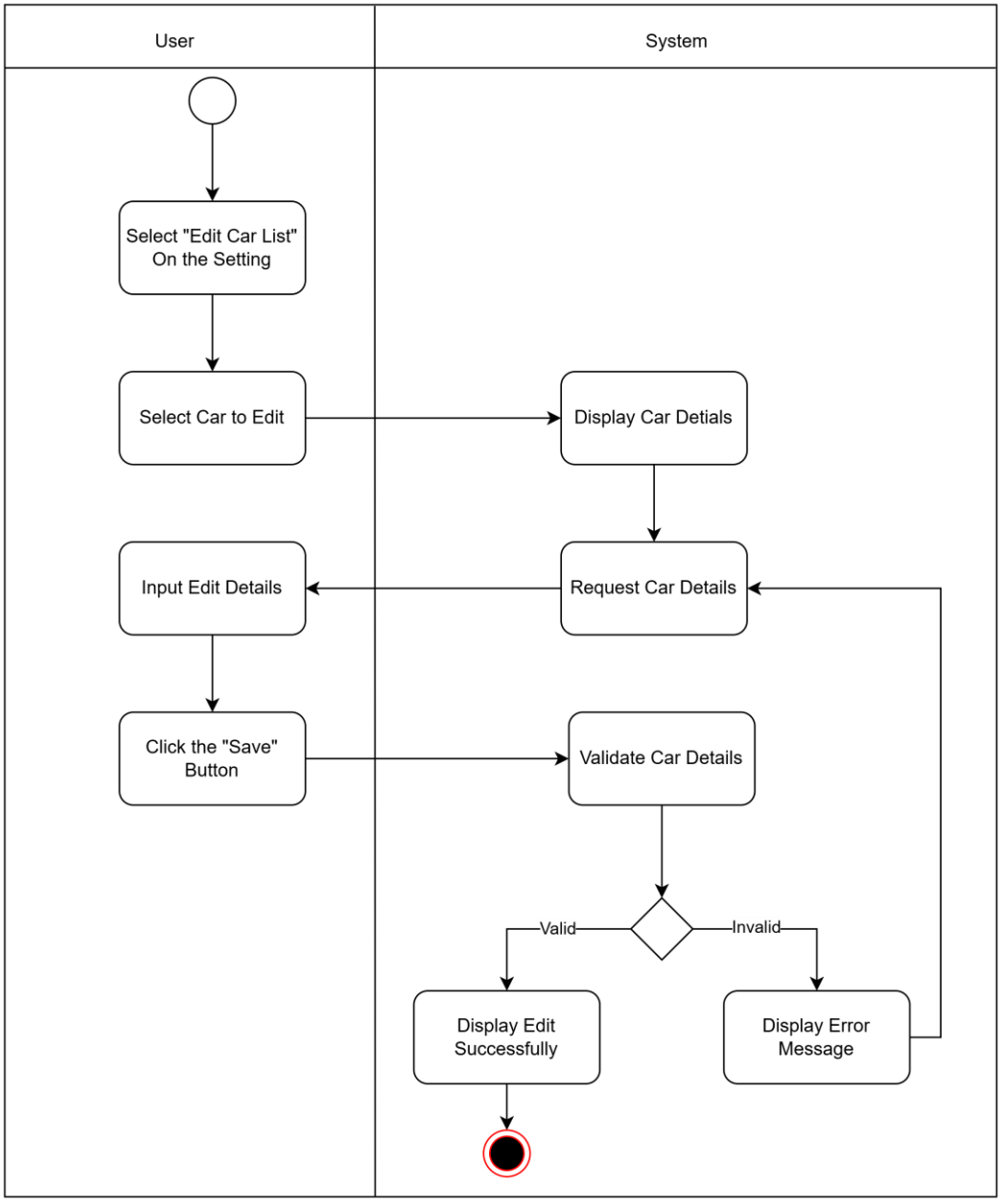


Figure 3.15 Edit Car Activity Diagram

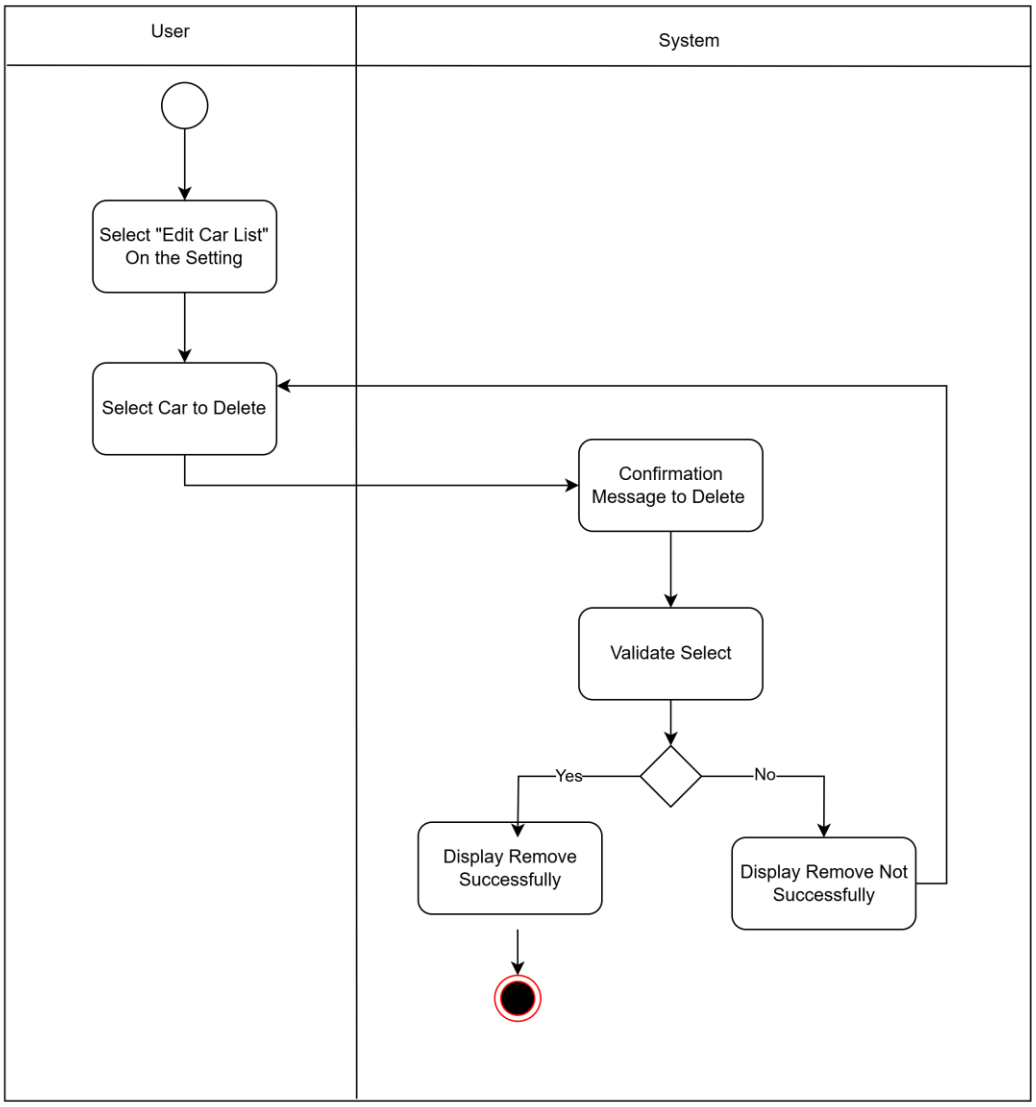


Figure 3.16 Delete Car Activity Diagram

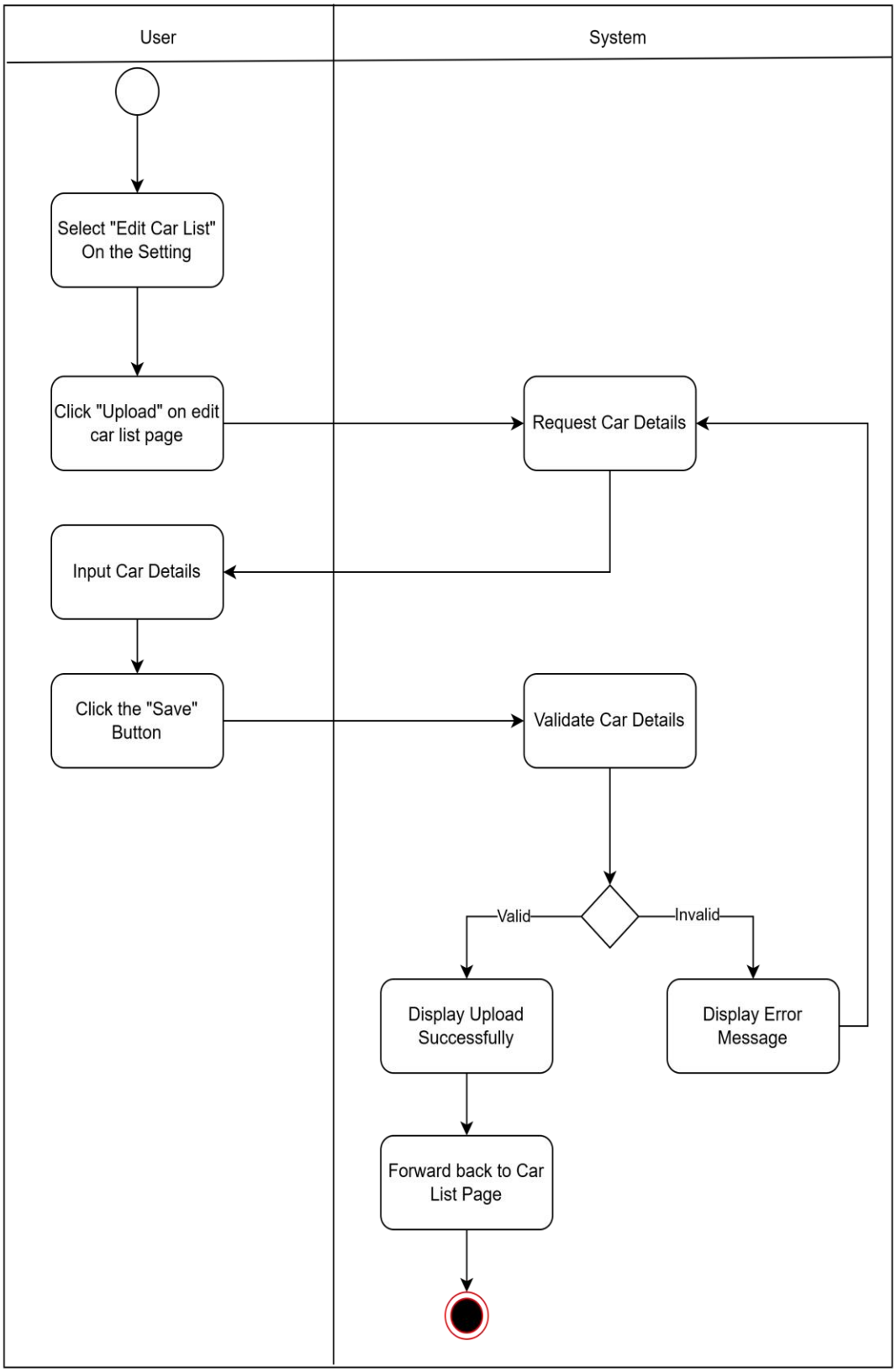


Figure 3.17 Add Car Activity Diagram

3.6.5 Manage Setting

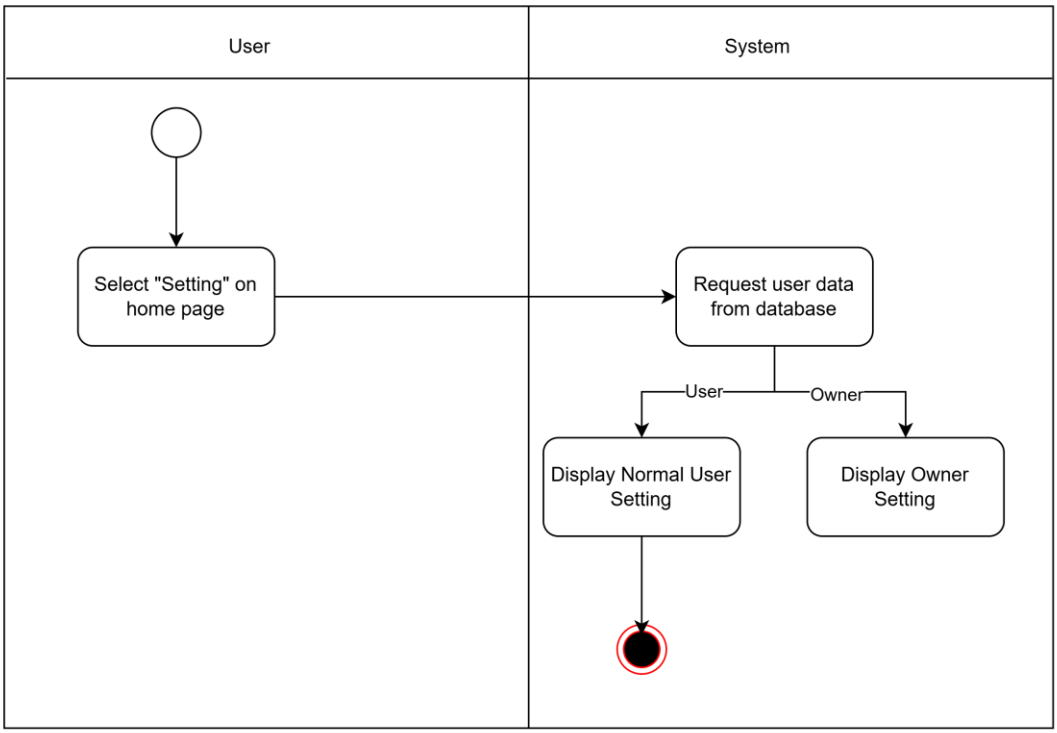


Figure 3.18 Setting Activity Diagram

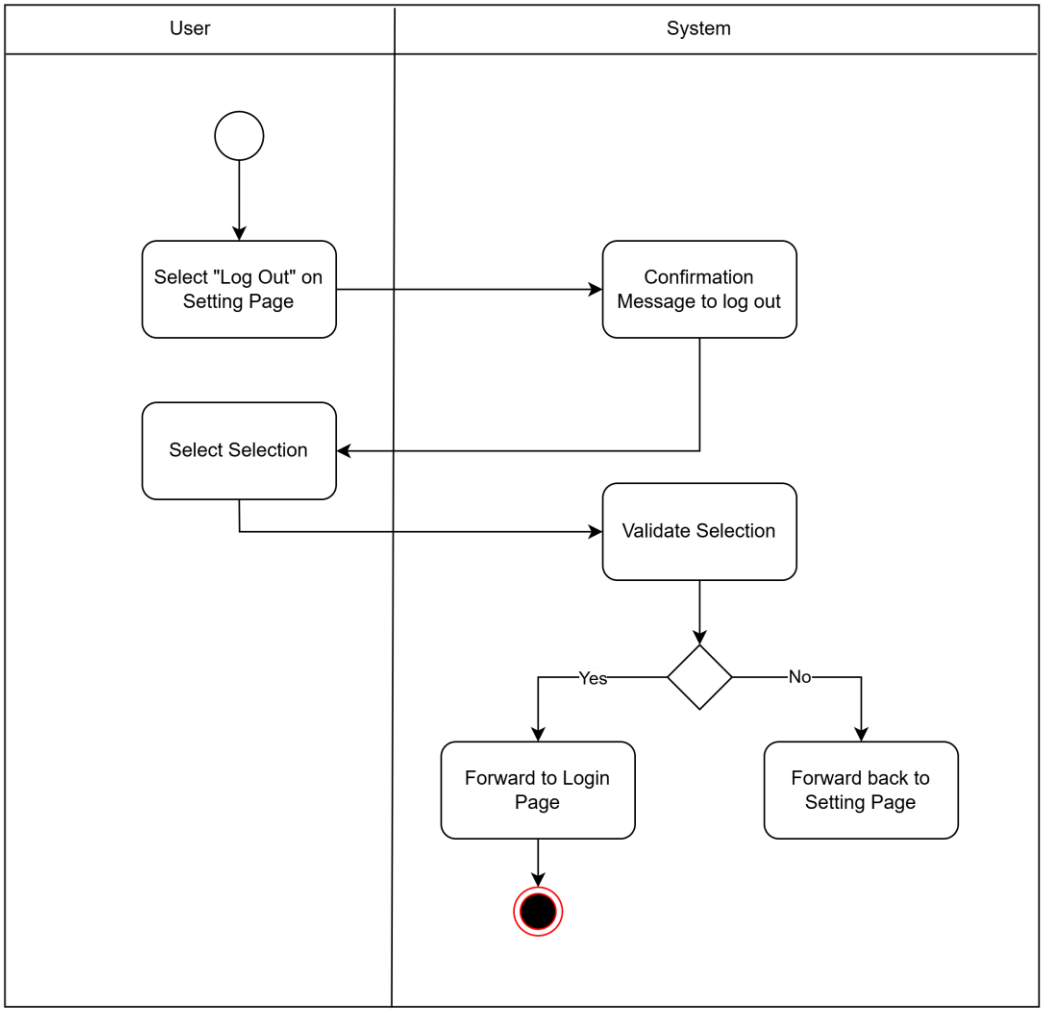


Figure 3.19 Log Out Activity Diagram

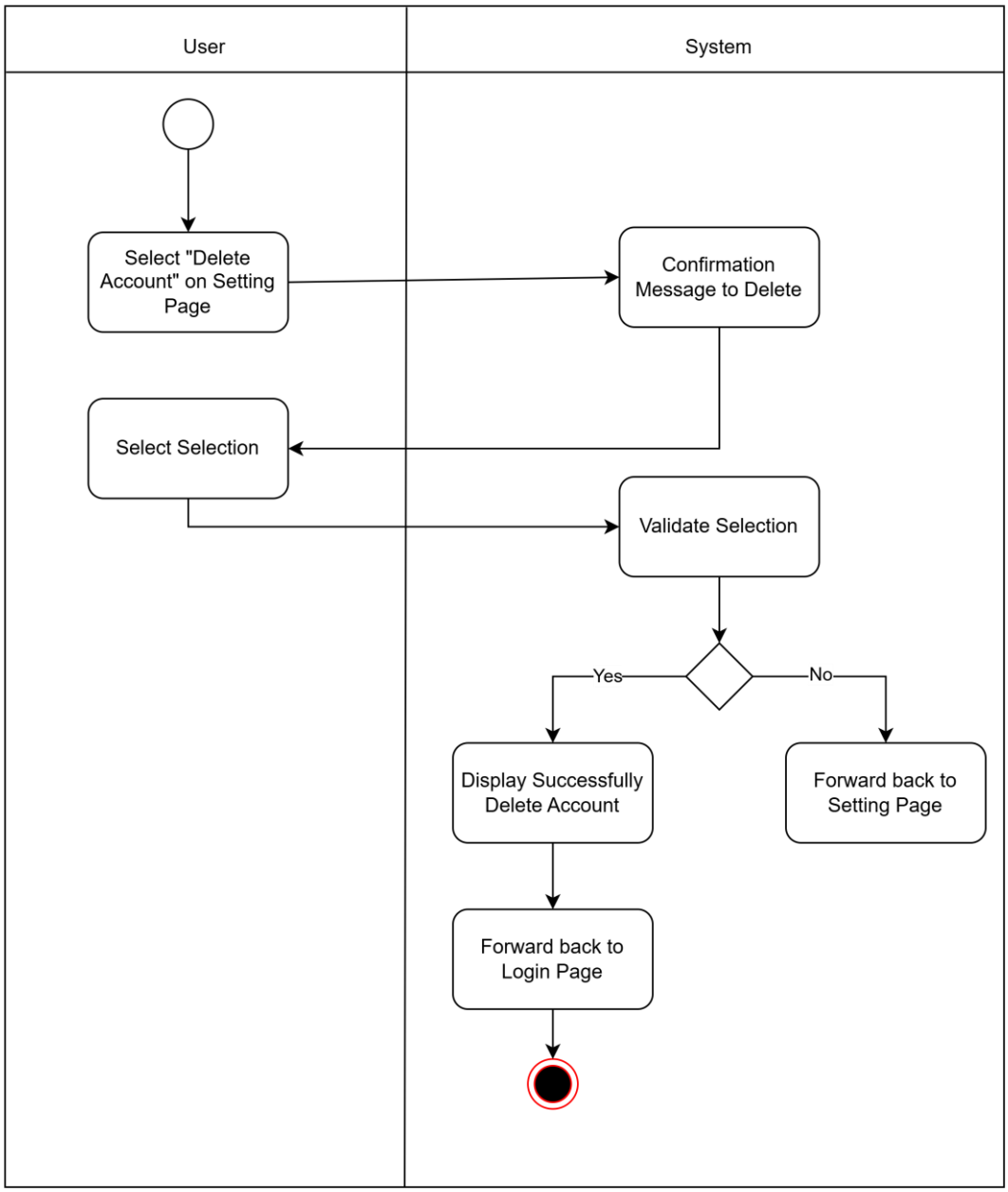


Figure 3.20 Delete Account Activity Diagram

3.6.6 Manage Notification

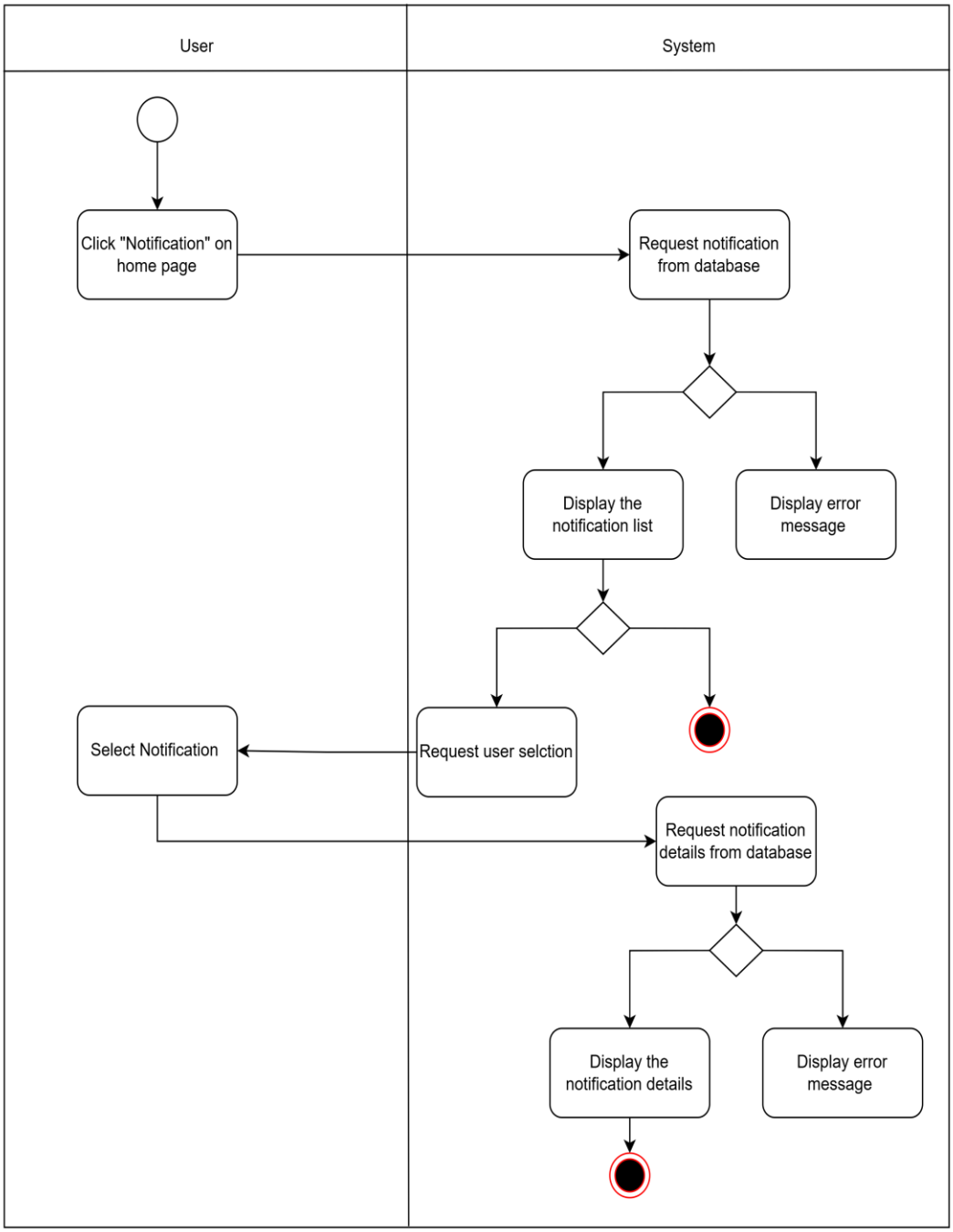


Figure 3.21 Notification Activity Diagram

3.6.7 Manage Booking Virtualization

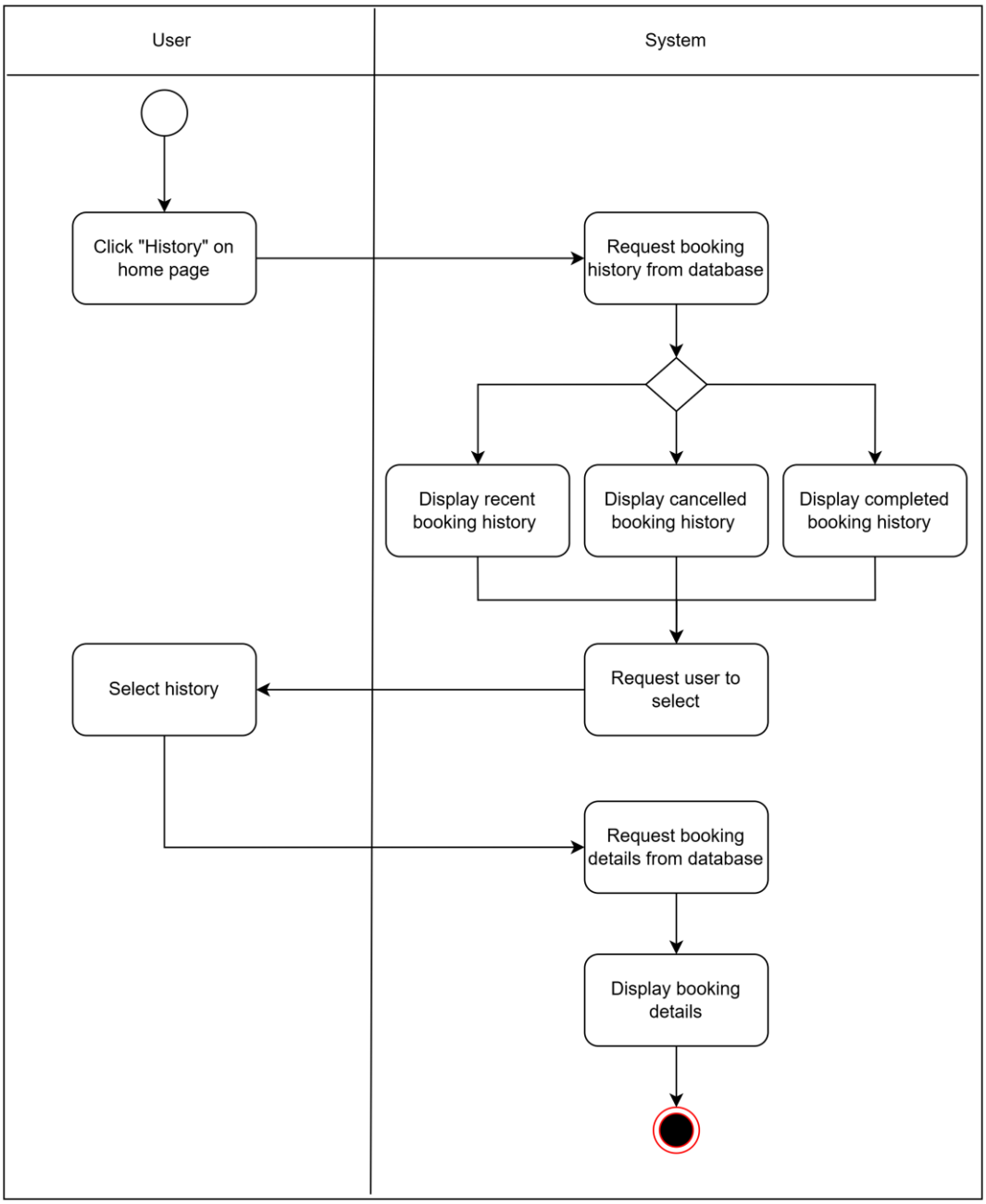


Figure 3.22 History Activity Diagram

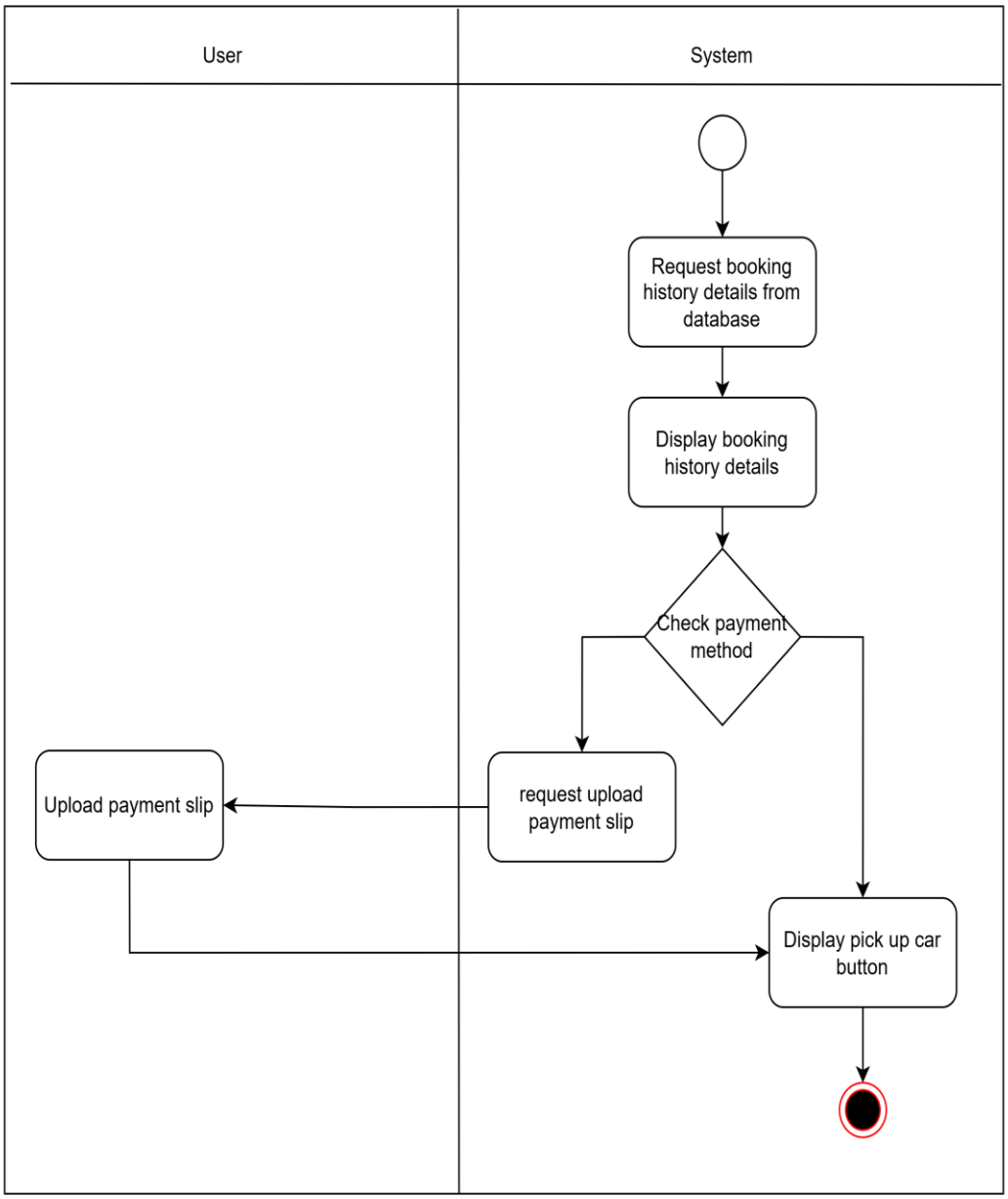


Figure 3.23 Upload Payment Slip Activity Diagram

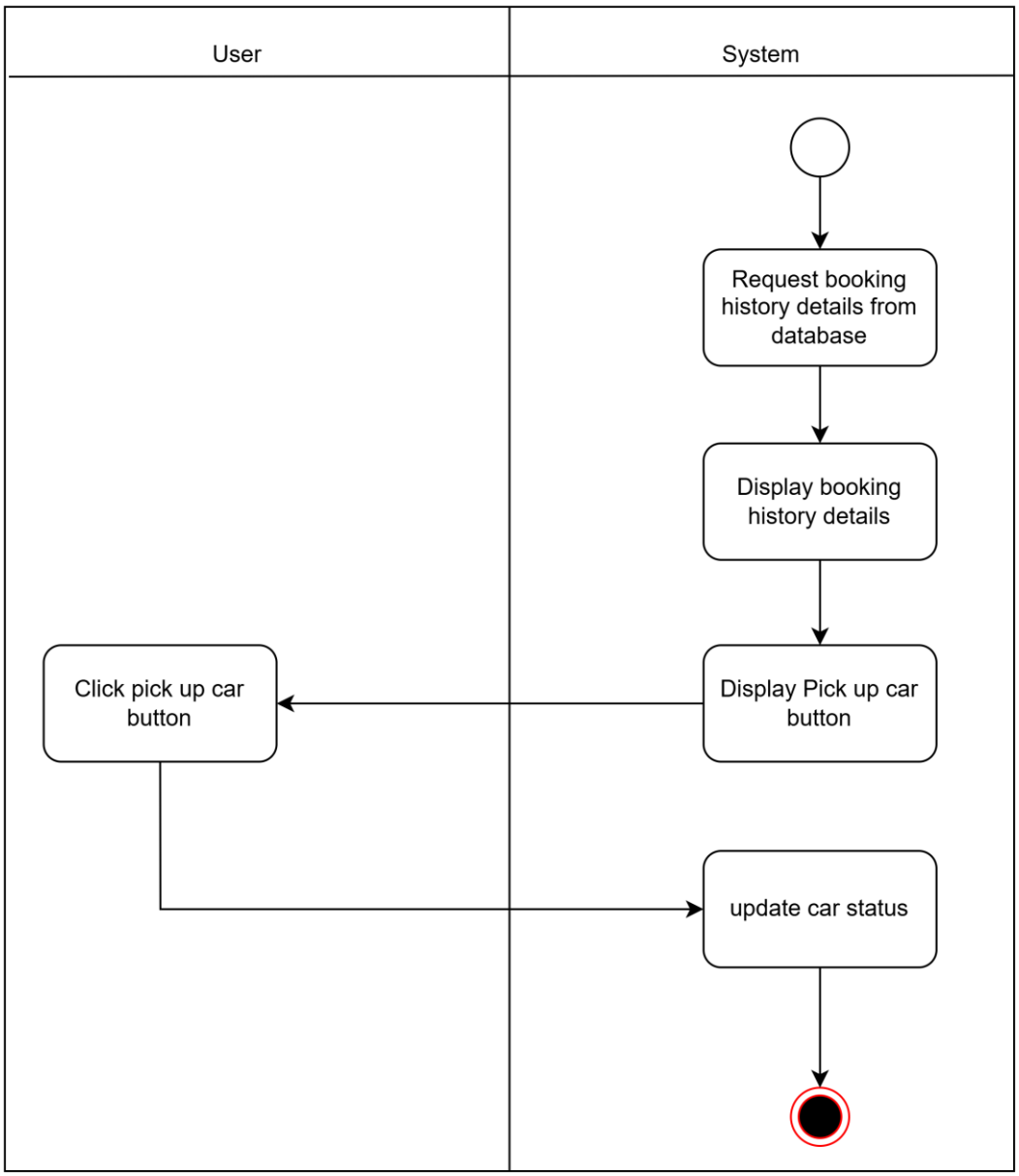


Figure 3.24 Pickup car Activity Diagram

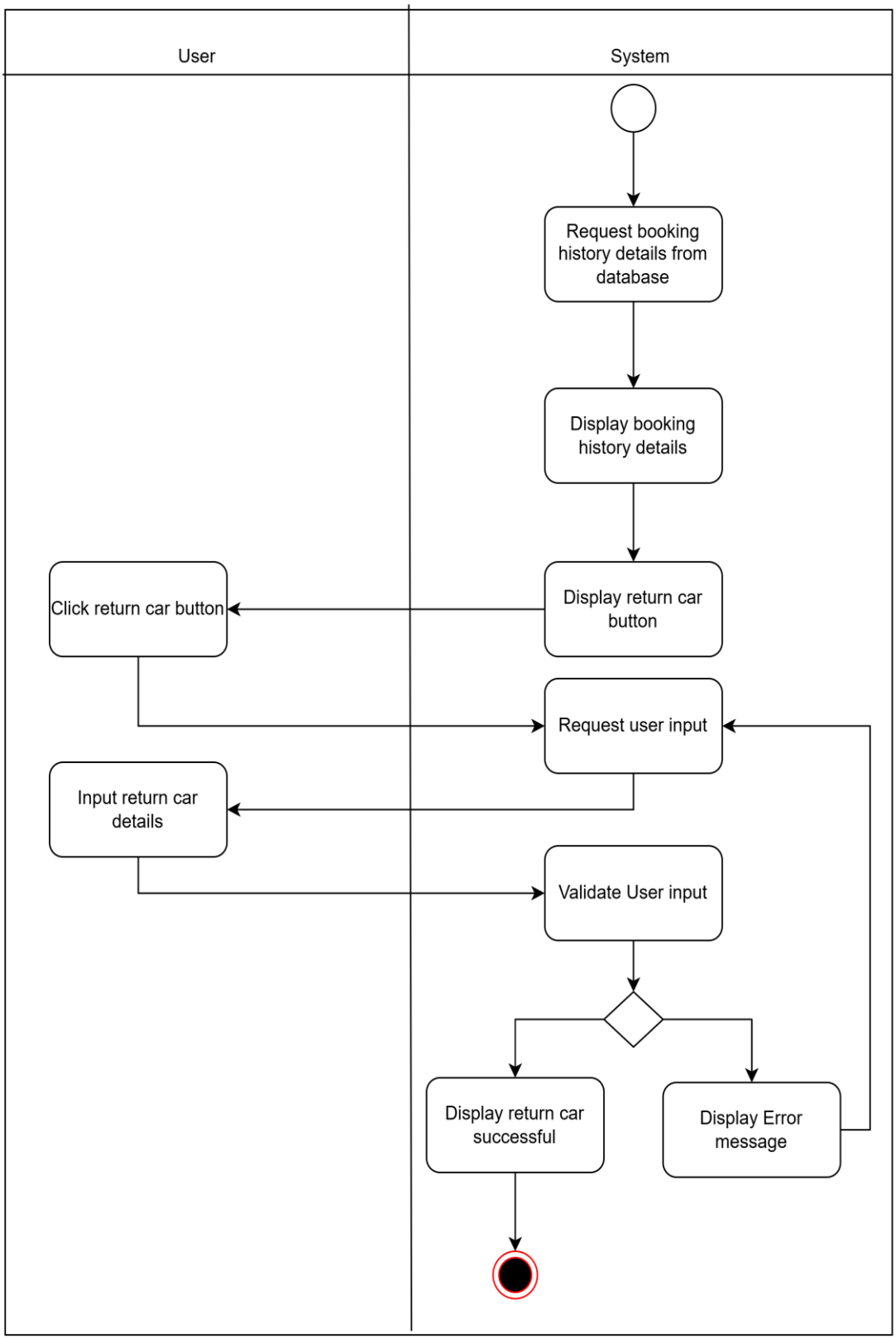


Figure 3.25 Return car Activity Diagram

Chapter 4

System Design

4.1 System Block Diagram

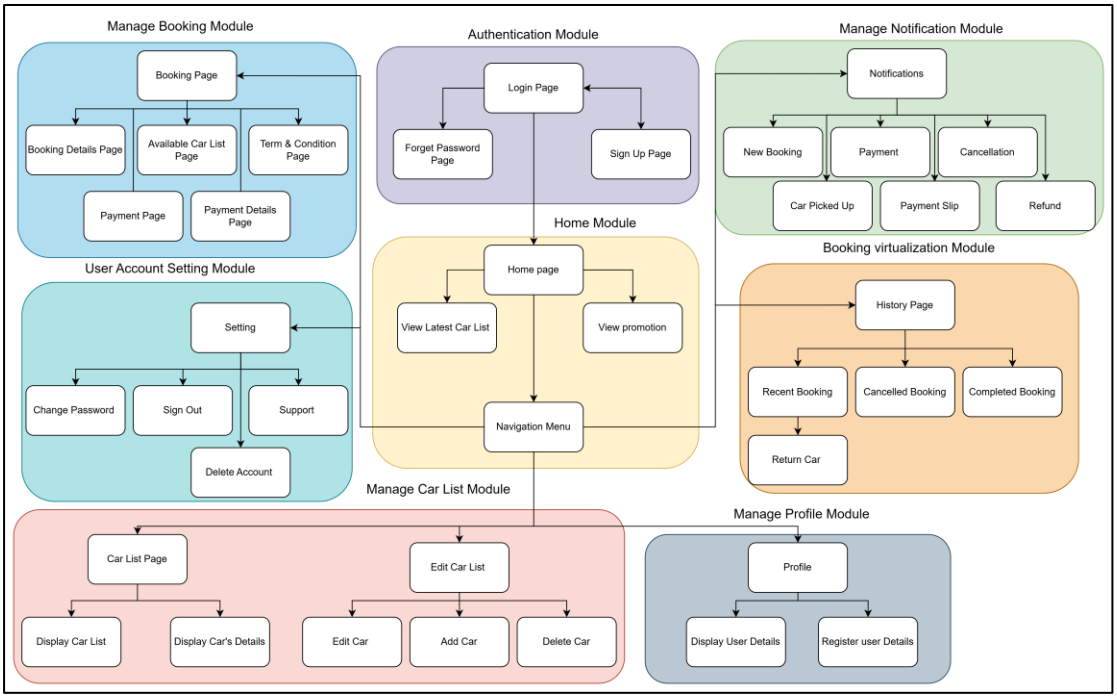


Figure 4.1 Block Diagram

The Figure 4.1 showed the overall block diagram of the proposed system. The system separated into 8 module which included Home Module, Authentication Module, Manage Booking Module, Manage Car List Module, Manage Notification Module, Booking Virtualization Module, User Account Setting Module and Manage Profile Module. Authentication module is a module that allows users to register and verify their identity before access into the system. In this module it includes login page which use to validate user's identity, while sign up page which use to create a new access account to gain access into the system and forget password page is allow user to reset the password if user forgotten. In addition, Home Module is a module that allows users to navigate to other modules by clicking the button at the navigation menu and display the latest car and promotion. Besides, Manage Booking Module allow user to make a new booking to book a car. In order to perform a booking user have to fill in the booking details, select the available car, read the term and condition, and make the payment,

after that the payment details confirmation will be provide. In the Booking Virtualization Module, system will provide three list of booking history to view, it will be separate to three part which is recent, cancel and completed. In the recent user are needed to click to the button for picked up and return car. While Notifications Module is a module that allow two users type to receive the notification. For normal user, it will receive the notification such as payment successful, booking cancellation, car picked up, upload payment slips, and refund, while car owner will receive notification such as new booking, payment successful, cancellation booking, refund, payment slip uploaded, and car picked up. In the Manage Car List module only car owner user can be managing their own car list. In order to perform add, modify and delete car, a list to view the car list data will be providing. In the Manage Profile Module is a module that allows users to view their profile details and register their full user profile details. Lasty user account setting module is a module more likely preferences setting of the system which allow user to delete account, log out account, and reset password.

4.2 System Flowchart

4.2.1 Flowchart of Authentication Module

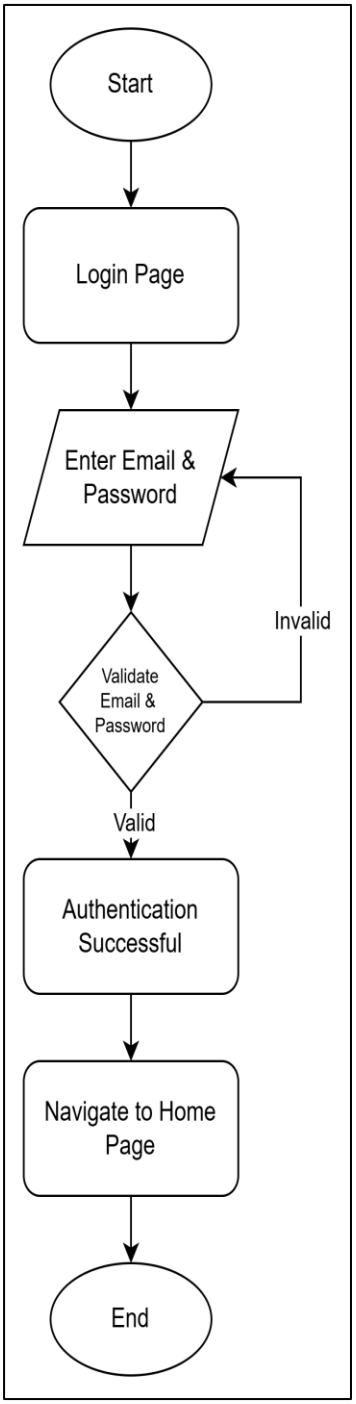


Figure 4.2 Login Flowchart

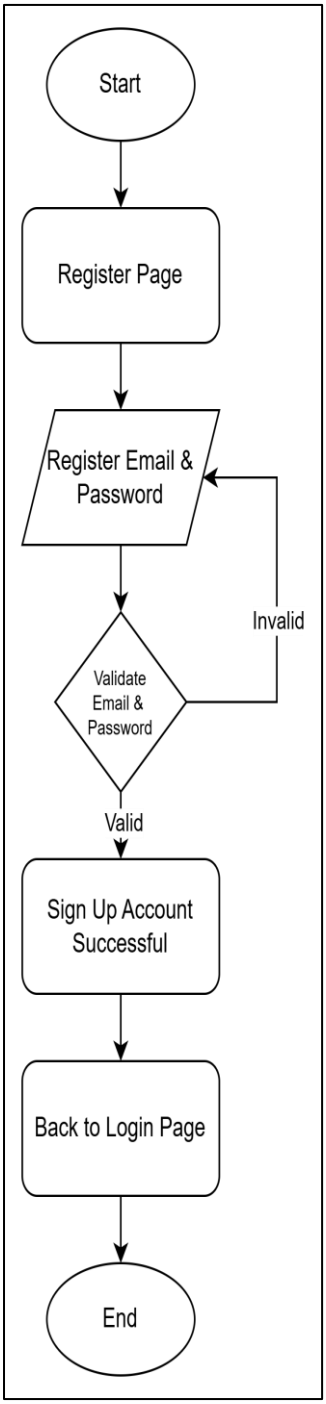


Figure 4.3 Sign up Flowchart

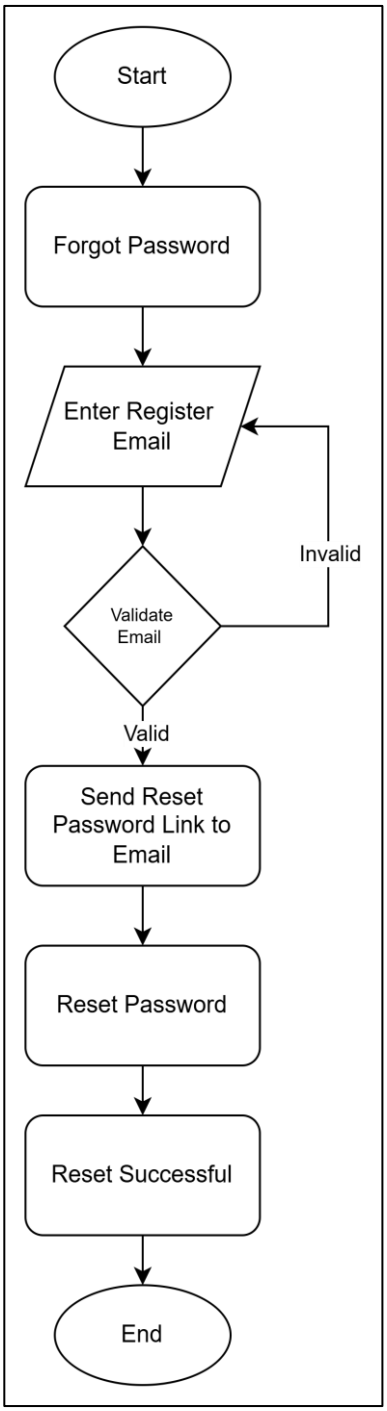


Figure 4.4 Forgot Password flowchart

4.2.2 Flowchart of Manage Profile Module

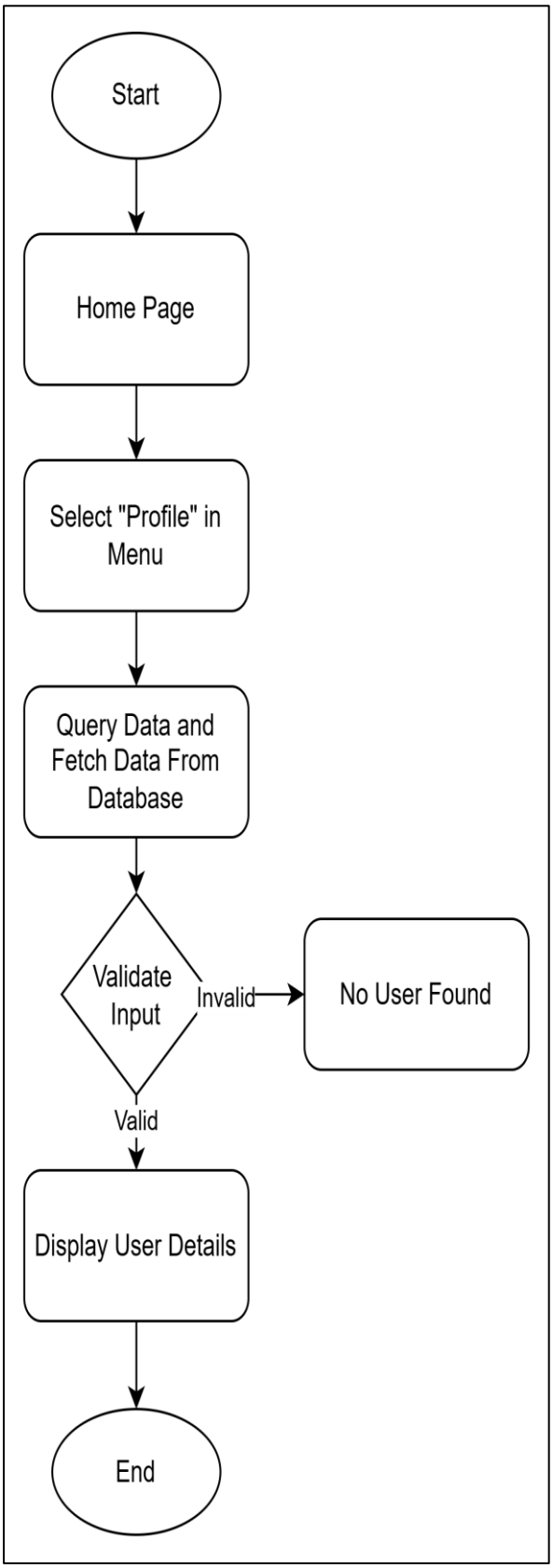


Figure 4.5 Profile Flowchart

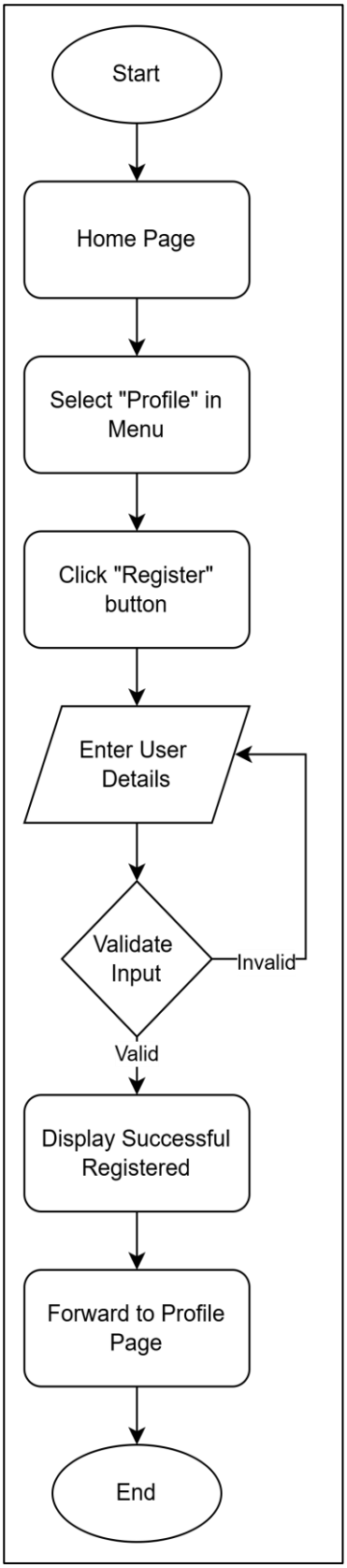


Figure 4.6 Profile Register Flowchart

4.2.3 Flowchart of Manage Car List Module

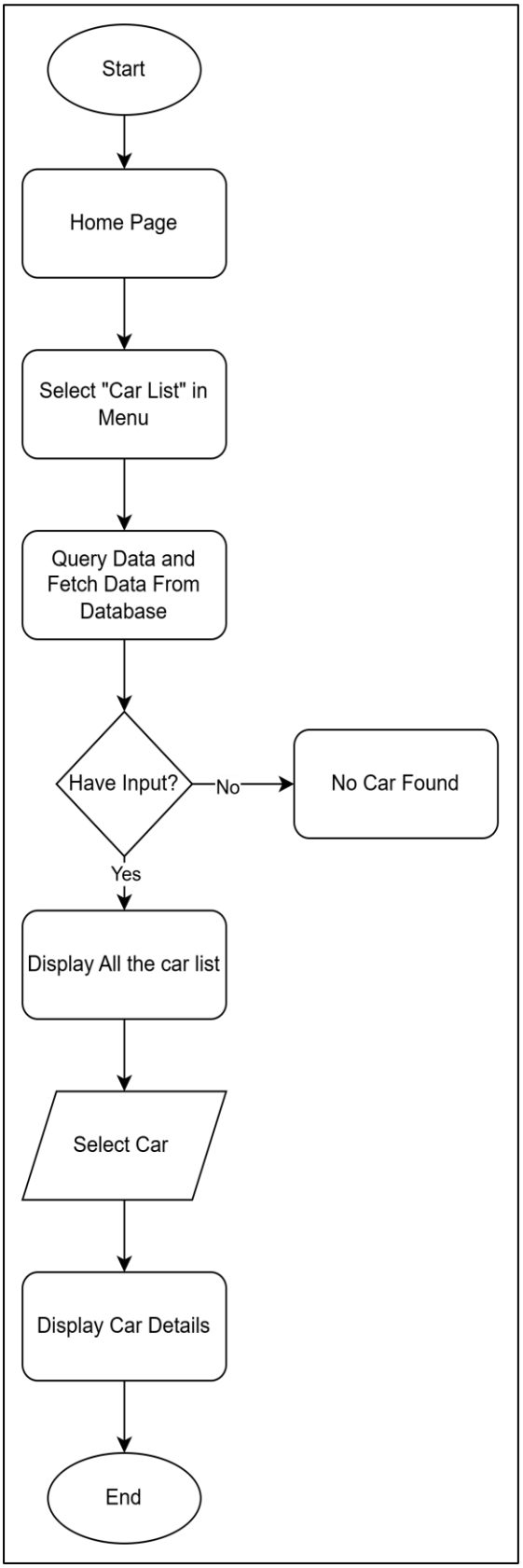


Figure 4.7 Car List Flowchart

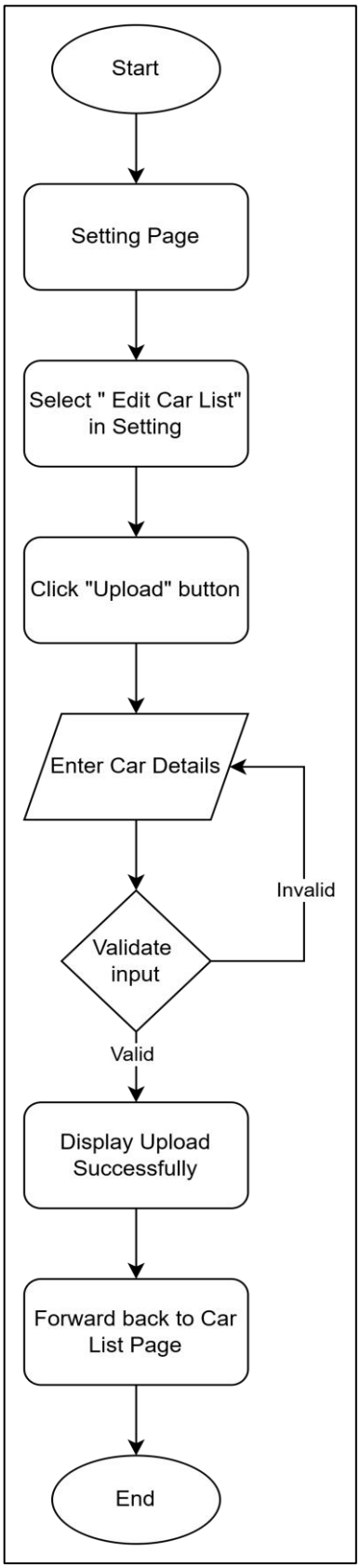


Figure 4.8 Add Car Flowchart

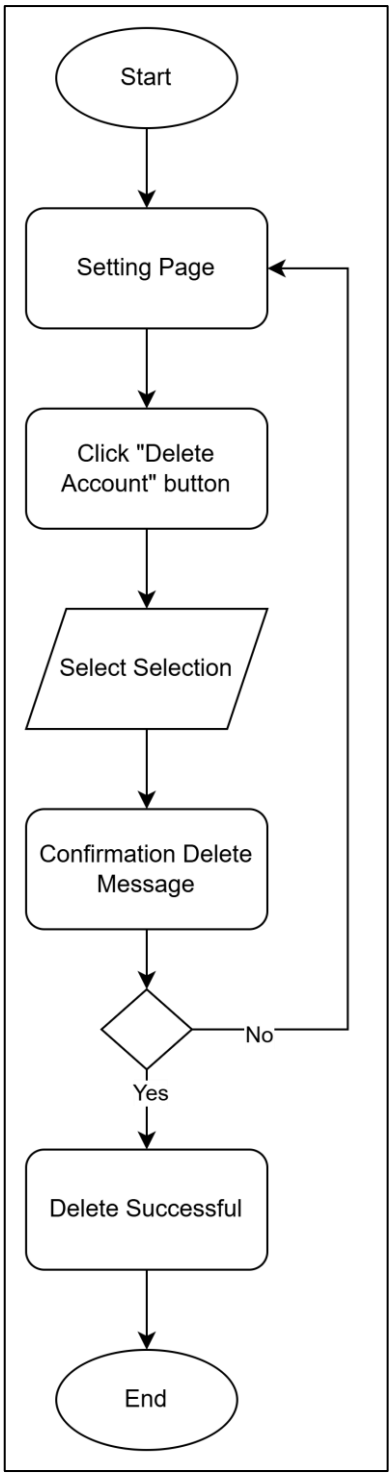


Figure 4.9 Delete car Flowchart

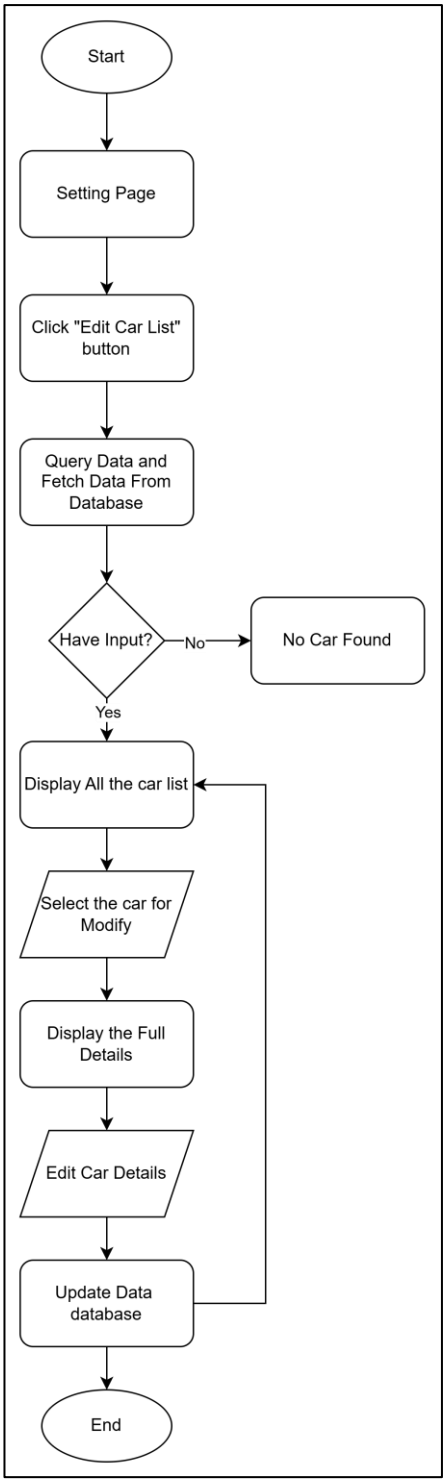


Figure 4.10 Edit car Flowchart

4.2.4 Flowchart of Manage Booking Module

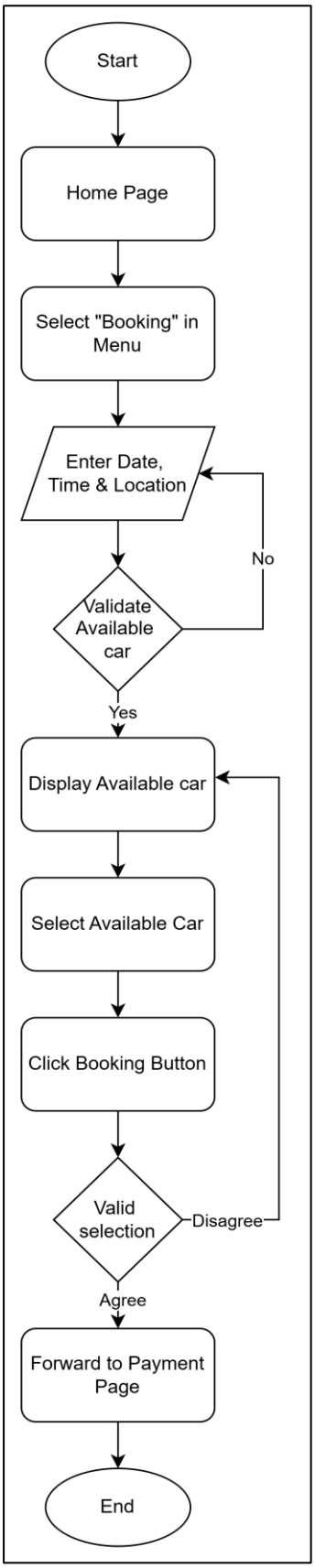


Figure 4.11 Booking Flowchart

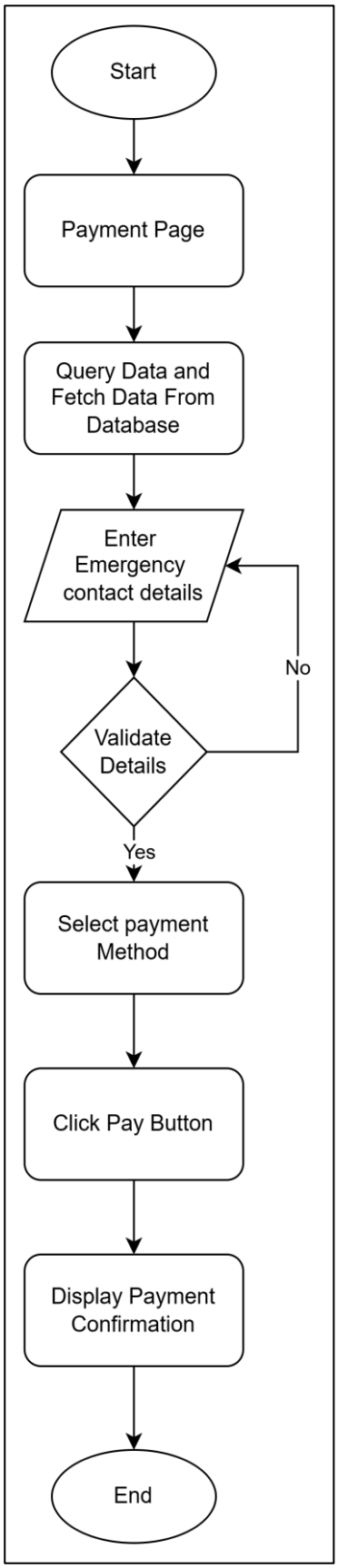


Figure 4.12 Payment Flowchart

4.2.5 Flowchart of Manage User Account Setting Module

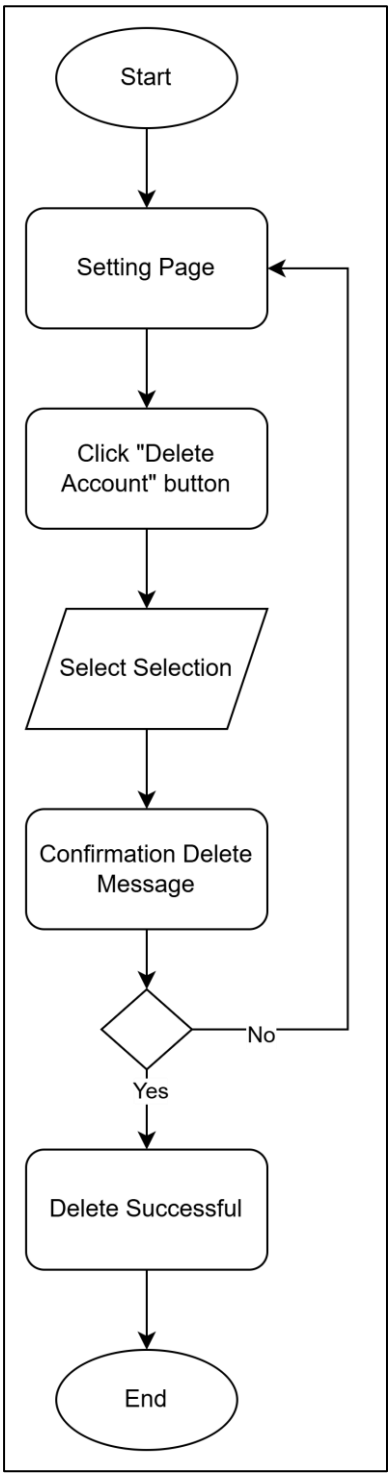


Figure 4.13 Delete Account Flowchart

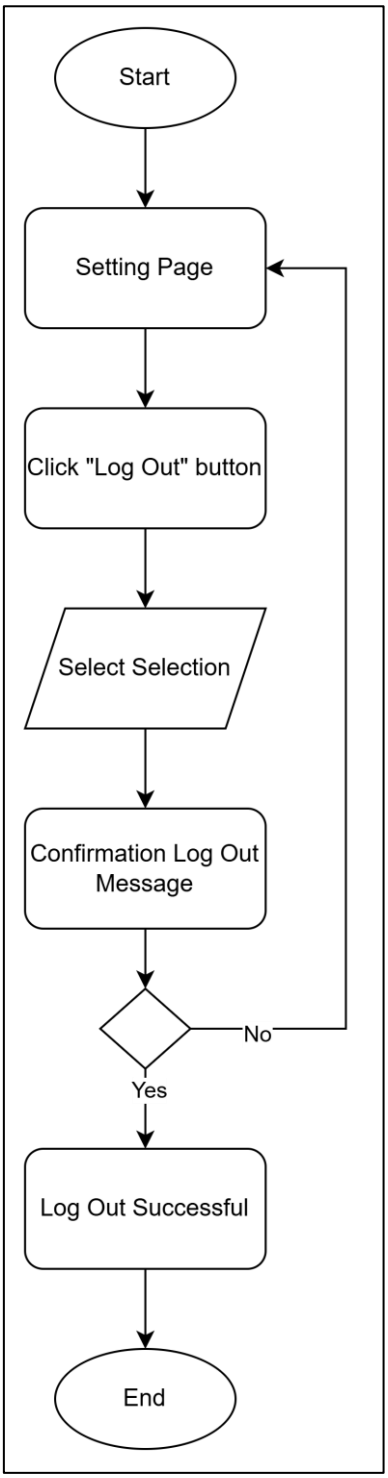


Figure 4.14 Log Out Flowchart

4.2.6 Flowchart of Manage Booking Virtualization Module

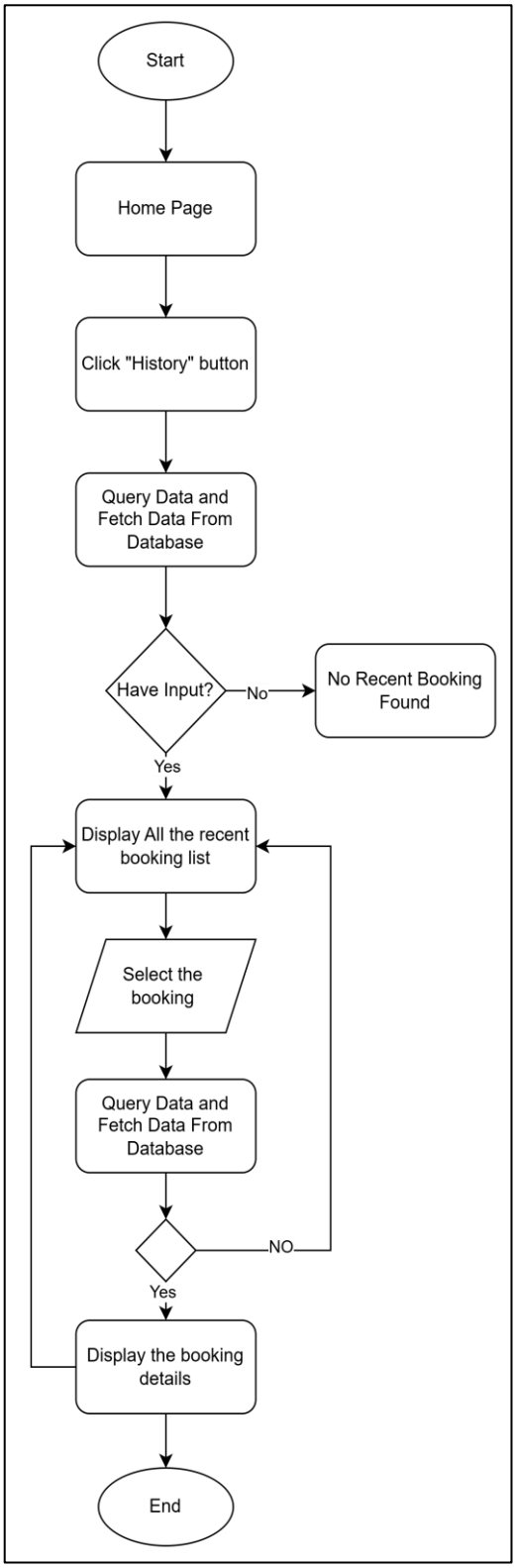


Figure 4.15 History "Recent" Flowchart

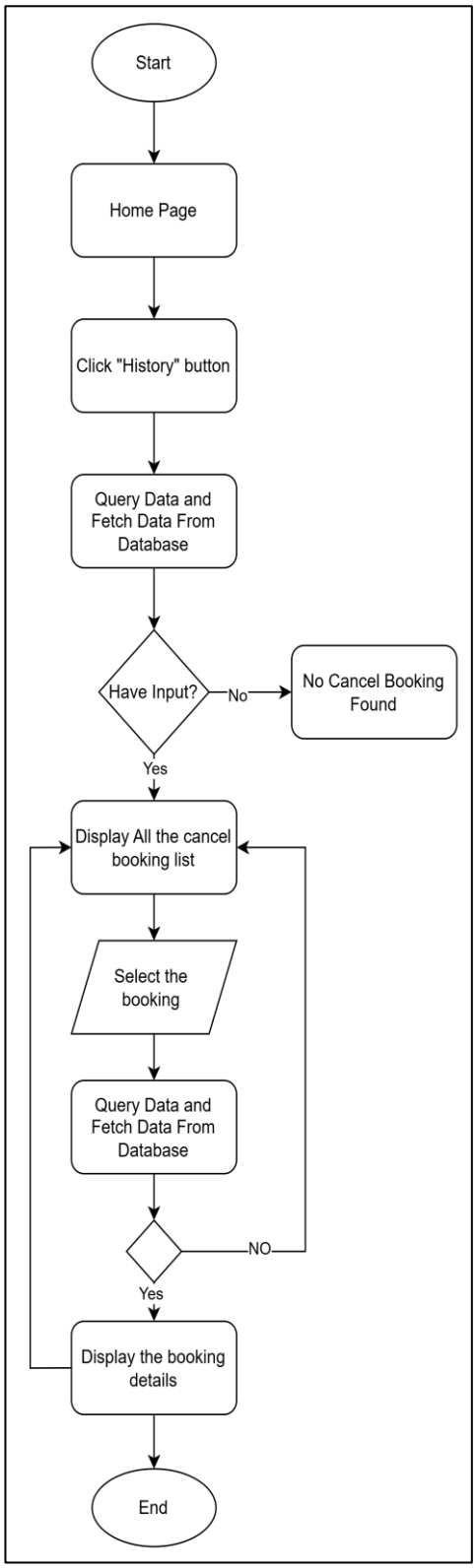


Figure 4.16 History "cancel" Flowchart

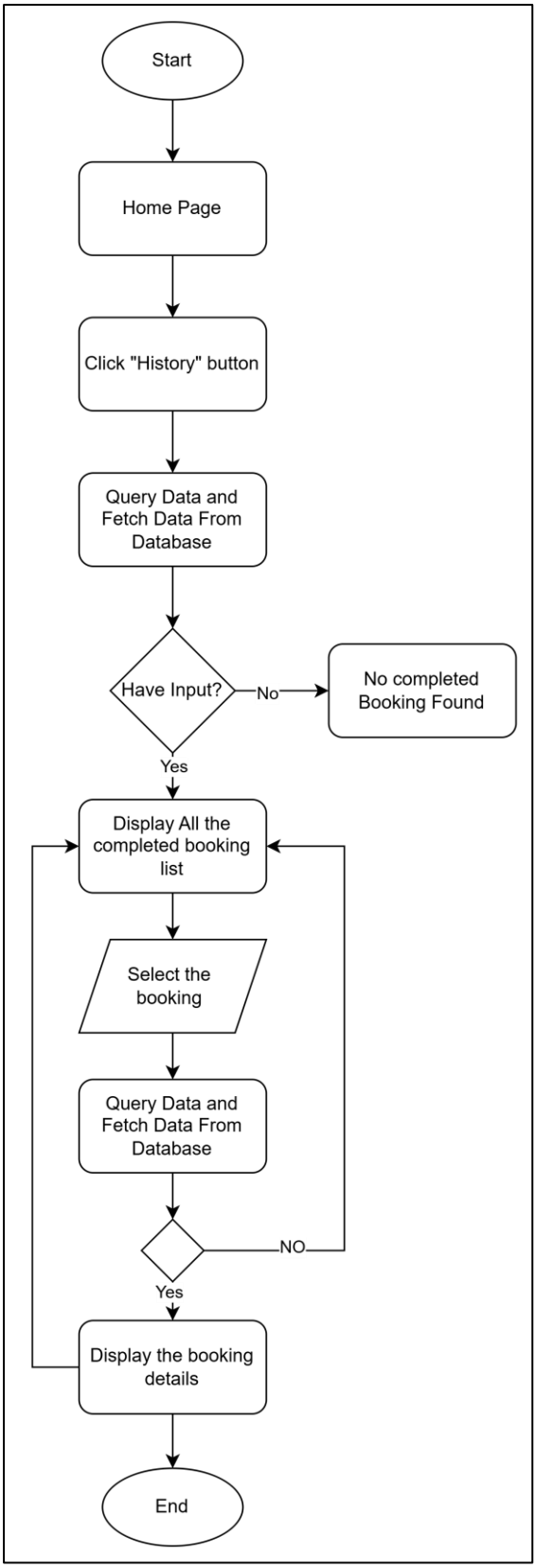


Figure 4.17 History "completed" Flowchart

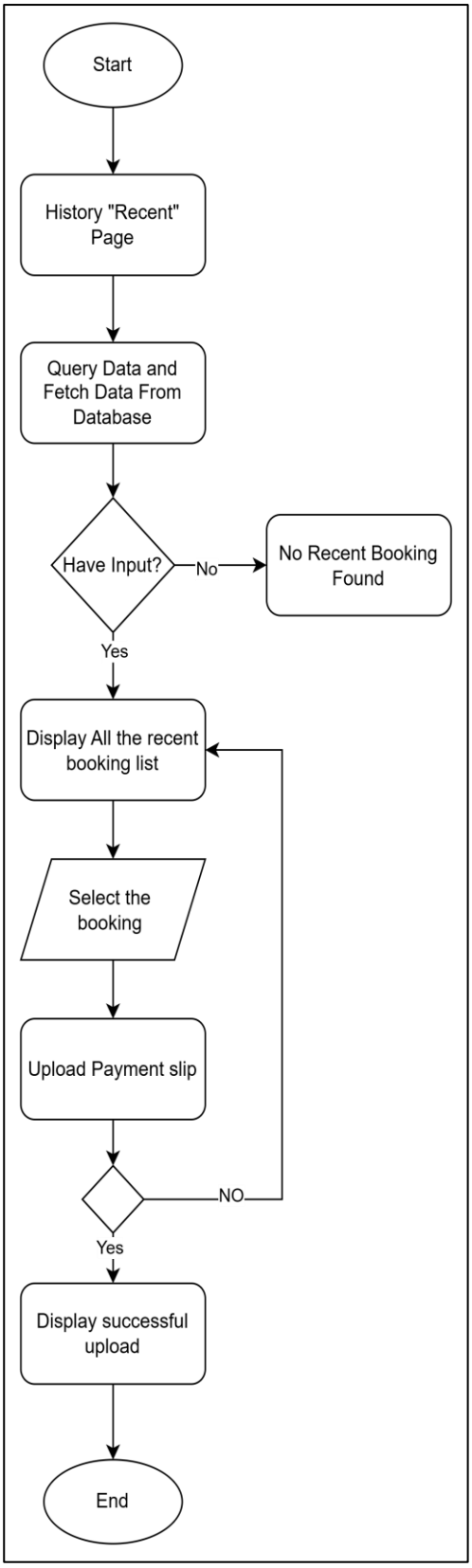


Figure 4.18 Upload Payment Slip Flowchart

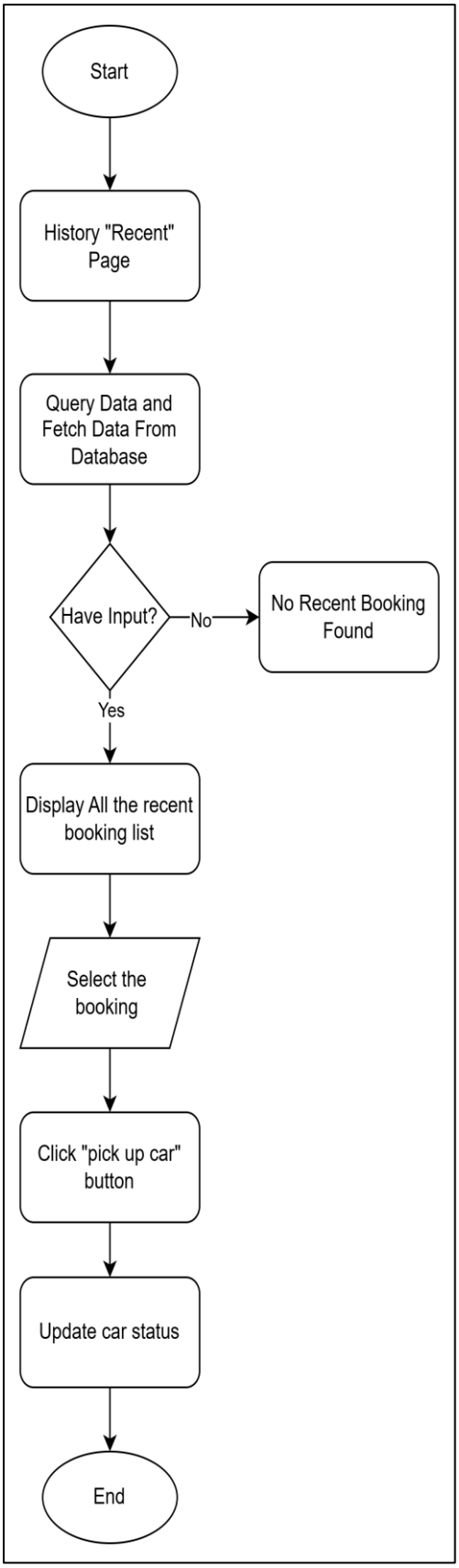


Figure 4.19 Pickup Car Flowchart

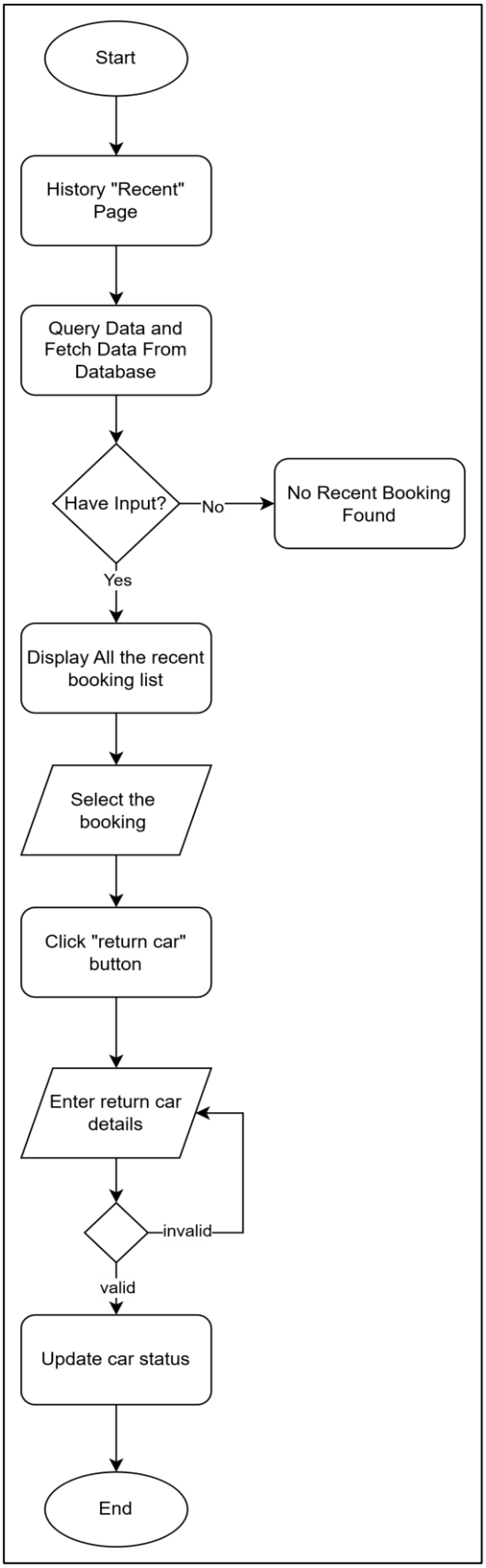


Figure 4.20 Return Car Flowchart

4.2.7 Flowchart of Manage Notification Module

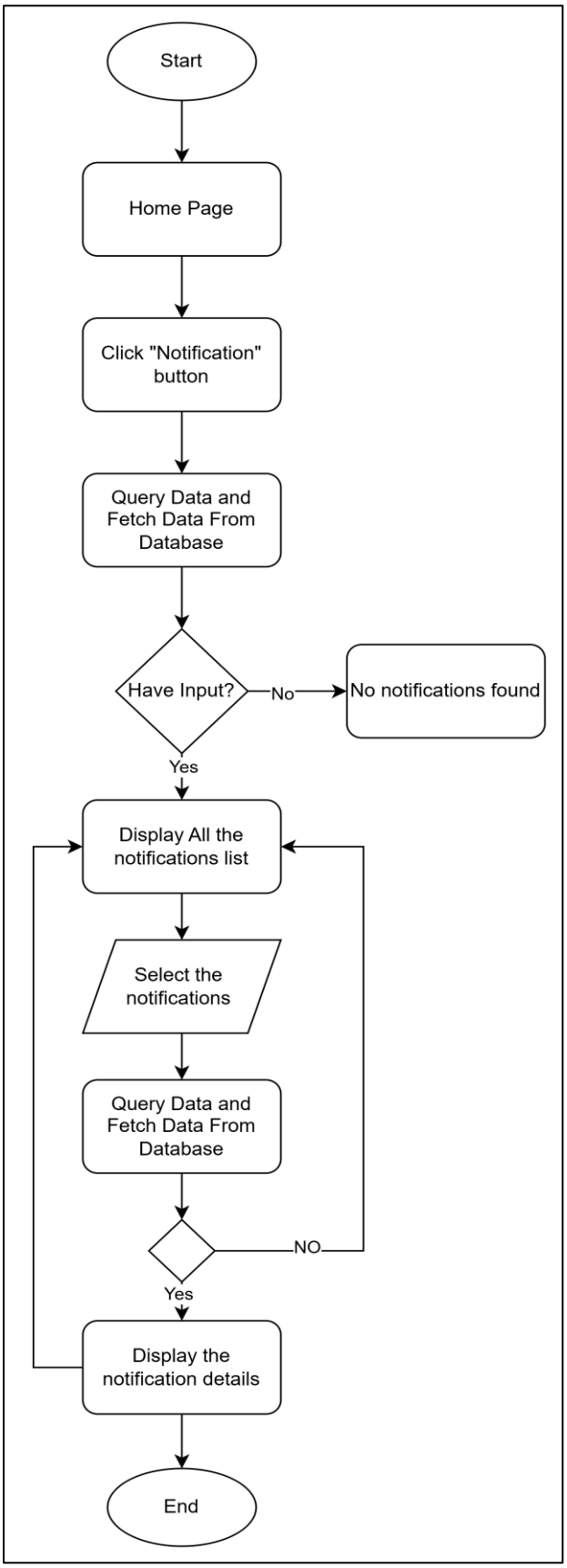


Figure 4.21 Notification Flowchart

4.3 Entity Relationship Diagram

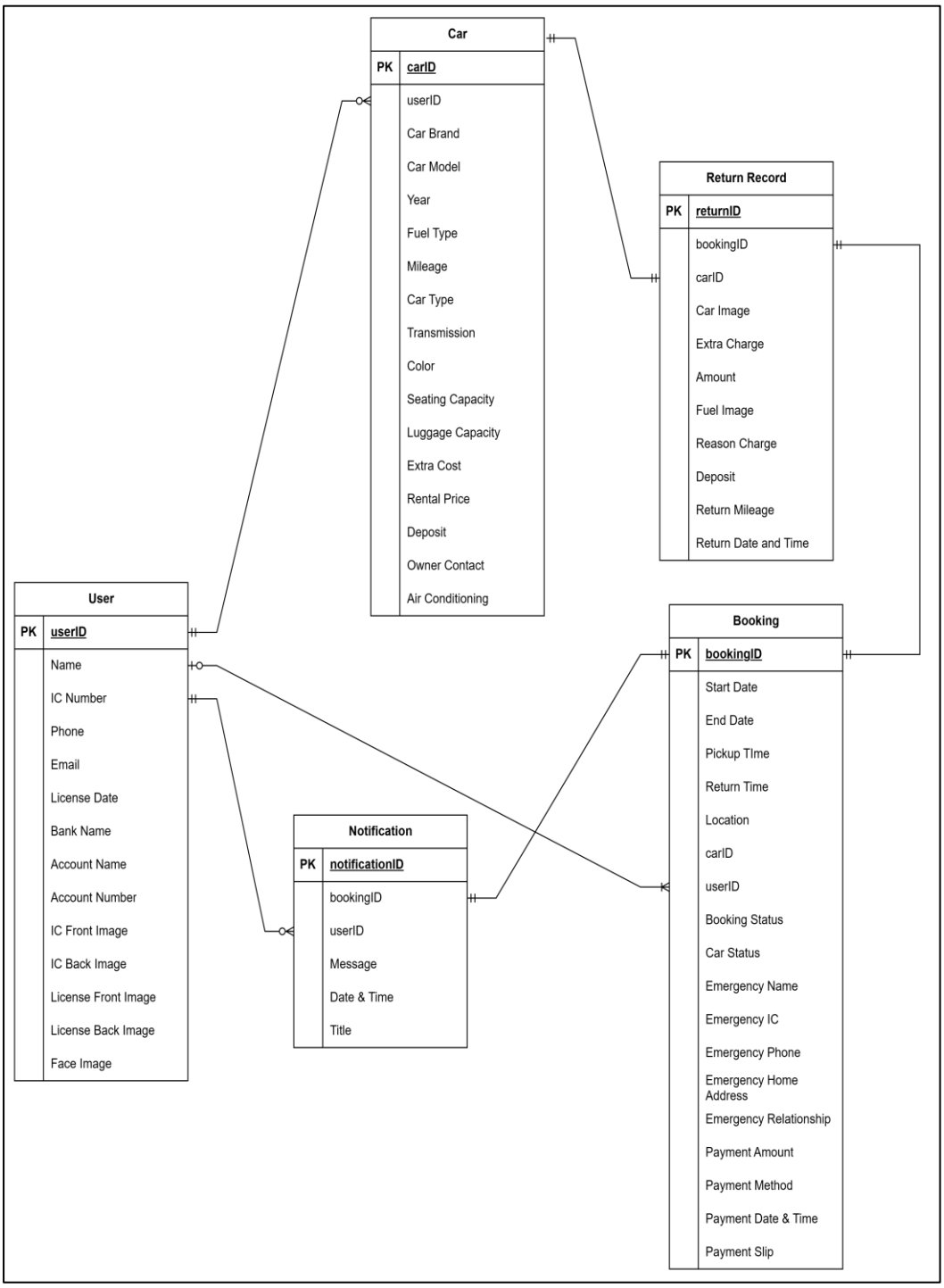


Figure 4.22 Entity Relationship Diagram

Figure 4.22 shows the database diagram for the Car Rental Booking system, which provides the connection of the relationships between various entities such as User, Car, Booking, Return Record, and Notification. The diagram fully shows the backend structure of the proposed system, which stores and manages the data for the application. In this system, the User entity serves with a primary key of userID that links to other entities. When a user registers an account, their information such as name, phone, email and other information will stored in the User table. This userID serves as a unique identifier for each user (normal user and car owner) and is used in various booking, such as booking a car, receiving notifications or check the history. The Car entity is linked to the User through the userID, indicating that the car belongs to the car owner user. The Car entity stores attributes such as car brand, model, fuel type, and rental price, with the carID as the primary key. The Booking table holds information about the car booking, including booking dates, pickup and return times, and payment details which used to car the car availability. The returnID is the primary key, and it is used for the Return Record table, which records the return details of the rental car, such as the amount, return date and time, and tracks the condition of the car when return, including charges for extra services. Finally, the Notification entity stores messages sent to users (normal user and car owner), such as booking confirmations or other reminders updates. This allows the system to notify users about their bookings, payments, and other relevant updates by referring the notificationID.

Chapter 5

System Implementation

5.1 Software Setup

Before starting the development of a mobile application, we need to install some of the software to help us to design and test run the code such as Android Studio, Flutter SDK, Git and Firebase.

5.1.1 Android studio

Step 1: System Requirements

1. Ensure the computer system meets the requirements: Windows 64-bits version of Window 8/10, 8 GB RAM or more, 4 GB of available disk space, and 1 GB for Android Emulator.

Step 2: Download Android Studio from the search engine.

1. Go to the official website of https://developer.android.com/studio .
2. Click the Download Android Studio button.
3. Read and agree to the terms and conditions, then click Download.

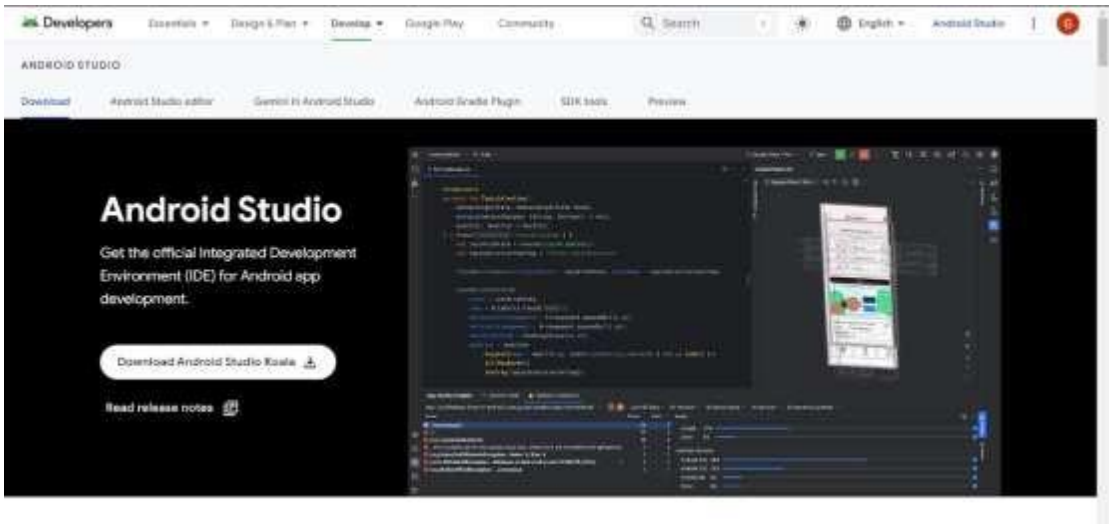


Figure 5.1 Download Android Studio

Step 3: Complete Installation

1. Click and launch the ".exe" file to install the Android Studio.
2. Wait for the installation to be finished, click "Next" it will start to run the Android Studio application.

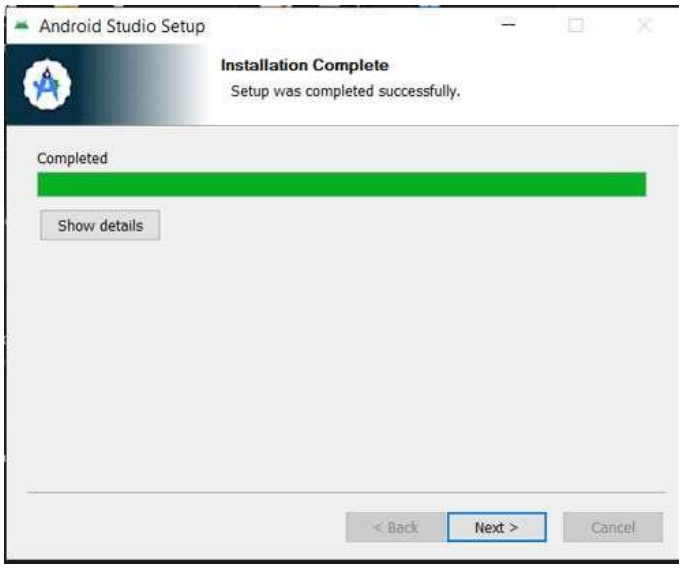


Figure 5.2 Android studio installation

Step 4: Start a New Project

1. Once the installation is complete, click on the "Start a new Android Studio project" to start developing the first Android application.

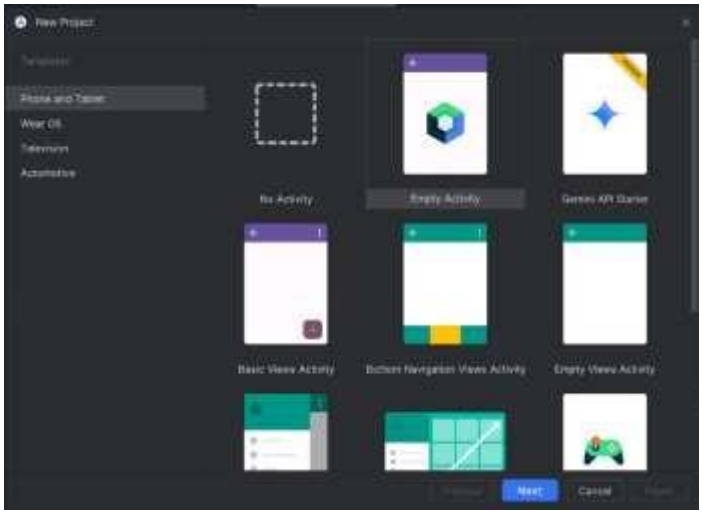


Figure 5.3 Main page of Android studio

5.2 Setting and Configuration

5.2.2 Firebase

1. Open the search engine to search https://console.firebase.google.com/ .
2. Open an account and select create a project to start a new project.
3. Go to the Android Studio and open a new project to connect with the firebase.
4. Go to the ‘Tools’ and select ‘Firebase’.
5. After that click on the ‘Connect your app to Firebase’, it will bring you back to the second step to connect to the selected project to make the connection.
6. When the connection is successful it will turn to green and show connected.
7. Next, click on the add the Firebase Authentication SDk to your app, it will automatically update in your project code.

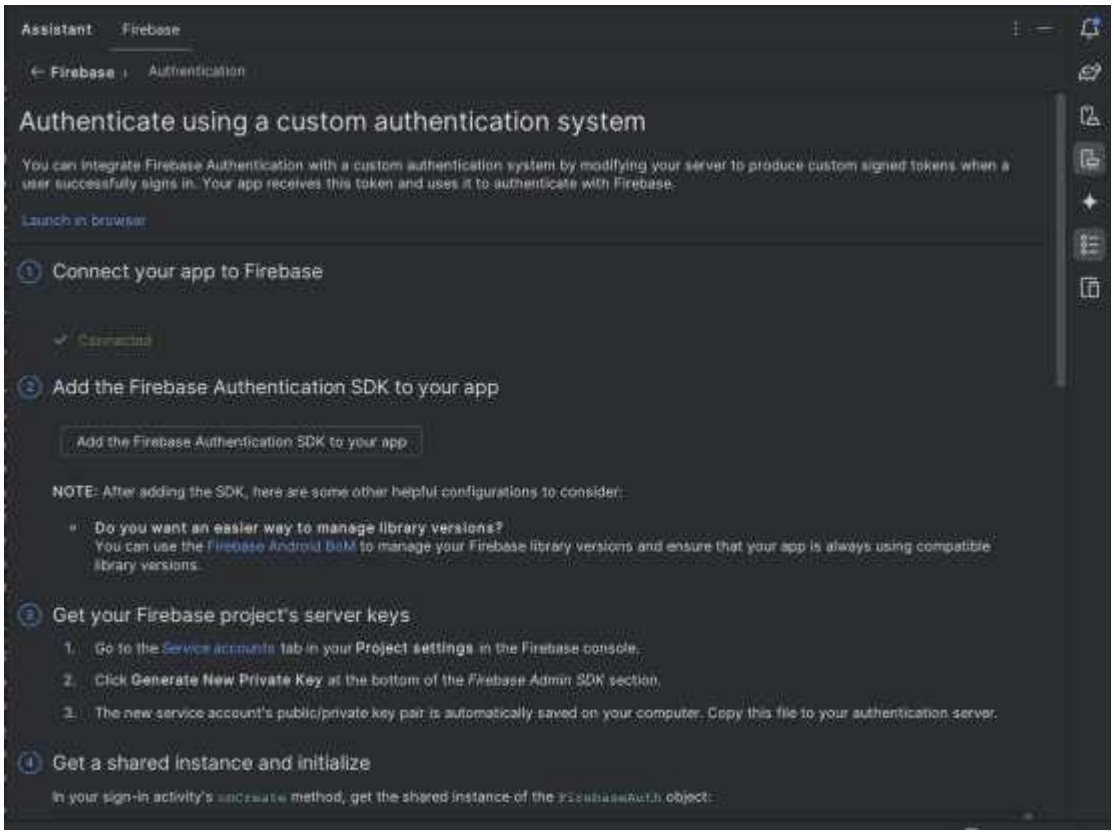


Figure 5.4 Firebase setup

5.3 System Operation

In this part, it will show the capture of the development design and used in car rental booking apps. It will include login activity, register activity, profile activity, and home activity.

5.3.1 Main Page

Figure 5.5 Main page

When the user launches the car rental application after installation in their mobile phone device. He or she will see a main page of the applications. In the main page users are allowed to click on the Login button at the top right corner of the page to login into the application.

5.3.2 Login Activity

Figure 5.6 Login page

When a user clicks on the login button at the main page, it will bring the user to the Login page. As a user of the app, he or she is required to sign in the account by email and password before login into the home page. As a new user, they can click on the " Don't have an account? SIGN UP" and will browse users to the register page to register their email and password. In case users forget their password, users can click on "Forgot Password" to receive the reset password link at their registered email.

5.3.3 Register Activity

Figure 5.7 Register page

As a new user, he or she needs to register an account by entering some of the following such as the user's email and password. All the data entered by the user will be saved in the firebase database. Each of the emails can be only registered one time in the database. Password security will be detected by the system as well if the user entered password is too short and didn't fulfil the requirements such as the password must be at least 8 characters and include a number, lowercase letter, uppercase letter, and symbol. After the user fills in all the information, users are allowed to click on the register button to register their account. If the user is registered successfully, it will pop out messages to give an alert message to the user. The user can click on the "Already have an account? LOGIN" to go back to the login page, users can click on this button to go back to the login page if the user accidentally goes to the register page.

5.3.4 Home activity

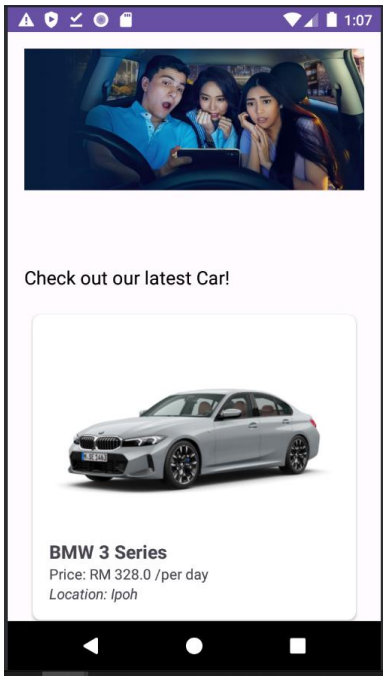


Figure 5.8 Home page

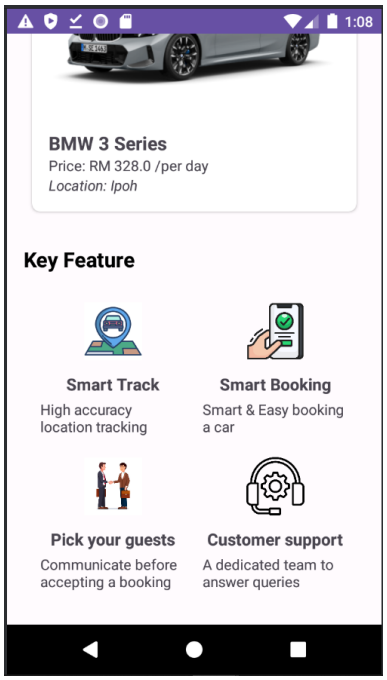


Figure 5.9 Home page (Key Feature)

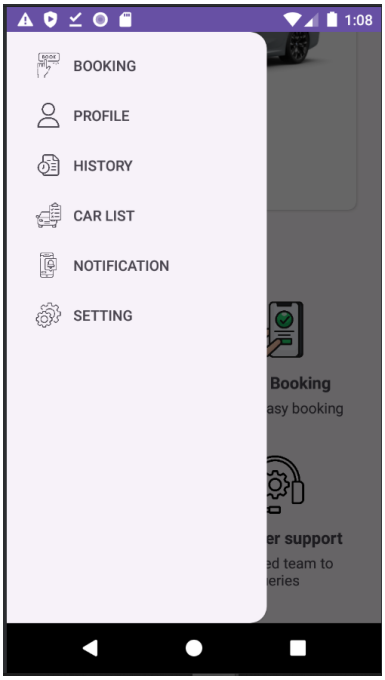


Figure 5.10 Home page (Menu)

In this page, all the cars will be displayed on the screen to allow the user to see, such as the car picture, category, price, model and brand will be displayed in the list to allow the user to see the car. There will be a menu that hides at the left side, users need to scroll from left to right to access the menu. In the menu, users are allowed to select which page they want to go to such as Booking page, Profile page, History page, Car List page, Setting Page.

5.3.5 Profile Activity

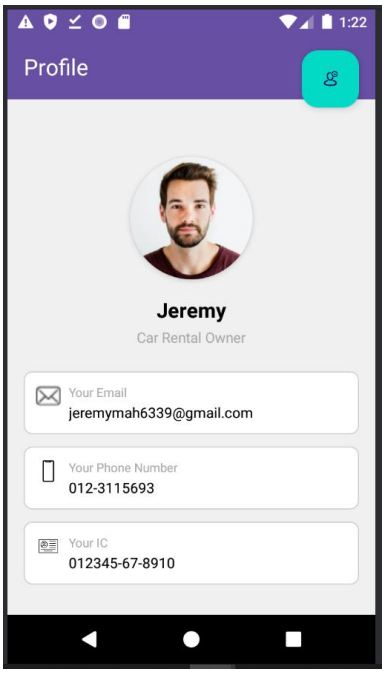


Figure 5.11 Profile page

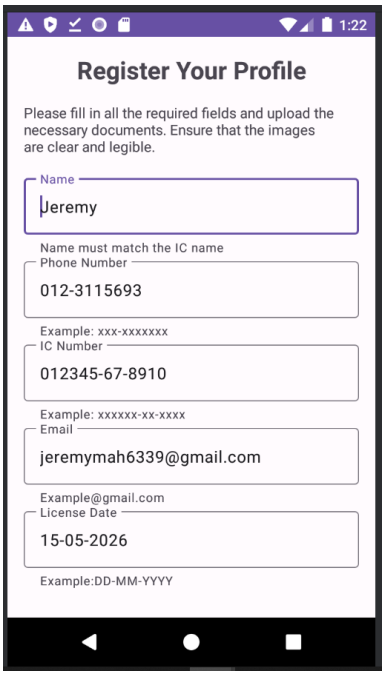


Figure 5.12 Registration page (1)

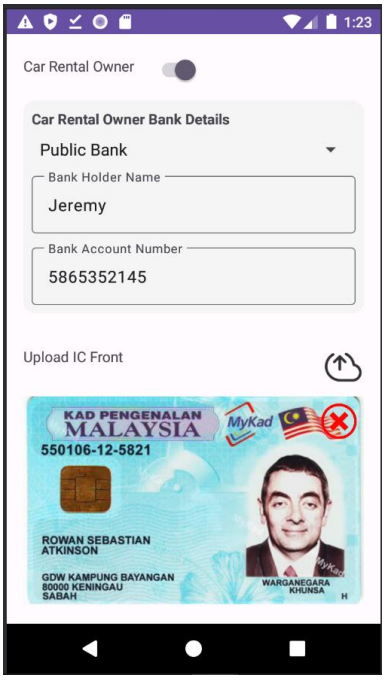


Figure 5.13 Registration page(3)

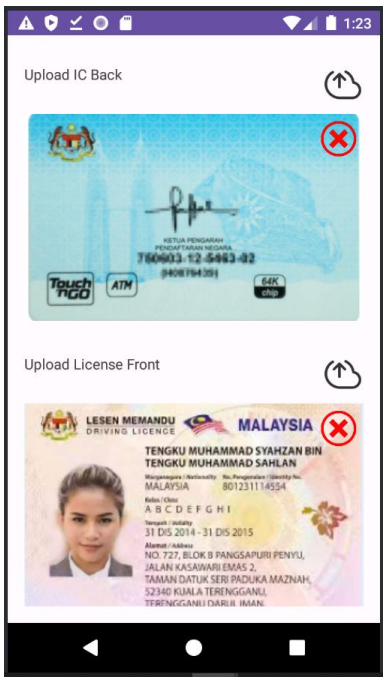


Figure 5.14 Registration page(3)

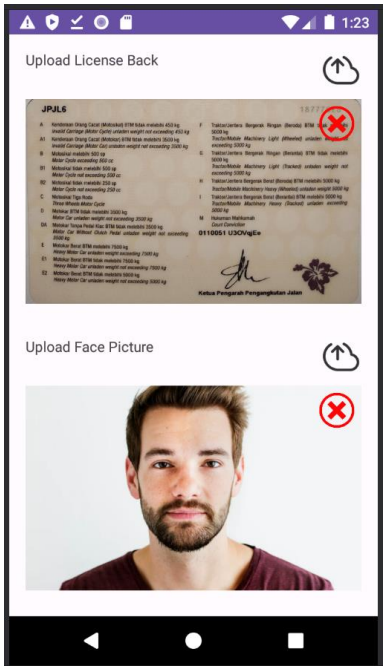


Figure 5.15 Registration page(4)

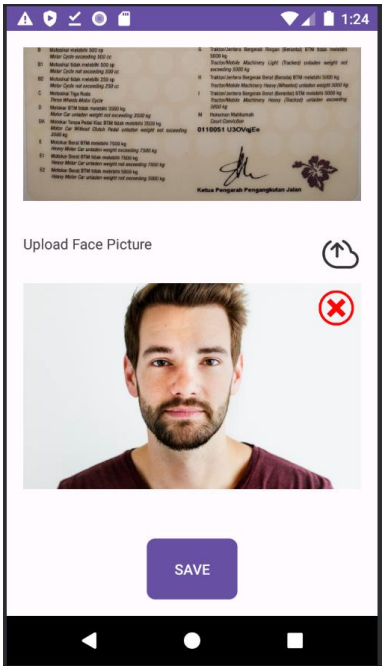


Figure 5.16 Registration page(5)

In this page, as a new user they need to register all their information in the registration page such as name, phone number, IC number, email, license expiration date, become a rental car owner or normal user, IC picture (front and back), license picture (front and back) and their face picture. There will be some of the rules that need to be followed when the user enters this information such as their name must be the same as their name on the IC card and only letters and spaces are allowed, phone number format: 000-0000000, IC number format: 000000-00-0000, Email must match the registered email, and the license expiry date must not be before now days. After the user fills all the information, users are allowed to click on the save button to save their information in the firebase database. After the registration, their face picture, name, phone number, ic number and email will be displayed on the profile page. If the user has registered before, they no need to register again at the register page, their data will automatically be retrieved from the firebase database and will be displayed automatically. If the user is selected to be a rental car owner, they can add their car to the car list for rent. While the user is registered as a normal user, they just can book the car.

5.3.6 Booking Activity

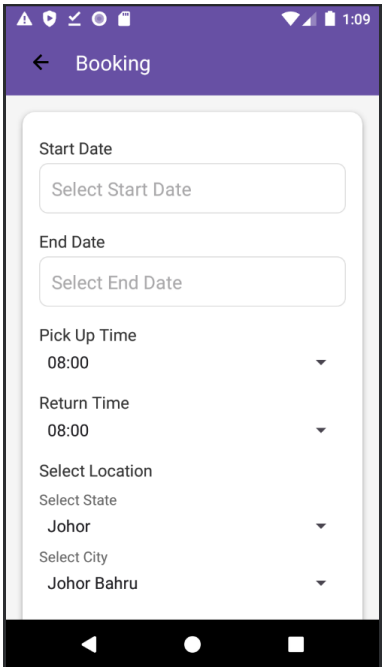


Figure 5.17 Booking Page(1)

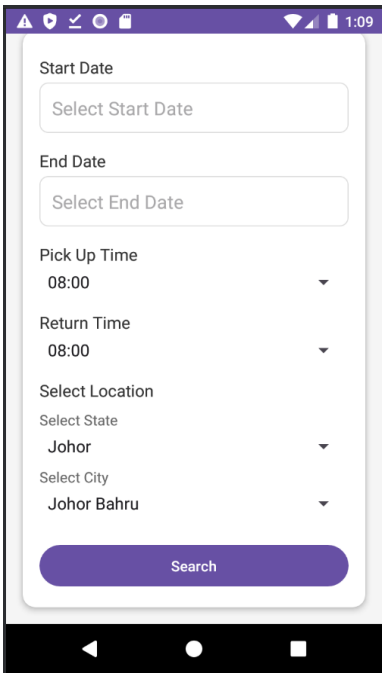


Figure 5.18 Booking Page(2)

In the booking page, it allows users to select the booking information such as Start Date, End Date, Pick Up time. Return Time, Location to find available car for renting. When user is click on the start date, it allows user to choose the start date when they want to start the renting car. While user is clicking on the return date, it will allow user to choose the date when they want to return the car. Similarly to the pickup time and return time, it allows user to pickup the car at the time they want and return the car they want. Not only that, but the pickup time also allow the car owner to prepare the car early for user and car owner can collect the car at the correct time. Besides that, user is allowed to choose the location such as state and city for find the available car to rent. User can click on the “Search” button when user is providing all the booking information, with this information, the system will display all the list of available car.

5.3.7 Available Car Activity

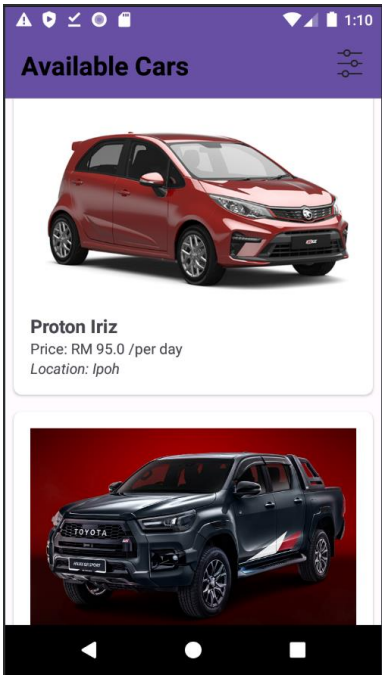


Figure 5.19 Available Car Page

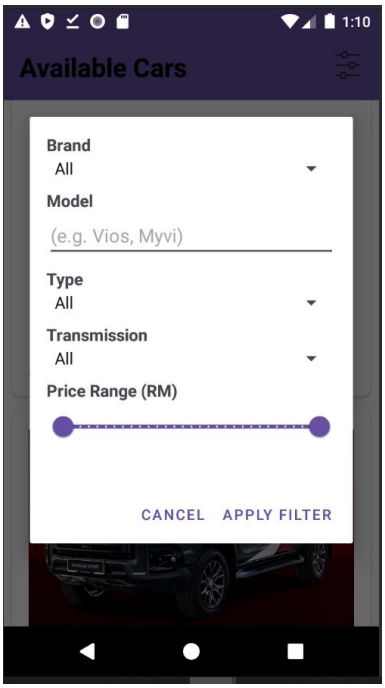


Figure 5.20 Pop out Filter

In the available car page, when user is provided the correct booking information such as start and end date, start and the system will check with the database to compare the information that provided by to user to list out the available car. When the system is finish compare, users can view the list of the available car in the page. Users are able to filter the car list by clicking the filter button at the top. The filter button in a car list allowed users to narrow down the list of available cars based on specific criteria such as filtering the brand, car type, transmission or even the car price.

5.3.8 Car List Activity

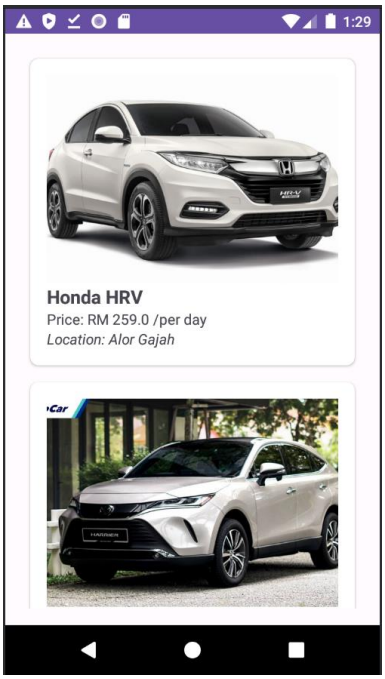


Figure 5.21 Car List Page

Figure 5.22 Car Details Page (1)

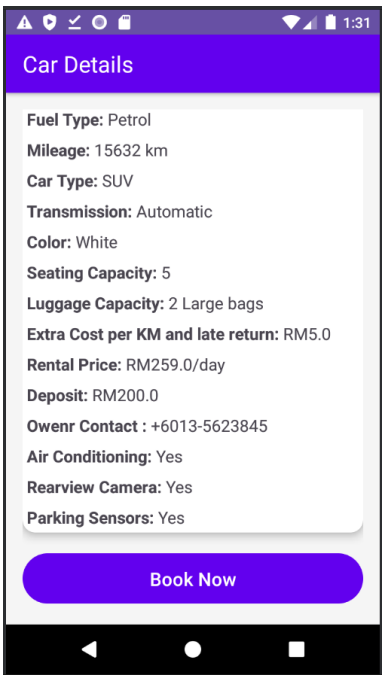


Figure 5.23 Car Details Page (2)

In the Car List page, the system will retrieve all the car list from the database that was added by the car owner. User is allowed to view all the car list in the Car List page. If user can want to view the details of car, user can click on the car container, so that user can view all the car details such as Brand, Model, Year, Fuel Type, Mileage, Car Type, Transmission, Colour, Seating Capacity, Luggage Capacity, Extra Cost per KM and Late return, Rental Price, Deposit, Owner Contact, Air conditioning, Rearview Camera, and Parking Sensors. If user want to book the car, User can click on the “Book Now” button to book the car.

5.3.9 Term and Conditions Activity

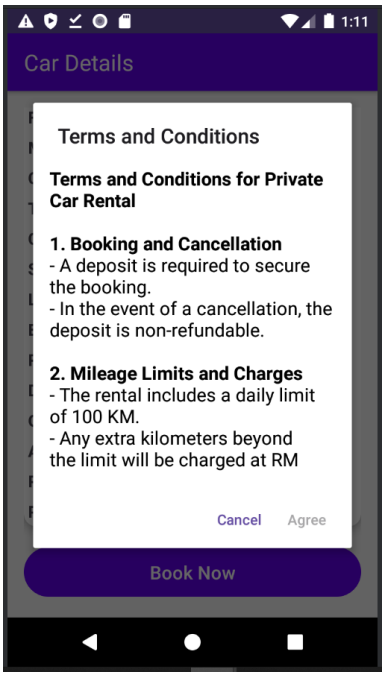


Figure 5.24 Term and Conditions (1)

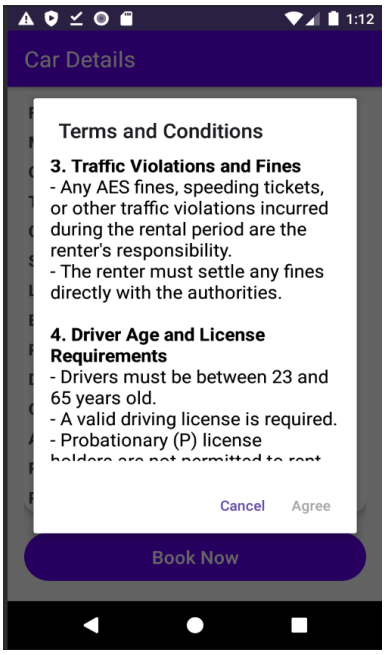


Figure 5.25 Term and Conditions (2)

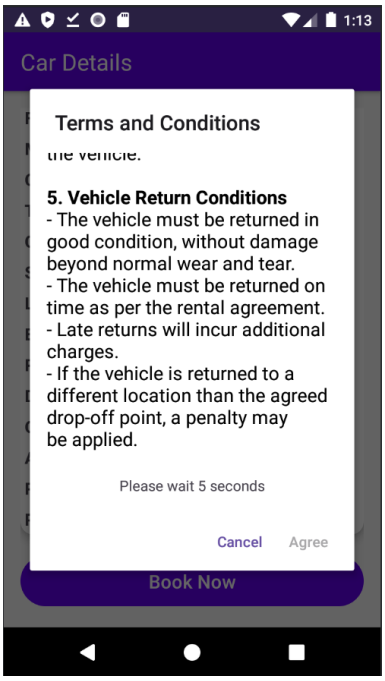


Figure 5.26 Term and Conditions (3)

In the term and conditions page, users need to read the term and conditions carefully before processing the payment to book the car. Its primary function is to inform users about important policies they need to agree to before proceeding with the rental. It outlines key details such as booking and cancellation terms, Mileage Limits and Charges, Traffic Violation and Fines, Driver Age and License Requirements and Vehicle Return Conditions. This step not only helps users make informed decisions but also provides legal protection for the car owner by confirming the user's agreement to the terms before allowing them to complete the booking. When the user is finished reading the term and conditions and after 10 seconds the user only can click on the Agree button. If the user agrees with the terms and conditions, it can process the booking payment, while users do not agree are not able to process the booking payment.

5.3.10 Payment Activity

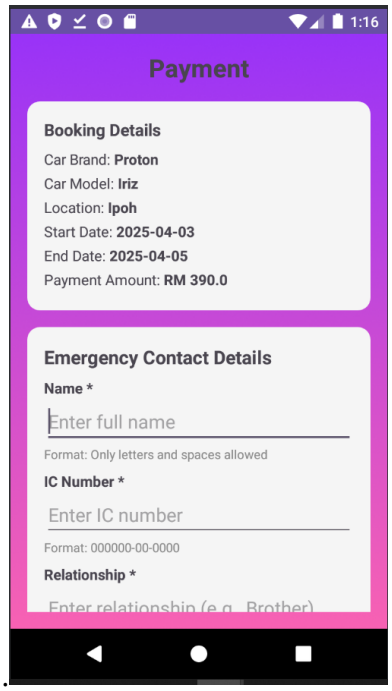


Figure 5.27 Payment page (1)

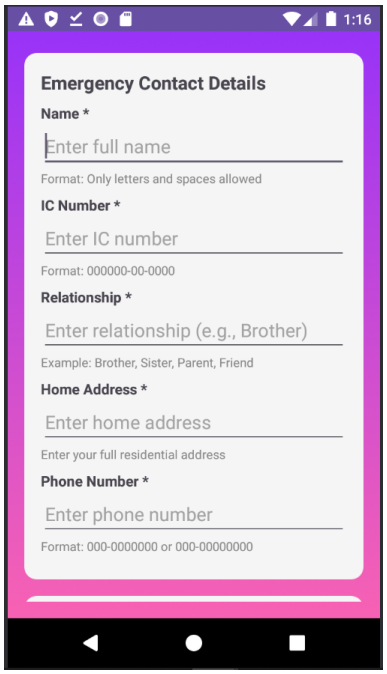


Figure 5.28 Emergency Contact Details

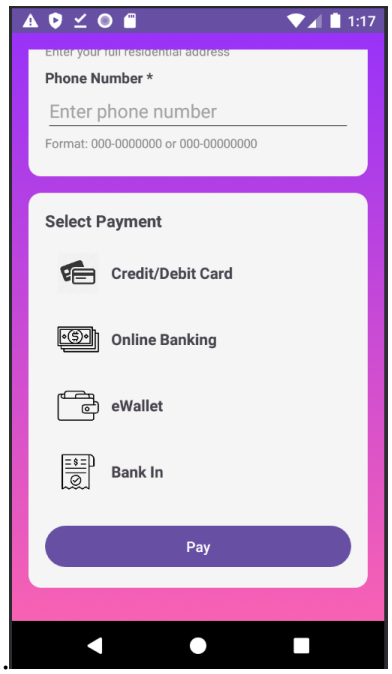


Figure 5.29 Payment Method

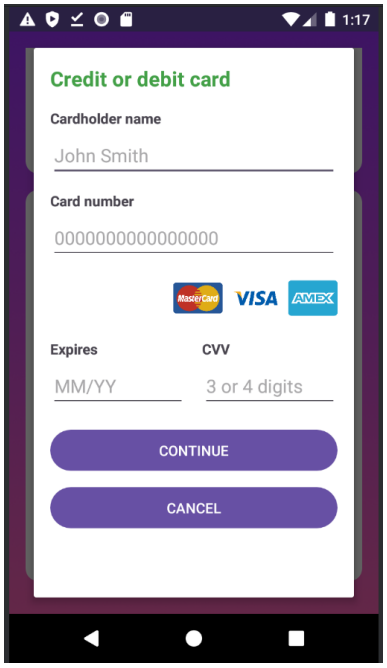


Figure 5.30 Credit Card Details

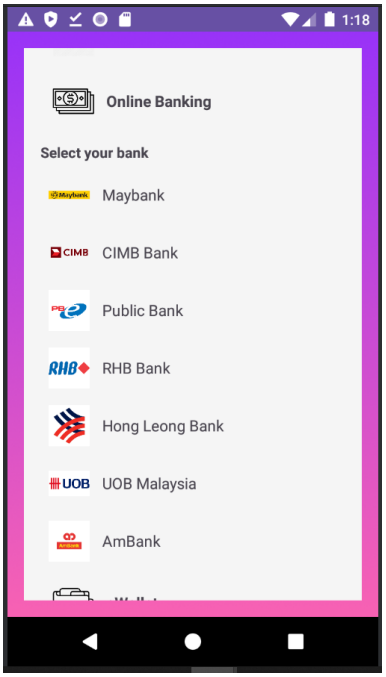


Figure 5.31 Online Banking Selection

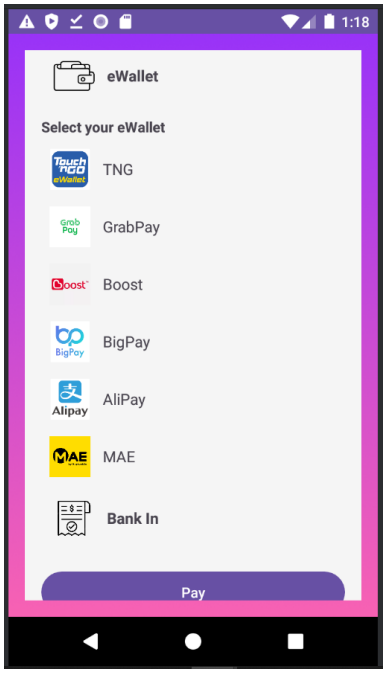


Figure 5.32 E-wallet Selection

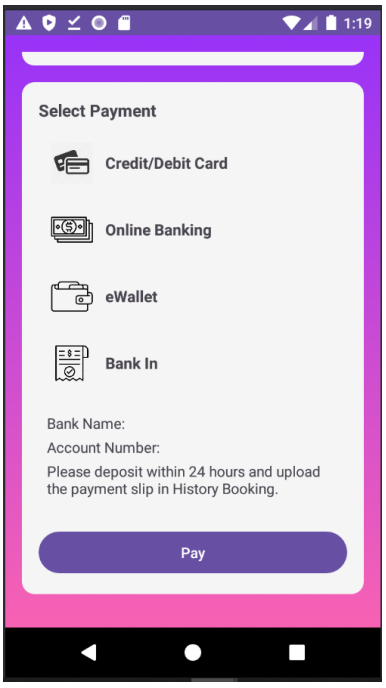


Figure 5.33 Bank in Details

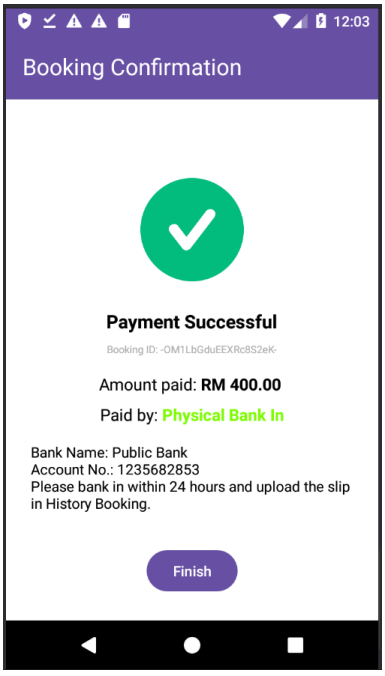


Figure 5.34 Confirmation Details (1)

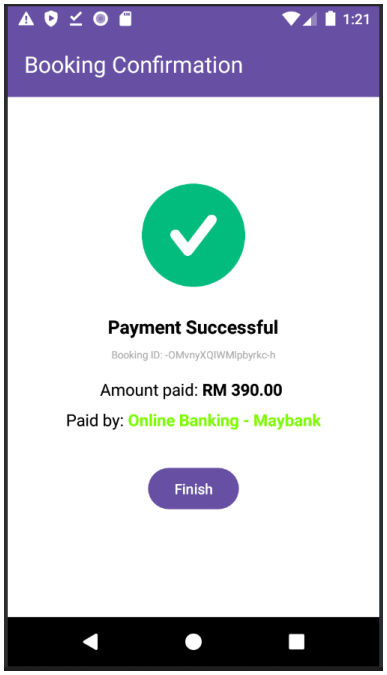


Figure 5.35 Confirmation Details (2)

In the payment page, user is able to view all the payment details such as Booking Details, Emergency Contact Details, select the payment method. In the booking details user able to view what car the user wants to book, location, start and end date and the total payment amount need to pay. In additional, users are needed to fill in the all the emergency contact details such as emergency contact name, emergency contact IC number, emergency contact relationship, emergency contact home address and emergency contact phone number. There will be some of the rules that need to be followed when the user enters this information such as their name must be the same as their name on the IC card and only letters and spaces are allowed, phone number format: 000- 0000000, IC number format: 000000-00-0000, relationship must be real and home address same as their address on the IC card. Lastly, users can select the payment method such as credit/debit card, online banking, e-wallet or bank in to process the payment. After the user is successful paying the payment, it will display the booking confirmation page. In the booking confirmation page, if user is paid by credit/debit card, online banking or e-wallet the booking confirmation message will display the booking id, amount paid and payment method while user using bank in the booking confirmation message will display booking id, amount paid, payment method and account information of the car owner.

5.3.11 History Activity

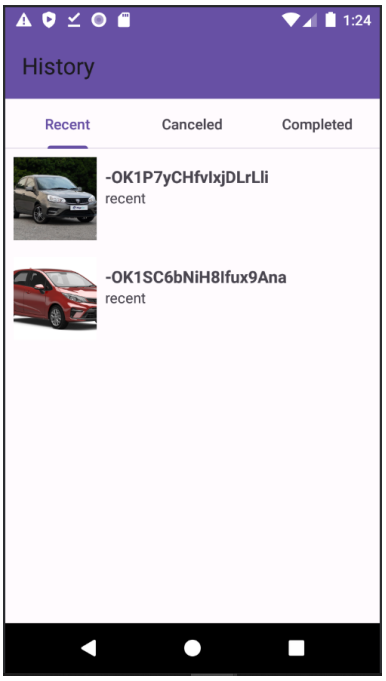


Figure 5.36 History Page (Recent)

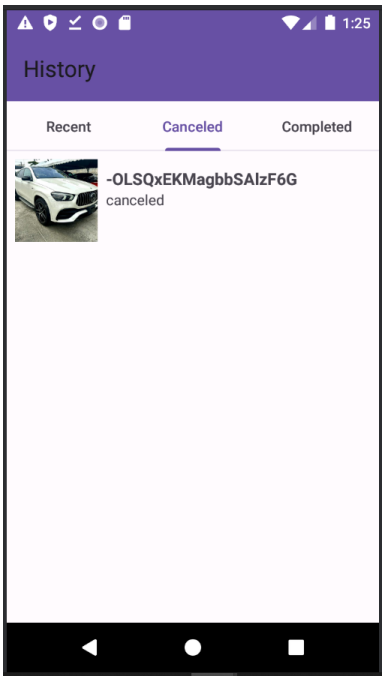


Figure 5.37 History Page (Canceled)

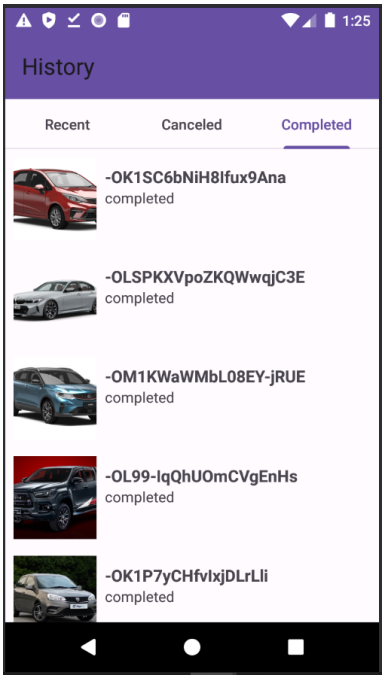


Figure 5.38 History Page (Completed)

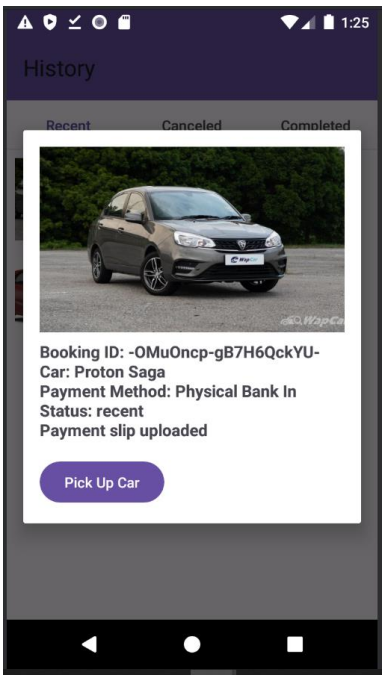


Figure 5.39 History Page (Pickup Car)

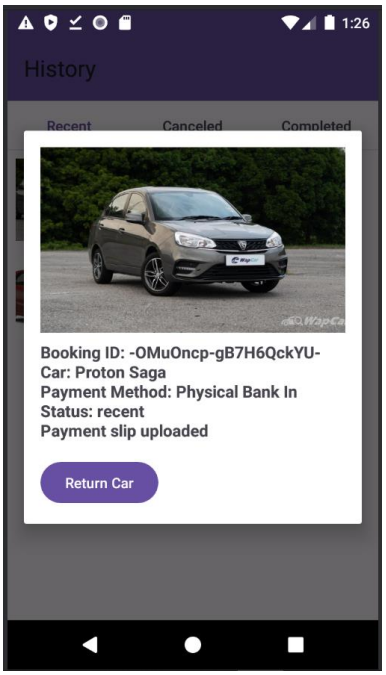


Figure 5.40 History Page (Return Car)

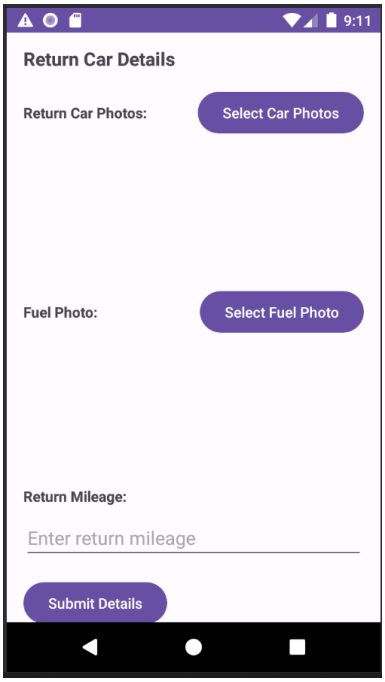


Figure 5.41 Return Car Details Page

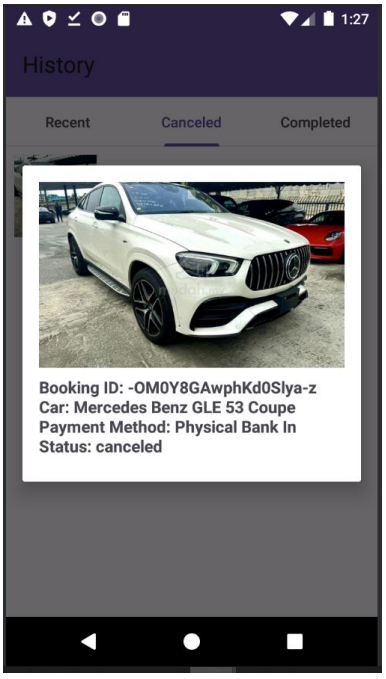


Figure 5.42 History Pop out Details (1)

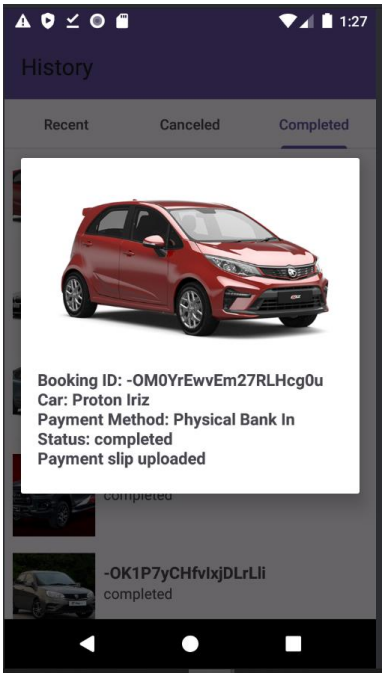


Figure 5.43 History Pop out Details(2)

In the history page, the user can view all the booking history in the different status which is “Recent”, “Cancelled”, and “Completed”. These three statuses allow user easier to view the booking history details. In the recent status, user can view the latest ongoing booking history details such as user can check the booking ID, Car Name, Payment method, Car Status and the additional function, which is uploading payment slip, pick up the car, and return the car. If the user is using the bank in payment method, it will need to upload the payment slip before it can pick up the car. User who is using the bank in payment method which they need to upload the payment slip within 24 hours, if the user fails to upload the payment slip the booking will automatically turn to the cancelled status. When user is ready pick up the car it needs to be clicking on the pickup car button to give notification to car owner and update the car status. After the user return the car it needs to click on the return car button to fill in the return car details such as the image of condition of the return car, the fuel of the car, and the return mileage to update car owner. After car is returned to car owner. The system will automatically calculate the refund or the penalty that need to be return to user or user need to be pay. The penalty includes late return or extra mileage used. While in the cancelled status, user is allowed to view the booking that has been cancelled. Lastly, in the completed page it allows user to view the booking that has been completed.

5.3.12 Notifications Activity

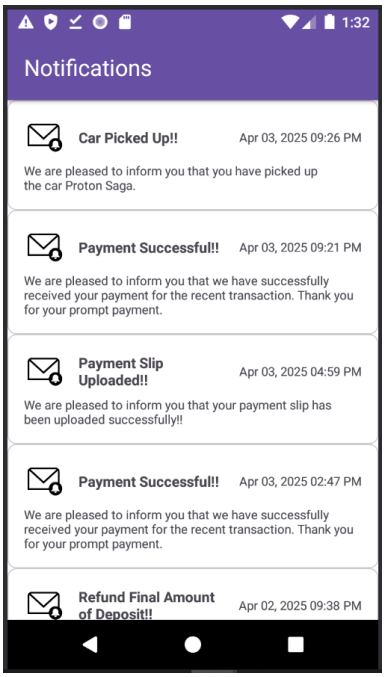


Figure 5.44 Notifications Page

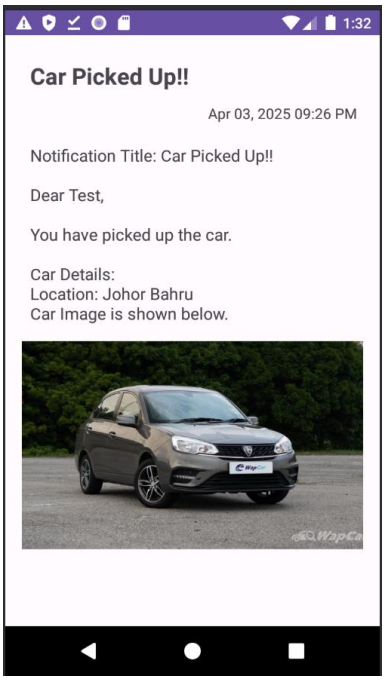


Figure 5.45 Car Picked up

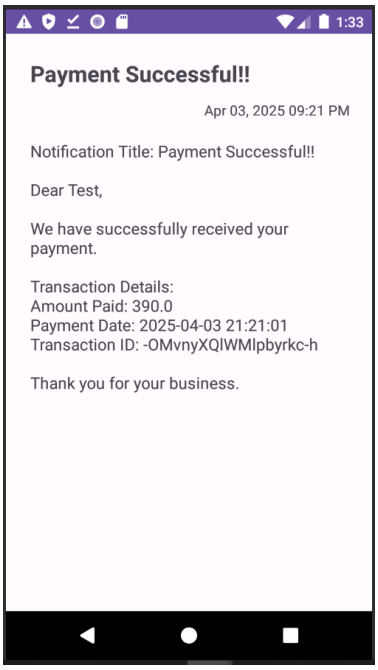


Figure 5.46 Payment Successful

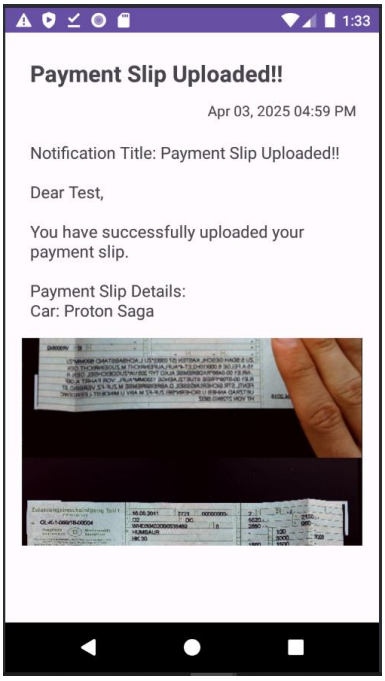


Figure 5.47 Payment Slip Uploaded

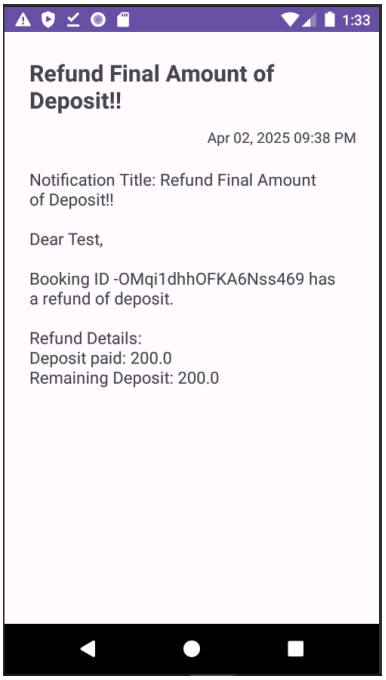


Figure 5.48 Refund

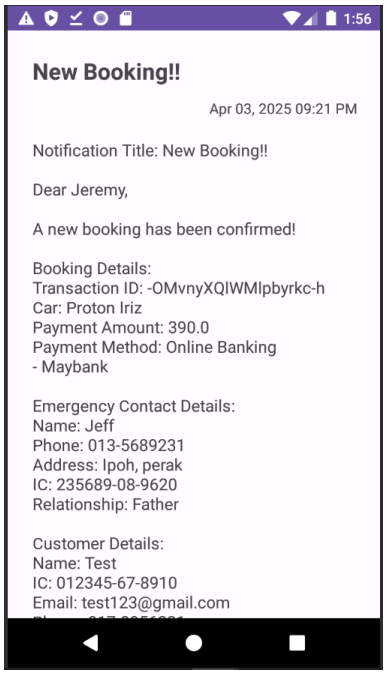


Figure 5.49 New Booking

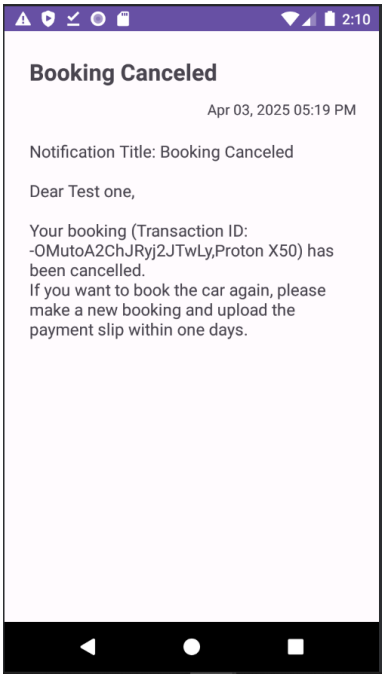


Figure 5.50 Booking Canceled

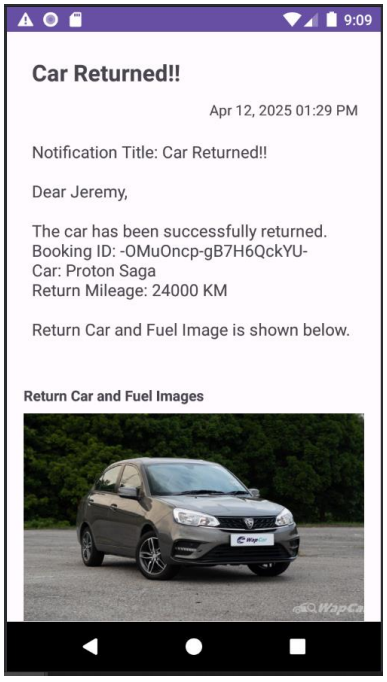


Figure 5.51 Car Returned

In the notification page, system will display a list of important updates related to the user's car rental transactions, such as “Car Picked Up”, “Payment Slip Uploaded”, “Refund Final Amount of Deposit”, “New Booking”, Booking Cancelled”, “Car Return”, and Extra Amount need to be Pay”. Users are allowed to view the list of message content by clicking on the notification message. In the notification details, users are allowed to view the notification that generates by the system based on the type of event detected in the title. When in "payment successful" notification, the message informs the customer that their payment was received successfully, and including the details such as the total amount, date, payment method, and booking ID. For a "new booking" notification, the car owner is notified with full booking details, such as booking ID, car details, payment amount, and emergency contact details of the customer. If the notification is "payment slip uploaded", it sends a message to the car owner and customer to confirming that the payment slip was uploaded and car details. When the notification is "car picked up", it confirms that the car has been picked up and includes the location and a photo of the car to the car owner and customer. In the case of a "refund final amount of deposit", the message informs either party about the deposit and the amount refunded. For "extra amount need to pay", the system notifies the customer and owner about additional charges with the amount and reason. If a "booking is cancelled", it explains to the customer that the booking was cancelled due to a late payment slip. Lastly, if the notification is "car returned", it confirms that the car was successfully returned, shows the return mileage, and includes a return car and fuel photo.

5.3.13 Setting Activity

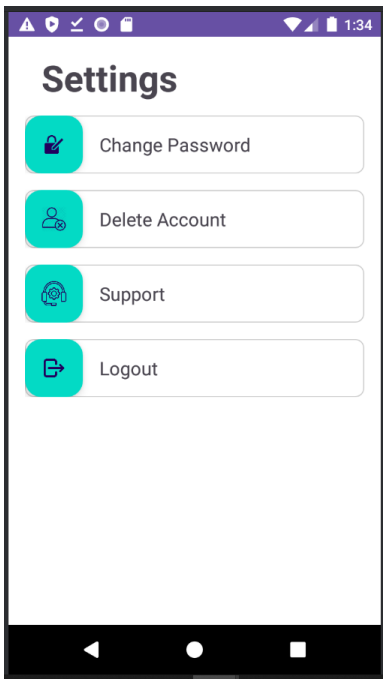


Figure 5.52 Setting page (Normal User)

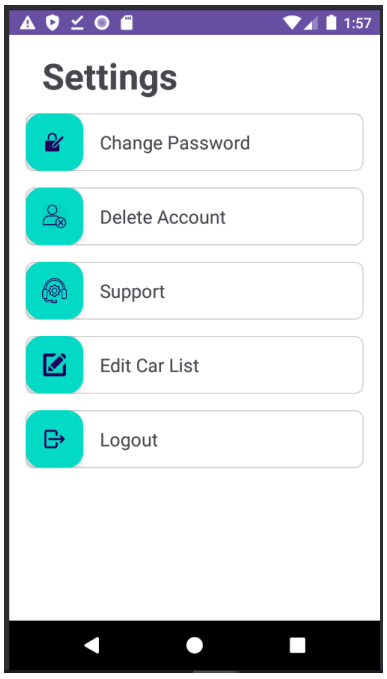


Figure 5.53 Setting page(Owner)

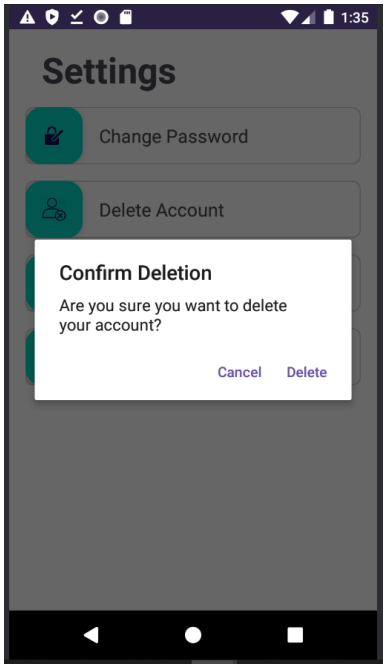


Figure 5.54 Pop out message (Delete Account)

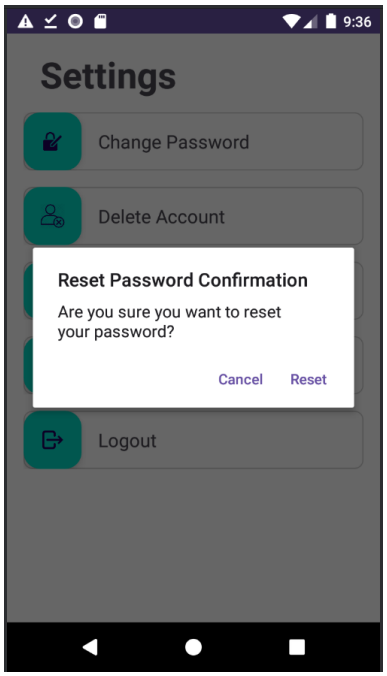


Figure 5.55 Pop out message (Reset Password)

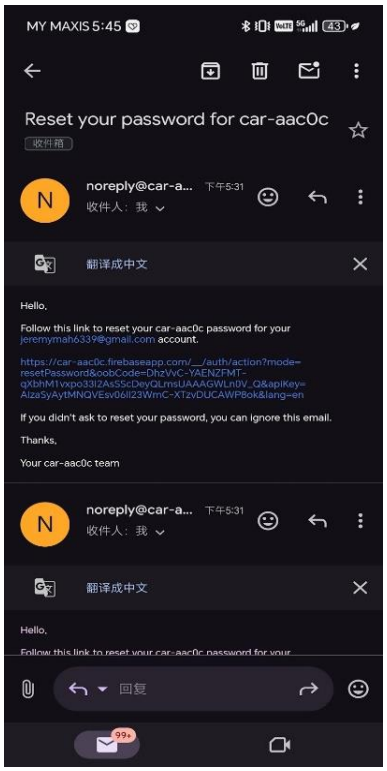


Figure 5.56 Reset Password Email

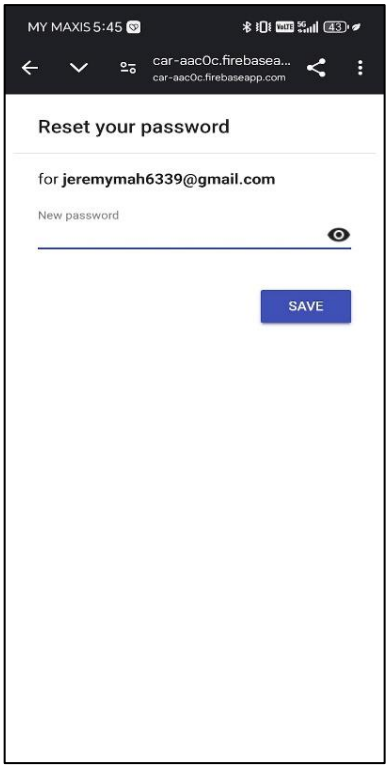


Figure 5.57 Reset Password Page

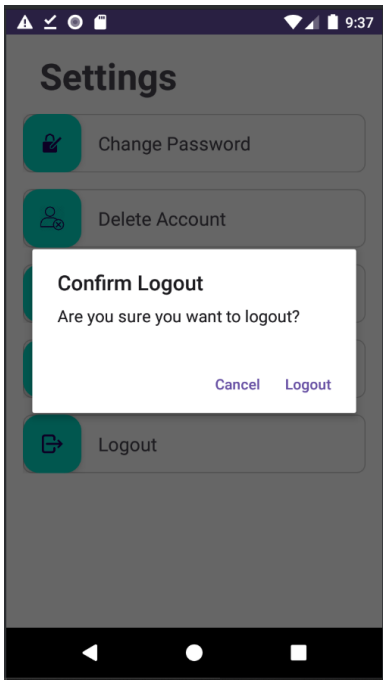


Figure 5.58 Logout Page

In the settings page users can change their account password, delete their account, to get a support or logout and if they are car owner it will have extra button which is “Edit Car List” button to allow user to add new car, delete car and modify car information. When the user clicks on “Change Password” to reset a new password, the users will receive a reset password link in their registered email account. Users will need to click on the link to reset the new password. When the user clicks on the “Delete Account”, it will pop out a confirmation message to users to double confirm to delete the account. If a user clicks on the Delete button it will delete the account and go back to the main page, while user clicks on the cancel button it will not do anything. This pop out message will prevent accidental account deletion, ensure thoughtful actions, and enhance user security. Similarly, when the user clicks on the Logout button, a pop out confirmation alert is displayed asking user whether they really want to log out the app. If they select “Logout,” the app signs them out, clears the activity back stack, and returns them to the main page, while user choosing “Cancel” it will dismiss the logout process. If the user clicks on the support button, it will allow users to quickly access help or customer service. It will provide direct assistance for issues, questions, or guidance, to the user who needs help at the time to improve user experience by offering a convenient way to resolve problems or get information. The extra function button will provide to the car owner only which allow user to view the car list, add new car, delete car and modify car information.

5.3.14 Edit Car Activity

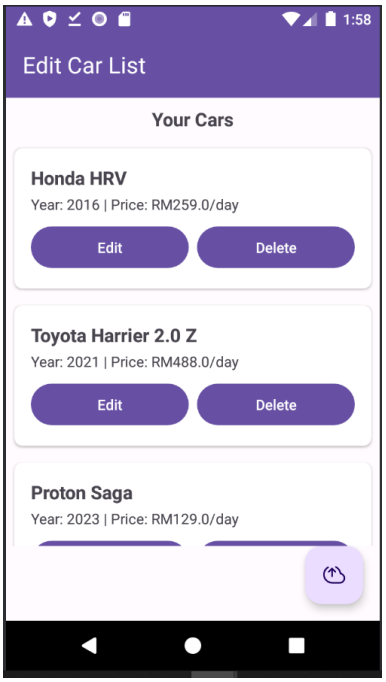


Figure 5.59 Edit Car List Page

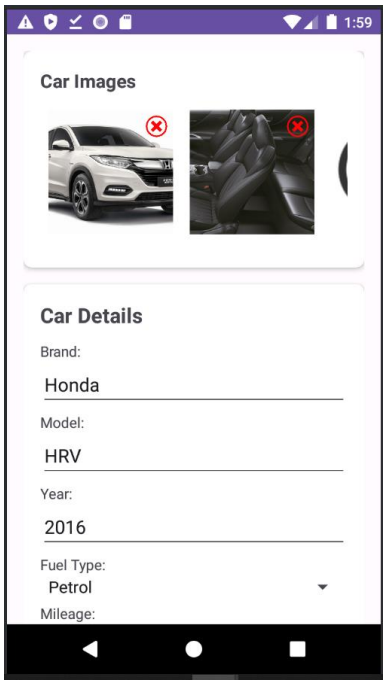


Figure 5.60 Edit Car Details (1)

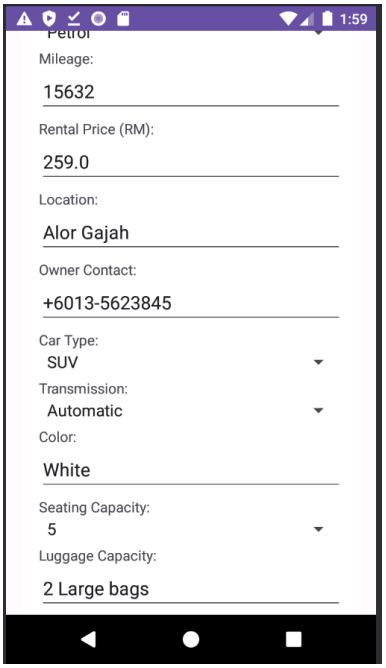


Figure 5.61 Edit Car Details (2)

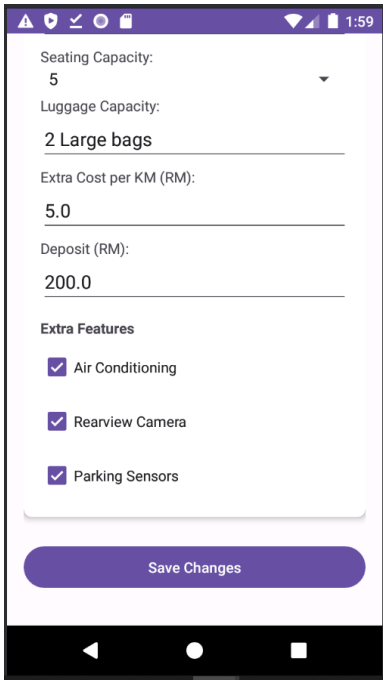


Figure 5.62 Edit Car Details (3)

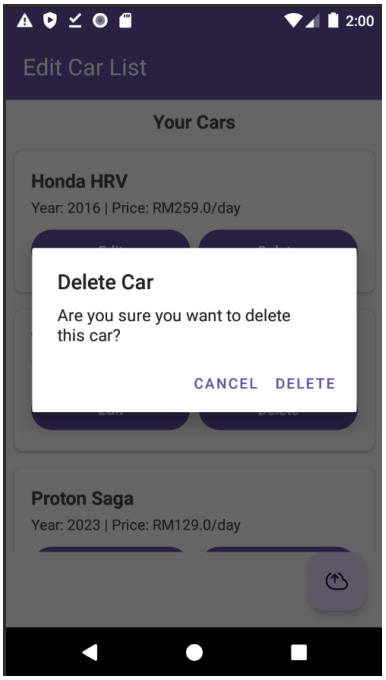


Figure 5.63 Delete Car

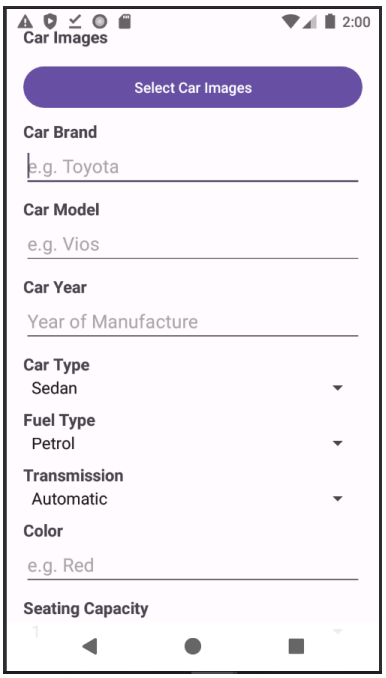


Figure 5.64 Add Car Page(1)

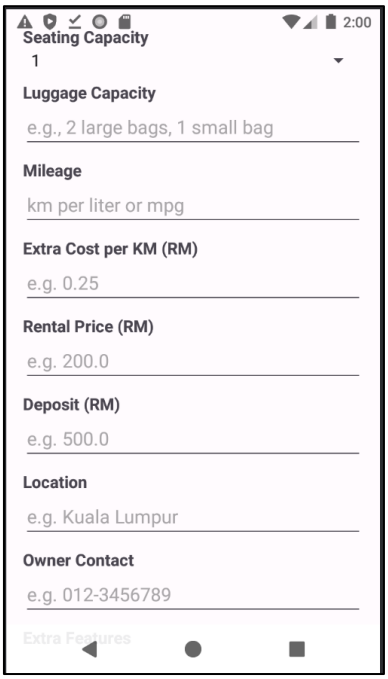


Figure 5.65 Add Car Page(2)

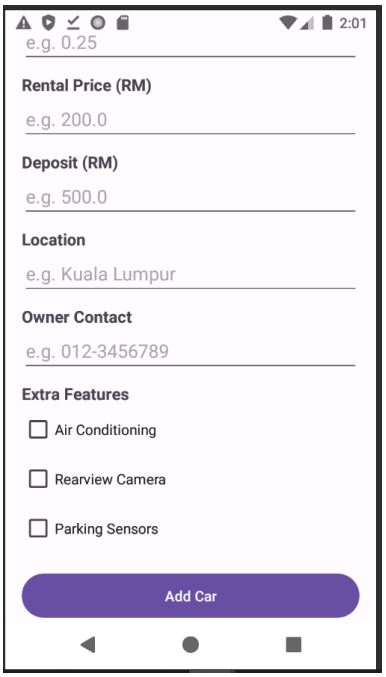


Figure 5.66 Add Car Page(3)

In the edit car list page, car owner can view the car list that added to the database. In the car list, user can view the normal car information such as car name, year and price to identify the car. In additional, car owner can add the car by clicking on the add button at the bottom. When user is clicked the add button, it will need owner to fill the car details such as car image, car brand, car model, year, fuel type, mileage, rental price, location, owner contact, car type, transmission, colour, seating capacity, luggage capacity, extra cost per km, deposit, and extra feature. After the owner is fill in all the car details, owner can click on the add car button to save in the database. Besides that, owner can edit the car details by clicking on the edit button to edit the car details such as car image, car brand, car model, year, fuel type, mileage, rental price, location, owner contact, car type, transmission, colour, seating capacity, luggage capacity, extra cost per km, deposit, and extra feature. After the owner finish edit the car details, owner can save the car details by clicking the save change button. If the owner wants to delete the car, owner can click on the delete button to delete the car from the database. When owner clicks on the delete button it will show the pop out confirmation message asks the owner if they are sure they want to delete the selected car. When the owner clicks on "CANCEL" it will stop the deletion process while owner clicks on "DELETE" to confirm and remove the car from the list. This confirmation step is important to prevent accidental deletions and ensure users have full control over their car listings.

5.3.15 Support Activity

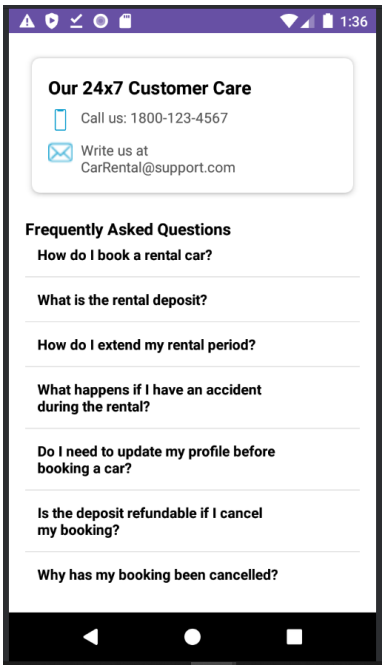


Figure 5.67 Support Page (1)

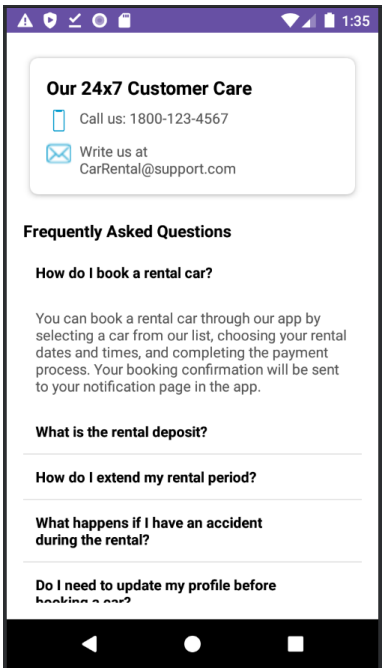


Figure 5.68 Support Page (2)

In the support page, users can directly click on the “call us” to get an on-time support by calling the phone number or click on the email to give feedback. In addition, users can get support by frequently asked questions, allowing users to find quick answers to common concerns. When user taps on a question, such as “How do I book a rental car?”, the system expands to show a detailed answer to user, explaining how to book process step-by-step. This feature helps users get immediate assistance without needing call or email support, improving the overall user experience by making important information easily accessible within the app.

Chapter 6

System Evaluation and Discussion

6.1 System Testing & Result

In this chapter, it will conduct various test cases that will be executed to ensure that the application runs smoothly and as per the requirements.

Table 6.1 Sign Up Page Test Case

Sign Up Test Case				
Role : New User				
No	**Test Cases**	**Expected Results**	**Actual Results**	**Pass/Fail**
1.	Click Sign Up without fill in any input fields	Error message conduct cannot be blank.	Email is required.	Pass
2.	Click Sign Up without fill in email.	Error message conduct email cannot be blank.	Email is required.	Pass
3.	Click Sign Up without fill in password.	Error message conduct password cannot be blank.	Password is required.	Pass
4.	Click Sign Up without fill in confirm password.	Error message conduct confirm password cannot be blank.	Confirm Password is required.	Pass
5.	Use email address already registered.	Error message conduct email has been registered.	The email address is already in use by another account.	Pass
6.	Click Sign Up without 8 characters of password.	Error message conduct password format.	Password must be at least 8 characters and include a number, lowercase letter, uppercase letter and symbol.	Pass
7.	Click Sign Up without including a number in the password.	Error message conduct password format.	Password must be at least 8 characters and include a number, lowercase letter, uppercase letter and symbol.	Pass
8.	Click Sign Up without including an uppercase in the password.	Error message conduct password format.	Password must be at least 8 characters and include a number, lowercase letter, uppercase letter and symbol.	Pass

9.	Click Sign Up without including a lowercase in the password.	Error message conduct password format.	Password must be at least 8 characters and include a number, lowercase letter, uppercase letter and symbol.	Pass
10.	Click Sign Up without including a symbol in the password.	Error message conduct password format.	Password must be at least 8 characters and include a number, lowercase letter, uppercase letter and symbol.	Pass
11.	Click Sign Up with different password and confirm password.	Error message conduct confirm password do not match.	Password do not match.	Pass
12.	Sign Up with valid email, password and confirm password.	Show successfully register message	Registration successfully	Pass
13.	Click hypertext Already have an Account.	Back to Login Page	Back to Login Page	Pass
14.	Click hypertext Sign Up	Back to Login Page	Back to Login Page	Pass

Table 6.2 Log In page Test Case

Log In Test Case				
Role : Current User				
No	**Test Cases**	**Expected Results**	**Actual Results**	**Pass/Fail**
1.	Click hypertext Login.	Forward to Login Page.	Forward to Login Page.	Pass
2.	Click Login without fill in any input fields.	Error message conduct input field cannot be blank.	An error occurred. Please enter your email.	Pass
3.	Click Login without fill in email fields.	Error message conduct input field cannot be blank.	An error occurred. Please enter your email.	Pass
4.	Click Login without fill in password fields.	Error message conduct input field cannot be blank.	An error occurred. Please enter your password.	Pass
5.	Use Invalid email address.	Error message conduct invalid email.	An error occurred. Login Failed.	Pass
6.	Use Invalid password address.	Error message conduct invalid password.	An error occurred. Login Failed.	Pass

7.	Click hypertext Don't have an account.	Forward to Register page.	Forward to Register page.	Pass
8.	Click hypertext Forgot Password.	Forward to Reset Password page.	Forward to Reset Password page.	Pass
9.	Login using correct email and password.	Successful login to system.	Login to Home page.	Pass

Table 6.3 Reset Password Page Test Case

Reset Password Test Case				
Role : Current User				
No	**Test Cases**	**Expected Results**	**Actual Results**	**Pass/Fail**
1.	Click Reset using invalid email format.	Show error message conduct invalid email format.	Invalid Email format.	Pass
2.	Click Reset using invalid email	Show error message conduct invalid email.	Invalid Email.	Pass
3.	Click Reset using valid email.	Show reset email link have been sent.	Password reset link sent.	Pass
4.	Click Reset without fill in email.	Show error message conduct email input field cannot be blank.	An error occurred. Pease enter your reset email to rest the password.	Pass

Table 6.4 Home Page Test Case

Home Page Test Case				
Use Case : Forward to Booking Page, Profile Page, History Page, Car List Page, Notification Page, Setting Page, Car Details Page				
Role : Current User				
No	**Test Cases**	**Expected Results**	**Actual Results**	**Pass/Fail**
1.	Click latest car display.	Forword to Car details page.	Forword to Car details page.	Pass
2.	Click selection "Booking".	Forword to Booking page.	Forword to Booking page.	Pass
3.	Click selection "Profile".	Forword to Profile page.	Forword to Profile page.	Pass
4.	Click selection "History".	Forword to History page.	Forword to History page.	Pass
5.	Click selection "Car List".	Forword to Car List page.	Forword to Car List page.	Pass
6.	Click selection "Notifications".	Forword to Notifications page.	Forword to Notifications page.	Pass
7.	Click selection "Setting".	Forword to Setting page.	Forword to Setting page.	Pass

Table 6.5 Booking Page Test Case

Booking Page Test Case				
Use Case : Add booking				
Role : Current User				
No	**Test Cases**	**Expected Results**	**Actual Results**	**Pass/Fail**
1.	Click Search button without fill in input field.	Error message as missing field.	Error message. Invalid date selection.	Pass
2.	Click Search button without fill in start date input field.	Error message as missing field.	Error message. Please fill the all field.	Pass
3.	Click Search button without fill in end date input field.	Error message as missing field.	Error message. Please fill the all field.	Pass
4.	Click Search button without select pick up time input field.	Error message as missing field.	Error message. Please fill the all field.	Pass
5.	Click Search button without select return time input field.	Error message as missing field.	Error message. Please fill the all field.	Pass
6.	Click Search button without select state input field.	Error message as missing field.	Error message. Please fill the all field.	Pass
7.	Click Search button without select city input field.	Error message as missing field.	Error message. Please fill the all field.	Pass
8.	Select start date with invalid date.	Error message as invalid date.	Error message. Invalid start date selection.	Pass
9.	Select end date with invalid date.	Error message as invalid date.	Error message. Invalid end date selection.	Pass
10.	Select start date and without end date.	Error message as invalid date.	Error message. Invalid date selection.	Pass
11.	Select with correct start and end date.	Forward to available car page.	Forward to available car page.	Pass

Table 6.6 Available Car Page Test Case

Available Car Page Test Case				
Use Case : Select Available Car, Apply Filter				
Role : Current User				
No	**Test Cases**	**Expected Results**	**Actual Results**	**Pass/Fail**
1.	Successful retrieve data from car database.	Display the car list.	Display the car list.	Pass
2.	Unsuccessful retrieve data from car database.	Error message as no car data found.	Error message as no car data found.	Pass
3.	Click “Filter” icon.	Pop out filter selection.	Pop out filter selection.	Pass
4.	Click “Apply Filter” button without fill in input field.	Display all car lists.	Display all car lists.	Pass
5.	Click “Apply Filter” button with brand selected input field.	Display the car based on the brand selected.	Display the car based on the brand selected.	Pass
6.	Click “Apply Filter” button with model selected input field.	Display the car based on the model selected.	Display the car based on the model selected.	Pass
7.	Click “Apply Filter” button with car type selected input field.	Display the car based on the car type selected.	Display the car based on the car type selected.	Pass
8.	Click “Apply Filter” button with transmission selected input field.	Display the car based on the transmission selected.	Display the car based on the transmission selected.	Pass
9.	Click “Apply Filter” button with price range selected input field.	Display the car based on the price range selected.	Display the car based on the price range selected.	Pass
10.	Click car container.	Display car details.	Display car details.	Pass

Table 6.7 Car Details Page Test Case

Car Details Page Test Case				
Use Case : Display Car Details, Booking a Car, View Term and Conditions				
Role : Current Users				
No	**Test Cases**	**Expected Results**	**Actual Results**	**Pass/Fail**
1.	Successful retrieve car details from car database.	Display car details.	Display car details.	Pass
2.	Unsuccessful retrieve car details from car database.	Error message as no car data found.	Error message as no car data found.	Pass
3.	Click “Book Now” Button.	Display term and conditions pop out container.	Display term and conditions pop out container.	Pass
4.	Scrolling the term and conditions details to end.	Allow system to run 10 second countdown for active the agree button.	Allow system to run 10 second countdown for active the agree button.	Pass
5.	Click “Cancel” button on the term and conditions pop out container.	Close the term and conditions pop out container.	Close the term and conditions pop out container.	Pass
6.	Click “Agree” button on the term and conditions pop out container.	Forward to Payment page.	Forward to Payment page.	Pass

Table 6.8 Payment Page Test Case

Payment Page Test Case				
Use Case : Fill in Emergency Contant Details, Select Payment Method, Make a Payment, View Booking Confirmation page				
Role : Current Users				
No	**Test Cases**	**Expected Results**	**Actual Results**	**Pass/Fail**
1.	Successful retrieve booking details from database.	Display booking details and payment details.	Display booking details and payment details.	Pass
2.	Unsuccessful retrieve booking details from database.	Error message as no booking data found.	Error message as no booking data found.	Pass
3.	Click “Pay” without fill in the emergency contact details.	Error message as missing field.	Error message. Please fill the all field.	Pass
4.	Click “Pay” without fill in the emergency contact name.	Error message as missing field.	Error message. Please fill the name field.	Pass
5.	Click “Pay” without fill in the emergency contact IC number.	Error message as missing field.	Error message. Please fill the IC number field.	Pass
6.	Click “Pay” without fill in the emergency contact Relationship.	Error message as missing field.	Error message. Please fill the Relationship field.	Pass
7.	Click “Pay” without fill in the emergency contact Home Address.	Error message as missing field.	Error message. Please fill the Home Address field.	Pass
8.	Click “Pay” without fill in the emergency contact Phone Number.	Error message as missing field.	Error message. Please fill the Phone number field.	Pass
9.	Click “Pay” without select the payment method.	Error message as missing payment method.	Error message. Please select a payment method.	Pass
10.	Fill in name input field with number or symbol.	Error message as invalid name field, only letter and space allowed.	Error message. Invalid name field, only letter and space allow.	Pass
11.	Fill in IC number input field with Character or symbol or invalid format.	Error message invalid IC number field, only number, “-” and xxxxxx-xx-xxxx allowed.	Error message. Invalid IC number field only only number ,“-” and xxxxxx-xx-xxxx allowed.	Pass
12.	Fill in Phone number input field with	Error message invalid Phone number field, only	Error message. Invalid Phone number field only	Pass

	Character or symbol or invalid format.	number, "-" and xxx-xxxxxxx allowed.	only number ,"-" and xxx-xxxxxxx allowed.	
13.	Click "Continue" without fill in credit/debit card input field.	Error message as missing field.	Error message. Please fill the all field.	Pass
14.	Click "Continue" without fill in credit/debit card name input field.	Error message as missing field.	Error message. Please fill the name field.	Pass
15.	Click "Continue" without fill in credit/debit card number input field.	Error message as missing field.	Error message. Please fill the card number field.	Pass
16.	Click "Continue" without fill in credit/debit card expires date input field.	Error message as missing field.	Error message. Please fill the expires date field.	Pass
17.	Click "Continue" without fill in credit/debit card cvv number input field.	Error message as missing field.	Error message. Please fill the cvv number field.	Pass
18.	Click "Pay" with fill in input field.	Forward to Booking Confirmation page.	Forward to Booking Confirmation page.	Pass
19.	Successful retrieve Payment details from payment page.	Display Booking Confirmation page.	Display Booking Confirmation page.	Pass
20.	Unsuccessful retrieve Payment details from payment page.	Error message as no payment data found.	Error message as no payment data found.	Pass

Table 6.9 Profile page Test Case

Profile Page Test Case				
Use Case : View Profile Page, Register Profile Page				
Role : Current Users and New Users				
No	**Test Cases**	**Expected Results**	**Actual Results**	**Pass/Fail**
1.	Successful retrieve user's details from database.	Display user's details.	Display user's details.	Pass
2.	Unsuccessful retrieve user's details from database.	Error message as no user's data found.	Error message as no user's data found.	Pass
3.	Click "Register" button on profile page.	Forward to Register Profile page.	Forward to Register Profile page.	Pass
4.	Click "Save" button without fill in input field.	Error message as missing field.	Error message. Please fill the all field.	Pass
5.	Click "Save" button without fill in name input field.	Error message as missing field.	Error message. Please fill the name field.	Pass
6.	Click "Save" button without fill in Phone number input field.	Error message as missing field.	Error message. Please fill the Phone number field.	Pass
7.	Click "Save" button without fill in IC number input field.	Error message as missing field.	Error message. Please fill the IC number field.	Pass
8.	Click "Save" button without fill in Email input field.	Error message as missing field.	Error message. Please fill the Email field.	Pass
9.	Click "Save" button without select License date input field.	Error message as missing field.	Error message. Please fill the License date field.	Pass
10.	Click "Save" button without select IC Front image input field.	Error message as missing field.	Error message. Please fill the IC Front image field.	Pass
11.	Click "Save" button without select IC Back image input field.	Error message as missing field.	Error message. Please fill the IC Back image field.	Pass
12.	Click "Save" button without select License Front image input field.	Error message as missing field.	Error message. Please fill the License Front image field.	Pass
13.	Click "Save" button without select	Error message as missing field.	Error message. Please fill the	Pass

	License Front image input field.		License Back image field.	
14.	Click “Save” without select Face image input field.	Error message as missing field.	Error message. Please fill the Face image field.	Pass
15.	Click “Save” with select become car rental owner.	Display extra Car Rental Owner Bank Details input field.	Display extra Car Rental Owner Bank Details input field.	Pass
16.	Click “Save” without select become car rental owner.	Save the user data as normal user.	Save the user data as normal user.	Pass
17.	Click “Save” without fill in Bank details input field.	Error message as missing field.	Error message. Please fill the Bank details field.	Pass
18.	Click “Save” without select Bank type field.	Error message as missing field.	Error message. Please select Bank type field.	Pass
19.	Click “Save” without fill in Bank holder name input field.	Error message as missing field.	Error message. Please fill in Bank holder name input field.	Pass
20.	Click “Save” without fill in Bank account number input field.	Error message as missing field.	Error message. Please fill in Bank account number input field.	Pass
21.	Fill in name input field with invalid format.	Error message as invalid name field, only letter and space allowed.	Error message. Invalid name field, only letter and space allow.	Pass
22.	Fill in Phone number input field with invalid format.	Error message invalid Phone number field, only number, “-” and xxx-xxxxxxx allowed.	Error message. Invalid Phone number field only only number ,“-” and xxx-xxxxxxx allowed.	Pass
23.	Fill in IC number input field with invalid format.	Error message invalid IC number field, only number, “-” and xxxxxx-xx-xxxx allowed.	Error message. Invalid IC number field only only number ,“-” and xxxxxx-xx-xxxx allowed.	Pass
24.	Fill in email input field with invalid registered email.	Error message as invalid registered email.	Error message as invalid registered email.	Pass
25.	Fill in License date input field with invalid date.	Error message as invalid date.	Error message. Your license is expired, please renew before registering.	Pass

26.	Fill in the Bank Holder name input field with invalid input.	Error message as invalid name field, only letter and space allowed.	Error message. Invalid name field, only letter and space allow.	Pass
27.	Fill in the Bank account number input field with invalid input.	Error message as invalid Bank account number field.	Error message as invalid Bank account number field.	Pass

Table 6.10 History Page Test Case

History Page Test Case				
Use Case : View Recent History, View Cancelled History, View Completed History, PickUp Car, Return Car, Upload Payment Slip				
Role : Current Users				
No	**Test Cases**	**Expected Results**	**Actual Results**	**Pass/Fail**
1.	Successful retrieve recent booking history from database.	Display the recent booking history list.	Display the recent booking history list.	Pass
2.	Successful retrieve cancelled booking history from database.	Display the cancelled booking history list.	Display the cancelled booking history list.	Pass
3.	Successful retrieve completed booking history from database.	Display the completed booking history list.	Display the completed booking history list.	Pass
4.	Unsuccessful retrieve recent booking history from database.	Display no data found.	Display no data found.	Pass
5.	Unsuccessful retrieve cancelled booking history from database.	Display no data found.	Display no data found.	Pass
6.	Unsuccessful retrieve completed booking history from database.	Display no data found.	Display no data found.	Pass
7.	Successful retrieve booking history car details from database.	Display the recent booking history car details.	Display the recent booking history car details.	Pass
8.	Unsuccessful retrieve booking history car details from database.	Display no data found.	Display no data found.	Pass
9.	Click “submit details” without fill in all input field.	Error message as missing field.	Error message as missing field.	Pass

10.	Click “submit details” without fill in fuel photos input field.	Error message as missing field.	Error message. Please upload the fuel photos.	Pass
11.	Click “submit details” without fill in return car photos input field.	Error message as missing field.	Error message. Please upload the return car photos.	Pass
12.	Click “submit details” without fill in return car mileage input field.	Error message as missing field.	Error message. Please enter the return car mileage.	Pass

Table 6.11 Car List Page Test Case

Car List Page Test Case				
Use Case : View Car List, View Car Details				
Role : Current User				
No	**Test Cases**	**Expected Results**	**Actual Results**	**Pass/Fail**
1.	Successful retrieve data from car database.	Display the car list.	Display the car list.	Pass
2.	Unsuccessful retrieve data from car database.	Error message as no car list found.	Error message as no car list found.	Pass
3.	Successful retrieve data from car database.	Display the car details.	Display the car details.	Pass
4.	Unsuccessful retrieve data from car database.	Error message as no car details found.	Error message as no car details found.	Pass

Table 6.12 Notification Page Test Case

Notifications Page Test Case				
Use Case : View Notification List, View Notification Details				
Role : Current User				
No	**Test Cases**	**Expected Results**	**Actual Results**	**Pass/Fail**
1.	Successful retrieve data from notifications database.	Display the notifications list.	Display the notifications list.	Pass
2.	Unsuccessful retrieve data from notifications database.	Error message as no notifications list found.	Error message as no notifications list found.	Pass
3.	Successful retrieve notification details from notifications database.	Display the notifications details.	Display the notifications details.	Pass
4.	Unsuccessful retrieve notification details from notifications database.	Error message as no notifications details found.	Error message as no notifications details found.	Pass

Table 6.13 Setting Page Test Case

Setting Page Test Case				
Use Case : View Car List, View Car Details, Edit Car, Add Car, Delete Car				
Role : Current User				
No	**Test Cases**	**Expected Results**	**Actual Results**	**Pass/Fail**
1.	Click “Change Password” button.	Display pop out reset message.	Display pop out reset message.	Pass
2.	Click “Delete Account” button.	Display pop out delete account message.	Display pop out delete account message.	Pass
3.	Click “Support” button.	Forward to Support Page.	Forward to Support Page.	Pass
4.	Click “Logout” button.	Display pop out Logout message.	Display pop out Logout message.	Pass
5.	Click “Edit Car List” button.	Forward to edit car page.	Forward to edit car page.	Pass
6.	Successful retrieve Owner’s car list from car database.	Display the owner car list.	Display the owner car list.	Pass
7.	Unsuccessful retrieve Owner’s car	Error message as no car list found.	Error message as no car list found.	Pass

	list from car database.			
8.	Click “Edit” on edit car list page.	Forward to edit car list.	Forward to edit car list.	Pass
9.	Click “Delete” on edit car list page.	Display pop out delete car message.	Display pop out delete car message.	Pass
10.	Click “Add car” without fill in all input field.	Error message as missing input field.	Error message as missing input field.	Pass
11.	Click “Add car” without fill in Car Brand input field.	Error message as missing Car Brand input field.	Error message as missing Car Brand input field.	Pass
12.	Click “Add car” without fill in Car Model input field.	Error message as missing Car Model input field.	Error message as missing Car Model input field.	Pass
13.	Click “Add car” without fill in Car Year input field.	Error message as missing Car Year input field.	Error message as missing Car Year input field.	Pass
14.	Click “Add car” without select Car Type input field.	Error message as missing Car Type input field.	Error message as missing Car Type input field.	Pass
15.	Click “Add car” without select Fuel Type input field.	Error message as missing Fuel Type input field.	Error message as missing Fuel Type input field.	Pass
16.	Click “Add car” without select Transmission input field.	Error message as missing Transmission input field.	Error message as missing Transmission input field.	Pass
17.	Click “Add car” without fill in Colour input field.	Error message as missing Colour input field.	Error message as missing Colour input field.	Pass
18.	Click “Add car” without select Seating Capacity input field.	Error message as missing select Seating Capacity input field.	Error message as missing select Seating Capacity input field.	Pass
19.	Click “Add car” without fill in Luggage Capacity input field.	Error message as missing Luggage Capacity input field.	Error message as missing Luggage Capacity input field.	Pass
20.	Click “Add car” without fill in Extra Cost per KM input field.	Error message as missing Extra Cost per KM input field.	Error message as missing Extra Cost per KM input field.	Pass
21.	Click “Add car” without fill in Rental Price input field.	Error message as missing Rental Price input field.	Error message as missing Rental Price input field.	Pass
22.	Click “Add car” without fill in Deposit input field.	Error message as missing Deposit input field.	Error message as missing Deposit input field.	Pass

23.	Click “Add car” without fill in Location input field.	Error message as missing Location input field.	Error message as missing Location input field.	Pass
24.	Click “Add car” without fill in Owner Contact input field.	Error message as missing Owner Contact input field.	Error message as missing Owner Contact input field.	Pass
25.	Fill in Car Brand with invalid format.	Error message as invalid Car Brand input field.	Error message. Brand can only contain letters and spaces.	Pass
26.	Fill in Car Model with invalid format.	Error message as invalid Car Model input field.	Error message. Model can only contain letters, numbers and spaces.	Pass
27.	Fill in Car Year with invalid format.	Error message as invalid Car Year input field.	Error message. Year must be 4 digits (xxxx) and only contain numbers.	Pass
28.	Fill in Colour with invalid format.	Error message as invalid Colour input field.	Error message. Color can only contain letters and spaces.	Pass
29.	Fill in Luggage Capacity with invalid format.	Error message as invalid Luggage Capacity input field.	Error message. Luggage Capacity can only contain letters, numbers and spaces.	Pass
30.	Fill in Mileage with invalid format.	Error message as invalid Mileage input field.	Error message. Mileage must be a positive number and only contain numbers.	Pass
31.	Fill in Extra Cost per KM with invalid format.	Error message as invalid Extra Cost per KM input field.	Error message. Extra Cost per KM must be a positive number and only contain numbers.	Pass
32.	Fill in Rental Price with invalid format.	Error message as invalid Rental Price input field.	Error message. Rental Price must be a positive number and only contain numbers.	Pass
33.	Fill in Deposit with invalid format.	Error message as invalid Deposit input field.	Error message. Deposit must be a positive number and only contain numbers.	Pass

34.	Fill in Location with invalid format.	Error message as invalid Location input field.	Error message. Location can only contain letters and spaces.	Pass
35.	Fill in Owner Contact with invalid format.	Error message as invalid Owner Contact input field.	Error message. Owner Contact only contain numbers and "-" with the format xxx-xxxxxxx.	Pass
36.	Click "Add car" with valid input field.	Successful Add car.	Successful Add car.	Pass

6.2 Project Challenges

One of the challenges in this project is ensuring smooth navigation and usability, particularly with the side menu that offers multiple features such as Booking, Profile, History, Car List, Notifications, and Settings. The menu needs to be simple and clear so users can easily move between different pages without getting lost. Another challenge is ensuring the app works well on different device sizes. Additionally, handling bookings and notifications can be tricky, requiring careful management of Firebase connections to avoid bugs and ensure real-time data updates.

Managing user profiles and sensitives information, such as booking and payment details need to be done securely. Furthermore, handling large amounts of data or real-time updates could potentially slow down the app's performance, so we need to ensure that the app runs smoothly even when processing large datasets. In addition, handling the code errors is a significant challenge due to the connection problems or data retrieval issues. To resolve these issues, we need to check the code line by line and add logging to capture error-level message. Lastly, testing the release APK file on different Android is crucial to ensure everything works as expected, from navigating the menu to receiving notifications.

6.3 Objective Evaluation

User Acceptance Testing is a phase that for the test-driven development strategy, where will give to the targeted users to test the system. This testing phase plays an important role in the development of the project which determines the success of the entire project meets the user's needs and expectations. Therefore, 3 testers has been targeted to perform the acceptance test and evaluate the car rental booking application accordingly. Below are the results from the three (3) testers after the user acceptance test.

Car Rental Application - User Acceptance Testing
Car Rental Using Mobile Application Development

Name : Eng Zi Yee	Position : Student
Date : 30-4-2025	Time: 12:35pm

Rating Scale: 1 (lowest/worst) to 5 (highest/excellent).
(Higher ratings indicate greater satisfaction.)

No.	Acceptance Criteria	Rating					Remarks
		1	2	3	4	5	
System Content							
1.	Readability				✓		
2.	Clarity of Information					✓	
3.	Layout Consistency				✓		
System Functionality							
4.	Overall Functionality					✓	
5.	Ease of Accessibility					✓	
6.	Booking Process Efficiency					✓	
7.	Multiple Payment Integration					✓	
8.	Real-time Updates					✓	Results are good and accurate
System Performance							
9.	Response Time					✓	Fast response
10.	Processing Speed				✓		
11.	Application Stability				✓		
12.	Error and Bug Handling			✓			
User Support							
13.	Helpfulness of FAQs					✓	
14.	Efficiency of Customer Support				✓		
Others							
15.	Achievement of Objectives					✓	
16.	Security & Privacy				✓		
17.	Overall User-friendliness					✓	

Comments / Suggestions:
An efficient tool for car renting.Nice design. Can add more features to look interesting.

Signature: 

Actions taken by developer:

The developer will use this feedback for future enhancements.

Figure 6.1 User Acceptance Testing Form (1)

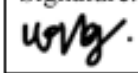
Car Rental Application - User Acceptance Testing Car Rental Using Mobile Application Development							
Name : Wong Hou Yan				Position : Student			
Date : 1-5-2025				Time: 4:08pm			
Rating Scale: 1 (lowest/worst) to 5 (highest/excellent).							
(Higher ratings indicate greater satisfaction.)							
No.	**Acceptance Criteria**	**Rating**					**Remarks**
		1	**2**	**3**	**4**	**5**	
System Content							
1.	Readability					✓	Good
2.	Clarity of Information				✓		Clear
3.	Layout Consistency				✓		Tidy
System Functionality							
4.	Overall Functionality					✓	Well function
5.	Ease of Accessibility					✓	
6.	Booking Process Efficiency					✓	
7.	Multiple Payment Integration				✓		
8.	Real-time Updates					✓	
System Performance							
9.	Response Time					✓	
10.	Processing Speed					✓	
11.	Application Stability				✓		
12.	Error and Bug Handling				✓		
User Support							
13.	Helpfulness of FAQs				✓		Helpful. But can add more FAQ.
14.	Efficiency of Customer Support					✓	
Others							
15.	Achievement of Objectives					✓	Align with objectives
16.	Security & Privacy				✓		
17.	Overall User-friendliness					✓	
Comments / Suggestions: Can add real time car tracking location. Overall is good.							
Signature: 							
Actions taken by developer: The developer will use this feedback for future enhancements.							

Figure 6.2 User Acceptance Testing Form (2)

Car Rental Application - User Acceptance Testing Car Rental Using Mobile Application Development							
Name : Choo Zhen Heng				Position : Student			
Date : 30-4-2025				Time: 10:23am			
Rating Scale: 1 (lowest/worst) to 5 (highest/excellent).							
(Higher ratings indicate greater satisfaction.)							
No.	**Acceptance Criteria**	**Rating**					**Remarks**
		1	**2**	**3**	**4**	**5**	
System Content							
1.	Readability				✓		
2.	Clarity of Information				✓		Quite clear
3.	Layout Consistency				✓		Nice layout.
System Functionality							
4.	Overall Functionality					✓	
5.	Ease of Accessibility				✓		
6.	Booking Process Efficiency					✓	
7.	Multiple Payment Integration					✓	Add more payment method.
8.	Real-time Updates					✓	Accurate data update
System Performance							
9.	Response Time				✓		
10.	Processing Speed				✓		
11.	Application Stability				✓		
12.	Error and Bug Handling			✓			So far no error
User Support							
13.	Helpfulness of FAQs				✓		
14.	Efficiency of Customer Support				✓		
Others							
15.	Achievement of Objectives					✓	
16.	Security & Privacy				✓		Can add 2 factor authentication when login
17.	Overall User-friendliness				✓		
Comments / Suggestions: Can make application more colourful.							
Signature: 							
Actions taken by developer: The developer will use this feedback for future enhancements.							

Figure 6.3 User Acceptance Testing Form (3)

6.3.1 To develop a streamlined and user-friendly car rental booking mobile application

The proposed system has achieved the following objectives, which demonstrate a clear, modern interface with easy navigation to each feature in the application. This streamlined and user-friendly design allows users to easily browse the available car and book a car directly from their mobile devices. The app's features include basic main functionality such as "Booking", " Profile", " Car List", " Notification" , " History" and "Setting" which can be accessible through simple menus from the home page. With the first approach of modern and mobile-first design helps users to make a booking easier by allowing the users quickly select a car and pay without extra step.

6.3.2 To develop a timely assistance customer support features for immediate user assistance

The presence of a "Support" feature in the setting which provide the comprehensive FAQ section, 24x7 customer care contact details, and also provide a clear layout for reaching support such as contact number or email that can directly use in the app, it shows that the app which can provide timely assistance. These features in this application meet the objective of enhancing customer support, which allows users to resolve issues quickly when users are facing any problem during using the application. With this support feature it can enhance the user experience and build trust.

6.3.3 To develop a multiple choice of payment system in car rental booking system

The integration of multiple of payment option such as credit/debit cards, online banking, e-wallets and bank in which meet the wide range of user needs who like to use the modern or traditional payment method and also achieves the objectives of providing multiple choice of payment system. A wide range of payment options allow user to choose their preferred payment with can promoting financial and meeting the needs of both modern and traditional users. The payment process in the app is straightforward and it provide a clear payment options for users to select based on their preferred method, which can enhance user's payment experience.

Chapter 7
Conclusion and Future Work

7.1 Conclusion

In conclusion, the development of the Car Rental Booking mobile applications has made a significant improvement in the car rental industry by addressing long-standing challenges and enhancing the overall customer experience when booking a car. This report provides the detailed process undertaken to create a modern, efficient, and user-friendly solution that simplifies and optimizes the user's car rental booking experience. By modernizing the traditional car rental system, the application provides a streamlined process that allows users to easily browse the available cars, make a booking, and enables car owners to manage rental vehicles directly from their mobile devices.

The integration of multiple payment options in the car rental booking application meets the needs of both digital-first users and those users who prefer traditional methods. This integration of multiple payment ensures inclusivity and flexibility, meeting the different preferences of modern consumers. Additionally, the enhanced customer support features, including FAQs, live chat, and easy access to help guides, provide immediate assistance to users when facing difficulty using the application, building trust and improving the overall user experience.

In summary, this report has demonstrated how to develop a car rental app that addresses current challenges and prepares for future needs. The app is designed to deliver faster, more efficient, and user-friendly service, improving operational efficiency and customer satisfaction. The successful development and implementation of this car rental app sets a new standard for the industry, making the car rental process simpler, more convenient, and future ready.

7.2 Recommendation

Although the proposed system has fulfilled the basic functions and design of a car rental booking application. For future recommendations, there are several key features that could enhance the car rental booking application which can improve the user's experience. Firstly, a real-time tracking feature could be implemented. By offering this feature, users would be able to track of their car's real-time location, providing greater security and transparency. In addition, with a real-time tracking system, the car owner could control the start and stop of the car, ensuring that the vehicle is returned on time and remains secure. For example, if a user exceeds the return period, the system will send an alert to the owner and offer the user an option to top up their rental time. If the user does not top up, the owner could manually disable the car, thereby improving operational efficiency and safety.

Another recommendation is to introduce a new payment method that allows users to top up money directly within the app. This feature would make the payment process faster, more convenient and would streamline the booking experience. At the same time, improving payment security is also important. Implementing stronger encryption, multi-factor authentication and also following the security standards will protect user's personal and financial data, enhancing trust and ensuring secure transactions.

Lastly, integrating an AI support system could significantly improve customer service. An AI chatbot could handle common user problems, guide users through the booking process or using the application, suggest vehicle options and assist with any issues user's encounter. This will reduce reliance on live customer support and provide faster response times and 24/7 availability. Additionally, with the AI chatbot could collect valuable user feedback, helping to continuously improve the application and making it more adaptive to customer preferences and needs.

References

[1] "Moovby - Car Sharing - Apps on Google Play," *play.google.com*. https://play.google.com/store/apps/details?id=com.moovby (accessed Apr. 13, 2024).

[2] "TREVO - Car Sharing Done Right - Apps on Google Play," *play.google.com*. https://play.google.com/store/apps/details?id=my.trevo.trevoapp (accessed Apr. 13, 2024).

[3] "gocar - Android Apps on Google Play," *play.google.com*. https://play.google.com/store/search?q=gocar&c=apps (accessed Apr. 13, 2024).

[4] "115442368-49930505-Car-Rental-System-Project-Report," *SlideShare*, Aug. 05, 2023. https://www.slideshare.net/ScottBou/1154423684993050 5carrentalsystemprojectreport (accessed Apr. 13, 2024).

[5] "Moovby - Car Sharing - Apps on Google Play," *play.google.com*. https://play.google.com/store/apps/details?id=com.moovby

[6] admin, "Explore top 25 Must-have features of Car Rental Management Software," *Adamo Software*, Dec. 14, 2024. https://adamosoft.com/blog/travel-software-development/must-have-features-of-car-rental-management-software/

[7] S. Shakthi, "Car Rental Management System [Development Of Car rental management system] LINGHURAJ P R ,ARAVIND K, GIRITHARAN R Guide: Mrs.T.Sathya AP/IT Department Of Information Technology Bachelor Of Technology," vol. 9, p. 200, 2024, Accessed: Jun. 09, 2024. [Online]. Available: https://www.ijnrd.org/papers/IJNRD2403623.pdf

[8] Aishwarya Mandave, Pranoti Wakodkar, Prachi Todkari, Samata G Kamble, and Prof. Anjali P. Kadam, "Vehicle Rental System," *International Journal of Advanced Research in Science, Communication and Technology*, pp. 339–343, May 2022, doi: https://doi.org/10.48175/ijarsct-3933.

POSTER


CAR RENTAL USING MOBILE APPLICATION DEVELOPMENT
Introduction
Simplifying the Car Rental Booking Process
Addressing Limited Payment Options
Improving Customer Support Channels
Objective
Multiple Choice of Payment Method
User Friendly
Access round-the-clock support
Car with best moments
Conclusion
Done booking with just a tap
User Friendly design
Simple and attractive design
By : Mah Guo Xiang