



Faculty of Computer Science and Information Technology

SMART CAR RENTAL APPS

Brandon Keane Anak Peron

Bachelor of Computer Science with Honours (Information Systems)

2025

UNIVERSITI MALAYSIA SARAWAK

THESIS STATUS ENDORSEMENT FORM

TITLE: SMART CAR RENTAL APPS

ACADEMIC SESSION: 2025

BRANDON KEANE ANAK PERON
(CAPITAL LETTERS)

hereby agree that this Thesis* shall be kept at the Centre for Academic Information Services, Universiti Malaysia Sarawak, subject to the following terms and conditions:

1. The Thesis is solely owned by Universiti Malaysia Sarawak
2. The Centre for Academic Information Services is given full rights to produce copies for educational purposes only
3. The Centre for Academic Information Services is given full rights to do digitization in order to develop local content database
4. The Centre for Academic Information Services is given full rights to produce copies of this Thesis as part of its exchange item program between Higher Learning Institutions [or for the purpose of interlibrary loan between HLI]
5. ** Please tick (√)

☐

CONFIDENTIAL

(Contains classified information bounded by the OFFICIAL SECRETS ACT 1972)

☐

RESTRICTED

(Contains restricted information as dictated by the body or organization where the research was conducted)

☒

UNRESTRICTED

(AUTHOR'S SIGNATURE)

Validated by

(SUPERVISOR'S SIGNATURE)

Dr. Fatimah bt. Ramli

Faculty of Education, Universiti Malaysia Sarawak

Permanent Address:

RUMAH ENTAU PUNGGU JINGGAU,
PANTU, 95000 SRI AMAN, SARAWAK

Date: 23 JUNE 2025

Date: 24 JULAI 2025

Note * Thesis refers to PhD, Master, and Bachelor Degree

** For Confidential or Restricted materials, please attach relevant documents from relevant organizations / authorities

SMART CAR RENTAL APPS

BRANDON KEANE ANAK PERON

This project is submitted in partial fulfillment of the requirements for the degree of
Bachelor of Computer Science with Honours (Information Systems)

Faculty of Computer Science and Information Technology

UNIVERSITI MALAYSIA SARAWAK

2025

DECLARATION

I hereby declare that this project is my original work. I have not copied from any other student's work or any other sources except where due to reference or acknowledgment is not made explicitly in the text, nor has any part had been written for me by another person.

.....
(BRANDON KEANE ANAK PERON)

(23/6/2025)

ACKNOWLEDGEMENTS

First and foremost, I would like to express my sincere gratitude to my supervisor, Dr. Fatihah Binti Ramli, and my examiner, Madam Noor Hazlini Bt Borhan, for their guidance, support, and encouragement throughout the development of this project. Her insights and constructive feedback were extremely valuable in helping me improve and complete my work. I also want to thank the Final Year Project coordinator, Professor Dr. Wang Yin Chai, for providing clear direction and assistance during this process.

I am also thankful to the examiners for their useful comments and suggestions, and to all my friends and course mates who supported me along the way. Lastly, I would like to thank my family for their constant love, motivation, and understanding, especially during the challenging moments of this project.

ABSTRACT

Smart Car Rental App is developed as a web-based platform for car rental activities using Rapid Application Development Model (RAD) methodology involving two user roles: Owner and Renter. By offering features like car listing management, a booking system with status tracking, and chat communication between the two individuals, the system aims to simplify the rental process. An AI-powered chatbot is also available to help users with basic support and pricing concerns. This system was created to solve problems with manual booking methods that were previously used, like tracking reservations, vehicle status, and renter information. Renters can easily search, filter, and reserve cars on this platform, and owners can manage their car listings. System testing yielded successful results, with all core features such as login, booking, status updates, chatbot and payment functioning correctly under various test cases. Additionally, the website makes it possible to replace paper-based records with data stored in a centralized database. In conclusion, by providing a more intelligent, effective, and user-friendly web solution, the suggested system assists vehicle rental companies in streamlining operations and enhancing the client experience.

ABSTRAK

Smart Car Rental App dibangunkan sebagai platform berasaskan web untuk aktiviti sewaan kereta menggunakan kaedah *Rapid Application Development Model (RAD)* yang melibatkan dua peranan pengguna: *Pemilik dan Penyewa*. Dengan menawarkan ciri seperti pengurusan senarai kereta, sistem tempahan dengan penjejakan status, dan komunikasi sembang antara kedua-dua individu, sistem ini bertujuan untuk memudahkan proses sewaan. Chatbot berkuasa AI juga disediakan untuk membantu pengguna dengan sokongan asas dan pertanyaan harga. Sistem ini dicipta untuk menyelesaikan masalah dengan kaedah tempahan manual yang digunakan sebelum ini, seperti menjejak tempahan, status kenderaan, dan maklumat penyewa. Penyewa boleh mencari, menapis, dan menempah kereta dengan mudah di platform ini, dan pemilik boleh mengurus senarai kereta mereka. Ujian sistem menunjukkan hasil yang berjaya, dengan semua ciri teras seperti log masuk, tempahan, kemas kini status, chatbot dan pembayaran berfungsi dengan betul di bawah pelbagai kes ujian. Selain itu, laman web ini memungkinkan penggantian rekod berasaskan kertas dengan data yang disimpan dalam pangkalan data berpusat. Kesimpulannya, dengan menyediakan penyelesaian web yang lebih pintar, berkesan, dan mesra pengguna, sistem yang dicadangkan membantu syarikat penyewaan kenderaan dalam memperkemas operasi dan meningkatkan pengalaman pelanggan.

Table of Contents

DECLARATION	iv
ACKNOWLEDGEMENTS	v
ABSTRACT	vi
ABSTRAK	vii
LIST OF FIGURES	iii
LIST OF TABLES	v
LIST OF ABBREVIATIONS	vi
Chapter 1: Introduction	1
1.1 Introduction.....	1
1.2 Problem Statement	1
1.3 Scope.....	2
1.4 Objective	3
1.5 Brief Methodology.....	3
1.6 Significant of Project	4
1.7 Project Schedule.....	4
1.8 Expected Outcome	5
Chapter 2: Literature Review.....	6
2.1 Introduction.....	6
2.2 Recommendation Systems in the Car Rental Business	7
2.3 Review of Existing Applications	8
2.4 Comparison Between Proposed and Existing Application	25
2.5 Technologies and Tools Used.....	27
2.6 Conclusion	30
Chapter 3: Methodology	31
3.1 Introduction.....	31
3.2 Methodology	31
3.3 Summary	48
Chapter 4: Implementation.....	49
4.1 Introduction.....	49
4.2 System Architecture	49
4.3 System Interfaces	50

4.4 Backend Implementation	61
4.5 Database Schema	62
4.6 Summary	62
Chapter 5: Testing and Evaluation	63
5.1 Introduction.....	63
5.2 Testing Strategy	63
5.3 Test Case and Expected Results.....	64
5.4 Non-Functional and User Acceptance Testing	67
5.5 Evaluation	70
5.6 Conclusion	72
Chapter 6: Conclusion and Future Works	73
6.1 Project Summary.....	73
6.2 Objective Achievement.....	74
6.3 Challenges and Limitations.....	75
6.4 Future Enhancements.....	75
6.5 Conclusion	76
Appendix A.....	77
Appendix B	78
References.....	79

LIST OF FIGURES

Figure 1.1 Example of searching for available cars.....	2
Figure 2.1 IkramTransports Dashboard	8
Figure 2. 2 IkramTransports Car List	9
Figure 2. 3 IkramTransports Book Information.....	10
Figure 2. 4 IkramTransports Car Condition.....	10
Figure 2. 5 SOCAR Dashboard	12
Figure 2. 6 SOCAR Car List.....	13
Figure 2. 7 SOCAR Car Information.....	14
Figure 2. 8 SOCAR Instructions	15
Figure 2. 9 Klezcar Dashboard	17
Figure 2. 10 Klezcar Car	17
Figure 2. 11 Klezcar Book Information.....	18
Figure 2. 12 Klezcar Add-on	19
Figure 2. 13 Kayak Dashboard	21
Figure 2. 14 Kayak Car Information.....	21
Figure 2. 15 Kayak Car List.....	22
Figure 2. 16 Kayak Book Choice.....	23
Figure 3.1 Rapid Application Development Model.....	32
Figure 3.2 Flowchart of Smart Car Rental App.....	37
Figure 3.3 Use Case Diagram	38
Figure 3.4 Entity-Relationship Diagram.....	39
Figure 3.5 Login Page.....	42
Figure 3.6 Register Page.....	43
Figure 3.7 Homepage.....	43
Figure 3.8 Available Cars	44
Figure 3.9 Booking Page.....	45
Figure 3.10 Chat Page.....	45
Figure 3.11 Logout Page.....	46
Figure 4.1 Smart Car Rental App’s login page	50
Figure 4.2 Smart Car Rental App’s registration form	51
Figure 4.3 Smart Car Rental App’s owner dashboard.....	52

Figure 4.4 Smart Car Rental App’s renter dashboard	53
Figure 4.5 Smart Car Rental App’s add/edit car	53
Figure 4.6 Smart Car Rental App’s list of cars	54
Figure 4.7 Smart Car Rental App’s search and filter car listing	55
Figure 4.8 Smart Car Rental App’s booking confirmation	55
Figure 4.9 Smart Car Rental App’s chat interface	56
Figure 4.10 Smart Car Rental App’s AI chatbot interface	57
Figure 4.11 Smart Car Rental App’s renter booking history page	67
Figure 4.12 Smart Car Rental App’s owner’s booking history	68
Figure 4.13 Smart Car Rental App’s forgot password	67
Figure 4.14 Smart Car Rental App’s payment	67
Figure 4.15 Smart Car Rental App’s rating.....	67
Figure 5.1 Car Renter UAT rating.....	69
Figure 5.2 Car Owner UAT rating	70
Figure 5.3 UAT Renters vs Owners	71

LIST OF TABLES

Table 1.1 Project Schedule	5
Table 2.1 Comparison Table.....	25
Table 3.1 Interviewee.....	33
Table 3.2 Functional and Non-Functional Requirements	34
Table 3.3 Hardware Requirements	36
Table 3.4 Software Requirements.....	36
Table 3.5 Data Dictionary.....	40
Table 5.1 Login Module	64
Table 5.2 Registration Module	64
Table 5.3 Car Management (Owner)	65
Table 5.4 Booking Process (Renter).....	65
Table 5.5 Payment and Status Update	66
Table 5.6 Chat and History	67
Table 5.7 Car Renter UAT Testing and Rating	68
Table 5.8 Car Owner UAT Testing and Rating	69
Table 6.1 Objective Achievement	74

LIST OF ABBREVIATIONS

Abbreviation	Word/Phrase
HTML	HyperText Markup Language
MySQL	My Structured Query Language
PHP	Hypertext Pre-Processor
SQL	Structured Query Language
XAMPP	X (cross platform), Apache, MySQL, PHP, Perl
CSS	Cascading Style Sheets
AI	Artificial Intelligence
SUV	Sports Utility Vehicle
API	Application Programming Interface
SDLC	System Development Life Cycle
RAD	Rapid Application Development
ERD	Entity-Relationship Diagram
UML	Unified Modeling Language
UAT	User Acceptance Testing

Chapter 1: Introduction

1.1 Introduction

The rise of the digital economy has transformed consumer expectations, efficiency and personalized experiences. Traditionally, car rental services often rely on manual booking systems, and this meets the struggle to meet these demands, leaving gaps in usability and customer satisfaction. Many other platforms lack advanced features like AI-driven personalization, dynamic pricing based on demand and 24/7 automated support, which nowadays is increasingly expected. These limitations hinder both customer experience and operational efficiency for rental providers.

The Smart Car Rental Apps project aims to bridge this gap by developing a fully online platform that streamlines the car rental process for users and businesses alike. This platform will offer personalized car recommendations, dynamic pricing based on demand, and 24/7 automated support, enhancing convenience for users, and user-friendly experience, while improving operational efficiency for car rental businesses.

1.2 Problem Statement

The car rental industry faces significant challenges, including time-consuming booking processes, lack of personal recommendations, and insufficient automated customer support. Most platforms fail to incorporate demand-based pricing, limiting their ability to offer competitive rates while maximizing utilization. The absence of 24/7 automated customer support further impacts user satisfaction, especially during urgent booking scenarios. By integrating AI-driven personalization, dynamic pricing models, and chatbot-based support to streamline the booking process will improve user experience. By addressing these challenges, the platform will meet the needs of both

consumers and businesses, making the rental process faster, more user-friendly, and adaptive to market demand. Figure 1.1 shows an example of a manual way to find available cars.

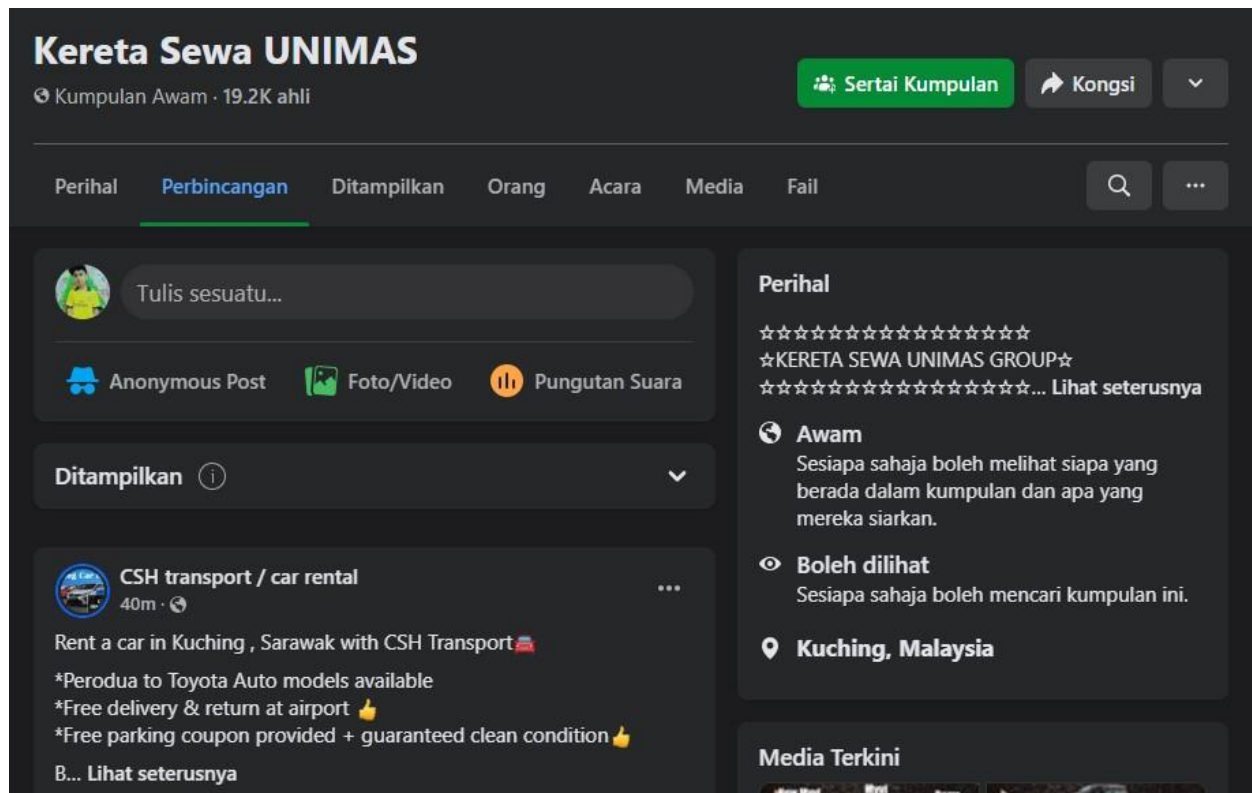


Figure 1.1 Example of searching for available cars

1.3 Scope

The project will focus on developing a fully functional car rental website that includes essential features for users and administrators. Users will be able to search for and browse available cars, make bookings, complete payments, receive personalized car suggestions and popular places visited at a distance. Administrators will have access to a dashboard for managing car inventory, tracking bookings, and generating reports. The scope includes:

- User and admin interfaces.

- Integration of AI-powered personalization features.
- A 24/7 chatbot for automated customer support.
- Dynamic pricing based on demand analysis.
- Payment processing for secure online transactions.

1.4 Objective

The primary aim of this project is to create an online platform that enhances car rental experience for users and streamlines the management process for car rental businesses. The specific objectives are:

- To develop a car rental website that enables users to browse, book, and pay for car rentals online.
- To implement AI features, such as personalized recommendations and chatbot that provides 24/7 automated support, to improve customer experience.
- To evaluate the effectiveness and user satisfaction of the developed platform.

1.5 Brief Methodology

The Smart Car Rental Apps project will follow an iterative development approach, with stages including requirements analysis, design, development, testing, and evaluation.

- Requirements Analysis: Identify and define key user and business requirements through research and stakeholder consultations.
- System Design: Develop both frontend and backend system designs, including wireframes, database schemas, and AI integration.

- Frontend Development: Use HTML, CSS, and JavaScript to create a responsive user interface.
- Backend Development: Implement server-side functionality with JavaScript, focusing on secure user sessions, book management, and AI-driven features.
- Database Management: Use MySQL for relational database management to store and retrieve booking and user data.
- AI Integration: Develop algorithms for car recommendation and dynamic pricing based on user preferences and demand.
- Testing and Evaluation: Conduct testing for usability, functionality, and performance, gathering user feedback to refine the platform.

1.6 Significant of Project

The significance of this project is to create a car rental platform with AI-driven features. By offering features such as personalized recommendations, dynamic pricing and automated customer support, this project aims to improve customer satisfaction and experience. A streamlined booking and management retention are offered to car rental businesses and reducing the reliance on manual processes and potentially lowering operational costs are also the significance of the project. The integration of a 24/7 automated chatbot ensures that users have access to assistance whenever needed, enhancing the overall user experience and adding a competitive advantage to car rental companies that adopt the platform.

1.7 Project Schedule

To complete a fully function website, the project is divided into the following key phases with estimated completion times:

Table 1.1 Project Schedule

Phase	Description	Estimated Duration	Timeline (Date)
Requirement Analysis	Gathering requirements and defining scope	14 days	14/10/24-27/10/24
Design Phase	System design, wireframes, and schematics	14 days	28/10/24-10/11/24
Frontend Development	Building user interface and responsive design	28 days	27/1/25-23/2/25
Backend Development	Server-side programming and database setup	21 days	24/2/25-16/3/25
AI Integration	Implementing recommendation and pricing AI	7 days	17/3/25-23/3/25
Testing and debugging	Functional and user testing, refinements	14 days	24/3/25-6/4/25
Evaluation and Deployment	Final evaluation and platform launch	14 days	7/4/25-20/4/25

1.8 Expected Outcome

Expected outcomes for the project are a fully functional and flexible car rental website that users can rely on for easy browsing, booking and payment. The system is a web-based application that run from browser with the inclusion of AI-driven personalized features is expected to enhance user satisfaction. The platform will also support efficient car inventory management, bookings oversight, pricing adjustment effectiveness, and reporting for administrators, helping car rental businesses run more effectively.

Chapter 2: Literature Review

2.1 Introduction

This chapter's purpose is to review the literature that is relevant to this project. The analysis begins by taking into consideration the body of previous studies, which shows that several attempts have been made to develop applications that are meant to improve vehicle rental systems. Additionally, car rental companies may promote their services on a single platform, and as Malaysia's car rental market grows, so will competition between them.

Three current apps have been reviewed for this project to offer recommendations for creating an appropriate solution for the current number of rental cars. The applications are IkramTransports, SOCAR, Klezcar, and Kayak. To gain useful information into the suggested smart car rental system, this chapter aims to describe their characteristics and functionalities, with a focus on areas like real-time updates, car availability, and integrated chat communication. Therefore, the purpose of this automatic rental application is to assist users, particularly students, in the process of renting a car.

2.2 Recommendation Systems in the Car Rental Business

By offering personalized vehicle recommendations based on customer interests and habits, recommendation systems have become essential to the vehicle rental sector. These systems may predict and suggest appropriate car selections by evaluating data like previous rentals, search history, and demographics of customers (Deshpande, Shiwankar, Patil, Pansare, & Shinde, 2023). This improves customer satisfaction and conversion rates. Machine learning systems, for example, can evaluate user data to provide customized car options that fit requirements and tastes. Customers' decision-making process is simplified by this individual approach, which also increases engagement and loyalty.

Car rental companies can also optimize their vehicle management and pricing tactics by implementing AI-driven recommendation systems. Businesses can more effectively use resources and flexibly adjust prices to optimize income by analyzing client demand and preferences. Furthermore, by automating the recommendation process and minimizing the need for person participation, these systems can improve operational efficiency. The adoption of price recommendation software in the car rental industry is not just a trend; it's a strategic move towards data-driven decision-making, resulting in increased profitability and customer satisfaction (Jane, 2024). For businesses hoping to keep a competitive advantage in the changing vehicle rental market, implementing such technologies is becoming more and more important.

Additionally, by ensuring that customers get options that best meet their needs, recommendation systems help to improve customer service and increase customer satisfaction. Recommendation systems are an essential part today's car rental services as the industry continues to embrace digital

transformation and the role of cutting-edge technologies like AI and machine learning in focusing on customer experiences and improving company operations is expected to grow (Yenra, 2024).

2.3 Review of Existing Applications

Three current applications have been selected for review and comparison in this topic. to determine the selected application's similarities, advantages, and disadvantages. We'll also talk about the features of the current and existing applications.

2.3.1 IkramTransports



Figure 2.1 IkramTransports Dashboard

Figure 2.1 displays the dashboard interface of IkramTransports, which serves as the main control panel for users. It provides a clean overview of car availability, user profile, and quick navigation to booking and vehicle management sections. This centralized layout helps users to manage their bookings and access car rental features efficiently.

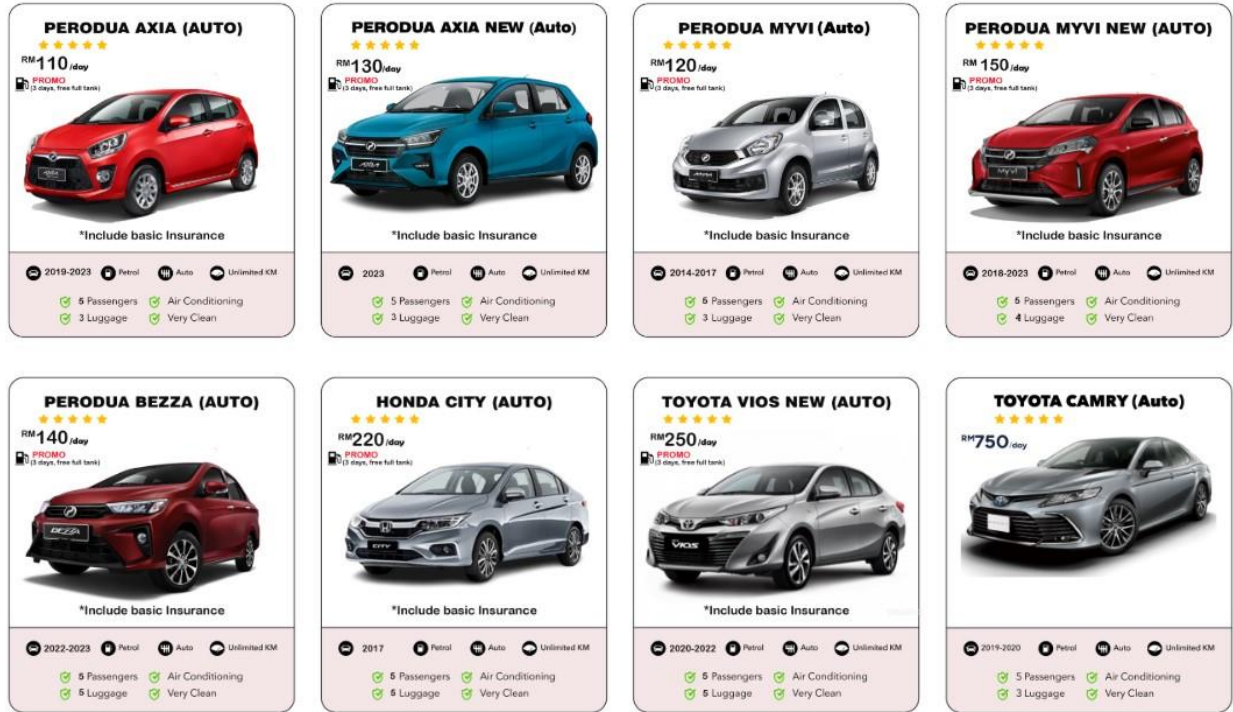


Figure 2.2 IkramTransports Car List

Figure 2.2 shows the list of available vehicles provided by IkramTransports. The page presents car details such as model, transmission type, and rental price. The layout allows renters to compare car options quickly, enhancing the decision-making process before booking.

PERODUA AXIA NEW (Auto)

★★★★★

RM130/day

PROMO

(3 days, free full tank)



*Include basic Insurance

2023

Petrol

Auto

Unlimited KM

5 Passengers

Air Conditioning

3 Luggage

Very Clean

New Axia 2023

New Axia 2023 (A)

From RM130.00

In order to make a booking for one-day, Please contact our admin

Pricing Info

Pickup Location

Pickup Location

Dropoff Location

Dropoff Location

Pickup Date & Time

Pickup Date

Pickup Time

Dropoff Date & Time

Drop-off Date

Drop-off Time

(Out Station Charges)

Betong

Mukah / Bintulu / Miri

Pusak / Saratok / Roban

Sibu

Sri Aman

RM150.00 - One Time

RM400.00 - One Time

RM200.00 - One Time

RM300.00 - One Time

RM100.00 - One Time

Figure 2.3 IkramTransports Book Information

Figure 2.3 illustrates the booking information form used by IkramTransports. This form captures rental dates, pick-up time, and contact information. It ensures that users provide the necessary data to proceed with a rental and serves as a confirmation point before submission.

Perodua Axia New 2023

Car Rental Kuching Sarawak – Perodua Axia 2023 is one of the latest collections from the Ikramtransports company. For now, our company only have 5 units. If you are interested in renting a Perodua Axia 2023, please make an early reservation before the reservation is full.

Car Rental Kuching – Perodua Axia 2023 is a new release in 2023. This car is very simple, quick and save in terms of fuel use. It can accommodate 4-5 people and has 2-3 modest luggage spaces.

Car Rental In Kuching – Perodua Axia 2023 are among the newest cars provided by our company.



Figure 2.4 IkramTransports Car Condition

Figure 2.4 highlights the car condition reporting section, where users can view or submit the current state of the vehicle. This helps maintain accountability and allows for inspection documentation before and after each rental.

Through the IkramTransports website, customers may conveniently rent vehicles in Kuching, Sarawak, Malaysia. It provides a variety of vehicles, including cars, SUVs, buses, vans, and costly options for transportation. The site's user-friendly style allows users to browse available cars, read brief descriptions, and request reservations. Users can get basic details such as prices and services despite the design's simplicity. The website also highlights its local knowledge and focusses to offering reliable, safe transportation for both solo and group travelers.

The IkramTransports website has notable limitations that impact its overall usability and functionality. A car rental platform may become less competitive if it lacks AI-driven features like dynamic pricing and tailored recommendations. To improve pricing tactics and draw in more clients, car rental companies are increasingly utilizing AI algorithms to constantly adjust prices based on a variety of criteria, such as market demand and changing seasons (Yenra, 2024). According to an article by Eliron Ekstein, published on Auto Rental News, AI and big data analytics are revolutionizing the car rental industry by enabling predictive maintenance, route optimization, and personalized customer experiences (Eliron, 2023). Additionally, the lack of 24/7 automated customer support, such as chatbots, can negatively impact user experience, as modern consumers expect immediate assistance at any time. Implementing AI-driven personalization and support systems can significantly enhance customer satisfaction and operational efficiency in the car rental industry.

To use the IkramTransports website, begin by navigating to the "Fleet" or "Vehicles" section to explore available options, each accompanied by descriptions and pricing details. Once you find a

suitable vehicle, click on it to view more specific information. Next, fill in the booking form with essential details such as your travel dates, destination, and personal information. After submitting the form, await confirmation via email or phone from IkramTransports. Upon receiving confirmation, follow the instructions provided to complete payment and finalize your booking.

2.3.2 SOCAR

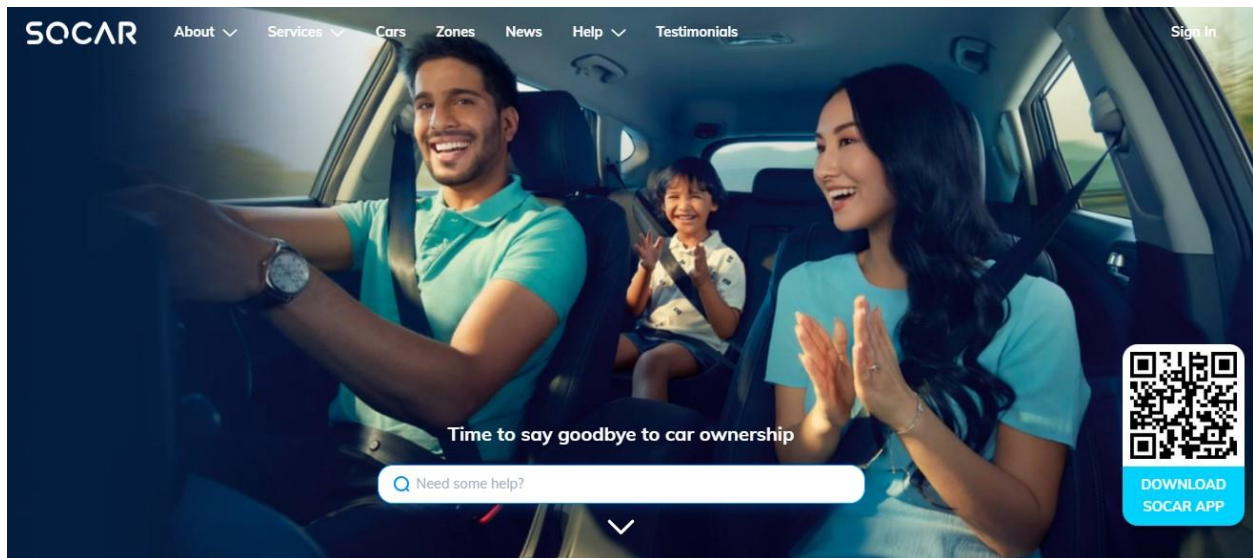


Figure 2.5 SOCAR Dashboard

Figure 2.5 presents the SOCAR dashboard, which gives users an overview of their bookings, available promotions, and frequently accessed features. The design is user-friendly, allowing both new and returning users to navigate effortlessly.

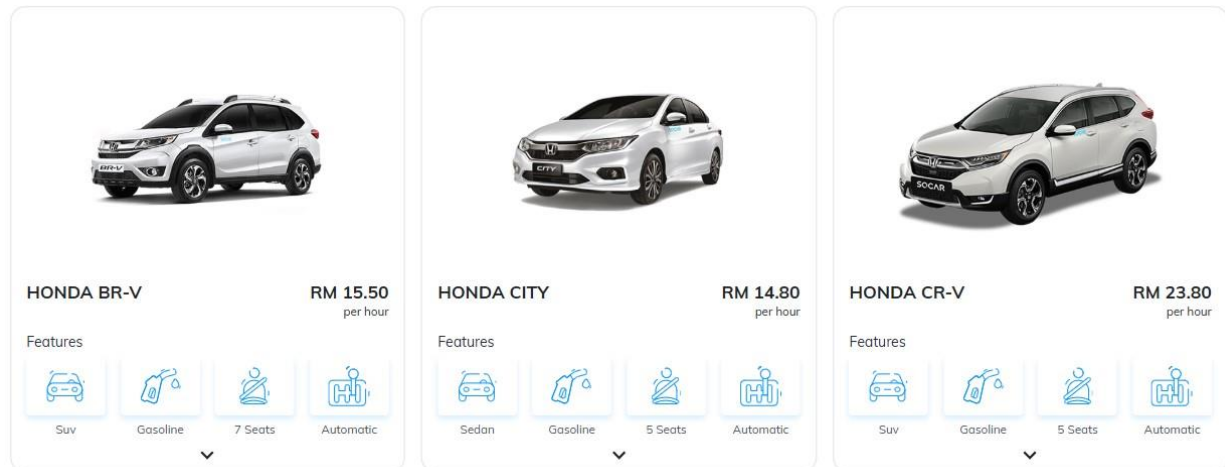


Figure 2.6 SOCAR Car List

Figure 2.6 shows the SOCAR vehicle listings. Each car is shown with clear details, including name, specifications, and pricing per hour. This list aids users in making informed rental choices based on their travel needs.



Figure 2.7 SOCAR Car Information

Figure 2.7 displays individual car information, including fuel level, seating capacity, and location. It ensures users understand what is included in their selection, promoting transparency and better planning.

Go Anywhere in 5 Easy Ways

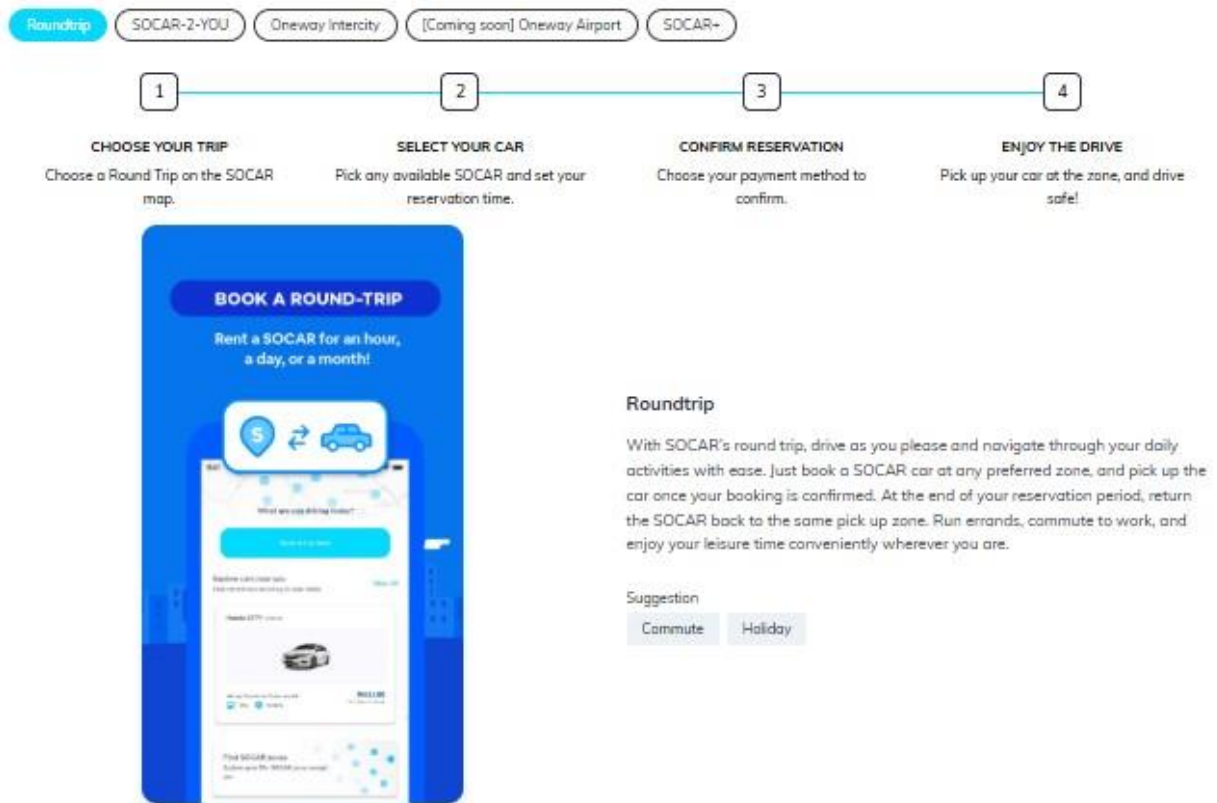


Figure 2.8 SOCAR Instructions

Figure 2.8 provides users with rental instructions, such as how to unlock the vehicle and where to return it. This guide is essential for first-time users and reduces confusion during the pick-up and return process.

Through its mobile app, SOCAR, a Malaysian car-sharing network, offers flexible car rentals as an environmentally friendly and efficient alternative for standard car rentals. To fulfill these various needs, the business provides more than 36 car models, including SUVs, sedans, and compact cars. An hourly booking system, fuel-inclusive rates, insurance coverage, and a real-time car finder are some of the main features. SOCAR users can streamline the rental process by simply unlocking and locking cars using the app. Child seats and one-way travel are further choices that

increase the service's adaptability and make it a popular alternative for both tourists and short-term users.

SOCAR has numerous limitations despite its unique characteristics. Its need for mobile devices for car booking and unlocking is an important limitation, which could be problematic in places with poor network service. Additionally, visitors or those who are not familiar with Malaysia's toll systems may find it inconvenient as users must have their own Touch 'n Go card in order to pay for tolls. The app's availability for users in more remote areas is further restricted by the fact that it mostly functions in urban areas. Furthermore, the usual rental rates might not be as affordable as those provided by many conventional rental firms, particularly for extended periods of time. Numerous user experiences and reviews have addressed these shortcomings.

To use SOCAR, download and install the app from the Google Play Store or Apple App Store, then register with your email and phone number. After verification, browse available cars by entering your location, date, and time. Once you've selected your preferred vehicle, review the total cost and confirm your booking. On the day of your rental, use the app to unlock the car at the pickup point and begin your journey. After your trip, return the car to the designated drop-off area, lock the car via the app, and complete your rental.

2.3.3 Klezcar

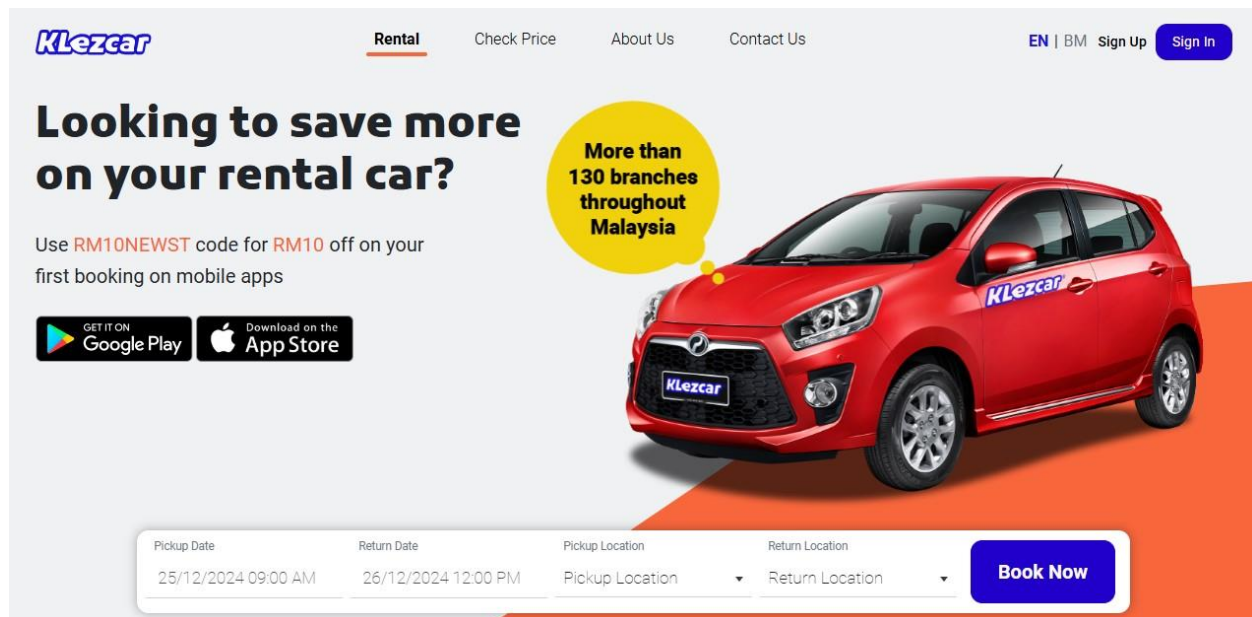


Figure 2.9 Klezcar Dashboard

Figure 2.9 illustrates the main dashboard of Klezcar, offering quick access to available cars, ongoing rentals, and support. The layout emphasizes ease of navigation and real-time rental status tracking.

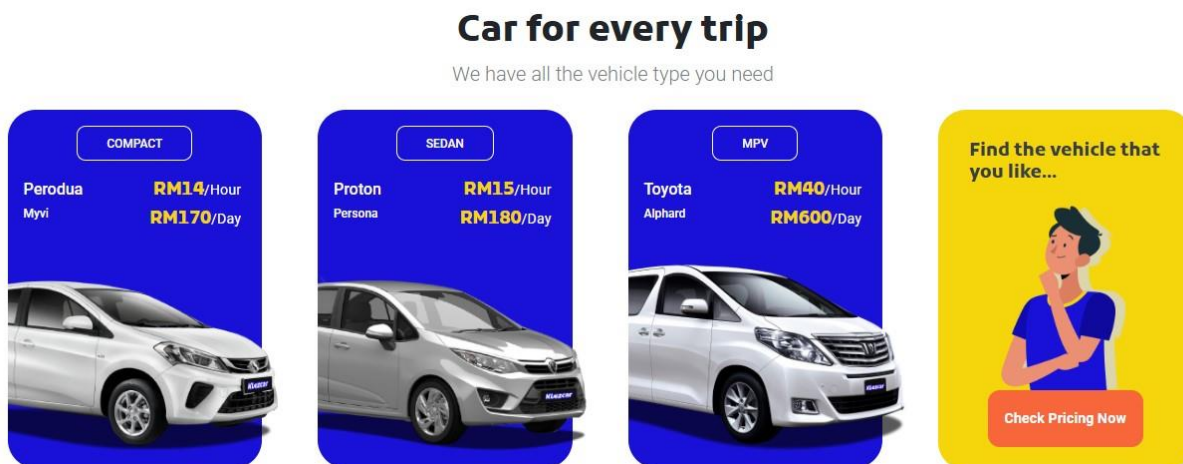


Figure 2.10 Klezcar Car List

Figure 2.10 shows a categorized list of cars offered by Klezcar. Cars are arranged by location and type, making it easier for users to browse based on proximity or preferences.

Rental Price Check

Day

Hour

1

9

Vehicle Class

× AXIA X 1.0 (A) 2023 - 2024

Search



AXIA X 1.0 (A) 2023 - 2024

Base Price:
RM 14.00 / Hour
RM 150.00 / Day

Total:
RM 276.00
for 1 Day 9 Hours

Book Now

Figure 2.11 Klezcar Book Information

Figure 2.11 displays the booking form used by Klezcar, where users input rental duration, pick-up point, and driver details. This form ensures complete and accurate user data for smooth reservation processing.

18

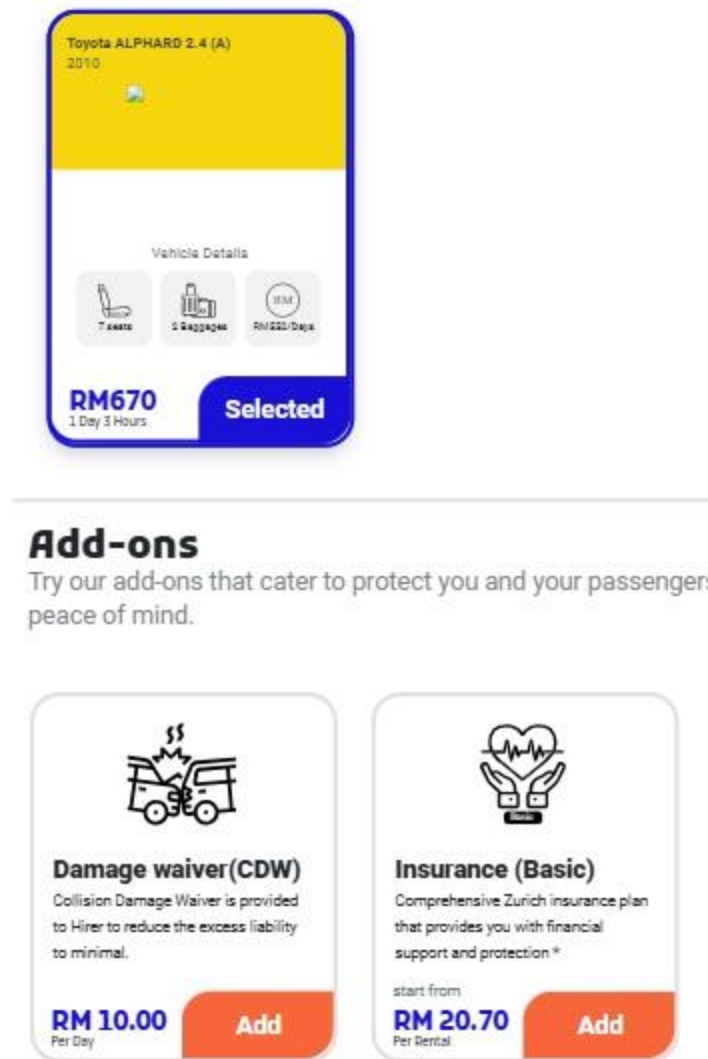


Figure 2.12 Klezcar Add-on

Figure 2.12 highlights the additional services or add-ons offered by Klezcar, such as insurance packages or GPS units. This allows users to customize their booking for added convenience or protection.

Klezcar is an online car rental service in Malaysia offering customers a seamless booking experience for short or long-term rentals. Users can select from a variety of cars with flexible rental options across multiple locations via the website and app. Features that make the entire process

quick and easy include simple booking confirmation, easy payment processing, and vehicle availability checks. Klezcar wants to serve both residents and tourists who require trustworthy transportation for various kinds of purposes.

Despite offering simple vehicle rental services, Klezcar's general accessibility is impacted by a few restrictions. Availability may be limited due to the limited number of locations, particularly in less popular areas. Additionally, booking may become more difficult during peak seasons due to strong demand for specific car models. Customers that require long-term vehicle choices may find the platform's emphasis on short-term rentals unsuitable. Additionally, users may experience annoyance due to rental time restrictions and late return fees (Klezcar, n.d.).

To use Klezcar, start by visiting their website or downloading the app. Sign up for an account to begin the booking process. Next, enter your preferred pick-up and return locations, along with the rental dates. Choose from a selection of cars that meet your preferences. Once you've selected a vehicle, proceed with payment and confirm your booking details. After confirmation, you will receive an email or app notification with your rental details. You can then collect your vehicle at the designated location at the specified time.

2.3.4 Kayak

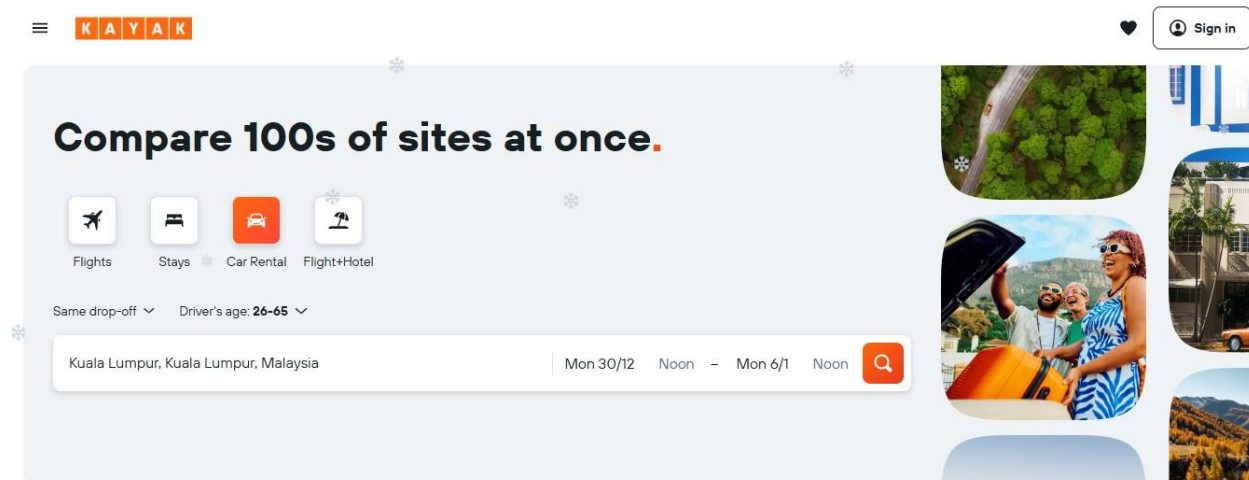


Figure 2.13 Kayak Dashboard

Figure 2.13 presents the Kayak car rental dashboard, integrating multiple rental companies into a single interface. Users can view the best offers, recent searches, and filter options, streamlining the rental comparison process.

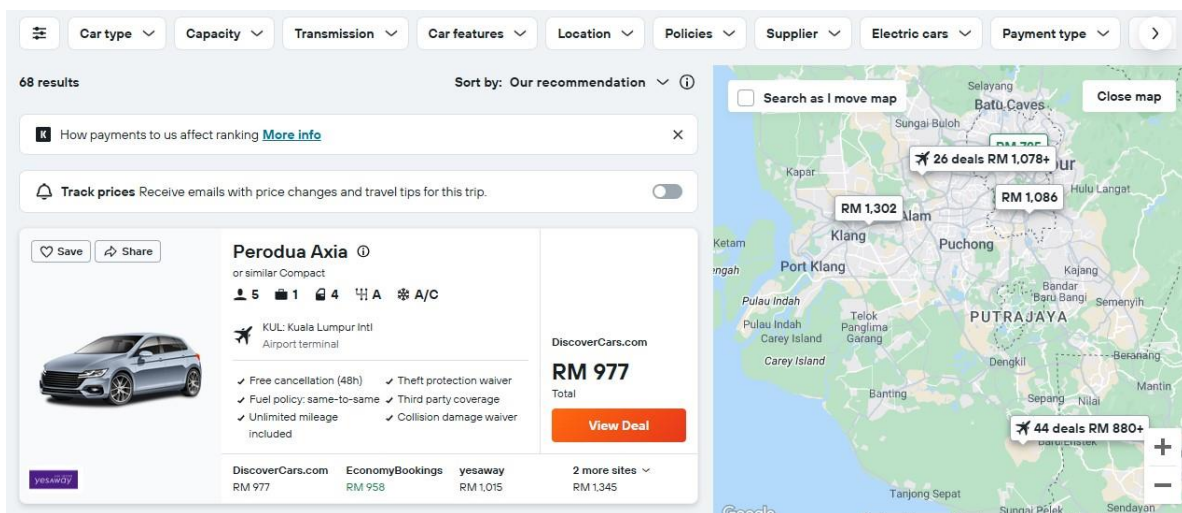


Figure 2.14 Kayak Car Information

Figure 2.14 shows detailed car descriptions for listings found on Kayak. Each car includes a rental provider, pickup policy, cancellation terms, and pricing details, which assist users in evaluating multiple providers.

Pick-up/Drop-off location Change

Kuala Lumpur Airport (KUL)

Pick-up date

Dec 30, 2024

12:00

▼

Drop-off date

Jan 6, 2025

12:00

▼

Driver's country of residence is Malaysia and age is 30-65 ⓘ

Search now

Popular filters

☐ Automatic Transmission MYR 640.75
☐ Debit card for deposit MYR 696.02
☐ 6+ seats MYR 1,487.96
☐ In terminal MYR 997.21
☐ 5 seats MYR 696.02
☐ Very good 8+ MYR 1,372.67
☐ Air Conditioning MYR 640.75

Small cars

4 1

from MYR 879.58

Medium cars

4 2

from MYR 640.75

Large cars

5 2

from MYR 696.02

SUVs

5 4

from MYR 2,068.25

Vans

7 1

from MYR 1,487.96

Premium cars

5 2

from MYR 2,498.03

Showing 110 out of 110 cars

Sort by Recommended ▼

✓ Your selected car

Perodua Axia or similar Mini

Automatic
 5 seats
 1 bag
 Air Conditioning
 4 doors

Rental Conditions

Kuala Lumpur Airport (KUL)
 Meet & Greet pick-up

Instant confirmation!

Unlimited mileage

Partial prepayment

yesAWAY

7.7 ★

Average

Total for 7 days

MYR 977.01

Free cancellation

View deal

Figure 2.15 Kayak Car List

Figure 2.15 illustrates the list of search results in Kayak. Users can scroll through options and refine results using filters like car type, price range, and rental company.

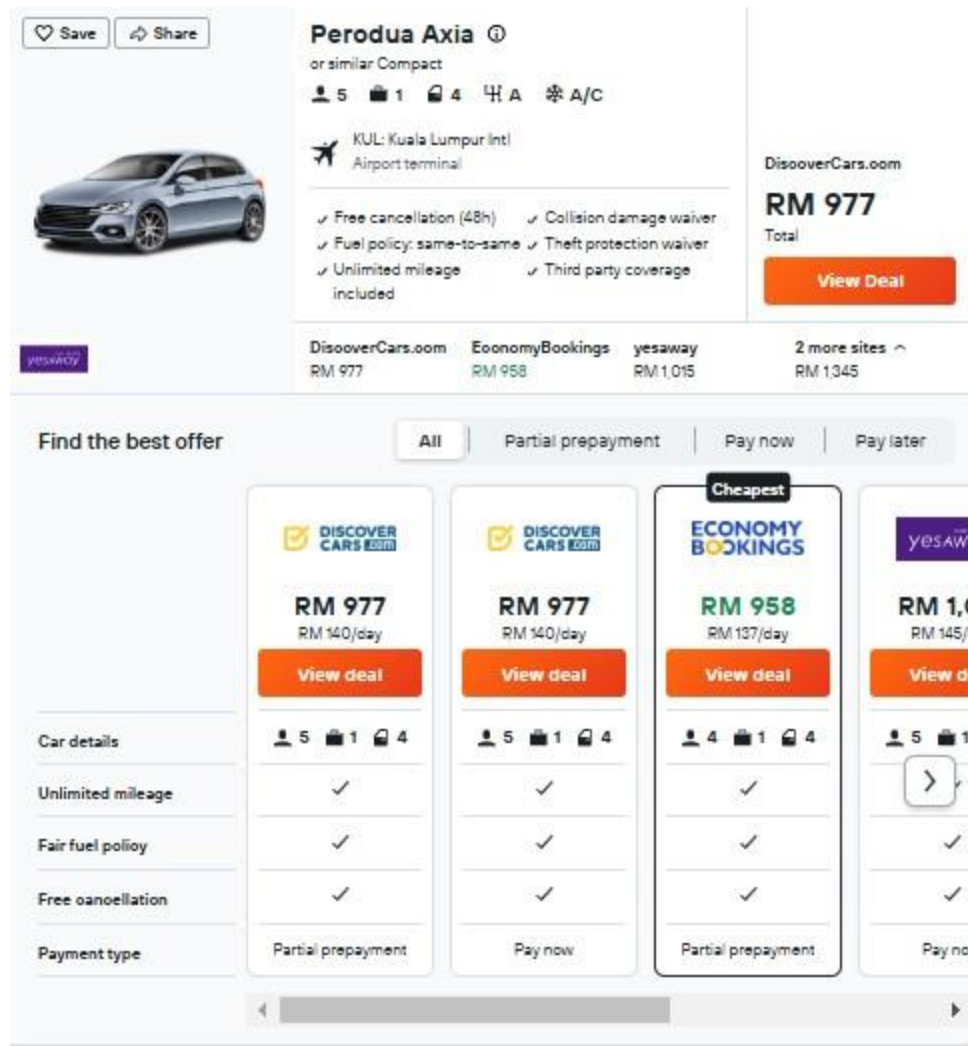


Figure 2.16 Kayak Book Choice

Figure 2.16 captures the booking selection screen in Kayak, where users finalize their car choice and proceed to confirm with the rental provider. It summarizes pricing, rental duration, and extras before redirecting to the payment gateway.

Users of Kayak's vehicle rental service can evaluate rental costs from several companies. It enables customers to filter results by location, car type, and rental business, and offers a large selection of vehicles, ranging from SUVs to economical vehicles. To improve the booking experience, Kayak

also offers customizable options like free cancellation filters. Its purpose is to assist tourists in locating the most affordable vehicle rental offers in Malaysia.

Although Kayak provides a wide range of rental choices, availability may limit the options you have. The time and location of the reservation affect the price, and there can be extra charges during pick-up. During periods of high travel demand, some users complain about inconsistent pricing or a lack of available vehicles. Additionally, during periods of high demand, customer service may take longer, which could limit the flexibility of bookings (source: "Car Rental Guide," 2020, Kayak).

To use Kayak's vehicle rental service, just go to the website, type in your rental dates and pick-up and drop-off locations, then click "Search." Numerous options, such as vehicle types and rental companies, will be displayed on the site. By using categories like car type, pricing range, and cancellation policies, users may narrow down the search results. Once they've selected their ideal vehicle, they can book it straight through the site.

2.4 Comparison Between Proposed and Existing Application

Table 2.1 Comparison Table

Feature	IkramTransports	SOCAR	Klezcar	Kayak	Smart Car Rental
User Interface (Modern)	No	Yes	No	Yes	Yes
Personalized Recommendations	No	Limited	No	Limited	Yes
Dynamic Pricing	No	Yes	No	Yes	Yes
Search and Filter Options	Limited	Yes	Limited	Yes	Yes
Seamless Booking Process	No	Yes	No	No	Yes
Multiple Payment Options	No	Yes	Limited	No	Yes
24/7 Customer Support	No	Yes	No	No	Yes
Mobile App	No	Yes	Limited	No	No
Admin Dashboard (Advanced)	No	Yes	Limited	No	Yes
Target Local and Global Users	Local	Local	Local	Global	Local and Global

Table 2.1 shows the section of a detailed comparison between the Smart Car Rental system and four existing car rental platforms: IkramTransports, SOCAR, Klezcar, and Kayak. The comparison focuses on critical features that influence usability, functionality, and user satisfaction, including interface quality, personalization, pricing strategy, support system, and accessibility. The goal is to

highlight the areas in which the proposed system offers significant improvements and to justify its development based on observed limitations in current platforms.

IkramTransports and Klezcar primarily serve local users with basic online booking functions. However, both platforms lack modern interfaces and key functionalities such as personalized car recommendations, dynamic pricing mechanisms, and seamless booking flows. Additionally, they do not support 24/7 customer service or advanced dashboards for administrative management, limiting their scalability and responsiveness to user needs.

SOCAR, on the other hand, represents a more modern solution, offering a user-friendly mobile app, dynamic pricing, and better booking integration. Despite this, SOCAR falls short in areas such as fully AI-driven personalization and global adaptability. Its support chatbot is limited in scope and is not available throughout the entire user journey. While it covers the core requirements of a car rental system, it does not cater to more advanced user expectations.

Kayak is designed as a global travel aggregator, integrating car rentals from multiple providers. It excels in comparison shopping and basic filters but lacks system-wide features such as live customer support, car status updates, and user roles beyond simple renters. Furthermore, Kayak does not offer control over the car listings themselves, as it relies on third-party suppliers.

In contrast, the proposed Smart Car Rental system combines the strengths of each platform and eliminates their weaknesses. It includes personalized recommendations based on user preferences and browsing history, a modern UI with responsive design, dynamic pricing according to demand and season, and an intelligent chatbot available 24/7. The system supports multiple payment options, including simulated online transactions, and includes an advanced admin dashboard for

car owners to manage inventory, bookings, and customer interactions. Furthermore, it targets both local and global users, offering scalability beyond the current reach of similar local-only platforms.

Through this comparison, it is evident that the Smart Car Rental App not only addresses existing gaps in the market but also provides a more complete, feature-rich experience for users and administrators alike. This justifies the system's relevance and potential impact in modernizing the car rental process for diverse audiences.

2.5 Technologies and Tools Used

The Smart Car Rental application utilizes a range of technologies and tools to deliver seamless user experience. MySQL ensures efficient database management, while Node.js powers the backend for real-time operations. Programming languages like JavaScript, Python, PHP, HTML, and CSS enhance functionality and user interface design. Development is streamlined with Visual Studio Code, while payment gateways iPay88 and PayPal enable secure, localized, and international transactions.

2.5.1 Database System

For the Smart Car Rental application, MySQL is chosen as the database system because of its reliability, efficiency, and familiarity among developers in Malaysia. It will store and manage critical information such as user profiles, car details, booking history, payment transactions, and personalized recommendations. MySQL's robust query capabilities ensure fast retrieval of data, which is essential for real-time features like car availability updates and dynamic pricing.

2.5.2 Web Server

The backend operations of the Smart Car Rental app will be powered by Node.js. This technology is ideal for handling asynchronous, real-time requests, which are critical for tasks like booking confirmations, personalized recommendations, and AI-driven pricing updates. Its ability to process multiple requests simultaneously ensures a seamless experience for users, even during high-demand periods.

2.5.3 Programming Languages

The application employs a combination of modern and versatile programming languages:

- JavaScript serves both the frontend and backend, enabling dynamic content updates, interactivity, and server-side logic within the Node.js environment.
- Python powers advanced AI features, such as the recommendation system and demand-based dynamic pricing, leveraging libraries.
- PHP is integrated for backend tasks, including handling API communication, user authentication, and admin dashboard functionalities.
- HTML and CSS are used to create a visually appealing and responsive user interface, ensuring the app works well on various devices.

2.5.4 Tools

Development for the app is streamlined using Visual Studio Code, a feature-rich IDE that supports all chosen technologies and facilitates debugging and code versioning. For payment solutions, iPay88, a trusted Malaysian payment gateway, ensures secure and localized transactions, while PayPal provides a globally recognized option for international users. These tools enhance the app's reliability and accessibility, ensuring smooth financial transactions for users.

By leveraging these technologies, the Smart Car Rental app ensures efficient performance, advanced AI-driven personalization, and secure payment processing, making it a competitive and innovative solution in the car rental industry.

2.6 Conclusion

The review of IkramTransports, SOCAR, Klezcar, and Kayak shows the different advantages and disadvantages of the current rental car services. Despite their increased efficiency and accessibility, these systems often lack dynamic pricing, smart personalized service, and simple management tools. The suggested Smart Car Rental application, which incorporates modern functions suited to user preferences and market demands, is one example of a creative solution made possible by these limitations. The Smart Car Rental app promises to improve customer experience and operational efficiency by filling in automation and user-centric design gaps.

The suggested Smart Car Rental app sets itself apart with features like real-time dynamic pricing and AI-driven personalized recommendations. These improvements guarantee affordable prices based on demand while allowing customers to locate cars that meet their specific requirements. Furthermore, a thorough management dashboard and automated customer service make the process easier for administrators and users alike. With these benefits, the app serves as a flexible solution that improves convenience and enjoyment for everyone involved.

In conclusion, in contrast to existing platforms, the Smart Car Rental application offers a more creative and user-focused approach. A smooth and effective car rental experience is produced by resolving the issues found in the systems under assessment. Its emphasis on customization, automation, and dynamic functioning guarantees that it meets the current needs of businesses as well as customers, making it a game-changing weapon in the vehicle rental market.

Chapter 3: Methodology

3.1 Introduction

The requirement analysis was covered in the first part of this chapter. The Rapid Application Development Model, one of the System Development Life Cycle (SDLC) methodologies, is used in the second part to describe how this project progressed during the development process.

3.2 Methodology

The Rapid Application Development (RAD) model is chosen as the methodology to develop the Smart Car Rental system due to its focus on quick prototyping and iterative feedback. This approach allows developers to provide clients with working prototypes at regular intervals, enabling continuous demonstrations and immediate feedback. By entering sample data and providing feedback on any necessary modifications or enhancements, users can test these prototypes. Compared to depending only on documentation, early prototypes offer a tangible picture of the system, even though they might not be fully functional. This helps clients comprehend the system better.

This methodology emphasizes active client collaboration throughout the development process. By involving clients in every phase, developers can ensure that the system meets all requirements and expectations. Clients can see features like dynamic pricing and personalized car suggestions with prototypes, which facilitate modifications. Developers move forward with implementing the system's complete functionality after the customer gives their approval to the prototype, making sure the finished product meets user expectations and needs. The diagram of the Rapid Application Development Model methodology is shown in Figure 3.1.

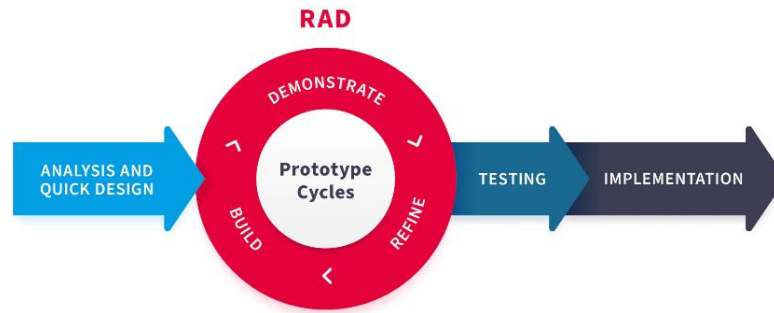


Figure 3.1 Rapid Application Development Model

3.2.1 Analysis and Quick Design

The first phase of the Rapid Application Development Model is analysis and design stage. Before analysis, user requirements and other system specifications are collected through several interview sections which were carried out in the UNIMAS to have better understanding of the situation and problems faced by the user. To create a new car rental system that meets user demands, expectations and user requirements are gathered and examined. The objective of the Kereta Sewa UNIMAS's Facebook (KSU) for Smart Car Rental App project is to improve the current system's car rental system.

Five users of the current car rental service have been invited to join the interview section. The users of the current car rental system are the UNIMAS students. There are several questions prepared to interview the users of the current system. The opinions and feedback of the users are collected, which helps to determine the requirements for the proposed system. The information about the interviewees is shown in Table 3.1 below.

Table 3.1 Interviewee

Name and Matrix Number	Faculty
Lovelyn Felorina Anak Julius (79913)	Faculty of Computer Science and Information Technology
Mohd Haziq Izzan bin Zamahari (81892)	Faculty of Computer Science and Information Technology
Nicholas Lim (80375)	Faculty of Computer Science and Information Technology
Machowell Gaing Anak Malang (79936)	Faculty of Engineering
Nur Altatika binti Abdullah (80523)	Faculty of Computer Science and Information Technology

During the interview, the users mentioned that KSU is using an old school method. It is hard to tell whether the car is still available or not. The posting there sometimes doesn't update, and the available car is no longer available. The user must scroll down the comment section to see whether the car is available or not and user had to contact directly to the owner.

To determine whether new features should be added to the software, or which changes should be made to the current features, all the requirements and problem statements must be studied after the target users' requirements have been met. During the requirement collection phase, the system and user requirements are identified.

i. User Requirements

There are two types of user requirements that will be discussed in this section, which are the functional and non-functional requirements as shown in Table 3.2.

Table 3.2 Functional and Non-Functional Requirements

Functional Requirement	Non-Functional Requirement
Login The system should only allow authorized users to access their accounts using secure credentials.	Availability The system must maintain high availability, especially during peak booking hours.
Car Search and Booking - The system enables users to search for available cars based on location, type, or preferences. - The system shall allow users to book selected cars for specified dates and times.	Accessibility The system shall be compatible with various devices, including desktops, tablets, and smartphones.
Personalized Recommendations The system should provide personalized recommendations based on user preferences and history.	Scalability The system shall be scaled to support additional cars, users, and locations as the business grows.
Payment - The system accepts various payment methods, including iPay88, PayPal, debit card, and credit card. - The system shall calculate payment details, including discounts and dynamic pricing, accurately.	Security The system shall enforce data encryption and secure payment processing to ensure user data protection.

User Roles and Management • The system should provide different interfaces and access permissions for customers, admins, and support staff. • The system shall allow admins to manage car inventories, bookings, and user accounts.	Validation • The system shall validate all user inputs, including dates, payment details, and booking information, and notify users of any errors.
Search and Filtering • The system should allow users to filter cars based on features, price range, and availability. • The system shall support searching rental history and transactions for both users and administrators.	Data Integrity • The system shall ensure accurate handling and storage of user, booking, and payment data.
Customer Support • The system shall include a chatbot for 24/7 assistance with queries and issues.	
Print a) The system shall allow users to print booking details, receipts, and invoices.	

ii. System Requirements

The system requirements outlined in this section detail the necessary software and hardware components essential for developing the proposed Smart Car Rental App. Tables 3.3 and Table 3.4 provide a comprehensive list of the hardware and software requirements for this project.

Table 3.3 Hardware Requirements

Hardware	Specification
Laptop	Yoga 6 13ALC6
RAM	8.0 GB
Processor	AMD Ryzen 5 5500U with Radeon Graphics 2.10 GHz
Operating System	Window 10

Table 3.4 Software Requirements

Software	Function
XAMPP	Cross platform of web browser
Visual Studio Code	To design coding for the interface and functions
Web Browser	Display web-based system
PHP, HTML, JavaScript	Programming language
MySQL	Database management system

Once the features to be added, modified, or removed have been identified, the updated workflow for the smart car rental system is designed, as illustrated in Figure 3.2. The features requiring revision or addition are prioritized based on importance. This step is completed when both the user and supervisor reach an agreement on the new system's features within the specified timeframe.

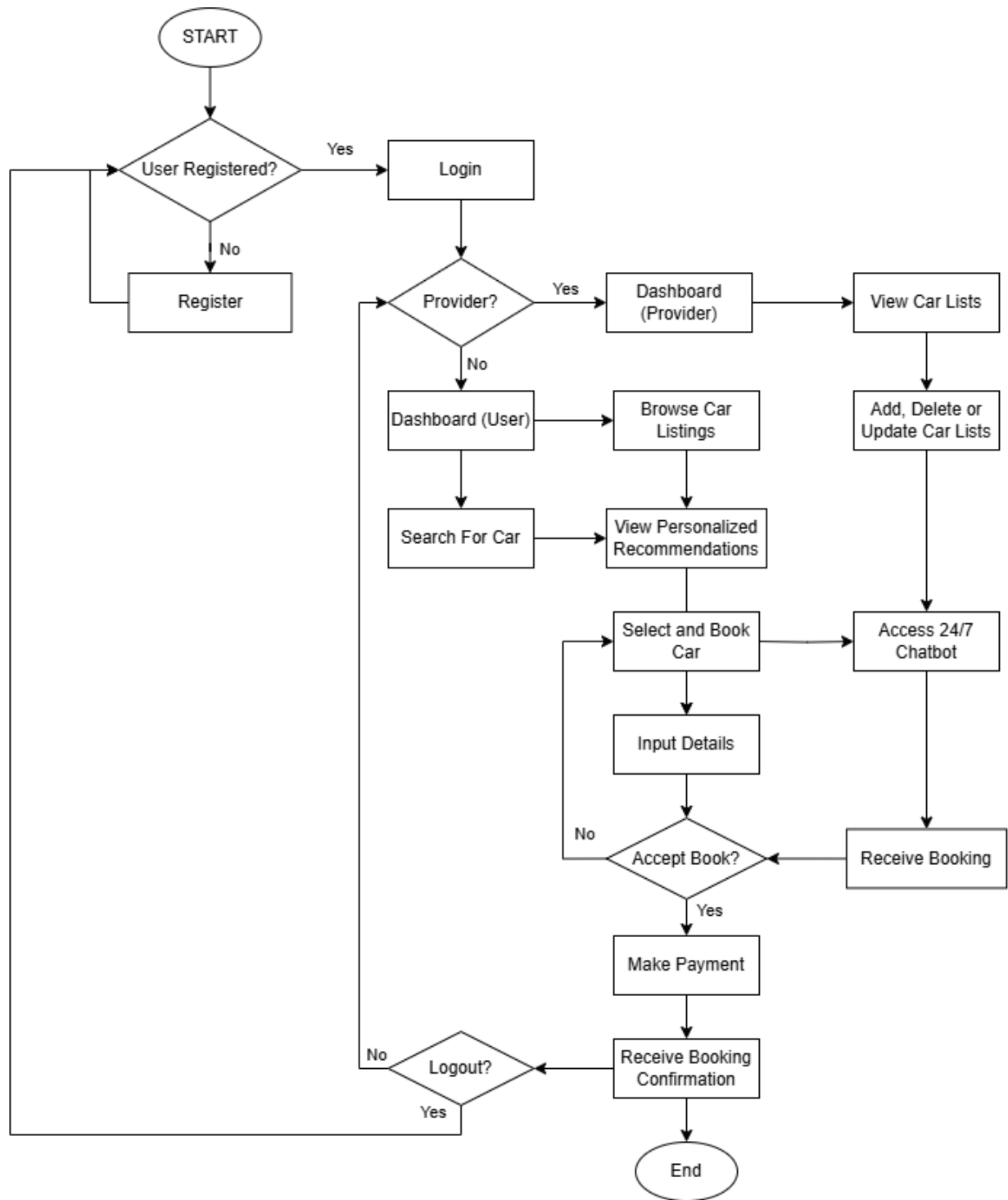


Figure 3.2 Flowchart of Smart Car Rental App

Following that, the new system's design is planned before any coding is done. Two diagrams and one table - the use case diagram, class diagram and data dictionary - are used to create the system architecture that satisfies user requirements. The suggested system's use cases are described in the use case diagrams. ERD are used in databases to design the relationships between all the classes in the system.

a) Use Case Diagram

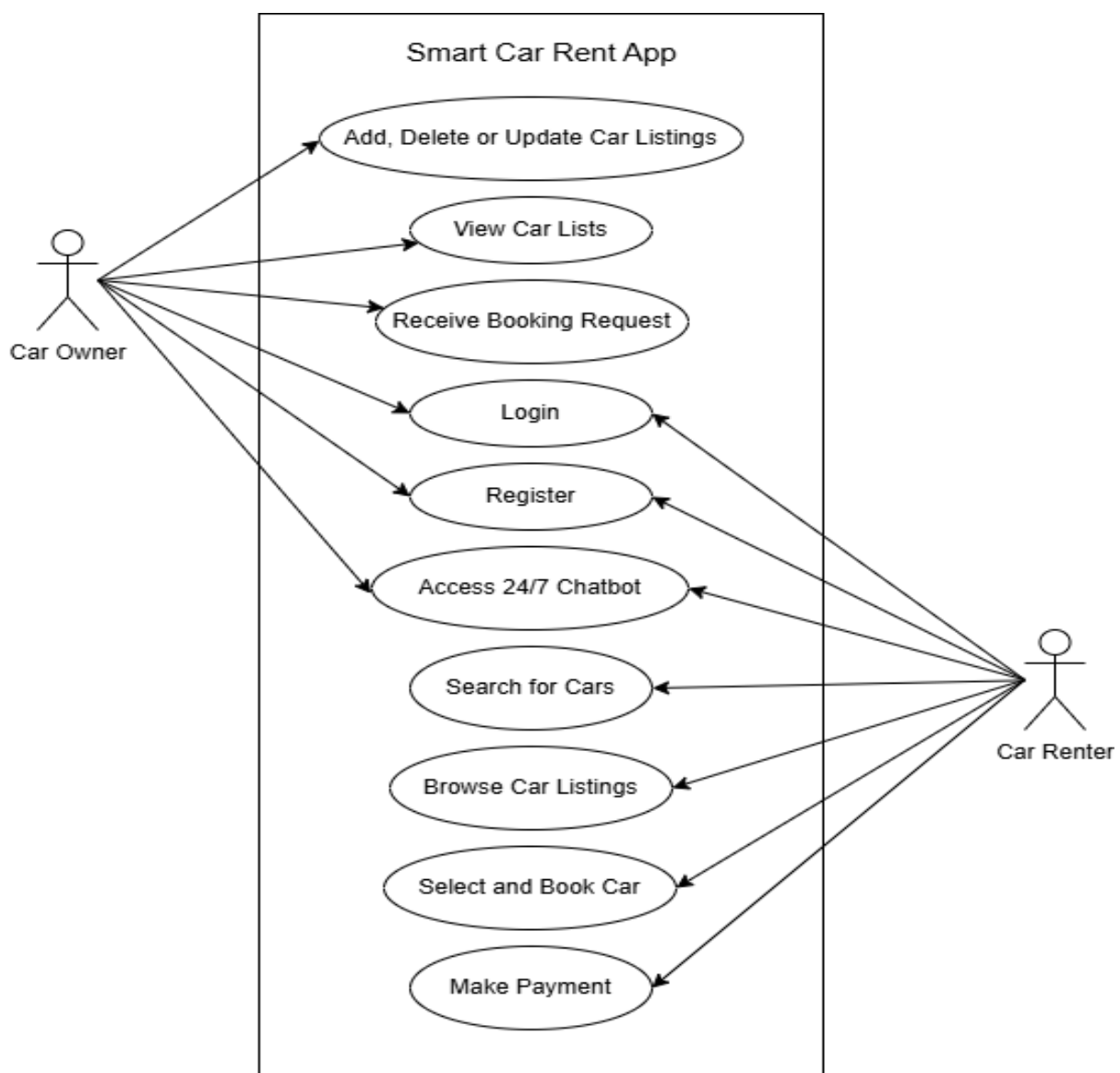


Figure 3.3 Use Case Diagram

b) Entity-Relationship Diagram

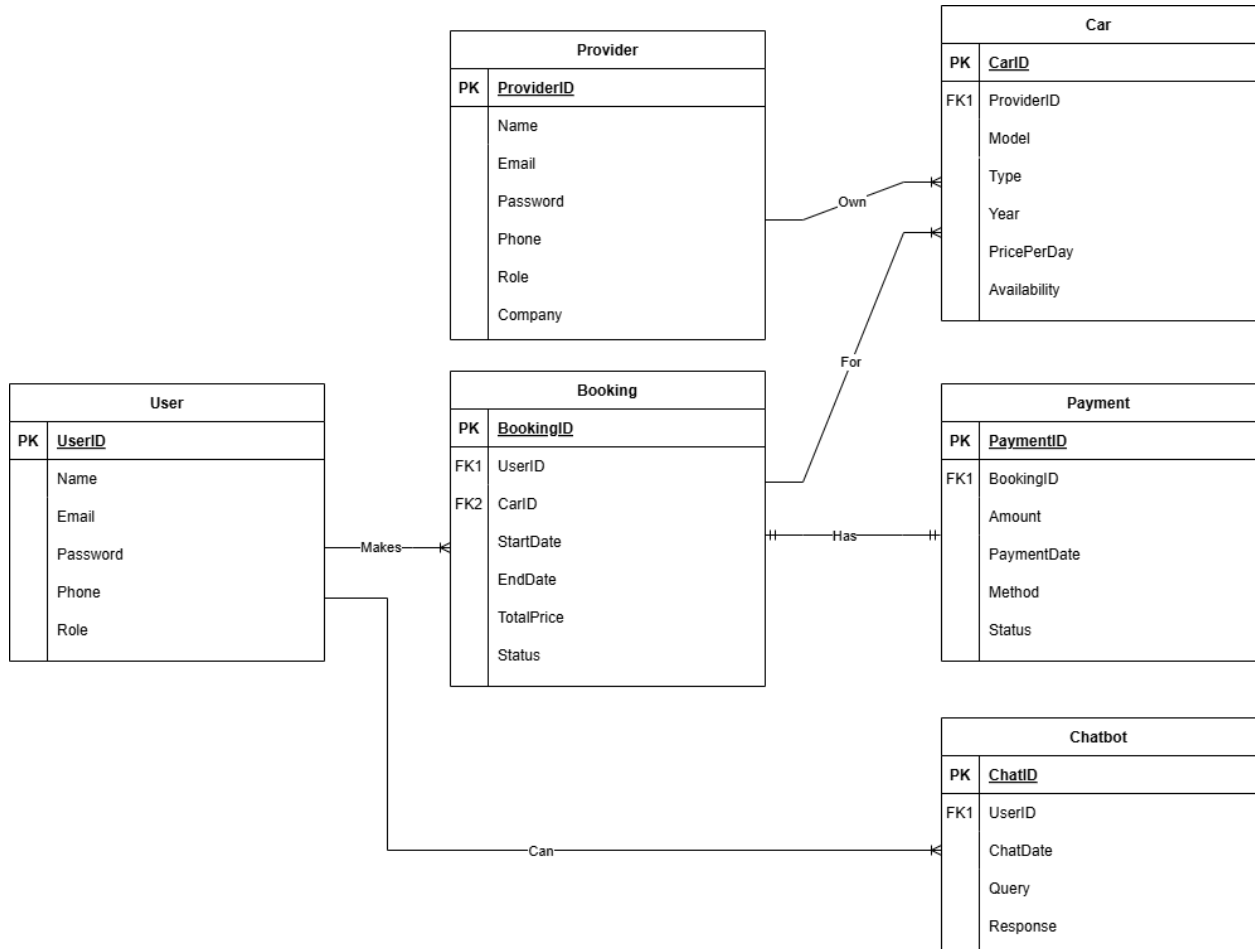


Figure 3.4 Entity-Relationship Diagram

Figure 3.4 shows the ERD of Smart Car Rental Apps. There are 6 classes in the database. In each class, there is one Primary Key and one or more Foreign Key that will be essential for the database.

c) Data Dictionary

There are 6 data dictionaries of 6 entities as shown below, which describe the objects stored in the database of the Smart Car Rental App.

Table 3.5 Data Dictionary

Entity	Attribute	Type
User	UserID	INT (PK)
	Name	VARCHAR (100)
	Email	VARCHAR (100)
	Password	VARCHAR (50)
	Role	VARCHAR (50)
Car	CarID	INT (PK)
	OwnerID	INT (FK)
	Model	VARCHAR (50)
	Year	INT (4)
	PricePerDay	DECIMAL (10,2)
	Availability	VARCHAR (10)
Booking	BookingID	INT (PK)
	UserID	INT (FK)
	CarID	INT (FK)
	StartDate	DATE
	EndDate	DATE
	TotalPrice	DECIMAL (10,2)
	Status	VARCHAR (50)

Payment	PaymentID	INT (PK)
	BookingID	INT (FK)
	PaymentMethod	VARCHAR (50)
	PaymentDate	DATE
	AmountPaid	DATE
Chatbot	ChatID	INT (PK)
	ChatDate	DATE
	Query	INT (100)
	Response	VARCHAR (100)

iii. Interface Design

Interface design is used to plan the design of the system which focuses on interaction and user experience. As mentioned by Interaction Design Foundation (2016), User interface (UI) design is the process designers use to build interfaces in software or computerized devices, focusing on looks or style.

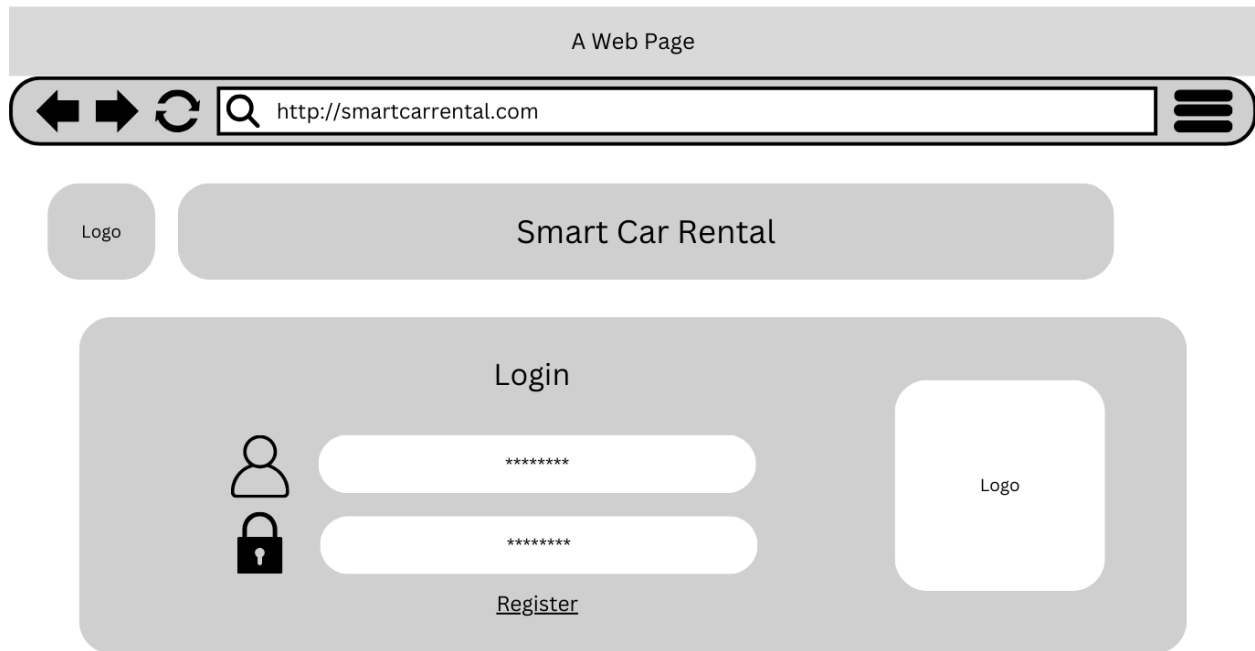


Figure 3.5 Login Page

Figure 3.5 shows the login interface ensures secure access to the system by requiring users to input their registered email and password. It incorporates validation to prevent unauthorized access. Users are notified of errors, such as incorrect credentials, and provided options to recover passwords or seek assistance through the chatbot.

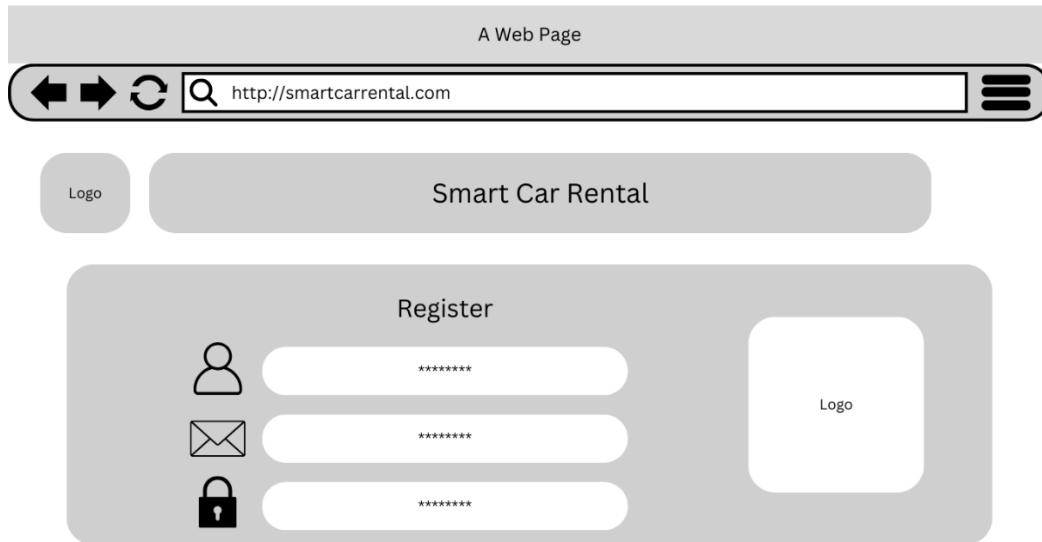


Figure 3.6 Register Page

Figure 3.6 shows the registration interface facilitates account creation for new users. It collects essential details, such as name, email, password, and user role (renter or owner). The interface validates inputs to ensure data accuracy and security. Users receive confirmation upon successful registration and are redirected to the login page.

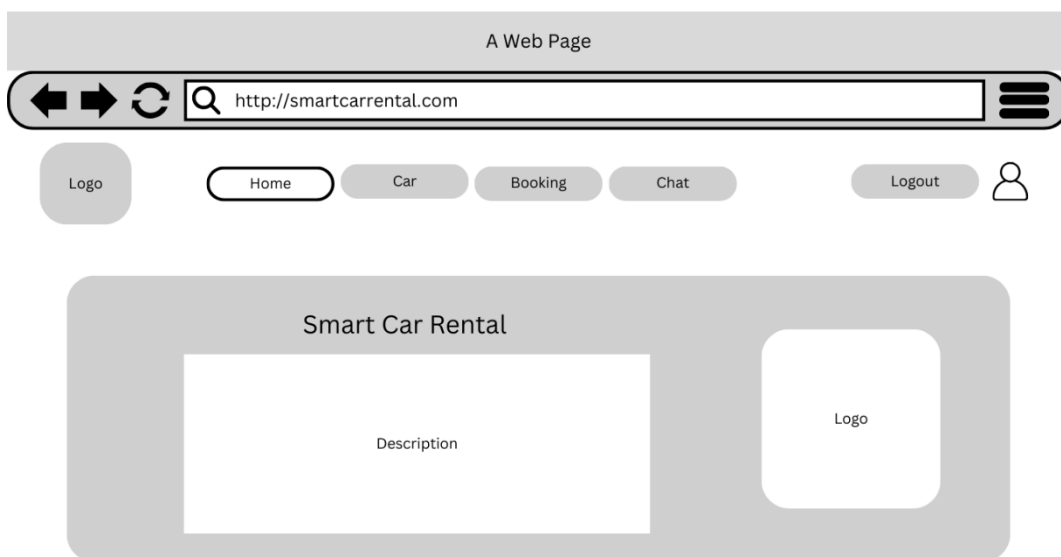


Figure 3.7 Homepage

Figure 3.7 shows the dashboard acts as the central hub for users, displaying personalized recommendations, shortcuts to key features, and user-specific notifications. Renters can view available cars or ongoing bookings, while car owners can manage listings.

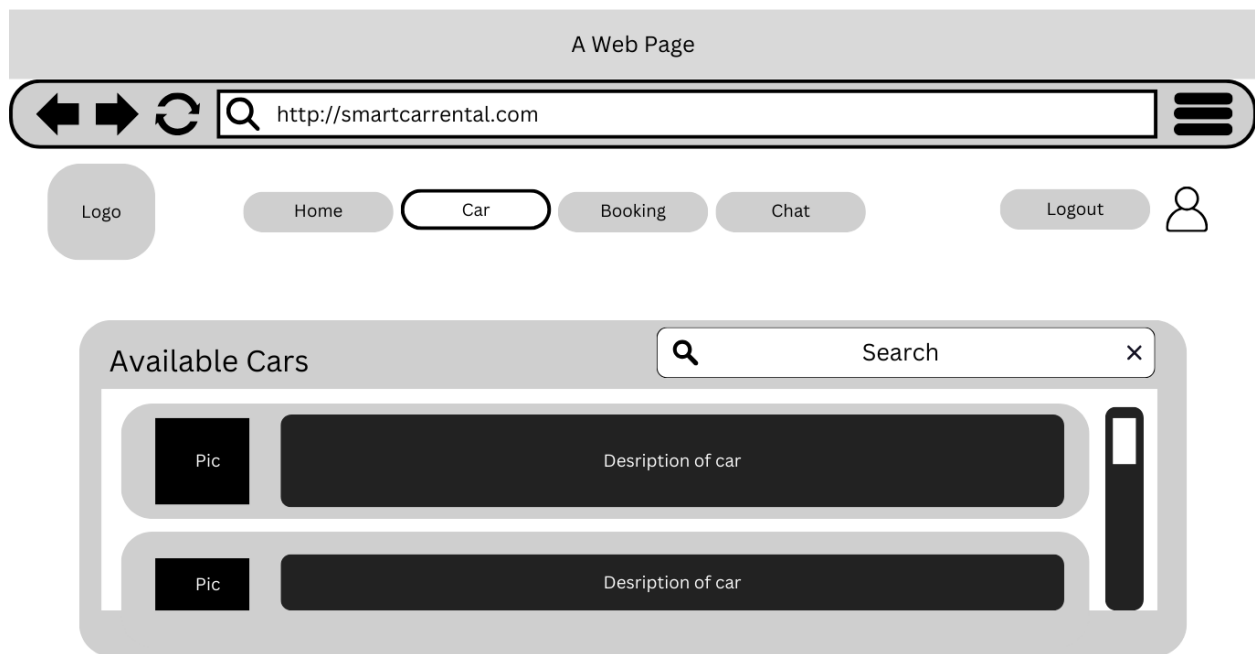


Figure 3.8 Available Cars

Figure 3.8 shows the interface showcases a catalog of cars available for rent. Users can view car details, such as model, price, and availability, with options to filter by preferences like location, price range, or car type.

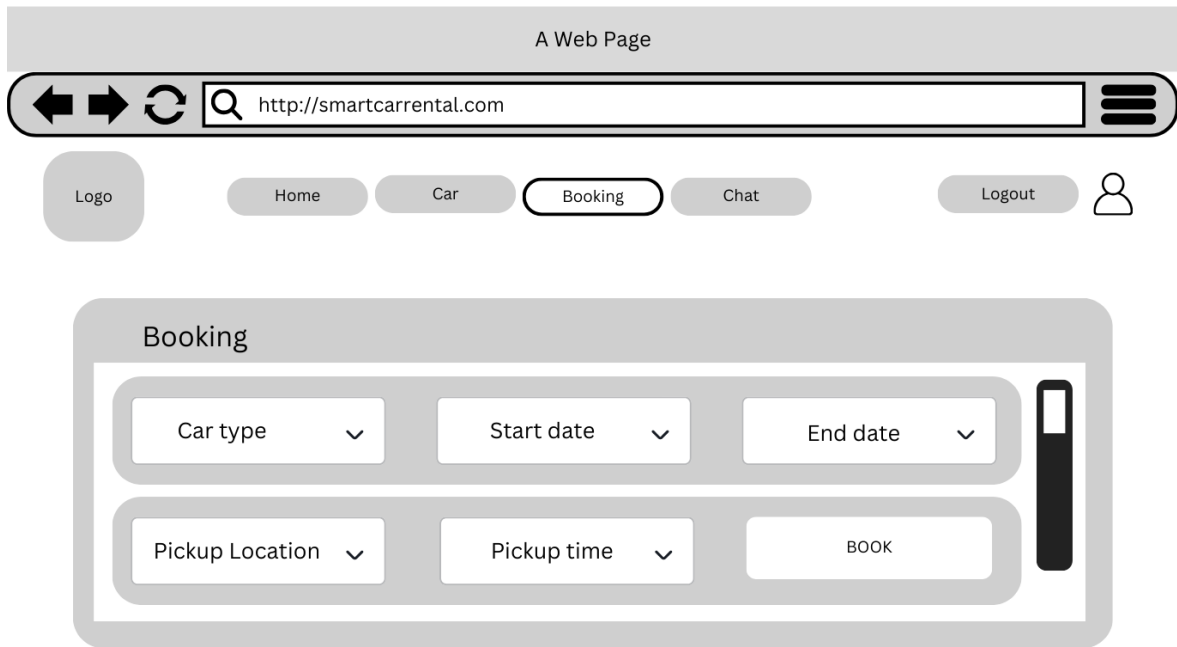


Figure 3.9 Booking Page

Figure 3.9 shows the booking interface allows users to select a car, specify rental dates, and review booking details, including pricing breakdowns. Users can confirm their reservation and make payments securely. This interface integrates with payment systems like iPay88 and PayPal, ensuring a seamless and trustworthy transaction process for renters.

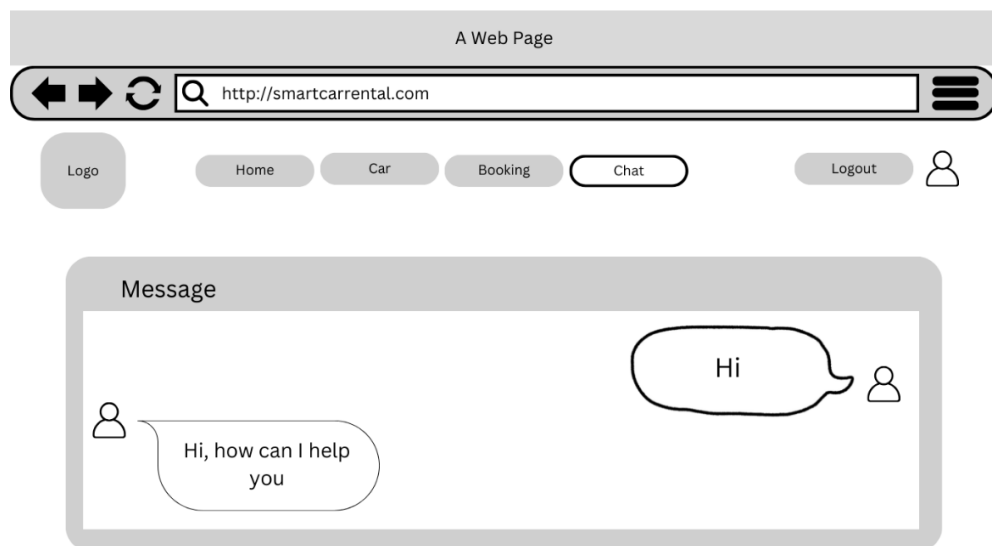


Figure 3.10 Chat Page

Figure 3.10 shows the chatbot interface offers round-the-clock assistance to users. It addresses common queries, provides system navigation guidance, and resolves issues promptly. Users can also receive personalized support, such as booking help or car management tips.

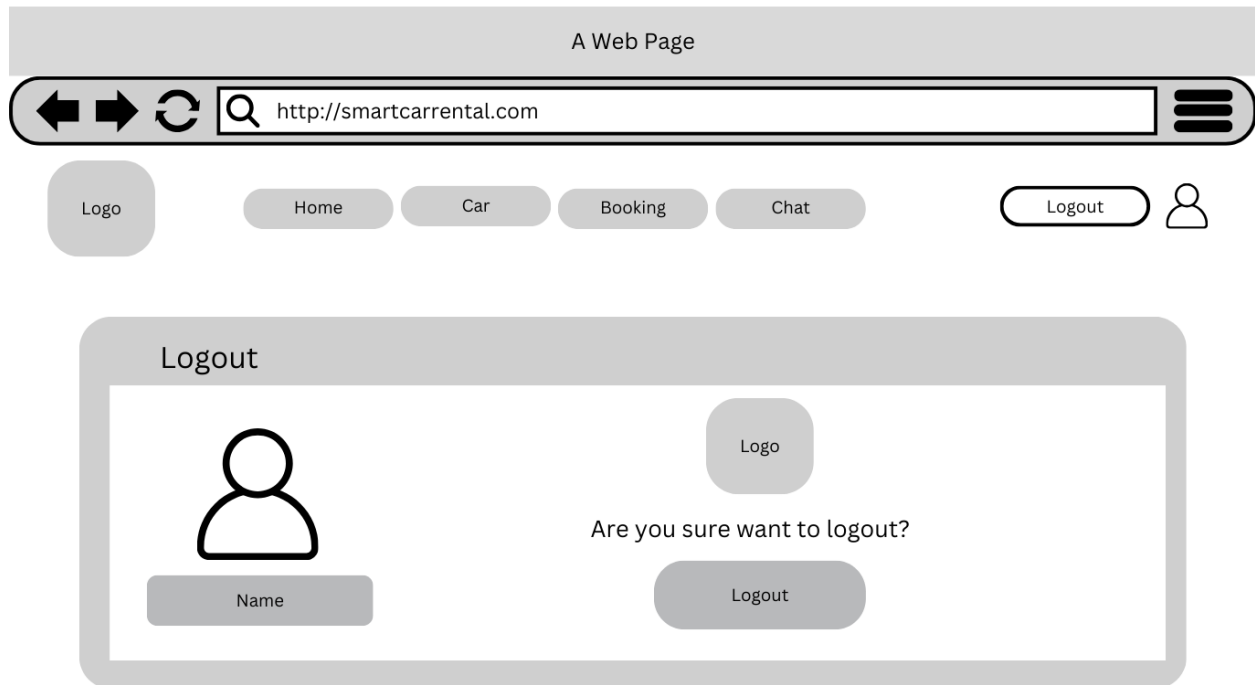


Figure 3.11 Logout Page

Figure 3.11 shows the logout interface securely ends the user's session. It ensures all data is saved before exit and provides a confirmation prompt to avoid accidental logouts.

3.2.2 Prototype Cycle

There are three primary processes in the prototype cycle: build, demonstrate, and refine. In the building process, the construction phase involves coding and debugging the selected features based on the analyzed requirements. For the smart car rental system, these features are implemented using PHP and HTML to ensure functionality and compatibility.

After the prototype is constructed, it is demonstrated to both clients and users. They review the prototype and provide valuable feedback, identifying any issues or areas for improvement. This feedback is crucial for the refined stage, where necessary modifications are made to align the system with user expectations. These phases are repeated until the client and users are fully satisfied with the product. Once no further feedback is received, the prototype moves to the software testing phase for final validation.

3.2.3 Software Testing

System testing and user acceptance testing are conducted to ensure the prototype is free from execution and logical errors. During the testing phase, the system is evaluated for functionality, reliability, efficiency, and usability to verify its readiness. These tests are essential steps before deploying the Smart Car Rental App.

3.2.4 Implementation

In the final stage of the Rapid Application Development Model, the Smart Car Rental App is implemented based on the finalized prototype approved by the client and users. The client validates and verifies the system to ensure it aligns with the objectives and requirements. Once approved, the system is delivered for integration. Initially, the new system operates alongside the existing system, and upon confirming its stability, a complete transition is made.

3.3 Summary

In conclusion, this chapter outlines the methodology and emphasizes the analysis of system requirements and design. Interviews were chosen as the requirement-gathering technique to understand user needs. Several UML diagrams were developed to provide a clearer representation of the proposed system. Additionally, focusing on the design at this stage helps minimize errors during implementation. The next chapter will delve into the implementation of the system.

Chapter 4: Implementation

4.1 Introduction

The Smart Car Rental App actual implementation is covered in this chapter, along with information on the technologies, system architecture, user roles, and construction of each module. The development process adhered to the Rapid Application Development (RAD) methodology, which was presented in Chapter 3 and places a strong emphasis on iterative prototyping and ongoing feedback.

Using HTML, CSS, JavaScript, PHP, and MySQL as the relational database, the system is constructed as a web-based platform. Owner and Renter are the two user roles that the app is made to support, and each has unique features and permissions.

4.2 System Architecture

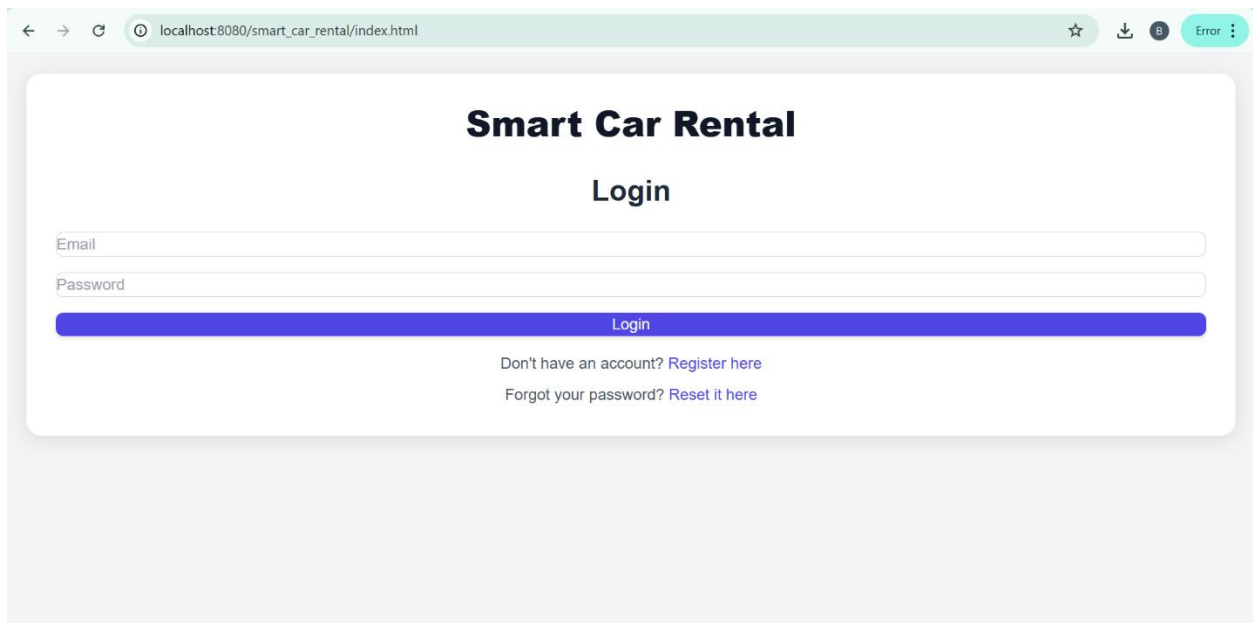
The web architecture used by the application is three-tiered:

- Frontend (client-side): UI files are created with HTML, CSS, and JavaScript.
- Backend (server-side): PHP was used in the development of the server-side backend to handle requests and communicate with the database.
- Database: User information, vehicle listings, reservations, chat messages, and payment history are all stored in a MySQL database.

4.3 System Interfaces

Both owners and renters can use the website's user-friendly interface. The main UI modules are listed below:

4.3.1 Login and Registration



The screenshot shows a web browser window with the address bar displaying 'localhost:8080/smart_car_rental/index.html'. The page content is centered and features the title 'Smart Car Rental' in a large, bold font. Below the title is the word 'Login' in a slightly smaller font. There are two input fields: 'Email' and 'Password'. Below these fields is a prominent blue button labeled 'Login'. At the bottom of the form, there are two links: 'Don't have an account? Register here' and 'Forgot your password? Reset it here'.

Figure 4.1 Smart Car Rental App's login page

Figure 4.1 shows the Smart Car Rental App's login page, which users must access by entering their email address and password. The backend verifies user roles for appropriate redirection, and the form employs validation to guarantee accurate input. An error message advising the user to try again will show up if they enter incorrect credentials.

Smart Car Rental

Register

Email

Password

Renter

Register

Already have an account? [Login here](#)

Figure 4.2 Smart Car Rental App's registration page

Figure 4.2 shows the registration form, allowing new users to create an account by providing their name, email address, password, and choice of user role (owner or renter). The system instantly alerts users to form input errors and verifies that emails are unique. The user is taken to the login page to view their dashboard after successfully completing the registration process.

4.3.2 Dashboard

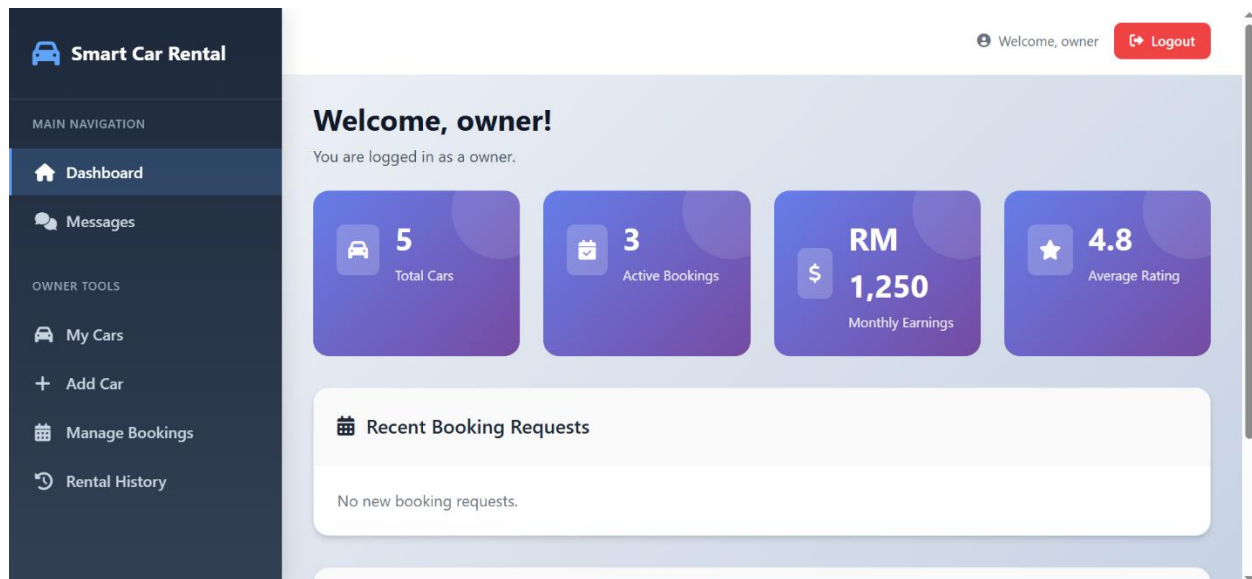


Figure 4.3 Smart Car Rental App's owner dashboard

Figure 4.3 shows the dashboard page for owners, where they can handle all car-related tasks. This includes accessing booking history, viewing current cars, and updating car statuses. A brief overview of available cars, pending confirmations, and current reservations is given by the interface. Owners can effectively manage their rentals without having to navigate through numerous pages thanks to the design.

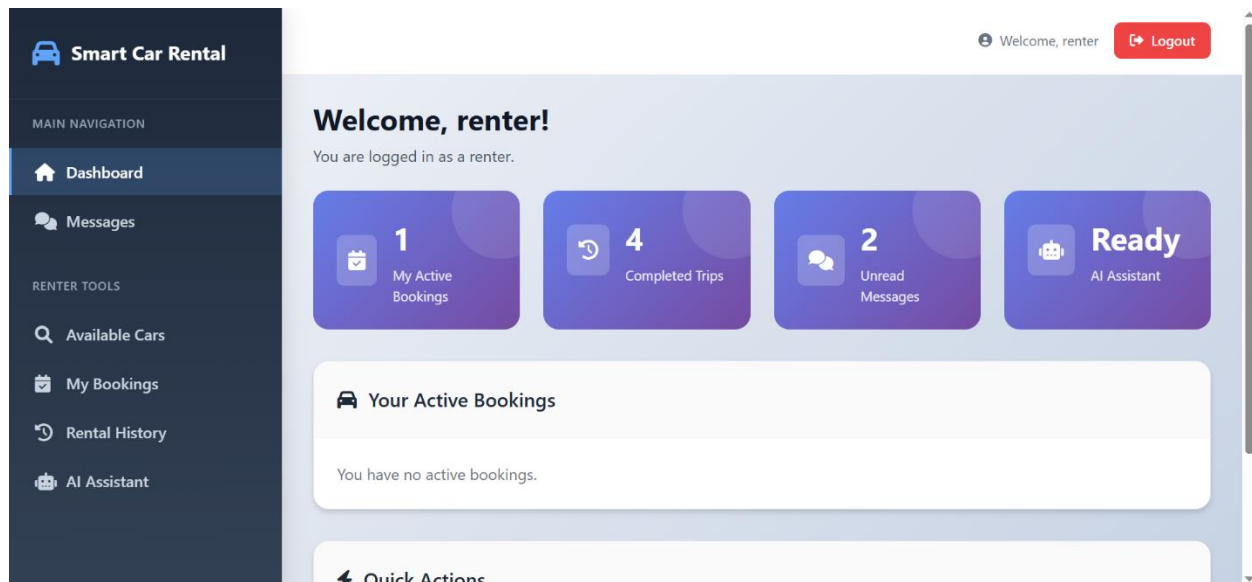


Figure 4.4 Smart Car Rental App's renter dashboard

The renter dashboard, which shows a summary of recent reservations, vehicle availability, and alerts for notifications, is depicted in Figure 4.4. Renters can quickly look through available cars, monitor the status of their reservations, and start conversations with owners. The layout is designed to improve responsiveness and user experience by making it easier to access the most used features.

4.3.3 Car Management (Owner)

Figure 4.5 Smart Car Rental App's add/edit car

The add/edit car form, which enables owners to list new cars or update details of existing listings, is shown in Figure 4.5. Car brands, type, transmission, color, price, and location are among the fields on the form. Validation helps to maintain the accuracy of the system's car listings by ensuring that all data entered is consistent. Quick updates are encouraged by the user-friendly interface.

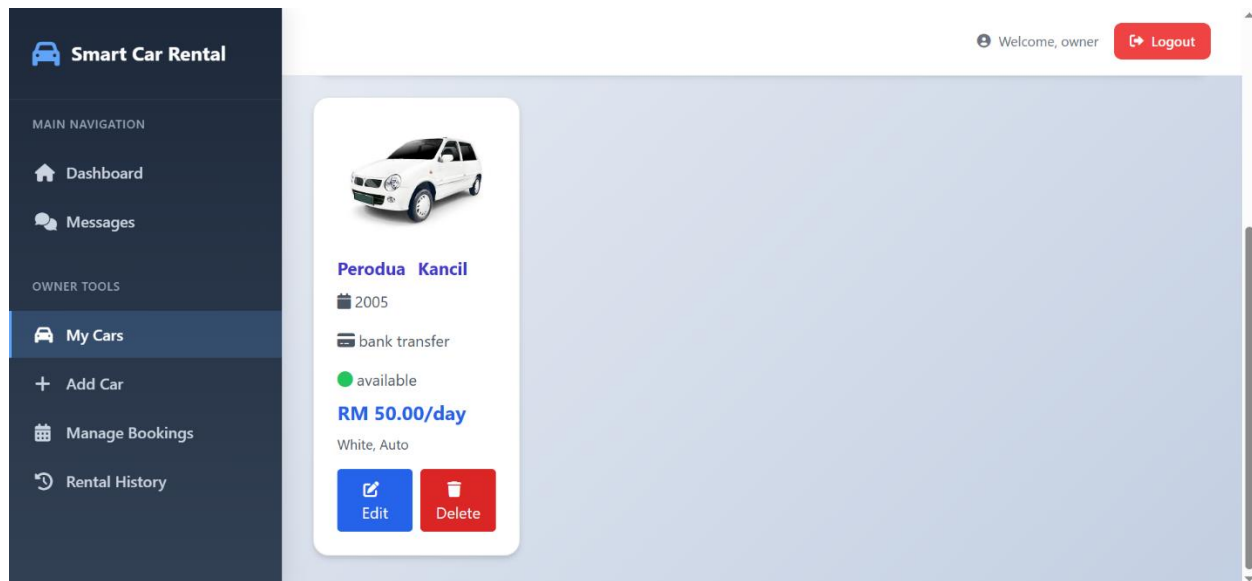


Figure 4.6 Smart Car Rental App's list of cars

The list of cars linked to a logged-in owner is shown in Figure 4.6. Basic details about the vehicle are displayed in each row, along with options to edit, delete, or update status. Car statuses include booked, confirmed, pending, and available. Owners can efficiently monitor and manage their car inventory thanks to the structured display.

4.3.4 Booking Process (Renter)

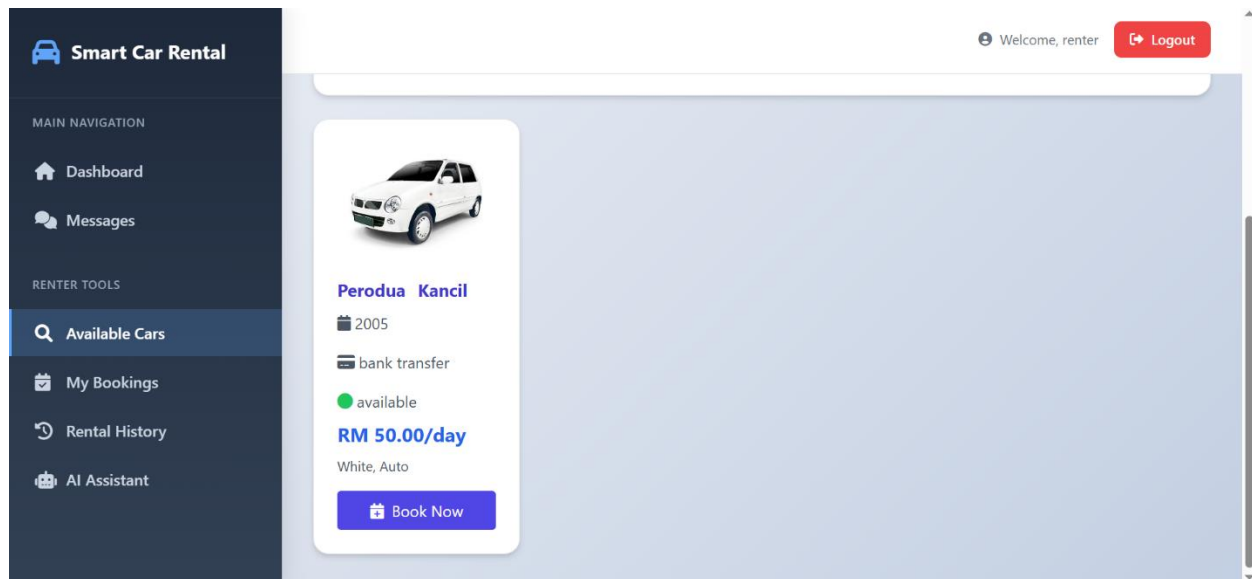


Figure 4.7 Smart Car Rental App's search and filter car listing

Figure 4.7 shows that renters can search and filter car listings from all owners in the available cars view. Renters can use filters to select vehicles based on brand, location, color, and gearbox type. Essential information like the daily car price and availability status are included in every listing so that renters can make well-informed reservations.

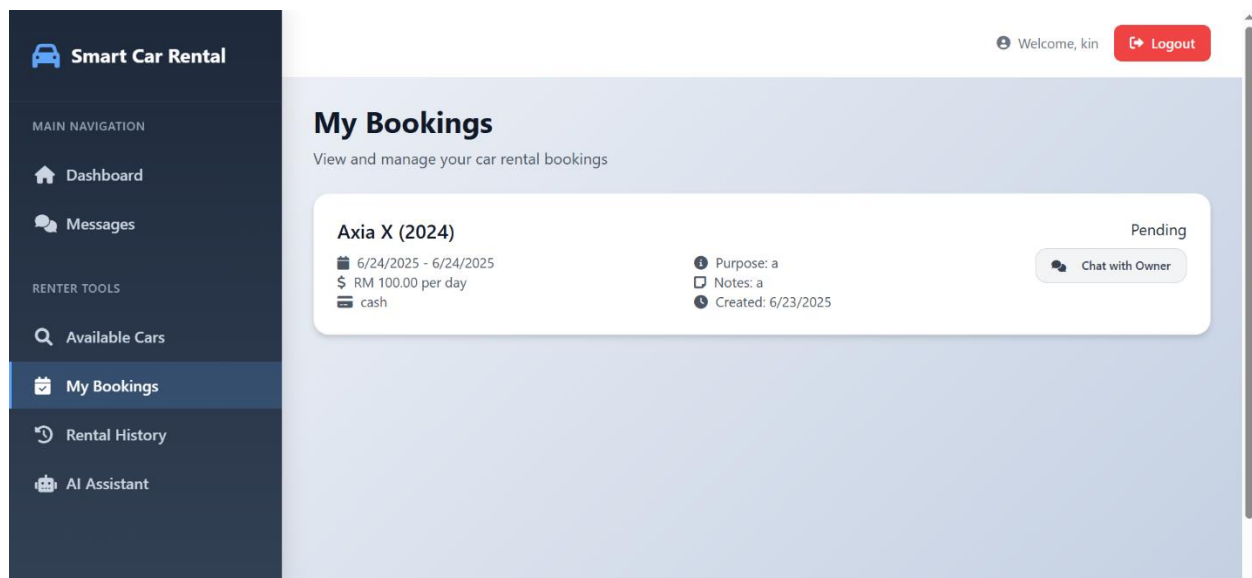


Figure 4.8 Smart Car Rental App's booking confirmation

The booking confirmation page, where renters enter their rental dates and submit a booking request, is shown in Figure 4.8. The booking status is set to pending after the backend determines the total rental cost. The status is changed to booked as soon as the owner agrees and payment is received. The form guarantees that the necessary data is accurately recorded.

4.3.5 Chat System

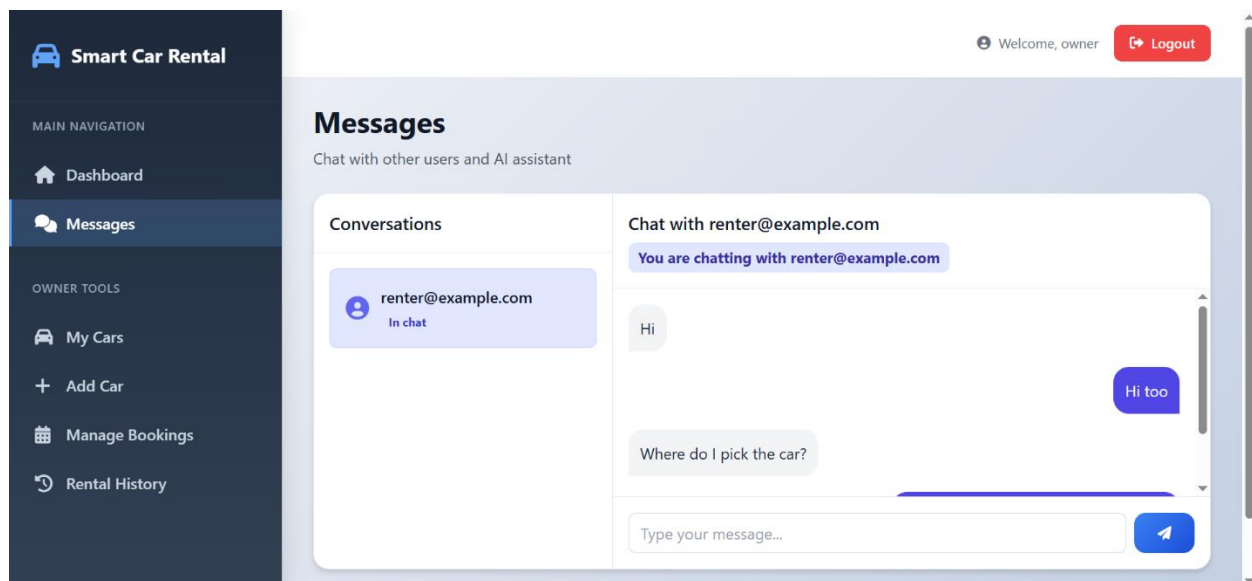


Figure 4.9 Smart Car Rental App's chat interface

The chat interface that facilitates real-time communication between renter and owner is shown in Figure 4.9. Both parties can keep track of conversations with the interface's support for message timestamps and display of message history. This feature facilitates more transparent communication and enables prompt resolution of queries or issues pertaining to reservations.

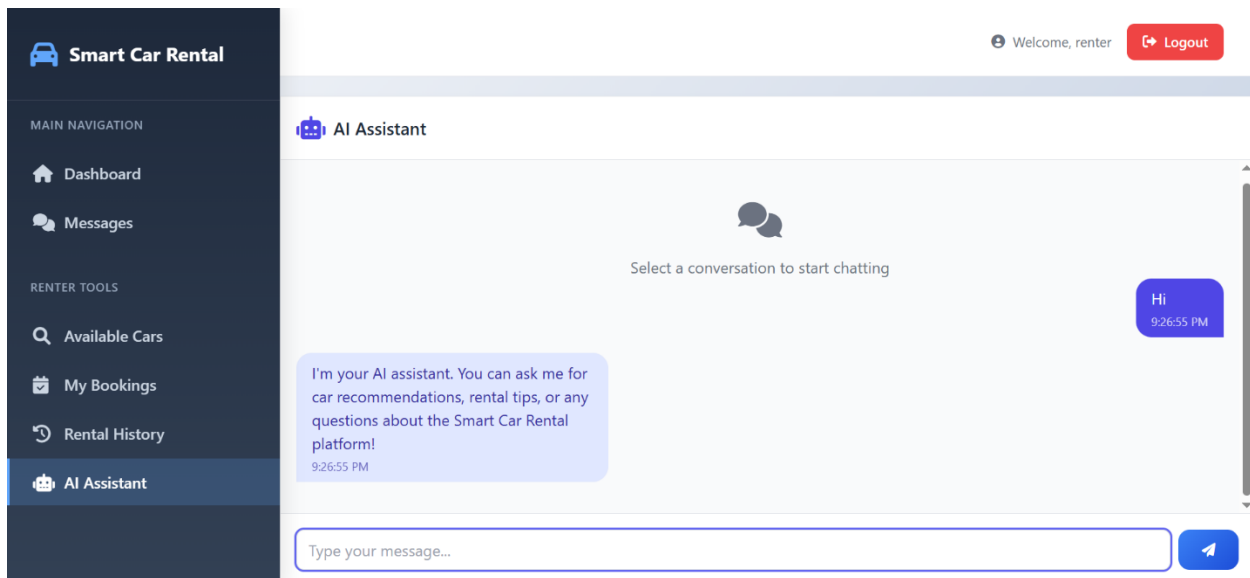


Figure 4.10 Smart Car Rental App's renter AI chatbot interface

Figure 4.10 illustrates the AI chatbot interface intended to help renters bargain for lower car prices or quickly locate pertinent information. It recommends price ranges and booking advice based on pre-established logic. By providing round-the-clock virtual support, particularly when the owner is not available, the chatbot enhances the user experience.

4.3.6 Booking History

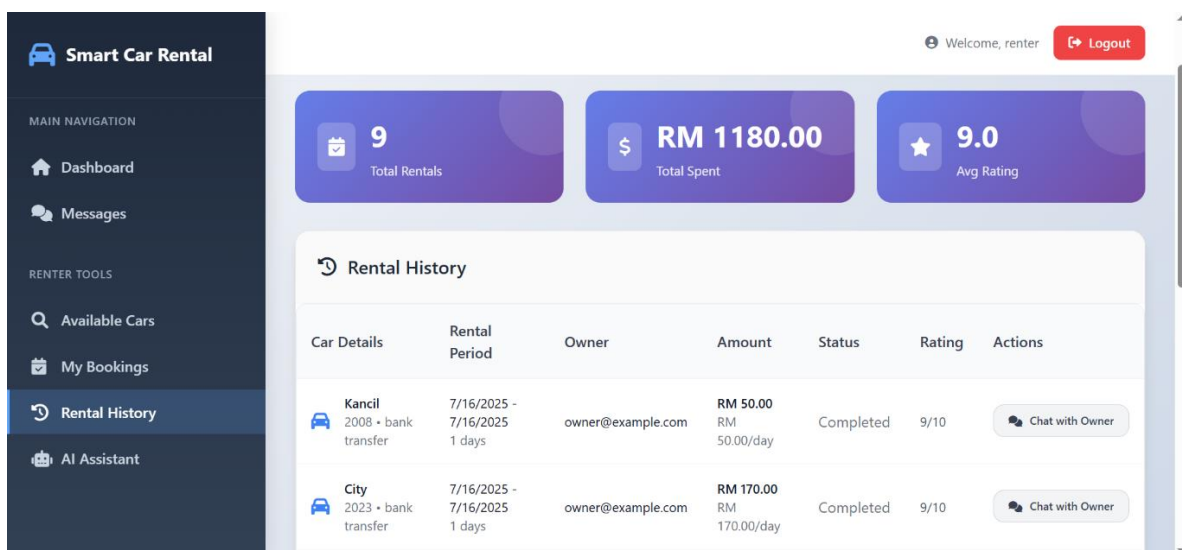


Figure 4.11 Smart Car Rental App's renter booking history page

Figure 4.11 illustrates the renter booking history page, which shows all the user's previous and present reservations. The vehicle's name, reservation date, rental duration, and status are displayed. This feature guarantees transparency in the booking process and assists renters in tracking their rental timeline.

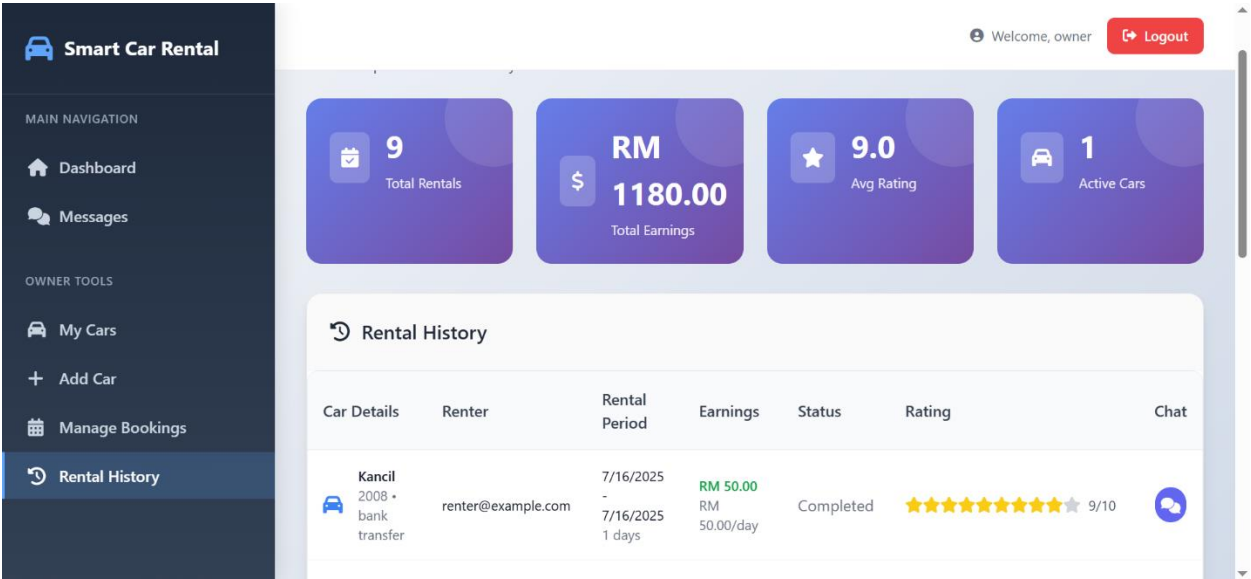
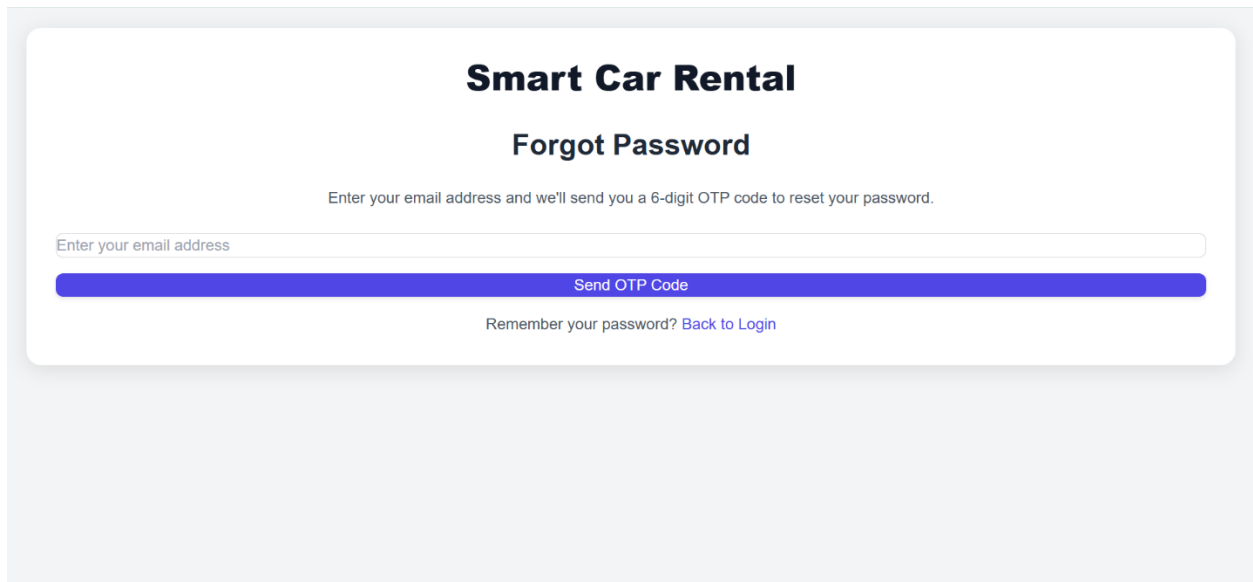


Figure 4.12 Smart Car Rental App’s owner booking history

Figure 4.12 shows the owner’s booking history, which includes all previous and present vehicle reservations across their fleet. Renter information, vehicle details, reservation dates, and status updates are compiled here. Owners can efficiently track pending confirmations and examine historical rental trends with this view.

4.3.7 Forgot Password



The image shows a mobile app interface for the 'Smart Car Rental' app. The title 'Smart Car Rental' is at the top in bold black text, followed by 'Forgot Password' in bold black text. Below this is a subtitle: 'Enter your email address and we'll send you a 6-digit OTP code to reset your password.' There is a text input field with the placeholder 'Enter your email address'. Below the input field is a blue button labeled 'Send OTP Code'. At the bottom, there is a link that says 'Remember your password? [Back to Login](#)'.

Figure 4.13 Smart Car Rental App's forgot password

Figure 4.13 shows Forgot Password interface that allows users to securely recover their accounts by submitting their registered email address. Upon submission, a reset link is sent to their email, guiding them through the password reset process. This feature enhances account security and user experience by ensuring access recovery is easy, especially for users who forget their login credentials during future interactions with the rental system.

4.3.8 Payment

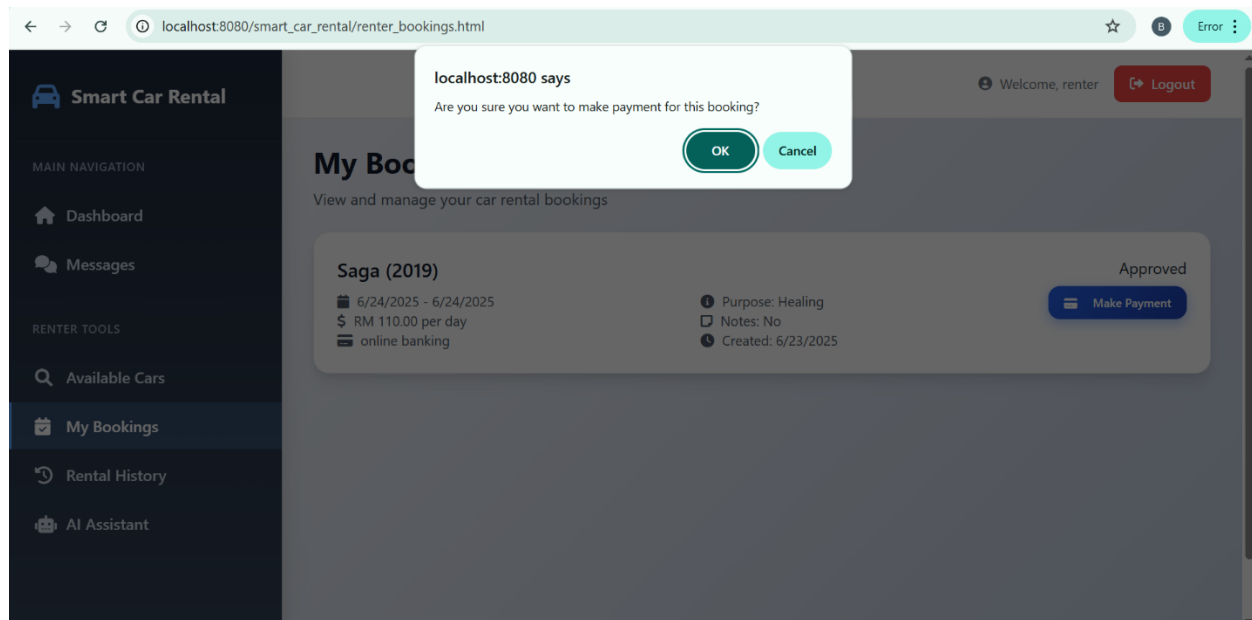


Figure 4.14 Smart Car Rental App's payment

Figure 4.14 shows a payment interface which allows renters to securely pay for their confirmed bookings. Once a booking is accepted by the owner, the renter is prompted to proceed with payment via available methods such as online transfer or credit/debit card (simulated through local integration). Payment confirmation updates the booking status to "Booked," completing the transaction. This interface ensures a seamless and reliable rental process from reservation to payment.

4.3.9 Rating

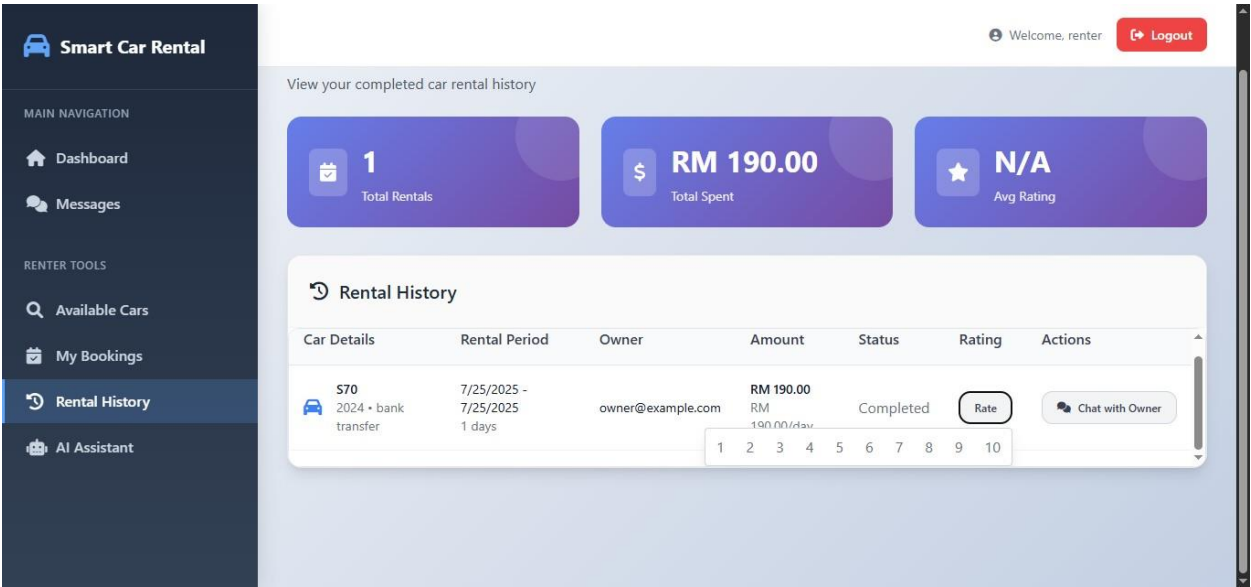


Figure 4.14 Smart Car Rental App’s rating

The Rating Process interface enables renters to provide feedback on their booking experience after a completed rental. Users can rate the car and service on a scale of 1 to 5 stars and leave optional comments. This feedback is stored in the system for owner review and system improvement. The rating feature promotes transparency, trust, and continuous service quality improvement between renters and car owners.

4.4 Backend Implementation

The backend is developed using PHP, handling business logic and database communication. It processes form submissions, updates car and booking statuses, manages user authentication, and returns JSON responses. Key backend scripts include those for user registration, login, booking updates, and chat handling. This layer ensures data is validated and properly stored in the MySQL database.

4.5 Database Schema

The MySQL database consists of tables like users, cars, bookings, payments, and messages. These tables are connected using foreign keys to ensure data integrity. Each car is linked to an owner, and each booking links a car to a renter. This relational design supports efficient data handling and allows smooth tracking of rental history and chat logs.

The database contains core tables such as:

- users (id, name, email, password, role)
- cars (id, owner_id, brand, type, price, status)
- bookings (id, car_id, renter_id, start_date, status)
- messages (chat system)
- payments (linked to confirmed bookings)

Refer to *Appendix B* for the full SQL schema.

4.6 Summary

The implementation of each module on the Smart Car Rental App was thoroughly explained in this chapter. Database integration, backend logic, frontend interfaces, and architecture were all covered. In line with the functional requirements covered in previous chapters, the design approach guarantees scalability, clarity, and role-based access control.

Chapter 5: Testing and Evaluation

5.1 Introduction

The testing techniques used to validate the Smart Car Rental App and make sure its operation meets with the goals covered in Chapter 1 are described in this chapter. Functional testing, black-box testing, and user acceptance testing (UAT) were among the various types of testing that were carried out. Every test was created to evaluate the system's dependability, performance, and usability from the viewpoints of the owner and renter.

5.2 Testing Strategy

To confirm that every system function operates in accordance with the specifications, functional testing was carried out. To test the system's responses without examining the internal code structure, black-box testing was also used. To make sure the system fulfilled practical requirements and offered seamless user experience, a user acceptance test was also carried out with a small sample of users who represented the target market.

5.3 Test Case and Expected Results

Table 5.1 Login Module outlines selected functional test cases that were conducted to verify the core features of the system:

Table 5.1 Login Module

Test Case ID	Description	Input	Expected Result	Actual Result	Status
TC01	Valid login credentials	<u>tzuyu@gmail.com</u> tzuyu123	Redirect to user dashboard	As expected	Passed
TC02	Valid email, incorrect password	<u>tzuyu@gmail.com</u> tzuyu12	Error message: "Invalid credentials"	As expected	Passed
TC03	Empty email and password fields	[blank] [blank]	Validation error messages shown	As expected	Passed

Table 5.2 Registration Module

Test Case ID	Description	Input	Expected Result	Actual Result	Status
TC04	Successful renter registration	Name, Email, Password, Role	Account created and redirected to login page	As Expected	Passed
TC05	Duplicate email registration	Existing Email	Error: "Email already exists"	As Expected	Passed

TC06	Missing required fields	Name only	Prompt: "Please fill in all fields"	As Expected	Passed
------	-------------------------	-----------	-------------------------------------	-------------	--------

Table 5.3 Car Management (Owner)

Test Case ID	Description	Input	Expected Result	Actual Result	Status
TC07	Add a new car with complete details	Brand, Type, Price, etc.	Car added and listed in inventory	As Expected	Passed
TC08	Edit an existing car	Update car type or status	Car information updated	As Expected	Passed
TC09	Delete car from inventory	Car ID	Car deleted from list	As Expected	Passed

Table 5.4 Booking Process (Renter)

Test Case ID	Description	Input	Expected Result	Actual Result	Status
TC10	Search available cars	Type, Brand, Location	Matching cars displayed	As Expected	Passed

TC11	Book selected car	Car ID, Dates	Booking added with status "pending"	As Expected	Passed
TC12	Submit booking without selecting dates	Car ID only	Error: "Please select rental period"	As Expected	Passed

Table 5.5 Payment and Status Update

Test Case ID	Description	Input	Expected Result	Actual Result	Status
TC13	Owner accepts booking	Booking ID	Booking status updated to "pending payment"	As Expected	Passed
TC14	Renter completes payment	Booking ID, Payment method	Status changed to "booked"	As Expected	Passed

Table 5.6 Chat and History

Test Case ID	Description	Input	Expected Result	Actual Result	Status
TC15	Renter sends a message	Message text	Message displayed with timestamp	Not As Expected	Failed
TC16	View renter booking history	User login	List of previous and active bookings displayed	As Expected	Passed
TC17	View owner booking history	Owner login	Booking list with renter details and status shown	As Expected	Passed

5.4 Non-Functional and User Acceptance Testing

Non-functional testing focused on aspects such as usability, responsiveness, and visual design. Usability testing was performed through informal observation and user feedback during UAT. 10 participants, 7 car renters and 3 car owners, were asked to rate their experience with specific areas of the system. The feedback was recorded in conversational format to better reflect real user impressions.

Table 5.7 Car Renter UAT Testing and Rating

Area Tested	Description	User Feedback Summary	Average Rating (1-5)
Ease of Navigation	How easily renters can navigate the platform	Most renters found the interface intuitive and easy to use.	4.5
Booking Process Clarity	Clarity in booking and status flow	Booking steps were clear, though a few wanted more visual guidance.	4.3
Chatbot Helpfulness	Usefulness of the AI chatbot in addressing questions	Users found the chatbot useful, though it lacked depth in some responses.	4.1
Interface Design	Layout and visual appeal of the platform	The design was rated modern and clean; some suggested more contrasts on buttons.	4.4
Overall Satisfaction	General satisfaction with using the system	Users were pleased overall and said they would prefer this over manual rentals.	4.6

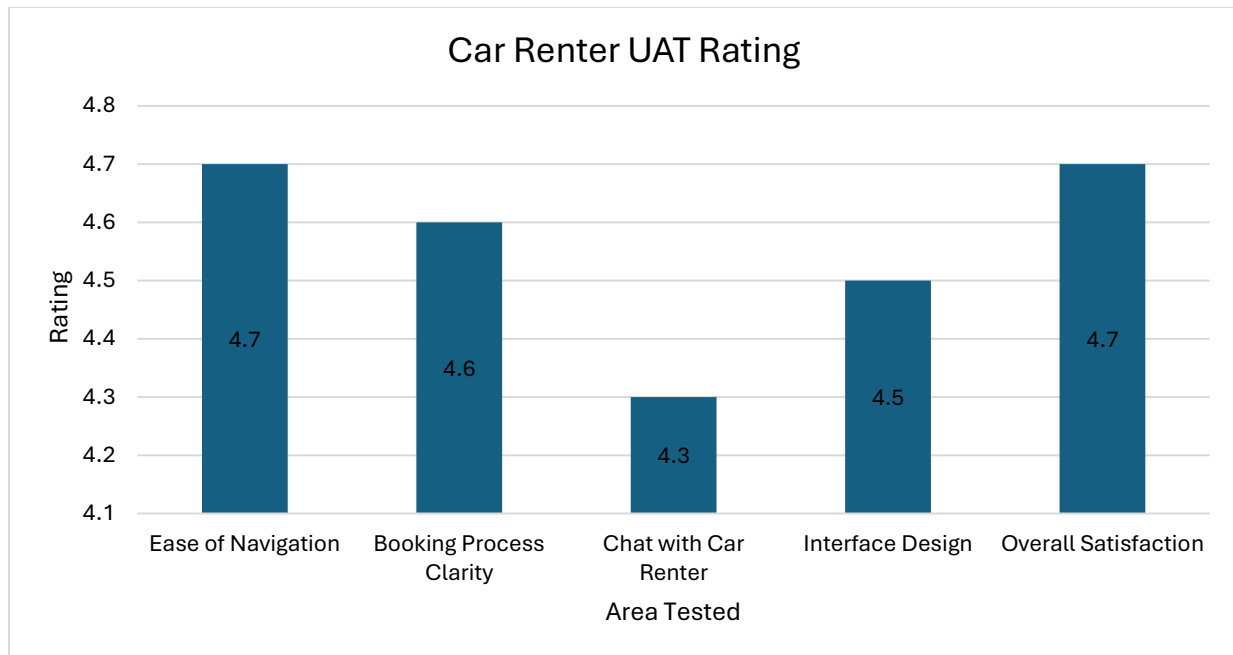


Figure 5.1 Car Renter UAT Rating

Table 5.8 Car Owner UAT Testing and Rating

Area Tested	Description	User Feedback Summary	Average Rating (1-5)
Ease of Navigation	How easily owners could access and manage their dashboards	Owners found car listing management and booking status easy to operate.	4.7
Booking Process Clarity	Ease of understanding renter bookings and status changes	Process was clear; a few requested auto-email confirmations for bookings.	4.6
Chat with Car Renter	Real-time messaging with renters for coordination	Owners liked being able to chat with renters directly for faster confirmation and updates.	4.3

Interface Design	Layout and visual appeal of the platform	Clean and organized dashboard, though some icons could be labeled more clearly.	4.5
Overall Satisfaction	General satisfaction with using the system	Overall satisfaction was high; they liked the status control and centralized access.	4.7

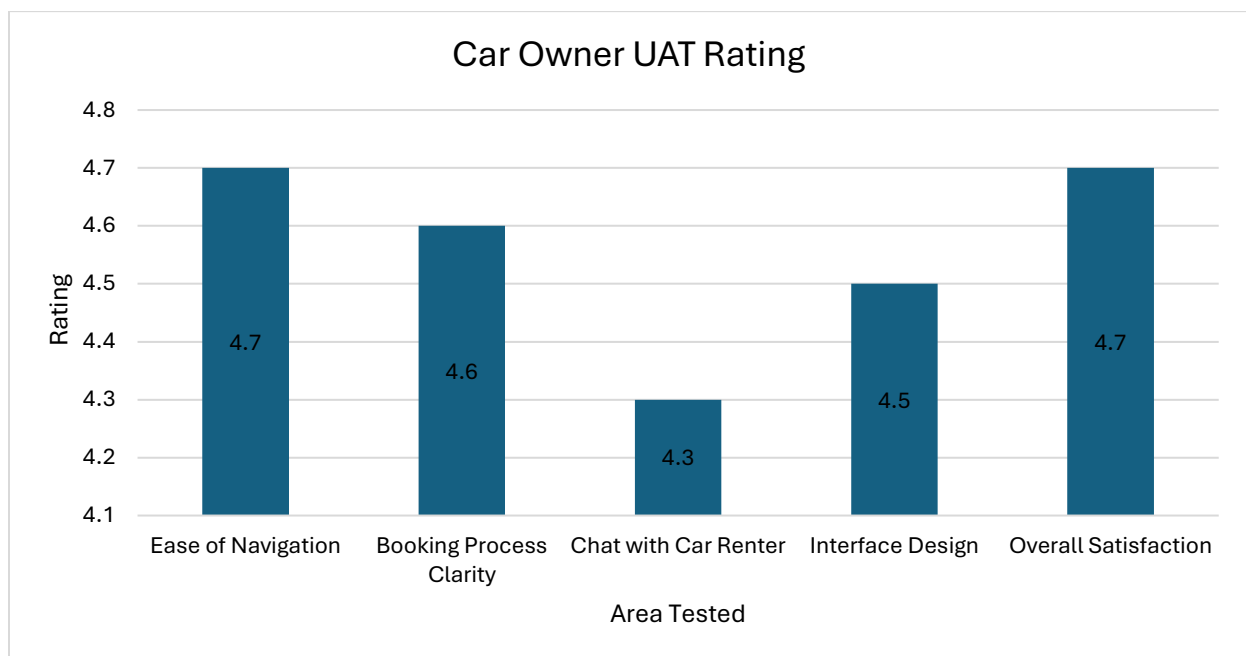


Figure 5.2 Car Owner UAT Rating

5.5 Evaluation

To evaluate the Smart Car Rental App, both functional and usability testing were conducted. Functional testing involved executing specific test cases to ensure each core module, such as login, booking, chatbot, payment, and car status tracking, functioned correctly. All critical test cases passed successfully, confirming the system meets its intended requirements. To assess user

satisfaction, informal usability testing was conducted through short conversations with 10 users. Users were asked to interact with the system and rate key features such as ease of use, clarity of interface, response time, and helpfulness of the chatbot on a scale of 1 to 5.

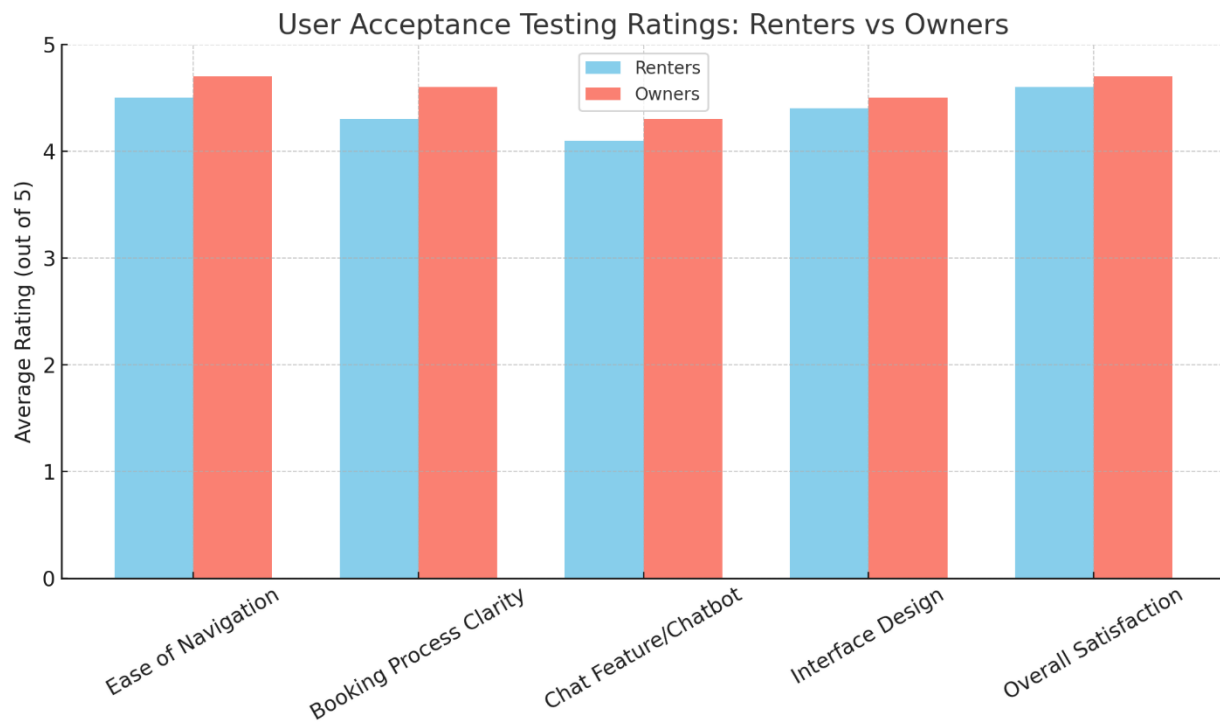


Figure 5.3 UAT Renters vs Owners

These results indicate a generally positive user experience. However, some users suggested minor improvements, such as booking history filters and expanding chatbot capabilities.

In conclusion, the evaluation reveals that the system performs reliably and is well-received by its users. It meets the functional goals and delivers a user-friendly experience, supporting the project's objective of creating an efficient and intelligent car rental platform.

5.6 Conclusion

The testing that was done on the Smart Car Rental App was presented in this chapter. It proved that every major feature worked according to plan, and user comments affirmed the system's effectiveness and usability. The successful testing outcomes confirm that the website is prepared for practical use and meets its original goals.

Chapter 6: Conclusion and Future Works

6.1 Project Summary

The Smart Car Rental App was created as a role-based platform to facilitate easy and effective communication between prospective renters and car owners. Owner and Renter were the two primary user roles that were used during the project. While renters can browse, search, and reserve cars based on their preferences, owners can register, manage their car listings, and handle bookings. The platform's effectiveness and interaction are improved by extra features like chat communication and an AI chatbot.

PHP was used for server-side logic, MySQL was used as the backend database, and HTML, CSS, and JavaScript were used for the front end of the website. Key features supported by the system include chat messaging, booking workflows, car management, registration, login, and transaction simulation. The testing revealed that the system met its initial goals, and all key features operated as expected.

6.2 Objective Achievement

Table 6.1 Objective Achievement shows how the objectives stated in Chapter 1 were achieved:

Table 6.1 Objective Achievement

Objective	Achievement Status	Description
To develop a car rental website that enables users to browse, book, and pay for car rentals online.	Achieved	A fully functional car rental website was successfully developed with features allowing renters to browse available cars, make bookings, and proceed with online payments. The platform supports multi-role access for both owners and renters.
To implement AI features, such as personalized recommendations and chatbot that provides 24/7 automated support, to improve customer experience.	Achieved	An AI-powered chatbot was integrated into the system to assist users with pricing and general inquiries. Additionally, the system provides recommendation features based on user interaction and car preferences to enhance usability and satisfaction.
To evaluate the effectiveness and user satisfaction of the developed platform.	Achieved	The system was tested using various functional test cases. Informal user feedback and usability testing were conducted, which revealed high levels of user satisfaction with the system's interface, performance, and accessibility.

6.3 Challenges and Limitations

During development, there were several difficulties, especially with backend integration and making sure the user experience was seamless. Careful database design and conditional logic were needed to handle chat messages and real-time booking status updates. The lack of actual payment gateway integration was another drawback; for testing purposes, a simulated payment module was used instead.

Implementing some advanced features, like live notifications and a fully intelligent chatbot, was also hampered by time constraints. Although it was only partially optimized, mobile responsiveness could be enhanced for lower screen resolutions.

6.4 Future Enhancements

To further enhance the system, the following improvements are recommended:

- Integration of a secure online payment gateway such as iPay88 or Stripe
- Implementation of real-time push notifications for booking updates and messages
- Deployment of a more advanced AI chatbot with natural language processing
- Optimization for full mobile responsiveness and support for PWA (Progressive Web App)
- Expansion of the system into a mobile app using React Native or Flutter

These features would not only enhance user experience but also make the system scalable for commercial deployment.

6.5 Conclusion

To sum up, the Smart Car Rental App was successful in fulfilling its objective of giving users a digital platform for easy car rental and management. Implementation and testing were done on the booking logic, role-based access control, core system architecture, and additional features like chat and history tracking. The platform is prepared for upcoming updates and practical implementation thanks to its user-centric design and ongoing feedback. This project has been a great way to learn about problem-solving and technical development.

Appendix A

Question for interviewee through face-to-face interview:

1. What challenges have you faced when renting cars online?
2. Do you prefer a mobile app or web-based platform for renting cars? Why?
3. What features are most important to you in a car rental system? (e.g., price, car variety, location, support)
4. Would you find a built-in chat feature helpful to communicate with car owners?
5. How likely are you to use an AI chatbot to ask about prices or find car options?
6. Do you think having a centralized platform that connects renters and car owners would be helpful in your area?
7. What concerns would you have about using such a system (security, trust, usability)?
8. Do you have any suggestions to improve a car rental platform that connects renters and owners directly?

Appendix B

Full SQL schema:

Table	Action	Rows	Type	Collation	Size	Overhead
☐ bookings	Browse Structure Search Insert Empty Drop	3	InnoDB	utf8mb4_general_ci	48.0 KiB	-
☐ cars	Browse Structure Search Insert Empty Drop	3	InnoDB	utf8mb4_general_ci	32.0 KiB	-
☐ chat_messages	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	32.0 KiB	-
☐ conversations	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	96.0 KiB	-
☐ users	Browse Structure Search Insert Empty Drop	3	InnoDB	utf8mb4_general_ci	32.0 KiB	-
5 tables	Sum	9	InnoDB	utf8mb4_general_ci	240.0 KiB	0 B

References

- Auto Rental News. (2023). *The future of car rentals: Big data and game-changing AI inspection technology*. Retrieved from <https://www.autorentalnews.com/10206598/the-future-of-car-rentals-big-data-and-game-changing-ai-inspection-technology>
- Deshpande, P., Shiwankar, D., Patil, N., Pansare, A., & Shinde, S. (2023). Location based recommendation system for car reservation using machine learning. *International Journal of Scientific Research in Engineering and Management (IJSREM)*, 7(6). <https://doi.org/10.55041/IJSREM23027>
- Hasan, H. (2021). *Car sharing in Malaysia with SOCAR: How to & what to look out for*. Medium. Retrieved from <https://helmihasan.medium.com/car-sharing-in-malaysia-with-socar-how-to-what-to-look-out-for-3a98289a7bc>
- Ikram Transports. (n.d.). *Ikram Transports*. Retrieved from <https://ikramtransports.com/en/>
- Interaction Design Foundation - IxDF. (2016, June 2). What is User Interface (UI) Design? Interaction Design Foundation - IxDF. <https://www.interaction-design.org/literature/topics/ui-design>
- Kayak. (n.d.). *Car rentals*. Retrieved from <https://www.kayak.com/cars>
- Klezcar. (n.d.). *Klezcar*. Retrieved from <https://www.klezcar.com/>
- Renttix. (2024). *Car rental price recommendation software*. Retrieved from https://blog.renttix.com/en-us/car-rental-price-recommendation-software/?utm_source=chatgpt.com
- SOCAR. (n.d.). *SOCAR consumer*. Retrieved from <https://socar.my/consumer>
- Yenra. (2024). *AI tech: Car rental systems: 10 Ways AI is Improving Car Rental Systems*. Retrieved from <https://yenra.com/ai-tech/car-rental-systems/>