A Major Project Final Report on

Rented Wheels

Submitted in Partial Fulfillment of the Requirements for the Degree of **Bachelor in Software Engineering**
Under Pokhara University

Submitted by:

Aakriti Thapa, 180701

Binamra Acharya, 181707

Prabesh Pokharel, 181727

Sunandan Ghimire, 181745

Under the supervision of
Mr. Birendra Bista

Date:
8th July, 2023

Department of Software Engineering
NEPAL COLLEGE OF
INFORMATION TECHNOLOGY

Balkumari, Lalitpur, Nepal.

Acknowledgement

We would like to express our sincere gratitude and appreciation to all those who have contributed to the successful completion of our project **'Rented Wheels'.**

First and foremost, we would like to extend our heartfelt thanks to our project Supervisor and Head of Department of Software Engineering **Mr. Birendra Bista** for his invaluable guidance, support, and expertise throughout the project. His continuous encouragement, insightful feedback, and patient mentorship have been instrumental in shaping this project and enhancing our understanding of the subject matter.

We would also like to thank our Project head **Dr. Roshan Chitrakar** for their valuable inputs, encouragement, and provision of necessary resources. His extensive knowledge and commitment to education have played a vital role in broadening our perspective and enriching the project.

We are grateful to each member of our group for our dedication, hard work, and collaboration. We as a team member brought unique skills, ideas, and contributions that greatly enhanced the project. The synergy and cooperation among us have been the key to our success.

To all those mentioned above and to anyone else who has contributed to this project in any way, we extend our heartfelt thanks. Your support and involvement have been invaluable, and we are truly grateful for the opportunity to work on this project together under such guidance and collaboration.

Abstract

As technology has advanced dramatically in recent years, significant improvements to Nepal's automobile rental systems are required. Currently, existing rental systems are manual, and some lack suitable software to manage the renting process. The purpose of our solution is to automate the existing manual system with the help of full-fledged computerized system. We will provide a user-friendly web application named Rented Wheels with an interface for renting and reserving vehicles, as well as a feedback system to assure customer satisfaction. With Rented Wheels, vehicles owners will be able to advertise their own vehicles for rent and manage bookings, allowing them to monetize their idle assets. This system will provide recommendation of the vehicles to the user based on the features and functionality of the vehicles. Furthermore, this system also includes record-keeping in order to monitor and analyze sales data in order to find patterns and chances for improvement. This will allow platform administrators to optimize the platform's features, functionality, and user experience in order to improve the platform's value proposition, attract more users, and create more income.

Rented Wheels provides an efficient and convenient way for customers to rent vehicles, while also providing rental companies with a powerful tool for managing their fleet and operations.

Keywords: *fledged, Rented Wheels, Feedback System, proposition, fleet*

Table of Contents

List of Figures

List of Tables

1. Introduction

Rented Wheels is the proposed web-based application that is designed to enable users to rent or add vehicles for rental purposes. The platform allows users to create accounts, search for available rental vehicles, and book them for a specified duration. Additionally, users can also register their vehicles on the platform, set rental prices, and manage bookings. The application provides a seamless booking process, secure payment options, and a feedback system to ensure customer satisfaction. The web application is user-friendly and responsive.

Incorporating features such as service ratings, vehicle filtering, availability checking, as well as personalized recommendations based on previous usage, will empower users to make more informed decisions. Additionally, the vehicle rental service platform's provides users with the opportunity to generate income by allowing them to list their vehicles for rent. The wide range of options from which customers can choose will undoubtedly help people to organize their travels freely and eliminate one of the most taxing aspects of planning, thereby speeding the process of effectively enjoying a trip .Overall, the Rented Wheels web application is an innovative solution that simplifies the rental process and provides a convenient and cost-effective option for vehicle rental needs.

We will use the Java framework Spring Boot for backend logic development, the JavaScript framework React JS for the frontend, and MY SQL for data storage. For the AI component, we will be using Python and its required algorithm to manage the recommendation system in our application.

Hence, Rented Wheels is an innovative and convenient solution that streamlines the vehicle rental process for customers and rental companies alike. Its user-friendly interface, advanced features, and customizable nature make it a powerful tool for the modern rental industry.

2. Problem Statement

The manual system used by an organization running a vehicle rental service makes it slow and difficult to provide monthly or annual reports due to a large number of clients. Additionally, the lack of an online booking system and record-keeping makes it hard to track rental car information. While there are many vehicle rental platforms available, only a few are properly managed and allow users to choose the vehicle of their choice transparently. Many of these platforms do not offer vehicle variations, which can make the renting process more tedious. Many people may need to rent a car for various reasons such as travel, temporary replacement of their car, or business purposes. However, existing vehicle rental services can be expensive, and finding a suitable rental vehicle can be time-consuming and challenging.

Some platforms require users to provide extensive information and wait for a quote, making it impossible to rent a vehicle in an emergency. To ensure transparency and help potential customers make informed decisions, users should be allowed to rate and review the service.

Furthermore, many car owners may have idle cars that they do not use regularly, and these cars could be a potential source of income if they could be rented out to others. However, the process of advertising, booking, and managing such a service can be complex and time-consuming.

3. Project Objectives

We aim to address the issues outlined above by introducing a new platform called 'Rented Wheels'. The platform will provide easy access to various types of vehicles suitable for specific journeys at particular destinations. Our goal is to establish an efficient system that facilitates a seamless connection between vehicle owners and renters. Rented wheels aims to provide a user-friendly platform with an interface for renting and reserving vehicles, as well as a feedback system to assure customer satisfaction. Rented Wheels enables vehicle owners to advertise their vehicles for rent and manage bookings, allowing them to generate income from their unused assets.

In addition, our platform offers vehicle recommendations to users based on the features and performance of the vehicles. This helps users make informed decisions when selecting a vehicle, based on the positive experiences of other users and the high performance of the vehicles.

The main objectives of this project include:

1. To develop a web based system which is user-friendly that will help users to rent and manage vehicles at a lower cost and helps customer to recommend a vehicle with good functionality and best user experience.
2. To design a system that assists users in managing their vehicles, which includes the features of a online payment processing system and to provide a scalable platform to protect and manage customer data from unauthorized access and can accommodate increasing demand and expand to new markets.

4. Significance of the Study

The study of 'Rented Wheels' has practical implications for everyday life as it addresses the difficulties of traveling from one location to another. Firstly, the proposed system helps to understand the current market trends and customer demands, and help to design a system that meets the needs of the target audience. Similarly, the study can help identify the strengths and weaknesses of existing vehicle rental systems, which can guide the development of a more effective and efficient system. Additionally, it can also be useful to identify potential hazards and difficulties linked to the other system and create mitigation plans for them.

Rented Wheels offers several other significant benefits, including convenience, income generation, improved accessibility, better customer experience, data analysis, and reduced environmental impact. The platform enables users to rent various types of vehicles at affordable rates and provides an opportunity for car owners to generate income and enhance the overall rental experience. Additionally, the platform's record-keeping allows administrators to monitor usage and performance, analyze sales data, and identify opportunities for improvement.

Overall, the Rented Wheels project and its system intends a significant benefit and eliminates the hassle of manual procedure for renting of vehicles and provides a successful platform for placing your own vehicles for renting purpose. The system also provides recommendation of the best vehicle for the customer based on the user experience, feedback and vehicles functionality

5. Scope and limitations

There are numerous successful online rental systems in Nepal's business industry, but building a website can present challenges and be impacted by external factors. Here are the potential opportunities and limitations for Rented Wheels:

Scope:

1. The vehicle rental system web application provides a comprehensive solution for individuals who require short-term or long-term transportation.
2. The platform offers an easy-to-use interface for renting a wide range of vehicles according to specific requirements.
3. The integration of a Payment Gateway.
4. Features such as service ratings and personalized recommendations improve the rental experience.
5. The implementation of a vehicle filtering.
6. The feedback system ensures customer satisfaction.
7. Record-keeping and data analysis capabilities allow administrators to monitor usage and performance, analyze sales data, and identify opportunities for improvement.

Limitations:

1. It is not possible to monitor the movement of vehicles.
2. Customers need to log in before they can rent a vehicle.
3. The system does not allow renting the same car on the same day.
4. Only phone calls and in-person communication are available as methods of communication.
5. You have to provide information every time you wish to make a vehicle reservation.
6. We do not have a comprehensive verification procedure for confirming the vehicles added by the user.

6. Literature Review

This section seeks to provide readers with both a theoretical foundation for the project and a clearer understanding of its purpose. It involves conducting a complete review of existing literature on a specific research topic in order to identify knowledge gaps and lay the groundwork for the proposed research.

6.1 Existing Systems

Some of the similar existing applications to Rented Wheels are mentioned below:

1. **Autocios [1]**

   It is one of Nepal's online automobile rental platforms. It's most likely the same as Rented wheels. It also delivers vehicles based on the user's preferences and the destination. In comparison to Rented Wheels, this platform is doesn't let the user to rate and review the vehicle and the system is confined to four-wheelers, whereas our platform offers two-wheeler services also. Users in Autocios are not notified about the availability of vehicles prior to the renting procedure, forcing them to wait until the procedure is complete, which takes time, whereas in Rented wheels, we show the availability of vehicles prior to booking them, making it more convenient and time saving.
2. **Bikemandu [2]**

   It is also one of Nepal's online vehicle rental platforms. Bikemandu offers a variety of bike services to customers at reasonable prices. This platform also offers 4-wheeler rental, however the selections are limited and the options are few. In Bikemandu there are no features like online payment and vehicle availability, so users aren't notified that the selected vehicle are available for rent, whereas our renting system displays vehicle availability before the booking procedure, and users can pay via online platform too.

6.2 Comparison with existing system

Our Rented Wheels has a number of advantages over the current process. The time-consuming manual processes and paper work are eliminated with our technology. This system offers two- and four-wheeled vehicles. Our solution ensures data security in an open environment, as well as online vehicle availability. After entering into the system, anyone can see the available vehicles for rent. This technology ensures payment security while also saving users time. Our platform also allows users to rent out their personal vehicles, which becomes a passive source of revenue for them. The current manual procedure will be completely replaced, and the current manual approach will be rendered inaccessible. Furthermore, because the records are kept in a database, we may update, delete, and add data as needed.

7. Project Methodology

For the successful plan, design, implementation, test and deployment of our project on the web we have decided to use following methodologies and technologies:

7.1 Software Development Life Cycle

The Software Development Life Cycle (SDLC) is a structure process that enables the production of high-quality, low-cost software, in the shortest period of time. The goal of the SDLC is to produce superior software that meets and exceeds all customer expectations and demand. We have decided to follow the principles of iterative development model for the overall tasks and activities that we may have to carry out during the development of this system.

7.1.1 Why Iterative and Incremental Model?

Iterative development [3] is a method or process for creating software in small and frequent increments. Each increment refers to the incorporation of one or more features or functionality into the system under development. The system to evolve becomes a more polished and comprehensive version of itself with each increment. Rather than developing an entire application or software in one go, which can be extremely difficult as the project size grows and requirements change, following the principles of helps to cope with changing system requirements as new requirements can be dealt with in new increments of development. Iterative development will also enable us to envisage the entire system in terms of features that will be added as the development process progresses.

Each increment of development process has following phases:

7.1.2 Analysis Phase

Analysis is performed prior to anything else during in an increment. Analysis phase consists of activities such as identifying and analyzing requirements, studying the application domain, figuring out the scope of the increment and so on. If changes are being made, the effects of these changes on the overall system are also studied.

7.1.3 Design Phase

After a thorough analysis of the requirements is done, the overall architecture of the increment is designed based on the analysis. The architecture shows the relationships between the components that make up the increment as well as the overall system and flow of data and information in the system.

7.1.4 Coding or Implementation Phase

In this phase, the planned increment is implemented by writing code using programming languages, libraries and all the tools available. The previous two phases have significant role in aiding in the implementation process.

7.1.5 Testing Phase

After a planned increment has been implemented, it is time to test whether the functional and non-functional requirements of the increment have been fulfilled. Various tools can be used for testing an increment before it is incorporated with the rest of the application or deployed. Testing may come in various forms and types.

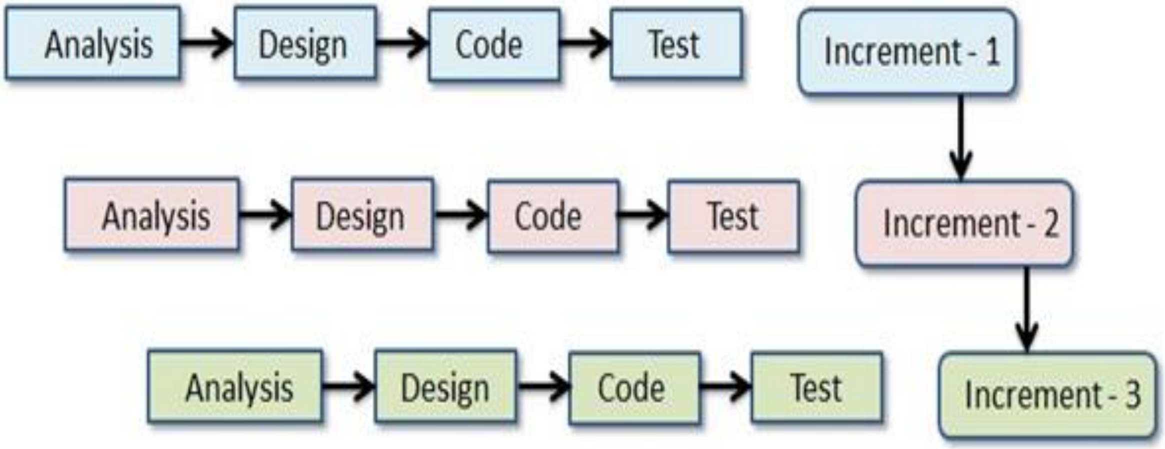


Figure 1: Incremental Development Model

7.2 Programming Language and Other Tools

7.2.1 Programming Language

For our project, we will be using JavaScript library i.e. React JS for the frontend development and for the backend development of our application we will be using the Java Spring Boot framework. For the management of data and its models we will be using MYSQL relational database. And for the AI part of our project we will be using Python and its necessary libraries. As a result, the combination of these technology stacks is strong enough to implement all of the project's stated objectives.

7.2.1.1 React JS

React JS [4] is a framework or toolkit for creating interactive user interfaces using highly reusable and effective components. A React JS-created UI is component-based, which means it is made up of numerous entities called components, each with its own state, which is a set of properties that describes the component's present appearance or behavior, that are linked together to form bigger elements of the UI.

7.2.1.2 Java Spring Boot

Java Spring Boot [5] is an open-source framework for building Java-based web applications. Spring Boot uses a convention-over-configuration approach, which means that developers can quickly get started with minimal setup and configuration. It also provides a range of modules and libraries for common tasks such as security, data access, and web services, making it easy to add additional functionality as needed. Overall, Spring Boot can help speed up development time, simplify the development process, and improve the scalability and reliability of our project.

7.2.1.3 MYSQL Database

MYSQL [6] is a relational database management system (RDBMS) which is suitable for Rented Wheels as multiple relationships between data models have to be addressed as the application is developed. We have chosen to work on MYSQL because it delivers a very fast, multithreaded, multi-user, and robust SQL (Structured Query Language) database servers and can integrate with other software platforms and programming languages.

7.2.1.4 Python

Python's [7] design philosophy emphasizes code readability with its notable use of significant whitespace. It is a powerful programming language for artificial intelligence (AI) projects because it has a wide range of libraries and frameworks which provide a several functionalities to handle different AI tasks, such as machine learning, natural language processing, computer vision, and deep learning. Python is flexible and is platform independent which makes us easier to develop our application and helps us to solve complex algorithm regarding AI and ML.

7.2.2 Tools Used

We are expecting to use following tools throughout the development process of our application:

Tools	**Application**
Visual Studio Code	It is a text editor, so it will be used to write code for the entire application.
Google Chrome	It is a web browser and will be used to debug and preview the client side of the application as it is being developed.
Edraw Max	Edraw Max will be used to create models or diagrams, such as use case diagrams, activity diagrams, class diagram, and sequence diagram and so on.
GitHub	It is a code hosting platform that allows us to manage our source code, track issues, and collaborate with team members.
Microsoft Word	It will be used for the documentation and final report of our project
MY SQL Workbench	It is a visual tool for database design, development, and administration of MySQL databases.

Table 1: Tools Used

7.3 Recommendation System

For our project Rented Wheels, we have used Content-based filtering algorithm [9] for recommendation purpose in our system. This algorithm suggests items to users based on the similarity of their attributes or content. It focuses on the characteristics or properties of the items themselves rather than considering user behavior or preferences. The underlying idea is that if a user has shown interest in certain items in the past, they are likely to be interested in similar items in the future. The algorithm leverages vehicle attributes and user preferences to deliver personalized vehicle recommendations to users.

Here's an overview of recommendation algorithm work in our system.

1. Data Collection: The algorithm collects comprehensive data about available vehicles in the inventory. We collected item data, including vehicle type, brand, model, year, and other features and descriptions, from reliable sources or our own. The item descriptions served as the primary content for similarity analysis.
2. User Profiling: User profiles are created based on preferences and historical data. Information such as preferred vehicle type, brand preferences, rating and review of vehicles categories are considered. These preferences were used as input for the recommendation process.
3. Similarity Calculation: After that, the algorithm calculates the similarity between vehicle features and user profiles. We applied cosine similarity, a common similarity measure, to quantify the similarity between item feature vectors. Cosine similarity ranges from 0 to 1, where a value of 1 indicates perfect similarity.
4. Recommendation Generation: Based on similarity scores, now the algorithm ranks vehicles to generate personalized recommendations. Vehicles that closely match user preferences, such as type, brand, and other features are suggested. Additional filters, like availability and pricing, may be applied for further refinement.
5. Personalization and Feedback Loop: User feedback is actively collected to improve the recommendation process. Feedback helps fine-tune the algorithm, making it more accurate and personalized over time.

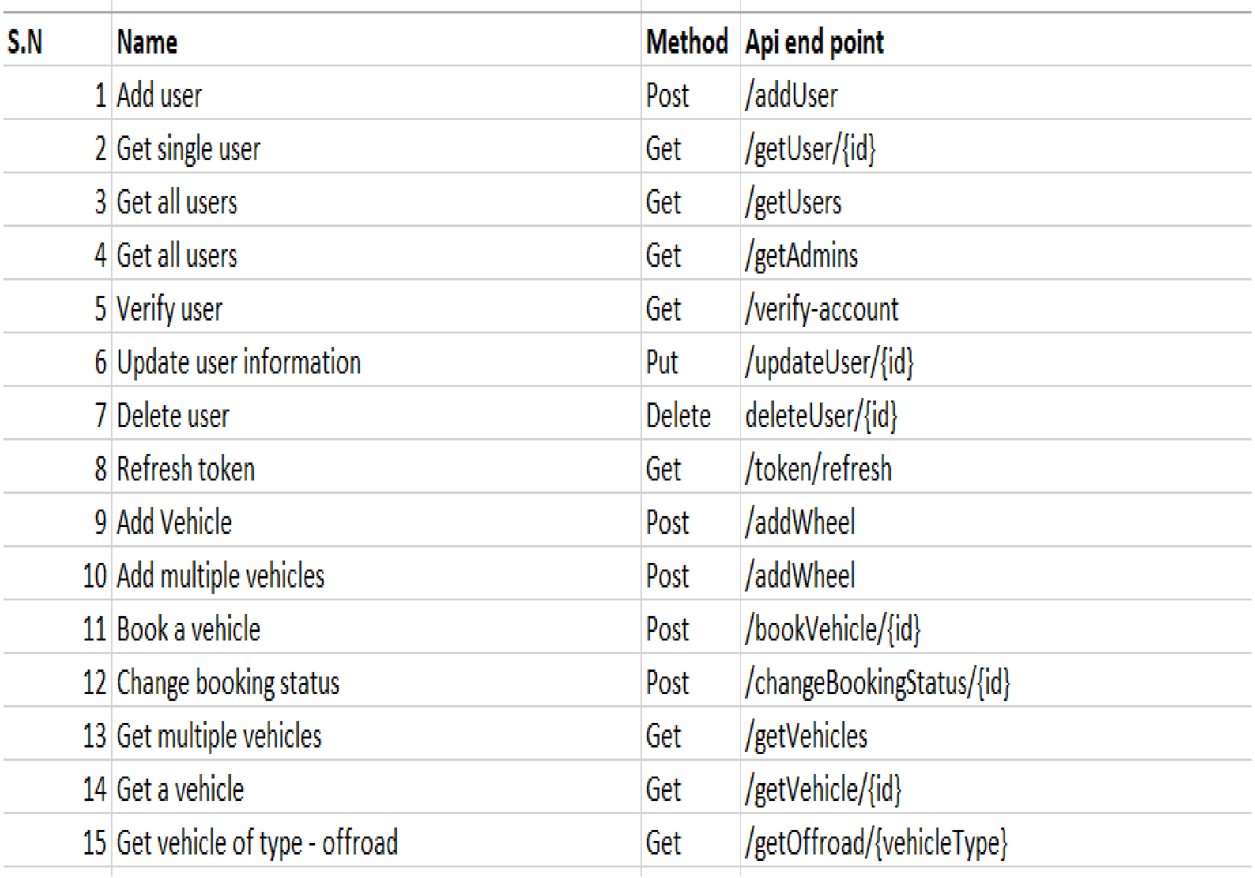
This is how the algorithm is designed for recommending vehicles to the customer in our system. The above overview shows the process for designing the required recommendation system for our project.

In conclusion, Content-based filtering is independent of user history, allowing recommendations for new or anonymous users. It provides transparent recommendations by utilizing explicit item attributes, fostering user trust and engagement. This approach effectively handles role or specialized items by considering specific attributes, matching unique user preferences. Content-based filtering also aids in mitigating the cold-start problem by generating meaningful recommendations without extensive historical data. Overall, content-based filtering algorithms provide valuable benefits in terms of personalization, transparency, and addressing various recommendation scenarios. However, it also has limitations, such as the inability to capture serendipitous recommendations or discover items outside the user's established preferences.

Moreover, further developments such as hybrid approaches can be adopted by combining content-based filtering with other recommendation techniques like collaborative filtering or hybrid systems, taking advantage of the strengths of different algorithms. Contextual recommendations can be implemented by considering factors such as time, location, or user context to provide more personalized and relevant suggestions. Deep learning models, such as convolutional or recurrent neural networks, can be employed to learn complex patterns and enhance recommendation accuracy. By pursuing these avenues, the system can be further developed to provide more accurate, personalized, and satisfactory recommendations to users.

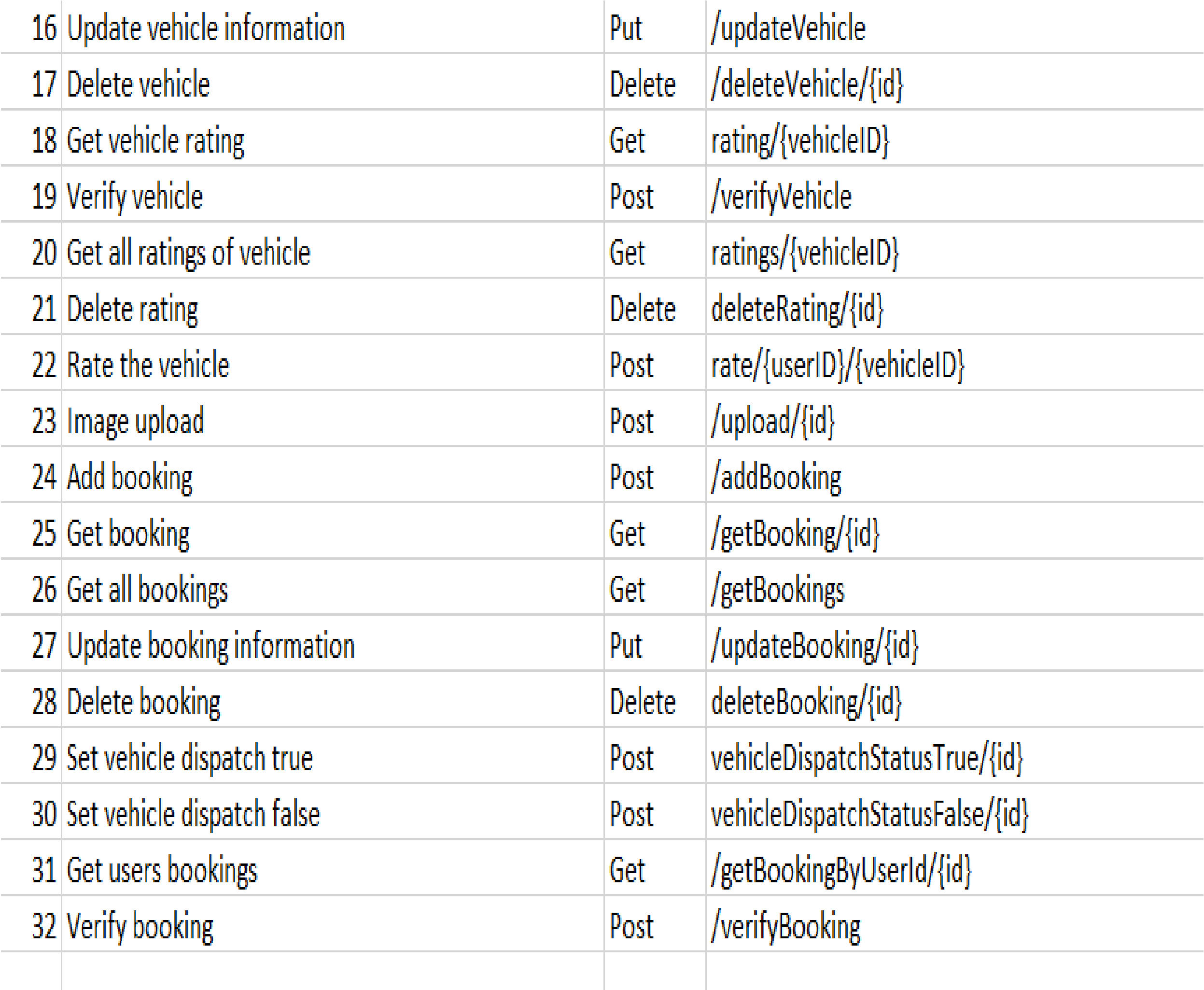
7.4 API (Application Programming Interface) endpoints.

The server-side of Rented Wheels has an API build using the Spring Boot framework on Java run-time environment.



S.N	Name	Method	Api end point
1	Add user	Post	/addUser
2	Get single user	Get	/getUser/{id}
3	Get all users	Get	/getUsers
4	Get all users	Get	/getAdmins
5	Verify user	Get	/verify-account
6	Update user information	Put	/updateUser/{id}
7	Delete user	Delete	deleteUser/{id}
8	Refresh token	Get	/token/refresh
9	Add Vehicle	Post	/addWheel
10	Add multiple vehicles	Post	/addWheel
11	Book a vehicle	Post	/bookVehicle/{id}
12	Change booking status	Post	/changeBookingStatus/{id}
13	Get multiple vehicles	Get	/getVehicles
14	Get a vehicle	Get	/getVehicle/{id}
15	Get vehicle of type - offroad	Get	/getOffroad/{vehicleType}

Figure 2: List of API endpoints 1



16	Update vehicle information	Put	/updateVehicle
17	Delete vehicle	Delete	/deleteVehicle/{id}
18	Get vehicle rating	Get	rating/{vehicleID}
19	Verify vehicle	Post	/verifyVehicle
20	Get all ratings of vehicle	Get	ratings/{vehicleID}
21	Delete rating	Delete	deleteRating/{id}
22	Rate the vehicle	Post	rate/{userID}/{vehicleID}
23	Image upload	Post	/upload/{id}
24	Add booking	Post	/addBooking
25	Get booking	Get	/getBooking/{id}
26	Get all bookings	Get	/getBookings
27	Update booking information	Put	/updateBooking/{id}
28	Delete booking	Delete	deleteBooking/{id}
29	Set vehicle dispatch true	Post	vehicleDispatchStatusTrue/{id}
30	Set vehicle dispatch false	Post	vehicleDispatchStatusFalse/{id}
31	Get users bookings	Get	/getBookingByUserId/{id}
32	Verify booking	Post	/verifyBooking

Figure 3: List of API endpoints 2

8. Results and Discussion

Rented Wheels project yielded positive results in the implementation of its key functionalities. User registration and login functionality were successfully incorporated, allowing users to create accounts and securely log in. The system efficiently managed the inventory of available vehicles by categorizing them based on type and providing detailed information about each listing. Users were able to search for vehicles, add their vehicles, book reservations, and receive confirmation. Secure payment processing was integrated, enabling users to complete transactions online and generate invoices and receipts. The system also incorporated a review and rating system, allowing users to provide feedback on their rented vehicles and overall experience. An admin dashboard was developed to provide comprehensive management of user accounts, vehicle listings, bookings, and payments. Security and privacy measures were implemented to protect user data and ensure secure access. Moreover, an AI based recommendation system has also been introduced which helps other user to choose the vehicles with good review and user satisfaction.

In terms of discussion, the user experience and interface were designed to be intuitive and user-friendly, incorporating feedback from users during the testing phase. Scalability and performance were considered, with optimization measures taken to enhance response times and ensure smooth operation under varying load conditions.

Overall, this project demonstrated successful implementation of core functionalities, providing a user-friendly experience for renting vehicles, efficient management of inventory and bookings, secure payment processing, and an administrative dashboard for system management. Future directions involve enhancing user experience, ensuring scalability, exploring third-party integrations, implementing effective marketing strategies, and adhering to legal and regulatory requirements.

9. Project Details and Diagrams

9.1 Use Case Diagram

A use case diagram is a dynamic or behavior diagram in UML. Use case diagrams model the functionality of a system using actors and use cases. Use cases are a set of actions that the system needs to perform. Use case is a list of steps, typically defining interaction between a role and a system, to achieve a goal. The graphical representation of what our system must actually do is represented as in use case below:

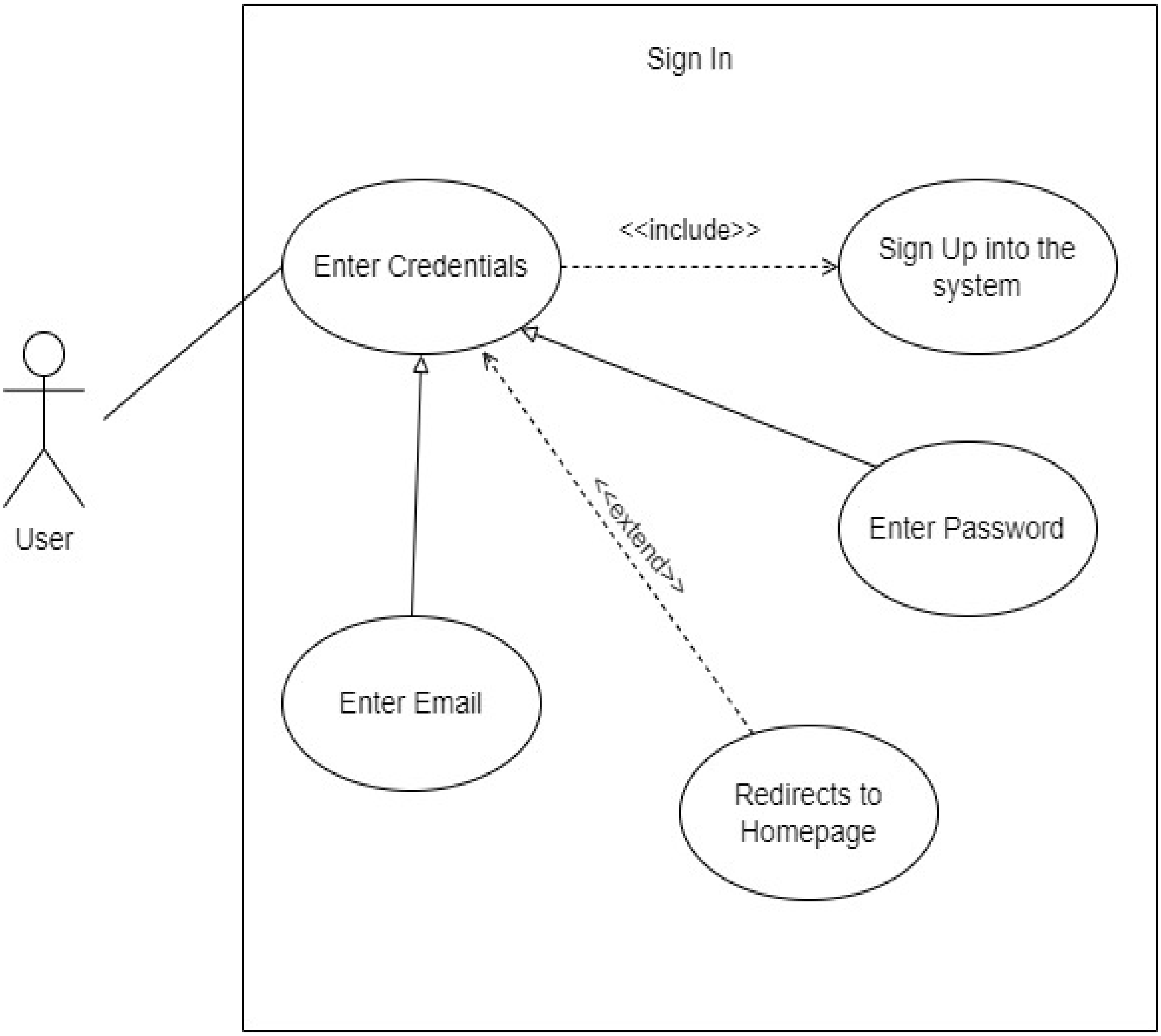


Figure 4: Sign In Use Case Diagram

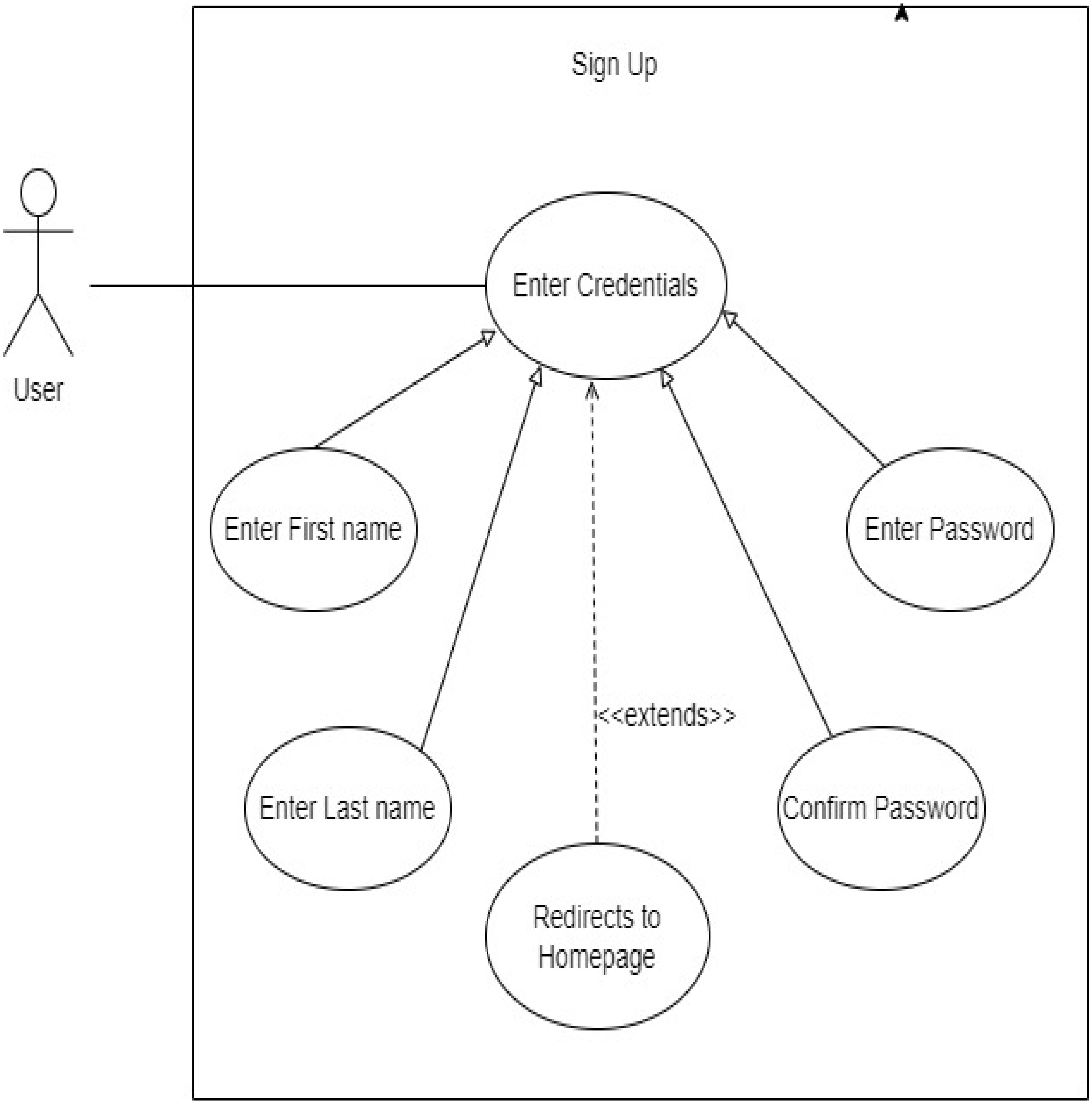


Figure 5: Signup Use Case Diagram

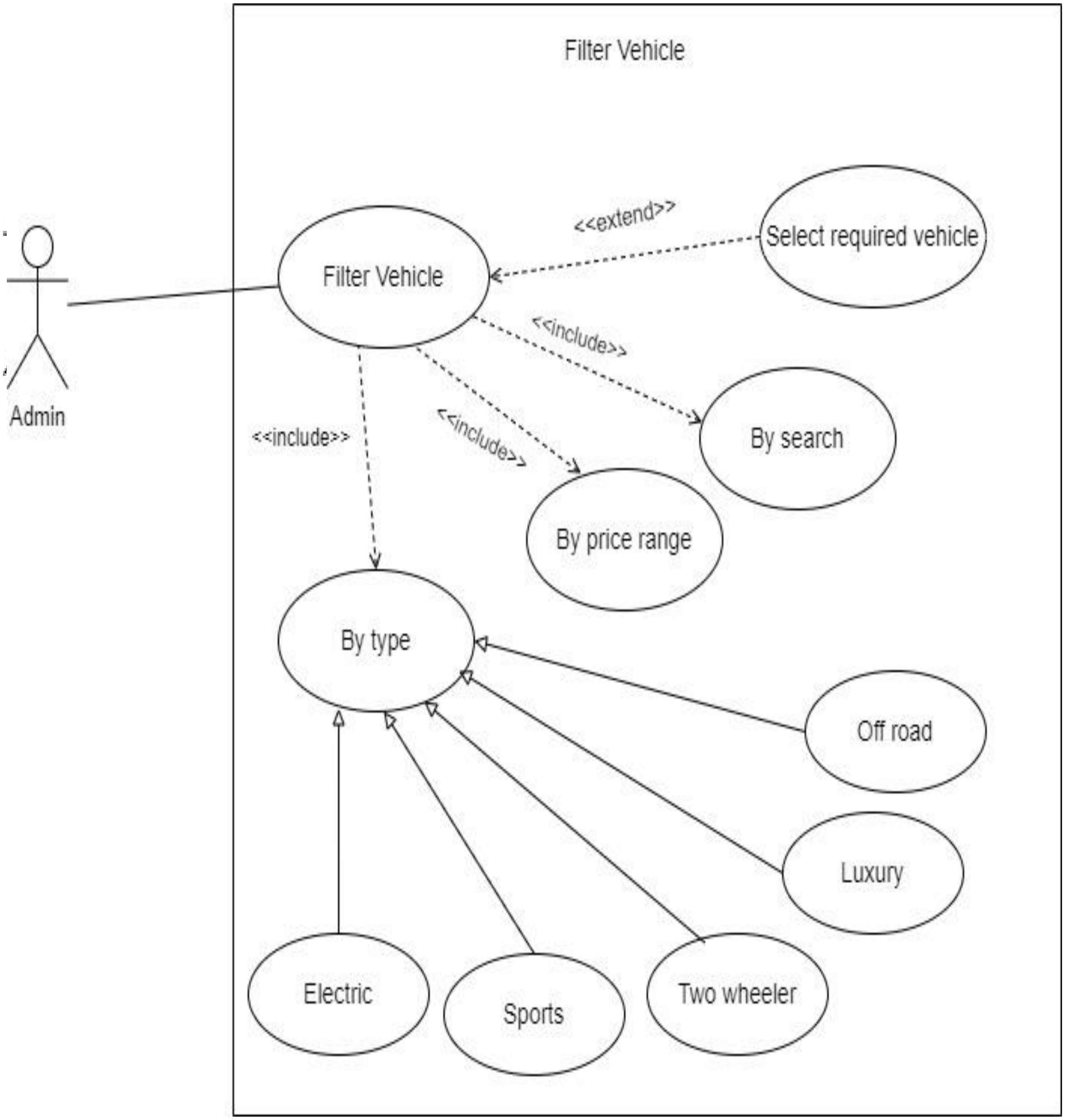


Figure 6: Filter Vehicle Use Case Diagram

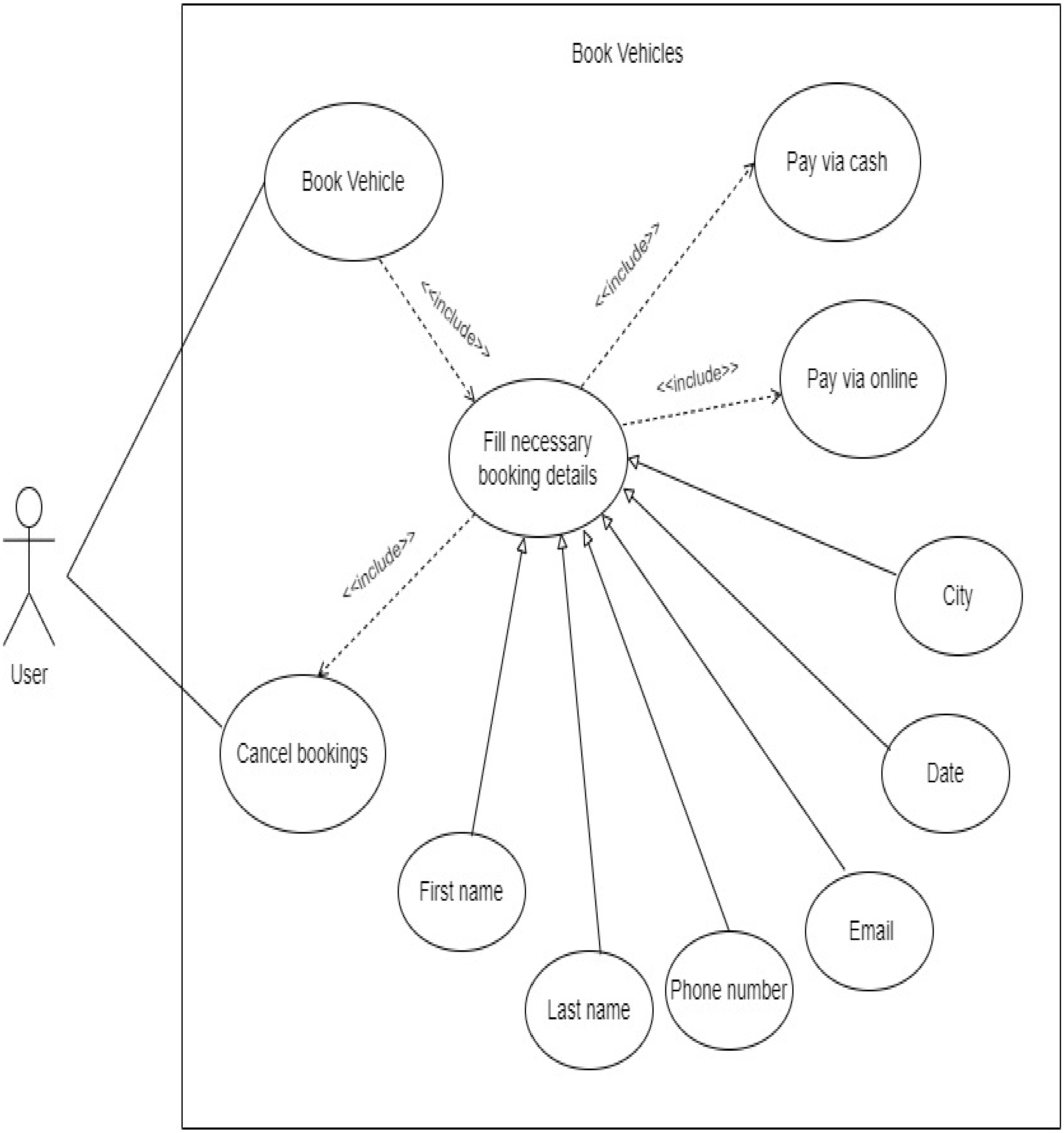


Figure 7: Book Vehicles Use Case Diagram

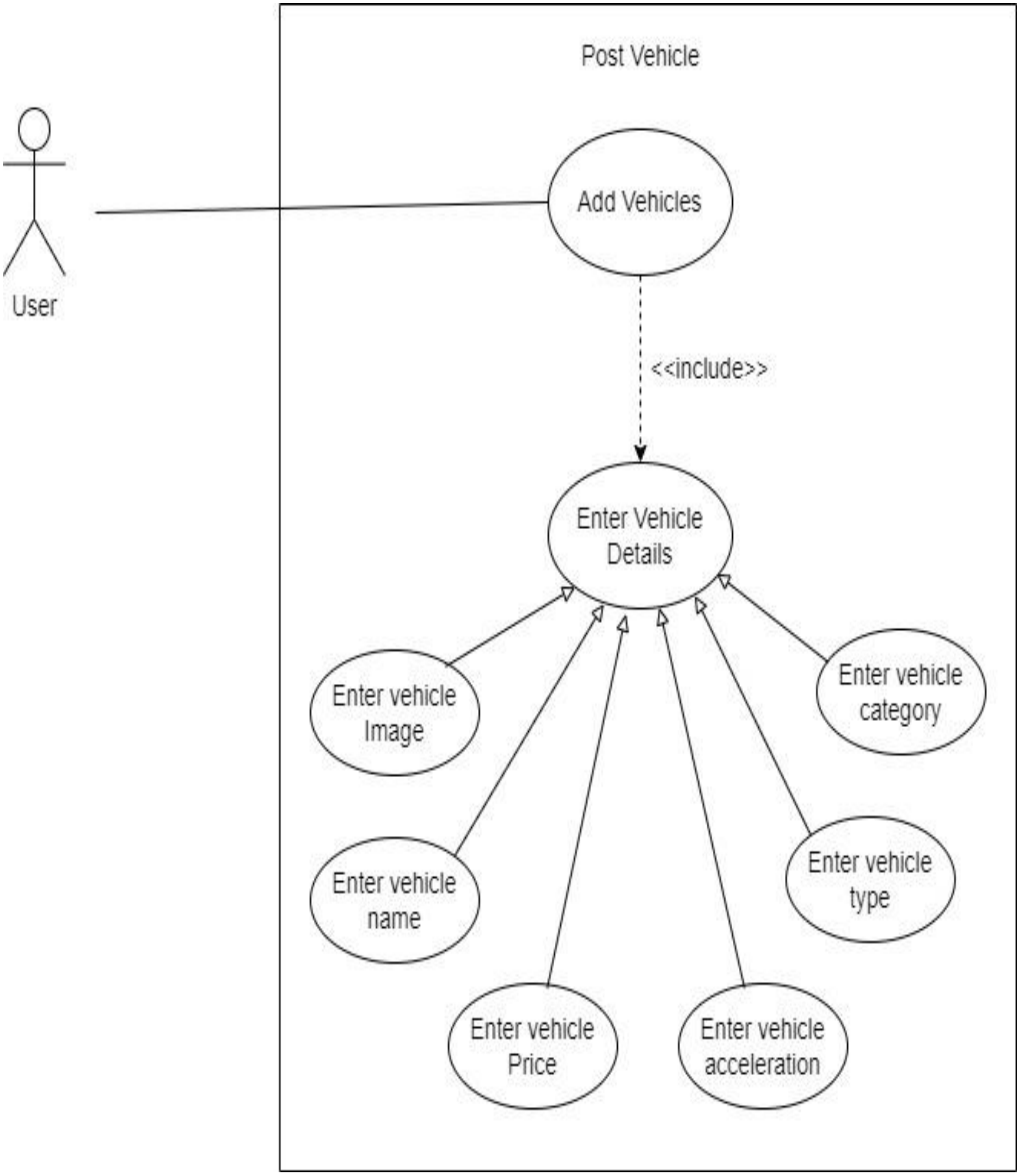


Figure 8: Post Vehicle Use Case Diagram

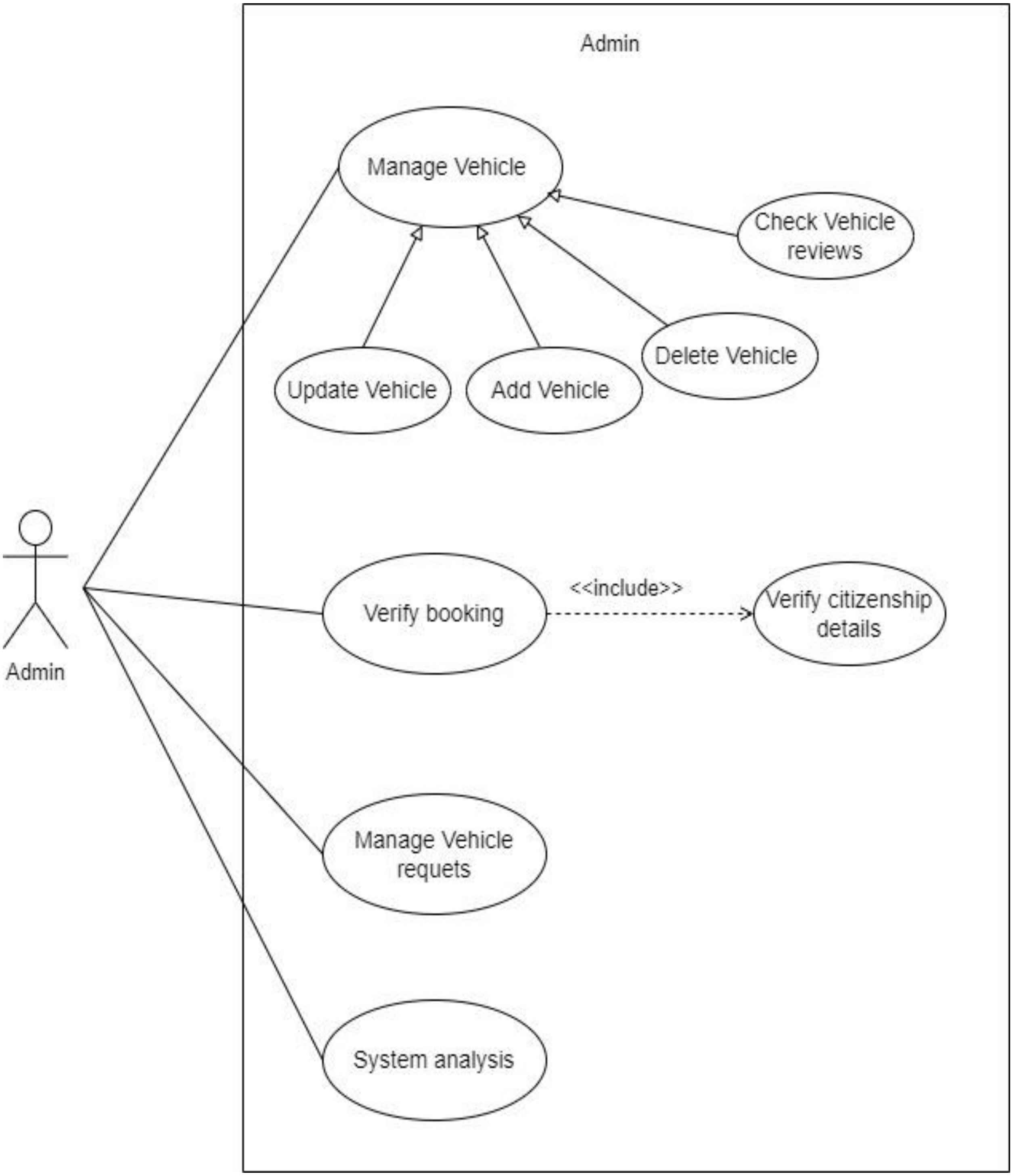


Figure 9: Admin Use Case Diagram

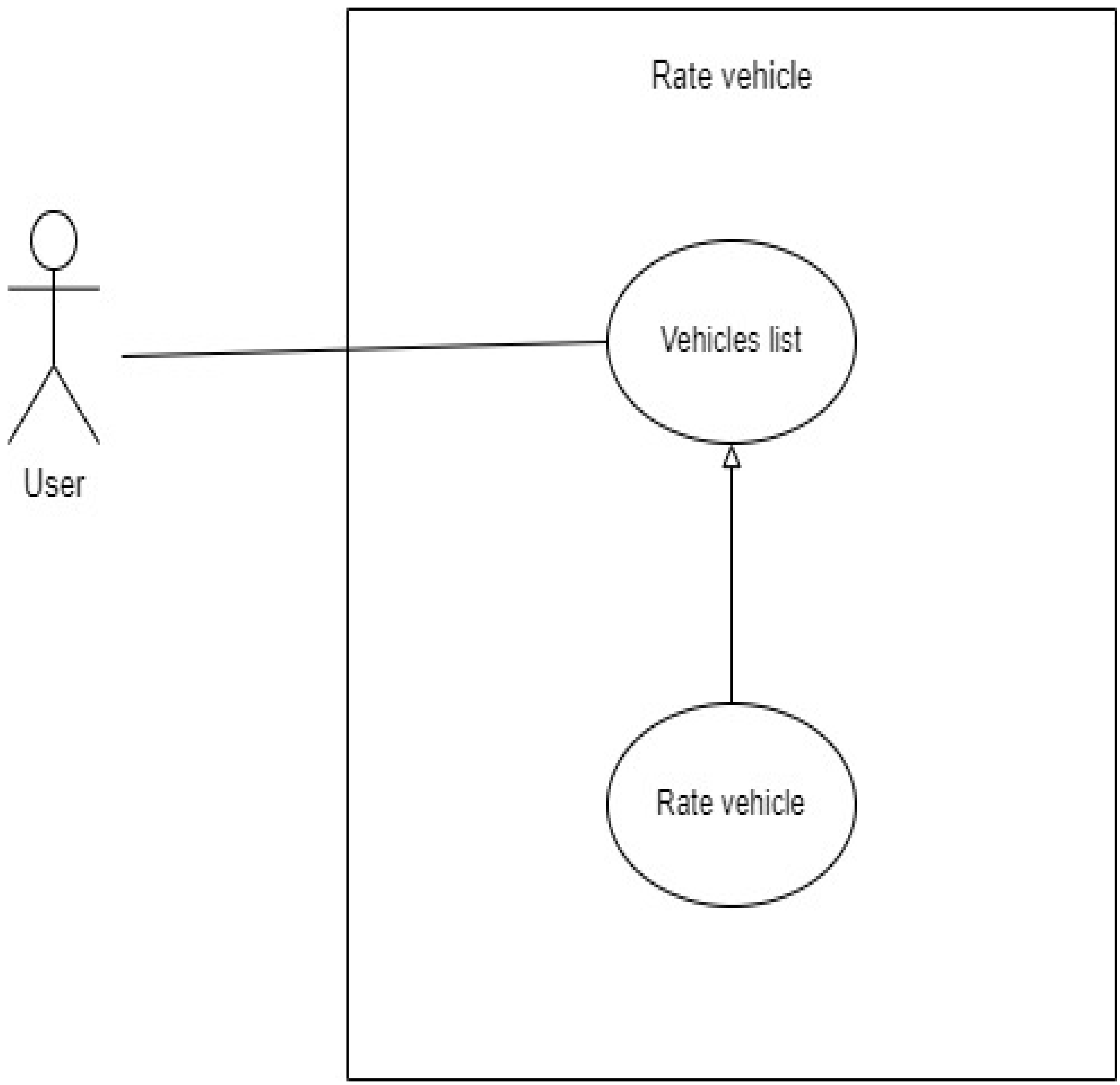


Figure 10: Rate Vehicle Use Case Diagram

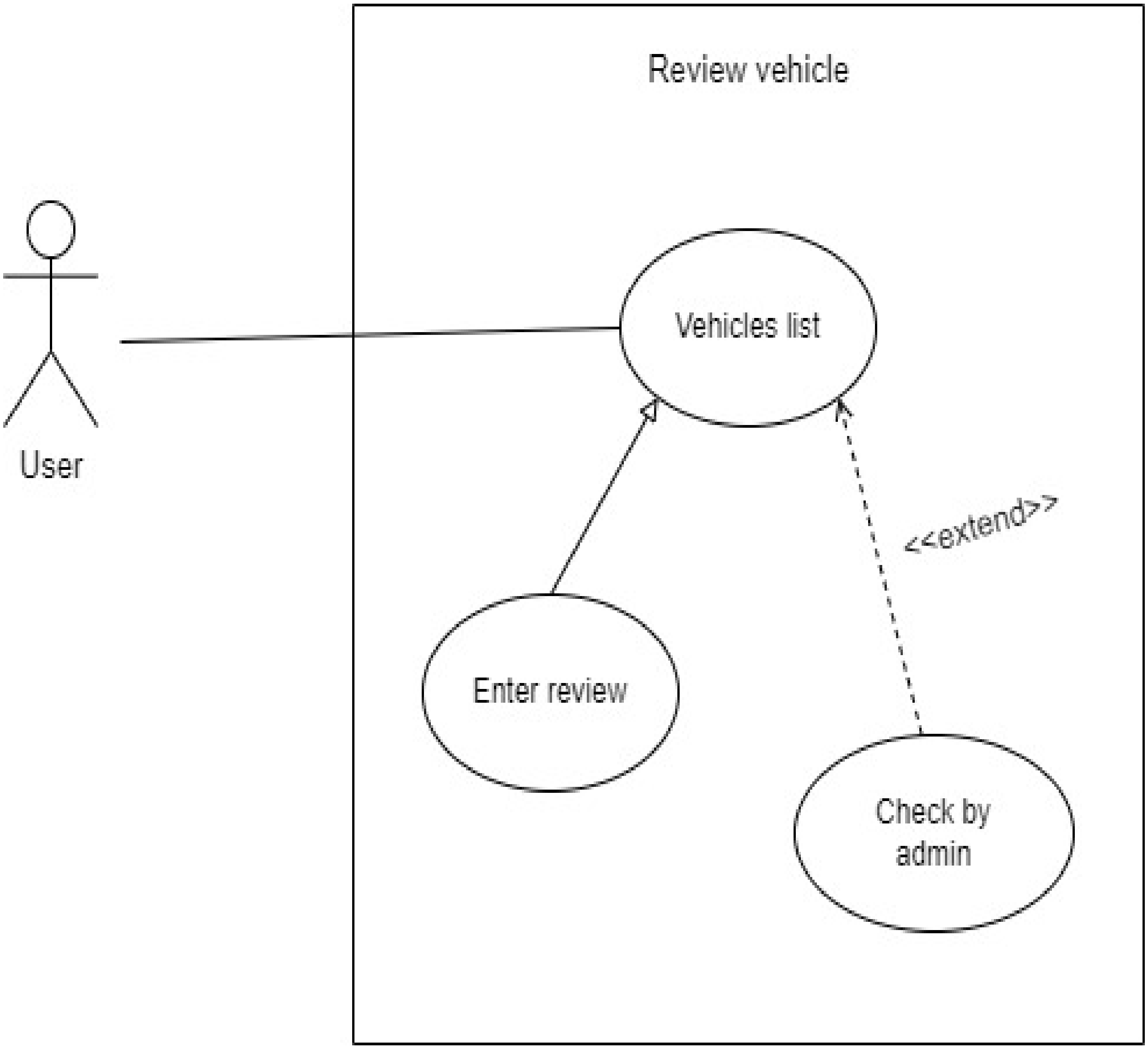


__Figure 11: Review Vehicle Use Case Diagram__

9.2 Entity Relationship Diagram

An Entity Relationship Diagram (ER Diagram) pictorially explains the relationship between entities to be stored in a database. Fundamentally, the ER Diagram is a structural design of the database. It acts as a framework created with specialized symbols for the purpose of defining the relationship between the database entities. ER diagram is created based on three principal components: entities, attributes, and relationships. An ER diagram of our system is represented as:

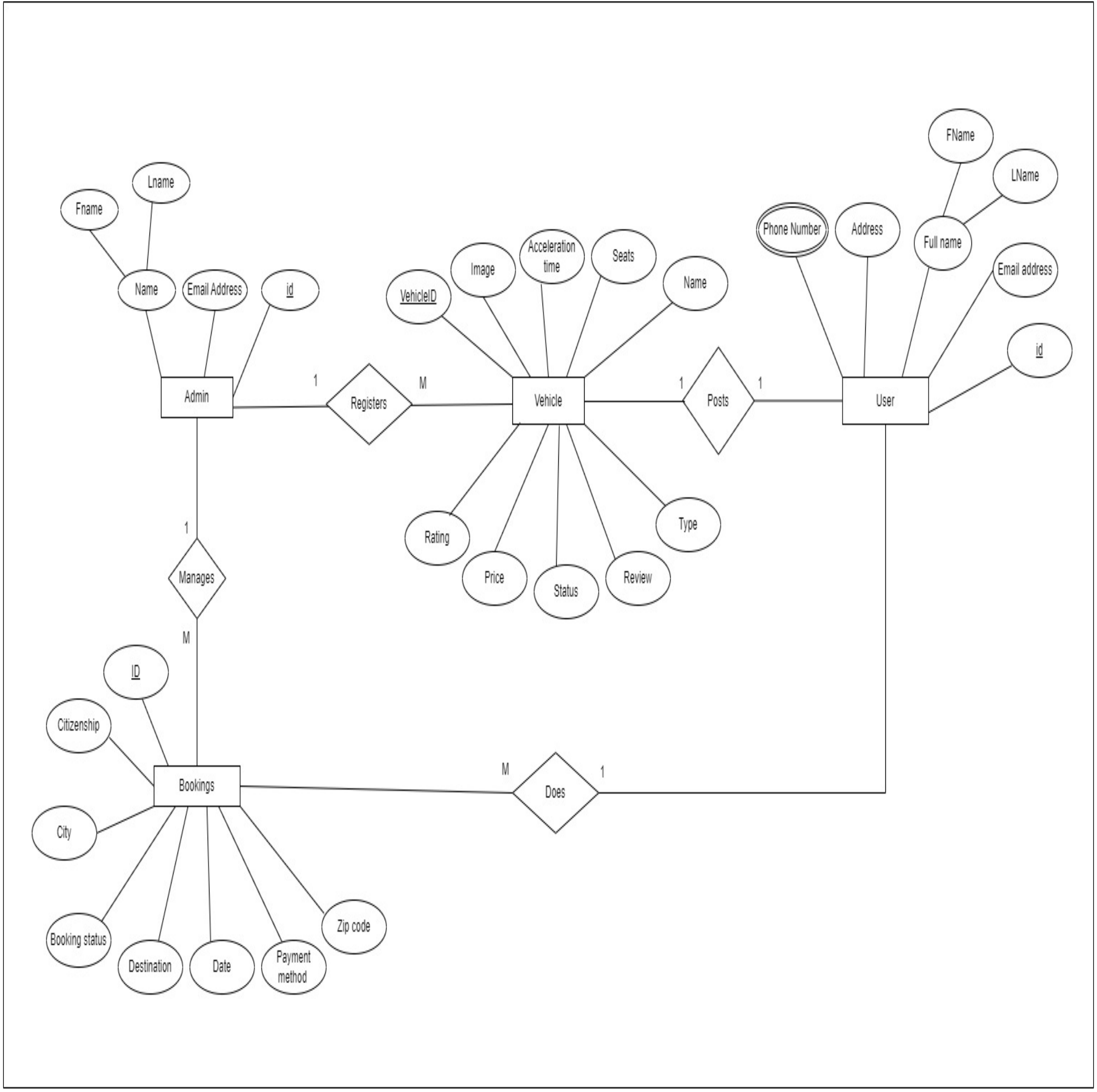


Figure 12: Entity Relationship Diagram

9.3 System Flow Diagram

A system flow diagram, also known as a system flowchart or process flow diagram, is a graphical representation of the flow of information, data, or materials within a system or process. It visually illustrates the sequence of steps, decisions, and interactions between different components or entities in the system.

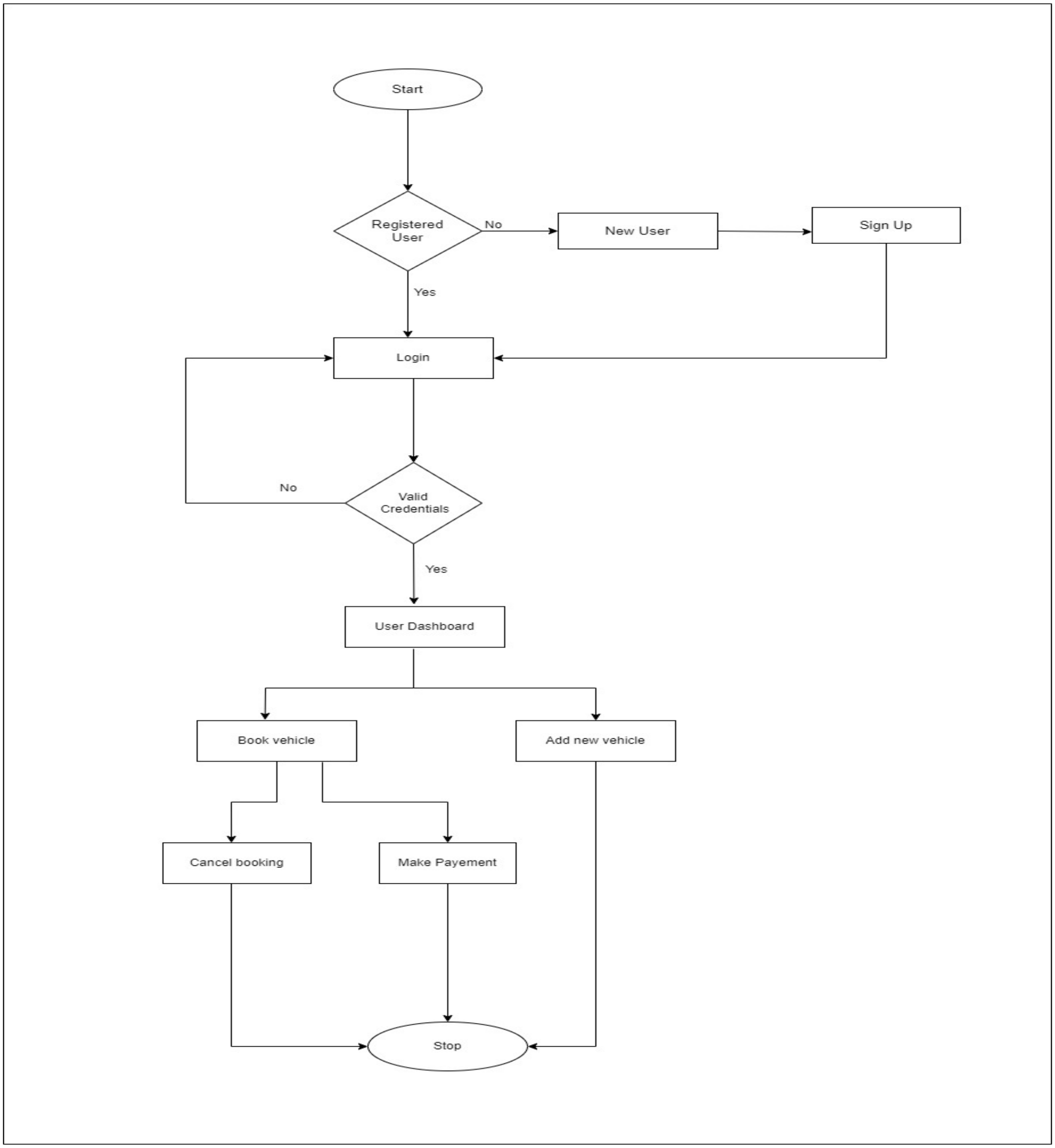


Figure 13: System Flow Diagram for Users

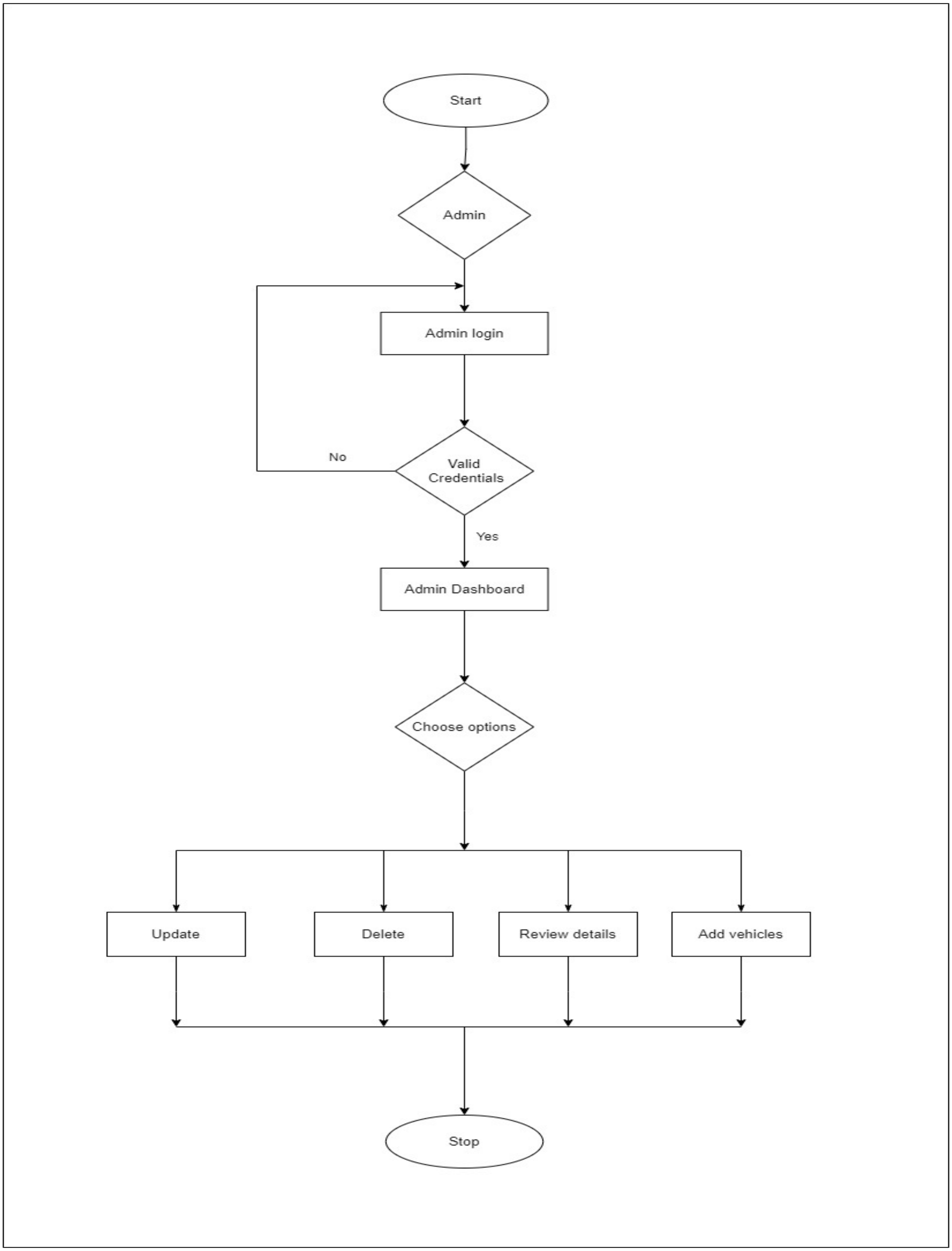


Figure 14: System Flow Diagram for Admin

9.4 Data Flow Diagram

A data flow diagram (DFD) shows how information moves through any system or process. Data flow diagrams can be as basic as hand-drawn process overviews or more complex, multi-level DFDs that progressively delve deeper into the data handling process. They can be used to model a new system or analyze an existing one. The data flow of our system is shown below:

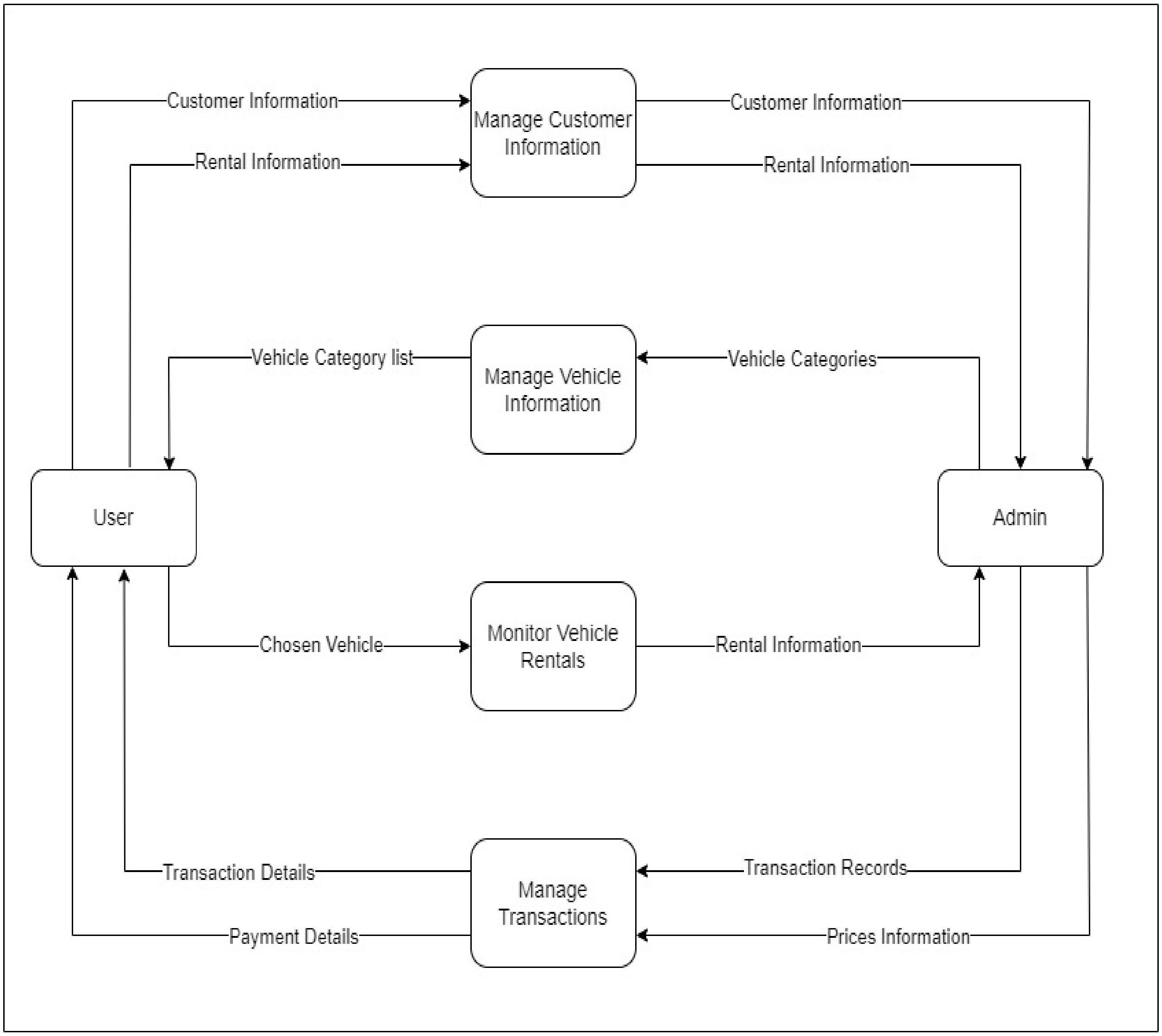


Figure 15: Data Flow Diagram

9.5 Activity Diagram

An activity diagram shows the flow from activity to activity within a system. An activity shows a set of activities, the sequential or branching flow from activity to activity, and objects that act and are acted upon shows what activities can be done in parallel, and any alternate paths through the flow. The activity diagram for the few of the scenarios while using system are as follows:

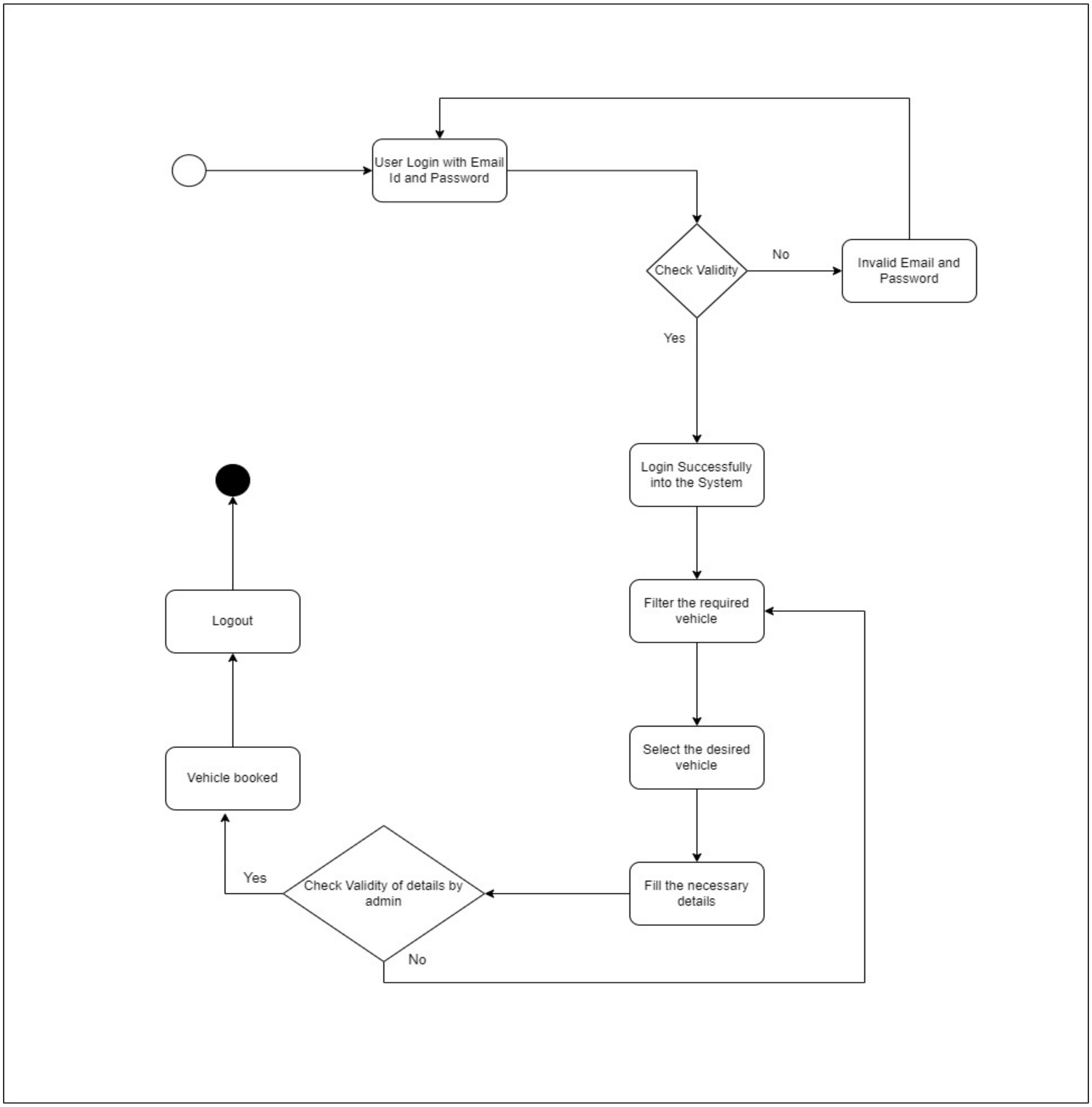

User Login with Email Id and Password
Check Validity
No
Invalid Email and Password
Yes
Login Successfully into the System
Filter the required vehicle
Select the desired vehicle
Fill the necessary details
Check Validity of details by admin
Yes
No
Vehicle booked
Logout

Figure 16: Activity Diagram of User for Booking Vehicle

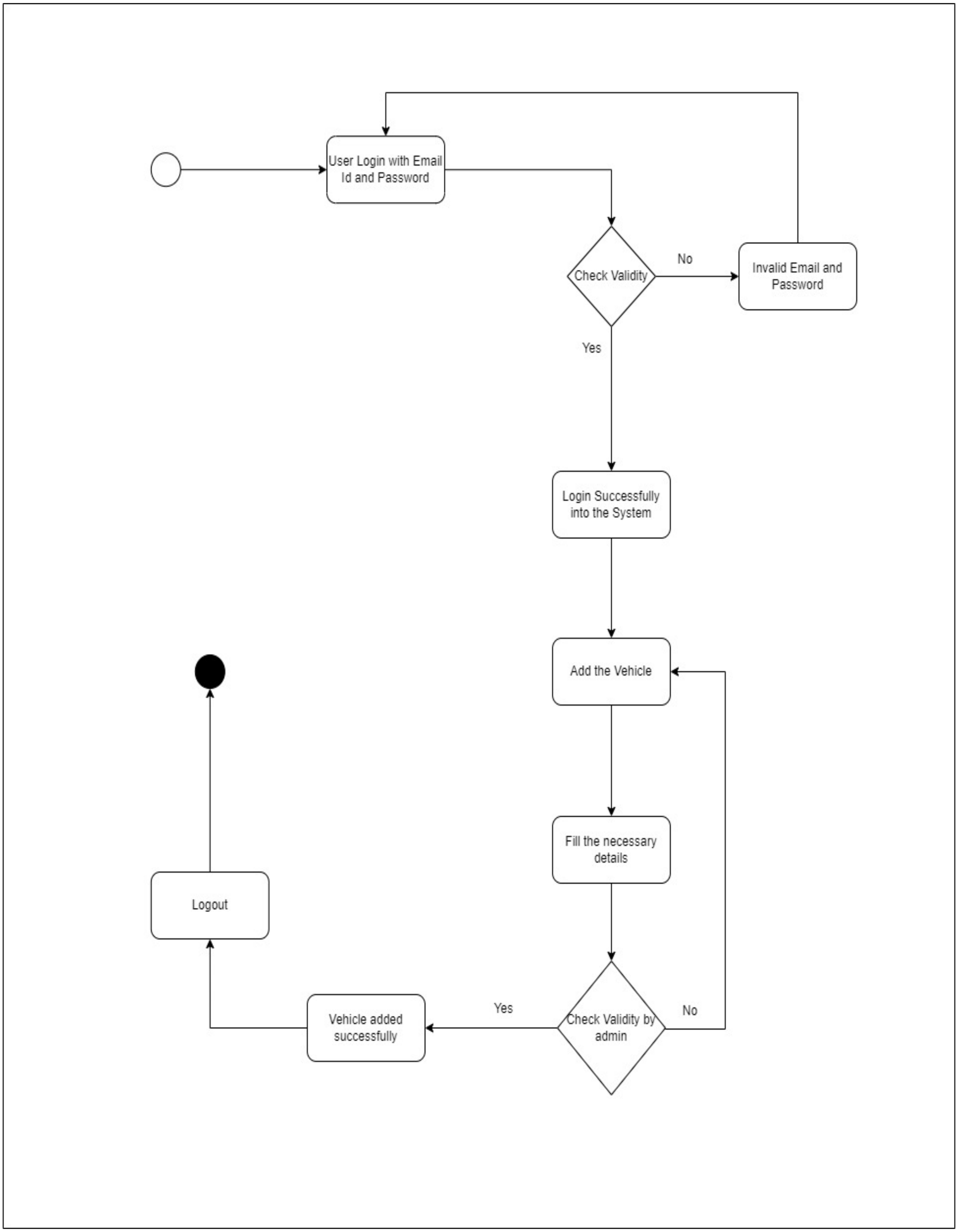

User Login with Email Id and Password
Check Validity
No
Invalid Email and Password
Yes
Login Successfully into the System
Add the Vehicle
Fill the necessary details
Check Validity by admin
No
Yes
Vehicle added successfully
Logout

Figure 17: Activity Diagram of User for Posting Vehicle

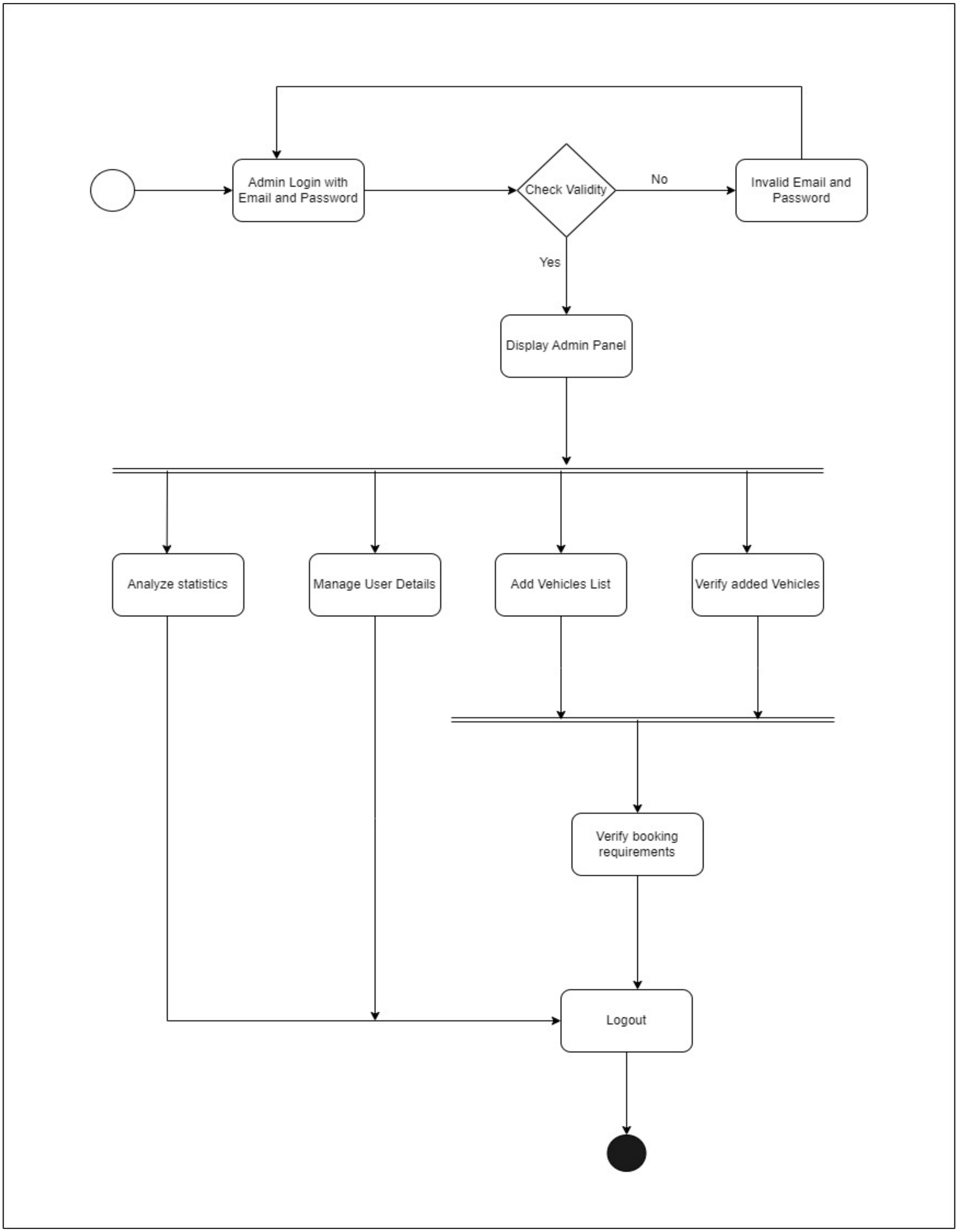


Figure 18: Activity Diagram of Admin

9.6 Sequence Diagram

A sequence diagram is an interaction diagram that emphasizes the time ordering of messages and shows a set of objects and the messages sent and received by those objects. The objects are typically named or anonymous instances of classes, but may also represent instances of other things, such as collaborations, components, and nodes.

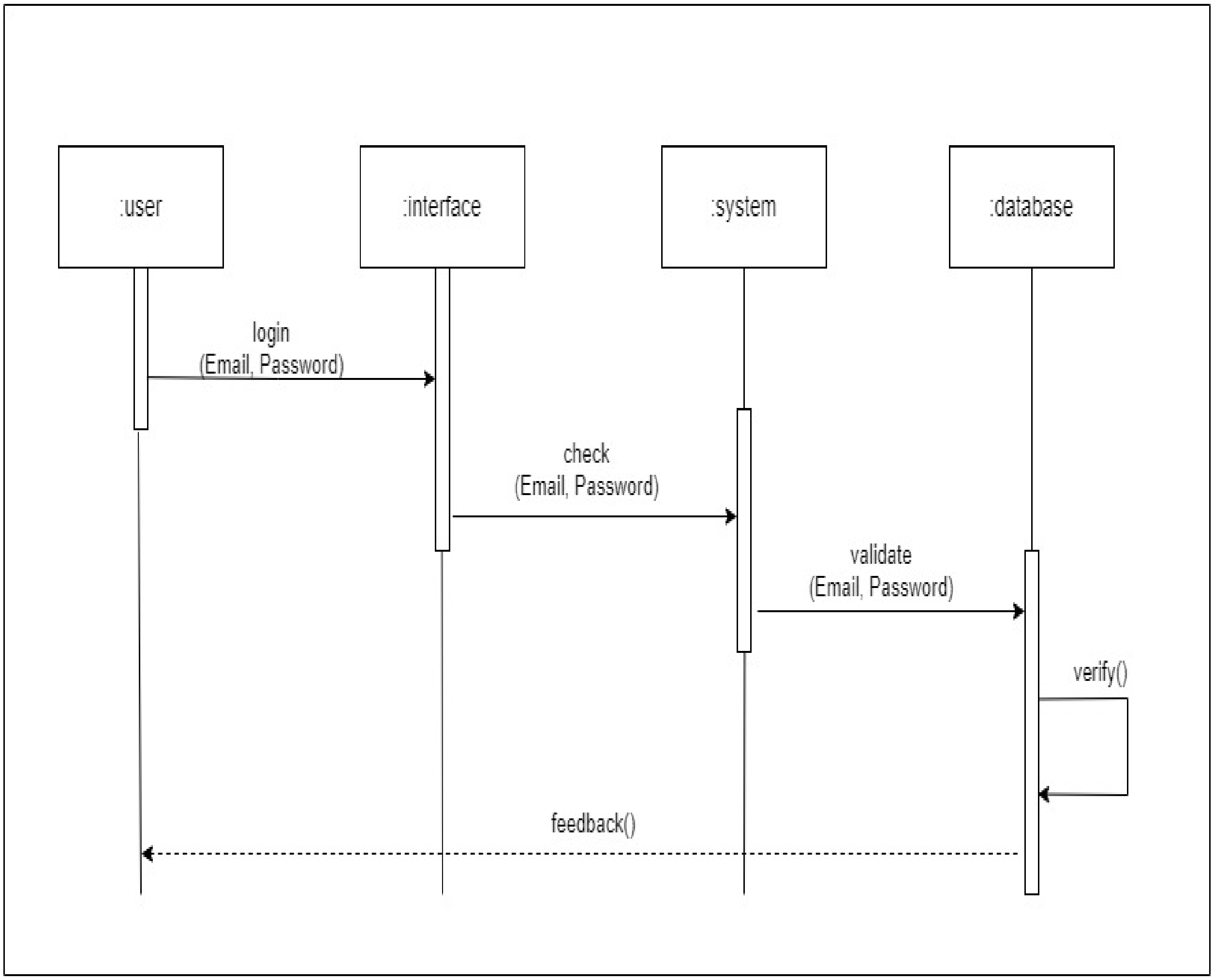


Figure 19: Sequence Diagram of User

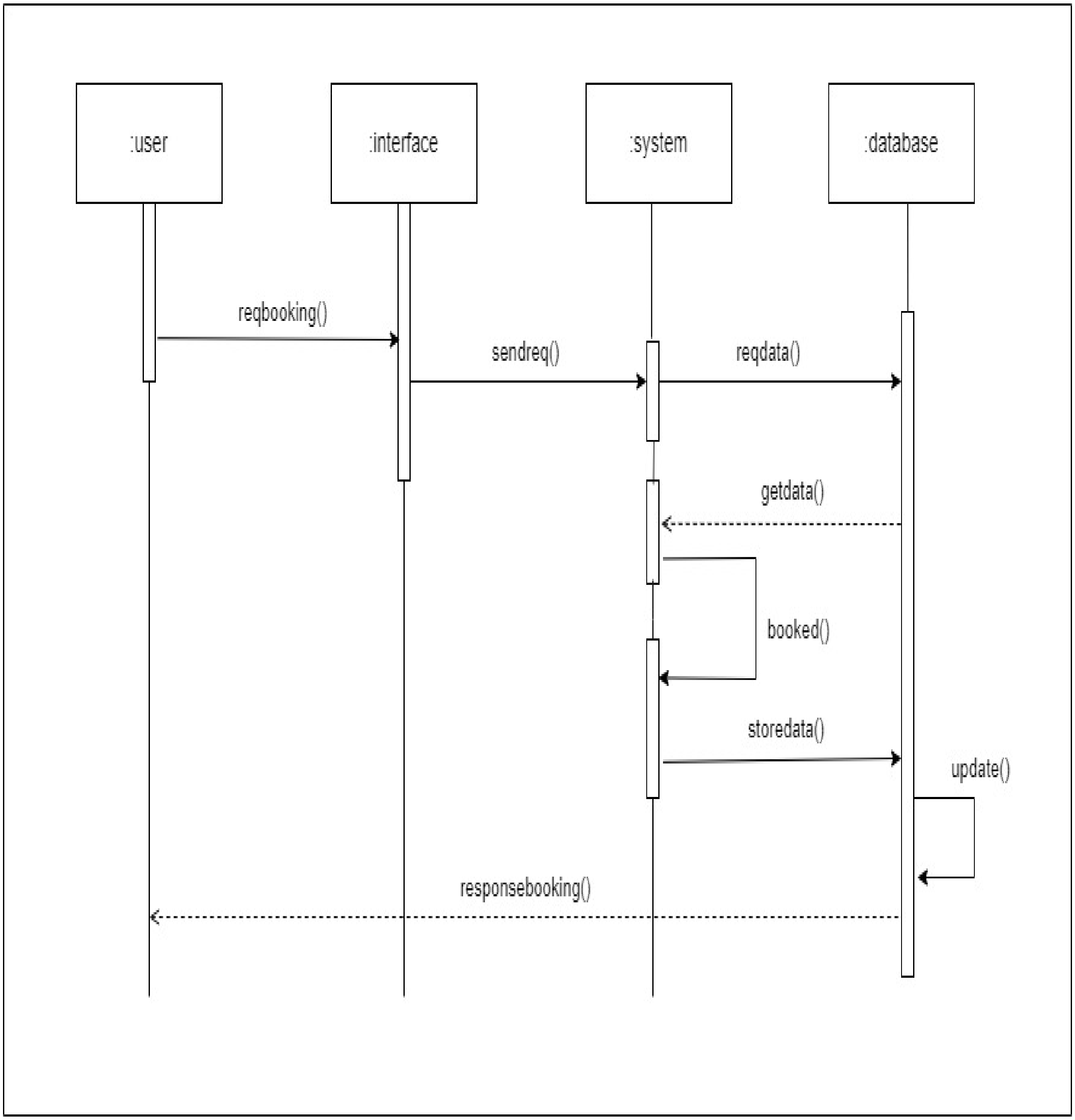
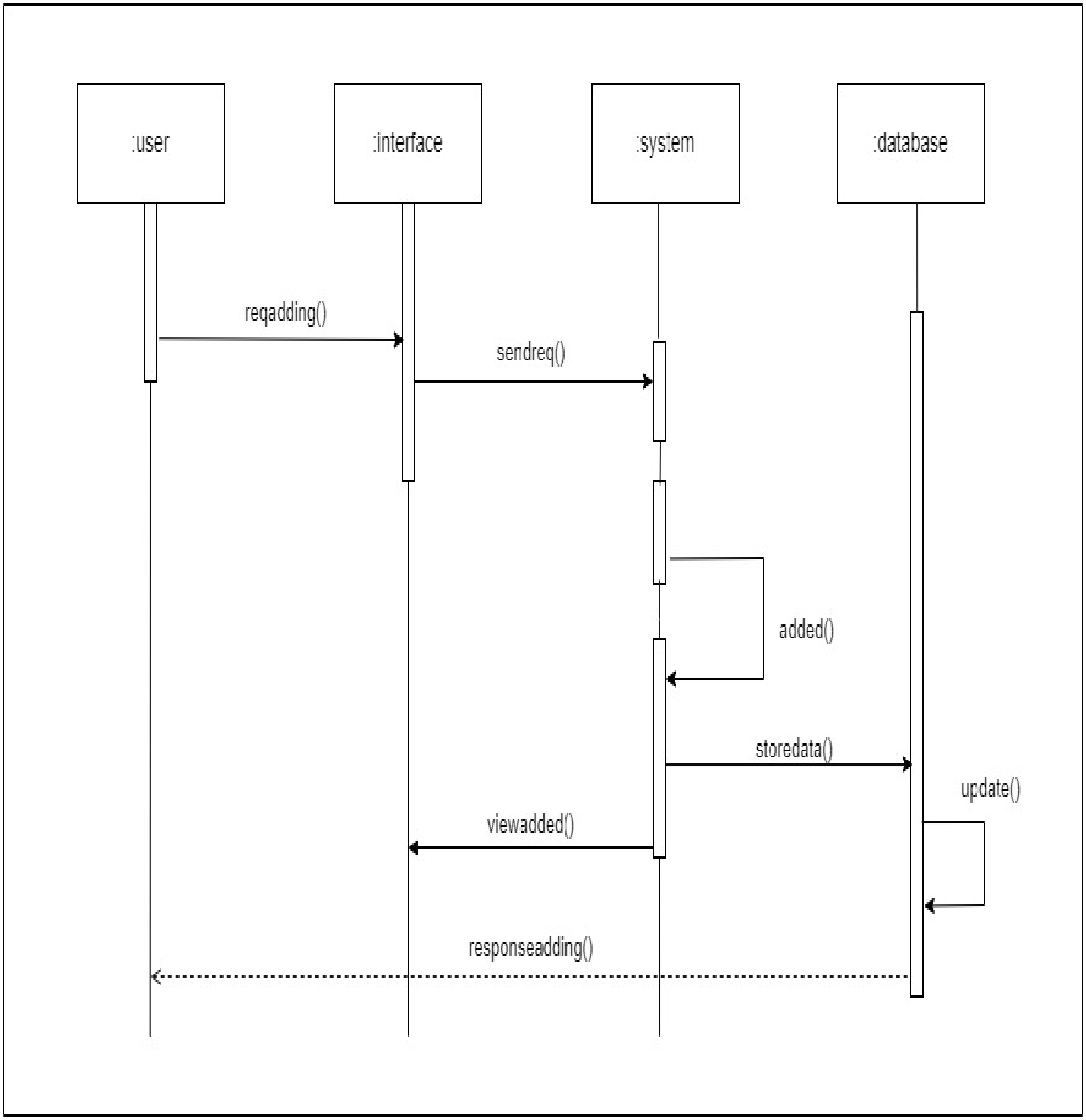


Figure 20: Sequence Diagram for Renting vehicle



<u>Figure 21: Sequence Diagram for Adding vehicle</u>

11. Deliverable

At the end of the project, we have ourselves a fully implemented application that has been deployed on the web so that it is accessible to everyone. The output of the project will be the application itself and the application will have all proposed features incorporated and requirements fulfilled. The overall proposed outcome of the project is an application that has the following features and provisions:

- User authentication, i.e., signing in and signing up.
- The proposed system allows users to add their own vehicles for rent, which can be an excellent source of passive income for vehicle owners.
- Users can rate and review the vehicles they rent, which can help future renters make informed decisions.
- Users will be able to view their booking history and current bookings in their profile, making it easy to manage their rentals and plan future trips.
- The admin will have the ability to add, update, and delete vehicles from the system, which can help keep the inventory up-to-date and relevant.
- The admin also has control over user reviews, meaning that if any inappropriate content is posted, it can be removed or edited.
- Users will also be recommended with the similar vehicles during the booking process.
- The admin can access to a dashboard where they can view users' booking information, allowing them to keep track of usage and identify any potential issues.

12. Project Iterations and Time Scheduled

The project schedule has been designed as per requirements and constraints involved. Requirement analysis have been given more emphasis. Research and database management is to be done first and well documented. Debugging and Testing is to be done prior to the completion of the project.

12.1 Project Iterations

This project is completed at 3 iterations. The overall work that we performed in these 3 iterations are:

12.1.1 Iteration 1

1. Make Plans for all the tech stack required for completing the project with given timeframe. It includes programming languages, design patterns, designs and algorithms associated to it.
2. Make a wire frame on the vehicle rental system that shows the clear picture of the schemas related to the associated Codes.
3. Make a database Structures with associated keys related to the application. This needs to be done on the first iteration as SQL is a rigid database structures that needs proper execution and planning. Modifying the schemas is a time taking and tough job.

12.1.2 Iteration 2

1. Make Backend API and the planned database wire framing as discussed and mentioned in Iteration 1.
2. Make a strong authentication and authorization system using the OAuth or JWT pattern for a strong application security.
3. Slight changes to be made (if necessary) in the planed database if necessary. Might be unavoidable.
4. Plan for the front end development.

12.1.3 Iteration 3

1. Make an executable front end using a proper and easy to use front end development programming language and associated framework.
2. Use Python and its necessary libraries to implement the AI related tasks and integrate it with Spring Boot Backend.
3. Complete the project by executing and combining designing, front end logic and back end logics.
4. Make necessary changes after doing a good integration testing with a feasible QA after the completion of project. Fix the bugs if required.

12.2 Expected Time Scheduled

The total development time for the completion of this project is 9 weeks. Design and coding has been given more emphasis due to the fact that we plan to add many features, but comparatively lesser time has been given to analysis and testing even though they are equally important. The schedule for each iteration is given below:

Task	Iteration 1	Iteration 2	Iteration 3	Total Approximated Duration (in days)
Requirements Analysis	4	2	2	8
System Design	3	3	2	8
Coding	11	11	9	31
Overall System Test	3	2	2	7
Producing Documentation	4	4	3	11

Table 2: Time Scheduled

12.3 Gantt chart

12.3.1 Iteration 1 Gantt chart

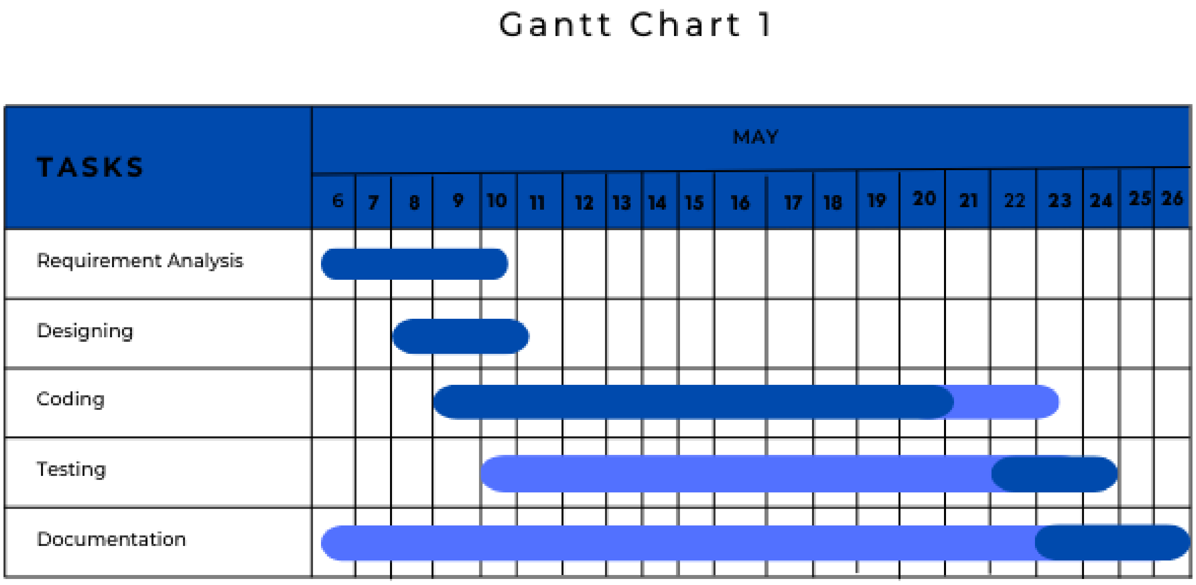


Figure 22: Gantt chart for Iteration 1

12.3.2 Iteration 2 Gantt chart

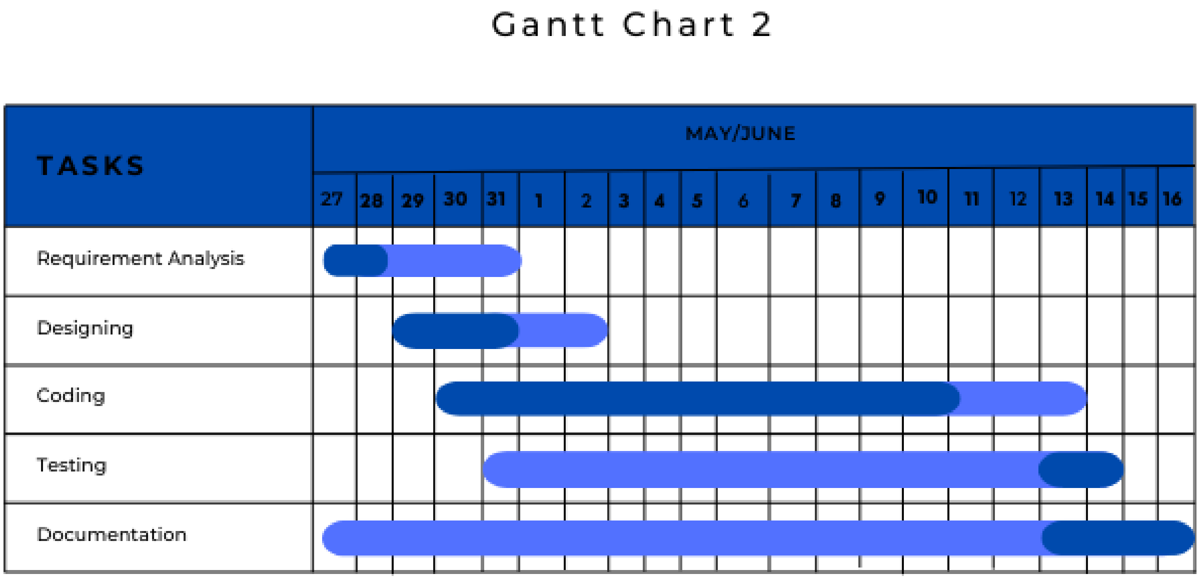


Figure 23: Gantt chart for Iteration 2

12.3.3 Iteration 3 Gantt chart

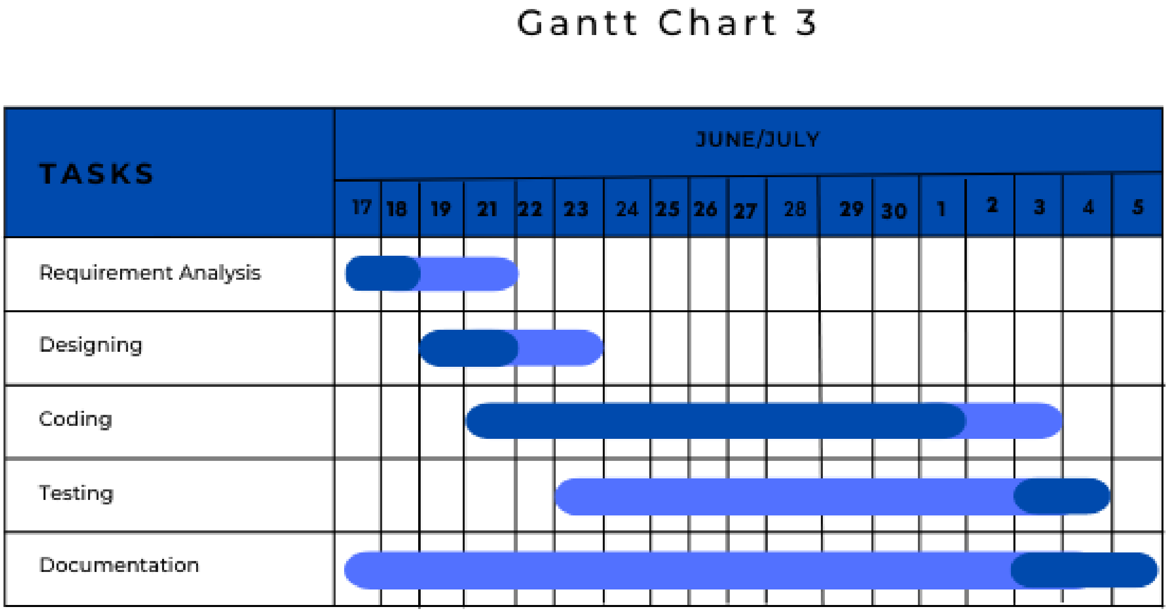


Figure 24: Gantt chart for Iteration 3

13. Testing

Testing is the systematic process of evaluating a system or its components to determine if they meet the specified requirements. Its purpose is to identify any discrepancies, gaps, errors, or missing elements in relation to the intended requirements. In essence, testing involves executing the system and comparing its actual behavior against the expected behavior to ensure its quality and compliance with the desired outcomes. Usually, following testing types are used to check the integrity of the system.

13.1 Unit Testing

A unit is the smallest testable part of an application. The primary goal of unit testing is to take the smallest piece of testable software in the application and isolate it from the remainder of the code, and determine whether it behaves exactly as expected.

S.N	Condition Tested	Expected Result	Actual Result	Remarks
1.	Invalid Email and Password is entered	Error message generated	As per the expected result	Pass
2.	Password is entered without any special character and numbers.	Error message generated	As per the expected result	Pass

Table 3: Unit Testing

13.2 Integration Testing

It is a logical extension of unit testing. Different units that have already been tested are combined into a component and the interface between them is tested. Integration testing identifies problems that occur when units are combined. Thus, integration testing is the phase where individual software modules are combined and tested as a group.

S.N	Condition Tested	Expected Result	Actual Result	Remarks
1.	Navigate to the home page after correct email and password is entered	Proceeds on correct entry	As per the expected result	Pass
2.	After a required vehicle is selected it should redirect to the booking page.	Redirects to the page.	As per the expected result	Pass

Table 4: Integration Testing

13.3 System Testing

The system test focuses on the behavior of the system as a whole. User scenarios will be executed against the system as well as error messages and navigations will be tested.

S.N	Condition Tested	Expected Result	Actual Result	Remarks
1.	Upload the vehicles by admin and saves into database	Successful	Successful	Data uploaded successfully.
2.	Vehicles being booked are shown unavailable on the dashboard.	Shows unavailable.	Shows unavailable.	Successful.
3.	Vehicles added by the users and accepted admin are shown dashboard.	Successful	Successful	Vehicles added successfully.

Table 5: System Testing

14. Conclusion

The development and implementation of the Rented Wheels have been a significant endeavor, resulting in a robust and efficient solution for managing vehicle rentals. Through the collaborative efforts of our team and the guidance of our project guide, we have successfully achieved our project objectives and delivered a high-quality system.

Rented Wheels provides numerous benefits and functionalities to the customers. It offers a user-friendly interface for customers to search, select, and book vehicles based on their preferences and requirements. The system incorporates features such as real-time availability, pricing information, and secure payment options, ensuring a seamless and convenient experience for users.

While we have successfully implemented the core functionalities of the vehicle renting system, we recognize that there is always room for improvement. Future enhancements could include integrating additional features such as vehicle tracking, and seamless integration with third-party platforms. Continuous updates and maintenance will be necessary to adapt to changing customer needs and technological advancements.

Overall, the Rented Wheels has been a valuable learning experience for our team, allowing us to apply our knowledge and skills in a real-world scenario. We are proud of the outcomes we have achieved and believe that this system will greatly benefit rental companies and customers alike in managing and accessing vehicles in a more efficient and convenient manner.

15. Further Works

While our project initially met the stated requirements, there are still areas that can be explored for further improvement. Change is inevitable, and there is always room for enhancement to cater to user needs, adapt to a changing environment, and incorporate new features. By considering these aspects, we can identify potential areas for improvement in the project.

Some of the features and improvements that we have planned to add in Rented Wheels for further works are:

1. An android or iOS App can be developed that increases the user engagement on our application.
2. Implement a notification system to send automated reminders to users about upcoming bookings, return dates, and other important information related to their rentals.
3. Continuously improve the user interface, navigation, and overall user experience based on user feedback and usability testing.

16. References

During the preparation of the project, we took references so as to learn a few important steps:

[1] "Autocios"; *Vehicle renting System* [Online]. Available at *https://autocios.com/about*

[2] "Bikemandu"; *Motorcycle Renting Agency* [Online]. Available at *https://www.bikemandu.com/*

[3] "Software Engineering –The incremental process model" [Online] Available: *https://www.geeksforgeeks.org/software-engineering-incremental-process-model*

[4] Pratik Sharad Maratkar , Pratibha Adkar "React JS- An Emerging Frontend JavaScript Library" *Iconic Research And Engineering Journals* Vol.4 Issue 12 2021 Page 99-102 [Online]. Available: *https://www.irejournals.com/paper-details/1702778*

[5] "Guide to Spring boot," [Online]. Available: *https://spring.io/guides/tutorials/spring-boot.*

[6] B. P. S. &. M. Rawat, ""MySQL Database Management System (DBMS)", *International Journal of Cyber and IT Service Management (IJCITSM)*, vol. 8, no. 11, pp. 173-179, 2021. [Online]. *Available: https://iiast-journal.org/ijcitsm/index.php/IJCITSM/article/view/47*

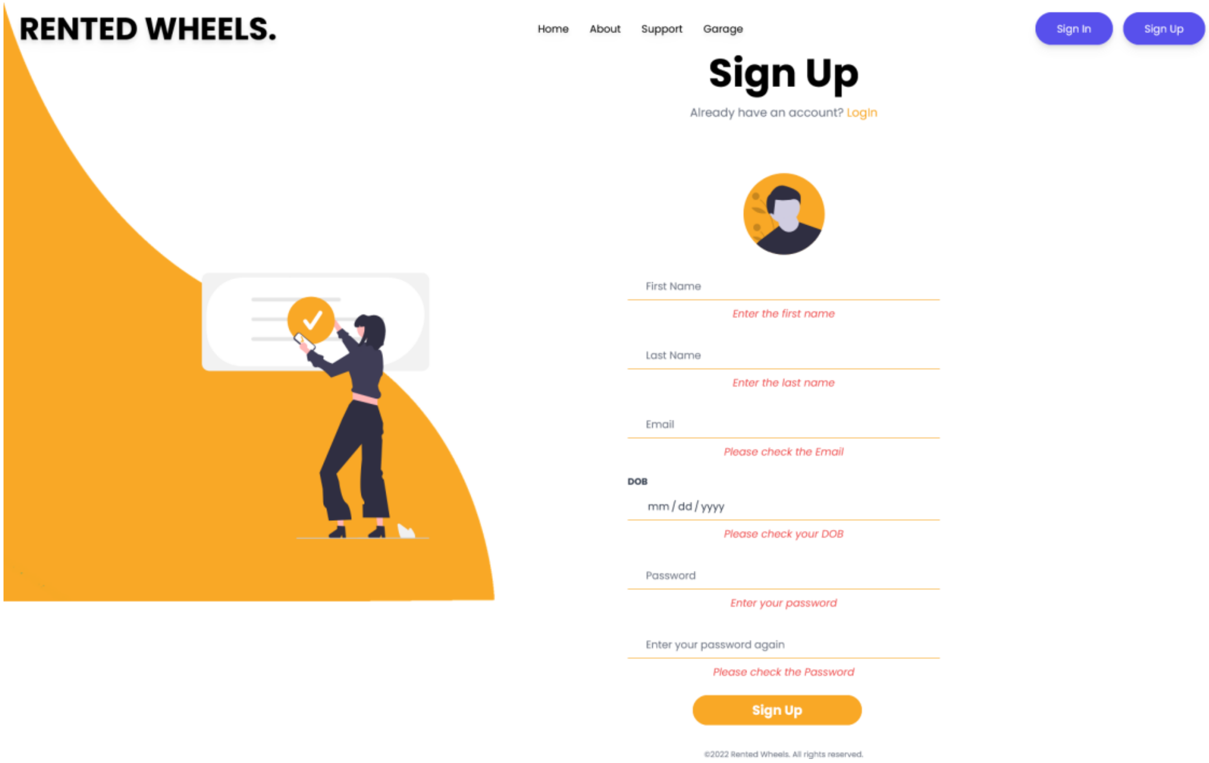
[7] "Article on Python for AI and Machine learning", [Online]. Available: *https://steelkiwi.com/blog/python-for-ai-and-machine-learning.*

[8] "Project Proposals Formats" [Online] Available at *http://lms.ncit.edu.np/moodle/*

[9] "A Guide to Content-Based Filtering In Recommender Systems", [Online]. Available: *https://www.turing.com/kb/content-based-filtering-in-recommender-systems*

17. Appendix

17.1. Appendix A User Registration



Sign Up

Already have an account? Login

First Name

Enter the first name

Last Name

Enter the last name

Email

Please check the Email

DOB

mm / dd / yyyy

Please check your DOB

Password

Enter your password

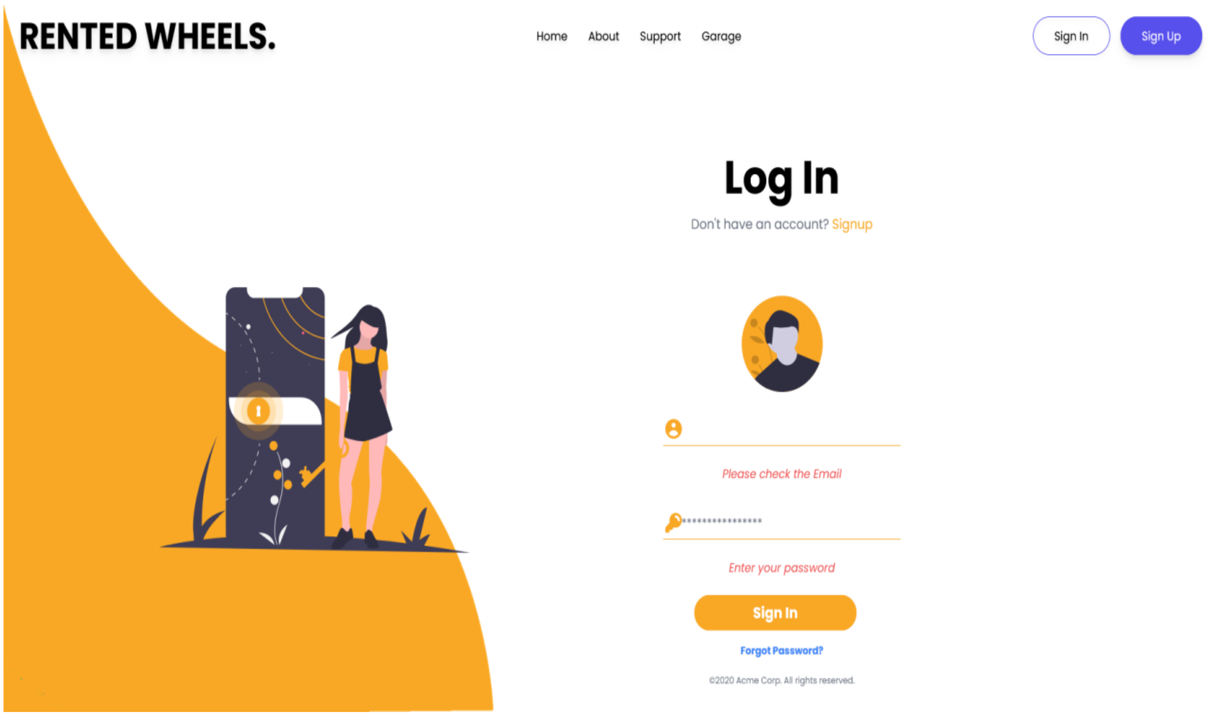
Enter your password again

Please check the Password

©2022 Rented Wheels. All rights reserved.

Figure 25: User Registration

17.2. Appendix B User Login



Log In

Don't have an account? Signup

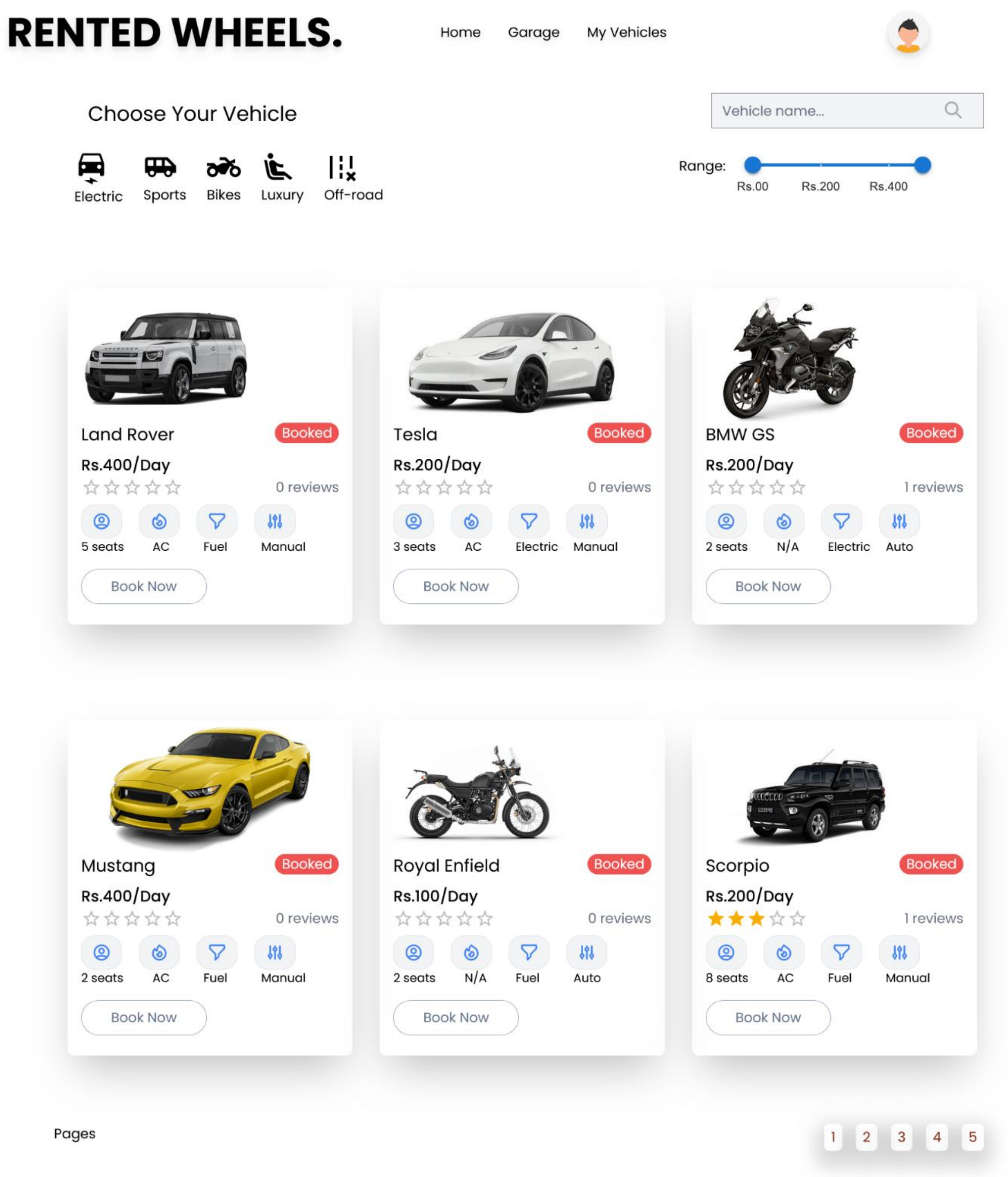
Please check the Email

Enter your password

©2020 Acme Corp. All rights reserved.

Figure 26: User Login

17.3. Appendix C User Dashboard



RENTED WHEELS.

Home Garage My Vehicles

Choose Your Vehicle

Vehicle name...

Electric Sports Bikes Luxury Off-road

Range: Rs.00 Rs.200 Rs.400

Land Rover Booked
Rs.400/Day
☆☆☆☆☆ 0 reviews
5 seats AC Fuel Manual
Book Now

Tesla Booked
Rs.200/Day
☆☆☆☆☆ 0 reviews
3 seats AC Electric Manual
Book Now

BMW GS Booked
Rs.200/Day
☆☆☆☆☆ 1 reviews
2 seats N/A Electric Auto
Book Now

Mustang Booked
Rs.400/Day
☆☆☆☆☆ 0 reviews
2 seats AC Fuel Manual
Book Now

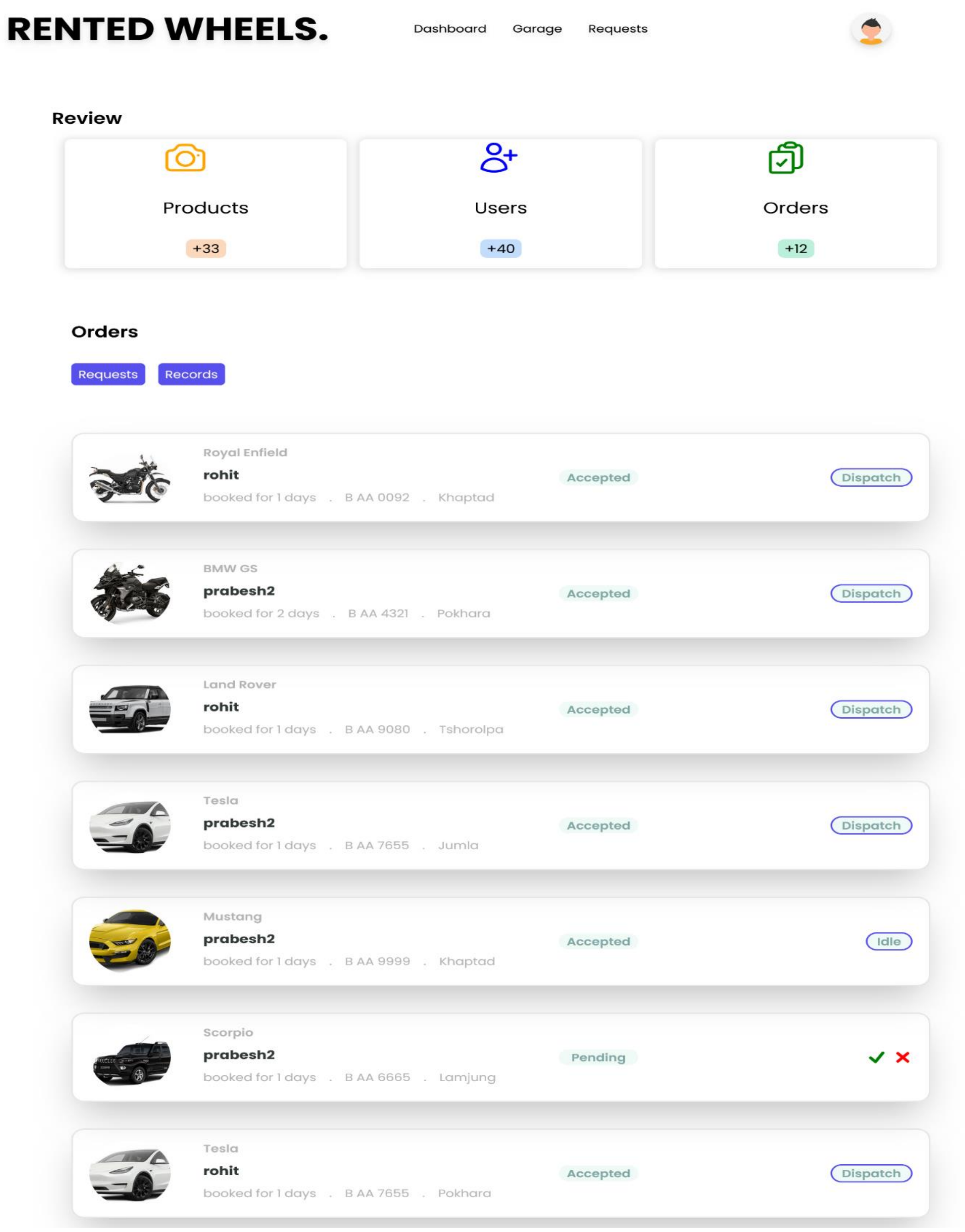
Royal Enfield Booked
Rs.100/Day
☆☆☆☆☆ 0 reviews
2 seats N/A Fuel Auto
Book Now

Scorpio Booked
Rs.200/Day
★★★☆☆ 1 reviews
8 seats AC Fuel Manual
Book Now

Pages 1 2 3 4 5

Figure 27: User Dashboard

17.4. Appendix D Admin Dashboard



RENTED WHEELS.

Review

Products
+33

Users
+40

Orders
+12

Orders

Royal Enfield
rohit
booked for 1 days . B AA 0092 . Khaptad
Accepted
Dispatch

BMW GS
prabesh2
booked for 2 days . B AA 4321 . Pokhara
Accepted
Dispatch

Land Rover
rohit
booked for 1 days . B AA 9080 . Tshorolpa
Accepted
Dispatch

Tesla
prabesh2
booked for 1 days . B AA 7655 . Jumla
Accepted
Dispatch

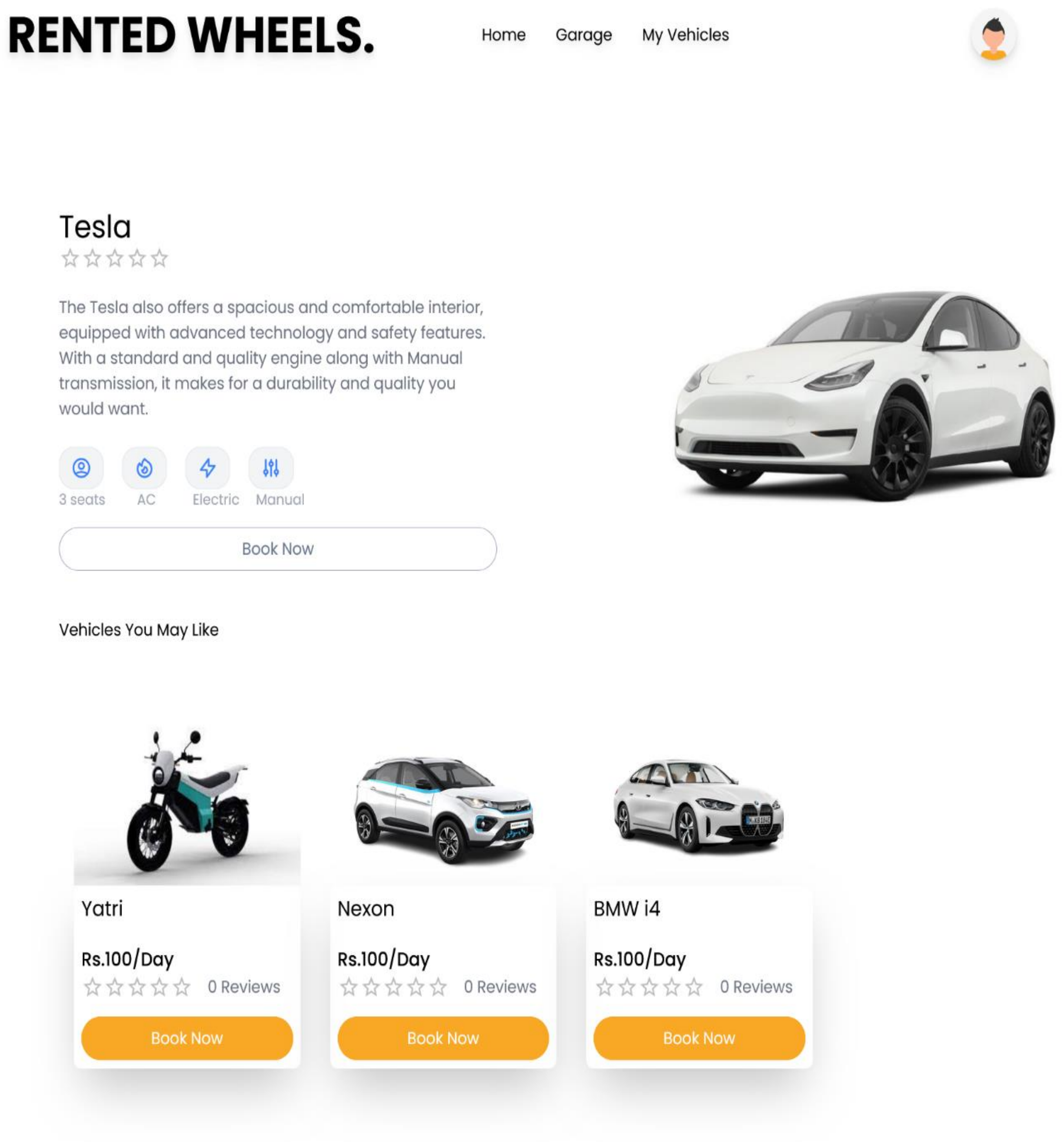
Mustang
prabesh2
booked for 1 days . B AA 9999 . Khaptad
Accepted
Idle

Scorpio
prabesh2
booked for 1 days . B AA 6665 . Lamjung
Pending
✓ ✗

Tesla
rohit
booked for 1 days . B AA 7655 . Pokhara
Accepted
Dispatch

Figure 28: Admin Dashboard

17.5. Appendix E Vehicle Information and Recommendation



The Tesla also offers a spacious and comfortable interior, equipped with advanced technology and safety features. With a standard and quality engine along with Manual transmission, it makes for a durability and quality you would want.

Figure 29: Vehicle Information and Recommendation

17.6. Appendix F Post Vehicle by user

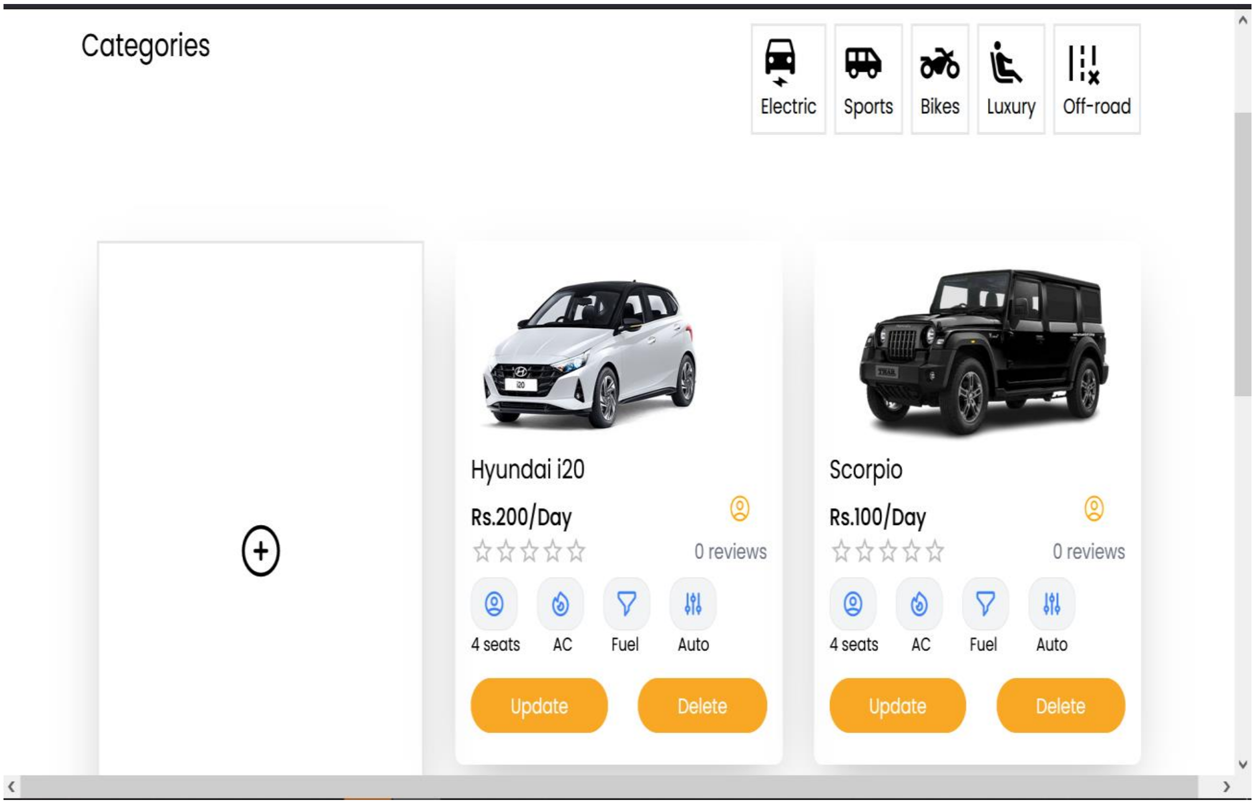


Figure 30: Post Vehicle by User