Software Requirements Specification

ODYSSEY

Prepared by

Nuzhat Nasrum BFH2125016F
Md Nazirul Islam Bijoy ASH21225023M
Md Masum Billah ASH2125033M
Md Imteaz Ahmed ASH25035M

Institute of Information Technology
Noakhali Science and Technology University

09.06.24

1. Introduction

The purpose of this document is to provide a detailed Software Requirements Specification (SRS) for our Odyssey platform. It outlines the functional and nonfunctional requirements, design constraints, and other critical specifications necessary for the development and implementation of the software. This document is intended for all stakeholders, including developers, and end-users, to ensure a comprehensive understanding of the software's capabilities and constraints.

1.1 Problem Statement

Odyssey seeks to enhance the car rental platform by addressing key challenges in user experience, booking processes, customer feedback, pricing, administrative oversight, and vehicle management. The goal is to create a seamless, user-friendly interface for customers to authenticate, search, and book vehicles, while providing robust tools for admins to manage promo codes, refunds, and user accounts. Additionally, the platform must offer effective vehicle management features for owners, including maintenance logging and booking management, all while ensuring transparent pricing and a fair fine system. By overcoming these challenges, Odyssey aims to deliver a reliable, efficient, and satisfying car rental experience for all stakeholders.

1.2 Purpose

The purpose of Odyssey is to provide a convenient and efficient solution for individuals and businesses seeking short-term vehicle rentals. By offering a user-friendly interface and a wide range of vehicle options, Odyssey aims to streamline the rental process, from searching and booking to payment. The platform aims to cater to diverse transportation needs and budgets, while also providing vehicle owners with a platform to list and manage their vehicles. Overall, Odyssey strives to deliver a seamless and reliable car rental experience for all stakeholders involved.

1.3 Project Scope

The "Odyssey" car rental platform is developed to provide comprehensive vehicle rental services and a seamless user experience.

The key features and capabilities of the platform include:

- Users can quickly search for their desired vehicle based on type, rental dates, locations, and price.
- Users can book online, specifying pick-up and drop-off dates and locations.

- Renters can view and manage their profiles, including checking rental history, current bookings, and any outstanding payments or dues.
- Users can send requests for specific types of vehicles.
- The platform supports secure online payment options to facilitate a smooth transaction process.
- Users can access customer support for any inquiries or issues related to their rental experience.

1.4 Glossary

This section provides definitions for all document names, acronyms, and abbreviations. The application domain's terms and concepts are defined.

XML - Extensible Markup Languag
PHP – Hypertext Preprocessor
SRS – Software Requirement Specification
UI – User Interface
API – Application Programming Interface
W3C – World Wide Web Consortium

1.6 Overview

The "Odyssey" Car Rental platform is designed to offer a convenient and efficient platform for both individuals and businesses seeking short-term vehicle rentals. This system allows users to browse through a wide variety of vehicles, providing flexibility in selecting rental dates and pick-up and drop-off locations. With an extensive range of vehicle options, Odyssey caters to different needs and budgets, ensuring that customers can find the right vehicle for their specific requirements.The platform emphasizes user-friendliness with its intuitive interface, making it easy for users to navigate and complete their rental transactions. Secure payment options are integrated to protect user data and facilitate smooth financial transactions. By streamlining the rental process, Odyssey aims to provide a seamless and hassle-free experience for customers, enhancing their ability to quickly and conveniently access transportation solutions.

2. Stakeholders and Characteristics

Owner:

- Individual or business that provides vehicles for rental.
- Responsible for the maintenance and upkeep of their vehicles.
- Ensure the availability and scheduling of their vehicles.
- Aim to generate income by renting out their vehicles.

Renter:

- Individuals or businesses seeking vehicle rentals.
- Book cars anytime and from anywhere.
- Access to a broad range of vehicles.
- Able to make booking requests, process payments, and extend rental periods.

Admin:

- Manages the overall operation of the platform.
- Oversees user accounts, ensuring compliance with platform policies.
- Responsible for handling customer inquiries and providing support.
- Generates reports to monitor platform performance and user engagement.

3. Design and Implementation Constraints

In order to ensure the project's success, we used design and implementation limitations. It can also refer to a tool that enables testers and developers to view and interact with the user interface components of an application.

3.1 Language

User interface Design, usually known as UI Design, is the visual organization of the parts of a website or technological product that a user could interact with. In other words, it is the visual layout of a website. On the other hand, the code that enables a computer program or application to run and cannot be viewed by a user is referred to as the back end. The back end of a computer system is where the majority of data and operating syntax are kept and accessed.

3.1.1 XML

Extensible Markup Language (XML) lets you define and store data in a shareable manner. XML supports information exchange between computer systems such as websites, databases, and third-party applications. Predefined rules make it easy to transmit data as XML files over any network because the recipient can use those rules to read the data accurately and efficiently.

3.1.2 Java

Java is a widely-used programming language for coding web applications. It has been a popular choice among developers for over two decades, with millions of Java applications in use today. Java is a multi-platform, object-oriented, and network-centric language that can be used as a platform in itself. It is a fast, secure, reliable programming language for coding everything from mobile apps and enterprise software to big data applications and server-side technologies.

3.2 Server-Side Technology

When an application is used, server-side development refers to the processes that happen in the background. Databases, scripting, website architecture, backend logic, APIs, and servers are the main topics covered.

3.2.1 PHP

PHP is a server scripting language, and a powerful tool for making dynamic and interactive Web pages. It was one of the first server-side languages that HTML could incorporate, making it simpler to add functionality without having to call outside files for information. One of the best things about PHP is how beginner-friendly it is while still offering many advanced features to professional programmers.

3.3 Database Server

3.3.1 MySQL

MySQL is an open-source relational database management system (RDBMS). A relational database arranges data into one or more tables where it is possible for the data to be connected to one another. Programmers use the SQL language to create, change, and extract data from relational databases and to manage user access to the databases.

4. Requirement Specification

4.1 Functional Requirement

Functional requirements are those that serve as examples for the system's internal operation. It comprises the task that the system should complete, the associated processes, the data that the system should store, and the user interfaces. These are the requirements that the end user specifically demands as basic facilities that the system should offer.

4.1.1 Authentication

FR-1	Users are able to register and authenticate themselves to use the application.		
Description	The system allows new users to register by providing necessary information such as name, email, role and password. It enables registered users to log in and log out securely. Additionally, the system offers a password recovery option for users who forget their passwords.		
Stakeholders	Renters, Owners, Admin	**Priority**	High

4.1.2 Vehicle Browsing

FR-2	Users are able to input search criteria such as location, rental dates and budget to find available vehicles that match their needs.		
Description	The system allows users to search for available vehicles based on criteria such as location, rental dates and budget. It displays information about filtered vehicles including images, specifications, rental rates, and availability.		
Stakeholders	Renters, Owners	**Priority**	High

4.1.3 Booking

FR-3	Users need to enter and confirm their rental details, including pick-up and drop-off dates and locations.		
Description	The system enables users to select a vehicle and make a booking for specific dates and locations. It provides a booking summary for users to review before confirming their booking and sends confirmation notification upon successful booking.		
Stakeholders	Renters	**Priority**	High

4.1.4 Profile Management

FR-4	Users are able to access their profile to view and update personal details such as name, email, password.		
Description	The system allows users to view and update their profile information, including personal details, contact information. Renters are able to view their rental history.		
Stakeholders	Renters, Owners, Admin	**Priority**	Medium

4.1.5 Vehicle Management

FR-5	Owners must be able to add new vehicles to the system.		
Description	The system must allow owners to add, update, and remove vehicle details. It should enable the management of vehicle availability, rental rates, and detailed information, ensuring accurate and up-to-date listings for users.		
Stakeholders	Owners	Priority	High

4.1.6 Notifications

FR-6	The system must notify users and administrators of important events.		
Description	The system must send notifications to users regarding various events and updates related to their rental activities. This includes notifications for booking confirmations, reminders for upcoming rentals, and alerts for changes to rental status.		
Stakeholders	Renters,Owners	**Priority**	Medium

4.2 Data Requirement

4.2.1 User Data

DR-1	Personal details		
Description	The system collects and securely stores detailed user information, including personal details, authentication credentials.		
Stakeholders	Renters,Owners,Admin	**Priority**	High

4.2.2 Vehicle Data

DR-2	Vehicle details, including specifications, availability, and rental rates.		
Description	Detailed information about each vehicle, including specifications, availability, and rental rates. This data ensures accurate listings, availability management, and pricing for users.		
Stakeholders	Renters	**Priority**	High

4.2.3 Booking and Reservation Data

DR-3	Booking
Description	The system stores detailed data related to each booking made by users. This includes information on booking IDs, user IDs, vehicle IDs, rental periods, locations, and status updates.

Stakeholders	Renters.Owners	**Priority**	High

4.2.4 Transaction Data

DR-4	Store detailed transaction data for all financial activities, including advance payment		
Description	The system stores detailed transaction data for all financial activities, including advance payments, total payment. This data must be securely stored and easily accessible for processing, reporting, and auditing purposes.		
Stakeholders	Renters,Owners,Admin	**Priority**	High

5. Requirement Engineering Process

The process of Requirements Engineering (RE) involves identifying and defining the necessary features and functionalities of a software system based on the specific demands or needs of the customer.

5.1 Requirement Elicitation Techniques

Requirements elicitation is the process of gathering and defining the requirements for a software system. The goal of requirements elicitation is to ensure that the software development process is based on a clear and comprehensive understanding of the customer's needs and requirements.

5.1.1 Hold Interviews

We offer discussions that can be conducted one-on-one or in small groups. These sessions are an efficient way to access services, focusing only on the key requirements of the program, thereby saving participants time.

We mainly perform our interview based on some specific criteria.

- Booking process
- Variety of payment
- Pickup and drop-off time and location

5.1.2 Perform Document Analysis

Existing documentation can help to show how systems are currently operating or what they are what I should do. Documents include written information about current programs, business processes, needs specifications, and competitor research. Review once textual analysis can help determine which performance should remain and functionality that isn’t in use. After existing document

In analysis, we found several problems with the existing system.

- Existing platform offer their own car
- Limited Vehicle Availability
- Poor Mobile Experience

5.2 Sample of requirement collection

5.2.1 Requirement Collection - 1

This report summarizes the results of the questionnaire distribution conducted to gather requirements for our system. The objective of the surveys was to identify the key needs and expectations of the stakeholders and to use this information to develop a comprehensive set of requirements for the system.

Methodology - Interview:

The purpose of this interview was to gather requirements from a car owner's perspective for our platform. The aim was to understand the features and functionalities desired by car owners to manage their vehicles and rentals effectively.

Participants:

A total of 3 interviews have been conducted to gather comprehensive project requirements for the car rental platform.

Key Points Discussed:

The following are the key findings from the interviews:

- **Partial Payment System:** Integration of a payment system that allows renters to pay 50% of the rental fee upfront at the time of booking due to toll fee & fuel cost.
- **Open Bidding:** An open bidding feature where car owners can list their vehicles for auction, allowing renters to place bids.

- **Booking Information:** A comprehensive booking form where renters provide pick-up and drop-off locations, start and end date/time of the rental period and number of passengers.
- **Route Planning:** Renters should have the option to provide the number of stoppages during their rental period.
- **Dual Review:** Both car owners and renters should be able to leave reviews and ratings for each other after the rental period.
- **Multiple Cars:** The platform should support car owners who have multiple vehicles.

Limitations:

Responses were collected less than the expectation.

Conclusion:

The car owner emphasized the need for a user-friendly platform that provides comprehensive management tools, secure transactions, and reliable support. These requirements will guide the development of our platform to ensure it meets the needs of car owners effectively.

5.2.2. Requirement Collection - 2

This report summarizes the results of the questionnaire distribution conducted to gather requirements for our system. The objective of the surveys was to identify the key needs and expectations of the stakeholders and to use this information to develop a comprehensive set of requirements for the system.

Methodology - Interview:

The purpose of this interview was to gather requirements from a car renter's perspective for our platform. The aim was to understand the features and functionalities desired by car renters to enhance their rental experience.

Participants:

A total of 2 interviews have been conducted to gather comprehensive project requirements for the car rental platform.

Key Points Discussed:

The following are the key findings from the interviews:

- **Filtering:** Renters want advanced search functionality to filter cars based on various criteria such as location, price range, car type.
- **Mobile App:** A companion mobile app for easy management on the go.
- **Waiting Time:** Allow renters/owners to document and report waiting times at gas stations, traffic jams through the mobile app.

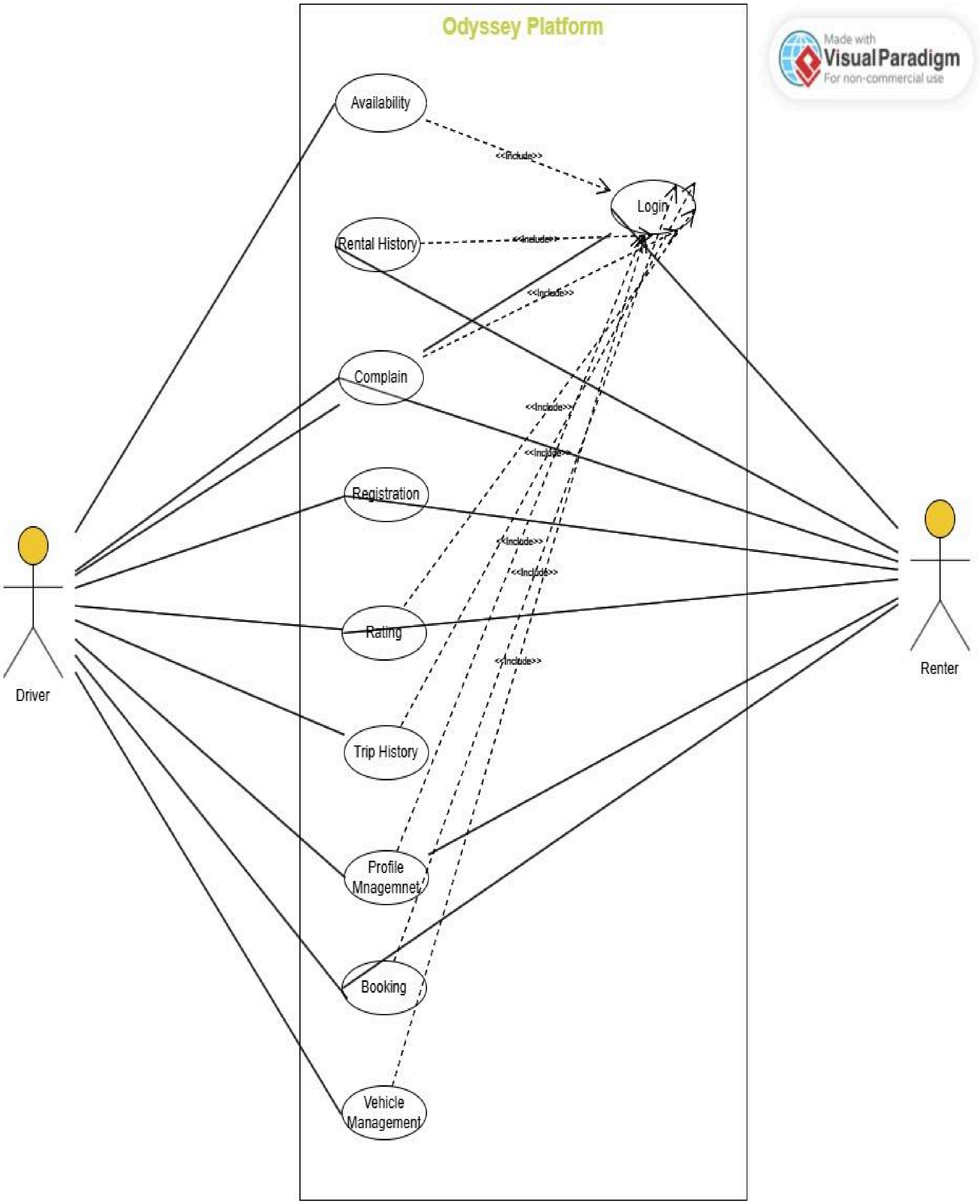
Limitations:

Responses were collected less than the expectation.

Conclusion:

The car renter emphasized the need for a user-friendly platform that provides comprehensive management tools, secure transactions, and reliable support. These requirements will guide the development of our platform to ensure it meets the needs of car owners effectively.

6. Use case Diagram:



7. Use Case Description

7.1 Registration

Use Case No	1	
Use Case	Registration	
Goal	Allow new users (Renter, Owner) to create an account on the Odyssey platform.	
Preconditions	The user must have access to the Odyssey platform via mobile interface.	
Success End Condition	The user is successfully registered and can log in to the platform with their credentials.	
Failed End Condition	The user is not registered due to incomplete information, system error, or invalid inputs	
Primary Actors	Renter, Owner	
Secondary Actors	N/A	
Trigger	The user clicks on the *Register* button	
Main Success Flow	**Step**	**Action**
	1	The user accesses the registration page on the Odyssey platform.
	2	The user clicks the *Register* button.
	3	The user provides necessary information (name, email, password, role).
	4	The system checks for validation and verifies that the email is unique.
	5	The system creates a new user account.

Alternative Flow	**Step**	**Action**
	1	The user provides invalid data.
	2	The system results in an *Error* message.
	3	The user cannot proceed with registration until valid data is provided.
Quality Requirements	**Step**	**Requirement**
	1	The system should provide clear instructions and validation errors for incorrect or incomplete inputs.

7.2 Login

Use Case No	2	
Use Case	Login	
Goal	Allow registered users (Renter, Owner, Admin) to log in to the Odyssey platform.	
Preconditions	The user must have a registered account on the Odyssey platform.	
Success End Condition	The user is successfully logged in and granted access.	
Failed End Condition	The user cannot log in due to incorrect credentials or system error.	
Primary Actors	Renter, Owner, Admin	
Secondary Actors	N/A	
Trigger	The user clicks the *Login* button.	
Main Success Flow	**Step**	**Action**

	1	The user accesses the login page on the Odyssey platform.
	2	The user clicks the *Login* button.
	3	The user provides necessary information (email, password).
	4	The system checks and validates the credentials.
	5	The system authenticates the user and grants access.
Alternative Flow	**Step**	**Action**
	1	The user provides invalid data.
	2	The system results in an *Error* message.
	3	The user cannot proceed with registration until valid data is provided.
Quality Requirements	**Step**	**Requirement**
	1	The system should provide clear instructions and validation errors for incorrect or incomplete inputs.

7.3 Profile Management

Use Case No	3
Use Case	Profile Management
Goal	Allow users (Renter, Owner, Admin) to view and update their profile information on the Odyssey platform.
Preconditions	The user must be logged in to the Odyssey platform.

Success End Condition	The user's profile information is displayed and the user is able to update data.	
Failed End Condition	The user's information does not display due to invalid data or system error.	
Primary Actors	Renter, Owner, Admin	
Secondary Actors	N/A	
Trigger	The user clicks the *Profile* or *Edit Profile* button.	
Main Success Flow	**Step**	**Action**
	1	The user accesses the profile management page.
	2	The user views their current profile information.
	3	The user clicks the *Edit Profile* to update their profile information
	4	The system updates data after successful validation.
Alternative Flow	**Step**	**Action**
	1	The user provides invalid or incomplete data.
	2	The system prints an error message indicating the specific issue.
	4	The user cannot save the changes until valid data is provided.
Quality Requirements	**Step**	**Requirement**
	1	The system should provide clear instructions and validation errors for incorrect or incomplete inputs.

7.4 Availability

Use Case No	4	
Use Case	Availability	
Goal	Allow owners to manage vehicle availability while renters can view and book them.	
Preconditions	The user must be logged in to the Odyssey platform.	
Success End Condition	The user sets and views the availability of vehicles for next 7 days.	
Failed End Condition	The user cannot retrieve the availability information due to system error.	
Primary Actors	Owner	
Secondary Actors	Renter	
Trigger	The user triggers the availability calendar.	
Main Success Flow	**Step**	**Action**
	1	The owner accesses the availability calendar.
	2	The owner selects a specific date to make it unavailable.
	3	The renter selects a specific date to make a booking
Alternative Flow	**Step**	**Action**
	1	The user enter invalid input
	2	The system prints an error message indicating the specific issue
Quality Requirements	**Step**	**Requirement**
	1	The system should provide clear instructions and validation errors for incorrect or incomplete inputs.

	2	The availability information should be accurate and up-to-date.

7.5 Rental History

Use Case No	5	
Use Case	Rental History	
Goal	Allow renters to view their past rental history.	
Preconditions	The user must be logged in to the Odyssey platform.	
Success End Condition	The user views their rental history with detailed information.	
Failed End Condition	The user cannot retrieve rental history due to system error or no past Renters found.	
Primary Actors	Renter	
Secondary Actors	N/A	
Trigger	The user navigates to their profile and selects the *Renter History* option.	
Main Success Flow	**Step**	**Action**
	1	The user logs in to the Odyssey platform and accesses their profile.
	2	The user selects the *Rental History* option from their profile menu.
	3	The system retrieves the user's past rental records from the database.
	4	The system displays the rental history, including details such as vehicle rented, rental period, cost, and status.
Alternative Flow	**Step**	**Action**

	1	The user has no past rentals.
	2	The system displays an error message and suggests the user try again later.
Quality Requirements	**Step**	**Requirement**
	1	The system should provide a clear and easy-to-navigate profile interface.
	2	The displayed rental history must be accurate and include all relevant details.

7.6 Trip History

Use Case No	6	
Use Case	Trip History	
Goal	Allow Owner to view their past trip transaction and trip details.	
Preconditions	The user must be logged in to the Odyssey platform.	
Success End Condition	The user successfully views their trip history details.	
Failed End Condition	The user cannot view trip history due to system error or incomplete data	
Primary Actors	Owner	
Secondary Actors	N/A	
Trigger	The user clicks the *Trip History* button.	
Main Success Flow	**Step**	**Action**
	1	The user logs in to the Odyssey platform and accesses their profile.
	2	The user selects the *Trip History* option from their profile menu.

	3	The system retrieves the user's past trip records from the database.
	4	The system displays the trip history, including details such as vehicle rented, rental period, cost, and status.
Alternative Flow	**Step**	**Action**
	1	The user has no past trip.
	2	The system displays an error message and suggests the user try again later.
Quality Requirements	**Step**	**Requirement**
	1	The system should retrieve trip history data.
	2	The displayed ride history details must be accurate and up-to-date.

7.7 Notification

Use Case No	7
Use Case	Notification
Goal	Notify users about important events.
Preconditions	The user must be logged in to the Odyssey platform.
Success End Condition	The user successfully receives and views the notifications.
Failed End Condition	The user does not receive notifications due to system error or incorrect settings.
Primary Actors	Renter, Owner
Secondary Actors	N/A
Trigger	An event occurs that requires a notification to be sent

Main Success Flow	**Step**	**Action**
	1	An event occurs that triggers notification
	2	The system generates a notification based on the event details.
	3	The system sends the notification to the user
	4	The user receives and views the notification.
Alternative Flow	**Step**	**Action**
	1	System error occurs while sending the notification
Quality Requirements	**Step**	**Requirement**
	1	Notifications should be generated promptly upon triggering events.
	2	The notification content must be clear, accurate, and include all relevant information.

7.8 Complain

Use Case No	8
Use Case	Complaint
Goal	Allow users (renter, owner) to submit complaints regarding their rental experience or issues on the Odyssey platform.
Preconditions	The user must be logged in to the Odyssey platform.
Success End Condition	The user successfully submits a complaint.
Failed End Condition	The user cannot submit a complaint due to incomplete information, system error, or invalid inputs.

Primary Actors	Renter, Owner	
Secondary Actors	Admin	
Trigger	The user navigates the complaint box from the profile.	
Main Success Flow	**Step**	**Action**
	1	The user accesses the complaint section on the Odyssey platform.
	2	The user provides necessary information (complaint category, description, relevant ride details).
	3	The system checks for validation and completeness of the complaint information.
Alternative Flow	**Step**	**Action**
	1	The user provides incomplete or invalid data.
	2	The system prints an "Error" message indicating what needs to be corrected.
	3	The user cannot proceed with complaint submission until valid data is provided
Quality Requirements	**Step**	**Requirement**
	1	The system should provide clear instructions and validation errors for incorrect or incomplete inputs.

7.9 Rating

Use Case No	9
Use Case	Rating

Goal	Allow users (Renters, Owners) to leave rating for their rental experiences on the Odyssey platform.	
Preconditions	The user must be logged in to the Odyssey platform and have completed a rental transaction.	
Success End Condition	The user successfully submits a rating.	
Failed End Condition	The user cannot submit a rating due to incomplete information, system error, or invalid inputs.	
Primary Actors	Renter, Owner	
Secondary Actors	N/A	
Trigger	The user complete a trip and click *Review* button	
Main Success Flow	**Step**	**Action**
	1	The user completes a trip & clear full payment.
	2	The user provides necessary information and the system validates it.
	3	The system submits the review and provides a confirmation message to the user.
Alternative Flow	**Step**	**Action**
	1	The user provides incomplete or invalid data.
	2	The system prints an error message indicating what needs to be corrected.
	3	The user cannot proceed with review submission until valid data is provided and the system encounters an error while processing the review.

Quality Requirements	Step	Requirement
	1	The system should provide clear instructions and validation errors for incorrect or incomplete inputs.
	2	Review information must be validated accurately

7.10 Filter

Use Case No	10	
Use Case	Filter	
Goal	Allow users to filter vehicle search results on the Odyssey platform.	
Preconditions	The user must have accessed the vehicle search page on the Odyssey platform.	
Success End Condition	The user successfully applies filters to refine search results.	
Failed End Condition	The user cannot apply filters due to system error or invalid inputs.	
Primary Actors	Renter	
Secondary Actors	N/A	
Trigger	The user provides filter criteria and clicks the apply button.	
Main Success Flow	**Step**	**Action**
	1	The user accesses the vehicle search page on the Odyssey platform.
	2	The user selects filter criteria such as vehicle type, location, price range, etc.
	3	The user clicks the *Apply Filters* button.

	4	The system applies the selected filters to the search results.
	5	The system displays refined search results based on the applied filters.
Alternative Flow	**Step**	**Action**
	1	The system encounters an error while applying filters.
	2	The system displays an error message and prompts the user to retry later.
Quality Requirements	**Step**	**Requirement**
	1	The system should provide clear and intuitive filter options for users to select.
	2	The system should accurately apply selected filters to search results.

7.11 Vehicle Management

Use Case No	11
Use Case	Vehicle Management
Goal	Allow owners to manage their vehicles on the Odyssey platform.
Preconditions	The user must be logged in to the Odyssey platform
Success End Condition	The owner successfully manages their vehicles on the platform.
Failed End Condition	The owner cannot manage their vehicles due to system error or invalid inputs.
Primary Actors	Owner
Secondary Actors	N/A

Trigger	The user clicks the *Manage Vehicles* button.	
Main Success Flow	**Step**	**Action**
	1	The user accesses the vehicle management page on the Odyssey platform.
	2	The system retrieves the list of vehicles associated with the user's account.
	3	The user performs the desired management actions
Alternative Flow	**Step**	**Action**
	1	The system encounters an error while retrieving the list of vehicles.
	2	The system displays an error message and prompts the user to retry later
Quality Requirements	**Step**	**Requirement**
	1	The system should provide an intuitive and user-friendly interface for managing vehicles.
	2	The system should accurately retrieve and display the list of vehicles associated with the user's account.

7.12 Booking

Use Case No	12
Use Case	Booking

Goal	Allow renters to book vehicles on the Odyssey platform.	
Preconditions	The user must be logged in to the Odyssey platform.	
Success End Condition	The user successfully books a vehicle.	
Failed End Condition	The user cannot book a vehicle due to system error, vehicle unavailability, or invalid inputs.	
Primary Actors	Renter	
Secondary Actors	Owner	
Trigger	The user clicks the *Book* button	
Main Success Flow	**Step**	**Action**
	1	The user accesses the booking page on the Odyssey platform.
	2	The user searches for available vehicles based on location, date, and other preferences.
	3	The user selects a vehicle from the list of available options.
	4	The user confirms the booking by providing necessary details
	5	The system verifies availability and processes the booking.
Alternative Flow	**Step**	**Action**
	1	The selected vehicle is no longer available for booking.
	2	The system suggests alternative vehicles.
	3	The user provides invalid booking details.

Quality Requirements	**Step**	**Requirement**
	1	The system should provide an intuitive and user-friendly interface for booking vehicles.
	2	The system should accurately display available vehicles based on user preferences.

7.13 Vehicle Verification

Use Case No	13	
Use Case	Vehicle Verification	
Goal	Allow admin to verify vehicles uploaded by owners on the Odyssey platform.	
Preconditions	The admin must be logged into the dashboard.	
Success End Condition	The vehicle is successfully verified and displayed as available for booking.	
Failed End Condition	The vehicle is not verified due to incomplete information, system error, or invalid inputs.	
Primary Actors	Admin	
Secondary Actors	Owner	
Trigger	A vehicle is uploaded by an owner.	
Main Success Flow	**Step**	**Action**
	1	Authorized admin navigates the vehicle verification section.
	2	The admin panel reviews the vehicle details and uploaded images.
	3	The system updates the vehicle status from unverified to verified.

Alternative Flow	**Step**	**Action**
	1	The admin panel identifies discrepancies or missing information in the vehicle details or images.
	2	The admin panel suspects fraudulent activity or invalid information.
Quality Requirements	**Step**	**Requirement**
	1	The vehicle details and images must be displayed clearly and accurately for review.
	2	The system should promptly update the vehicle status upon verification completion.

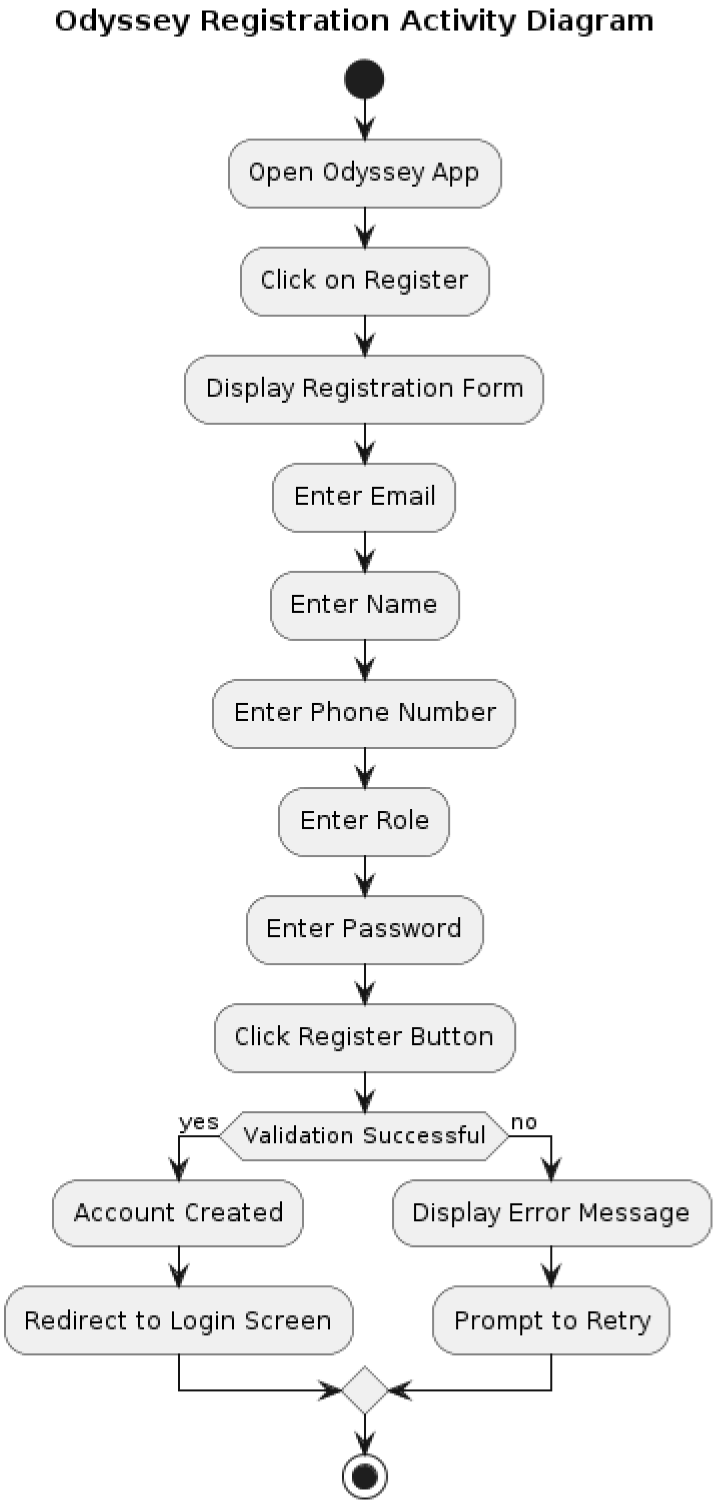
7.15 Suspend User

Use Case No	15	
Use Case	Suspend User	
Goal	Allow admin to suspend user accounts on the Odyssey platform.	
Preconditions	The admin must be logged into the dashboard.	
Success End Condition	The user account is successfully suspended in the system.	
Failed End Condition	The user account is not suspended due to system error or invalid inputs.	
Primary Actors	Admin	
Secondary Actors	N/A	
Trigger	Admin marked a user suspend	
Main Success Flow	**Step**	**Action**

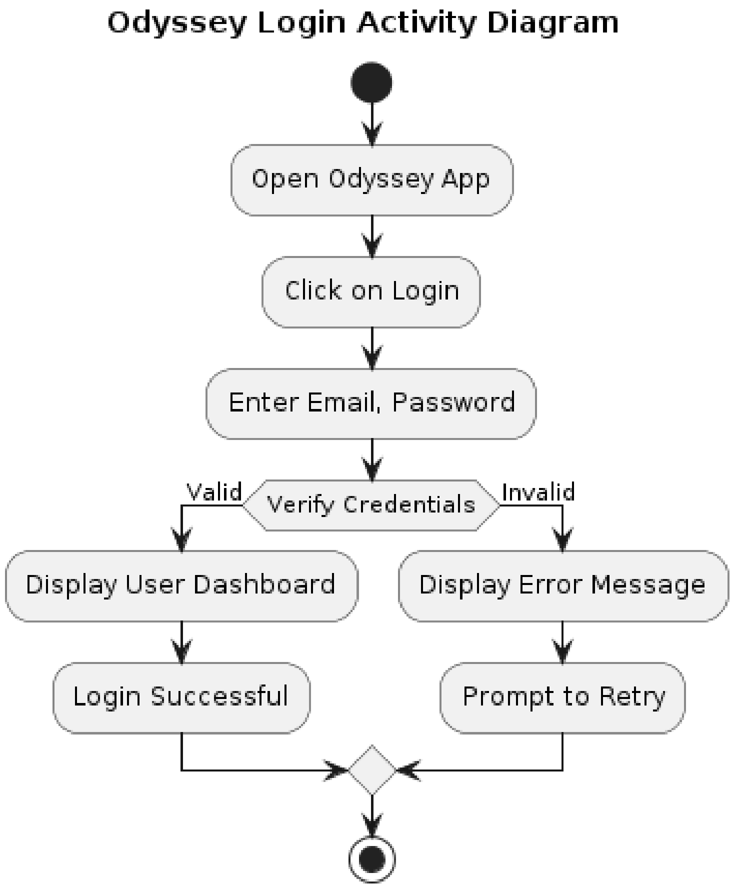
	1	Admin navigate to user management section
	2	Admin marks a user suspend and provides validity of suspension.
	3	The system verifies the user's identity and checks if the account exists.
	4	The system suspends the user account.
Alternative Flow	**Step**	**Action**
	1	The user account does not exist.
	2	The system displays an error message indicating that the user account could not be found.
	3	The system displays an error message indicating that the data provided is invalid or incomplete.
Quality Requirements	**Step**	**Requirement**
	1	The user management interface should be intuitive and easy to navigate for the admin panel user.
	2	User suspension operations should be executed efficiently, with the account status updated immediately in the system.

8. Activity Diagram

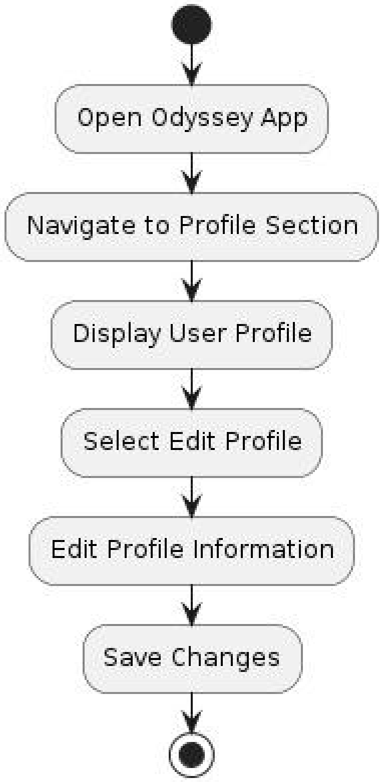
8.1 Registration



8.2 Login

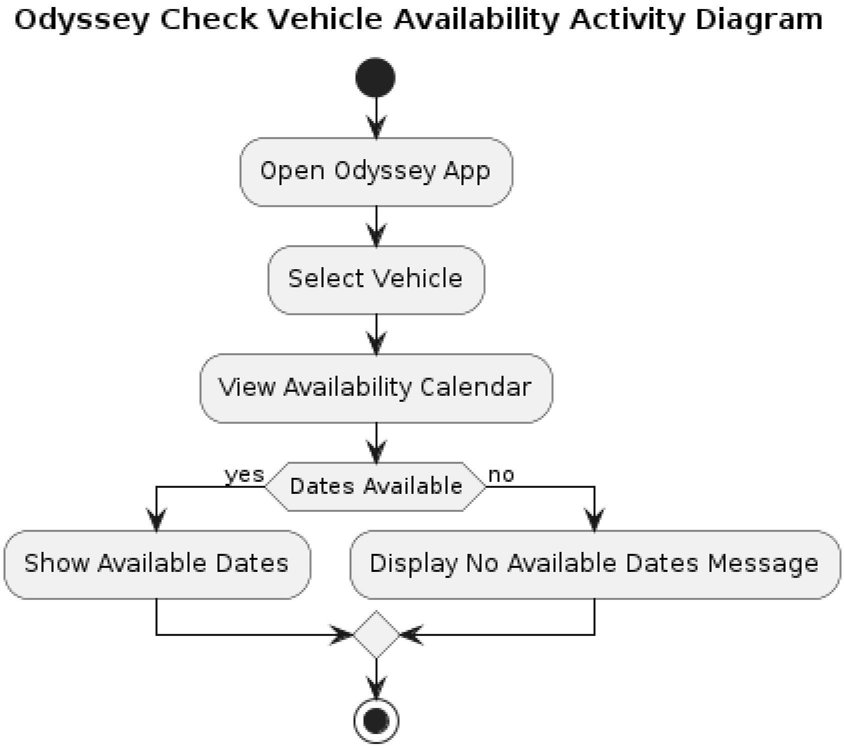


8.3 Profile Management

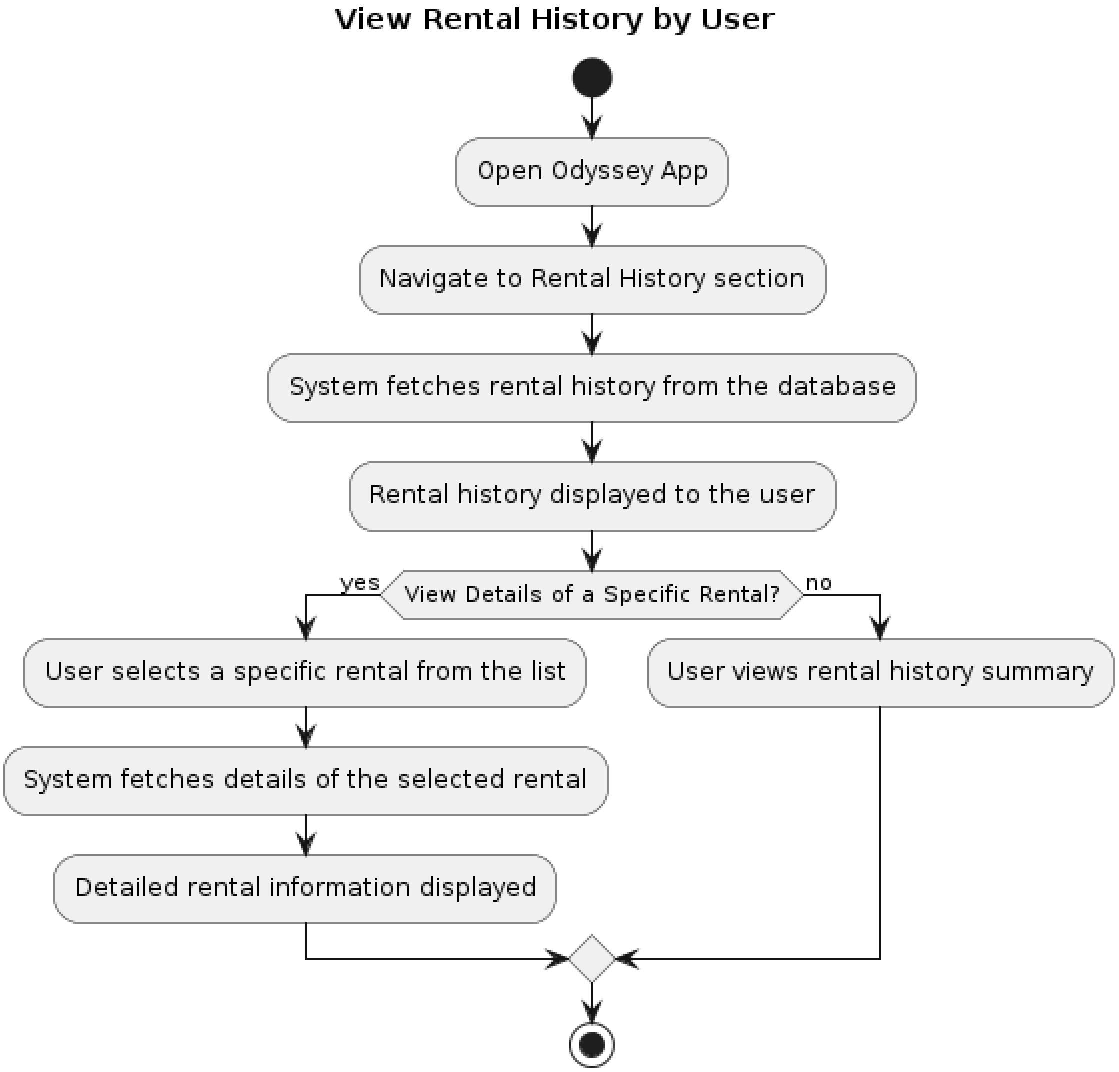


Odyssey Profile Management Activity Diagram

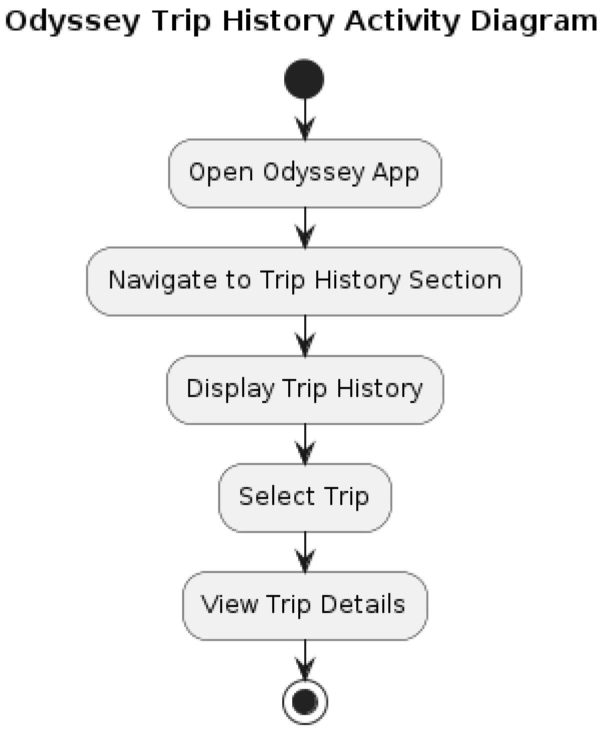
8.4 Availability


Odyssey Check Vehicle Availability Activity Diagram
Open Odyssey App
Select Vehicle
View Availability Calendar
yes
Dates Available
no
Show Available Dates
Display No Available Dates Message

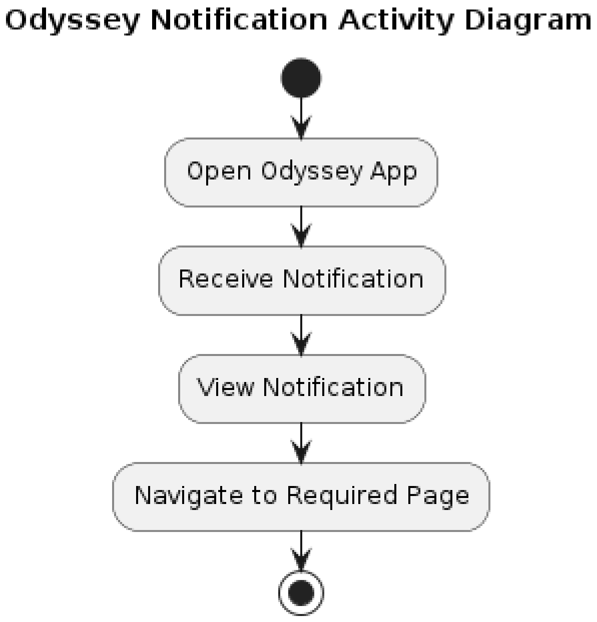
8.5 Rental History


View Rental History by User
Open Odyssey App
Navigate to Rental History section
System fetches rental history from the database
Rental history displayed to the user
View Details of a Specific Rental?
yes
no
User selects a specific rental from the list
User views rental history summary
System fetches details of the selected rental
Detailed rental information displayed

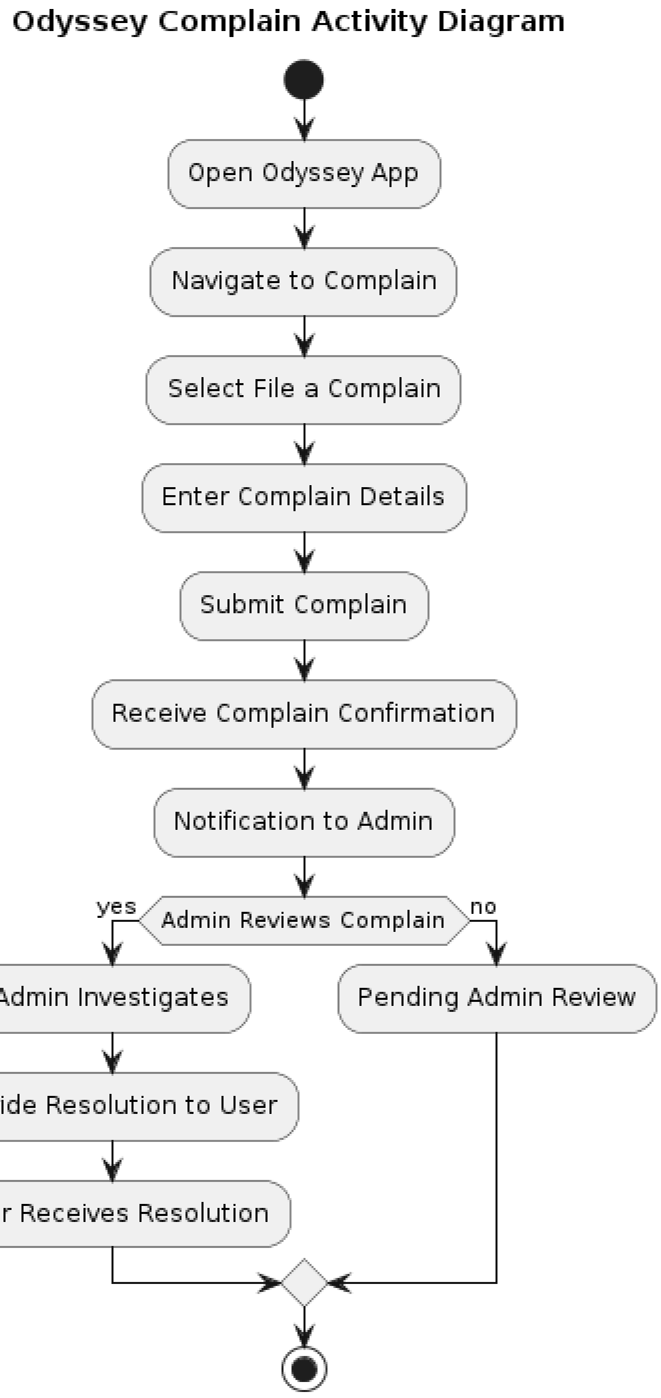
8.6 Trip History


Odyssey Trip History Activity Diagram
Open Odyssey App
Navigate to Trip History Section
Display Trip History
Select Trip
View Trip Details

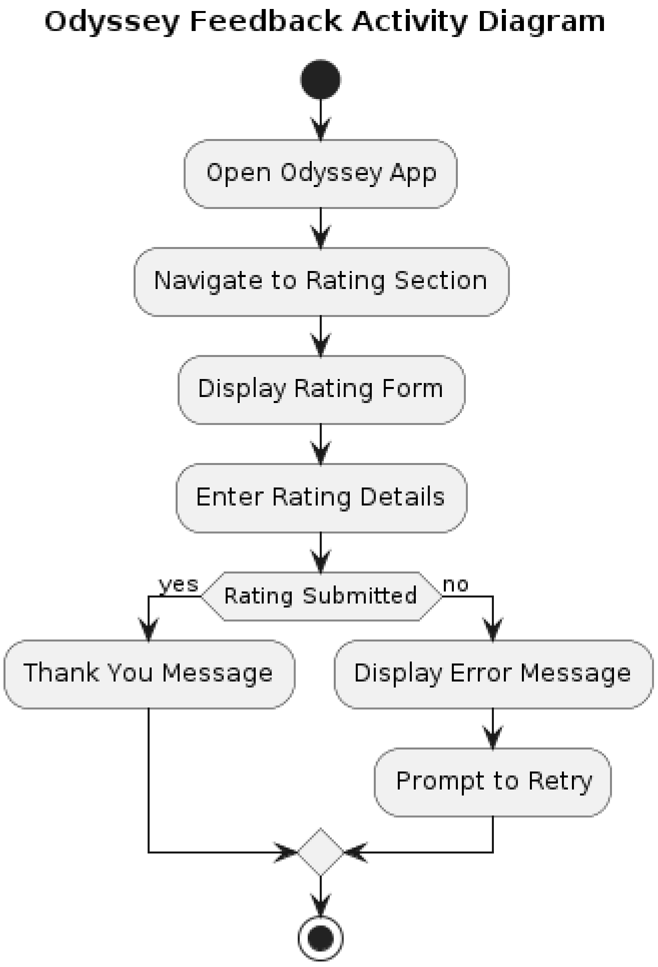
8.7 Notification


Odyssey Notification Activity Diagram
Open Odyssey App
Receive Notification
View Notification
Navigate to Required Page

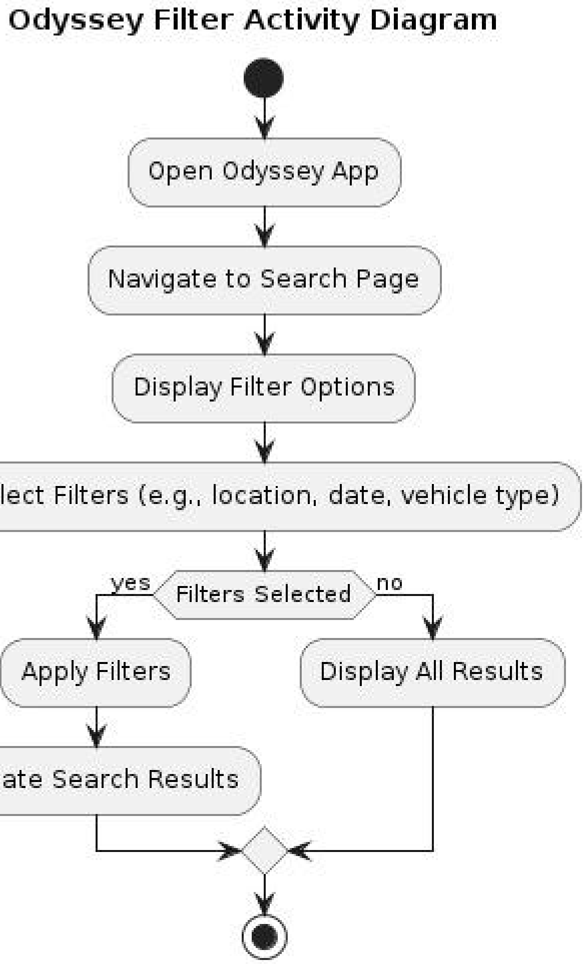
8.8 Complain


Odyssey Complain Activity Diagram
Open Odyssey App
Navigate to Complain
Select File a Complain
Enter Complain Details
Submit Complain
Receive Complain Confirmation
Notification to Admin
Admin Reviews Complain
yes
no
Admin Investigates
Pending Admin Review
Provide Resolution to User
User Receives Resolution

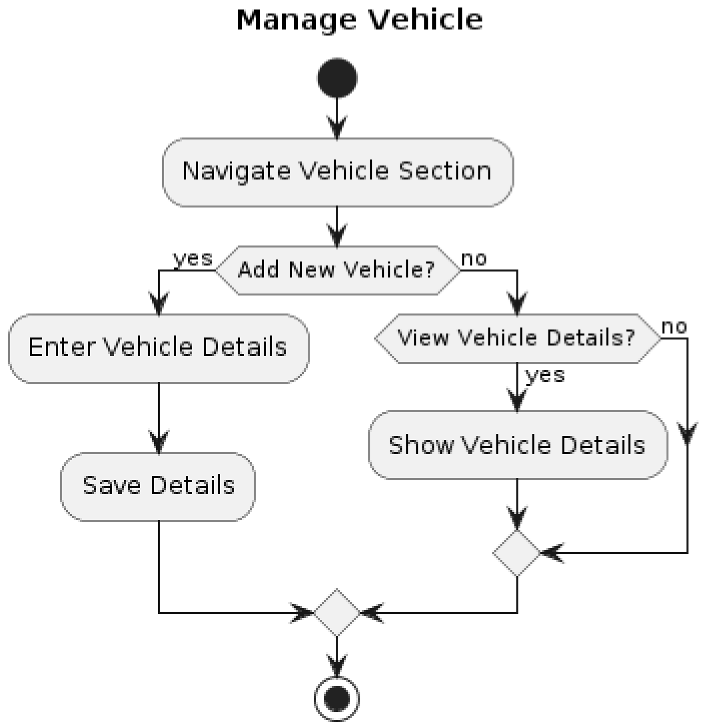
8.9 Rating


Odyssey Feedback Activity Diagram
Open Odyssey App
Navigate to Rating Section
Display Rating Form
Enter Rating Details
Rating Submitted
yes
no
Thank You Message
Display Error Message
Prompt to Retry

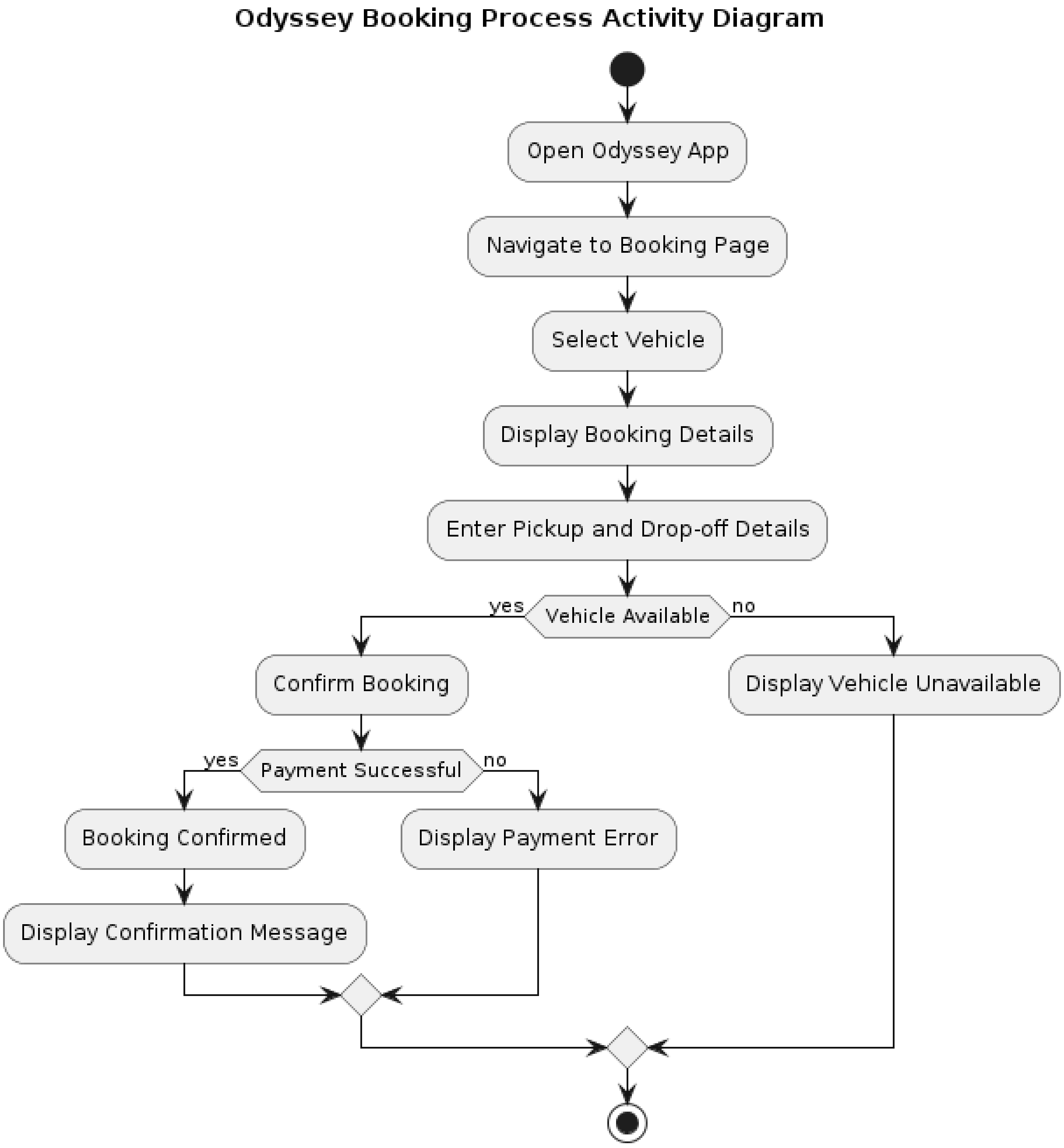
8.10 Filter


Odyssey Filter Activity Diagram
Open Odyssey App
Navigate to Search Page
Display Filter Options
Select Filters (e.g., location, date, vehicle type)
yes
Filters Selected
no
Apply Filters
Display All Results
Update Search Results

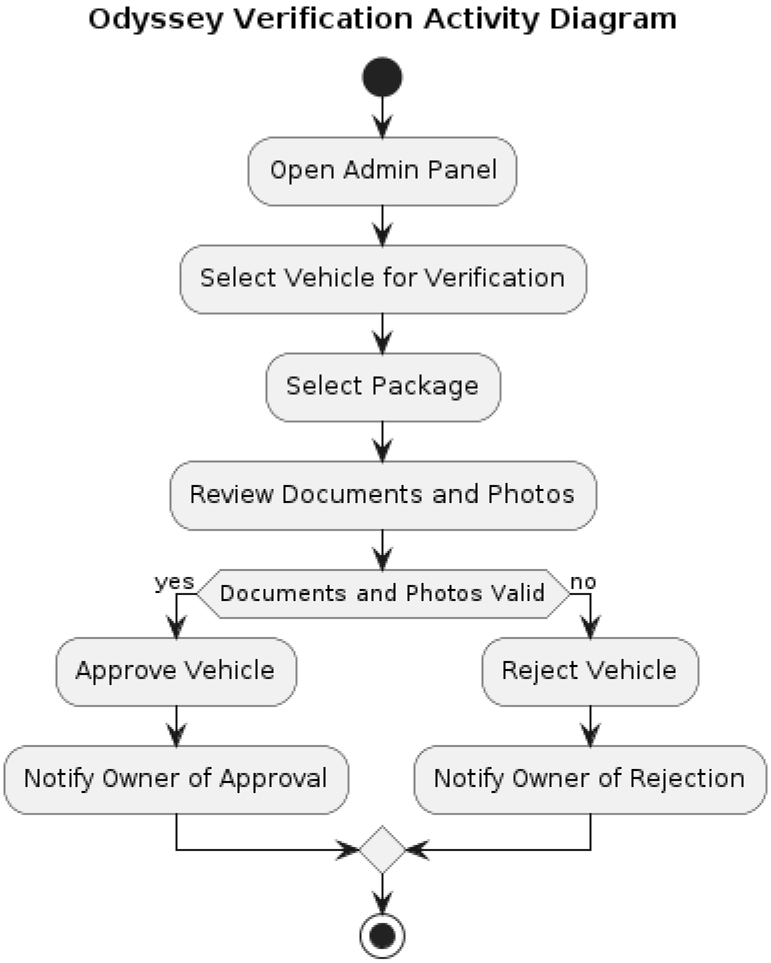
8.11 Vehicle Management



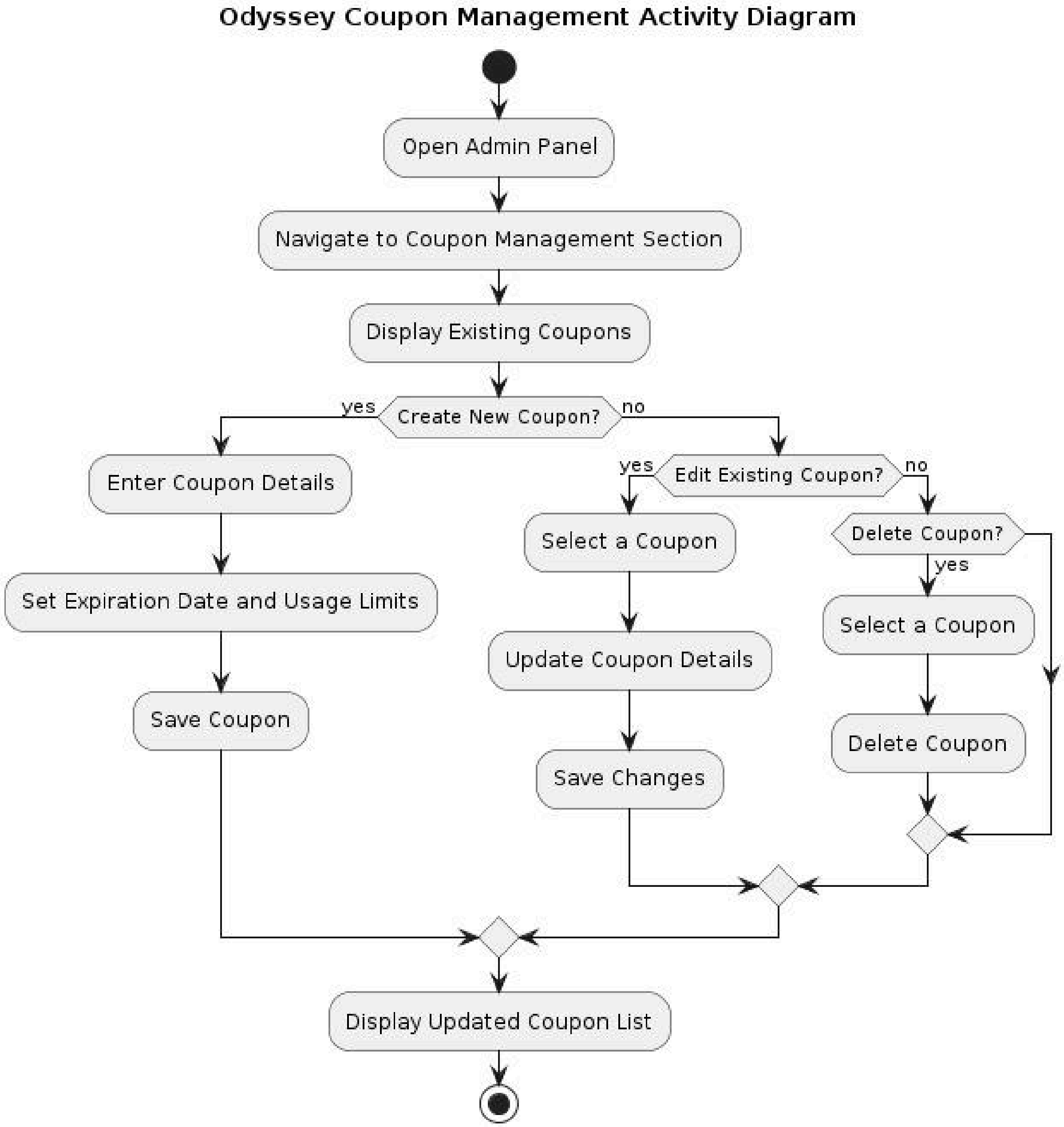
8.12 Booking


Odyssey Booking Process Activity Diagram
Open Odyssey App
Navigate to Booking Page
Select Vehicle
Display Booking Details
Enter Pickup and Drop-off Details
yes
Vehicle Available
no
Confirm Booking
Display Vehicle Unavailable
yes
Payment Successful
no
Booking Confirmed
Display Payment Error
Display Confirmation Message

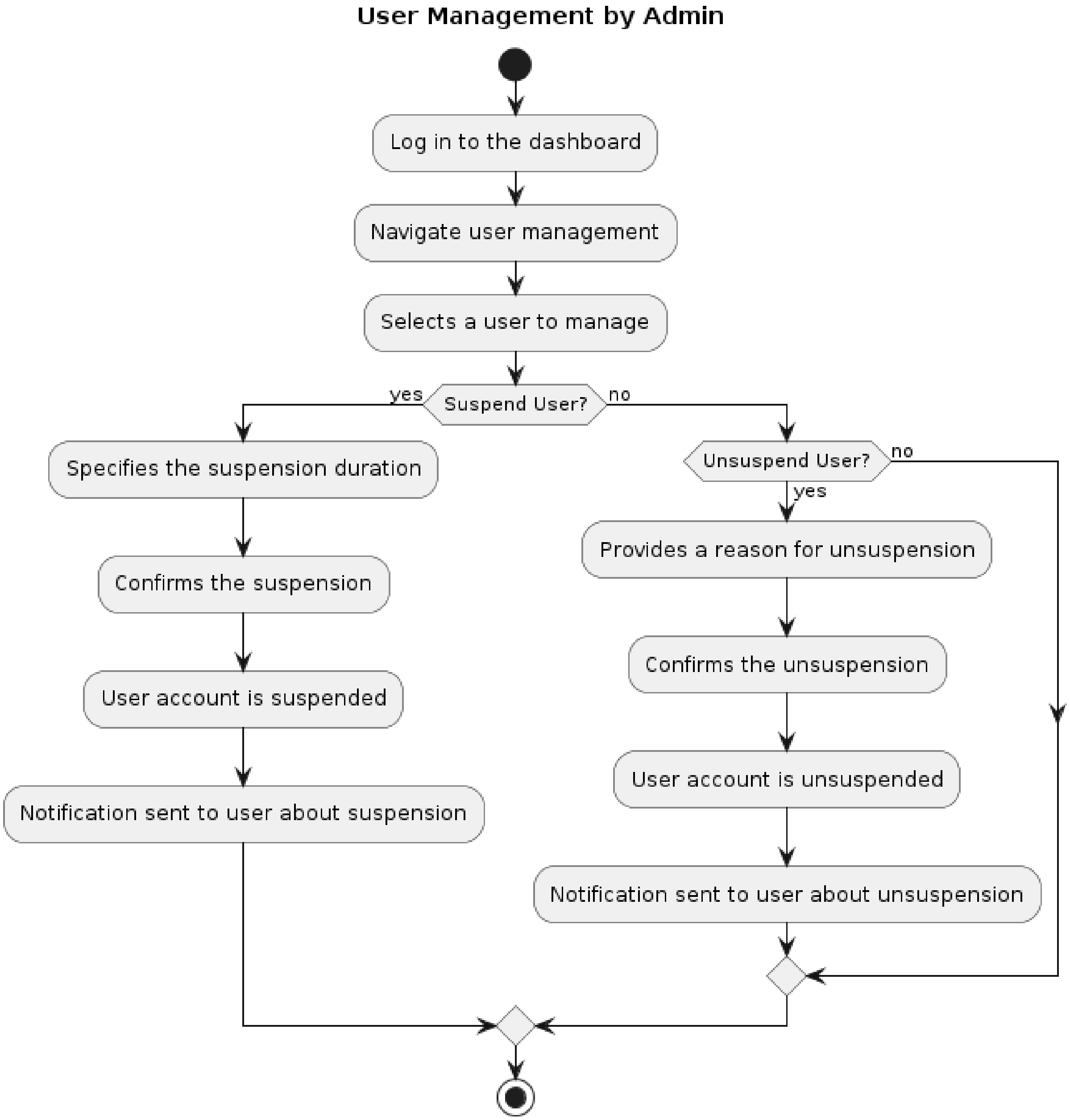
8.13 Vehicle Verification


Odyssey Verification Activity Diagram
Open Admin Panel
Select Vehicle for Verification
Select Package
Review Documents and Photos
Documents and Photos Valid
yes
no
Approve Vehicle
Reject Vehicle
Notify Owner of Approval
Notify Owner of Rejection

8.14 Coupon Management

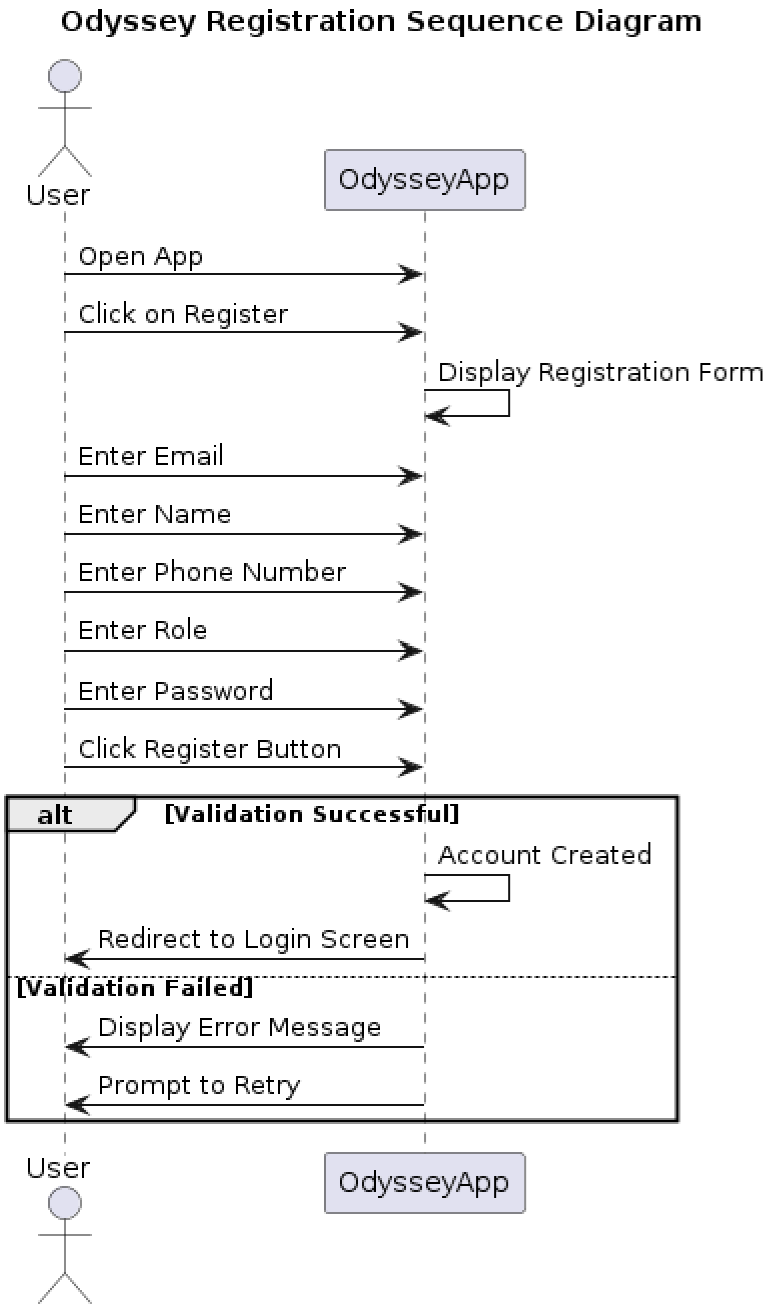


8.15 Suspend User

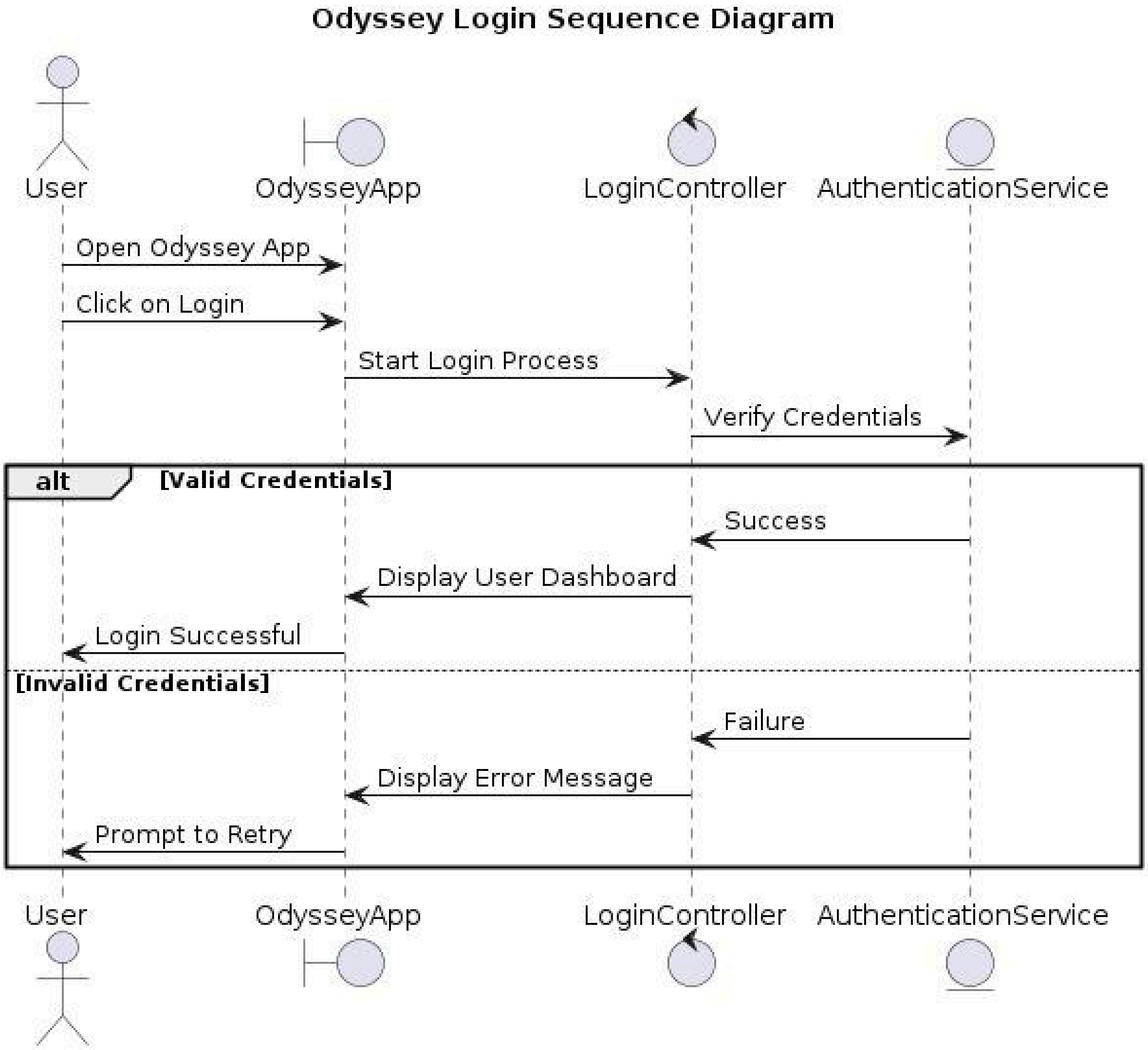

User Management by Admin
Log in to the dashboard
Navigate user management
Selects a user to manage
Suspend User?
yes
no
Specifies the suspension duration
Confirms the suspension
User account is suspended
Notification sent to user about suspension
Unsuspend User?
no
yes
Provides a reason for unsuspension
Confirms the unsuspension
User account is unsuspended
Notification sent to user about unsuspension

9. Sequence Diagram

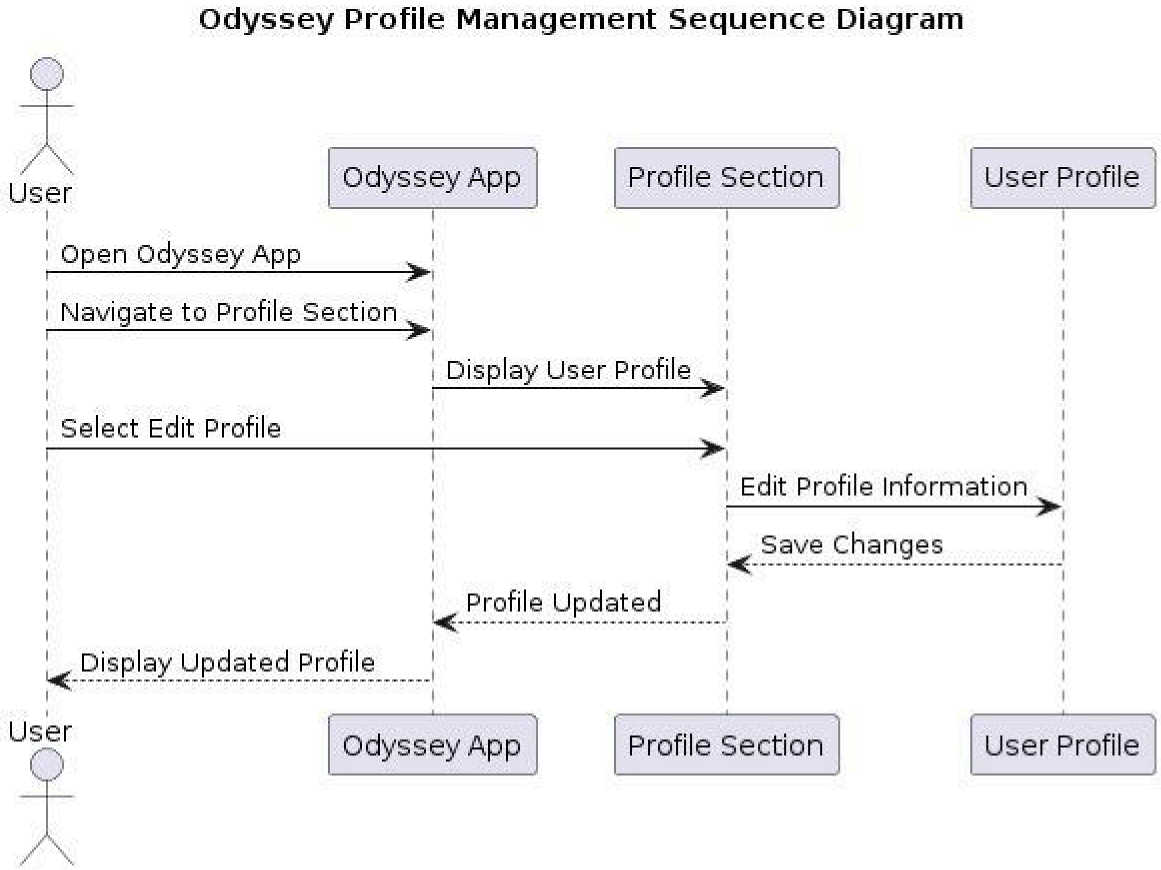
9.1 Registration



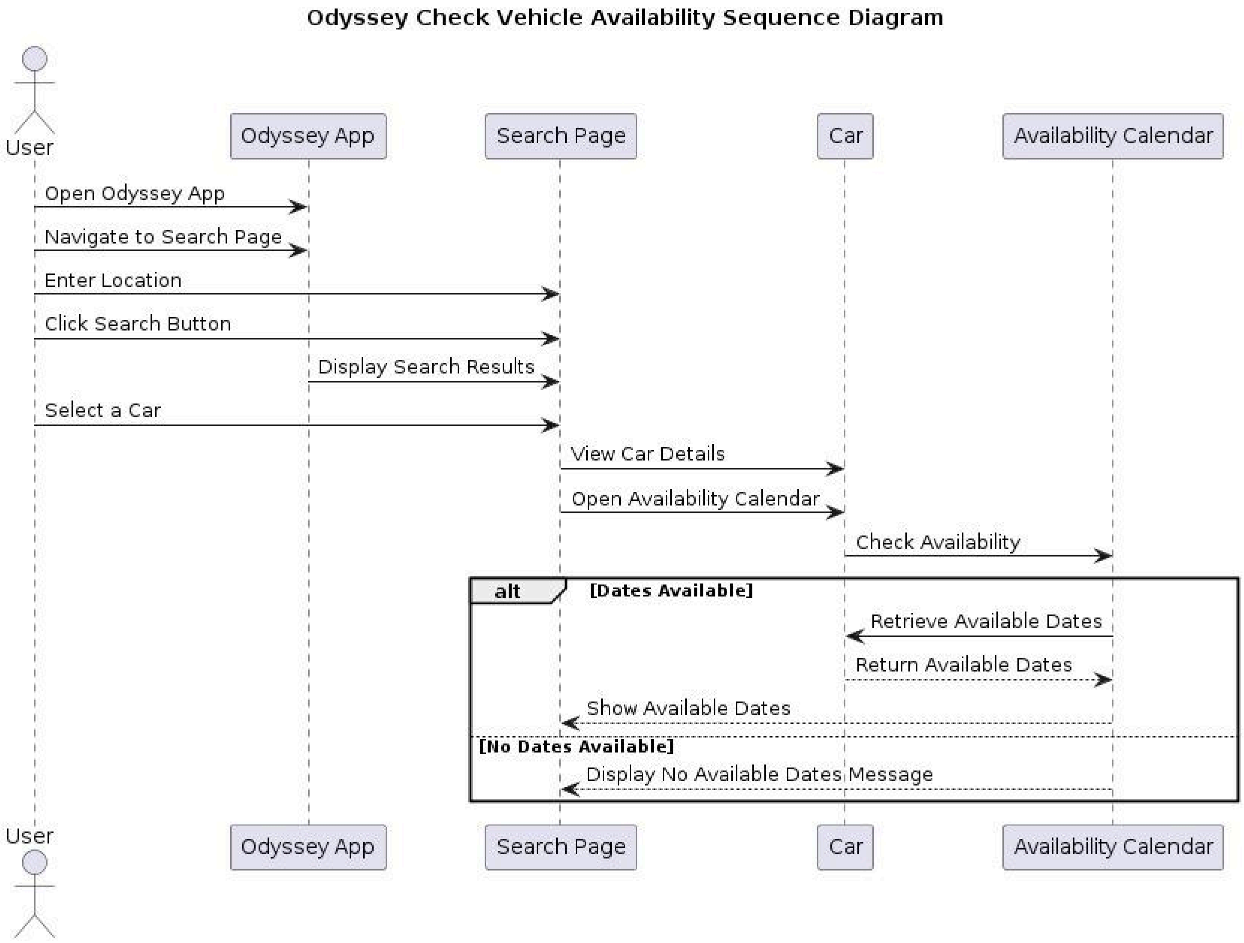
9.2 Login


Odyssey Login Sequence Diagram
User
OdysseyApp
LoginController
AuthenticationService
Open Odyssey App
Click on Login
Start Login Process
Verify Credentials
alt
[Valid Credentials]
Success
Display User Dashboard
Login Successful
[Invalid Credentials]
Failure
Display Error Message
Prompt to Retry
User
OdysseyApp
LoginController
AuthenticationService

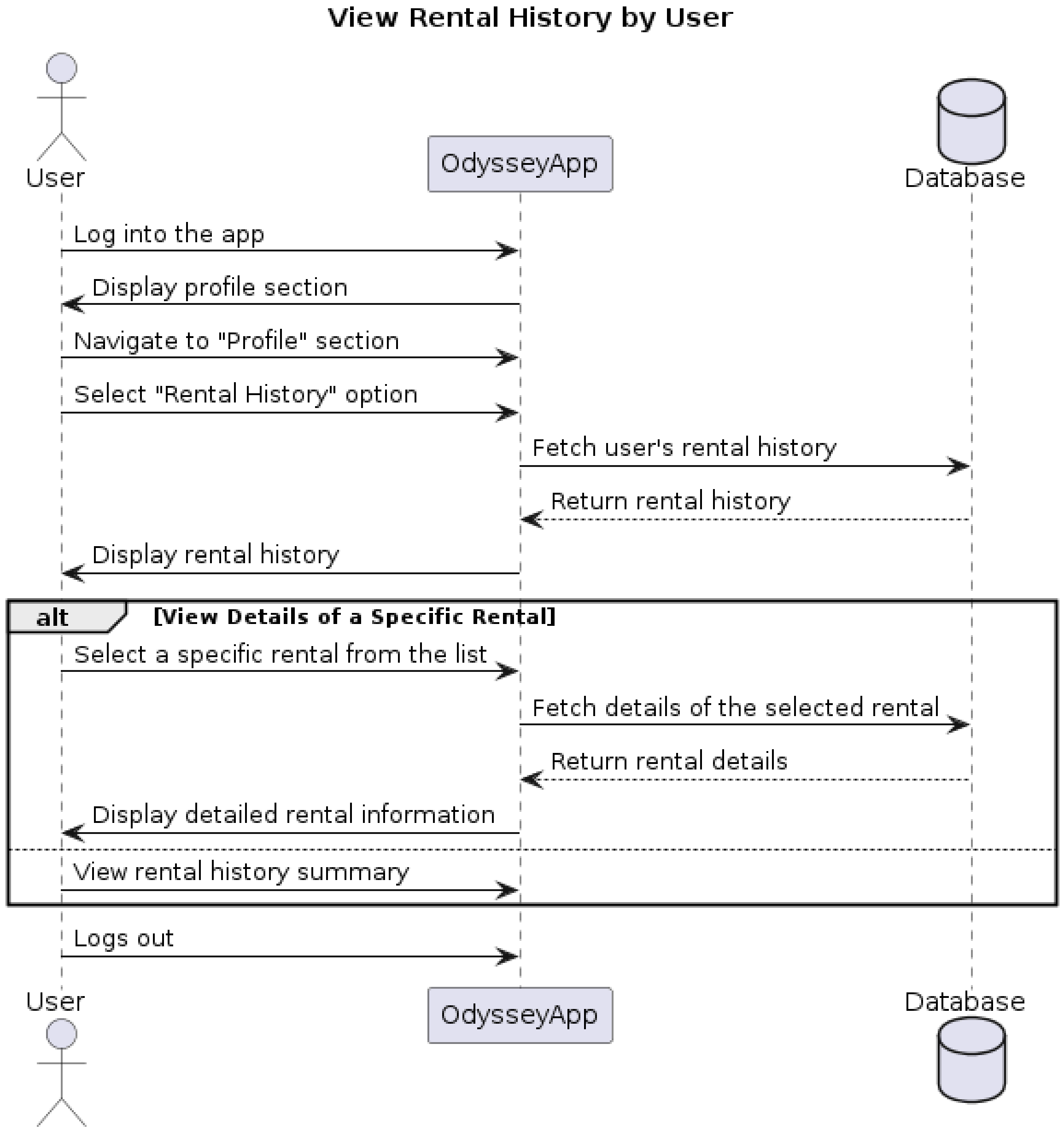
9.3 Profile Management



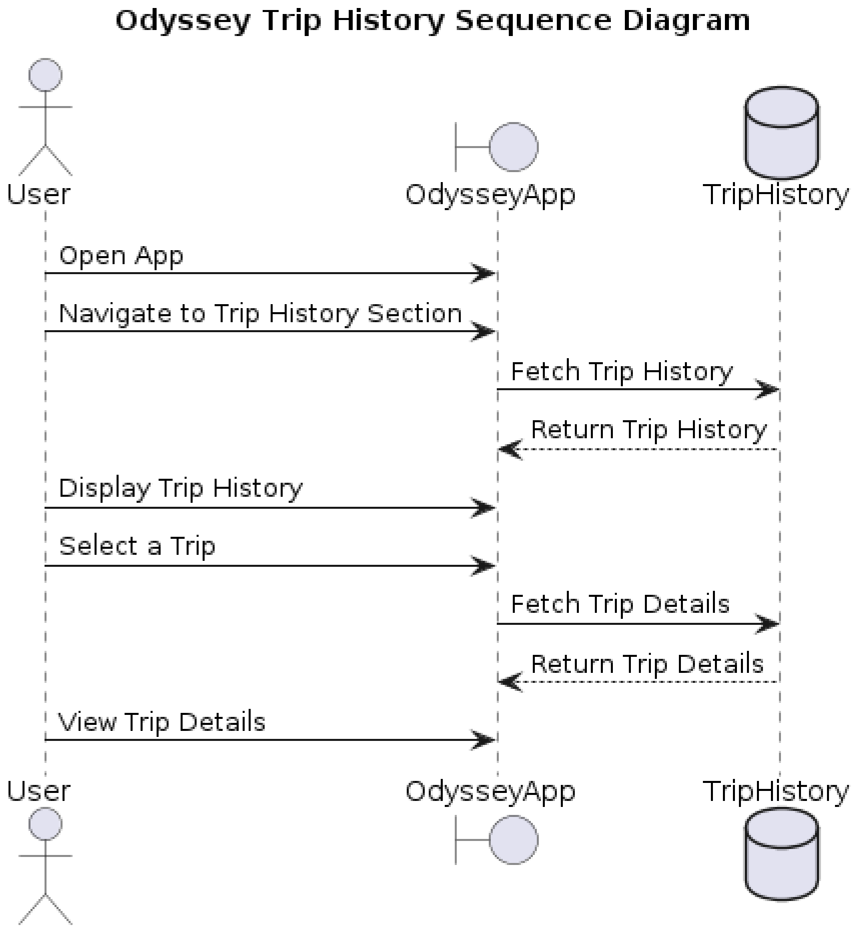
9.4 Availability



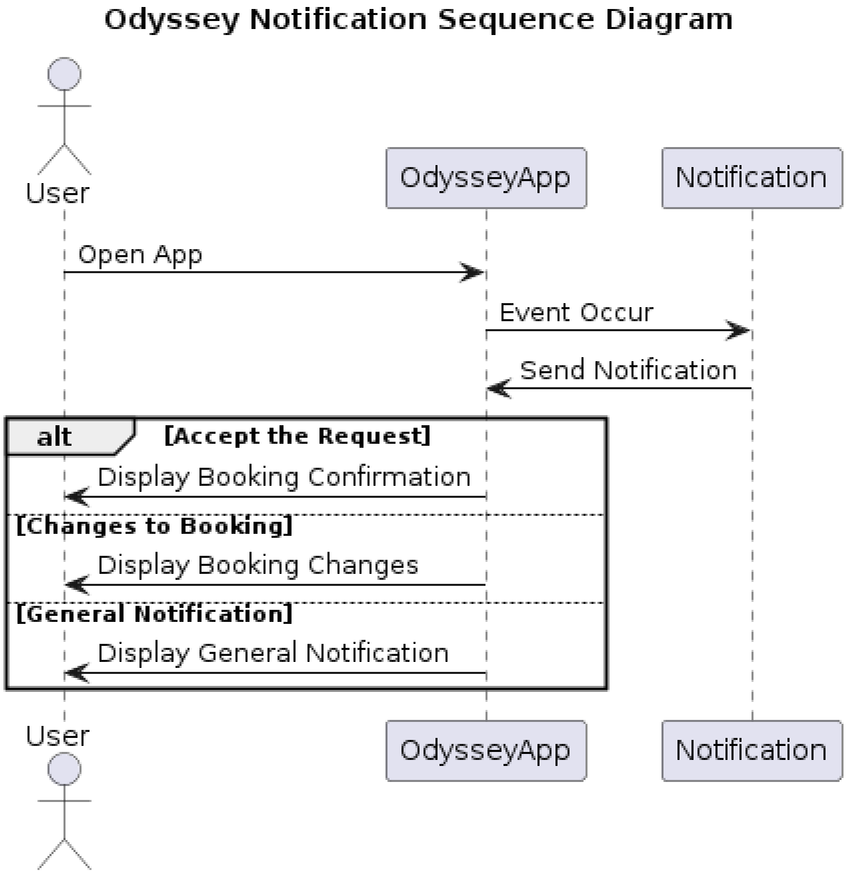
9.5 Rental History


View Rental History by User
User
OdysseyApp
Database
Log into the app
Display profile section
Navigate to "Profile" section
Select "Rental History" option
Fetch user's rental history
Return rental history
Display rental history
alt
[View Details of a Specific Rental]
Select a specific rental from the list
Fetch details of the selected rental
Return rental details
Display detailed rental information
View rental history summary
Logs out
User
OdysseyApp
Database

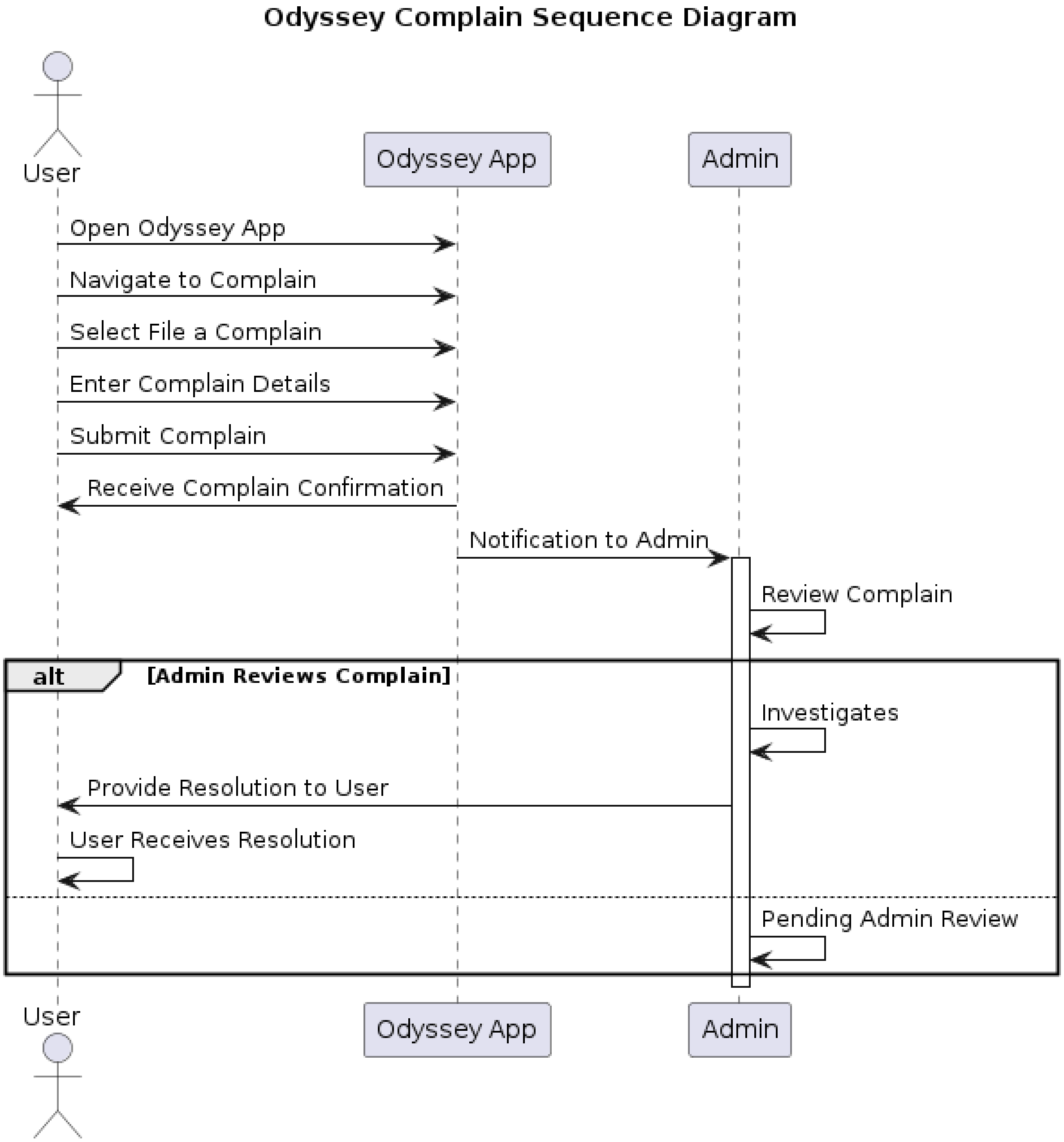
9.6 Trip History



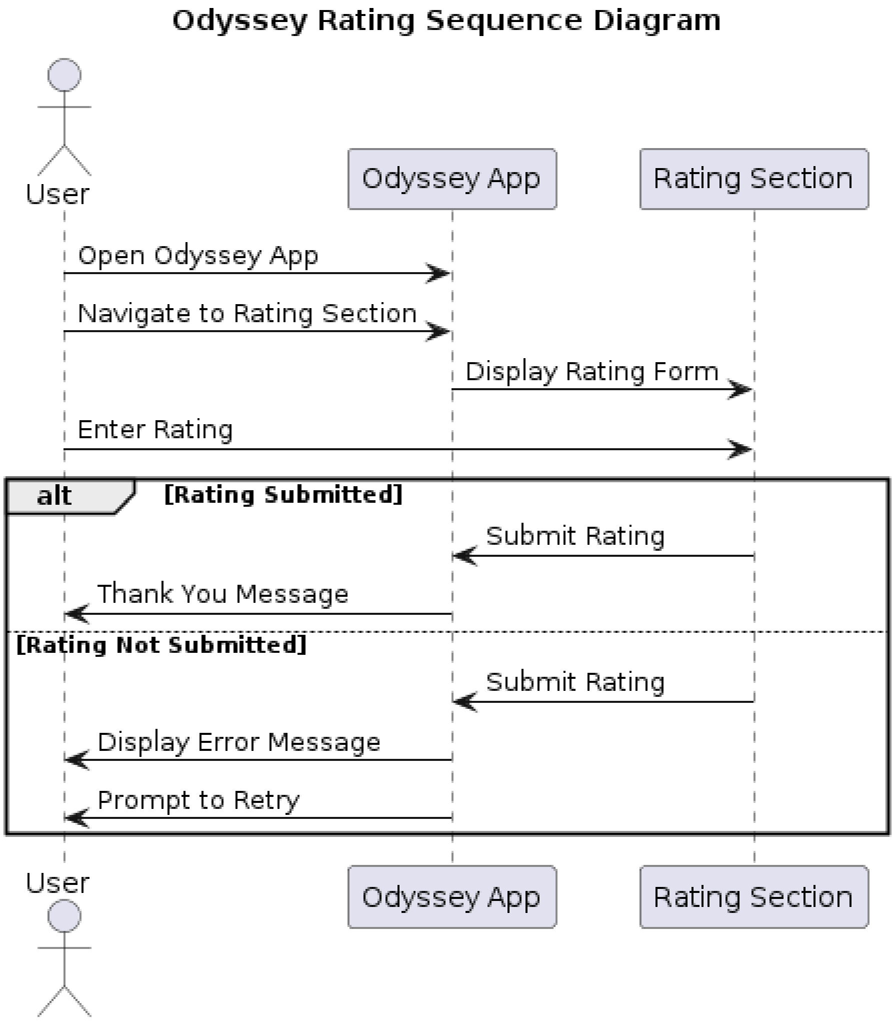
9.7 Notification



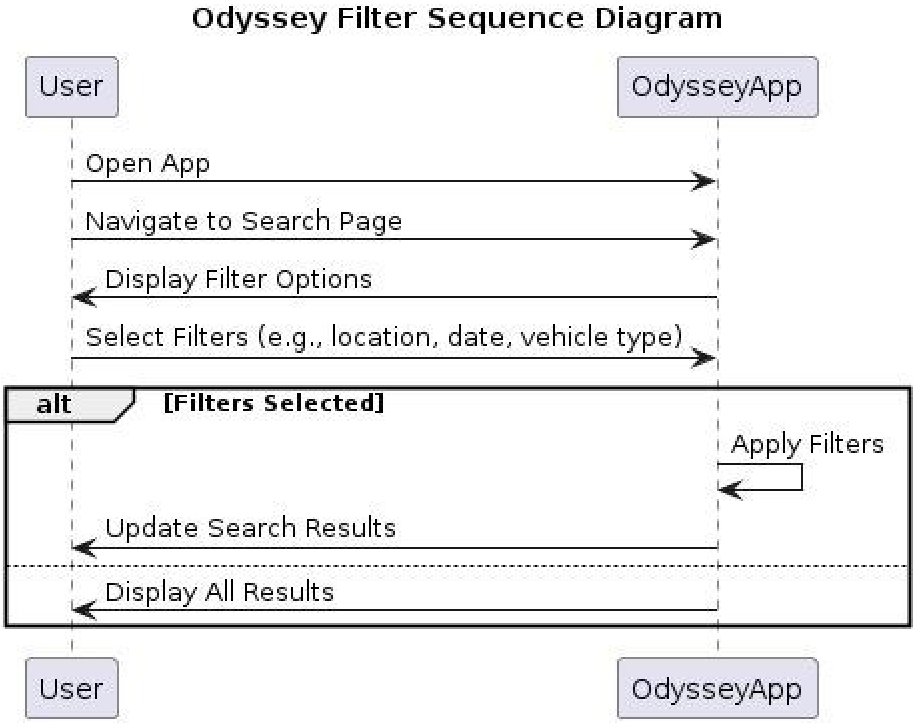
9.8 Complain


Odyssey Complain Sequence Diagram
User
Odyssey App
Admin
Open Odyssey App
Navigate to Complain
Select File a Complain
Enter Complain Details
Submit Complain
Receive Complain Confirmation
Notification to Admin
Review Complain
alt
[Admin Reviews Complain]
Investigates
Provide Resolution to User
User Receives Resolution
Pending Admin Review
User
Odyssey App
Admin

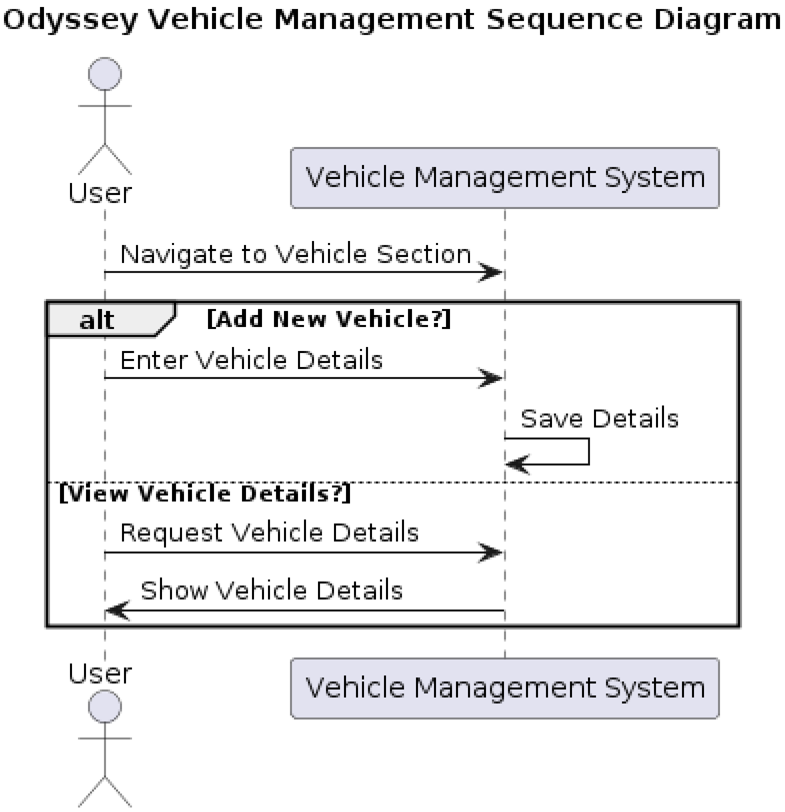
9.9 Rating


Odyssey Rating Sequence Diagram
User
Odyssey App
Rating Section
Open Odyssey App
Navigate to Rating Section
Display Rating Form
Enter Rating
alt
[Rating Submitted]
Submit Rating
Thank You Message
[Rating Not Submitted]
Submit Rating
Display Error Message
Prompt to Retry
User
Odyssey App
Rating Section

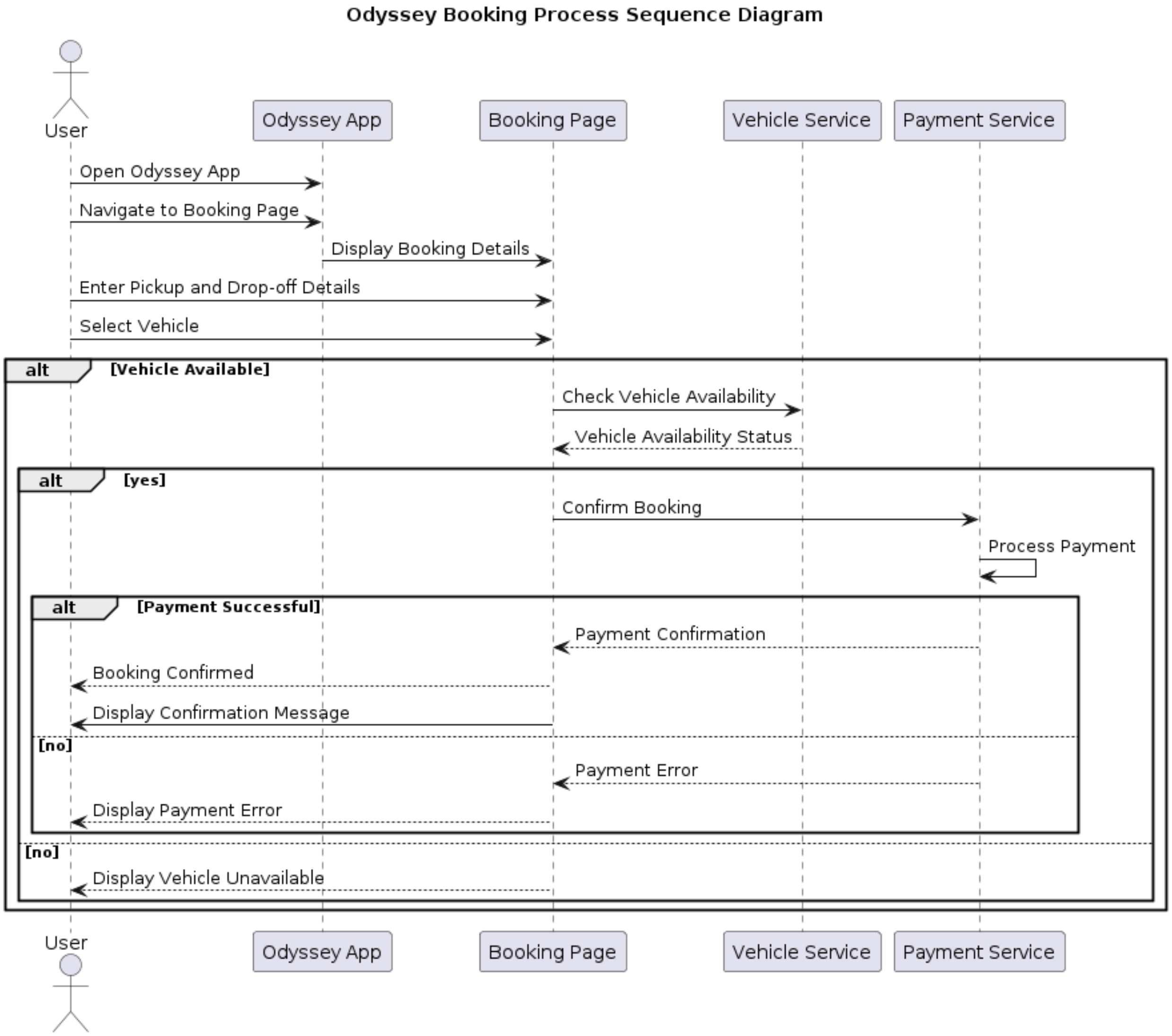
9.10 Filter


Odyssey Filter Sequence Diagram
User
OdysseyApp
Open App
Navigate to Search Page
Display Filter Options
Select Filters (e.g., location, date, vehicle type)
alt
[Filters Selected]
Apply Filters
Update Search Results
Display All Results
User
OdysseyApp

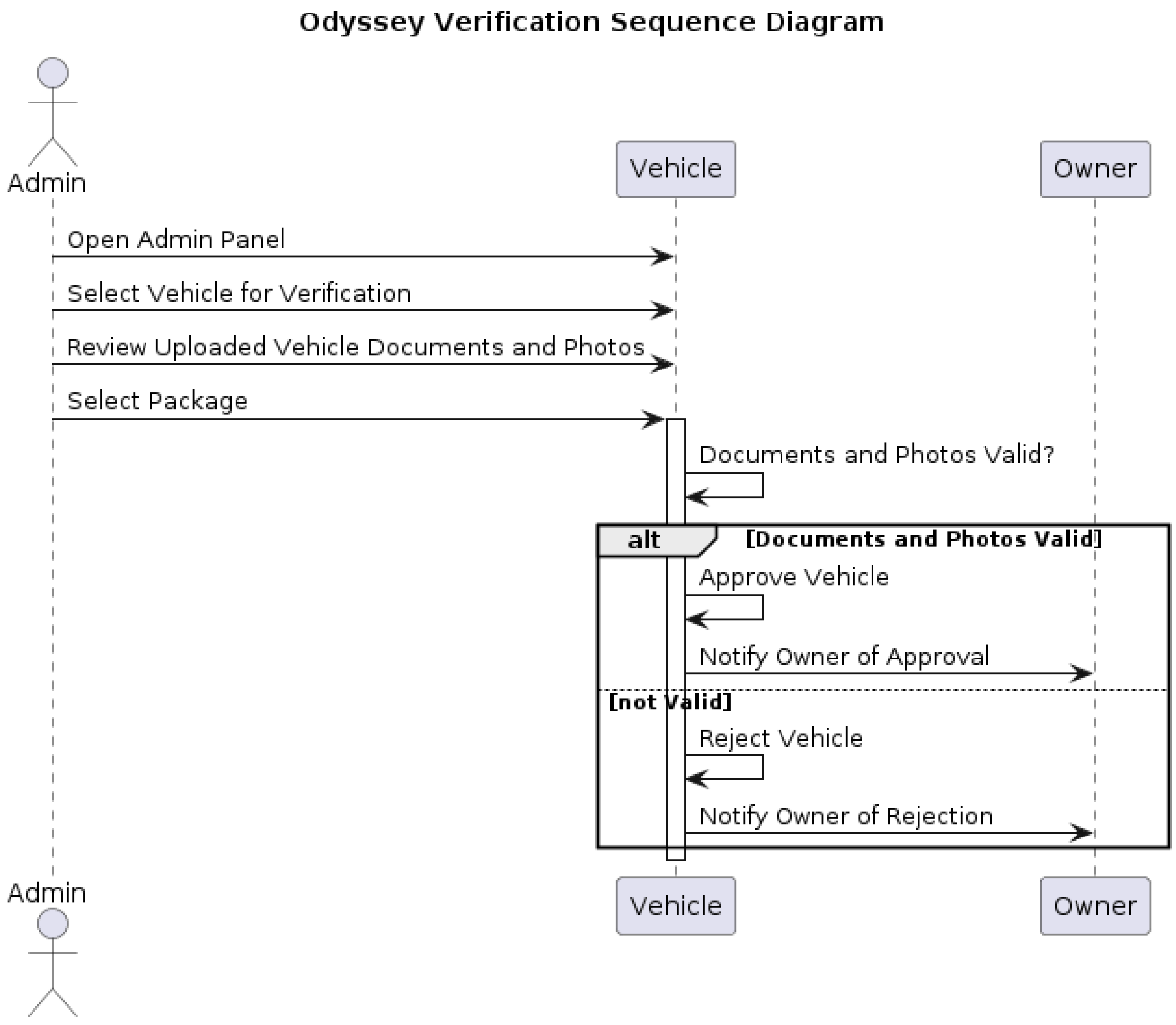
9.11 Vehicle Management



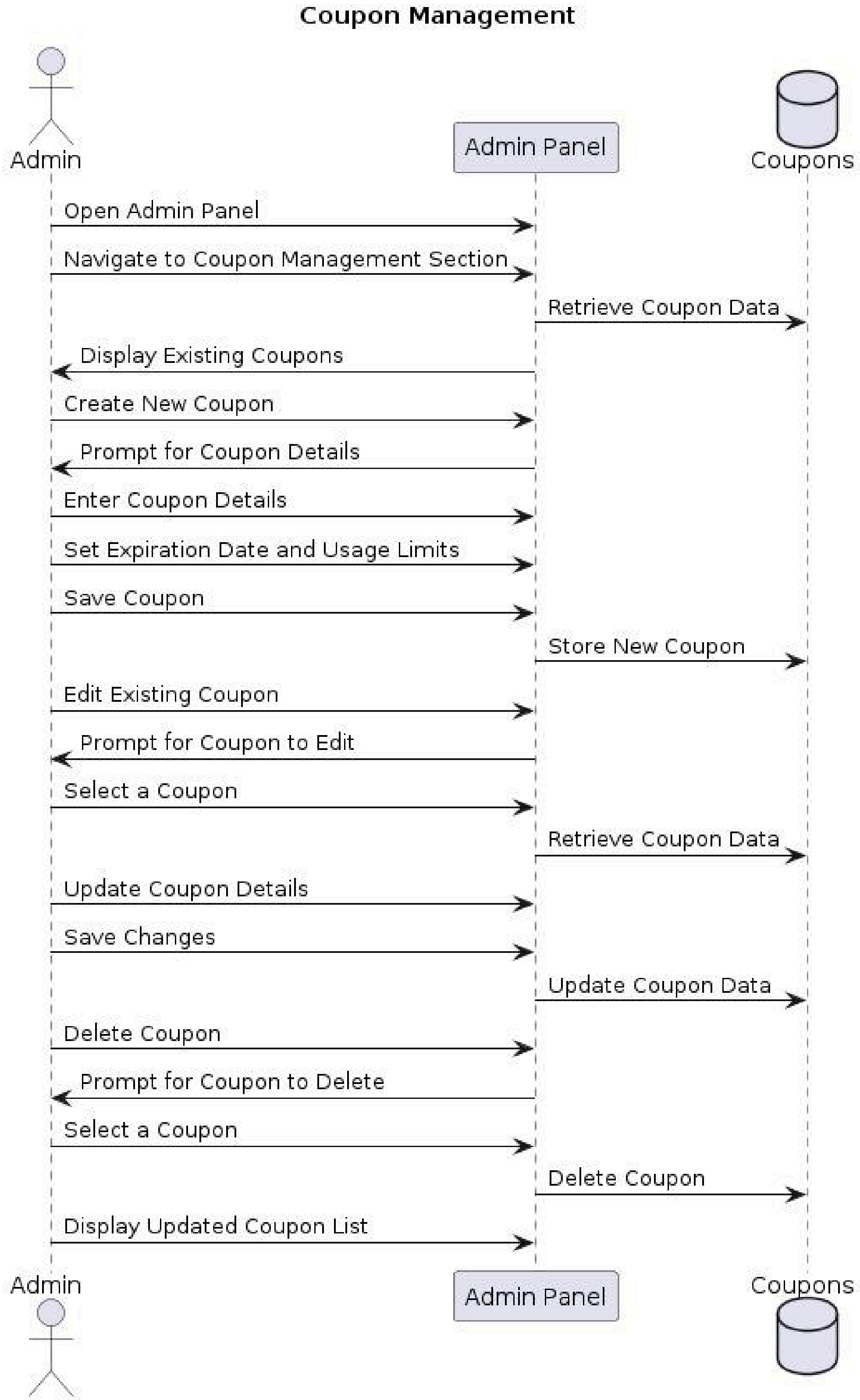
9.12 Booking


Odyssey Booking Process Sequence Diagram
User
Odyssey App
Booking Page
Vehicle Service
Payment Service
Open Odyssey App
Navigate to Booking Page
Display Booking Details
Enter Pickup and Drop-off Details
Select Vehicle
alt [Vehicle Available]
Check Vehicle Availability
Vehicle Availability Status
alt [yes]
Confirm Booking
Process Payment
alt [Payment Successful]
Payment Confirmation
Booking Confirmed
Display Confirmation Message
[no]
Payment Error
Display Payment Error
[no]
Display Vehicle Unavailable
User
Odyssey App
Booking Page
Vehicle Service
Payment Service

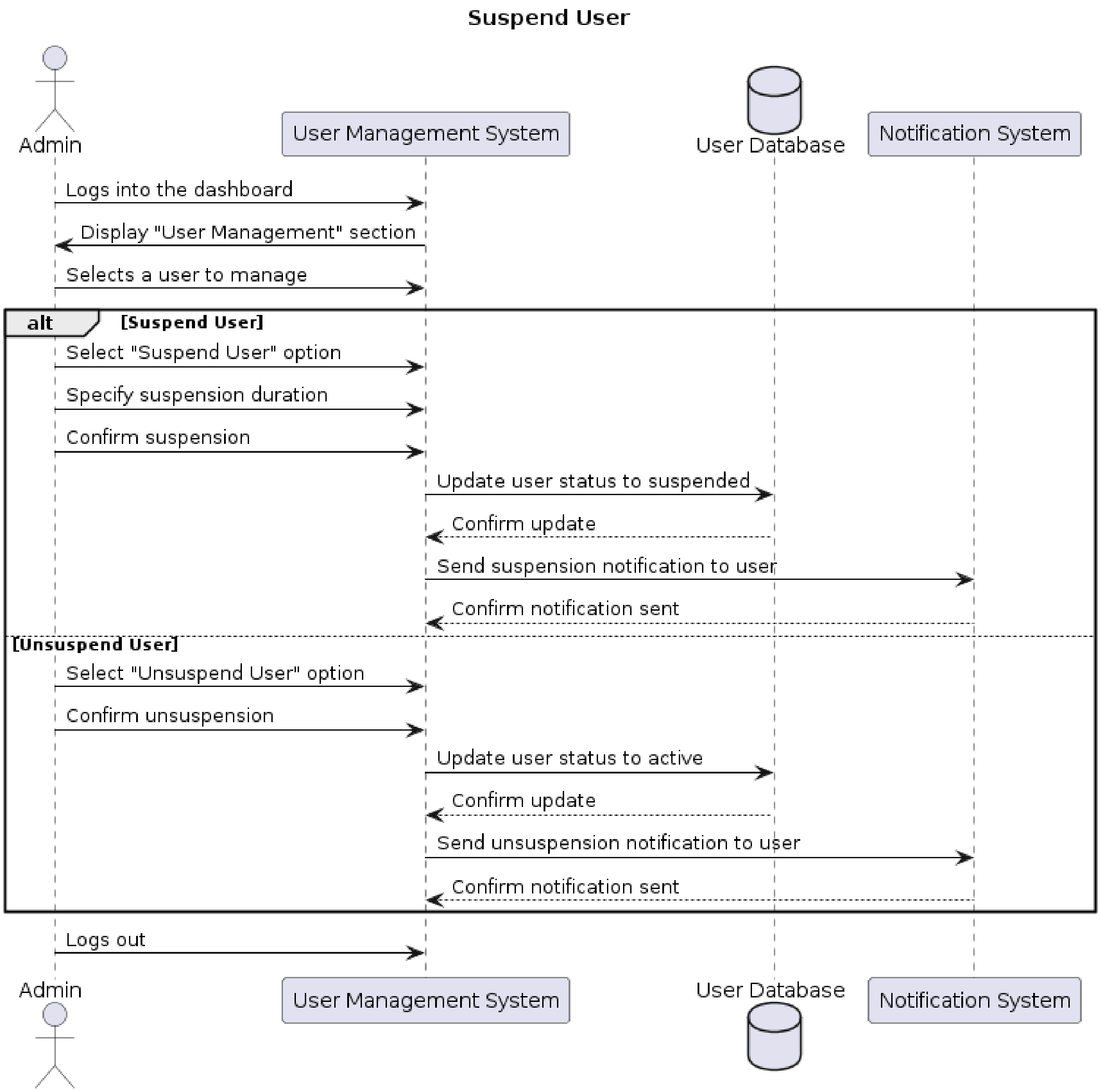
9.13 Vehicle Verification


Odyssey Verification Sequence Diagram
Admin
Vehicle
Owner
Open Admin Panel
Select Vehicle for Verification
Review Uploaded Vehicle Documents and Photos
Select Package
Documents and Photos Valid?
alt
[Documents and Photos Valid]
Approve Vehicle
Notify Owner of Approval
[not Valid]
Reject Vehicle
Notify Owner of Rejection
Admin
Vehicle
Owner

9.14 Coupon Management



9.15 Suspend User


Suspend User
Admin
User Management System
User Database
Notification System
Logs into the dashboard
Display "User Management" section
Selects a user to manage
alt
[Suspend User]
Select "Suspend User" option
Specify suspension duration
Confirm suspension
Update user status to suspended
Confirm update
Send suspension notification to user
Confirm notification sent
[Unsuspend User]
Select "Unsuspend User" option
Confirm unsuspension
Update user status to active
Confirm update
Send unsuspension notification to user
Confirm notification sent
Logs out
Admin
User Management System
User Database
Notification System

10.Requirement Traceability Matrix

A traceability matrix is a document, usually in the form of a table, used to assist in determining the completeness of a relationship by correlating any two baselined documents using a many-to-many relationship comparison. It is often used with high-level requirements (these often consist of marketing requirements) and detailed requirements of the product to the matching parts of high-level design, detailed design, test plan, and test cases.

Requirements Traceability Matrix					
Project Name	**Odyssey**	**Business Area**		**Local**	
Functional Activity	**Use Case Reference**	**Design Document Reference**	**Code Module/ Reference**	**User Acceptance Validation**	**Comments**
FR1	UC1, UC2			Pass	
FR2	UC10			Verified	
FR3	UC12			Verified	
FR4	UC3			Verified	
FR5	UC11			Verified	
FR6	UC7			Verified	

11.Appendix

11.1 Prioritization of requirements

We've prioritized the requirements by following Three-level Scale technique.

11.1.1 Three-level Scale

When a Business Analyst categorizes the requirements in any of the ordering or ranking scale, it is subject to the analyst's understanding of the business. Many analysts suggest that this method has some drawbacks and advocate methods that have more than one scale.

11.1.2 Prioritization of the requirements of Odyssey

FR1 – High priority

FR2 – High priority

FR3 – High priority

FR4 – Medium priority

FR5 – High priority

FR6 – Medium priority

DR1 – High priority

DR2 – High priority

DR3 – High priority

DR4 – High priority