VEHICLE PORTAL

PROJECT THESIS

SUBMITTED

TO

AWH ENGINEERING COLLEGE

KUTTIKKATTOOR, KOZHIKODE - 8

IN PARTIAL FULFILLMENT OF THE

REQUIREMENTS FOR THE AWARD OF THE DEGREE

OF

Master of Computer Applications

BY

ABHINAV S

DEPARTMENT OF COMPUTER APPLICATIONS

AWH ENGINEERING COLLEGE, KUTTIKKATTOOR,
KOZHIKODE

MAY 2023

DEPARTMENT OF COMPUTER APPLICATIONS

AWHENGINEERINGCOLLEGE

KOZHIKODE

CERTIFICATE

*This is to certify that this thesis entitled "**VEHICLES PORTAL**" submitted herewith is an authentic record of the thesis work done by **ABHINAV S(AWH21MCA-2002)**under our guidance in partial fulfillment of the requirements for the award of **Master of Computer Applications** from APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY during the academic year 2023.*

Mrs. Sruti Sudevan	**Mrs. Keerthi Sankar K**
Assistant Professor	Assistant Professor
Head of the Department	Project Guide
External Examiner	**Internal Examiner**

ACKNOWLEDGEMENT

I express my sincere gratitude to our beloved principal **Dr. Sabeena M V** for providing me an opportunity with the required facilities for doing this project. I express my hearty thanks to **Mrs. Sruti Sudevan**, Head of department of MCA and **Mrs. Keerthi Sankar K,** Assistant Professor department of MCA for her guidance. I am thankful to all other staff of MCA department for their encouragement, timely guidance, valuable suggestions and inspiring ideas given throughout this mini project. I am grateful to my friends for the way they have co-operated, expected me to achieve success and have always stirred my ambition to do the best. Above all, I am grateful to the almighty, who has showered his blessings on me through my life and throughout the project.

ABHINAV S

ABSTRACT

The vehicles rental and transportation project is a web-based application designed to simplify the process of renting and managing vehicles for transportation purposes. The system aims to provide an easy-to-use interface for customers to rent vehicles and for the rental company to manage their fleet. The project includes features such as vehicle booking, payment management, Customers can browse available vehicles, select a vehicle according to their requirements, and book it for a specific time period. The payment management system allows customers to make payments for their selected vehicles, and the rental company to keep track of the payment status.

The vehicles rental and transportation project will improve the efficiency of vehicle rental and transportation services, reduce errors, and provide transparency to customers and rental companies. The system will save time and resources for the rental company and provide a better overall experience for customers. It will also help rental companies to manage their fleet effectively and ensure that their vehicles are well-maintained and in good condition.

CONTENTS

Page No.

INTRODUCTION

1.INTRODUCTION

The Vehicles Rental and Transportation Project is an online platform created with the aim of streamlining the procedures involved in renting and overseeing vehicles for transportation needs. Its user-friendly interface allows customers to browse available vehicles, select a suitable vehicle based on their needs, and book it for a specific time period. The system also features a payment management system that enables customers to pay for their selected vehicles, while allowing the rental company to track the payment status.

The main goal of the Vehicles Rental and Transportation Project is to enhance the efficiency of vehicle rental and transportation services, minimize errors, and promote transparency for both customers and rental companies. The system can save time and resources for the rental company and provide a better rental experience for customers. Moreover, the system can help rental companies manage their fleet effectively, ensuring that their vehicles are well-maintained and in good condition. With the Vehicles Rental and Transportation Project, customers can avoid the headaches and frustrations typically associated with traditional vehicle rental and transportation services. Instead, they can enjoy a seamless and stress-free experience.

SYSTEM ANALYSIS

2. SYSTEM ANALYSIS

2.1 Existing system

The current system only for booking cabs only otherwise only a particular Services can performed. Is an online reservation system which helps the customers to book cabs and vehicles online rather than using the traditional manual system of vehicle reservation.

In this system cannot find nearby available vehicles and other services. In traditional system which require lots physical and mental efforts. The system many chances of human errors like wrong entry of journey date, journey time and destination as everything is recorded manually in a register by an employee

2.2 Proposed system

In this system proposed all types of vehicles such as car, bikes, carry van, truck all the type of vehicles booking for services and rent is online and it is very easy to booking vehicles and other services.

We can very easy to find available vehicles and other services in nearby in this system journey date journey time and destination as everything is recorded online in a registered by receiver. The main advantage is that the user shall be able to choose a vehicle depending on his budget.

2.3Module description

This project has 3 modules:

ADMIN

- Login
- Add Rental Vehicles

- Add Transportation Vehicles
- Add Notification
- Manage Rental vehicles
- Manage Truck
- Manage Transportation Vehicles
- View Payment

PROVIDER

- Register
- Login
- Add Rental Vehicles
- Add Truck
- Add Transportation Vehicles
- View Notification

USER

- Register
- Login
- Book Rental Vehicles
- Book Transportation Vehicles
- Book Truck
- View Notification
- Payment
- View Booking

2.2 Sprint

Sprint 1

Module	Task	Pending task if any	Hours for Completi-on	Expected date of Completion	Actual date of Completio-n	Reason for deviation
Requirement Study	Requirement Study	-	11 hours.	06/02/2023	06/02/2023	-
	Figma UI		10 hours.	09/02/2023	09/02/2023	
Admin, Provider, User	Login	-	10 hours.	14/02/2023	14/02/2023	-
Admin	Add Rental Vehicles	-	12 hours.	17/02/2023	17/02/2023	-
	Add Transportation Vehicles	-	12 hours.	21/02/2023	21/02/2023	-
	Add Truck	-	12 hours.	24/02/2023	24/02/2023	
	Manage Rental Vehicles	-	12 hours.	28/02/2023	28/02/2023	-
	Manage Transportation Vehicles	-	12 hours.	06/03/2023	06/03/2023	-
	Manage Truck	-	12 hours.	08/03/2023	08/03/2023	-
	Add notifications	-	12 hours	10/03/2023	10/03/2023	-

	View Rental Vehicles		10 hours	14/03/2023	14/03/2023	-
	View Transportation Vehicles		10 hours	16/03/2023	16/03/2023	-

Sprint 2

Module	Task	Pending task if any	Hours for Completion	Expected date of Completion	Actual date of Completion	Reason for deviation
Provider	Register	-	10 hours.	20/03/2023	20/03/2023	-
	Add Rental Vehicles	-	11 hours.	23/03/2023	23/03/2023	-
	Add Truck	-	11 hours.	27/03/2023	27/03/2023	
	Add Transportatio-n Vehicles	-	11 hours.	29/03/2023	29/03/2023	-
	View notifications	-	10 hours.	04/04/2023	04/04/2023	-
	Add Truck	-	11 hours.	10/04/2023	10/04/2023	-
User	Register	-	10 hours.	11/04/2023	11/04/2023	-
	BookTransportatio-n Vehicles	-	10 hours.	13/04/2023	13/04/2023	-
	Book Rental Vehicles	-	10 hours.	18/04/2023	18/04/2023	-

	Book Truck	-	10 hours.	20/04/2023	20/04/2023	-
	View Notification	-	10 hours.	24/04/2023	24/04/2023	-
	Payment	-	11 hours.	26/04/2023	26/04/2023	
	View Booking	-	10 hours.	28/04/2023	28/04/2023	-

2.5 User Stories

The Vehicles Portal was a mobile and web application that consisted of 3 modules such as Admin, Provider, User. The Admin logged into the system and added details of rental and transportation and also added truck details. Then the admin managed the rental and transportation of the vehicle also truck, accepting or rejecting the vehicles that were added by the provider. The admin also added notifications based on seasonal offers, population density, and other relevant factors. Managed the rental and transportation process, including viewing the accepted and rejected rental and transportation vehicles and trucks. The admin was able to view the accepted and rejected rental and transportation vehicle and truck that were added by the admin or provider. The Admin could also add notifications and view payment details of the user.

The Provider could also register and log in to the system and add rental and transportation vehicles and trucks. The Provider could view notifications that were added by the admin. The User could also register and log in to the system.

The User booked the rental and transportation vehicles and trucks and also viewed notifications that were added by the admin. The user was also able to view the rental booking details such as whether the vehicle was successfully booked or not. The User made a one-time payment for the rental services. The user could book the rental and transportation vehicles and trucks based on the location that were added while registering.

FEASIBILITY STUDY

3. FEASIBILITY STUDY

A feasibility study is an important step in evaluating the viability and potential success of a project. In the case of a rental and transportation vehicle system using the Flutter framework. Assess the demand for rental and transportation services in your target market. Research the existing competition, their offerings, and their customer base. Determine if there is a need for your system and identify any unique features or advantages you can provide.

3.1 Economical Feasibility

Economic feasibility of a rental and transportation vehicle system involves evaluating the financial aspects and determining whether the project is economically viable. Determine the potential revenue streams for the rental and transportation system. This could include rental fees, transaction commissions, premium features or services, advertising, or partnerships. Estimate the revenue based on market demand, competition analysis, and projected user adoption. Calculate the costs associated with developing and maintaining the system, including software development, infrastructure, marketing, and ongoing operational expenses. Compare these costs with the potential benefits, such as revenue from rentals, commissions, or other monetization strategies. Assess the projected profitability of the system and its ability to generate a positive return on investment.

3.2 Technical Feasibility

Assessing the technical feasibility of a rental and transportation vehicle system built with Flutter involves evaluating whether the system can be developed and implemented effectively. Evaluate the suitability of the Flutter framework for building a rental and transportation system. Consider factors such as the availability of Flutter packages or plugins that can facilitate the development of key features like user authentication, real-time updates, mapping, and payment integration. Assess the maturity of the framework and its ability to handle the required functionalities.

3.3 Operational Feasibility

Operational feasibility of a rental and transportation vehicle system involves evaluating whether the system can be implemented and operated smoothly within the existing operational environment. Assess the system's ability to scale and accommodate growth in the future. Determine if the system can handle an increasing number of vehicles, customers, and transactions without significant disruptions or performance issues. Plan for future enhancements and adjustments to meet the evolving needs of the business.

3.4 Behavioural Feasibility

Behavioral feasibility, also known as user acceptance or human factors feasibility, assesses the willingness and ability of users to adopt and effectively use a rental and transportation vehicle system. Ensure that the system's user interface (UI) and user experience (UX) are intuitive, user-friendly, and aligned with the target users' expectations. Design the system with clear navigation and logical workflows. Conduct usability testing to identify any usability issues and make necessary improvements.

3.5 Software Feasibility

Software feasibility refers to evaluating the technical aspects of developing the rental and transportation vehicle system using software. Define and document the system requirements, including functional requirements (features, user interactions, and system behavior) and non-functional requirements (performance, security, scalability, and usability). Ensure that the chosen software technologies, including Flutter, can meet these requirements effectively.

SOFTWARE ENGINEERING PARADIGM

4. SOFTWARE ENGINEERING PARADIGM

The software engineering paradigm which is also referred to as a software process model or Software Development Life Cycle (SDLC) model is the development strategy that encompasses the process, methods and tools. SDLC describes the period of time that starts with the software system being conceptualized.

4.1 Agile model

Agile SDLC model is a combination of iterative and incremental process models with focus on process adaptability and customer satisfaction by rapid delivery of working software product. Agile methods break the product into small incremental builds. These builds are provided in iterations. Each iteration typically lasts from about one to three weeks. At the end of the iteration, a working product is displayed to the customer and important stakeholders. Agile methods break the product into small incremental builds. These builds are provided in iterations. Each iteration typically lasts from about one to three weeks. Every iteration involves cross functional teams working simultaneously on various areas like

- Planning
- Requirement Analysis
- Design
- Coding
- Unit Testing
- Acceptance Testing

At the end of the iteration, a working product is displayed to the customer and important stakeholders. Agile model believes that every project needs to be handled differently and the existing methods need to be tailored to best suit the project requirements. In agile, the tasks are divided to time boxes (small time frames) to deliver specific features for a release.

In agile, the tasks are divided to time boxes (small time frames) to deliver specific features for a release. Iterative approach is taken and working software build is delivered after each iteration. Each build is incremental in terms of features; the final build holds all the features required by the customer.

Agile software development is more than frameworks such as scrum, extreme programming, or feature-driven development (fdd).agile software development is more than practices such as pair programming, test-driven development, stand-ups, planning sessions, and sprints.

The most popular agile methods include rational unified process (1994), scrum (1995), crystal clear, extreme programming (1996), adaptive software development, feature driven development, and dynamic systems development method (dsdm) (1995). These are now collectively referred to as agile methodologies, after the agile manifesto was published in 2001. Following are the

Agile manifesto principles:

- individuals and interactions − in agile development, self-organization and motivation are important, as are interactions like co-location and pair programming.
- working software − demo working software is considered the best means of communication with the customers to understand their requirements, instead of just depending on documentation.
- customer collaboration − as the requirements cannot be gathered completely in the beginning of the project due to various factors, continuous customer interaction is very important to get proper product requirements.
- responding to change − agile development is focused on quick responses to change and continuous development.

4.2 Scrum

Scrum is an agile framework for managing knowledge work, with an emphasis on software development. It is designed for teams of three to nine members, who break their work into actions that can be completed within time boxed iterations, called "sprints", no longer than one month and most commonly two weeks, then track progress and re-plan in 15-minute stand-up meetings, called daily scrums.

Scrum is an iterative and incremental framework for managing product development. It defines "a flexible, holistic product development strategy where a development team works as a unit to reach a common goal", challenges assumptions of the"traditional, sequential approach to product development, and enables teams to self organize by encouraging physical co-location or close online collaboration of all team members, as well as daily face-to-face communication among all team members and disciplines involved.

Scrum is a framework that helps teams work together. Much like a rugby team (where it gets its name) training for the big game, scrum encourages teams to learn through experiences, self-organize while working on a problem, and reflect on their wins and losses to continuously improve.

While the scrum is talking about is most frequently used by software development teams, its principles and lessons can be applied to all kinds of teamwork. This is one of the reasons scrum is so popular. Often thought of as an agile project management framework, scrum describes a set of meetings, tools, and roles that work in concert to help teams structure and manage their work.

Scrum is the most common Agile framework, and the one most people start with. Agile practices on the other hand, are techniques applied during phases of the software development lifecycle. Planning Poker for example, is a collaborative estimation practice designed to encourage team members to share their understanding of what done means. The process is quite fun, and has proven to help foster teamwork and better estimates. Continuous Integration (also known as CI) is a common Agile engineering practice where code changes are integrated into the main branch frequently. An automated build verifies changes, leading to a reduction in integration debt and a continually shippable main branch. These practices, like all Agile practices, carry the Agile label, because they are consistent with the principles in the Agile manifesto. In the project management, scrum, sometimes written Scrum or SCRUM, is a framework for developing, delivering, and sustaining products in a complex environment, with an initial emphasis on software development, although it has been used in other fields including research, sales, marketing and advanced technologies. It is designed for teams of ten or fewer members, who break their work into goals that can be completed within time-boxed iterations, called sprints, no longer than one month and most commonly two weeks. The scrum team assess progress in time-boxed daily meetings of 15 minutes or less, called daily scrums (a form of stand-up meeting). At the end of the sprint, the team holds two further meetings: the sprint review which demonstrates the work done to stakeholders to elicit feedback, and sprint retrospective which enables the team to reflect and improve. A key principle of Scrum is the dual recognition that customers will change their minds about what they want or need and that there will be unpredictable challenges-for which a predictive or planned approach is not suited. As such, Scrum adopts an evidence based empirical approach accepting that the problem cannot be fully understood or defined up front, and instead focusing on how to maximize the team's ability to deliver quickly, to respond to emerging requirements, and to adapt to evolving technologies and changes in market conditions. Many of the terms used in Scrum (e.g., scrum master) are typically written with leading capitals (e.g., Scrum Master) or as conjoint words written in camel case (e.g., Scrum Master). To maintain an encyclopaedic tone, however, this article uses normal sentence case for these terms-unless they are recognized marks. This is occasionally seen written in all -capitals, as SCRUM. The word is not an acronym, so this is not correct; however, it likely arose due to an early paper by Ken Schwaber which capitalized SCRUM in its title. While the trademark on the term Scrum itself has been allowed to lapse, so that it is deemed as owned by the wider community rather than an individual, the leading capital is retained-except when used with other words.

SYSTEM REQUIREMENT

SPECIFICATION

5. SYSTEM REQUIREMENTS SPECIFICATION

5.1 Software Requirements

• Operating system	: Android.10 or Windows 7 or above
• Frontend	: Flutter, PHP
• Backend	: MySQL
• IDE	: Android studio, Visual studio, MySQL

5.2 Hardware Requirements

• A device with an internet connection	
• Processor	:Intel core i3 or above
• Hard Disk	:128GB and above processor Intel Core i3 and above
• RAM	:4 GB
• Storage	:500GB Hard disk

SYSTEM DESIGN

6. SYSTEM DESIGN

System design is the first in the development phase for many engineered product or system. It may define the process of applying various techniques and principles for the purpose of defining a device, a process or system in sufficient detail to permit its physical realization.

6.1 Database Design

Database design is the process of producing a detailed data model of a database. This logical data model contains all the needed logical and physical design choices and physical storage parameters needed to generate a design in a data definition language, which can then be used to create a database. The term database design can be used to describe many different parts of the design of an overall database system. Principally, and most correctly, it can be thought of as the logical design of the base data structures used to store the data.

In the relational model these are the tables and views. In an object database the entities and relationships map directly to object classes and named relationships. However, the term database design could also be used to apply to the overall process of designing,not just the base data structures, but also the forms and queries used as part of the overall database application within the database management system. The process of doing database design generally consists of a number of steps which will be carried out by the database designer. Usually, the designer must: Determine the relationships between the different data elements and superimpose a logical structure upon the data on the basis of these relationships.

Normalization

It is a process of converting a relation to a standard form. The process is used to handle the problems that can arise due to data redundancy i.e., repetition of data in the database, maintain data integrity as well as handling problems that can arise due to insertion, updation, deletion anomalies.

Insertion anomaly: Inability to add data to the database due to absence of other data.

Deletion anomaly: Unintended loss of data due to deletion of other data.

Update anomaly: Data inconsistency resulting from data redundancy and partial update.

Normal Forms: These are the rules for structuring relations that eliminate anomalies.

1. First Normal Form (1NF)

A relation is said to be in first normal form if the values in the relation are atomic for every attribute in the relation. This mean simply that no attribute value can be a set of values or, as it is sometimes expressed, a repeating group.

2. Second Normal Form (2NF)

A relation is said to be in second Normal form is it is in first normal form and itshould satisfy any one of the following rules.

- Primary key is a not a composite primary key
- No non key attributes are present
- Every non key attribute is fully functionally dependent on full set of primarykeys.

3. Third normal Form(3NF)

A relation is said to be in third normal form if there exist no transitive dependencies.

Transitive dependency: If two non-key attributes depend on each other as well on the primary key then they are said to be transitively dependent. the above normalization principle was applied to decompose the data in multiple tables thereby making the data tobe maintained in a consistent state.

6.1 Tables

Login:

SL NO	FIELD NAME	DATATYPE	CONSTRAINS
1	Login_id	Integer(11)	Primary Key
2	Email	Varchar(50)	Not null
3	Password	Varchar(50)	Not null
4	Type	Varchar(50)	Not null

Registration:

SL NO	FIELD NAME	DATATYPE	CONSTRAINS
1	Regi_id	Integer(11)	Primary key
2	Login_id	Varchar(50)	Foreign Key
3	First_Name	Varchar(50)	Not null
4	Last_Name	Varchar(50)	Not null
5	Street_Address	Varchar(50)	Not null
6	District	Varchar(50)	Not null
7	Phone_Number	Varchar(50)	Not null
8	Email	Varchar(50)	Not null

9	Password	Varchar(50)	Not null

Add Rental Vehicles:

SL NO	FIELD NAME	DATATYPE	CONSTRAINS
1	Rental_id	Integer(11)	Primary key
2	Login_id	Integer(11)	Foreign Key
3	Name	Varchar(50)	Not null
4	Price	Varchar(50)	Not null
5	Vehicle_Type	Varchar(50)	Not null
6	Type_of_Gear	Varchar(50)	Not null
7	Color_of_vehicle	Varchar(50)	Not null
8	Seat_of_Vehicle	Varchar(50)	Not null
9	Fuel_of_Vehicle	Varchar(50)	Not null
10	Location	Varchar(50)	Not null
11	RC	Varchar(50)	Not null
12	Insurance	Varchar(50)	Not null
13	Driving_Licence	Varchar(50)	Not null

14	Upload_Photo	File	Not null
15	Status	Varchar(50)	Not null

Add Transportation Vehicles:

SL NO	FIELD NAME	DATATYPE	CONSTRAINS
1	Transportation_id	Integer(11)	Primary key
2	Login_id	Integer(11)	Foreign Key
3	Vehicle	Varchar(50)	Not null
4	Type	Varchar(50)	Not null
5	Price	Varchar(50)	Not null
6	Seats	Varchar(50)	Not null
7	Location	Varchar(50)	Not null
8	RC	Varchar(50)	Not null
9	Insurarnce	Varchar(50)	Not null
10	Upload_Photo	File	Not null
11	Status	Varchar(50)	Not null

Truck

SL NO	FIELD NAME	DATATYPE	CONSTRAINS
1	Truck_id	Integer(11)	Primary key
2	Login_id	Integer(11)	Foreign Key
3	Name	Varchar(50)	Not null
4	Capacity	Varchar(50)	Not null
5	Image	Varchar(50)	Not null
6	Price	File	Not null
7	RC	Varchar(50)	Not null
8	Place	Varchar(50)	Not null
9	Insurance	Varchar(50)	Not null
10	Location	Varchar(50)	Not null
11	Status	Varchar(50)	Not null

Payment Rental

SL NO	FIELD NAME	DATATYPE	CONSTRAINS
1	rid	Integer(11)	Primary key
2	Login_id	Integer(11)	Foreign Key
3	Rental_id	Integer(11)	Foreign Key
4	Date	Varchar(50)	Not null
5	Status	Varchar(50)	Not null

Payment Transportation

SL NO	FIELD NAME	DATATYPE	CONSTRAINS
1	pid	Integer(11)	Primary key
2	Login_id	Integer(11)	Foreign key
3	Transaction_id	Integer(11)	Foreign key
4	Status	Varchar(50)	Not null

Payment Truck

SL NO	FIELD NAME	DATATYPE	CONSTRAINS
1	id	Integer(11)	Primary key
2	Truck_d	Integer(11)	Foreign key
3	Login_id	Integer(11)	Foreign key
4	Status	Varchar(50)	Not null

Rental Booking

SL NO	FIELD NAME	DATATYPE	CONSTRAINS
1	rid	Integer(11)	Primary key
2	Rental_id	Integer(11)	Foreign key
3	Login_id	Varchar(50)	Foreign key
4	Date	Varchar(50)	Not null
5	Last_Date	Varchar(50)	Not null
6	Status	Varchar(50)	Not null

Transportation Booking

SL NO	FIELD NAME	DATATYPE	CONSTRAINS
1	tid	Integer(11)	Primary key
2	Transportation_id	Integer(11)	Foreign key
3	Login_id	Varchar(50)	Foreign key
4	Date	Varchar(50)	Not null
5	Status	Varchar(50)	Not null

Truck Booking

SL NO	FIELD NAME	DATATYPE	CONSTRAINS
1	id	Integer(11)	Primary key
2	Login_id	Integer(11)	Foreign key
3	Truck_id	Varchar(50)	Foreign key
4	Date	Varchar(50)	Not null
5	Status	Varchar(50)	Not null

Notification

SL NO	FIELD NAME	DATATYPE	CONSTRAINS
1	Notification_id	Integer(11)	Primary key
2	Titile	Varchar(50)	Not null
3	Message	Varchar(50)	Not null
4	Date	Varchar(50)	Not null

6.3 UML Designs

The unified Modeling Language (UML) is a standard language for specifying, visualizing, constructing, and documenting the artifacts of the software systems, as well as for business modeling and other non-software systems. The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems. The UML is a very important part of developing objects-oriented software and software development process. The UML uses mostly graphical notations to express the design of software projects. Using the UML helps project teams communicate, explore potential designs, and validate the architectural design of the software.

Use Case Diagram

A use case diagram is a visual representation in the Unified Modeling Language (UML) that depicts the interactions between actors (users, systems, or external entities) and a system. It showcases the various use cases or functionalities provided by the system and the actors involved in those interactions.

Use case diagrams are valuable for understanding the system's requirements and capturing the system's behavior from a user's perspective. They help stakeholders visualize how the system will be used and identify the different actors and their roles.

In a use case diagram, use cases are represented as ovals, and actors are represented as stick figures or other shapes. The actors are connected to the use cases by lines, indicating their participation in those use cases. Additionally, relationships such as generalization (inheritance), association, and dependencies can be depicted to show connections between actors and use cases.

Use case diagrams provide a high-level view of the system's functionality and the interactions between actors and the system. They assist in identifying the core features of the system and the external entities involved. Use case diagrams can also highlight different scenarios or paths that users can take while interacting with the system.

Sequence Diagram

A sequence diagram is a type of interaction diagram in the Unified Modeling Language (UML) that illustrates the dynamic behavior of a system by showing the interactions between objects or components over time. It portrays the flow of messages between objects in a chronological sequence, representing the order of events during a particular scenario or use case.

In a sequence diagram, the participants or objects involved in the system are depicted as vertical lifelines. Each lifeline represents an instance of a class or component and is labeled with the name of the object or component it represents.

The lifelines are arranged from top to bottom, reflecting the chronological order of their participation.

The interactions between objects are shown as horizontal arrows, called messages, that travel vertically between the lifelines. These messages represent the method calls or communications exchanged between objects. Messages can be synchronous, asynchronous, or self-referential, depending on the nature of the interaction.

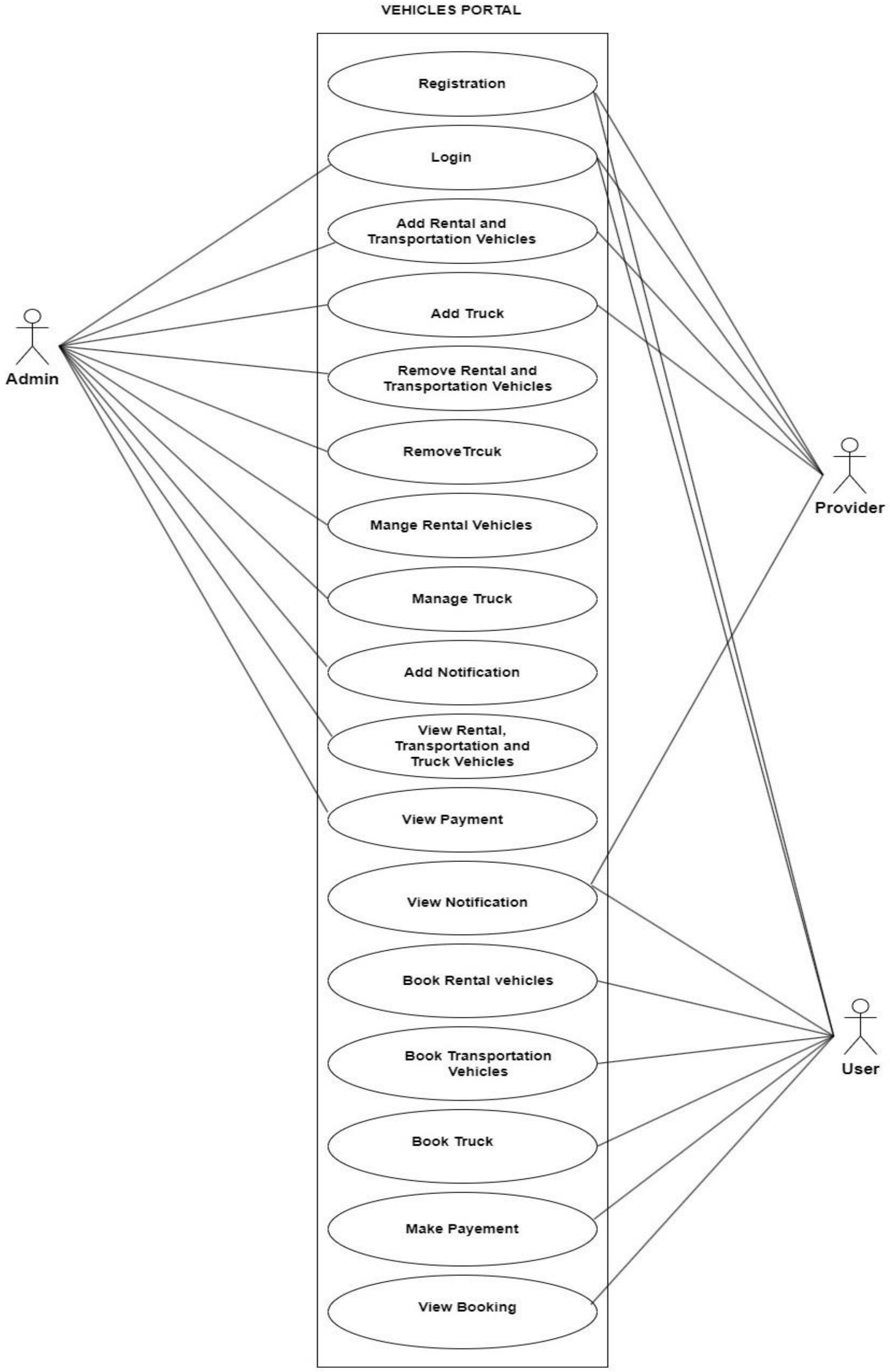
Additionally, sequence diagrams can include various constructs to enhance their expressiveness, such as:

- Activation boxes: Representing the period of time when an object is actively processing a message.
- Conditionals and loops: Showing conditional statements or repetitive behaviors

within the sequence.

- Return messages: Indicating the response or return value from a method call.
- Sequence diagrams are valuable for understanding the dynamic behavior and interactions between objects in a system. They help in visualizing the control flow, identifying potential bottlenecks or issues, and verifying the correctness of the system's design and communication between components.

6.4 Use case diagram



6.5 Scenario

Admin:

- Login
- Add Rental Vehicles
- Add Transportation Vehicles
- Manage Truck–Add and Reject Truck
- Manage Rental Vehicles– Accept and Reject Rental Vehicles
- Reject Transportation Vehicles
- View Transportation Vehicles
- Add Notification

Provider:

- Registration
- Login
- Add Rental Vehicles
- Add Transportation Vehicles
- Add Truck
- View Notification
- View Payment

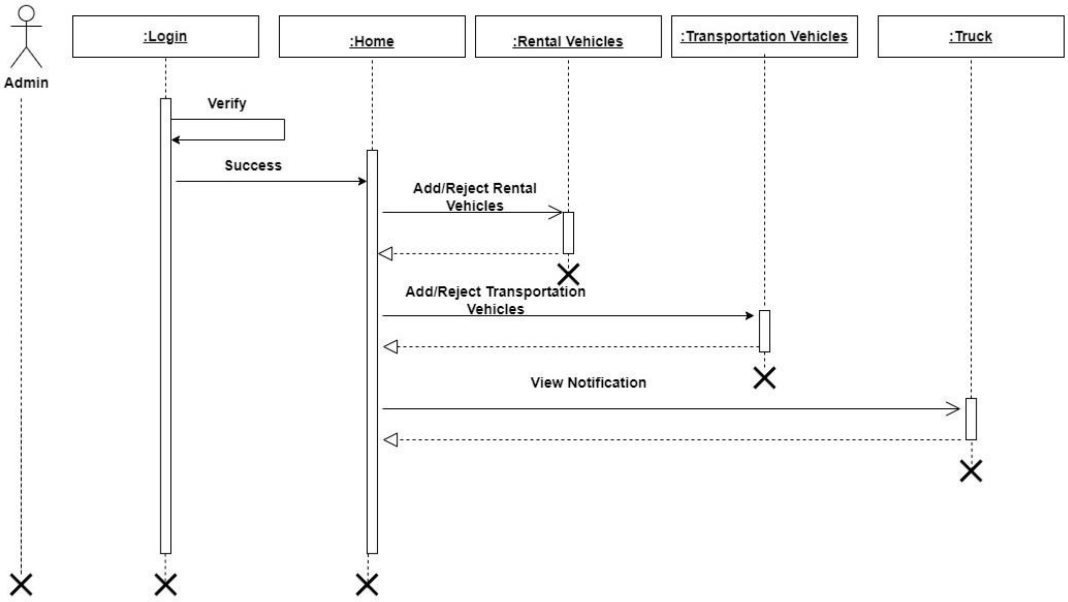
Use:

- Registration
- Login
- Book Rental Vehicles
- Book Transportation Vehicles
- Book Truck
- View Notification
- Payment
- View Booking

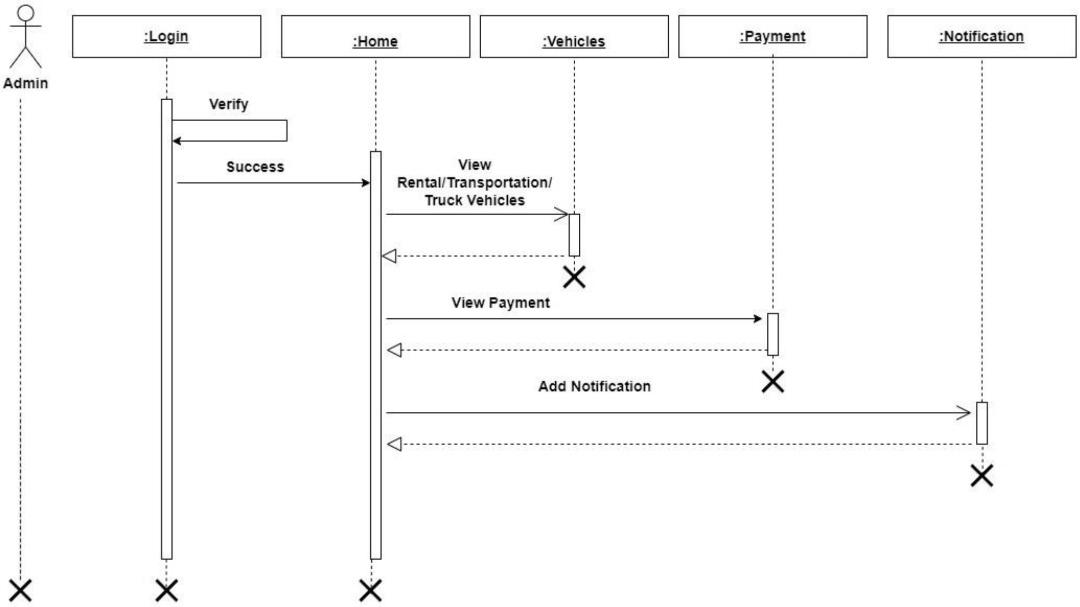
6.4 Sequence Diagram

ADMIN

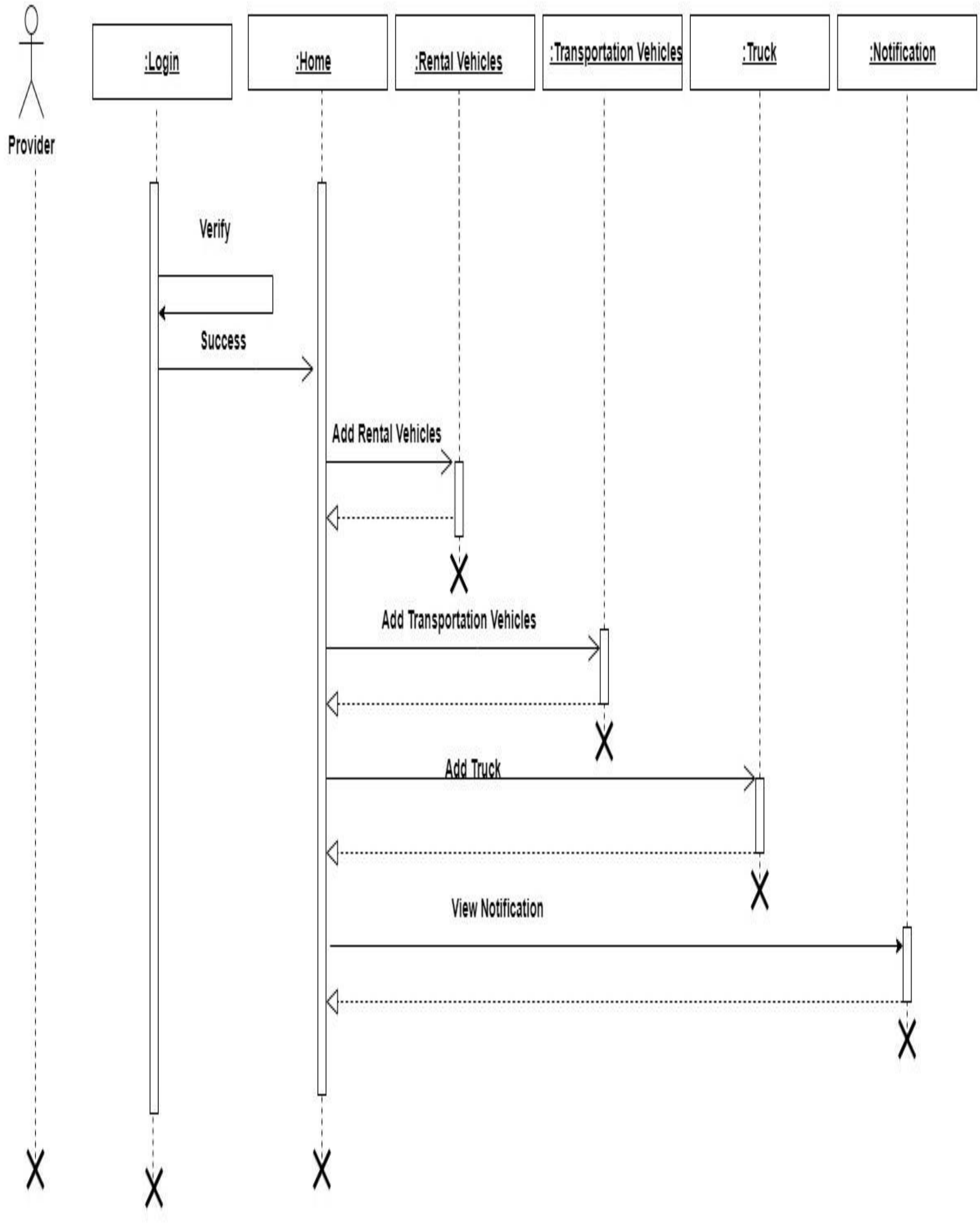
1)



2)

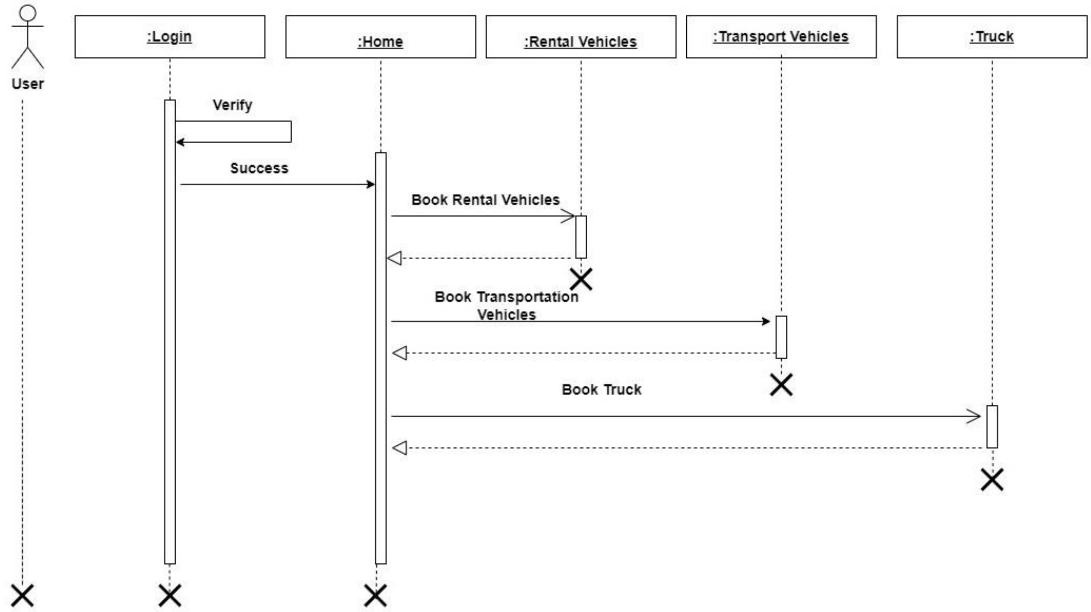


PROVIDER

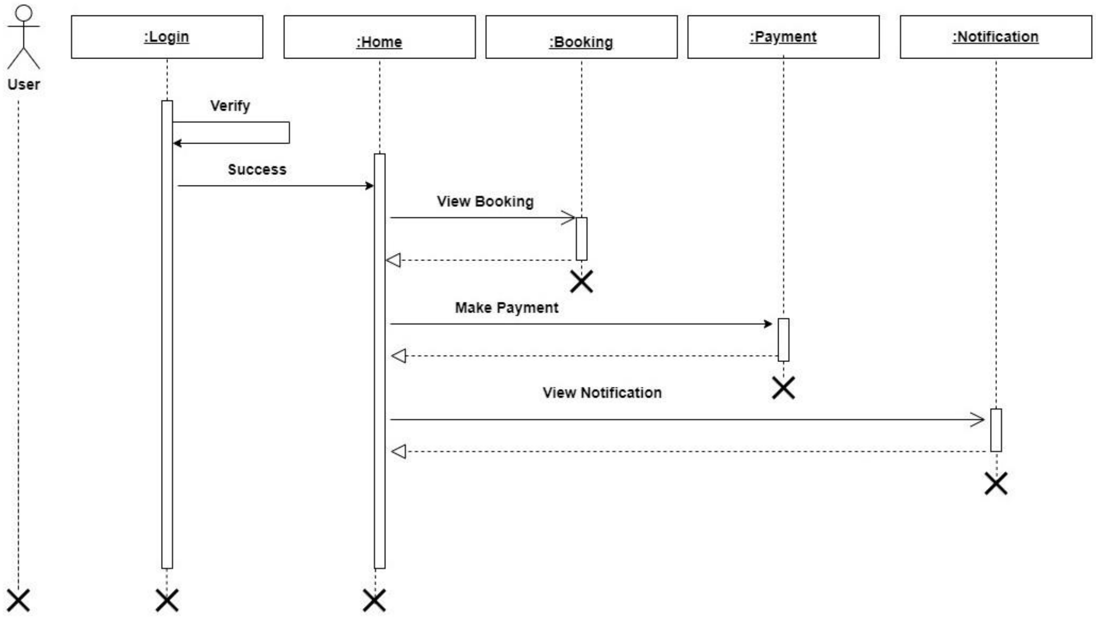

Provider
:Login
:Home
:Rental Vehicles
:Transportation Vehicles
:Truck
:Notification
Verify
Success
Add Rental Vehicles
Add Transportation Vehicles
Add Truck
View Notification

USER

1)


User
:Login
:Home
:Rental Vehicles
:Transport Vehicles
:Truck
Verify
Success
Book Rental Vehicles
Book Transportation Vehicles
Book Truck

2)


User
:Login
:Home
:Booking
:Payment
:Notification
Verify
Success
View Booking
Make Payment
View Notification

SYSTEM DEVELOPMENT

7. SYSTEM DEVELOPMENT

System development is series of operations to manipulate data to produce output from computer system. The principal activities performed during the development phasecan be divided into two major related sequences.

- External system development
- Internal system development

The major external system activities are:

- Implementation
- Planning
- Equipment acquisition
- installation

7.1 Coding

The purpose of code is to facilitate the identification and retrieval of items of information. A code is an ordered collection of symbols designed to provide unique identification of entity or an attribute. Code also show interrelationship among different items. Codes are used to identify, access, sort, matching records. The code ensures that only one value of code with a single meaning is applied to give entity or attribute as described in various ways.

Flutter

Flutter uses the Dart programming language, which is also developed by Google and is known for its fast performance, easy-to-learn syntax, and high productivity. Dart allows Flutter to compile natively to ARM and x86 code, making it possible to create high-performance, visually appealing apps for multiple platforms using a single codebase.

One of the unique features of Flutter is its widgets. Flutter widgets are reusable building blocks that developers can use to create complex UI designs easily. Widgets can be customized and combined in a variety of ways to create beautiful and responsive user interfaces. Flutter also has a large collection of pre-built widgets that developers can use out-of-the-box or modify to suit their needs. Flutter's hot reload feature allows developers to make changes to the code and instantly see the results without having to restart the app. This feature speeds up the development process and helps developers to iterate quickly on their designs and features.

Flutter is also known for its strong community support and extensive documentation, making it easy for developers to get started with the framework and find solutions to common problems.

PHP

PHP is primarily used for creating dynamic web pages and server-side scripting, allowing developers to create web applications that interact with databases, handle forms, manage sessions, and more. PHP is also widely used in content management systems (CMS) such as WordPress, Drupal, and Joomla, which are used to create and manage websites.

One of the advantages of PHP is its ease of use and flexibility. It has a simple syntax that is easy to learn and understand, even for beginners. It also supports a wide range of platforms and databases, including MySQL, PostgreSQL, Oracle, and Microsoft SQL Server. Another benefit of PHP is its strong community support and extensive documentation. There are many online resources available for learning PHP and finding solutions to common problems, making it easy for developers to get started with the language. However, PHP has also been criticized for its security vulnerabilities, especially in older versions. It is important for developers to stay up to date with the latest security updates and best practices to ensure the security of their applications.

Dart

Dart is a programming language developed by Google and released in 2011. It is primarily used for building web and mobile applications, as well as server-side applications and command-line tools. Dart is an object-oriented language with a syntax similar to C-style languages like Java and C++. It also has some features from other programming languages, such as JavaScript and Python. Dart is designed to be easy to learn and use, with a simple and clean syntax. One of the main features of Dart is its support for both AOT (Ahead of Time) and JIT (Just In Time) compilation. AOT compilation allows Dart code to be compiled into native machine code, which can improve performance, especially for mobile applications. JIT compilation, on the other hand, allows for faster development cycles, with the ability to quickly compile and run code during development.

Dart also has a built-in asynchronous programming model, which makes it easy to write code that handles asynchronous events, such as network requests or user input, without blocking the main thread. This can improve the performance and responsiveness of applications. Dart is often used in conjunction with the Flutter framework for building mobile and web applications. Flutter is a UI toolkit that allows developers to build beautiful, responsive, and fast applications for multiple platforms using a single codebase. Flutter uses Dart as its primary programming language, and the two technologies work seamlessly together.

MySQL Database

MySQL is a popular open-source relational database management system (RDBMS) that is used to store, manage, and retrieve data. It was created by Swedish developers Michael Widenius and David Axmark in the mid-1990s and has since become one of the most widely used database systems in the world. MySQL uses the SQL (Structured Query Language) to communicate with its databases. SQL is a standard language that is used to create, modify, and manipulate databases. MySQL supports a wide range of SQL commands, including SELECT, INSERT, UPDATE, DELETE, and JOIN, which are used to retrieve, insert, update, delete, and combine data in the database.

MySQL is known for its performance, scalability, and reliability. It can handle large amounts of data and is designed to run on a variety of operating systems, including Windows, Linux, and macOS. It is commonly used in web applications, content management systems, and other types of software that require data storage and retrieval. MySQL is also known for its security features, including user authentication, access control, and encryption. It offers a range of authentication methods, including password-based authentication, SSL/TLS encryption, and LDAP authentication, among others. Overall, MySQL is a powerful and flexible database system that is widely used by developers and businesses of all sizes. Its open-source nature and active community of developers make it a popular choice for building web applications and other software projects. MySQL is the world’s most popular open source database. According to DB-Engines, MySQL ranks as the second-most-popular database, behind Oracle Database. MySQL powers many of the most accessed applications, including Facebook, Twitter, Netflix, Uber, Airbnb, Shopify, and Booking.com.

MySQL is open source, it includes numerous features developed in close cooperation with users over more than 25 years. So it’s very likely that your favorite application or programming language is supported by MySQL Database. Databases are the essential data repository for all software applications. For example, whenever someone conducts a web search, logs in to an account, or completes a transaction, a database system is storing the information so it can be accessed in the future.

A relational database stores data in separate tables rather than putting all the data in one big storeroom. The database structure is organized into physical files optimized for speed. The logical data model, with objects such as data tables, views, rows, and columns, offers a flexible programming environment. You set up rules governing the relationships between different data fields, such as one to one, one to many, unique, required, or optional, and “pointers” between different tables. The database enforces these rules so that with a well-designed database your application never sees data that’s inconsistent, duplicated, orphaned, out of date, or missing.

The “SQL” part of “MySQL” stands for “Structured Query Language.” SQL is the most common standardized language used to access databases. Depending on your programming environment, you might enter SQL directly (for example, to generate reports), embed SQL statements into code written in another language, or use a language-specific API that hides the SQL syntax.

Open source means it’s possible for anyone to use and modify the software. Anybody can download MySQL software from the internet and use it without paying for it. You can also change its source code to suit your needs. MySQL software uses the GNU General Public License (GPL) to define what you may and may not do with the software in different situations.

If you feel uncomfortable with the GNU GPL or need to embed MySQL code into a commercial application, you can buy a commercially licensed version from Oracle. See the MySQL Licensing Information section for more information.

MySQL is currently the most popular database management system software used for managing the relational database. It is open-source database software, which is supported by Oracle Company. It is fast, scalable, and easy to use database management system in comparison with Microsoft SQL Server and Oracle Database. It is commonly used in conjunction with PHP scripts for creating powerful and dynamic server-side or web-based enterprise applications.

It is developed, marketed, and supported by MySQL AB, a Swedish company, and written in C programming language and C++ programming language. The official pronunciation of MySQL is not the My Sequel; it is My Ess Que Ell. However, you can pronounce it in your way. Many small and big companies use MySQL. MySQL supports many Operating Systems like Windows, Linux, MacOS, etc. with C, C++, and Java languages.

MySQL is a Relational Database Management System (RDBMS) software that provides many things, which are as follows:

- It allows us to implement database operations on tables, rows, columns, and indexes
- It defines the database relationship in the form of tables (collection of rows and columns), also known as relations.
- It provides the Referential Integrity between rows or columns of various tables.
- It allows us to updates the table indexes automatically.

SYSTEM TESTING AND IMPLEMENTATION

8. SYSTEM TESTING AND IMPLEMENTATION

Testing is the vital to the success of the system. It makes a logical assumption that ifall the parts of the system are correct, the goal will be successfully achieved in this project. It is the stage of implementation, which ensures that system works accurately and effectivelybefore the live operation commences. It is a confirmation that all are correct and opportunityto show users that the system must be tested and show that the system will operatesuccessfully and produce expected results under expected conditions. Software testing is a crucial element of software quality assurance and represents the unlimited review of specification, design and coding. Testing represents an interesting anomaly for the software.During the earlier definition and development phase, it was attempted to build the software from an abstract concept to implement.

Testing is a set of activity that can be planned in advance and conducted. Systematically, this is aimed at ensuring that the system works accurately and efficiently before live operations commences.

8.1 Types of Testing

Different types of testing are,

- Unit testing
- Integration testing
- Functional testing
- Performance testing
- Validation testing
- Black Box testing
- White Box testing

Unit Testing

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases. All modules were tested and individually as soon as they were completed and were checked for their correct functionality. Unit testing deals with testing a unit as a whole. This would test the interaction of many functions but confine the test within one unit. This testing is carried out during programming stage itself. In this testingstep each Module is found to be working satisfactorily as regard to the expected output fromthe module.

Integration Testing

Integration testing is a systematic technique for testing to overcome the errors associated within the interface. In this System all the modules such as login, registration, admin in web, user in web, and the final hardware and software are combined and then the entire program is tested as a whole. Thus, in the integration testing step all the errors in theimplementation of the system are corrected. Data can be lost across an interface one modulecan have an adverse effect on others; sub-functions when combined may not produce the desired major functions integration testing is a systematic testing for constructing the program structure. The objective is to take unit tested modules and to combine them and test it as a whole.

- The system contains various components and they have to be combined and tested.
- The software is embedded into hardware and tested.
- The modules are combined and tested.
- The hardware part tested separately.

Functional Testing

Functional testing is a type of software testing that verifies that an application or system functions correctly and meets the specified requirements. It is a black-box testing technique that focuses on the external behavior of the system and ensures that it performs the tasks it is intended to do. Functional testing involves creating test cases that cover various scenarios, inputs, and outputs to validate the expected behavior of the system. The goal of functional testing is to ensure that the application or system works as expected from the user's perspective and that it meets the business requirements. Functional testing can be manual or automated and can be performed at different stages of the software development life cycle, from unit testing to system testing and acceptance testing.

Performance Testing

Performance testing is a type of software testing that evaluates the speed, responsiveness, stability, scalability, and resource usage of an application or system under different conditions. It is typically conducted to identify performance bottlenecks and measure the system's performance in terms of response time, throughput, and resource utilization. Performance testing can involve different types of testing, such as load testing, stress testing, endurance testing, and spike testing. It involves simulating realistic workloads and traffic patterns to measure how the system behaves and performs under different scenarios. The goal of performance testing is to ensure that the system meets the performance requirements and can handle the expected workload and user traffic. It is an essential part of the software development life cycle and helps to identify and address performance issues before they affect the end-user experience.

Validation Testing

Validation testing is the process of assessing a new software product to ensure that its performance matches consumer needs. Product development teams might perform validation testing to learn about the integrity of the product itself and its performance in different environments. Developers can perform validation testing themselves, or collaborate with quality assurance professionals, external validation testing professionals or clients to identify elements of the code to improve. Developers can also combine this type of testing with other useful techniques like product verification, debugging and certification to help ensure the product is ready for the market.

Black Box Testing

Black box testing is a method of software testing that examines the functionality of an application without peering into its internal structures or workings. This method of test can be applied virtually to every level of software testing: unit, integration, system and acceptance. It is sometimes referred to as specification-based testing. In black box testing, the tester does not know how the software works internally. They only know what the software is supposed to do. The tester provides inputs to the software and observes the outputs. They then compare the outputs to the expected outputs to determine if the software is working correctly. Black box testing is a valuable tool for finding defects in software. It can be used to test a wide range of software, from simple websites to complex enterprise applications. Black box testing is also a relatively inexpensive method of testing, making it a cost-effective way to improve the quality of software.

White Box Testing

White box testing is a method of software testing that examines the internal structures or workings of an application, as opposed to its functionality. In white box testing, an internal perspective of the system is used to design test cases. White box testing is also known as clear box testing, open box testing, transparent box testing, code-based testing, and glass box testing. White box testing is typically performed by software developers who have access to the source code. However, it can also be performed by testers who have access to other documentation, such as design documents or test plans.

8.2 Implementation

Implementation is the stage of project, when theoretical design is turned in to a working system. The most crucial stage is achieving a successful system and confidencethat the new system will be work effectively. It involves careful planning, investigation of the manual system and to new system. Implementation means converting a new or revised system design into an operational one. The implementation includes all those activities that take place to convert from the old system to the new one.

There are several activities involved while implementing a project:

- Careful planning.
- Investigating the current system and its constraints on implementation.
- Design of methods to achieve the changeover.
- Training of the staff in the changeover procedure and evaluation of change over method.

Implementation is the final stage and it is an importance phase. The first task in implementation was the implementation planning, that is deciding on methods to be adopted. After the system was implemented successfully, training of the user was one of the most important subtasks of the developer. For this purpose, the user or system manual were prepared and handled over to the user to operate the developed system. So, change over plays a vital role, which checks the developed tool for the following requirements, and then only the user accepted the developed tool. The changeover took place only when the system had been proved to the satisfaction of the system analysis and other implementation activities have been completed.

SYSTEM MAINTENANCE

9. SYSTEM MAINTENANCE

Maintenance is making adaptation of the software for external changes (requirements changes or enhancements) and internal changes (fixing bugs). When changes are made during the maintenance phase all preceding steps of the model must be revisited.

There are 3 types of maintenance:

- Corrective (Fixing bugs/errors)
- Adaptive (Updates due to environment changes)
- Perfective (Enhancements, requirements changes)

Maintenance is enigma of the system development. The definition of the software maintenance can be given describing four activities that are undertaken after the program is released for use.

The maintenance activity occurs since it is unreasonable to assume that software testing will uncover all in a large system. The second activity that contributes the definition of maintenance occurs since rapid changes are encountered in every aspects of computing. The third activity involves recommendation for new capabilities, modification to the existing functions and general enhancements when the software is used. The fourth maintenance activity occurs when software is changed to improve futuremaintainability or reliability.

FUTURE ENHANCEMENT

10. FUTURE ENHANCEMENT

In future, if the vehicle breaks down during the journey, it can be arranged for another vehicle immediately, making the journey smoother and providing additional services to the user such as room and food. A discount will be given if the user books rent for more than one month . Implement real-time GPS tracking of vehicles to allow users to track the location of their rented vehicle. This feature provides transparency and helps users plan their journey more effectively. Allow users to provide ratings and reviews for vehicles and their rental experience. This feedback can help future users make informed decisions and improve the overall quality of the rental service. Implement an in-app chat feature that allows users to communicate directly with customer support representatives or rental agents. This feature enables users to seek assistance, ask questions and resolve issues in real time without the need for external communication.

CONCLUSION

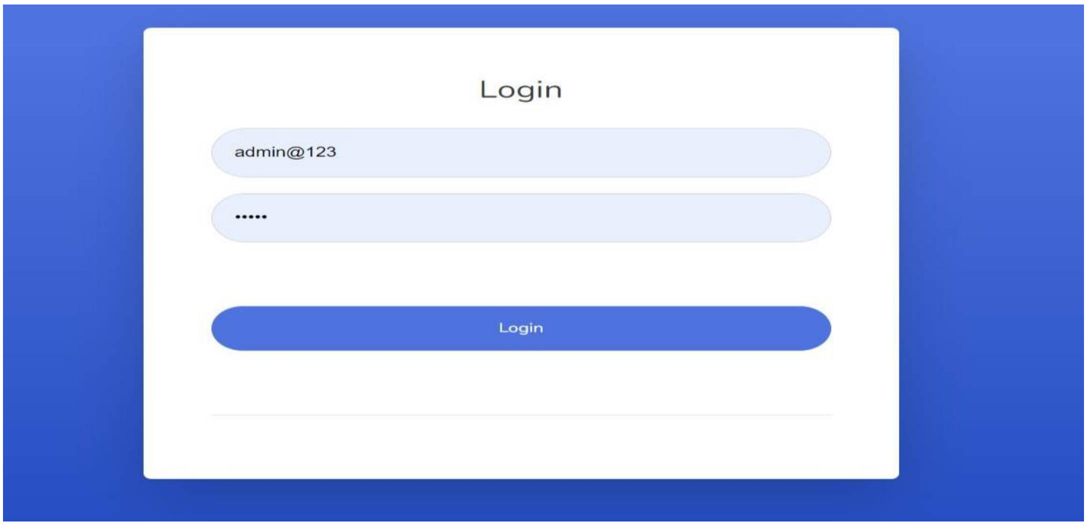
11.CONCLUSION

The vehicle rental and transportation system plays a crucial role in meeting the mobility needs of individuals and societies. The rental system offers a flexible and cost-effective way for people to access vehicles without the commitment and expenses of ownership. It provides a diverse range of vehicles to cater to different transportation requirements and promotes sustainable practices by encouraging shared vehicle usage and offering eco-friendly options. The transportation system offers various modes of public transportation, providing efficient and sustainable options for moving people within and between cities. It enhances accessibility, reduces traffic congestion, and supports inclusive mobility for individuals with disabilities or limited access to personal vehicles.

APPENDIX

Admin

Login

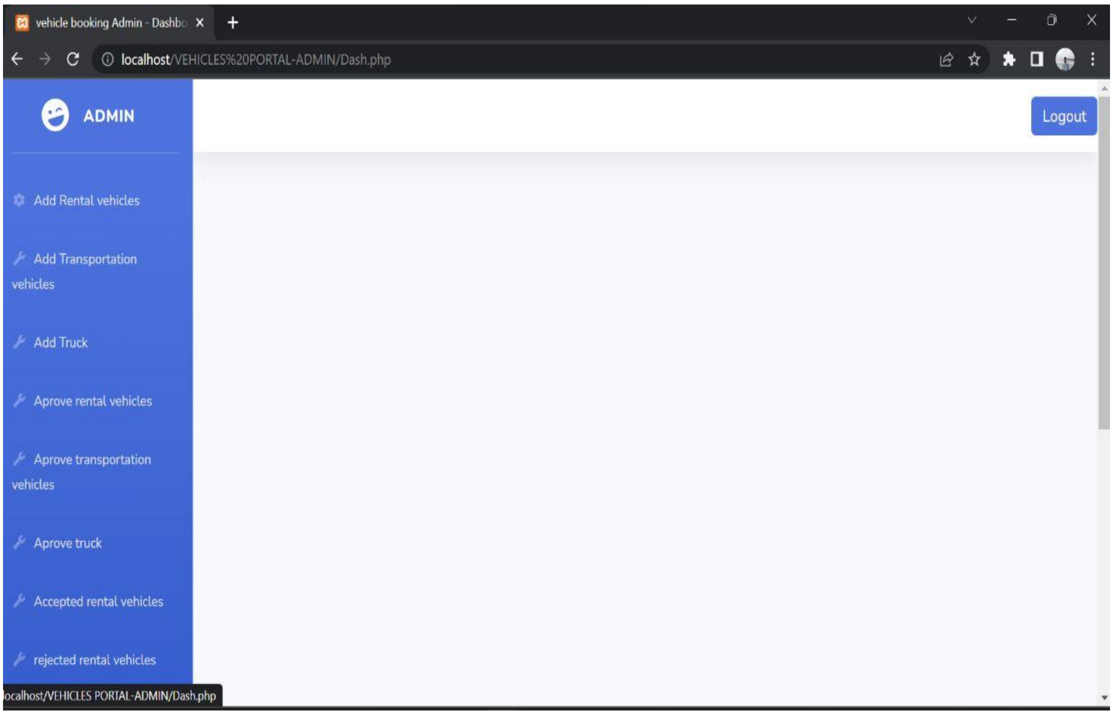


Login

admin@123

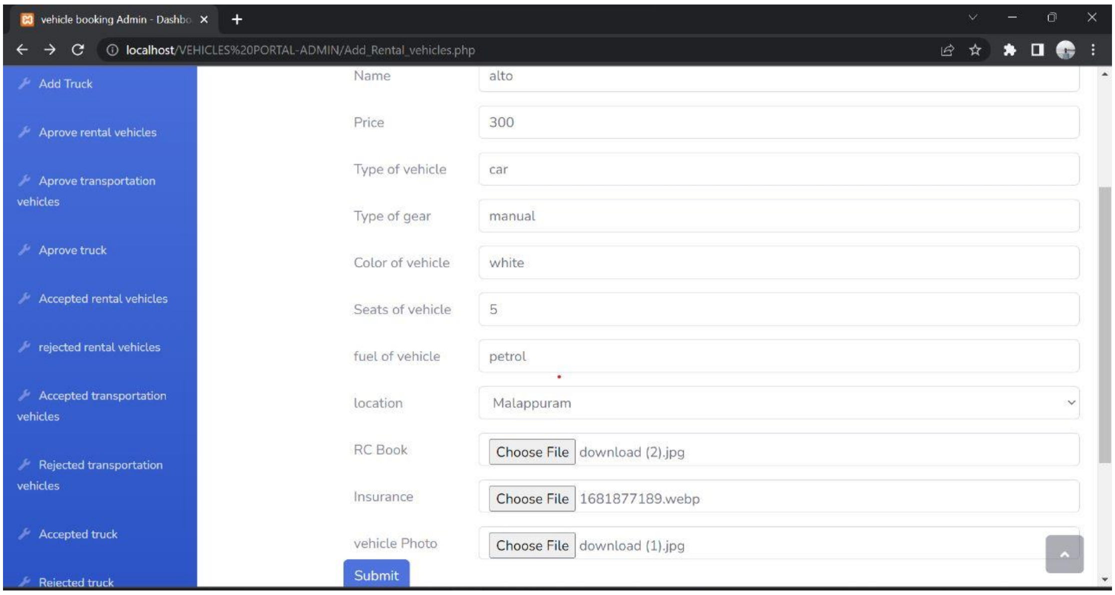
•••••

Home



vehicle booking Admin - Dashbo
localhost/VEHICLES%20PORTAL-ADMIN/Dash.php
ADMIN
Logout
Add Rental vehicles
Add Transportation vehicles
Add Truck
Aprove rental vehicles
Aprove transportation vehicles
Aprove truck
Accepted rental vehicles
rejected rental vehicles
localhost/VEHICLES PORTAL-ADMIN/Dash.php

Add Rental Vehicles


vehicle booking Admin - Dashbo
localhost/VEHICLES%20PORTAL-ADMIN/Add_Rental_vehicles.php
Add Truck
Aprove rental vehicles
Aprove transportation vehicles
Aprove truck
Accepted rental vehicles
rejected rental vehicles
Accepted transportation vehicles
Rejected transportation vehicles
Accepted truck
Rejected truck
Submit

Name: alto

Price: 300

Type of vehicle: car

Type of gear: manual

Color of vehicle: white

Seats of vehicle: 5

fuel of vehicle: petrol

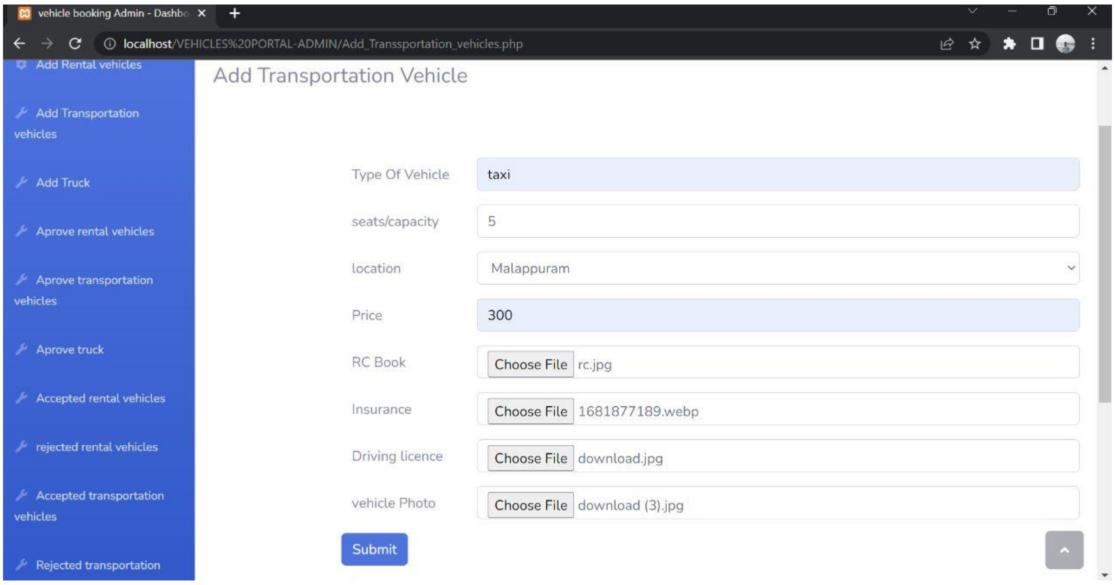
location: Malappuram

RC Book: Choose File download (2).jpg

Insurance: Choose File 1681877189.webp

vehicle Photo: Choose File download (1).jpg

Add Transportation Vehicles


vehicle booking Admin - Dashbo
localhost/VEHICLES%20PORTAL-ADMIN/Add_Transsportation_vehicles.php
Add Rental vehicles
Add Transportation vehicles
Add Truck
Aprove rental vehicles
Aprove transportation vehicles
Aprove truck
Accepted rental vehicles
rejected rental vehicles
Accepted transportation vehicles
Rejected transportation
Submit

Add Transportation Vehicle

Type Of Vehicle: taxi

seats/capacity: 5

location: Malappuram

Price: 300

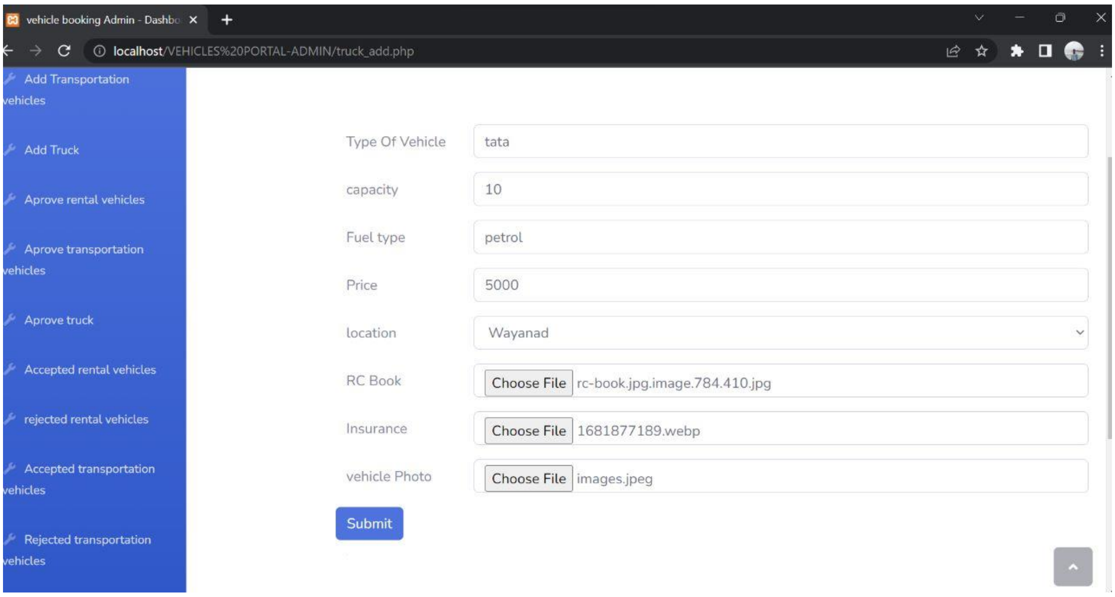
RC Book: Choose File rc.jpg

Insurance: Choose File 1681877189.webp

Driving licence: Choose File download.jpg

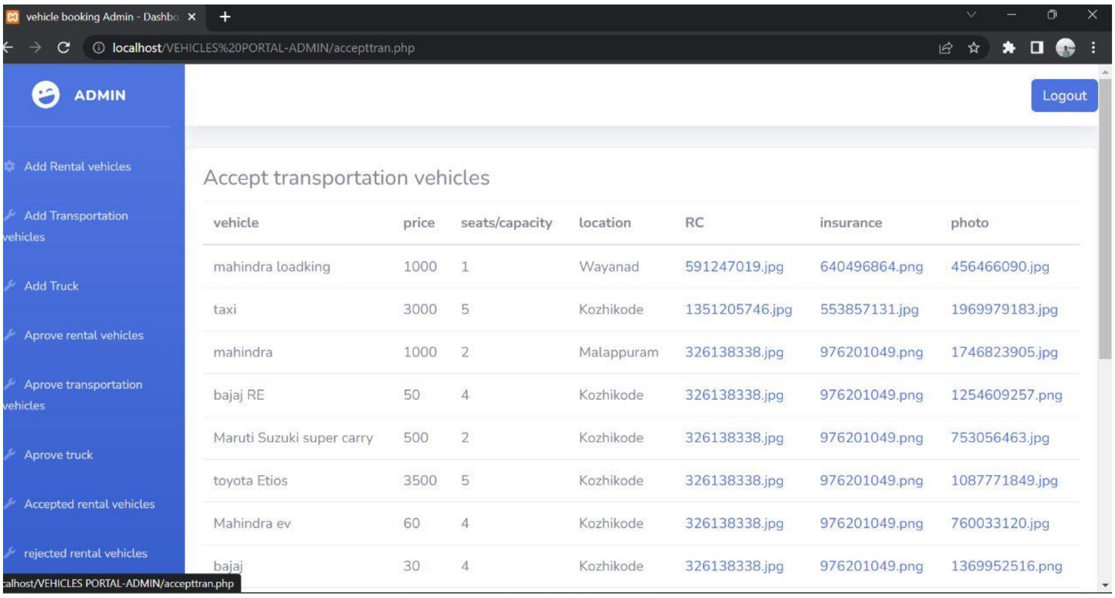
vehicle Photo: Choose File download (3).jpg

Add Truck


vehicle booking Admin - Dashbo
localhost/VEHICLES%20PORTAL-ADMIN/truck_add.php
Add Transportation vehicles
Add Truck
Aprove rental vehicles
Aprove transportation vehicles
Aprove truck
Accepted rental vehicles
rejected rental vehicles
Accepted transportation vehicles
Rejected transportation vehicles

Type Of Vehicle	tata
capacity	10
Fuel type	petrol
Price	5000
location	Wayanad
RC Book	Choose File rc-book.jpg.image.784.410.jpg
Insurance	Choose File 1681877189.webp
vehicle Photo	Choose File images.jpeg

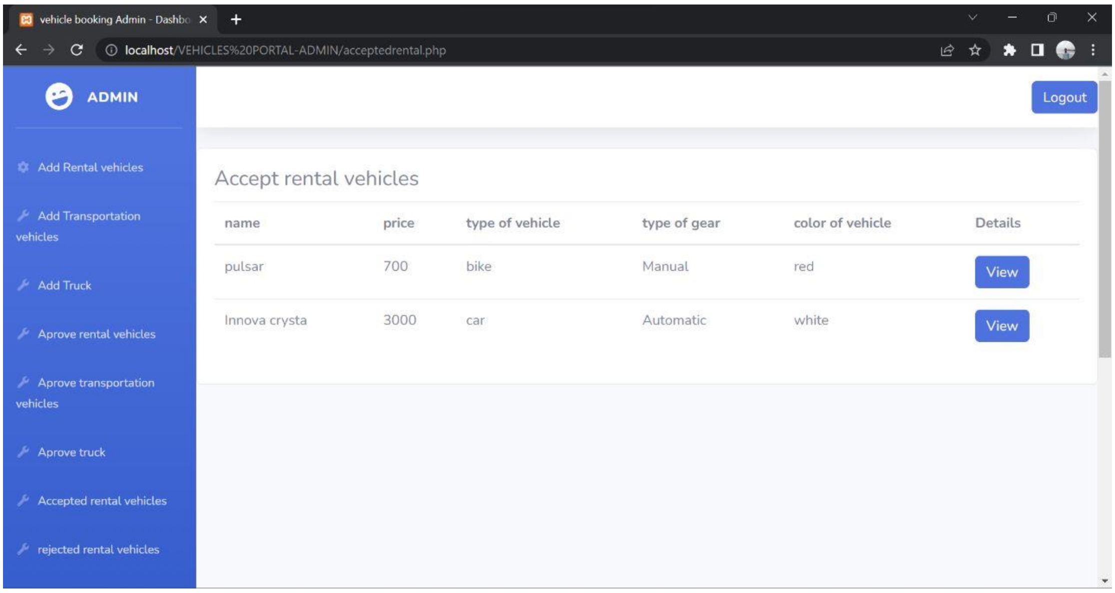
Accept Transportation Vehicles


vehicle booking Admin - Dashbo
localhost/VEHICLES%20PORTAL-ADMIN/accepttran.php
ADMIN
Logout
Add Rental vehicles
Add Transportation vehicles
Add Truck
Aprove rental vehicles
Aprove transportation vehicles
Aprove truck
Accepted rental vehicles
rejected rental vehicles
calhost/VEHICLES PORTAL-ADMIN/accepttran.php

Accept transportation vehicles

vehicle	price	seats/capacity	location	RC	insurance	photo
mahindra loadking	1000	1	Wayanad	591247019.jpg	640496864.png	456466090.jpg
taxi	3000	5	Kozhikode	1351205746.jpg	553857131.jpg	1969979183.jpg
mahindra	1000	2	Malappuram	326138338.jpg	976201049.png	1746823905.jpg
bajaj RE	50	4	Kozhikode	326138338.jpg	976201049.png	1254609257.png
Maruti Suzuki super carry	500	2	Kozhikode	326138338.jpg	976201049.png	753056463.jpg
toyota Etios	3500	5	Kozhikode	326138338.jpg	976201049.png	1087771849.jpg
Mahindra ev	60	4	Kozhikode	326138338.jpg	976201049.png	760033120.jpg
bajaj	30	4	Kozhikode	326138338.jpg	976201049.png	1369952516.png

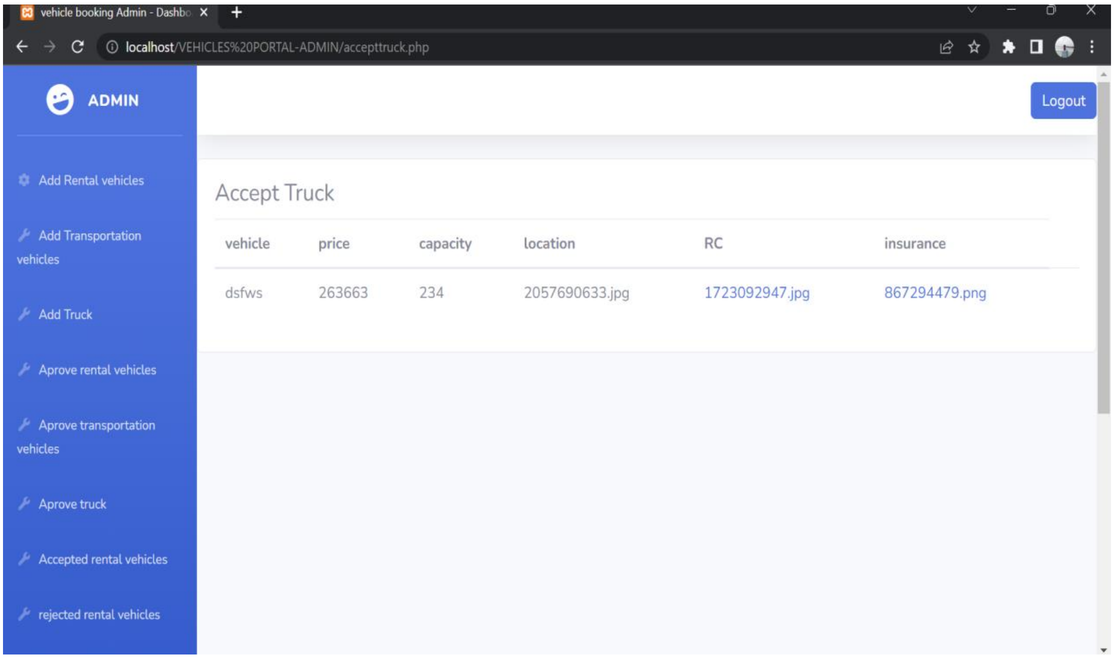
Accept Rental Vehicles


vehicle booking Admin - Dashbo
localhost/VEHICLES%20PORTAL-ADMIN/acceptedrental.php
ADMIN
Logout
Add Rental vehicles
Add Transportation vehicles
Add Truck
Aprove rental vehicles
Aprove transportation vehicles
Aprove truck
Accepted rental vehicles
rejected rental vehicles

Accept rental vehicles

name	price	type of vehicle	type of gear	color of vehicle	Details
pulsar	700	bike	Manual	red	View
Innova crysta	3000	car	Automatic	white	View

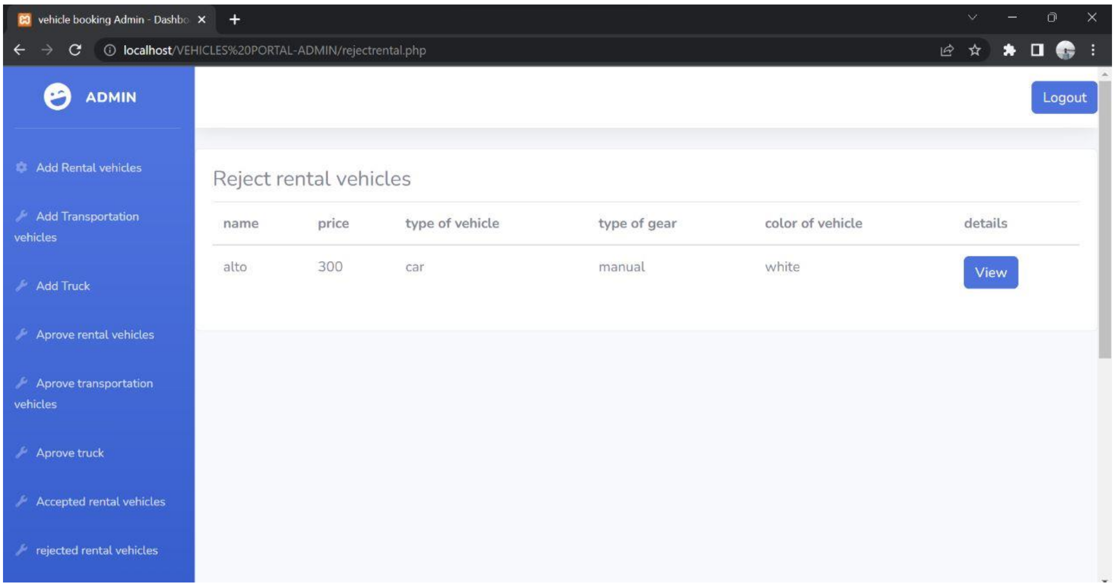
Accept Truck


vehicle booking Admin - Dashbo
localhost/VEHICLES%20PORTAL-ADMIN/accepttruck.php
ADMIN
Logout
Add Rental vehicles
Add Transportation vehicles
Add Truck
Aprove rental vehicles
Aprove transportation vehicles
Aprove truck
Accepted rental vehicles
rejected rental vehicles

Accept Truck

vehicle	price	capacity	location	RC	insurance
dsfws	263663	234	2057690633.jpg	1723092947.jpg	867294479.png

Reject Rental vehicles

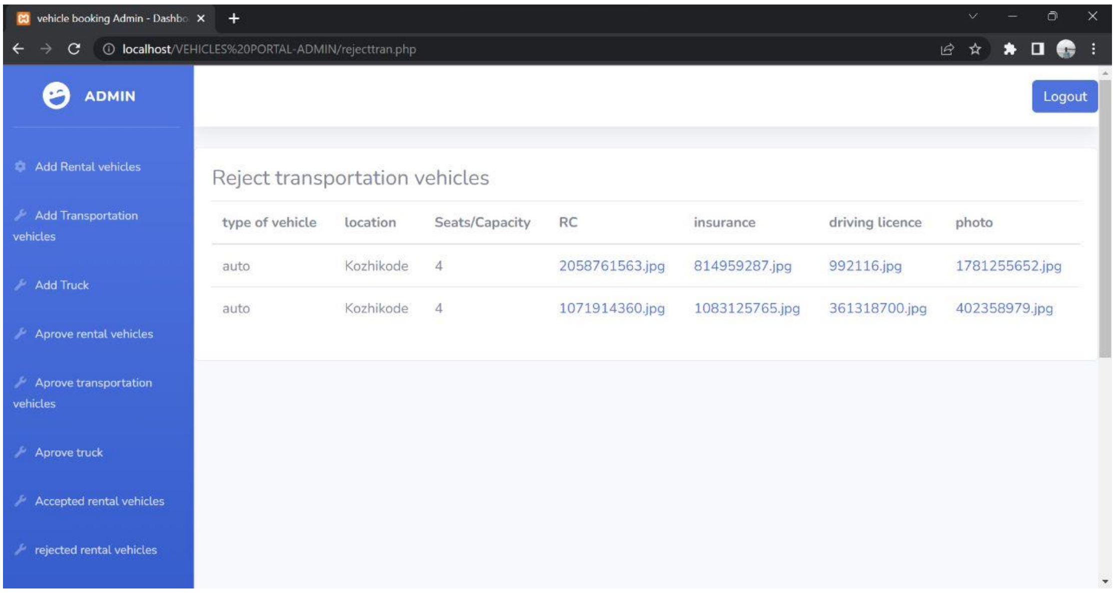


vehicle booking Admin - Dashbo
localhost/VEHICLES%20PORTAL-ADMIN/rejectrental.php
ADMIN
Logout
Add Rental vehicles
Add Transportation vehicles
Add Truck
Aprove rental vehicles
Aprove transportation vehicles
Aprove truck
Accepted rental vehicles
rejected rental vehicles

Reject rental vehicles

name	price	type of vehicle	type of gear	color of vehicle	details
alto	300	car	manual	white	View

Reject Transportation vehicles

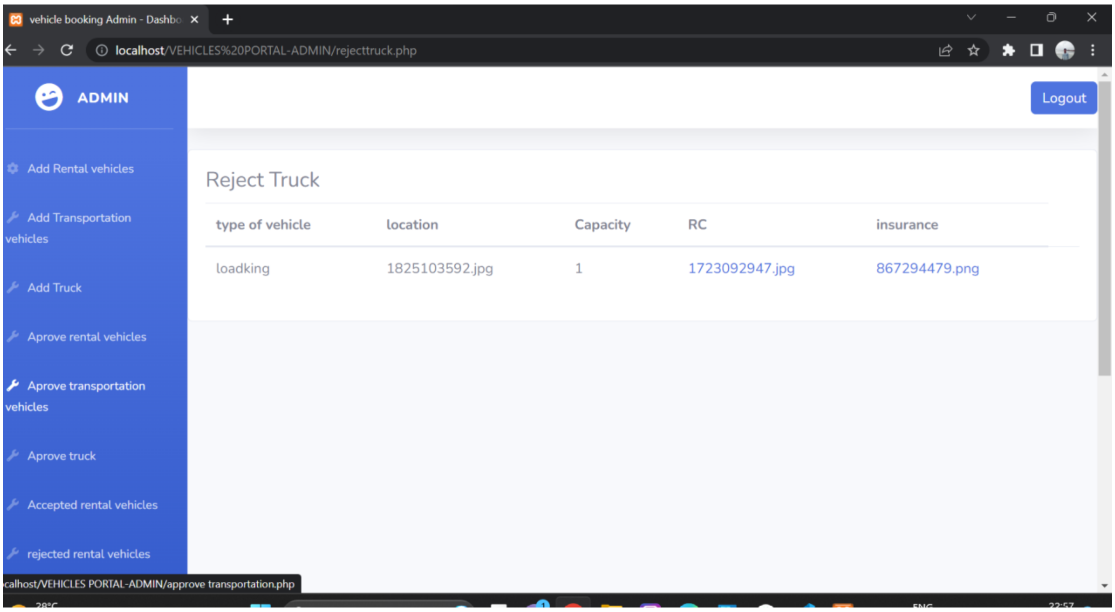


vehicle booking Admin - Dashbo
localhost/VEHICLES%20PORTAL-ADMIN/rejecttran.php
ADMIN
Logout
Add Rental vehicles
Add Transportation vehicles
Add Truck
Aprove rental vehicles
Aprove transportation vehicles
Aprove truck
Accepted rental vehicles
rejected rental vehicles

Reject transportation vehicles

type of vehicle	location	Seats/Capacity	RC	insurance	driving licence	photo
auto	Kozhikode	4	2058761563.jpg	814959287.jpg	992116.jpg	1781255652.jpg
auto	Kozhikode	4	1071914360.jpg	1083125765.jpg	361318700.jpg	402358979.jpg

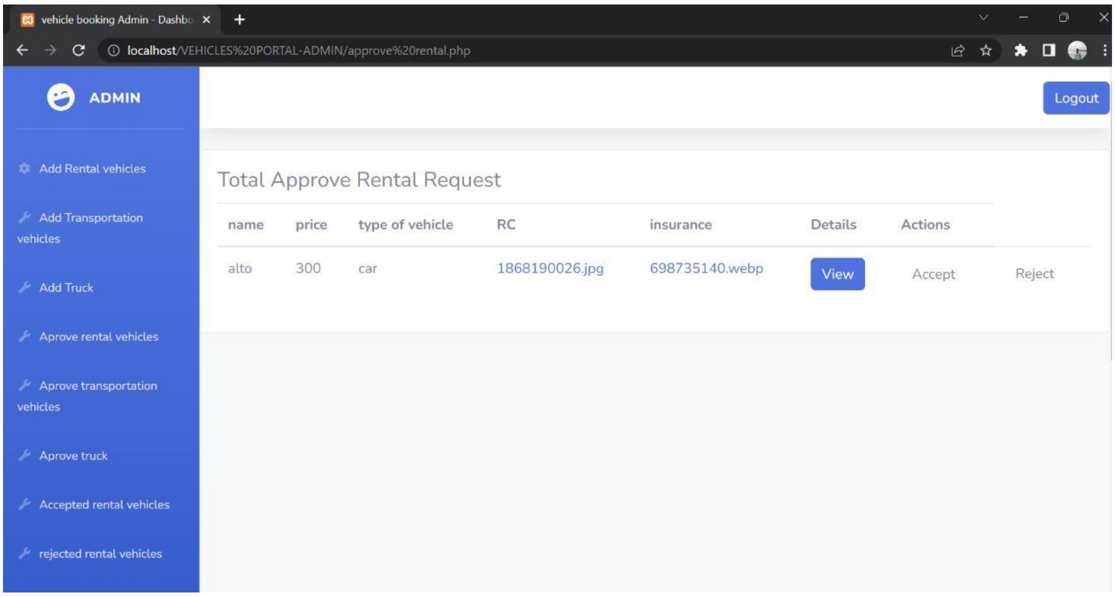
Reject Truck


vehicle booking Admin - Dashbo
localhost/VEHICLES%20PORTAL-ADMIN/rejecttruck.php
ADMIN
Logout
Add Rental vehicles
Add Transportation vehicles
Add Truck
Aprove rental vehicles
Aprove transportation vehicles
Aprove truck
Accepted rental vehicles
rejected rental vehicles
calhost/VEHICLES PORTAL-ADMIN/approve transportation.php

Reject Truck

type of vehicle	location	Capacity	RC	insurance
loadking	1825103592.jpg	1	1723092947.jpg	867294479.png

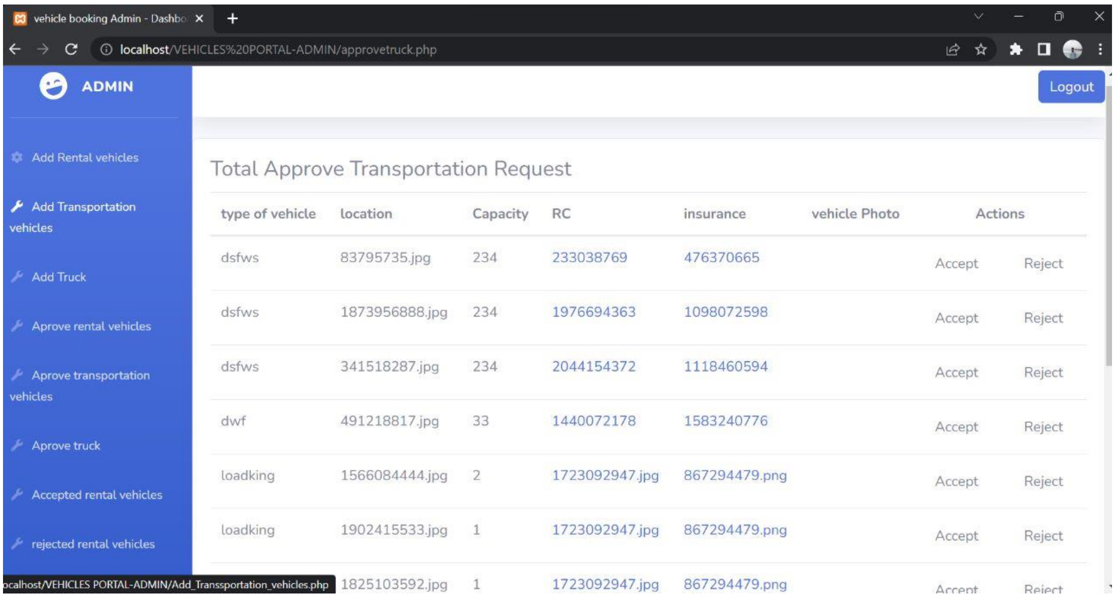
Total Approve Rental Vehicles


vehicle booking Admin - Dashbo
localhost/VEHICLES%20PORTAL-ADMIN/approve%20rental.php
ADMIN
Logout
Add Rental vehicles
Add Transportation vehicles
Add Truck
Aprove rental vehicles
Aprove transportation vehicles
Aprove truck
Accepted rental vehicles
rejected rental vehicles

Total Approve Rental Request

name	price	type of vehicle	RC	insurance	Details	Actions	
alto	300	car	1868190026.jpg	698735140.webp	View	Accept	Reject

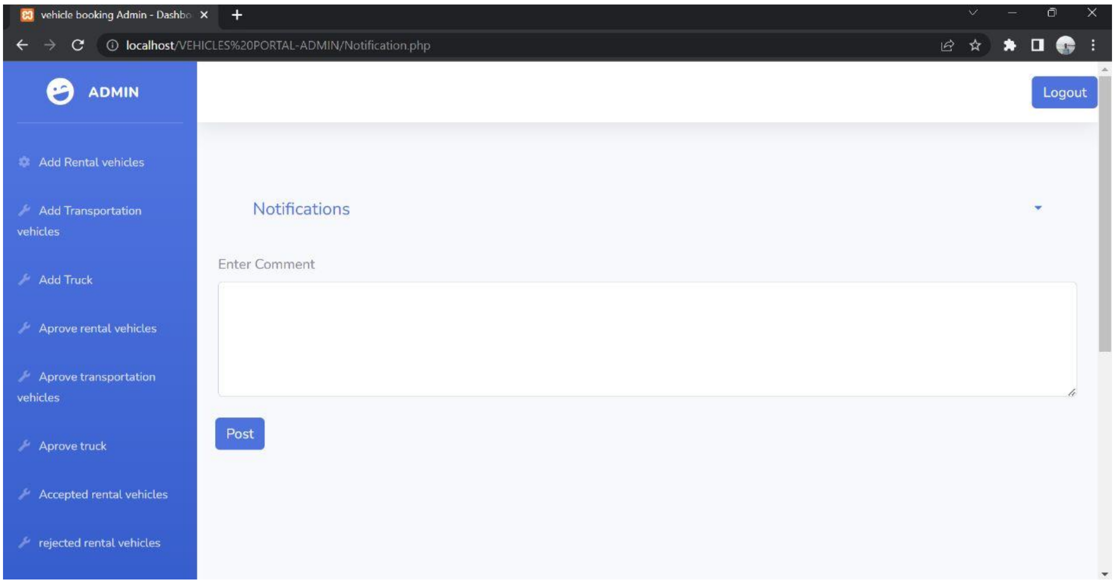
Total Approve Transportation Vehicles


vehicle booking Admin - Dashbo
localhost/VEHICLES%20PORTAL-ADMIN/approvetruck.php
ADMIN
Logout
Add Rental vehicles
Add Transportation vehicles
Add Truck
Aprove rental vehicles
Aprove transportation vehicles
Aprove truck
Accepted rental vehicles
rejected rental vehicles
ocalhost/VEHICLES PORTAL-ADMIN/Add_Transsportation_vehicles.php

Total Approve Transportation Request

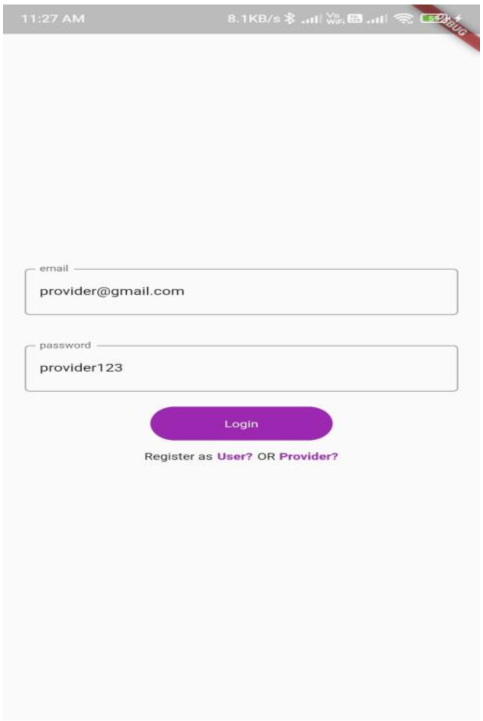
type of vehicle	location	Capacity	RC	insurance	vehicle Photo	Actions
dsfws	83795735.jpg	234	233038769	476370665		Accept Reject
dsfws	1873956888.jpg	234	1976694363	1098072598		Accept Reject
dsfws	341518287.jpg	234	2044154372	1118460594		Accept Reject
dwf	491218817.jpg	33	1440072178	1583240776		Accept Reject
loadking	1566084444.jpg	2	1723092947.jpg	867294479.png		Accept Reject
loadking	1902415533.jpg	1	1723092947.jpg	867294479.png		Accept Reject
	1825103592.jpg	1	1723092947.jpg	867294479.png		Accept Reject

Add Notification



Provider

Login



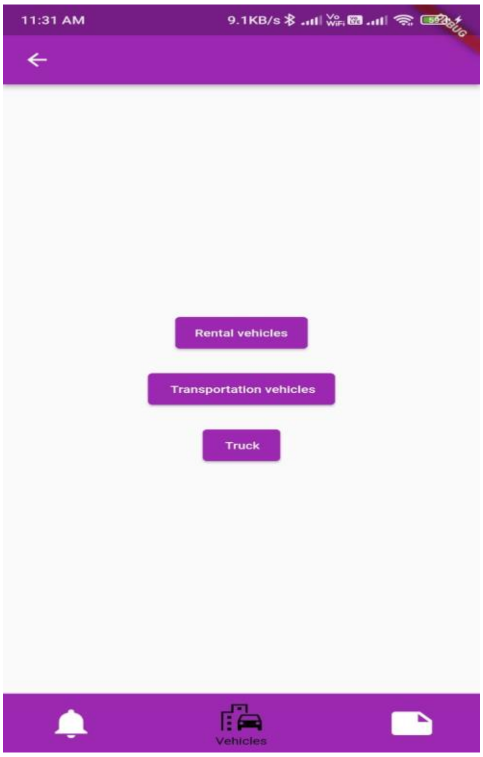
email

provider@gmail.com

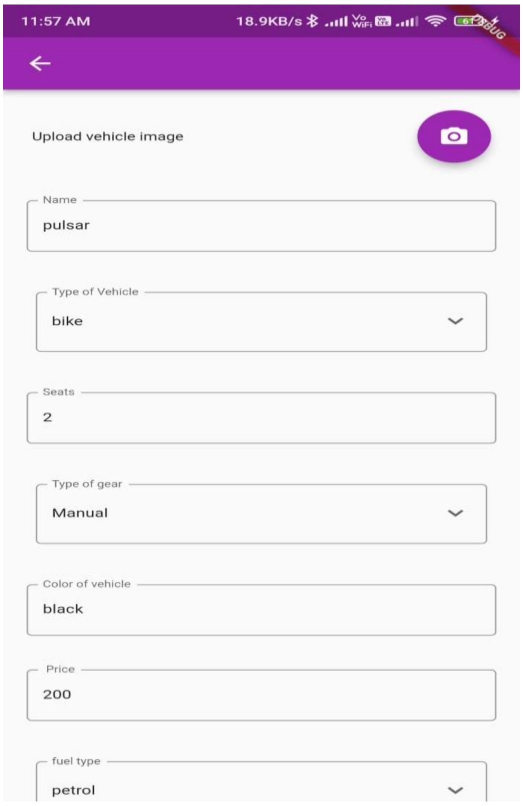
password

provider123

Home



Add Rental Vehicles



Upload vehicle image

Name: pulsar

Type of Vehicle: bike

Seats: 2

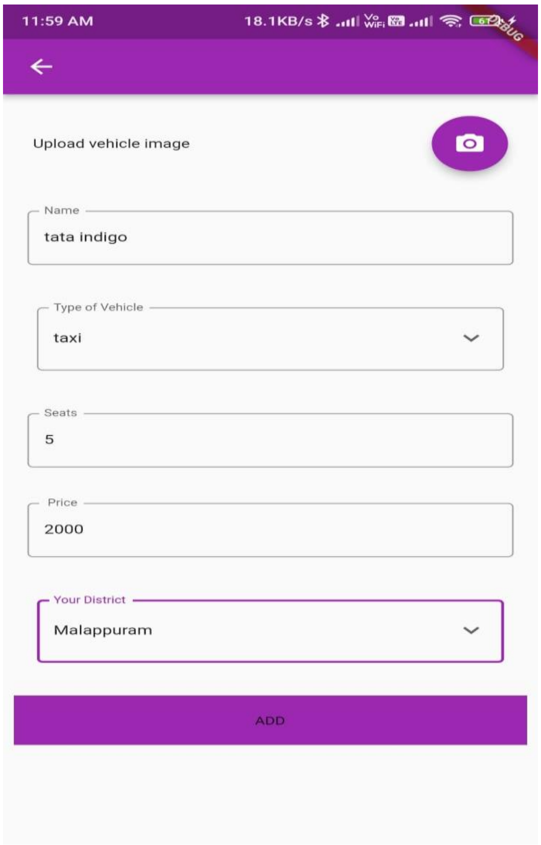
Type of gear: Manual

Color of vehicle: black

Price: 200

fuel type: petrol

Add Transportation Vehicles



Upload vehicle image

Name: tata indigo

Type of Vehicle: taxi

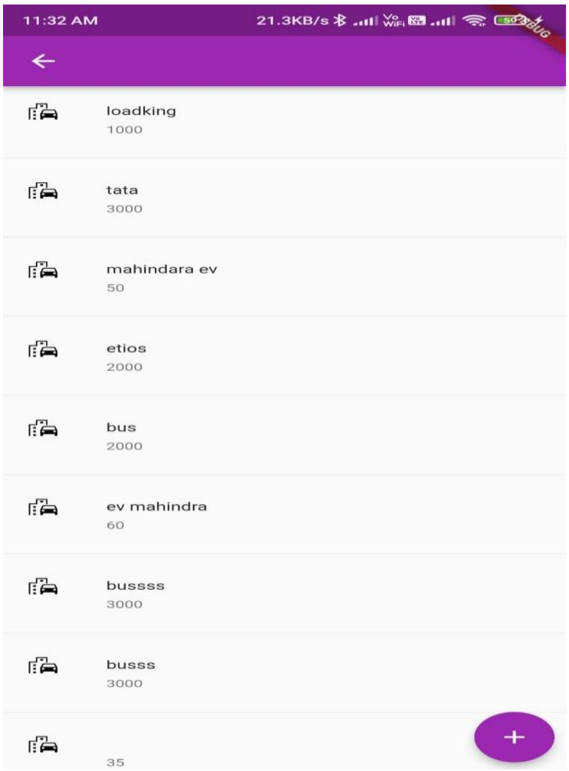
Seats: 5

Price: 2000

Your District: Malappuram

ADD

View Transportation Vehicles



11:32 AM 21.3KB/s

loadking
1000

tata
3000

mahindara ev
50

etios
2000

bus
2000

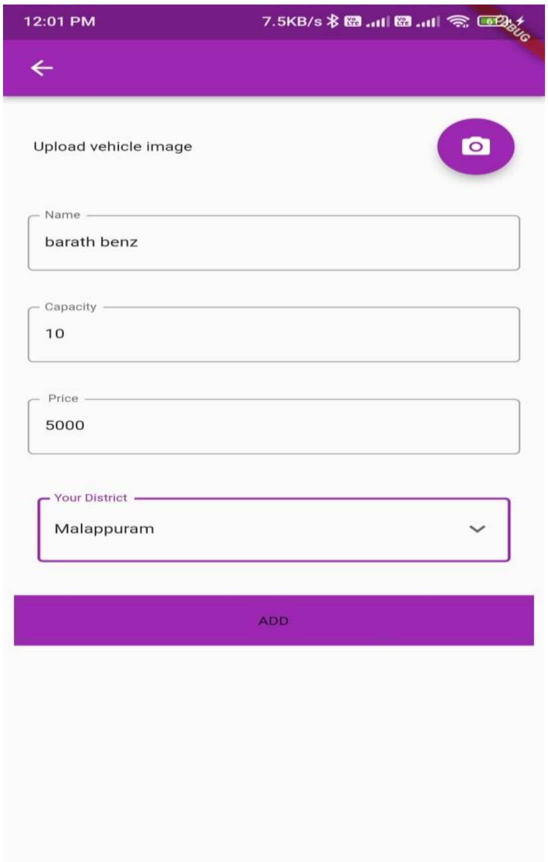
ev mahindra
60

bussss
3000

busss
3000

35

Add Truck Vehicles



12:01 PM 7.5KB/s

Upload vehicle image

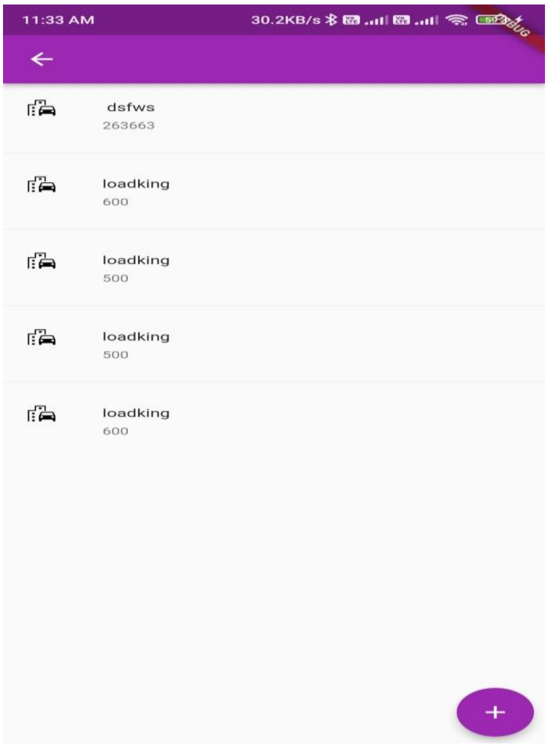
Name: barath benz

Capacity: 10

Price: 5000

Your District: Malappuram

View Truck Vehicles



dsfws
263663

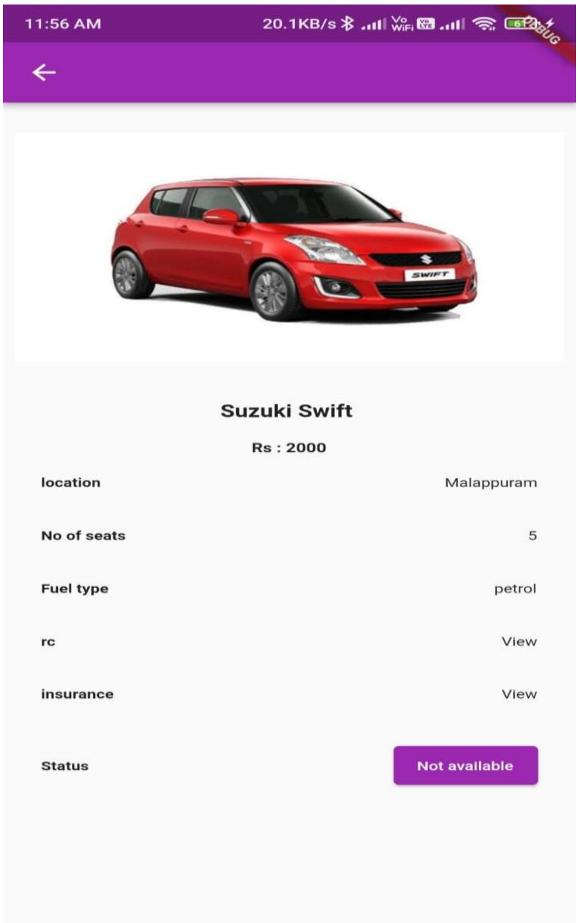
loadking
600

loadking
500

loadking
500

loadking
600

View Notification

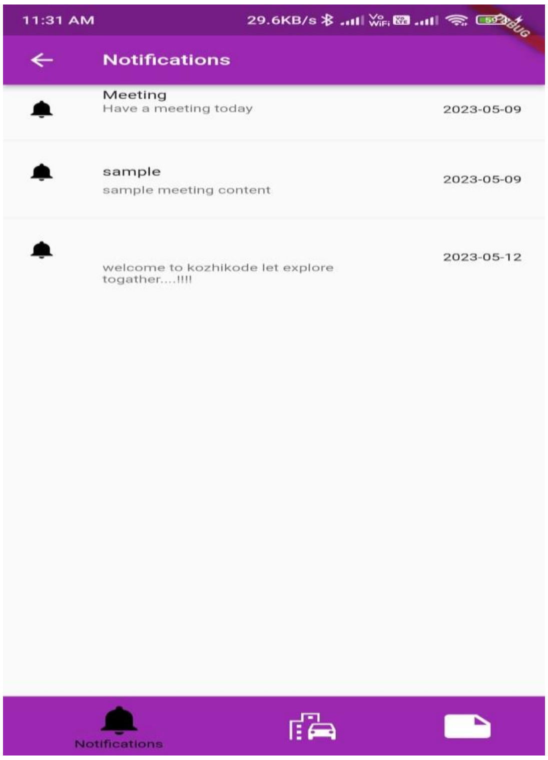


Suzuki Swift

Rs : 2000

location	Malappuram
No of seats	5
Fuel type	petrol
rc	View
insurance	View
Status	Not available

View Notification



11:31 AM 29.6KB/s

Notifications

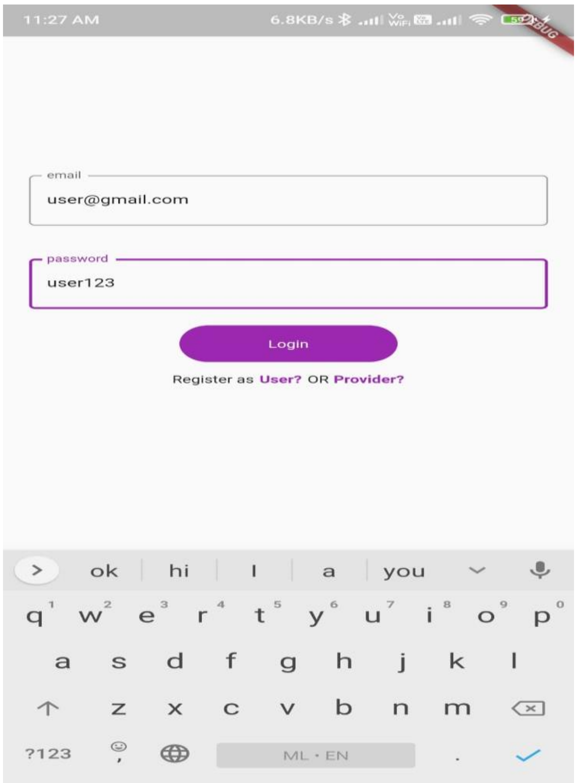
Meeting
Have a meeting today 2023-05-09

sample
sample meeting content 2023-05-09

welcome to kozhikode let explore togather....!!!! 2023-05-12

User

Login

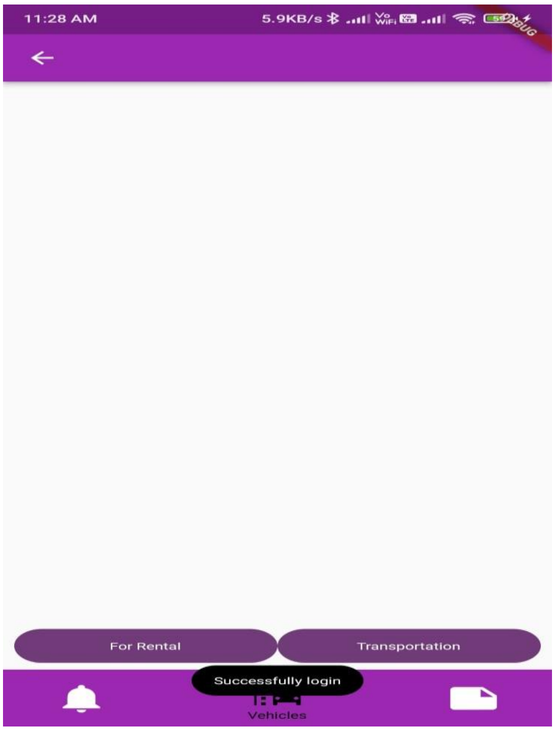


11:27 AM 6.8KB/s

email: user@gmail.com

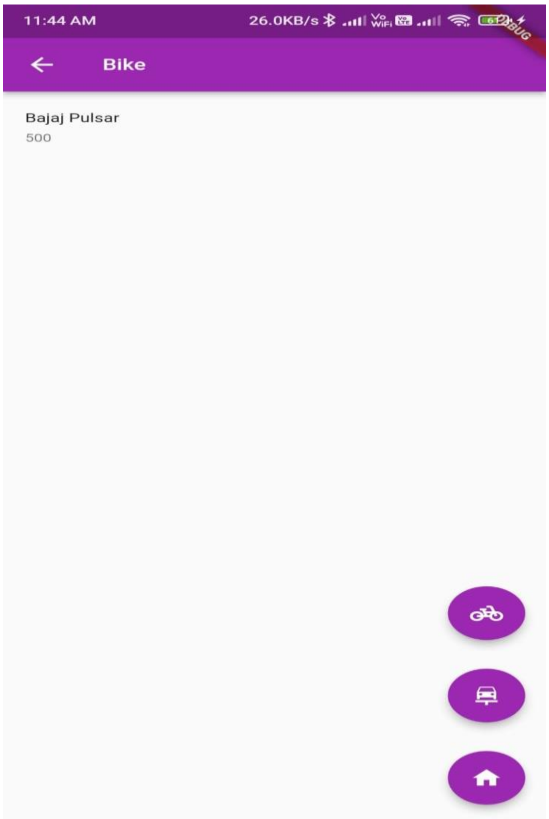
password: user123

Home



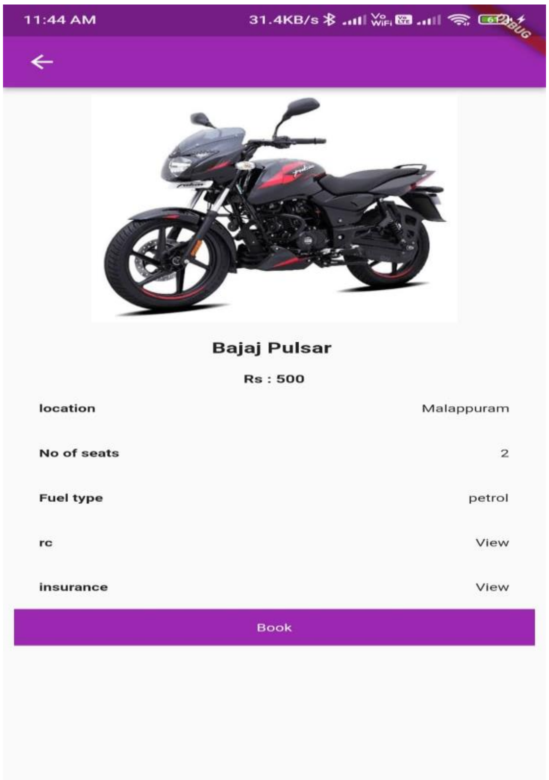
Successfully login

Rental Vehicles



Bajaj Pulsar
500

Vehicle View

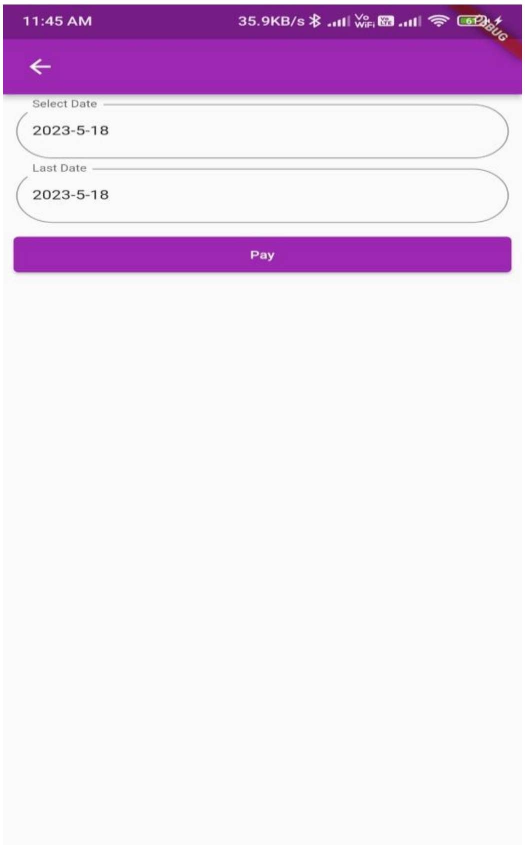


Bajaj Pulsar

Rs : 500

location	Malappuram
No of seats	2
Fuel type	petrol
rc	View
insurance	View

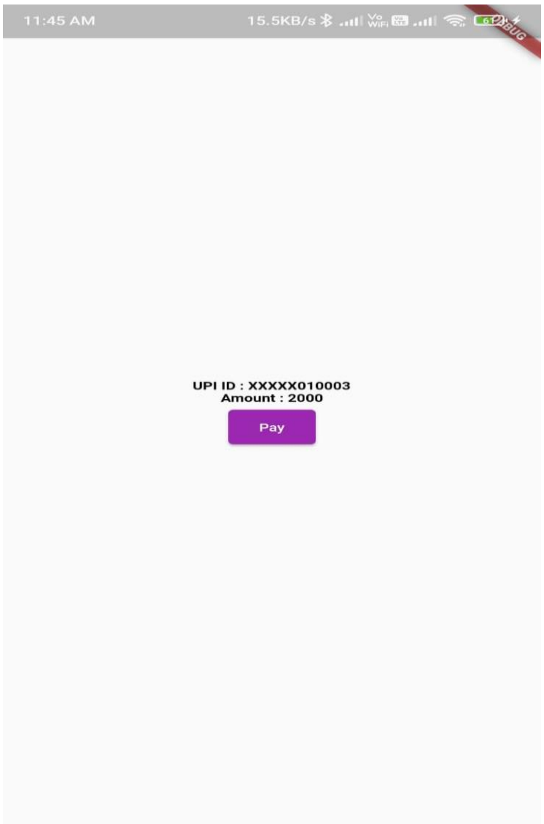
Rental Booking Date



Select Date: 2023-5-18

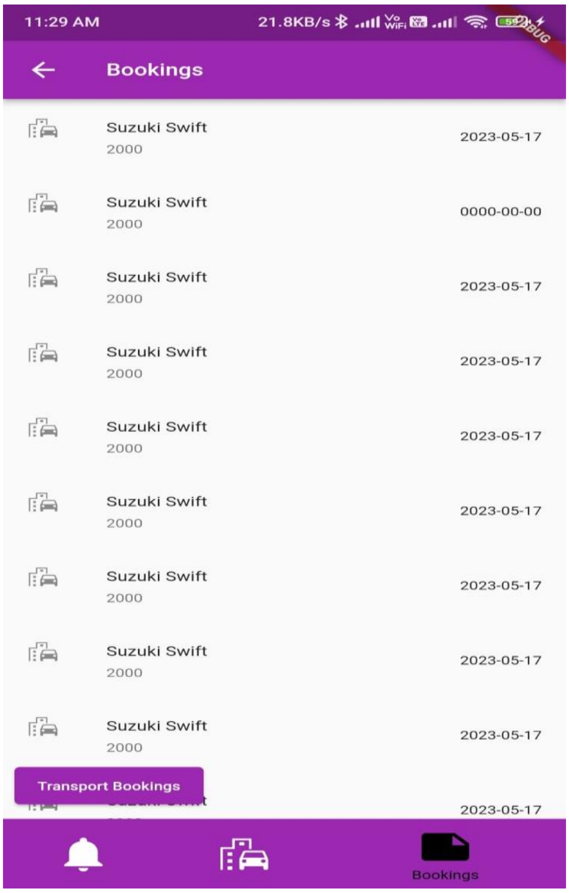
Last Date: 2023-5-18

Payment



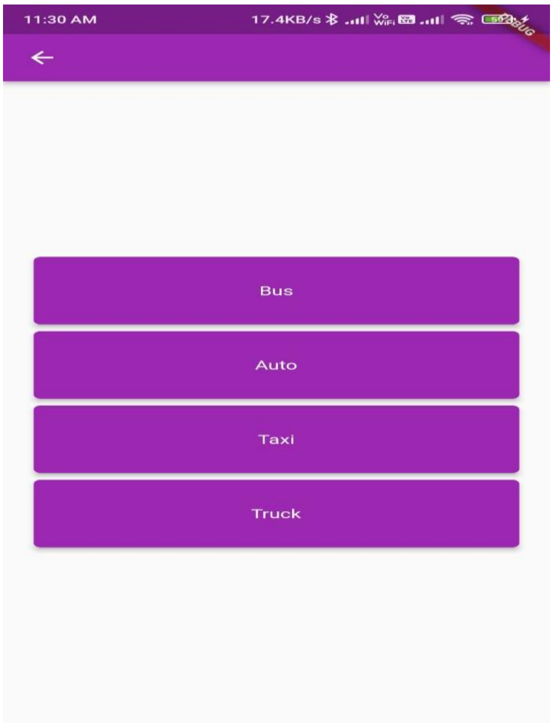
UPI ID : XXXXX010003
Amount : 2000

User Rental Booking

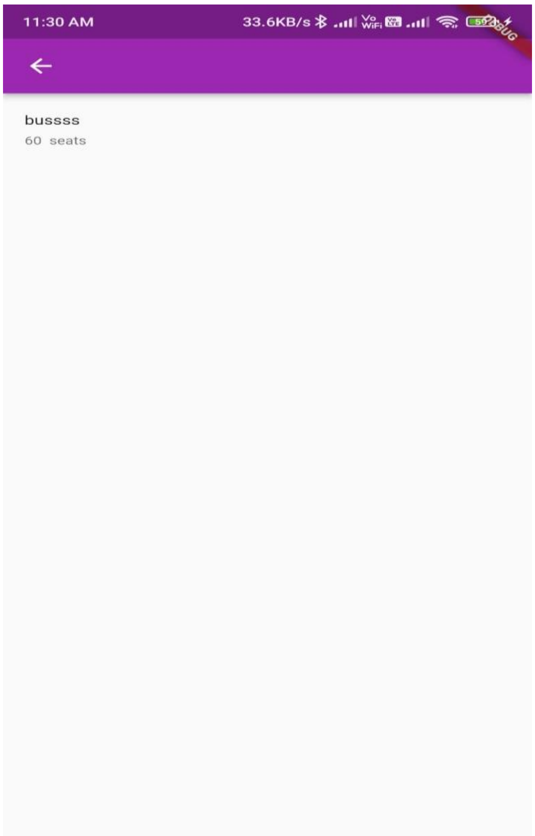


Suzuki Swift 2000	2023-05-17
Suzuki Swift 2000	0000-00-00
Suzuki Swift 2000	2023-05-17
Suzuki Swift 2000	2023-05-17
Suzuki Swift 2000	2023-05-17
Suzuki Swift 2000	2023-05-17
Suzuki Swift 2000	2023-05-17
Suzuki Swift 2000	2023-05-17
Suzuki Swift 2000	2023-05-17
	2023-05-17

Transportation

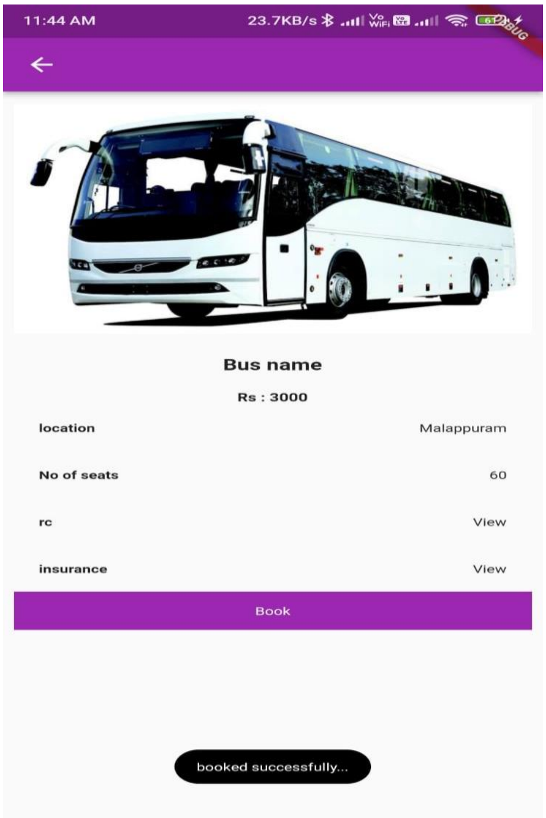


Bus List



bussss
60 seats

Bus View



11:44 AM

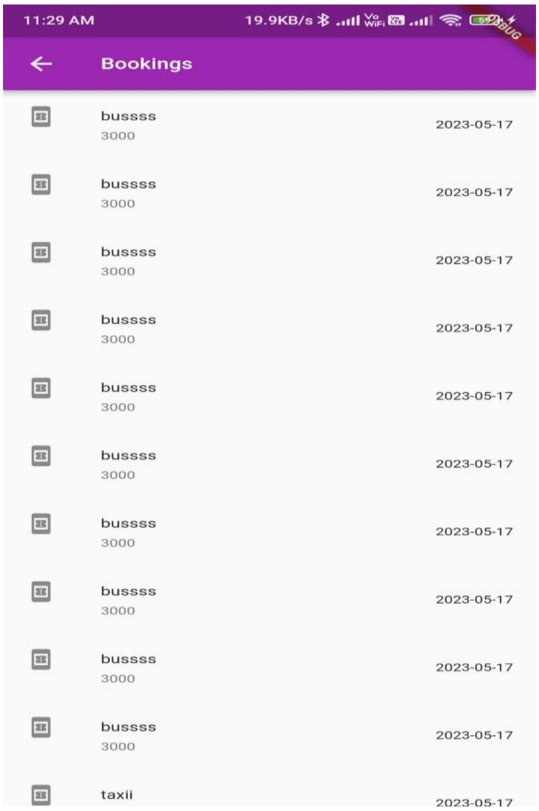
Bus name

Rs : 3000

location	Malappuram
No of seats	60
rc	View
insurance	View

booked successfully...

Transportation Booking

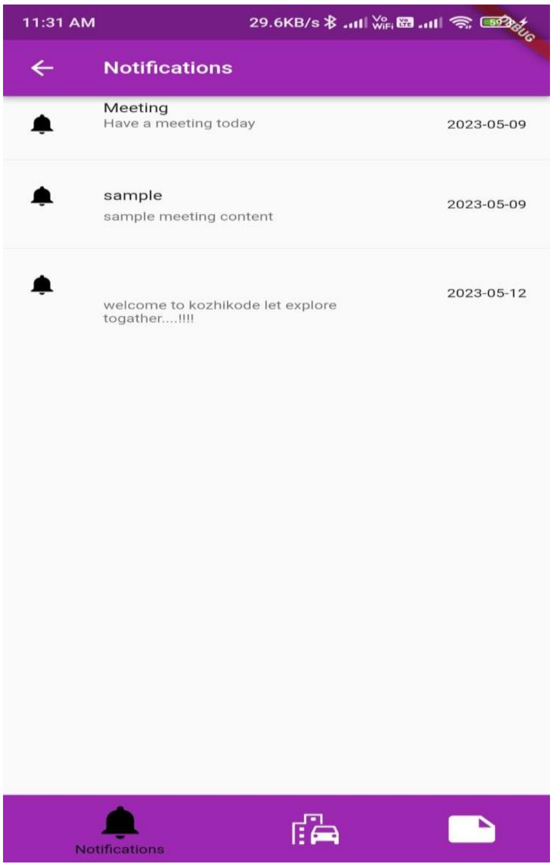


11:29 AM

Bookings

bussss 3000	2023-05-17
bussss 3000	2023-05-17
bussss 3000	2023-05-17
bussss 3000	2023-05-17
bussss 3000	2023-05-17
bussss 3000	2023-05-17
bussss 3000	2023-05-17
bussss 3000	2023-05-17
bussss 3000	2023-05-17
bussss 3000	2023-05-17
taxii	2023-05-17

Notification



Meeting
Have a meeting today
2023-05-09

sample
sample meeting content
2023-05-09

welcome to kozhikode let explore togather....!!!!
2023-05-12

BIBLIOGRAPHY

13.BIBLIOGRAPHY

Websites

[1] https://www.w3schools.com

[2] https://stackoverflow.com

[3] https://www.javapoint.com/mysql-tutorial

[4] https://www.geeksforgeeks.org

[5] https://docs.flutter.dev/packages-and-plugins/using-packages

Books

[1] Android Programming: The Big Nerd Ranch Guide (5th Edition)

[2] Learn Google Flutter Fast: 65 Example Apps Paperback – April 1, 2019

[3] Apps Beginning App Development with Flutter: Create Cross-Platform

[4] Beginning Flutter by Marco L. Napoli Released October 2019

[5] Practical Flutter: Improve your Mobile Development with Google's Latest Open-Source SDK by Frank W. Zammetti