



**Daffodil**  
*International*  
**University**

**“Easy Shifter”**

**A Smart Web Solution to shift your goods.**

**Supervised by**

Ms. Fatama Binta Rafiq  
Lecturer

Department of Software Engineering  
Faculty of Science & Information Technology  
Daffodil International University

**Submitted by**

Ahmad Musa

ID: 151-35-1030

B.Sc. in Software Engineering  
Daffodil International University

## APPROVAL

This **Project** titled “**Easy Shifter**”, submitted by **Ahmad Musa, 151-35-1030** to the Department of Software Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc in Software Engineering and approved as to its style and contents.

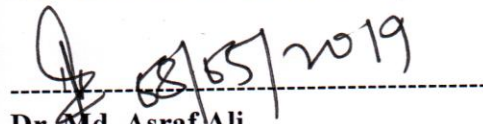
## BOARD OF EXAMINERS



**Dr. Touhid Bhuiyan**  
**Professor and Head**

Department of Software Engineering  
Faculty of Science and Information Technology  
Daffodil International University

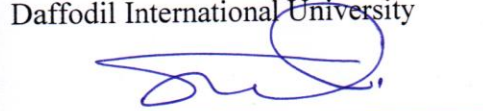
**Chairman**



**Dr. Md. Asraf Ali**  
**Associate Professor**

Department of Software Engineering  
Faculty of Science and Information Technology  
Daffodil International University

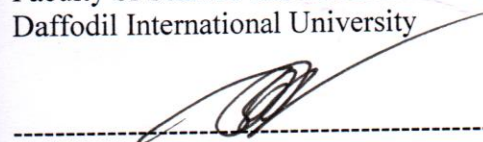
**Internal Examiner 1**



**Mohammad Khaled Sohel**  
**Assistant Professor**

Department of Software Engineering  
Faculty of Science and Information Technology  
Daffodil International University

**Internal Examiner 2**



**Prof Dr. Mohammad Abul Kashem**  
**Professor**

Department of Computer Science and Engineering  
Faculty of Electrical and Electronic Engineering  
Dhaka University of Engineering & Technology, Gazipur

**External Examiner**

## **ACKNOWLEDGEMENT**

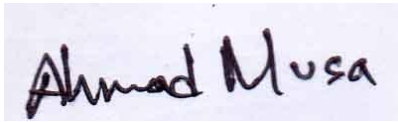
It stands to reason that the completion of main project needs the support of many people. I take this as an opportunity to express my heartiest thanks and to each and everyone, who helped me a lot in successful completion of my final project. I am really happy to acknowledge the help of all the individuals to fulfill this attempt.

I would like to take the opportunity to express my gratitude to Ms. Fatama Binta Rafiq, Lecturer. Department of Software Engineering, Daffodil International University. Though she was taken with many other causes, she gave me more than enough time to fulfill this work. She just not only gave time, but also enlightened me with her proper guidance and valuable advice whenever I got stuck with any kind of difficulty. Her guidance helped me a lot during this project.

Finally, I heartily thank to my parents for supporting me throughout all my studies at Daffodil International University. Thanks to my teachers and friends.

# DECLARATION

I have taken this project under the supervision of Ms. Fatama Binta Rafiq, Lecturer, Department of Software Engineering, Daffodil International University, I proclaim that neither this report nor any part of this has been submitted elsewhere for award of any degree.



.....

Ahmad Musa

ID: 151-35-1030

Batch: 16<sup>th</sup>

Department of Software Engineering

Certified by:



.....

Ms. Fatama Binta Rafiq

Lecturer

Department of Software Engineering

Faculty of Science & Information Technology

Daffodil International University

## **ABSTRACT**

This project entitled “Easy Shifter” is a web based application for hiring trucks and pickups from online at a reasonable cost by giving much less effort. People face many issues when they need to shift their goods or products from one place to another. Among those, hiring a vehicle at a reasonable price is the most troublesome one. “Easy Shifter” will help them to get rid of those hazards. People can hire big, mid or small size trucks or pickups from here at a reasonable cost without any effort. Users can also give their valuable opinion about our service by using the comment or feedback section.

I’ve used Laravel 5.6 (PHP framework), HTML 5, CSS 3 and Bootstrap for this website. I also used Client side scripting language: JavaScript to make it more user friendly.

# LIST OF CONTENTS

|                          |     |
|--------------------------|-----|
| Acknowledgement .....    | i   |
| Abstract .....           | ii  |
| Declaration .....        | iii |
| List of contents .....   | iv  |
| List of the figure ..... | vi  |
| List of the table .....  | vii |

## Chapter

|                                             |       |
|---------------------------------------------|-------|
| 1. INTRODUCTION .....                       | 1-5   |
| 1.1 Project Overview .....                  | 1     |
| 1.2 Purpose.....                            | 1     |
| 1.2.1 Background.....                       | 1     |
| 1.2.2 Benefits and Beneficiaries.....       | 1     |
| 1.2.3 Goal.....                             | 2     |
| 1.3 Stakeholders .....                      | 2     |
| 1.4 Project schedule .....                  | 2     |
| 1.4.1 Initial Step.....                     | 2     |
| 1.4.2 Idea Proposal .....                   | 3     |
| 1.4.3 Requirement Gathering.....            | 3     |
| 1.4.4 Physical System Design.....           | 4     |
| 1.4.5 Logical System design .....           | 4     |
| 1.4.6 Development Phase.....                | 5     |
| 1.4.7 System Testing.....                   | 5     |
| 2. SOFTWARE REQUIREMENT SPECIFICATION ..... | 6-8   |
| 2.1 SRS.....                                | 6     |
| 2.2 Studying similar projects.....          | 6     |
| 2.3 Expert's opinion.....                   | 6     |
| 2.4 Brainstorming.....                      | 6     |
| 2.5 Software Requirement Specification..... | 7     |
| 2.6 Used Technology.....                    | 7     |
| 3. SYSTEM ANALYSIS.....                     | 9-10  |
| 3.1 Use Case Diagram.....                   | 9     |
| 3.2 Activity Diagram.....                   | 10    |
| 4. SYSTEM DESIGN SPECIFICATION.....         | 11-23 |
| 4.1 Sequence Diagram.....                   | 11    |
| 4.2 Class Diagram.....                      | 20    |
| 4.3 Database Design Diagram.....            | 21    |
| 4.4 Data Flow Diagram.....                  | 22    |
| 5. SYSTEM TESTING.....                      | 24-25 |
| 6. USER MANUAL.....                         | 26-30 |

|                       |    |
|-----------------------|----|
| 7. CONCLUSION.....    | 31 |
| 7.1 Future Scope..... | 31 |
| 7.2 Limitations.....  | 31 |
| 7.3 GitHub Link.....  | 31 |

## LIST OF FIGURE

| Figure                                           | Page |
|--------------------------------------------------|------|
| <b>Figure 3.1:</b> Use case diagram .....        | 16   |
| <b>Figure 3.2:</b> Activity diagram .....        | 17   |
| <b>Figure 4.1:</b> Sequence diagram .....        | 18   |
| <b>Figure 4.2:</b> Class diagram .....           | 26   |
| <b>Figure 4.3:</b> Database Design diagram ..... | 27   |
| <b>Figure 4.4:</b> Data Flow Diagram .....       | 28   |
| <b>Figure 6.1:</b> Home Page.....                | 32   |
| <b>Figure 6.2:</b> Deals .....                   | 32   |
| <b>Figure 6.3:</b> Available Vehicles.....       | 33   |
| <b>Figure 6.4:</b> Registration .....            | 33   |
| <b>Figure 6.5:</b> Login .....                   | 34   |
| <b>Figure 6.6:</b> Ordering Vehicles.....        | 34   |
| <b>Figure 6.7:</b> Owner's Panel.....            | 35   |
| <b>Figure 6.8:</b> Feedback .....                | 36   |

## LIST OF TABLES

| Table                                                           | Page |
|-----------------------------------------------------------------|------|
| <b>Table 1.4.1</b> Initial Step.....                            | 10   |
| <b>Table 1.4.2</b> Idea Proposal.....                           | 11   |
| <b>Table 1.4.3</b> Requirement Gathering .....                  | 11   |
| <b>Table 1.4.4</b> Physical System Design .....                 | 12   |
| <b>Table 1.4.5</b> Logical System design .....                  | 12   |
| <b>Table 1.4.6</b> Development Phase .....                      | 13   |
| <b>Table 1.4.7</b> System Testing .....                         | 13   |
| <b>Table 2.5</b> Software Requirement Specification .....       | 15   |
| <b>Table 5.1</b> Unit Test Case for “Authentication” .....      | 30   |
| <b>Table 5.2</b> Unit Test Case for “Adding vehicles” .....     | 31   |
| <b>Table 5.3</b> Unit Test Case for “Ordering vehicles” .....   | 31   |
| <b>Table 5.4</b> Unit Test Case for “Customer’s feedback” ..... | 3    |

# CHAPTER 1

## INTRODUCTION

### 1.1 Project Overview

This project entitled “Easy Shifter” is a web based application for hiring trucks or pickups from online just only by sitting home. People face many issues when they need to shift their goods or products from one place to another. Among those, hiring a vehicle at a reasonable price is the most troublesome one. “Easy Shifter” will help them to get rid of those hazards. People can hire big, mid or small size trucks or pickups from here at a reasonable cost without any effort. Users can also give their opinion about our service by using the comment or feedback section.

I’ve used Laravel 5.6 (PHP framework), HTML 5, CSS 3 and Bootstrap for this website. I also used Client side scripting language: JavaScript to make it more user friendly.

### 1.2 Purpose

My purpose of building this system was to free people from so much trouble of hiring a vehicle for shifting.

#### 1.2.1 Background

As a tenant, I need to shift my furniture quite often. Hiring a vehicle at a reasonable price is so much trouble. And if I’m to find a vehicle at a reasonable cost, I’d have to give so much effort. That’s why I first thought about this project. And I think it’s an effective solution for reducing people’s effort.

#### 1.2.2 Benefits and Beneficiaries

##### Benefits:

- i) Easy to use.
- ii) Safe and secure.
- iii) Saves a lots of time, money and labor.
- iv) Reduces efforts.
- v) Fully functional and flexible system.

**Beneficiaries:**

The primary beneficiaries will be the users and vehicle owners.

**1.2.3 Goal**

Implementing a system that is user friendly, visually appealing and easy to use. A system that'll reduce people's effort and save their valuable time. In future I've a plan to develop this system in android platform and I want to add online payment system in this application.

**1.3 Stakeholder**

This system would be used by only customers and vehicle owners. So this system has two stakeholders.

**1.4 Project Schedule****1.4.1 Initial Step**

| Serial | Work Description          | Start (date) | End (date) |
|--------|---------------------------|--------------|------------|
| 1      | Idea Finding              | 01/07/2018   | 08/08/2018 |
| 2      | Feasibility study         | 08/08/2018   | 12/08/2018 |
| 3      | Available Source<br>Check | 13/08/2018   | 18/09/2018 |
| 4      | Mind Mapping              | 19/09/2018   | 26/09/2018 |
| 5      | Similar site analysis     | 27/09/2018   | 30/09/2018 |

### 1.4.2 Idea Proposal

| Serial     | Work Description                   | Start (date) | End (date) | Total day |
|------------|------------------------------------|--------------|------------|-----------|
| 1          | Idea Finding With Supervisor       | 01/09/2018   | 04/09/2018 | 3         |
| 2          | Feasibility study With Supervisor  | 05/09/2018   | 07/09/2018 | 2         |
| 3          | Feature Discussion With Supervisor | 08/09/2018   | 10/09/2018 | 2         |
| 4          | Work Flow maintenance              | 11/09/2018   | 16/09/2018 | 5         |
| 5          | SDLC Selection                     | 17/09/2018   | 19/09/2018 | 2         |
| Total Days |                                    |              |            | 14        |

### 1.4.3 Requirement Gathering

| Serial     | Work Description                          | Start (date) | End (date) | Total day |
|------------|-------------------------------------------|--------------|------------|-----------|
| 1          | System Flow sketch                        | 20/08/2018   | 28/08/2018 | 8         |
| 2          | Requirement gathering for proposed system | 29/08/2018   | 05/09/2018 | 7         |
| 3          | Requirement Collection                    | 06/09/2018   | 08/09/2018 | 2         |
| 4          | SRS                                       | 09/09/2018   | 11/09/2018 | 2         |
| 5          | all requirement and Information           | 12/09/2018   | 16/09/2018 | 4         |
| Total Days |                                           |              |            | 23        |

#### 1.4.4 Physical System Design

| Serial     | Work Description               | Start (date) | End (date) | Total day |
|------------|--------------------------------|--------------|------------|-----------|
| 1          | Designing Prototype            | 17/08/2018   | 22/08/2018 | 5         |
| 2          | GUI (Graphical user Interface) | 23/08/2018   | 07/08/2018 | 14        |
| 3          | Process Design                 | 07/08/2018   | 10/08/2018 | 3         |
| Total Days |                                |              |            | 22        |

#### 1.4.5 Logical System design

| Serial     | Work Description        | Start (date) | End (date) | Total day |
|------------|-------------------------|--------------|------------|-----------|
| 1          | Use case Diagram Design | 09/08/2018   | 10/08/2018 | 1         |
| 2          | Activity Diagram        | 11/08/2018   | 12/08/2018 | 1         |
| 3          | Schema Diagram          | 13/08/2018   | 14/08/2018 | 1         |
| 4          | Class Diagram           | 15/08/2018   | 16/08/2018 | 1         |
| Total Days |                         |              |            | 4         |

#### 1.4.6 Development Phase

| Serial     | Work Description        | Start (date) | End (date) | Total day |
|------------|-------------------------|--------------|------------|-----------|
| 1          | Customer module         | 01/01/2019   | 15/01/2019 | 15        |
| 2          | Owner module            | 16/01/2019   | 31/01/2019 | 15        |
| 3          | Admin module            | 01/02/2019   | 15/02/2019 | 15        |
| 4          | Multiple authentication | 15/02/2019   | 30/02/2019 | 15        |
| 5          | Database integration    |              |            |           |
| 6          | Refactoring code        |              |            |           |
| Total Days |                         |              |            | 60        |

#### 1.4.7 System Testing

| Serial                 | Work Description       | Start (date) | End (date) | Total day |
|------------------------|------------------------|--------------|------------|-----------|
| 1                      | Separate Module Test   | 10/09/2018   | 12/09/2018 | 2         |
| 2                      | Boundary Value Testing | 13/09/2018   | 16/09/2018 | 3         |
| 3                      | Functionality Test     | 17/09/2018   | 19/09/2018 | 2         |
| Total Days             |                        |              |            | 07        |
| Total Working Days 137 |                        |              |            |           |

# **CHAPTER 2**

## **SOFTWARE REQUIREMENT SPECIFICATION**

### **2.1 SRS**

Software Requirement Specification or SRS is a technique to find out the requirements and functionalities of a system. Also known as stakeholder's requirement specification. There're basically two types of requirements. They are – functional requirement and non-functional requirement.

This system has three types of users –

- i) Owners
- ii) Customers and
- iii) Admin

### **2.2 Studying similar projects**

Most of the requirements of this project were collected by studying similar websites or projects. I observed their system, tried to find out their lackings and tried to make this project advance in some particular way.

Name of some websites I studied:

- i) <https://gari.com.bd/>
- ii) <https://trucklagbe.com/>
- iii) <https://khaledrentacar.com/>

### **2.3 Expert's opinion**

I talked with many developers about my project. I tried to understand what they say about my project or their suggestions. This helped me a lot to evaluate my work on this project.

### **2.4 Brainstorming**

Besides studying similar projects and having expert's opinion, I had to do a lot of brainstorming. I jot down everything in my notebook and the evaluated very carefully. Then I sorted out the best possible outcomes and compared that with real life issues.

## 2.5 Software Requirement Specification

| Requirement No. | Requirement Name | Requirement Description                                                                                            | Requirement Type           |
|-----------------|------------------|--------------------------------------------------------------------------------------------------------------------|----------------------------|
| 1               | Authentication   | Customers and owners must be authenticated.                                                                        | Functional requirement     |
| 2               | Booking          | Customers have to book their preferred vehicle.                                                                    | Functional requirement     |
| 3               | Manage vehicle   | Vehicle owner can create, delete and update vehicle's information. Admin will be able to monitor the whole system. | Functional requirement     |
| 4               | Authorization    | Admin will have the authorization power.                                                                           | Functional requirement     |
| 5               | Usability        | Easy to use.                                                                                                       | Non-functional requirement |
| 6               | Capacity         | MySQL default capacity                                                                                             | Non-functional requirement |
| 7               | Performance      | Will have the ability to maintain mass amount of customer.                                                         | Non-functional requirement |
| 8               | Quality          | Maintain a user friendly environment. Easy to see and use navigation.                                              | Non-functional requirement |
| 9               | Reliability      | The system would be reliable for the owners and customers.                                                         | Non-functional requirement |
| 10              | User interface   | The UI would be visually appealing.                                                                                | Non-functional requirement |
| 11              | User experience  | User experience should be satisfactory                                                                             | Non-functional requirement |
| 12              | Security         | Prevention of any potential threat. Such as – MySQL injections through the search boxes and forms.                 | Non-functional requirement |

## 2.6 Used Technology

Technologies that are used to build this system is given below -

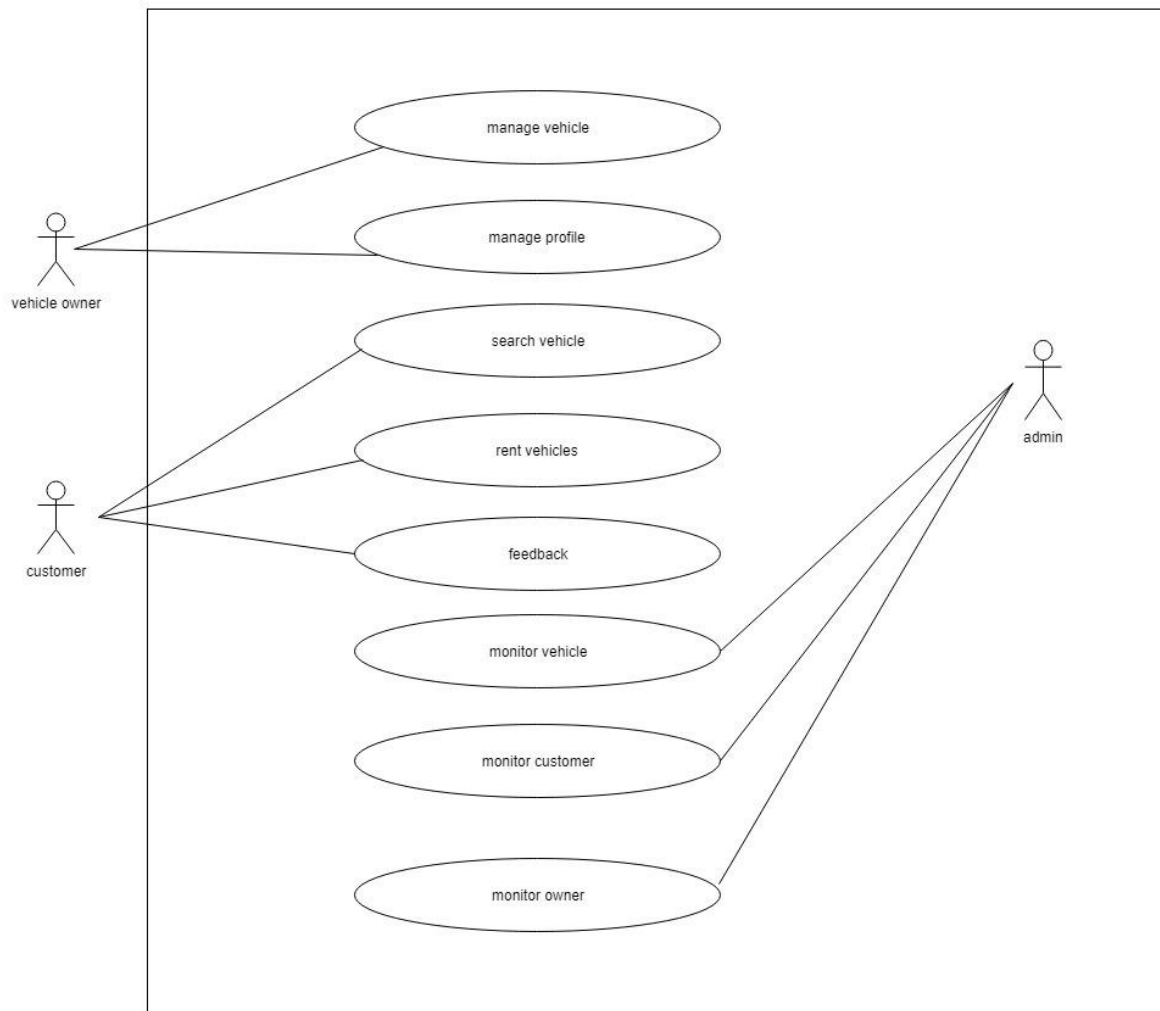
- Programming language: PHP 7.2
- Programming language Framework: LARAVEL 5.7

- Web Design: HTML5, CSS3, Bootstrap 3
- Client side scripting language: JavaScript
- Database : MySQL
- Web server: Apache.

## CHAPTER 3

### SYSTEM ANALYSIS

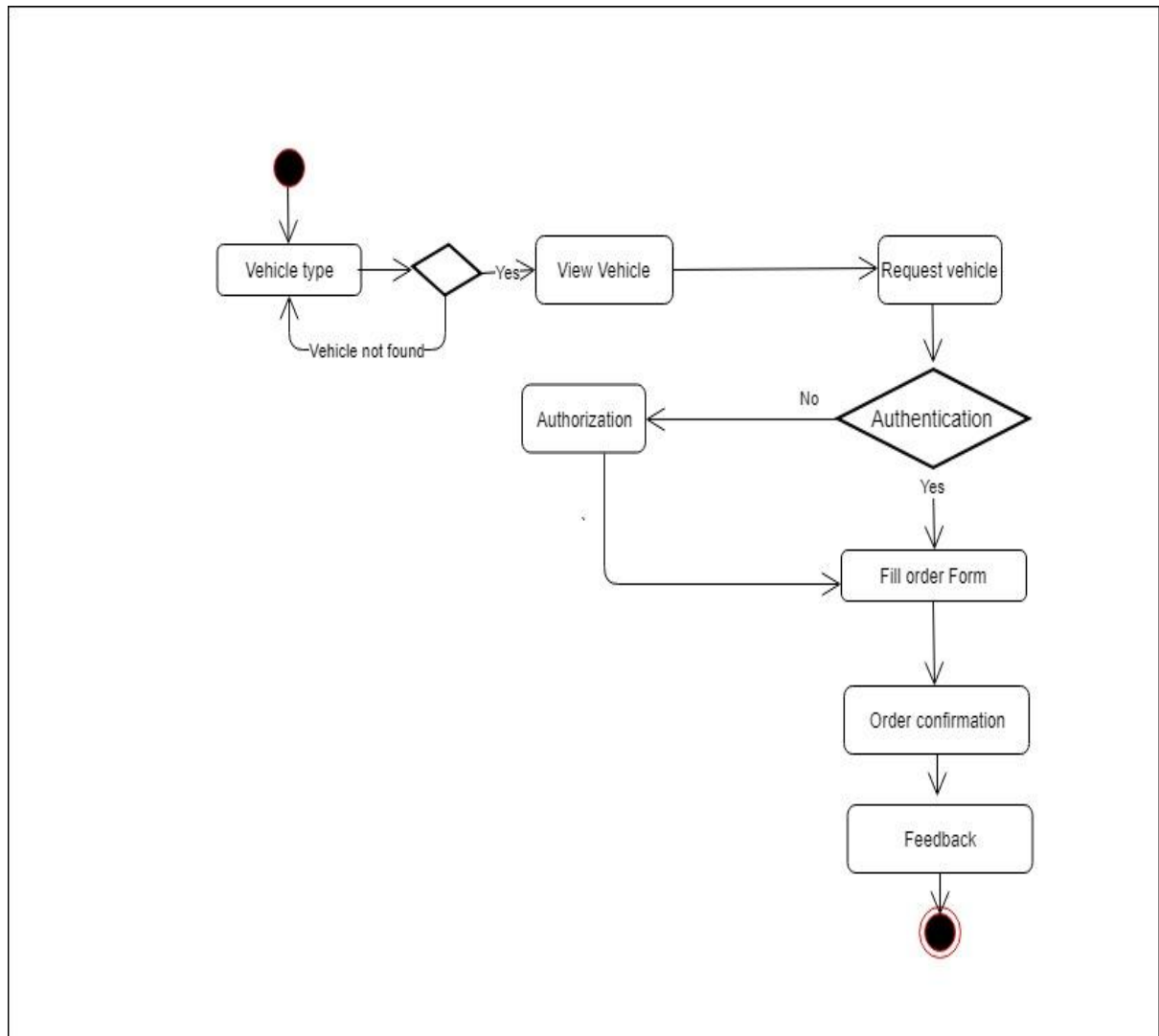
#### 3.1 Use Case Diagram



**Figure 3.1** Use case diagram

Actors of this system are vehicle owners, customers and admin. Actions and functions of this use case are – Manage vehicle, manage profile, vehicle type, rent vehicle, feedback, monitor vehicle, monitor customer. Those lines above in the use case indicates the relationship between actors and actions.

### 3.2 Activity Diagram



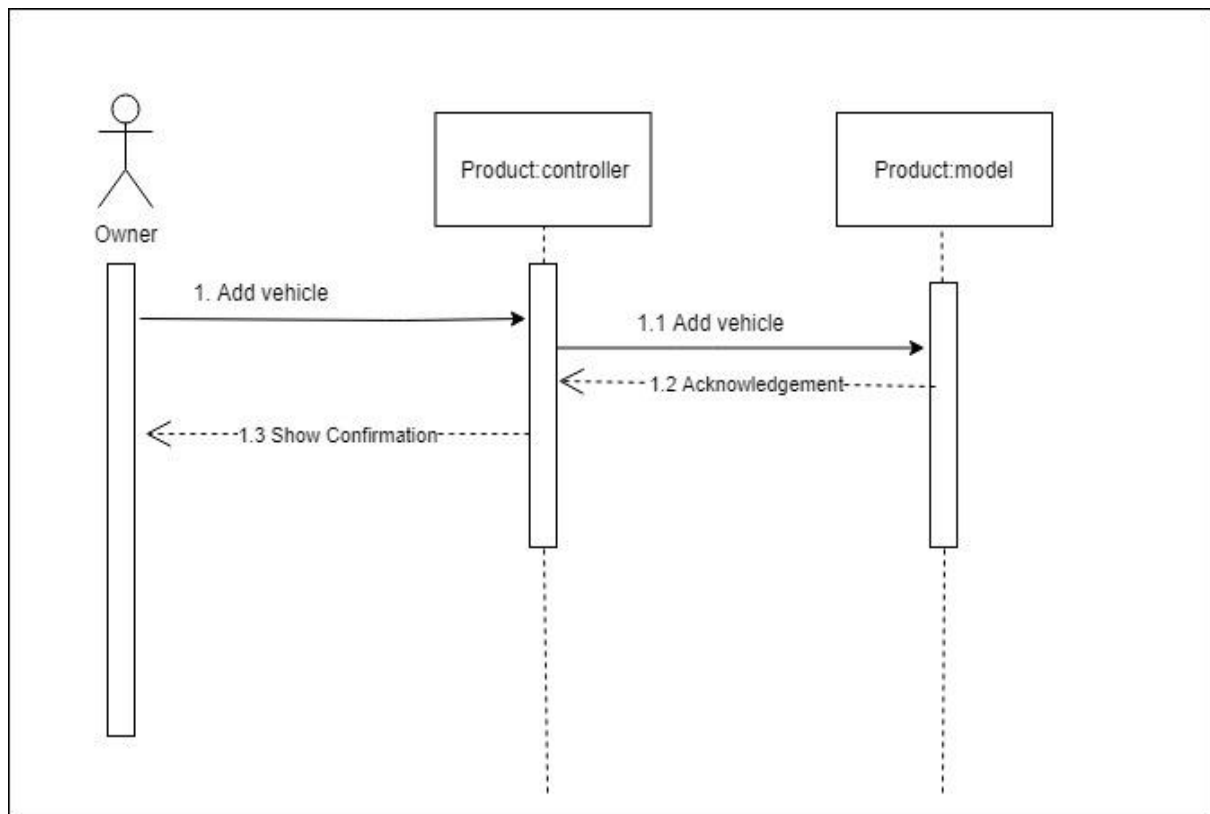
**Figure 3.2** Activity diagram

Activity diagram is to understand the whole system sequentially. From the activity diagram we can understand the sequential activities of the system.

## CHAPTER 4

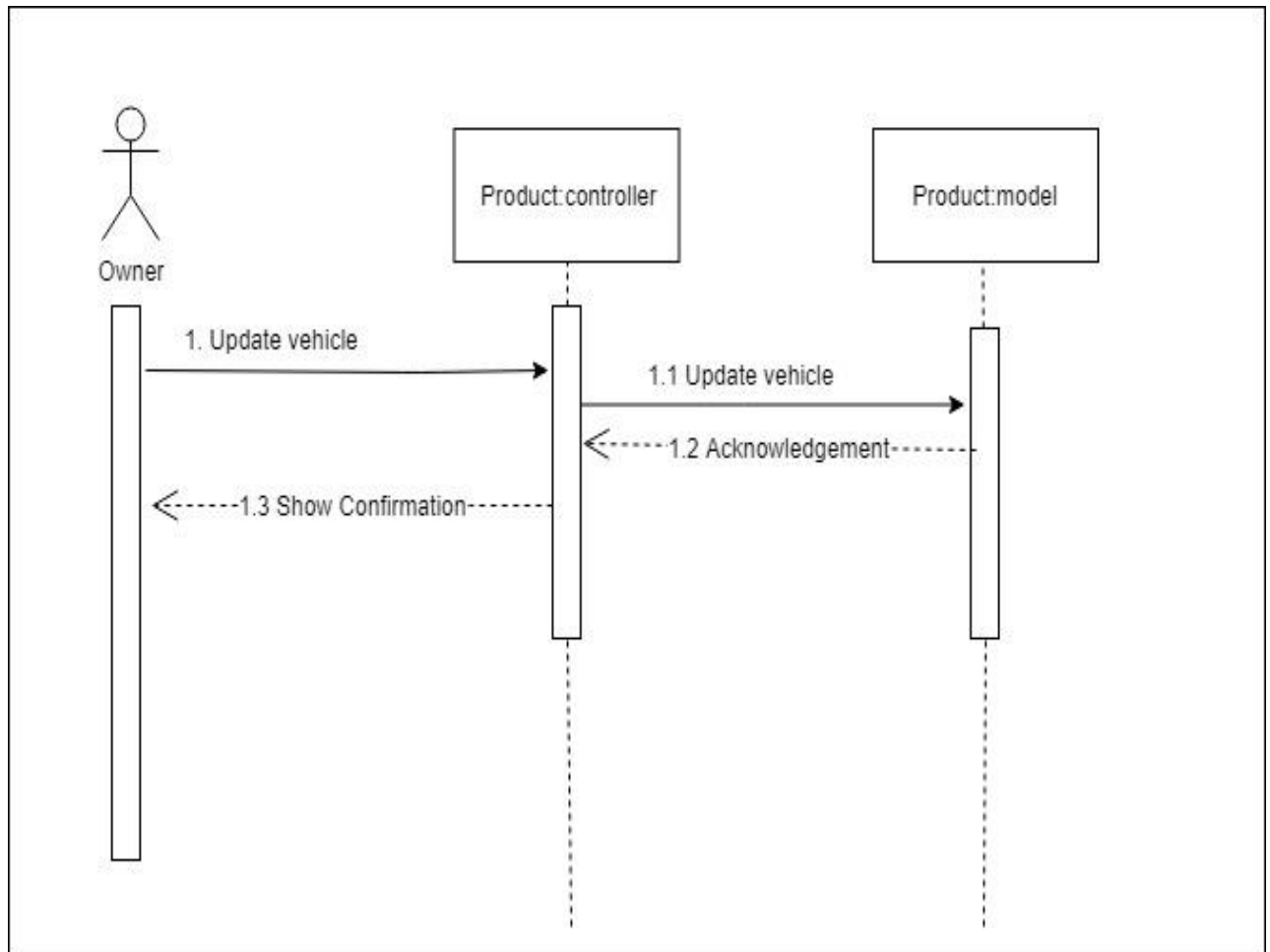
### SYSTEM DESIGN SPECIFICATION

#### 4.1 Sequence Diagram



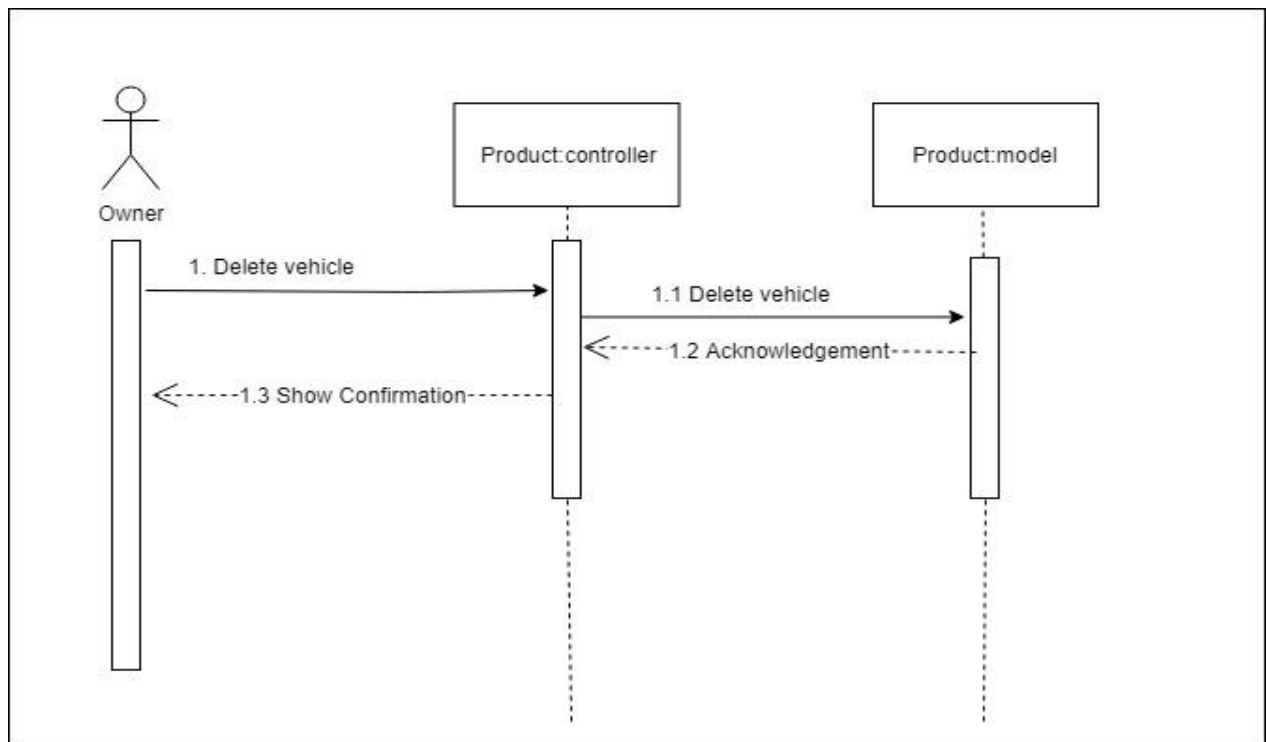
**Figure 4.1.1** Add vehicle

This diagram is showing that vehicle owner can vehicle through Product controller and Product model.



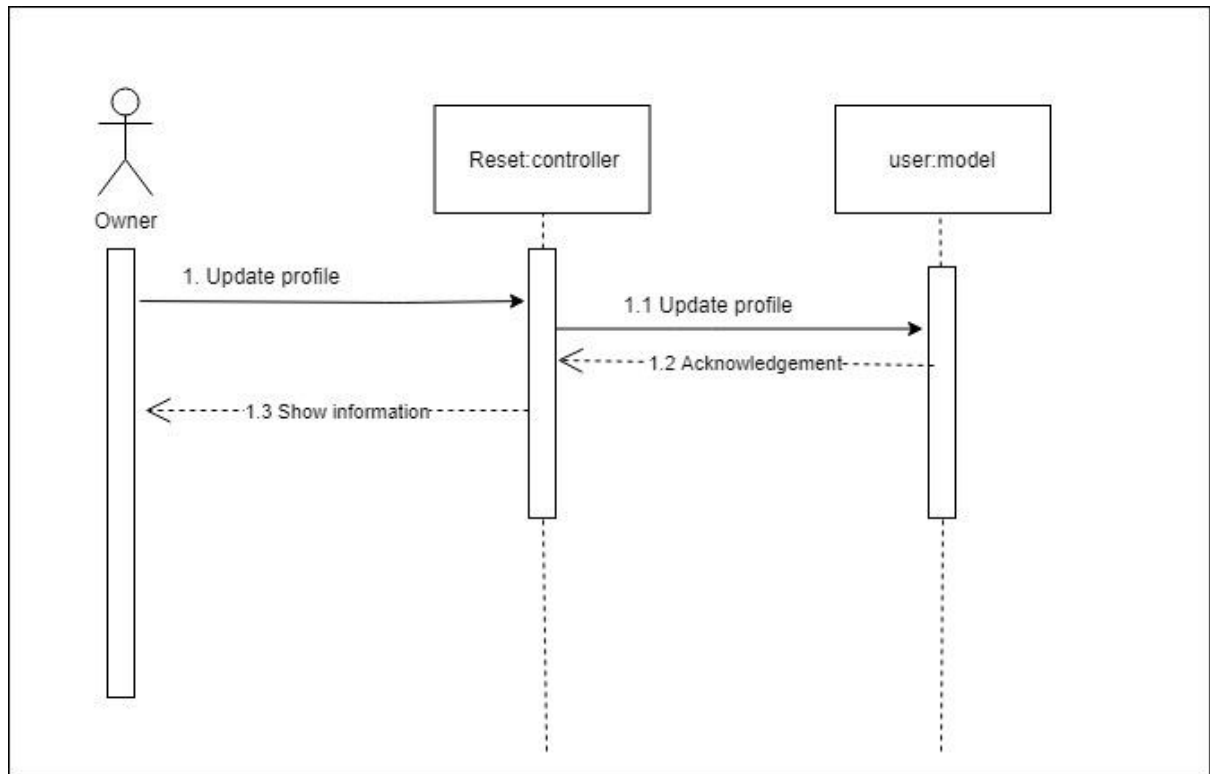
**Figure 4.1.2** Update vehicle

Vehicle owner can update vehicle information via product controller and product model.



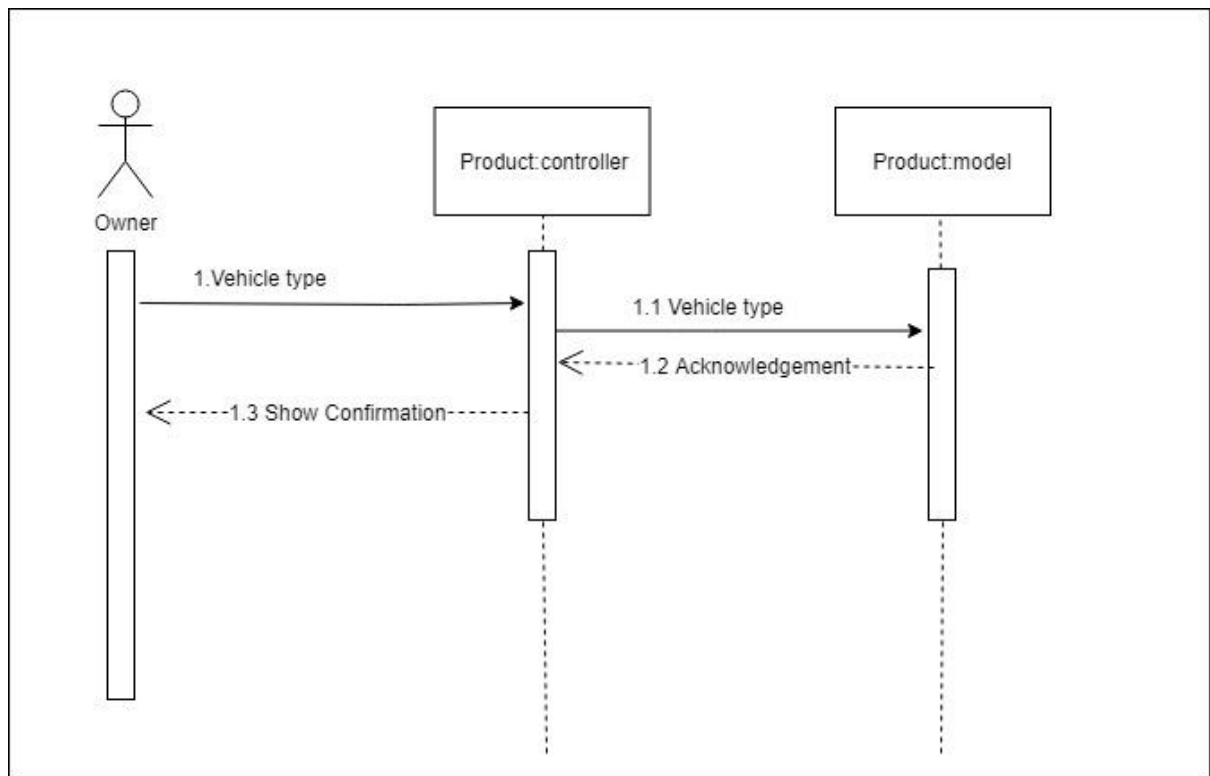
**Figure 4.1.3** Delete vehicle

Vehicle owner can delete vehicle via product controller and product model.



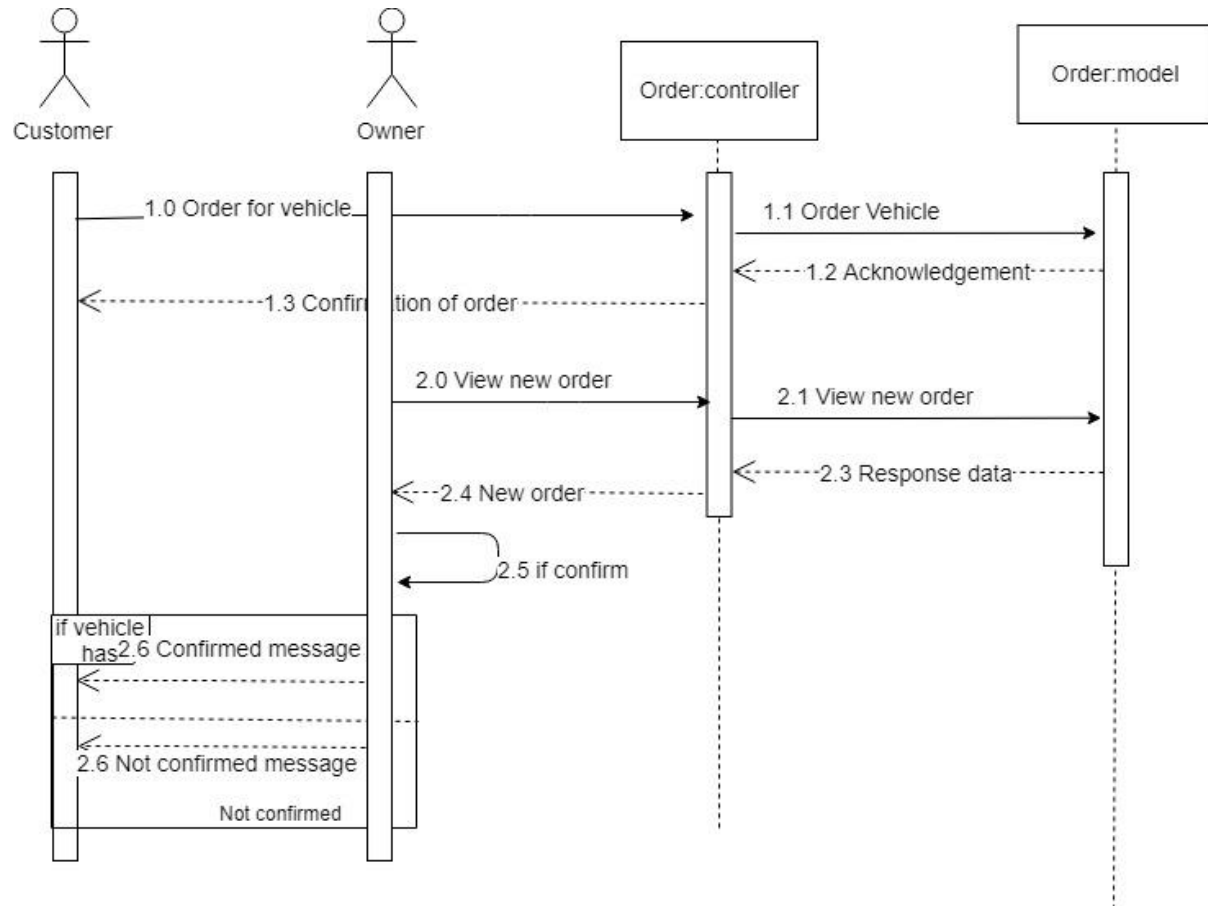
**Figure 4.1.4** Manage Profile

Vehicle owners can reset their password through reset controller and user model.



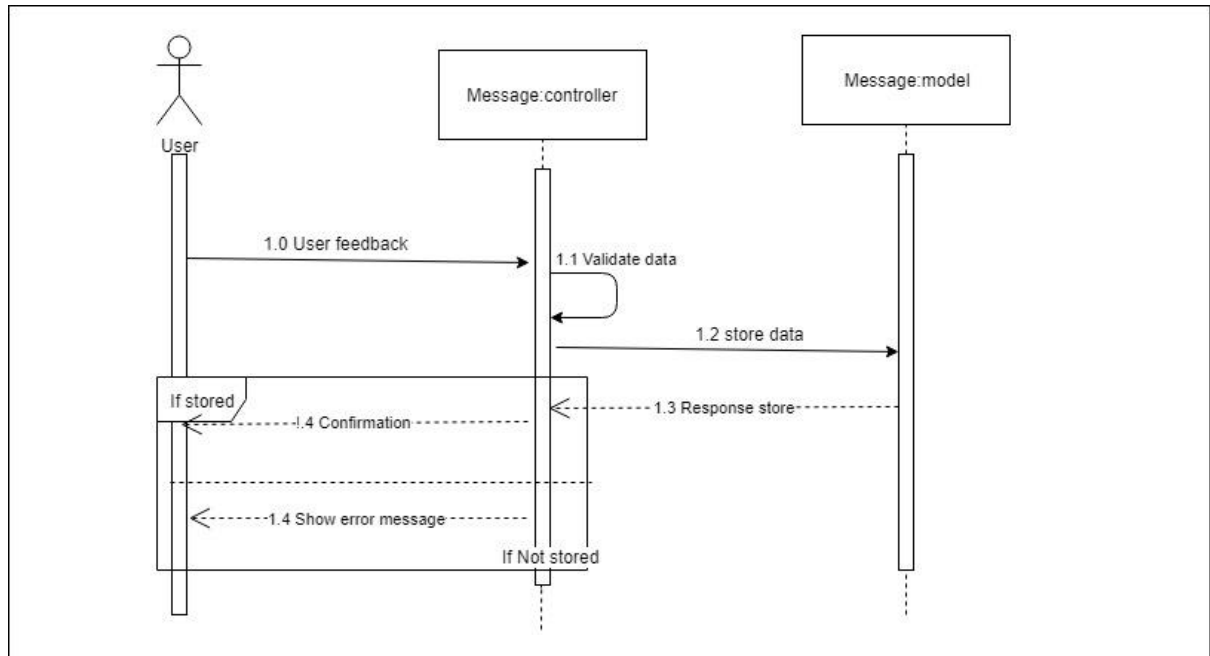
**Figure 4.1.5** vehicle type

Owners will have to include vehicles to a particular type. Such as large, medium or small.



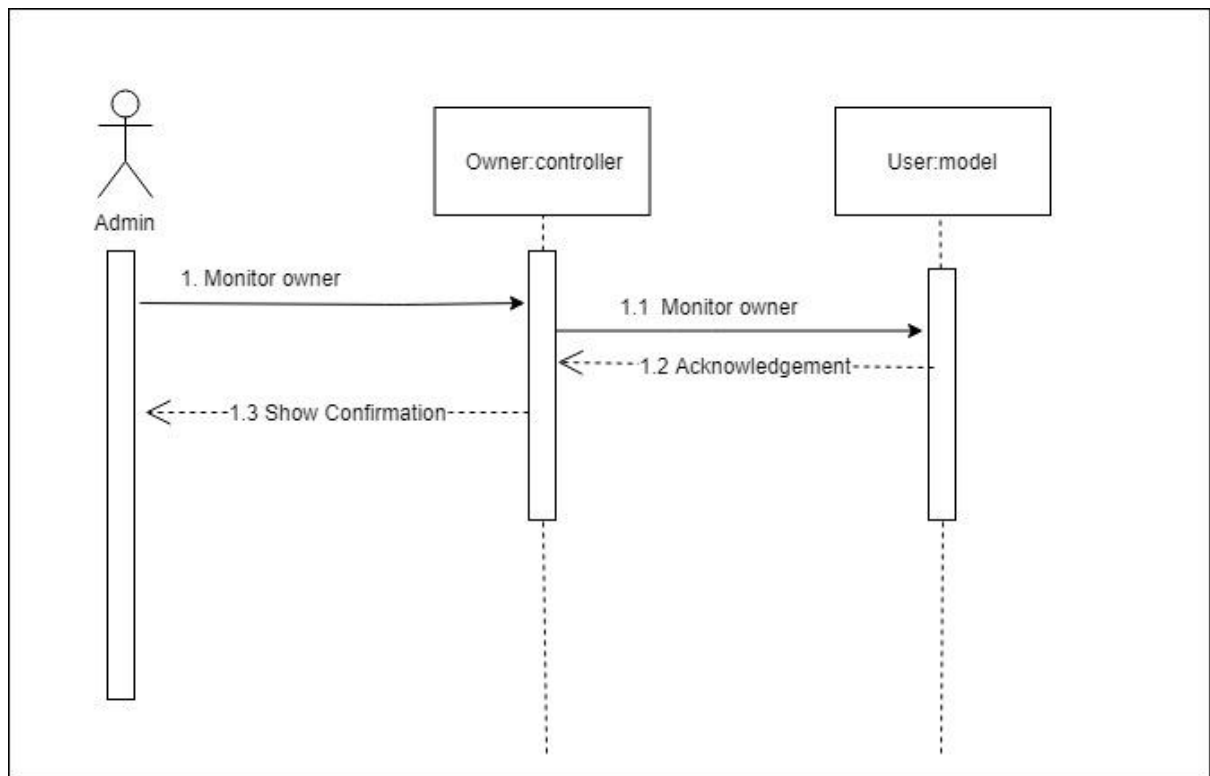
**Figure 4.1.6** Rent vehicle

Customer can rent vehicles through order controller and order model.



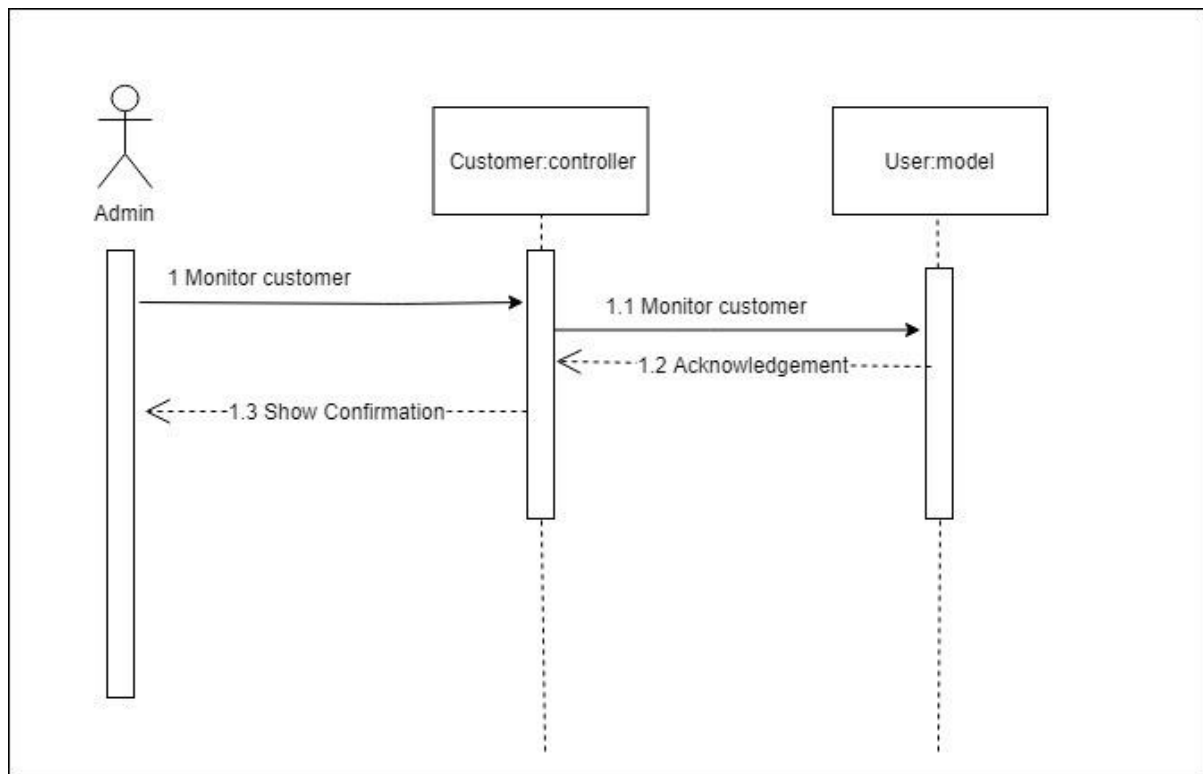
**Figure 4.1.7** Customer feedback

Customer can give feedback through message controller and message model.



**Figure 4.1.8** Monitor owner

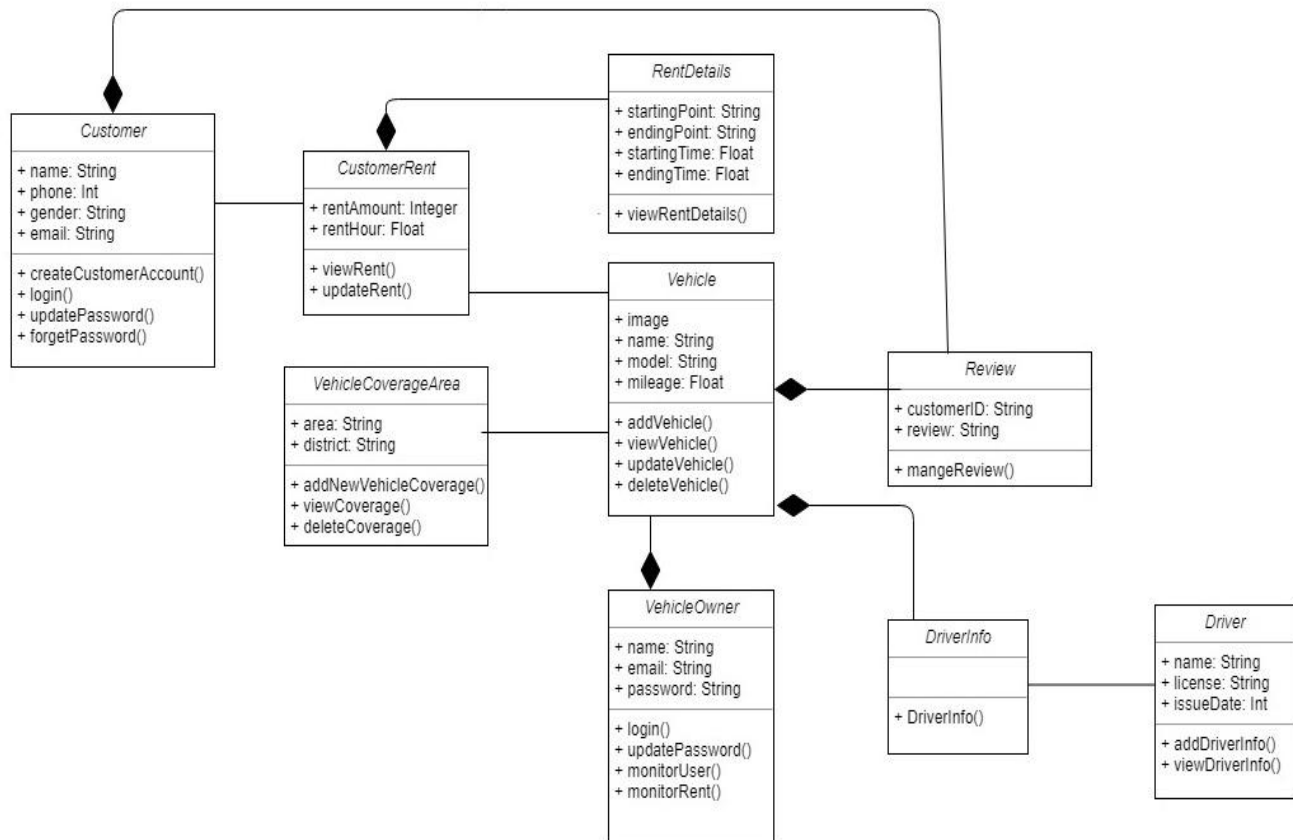
Admin can monitor owner via owner controller and user model.



**Figure 4.1.10** Monitor Customer

Admin can monitor customer via customer controller and user model.

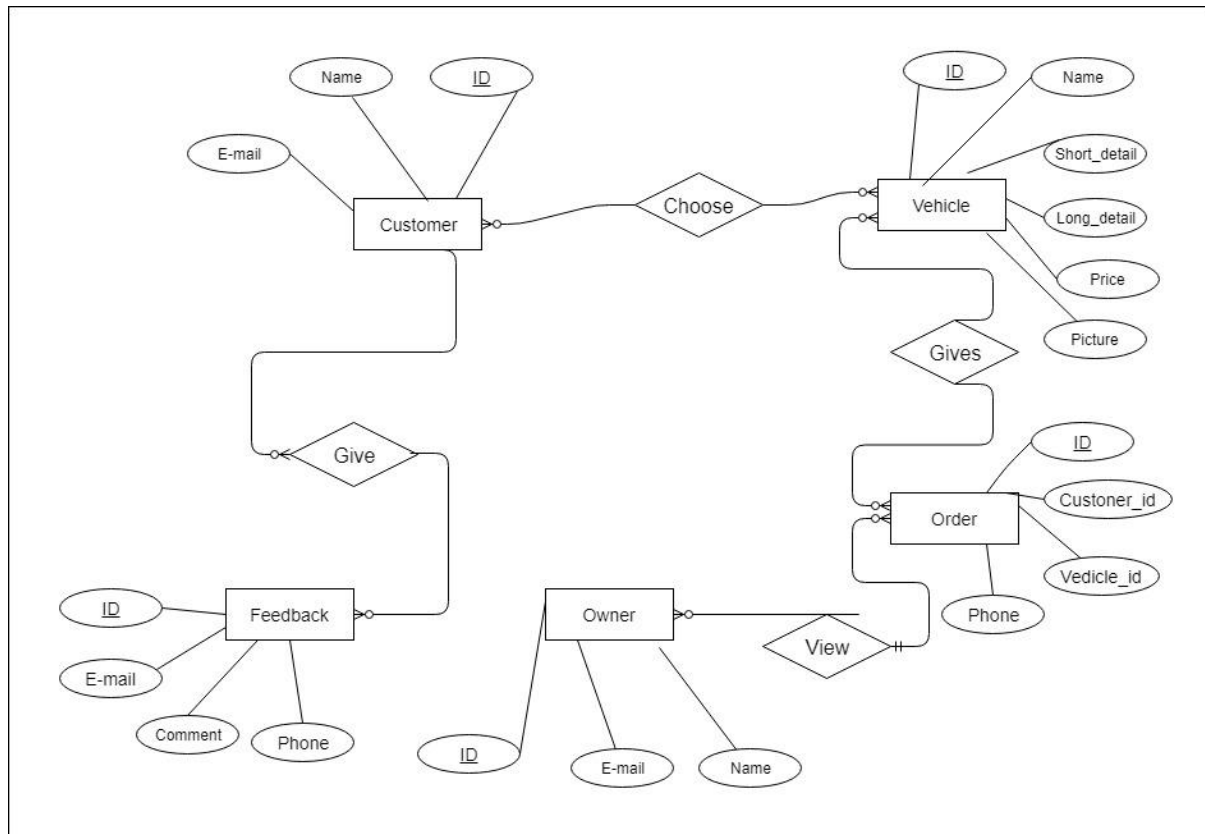
## 4.2 Class Diagram



**Figure 4.2** Class Diagram

Class diagram contains all the classes of this system. Class diagram shows the relationships between classes.

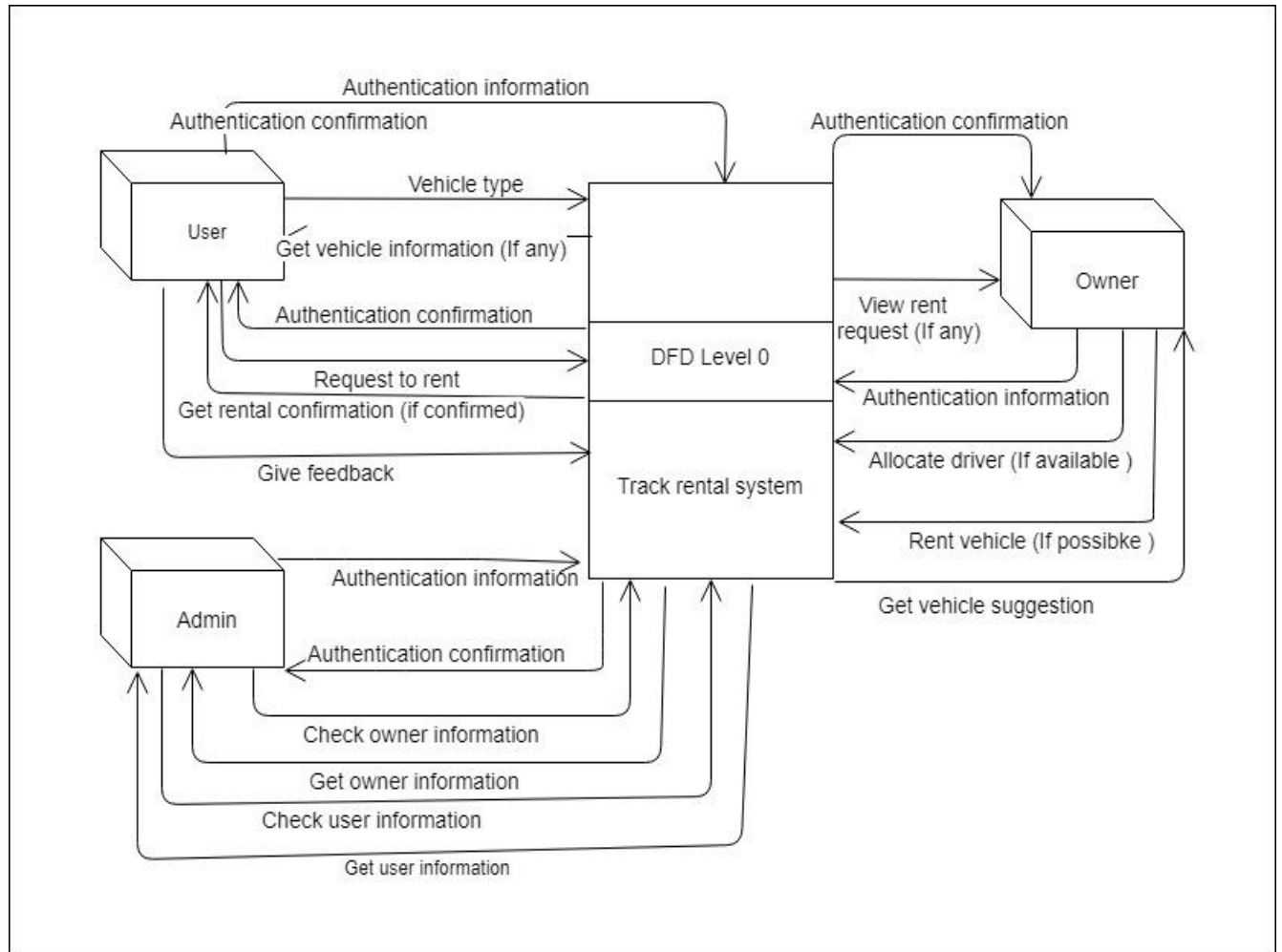
### 4.3 Database Design Diagram (ER Diagram)



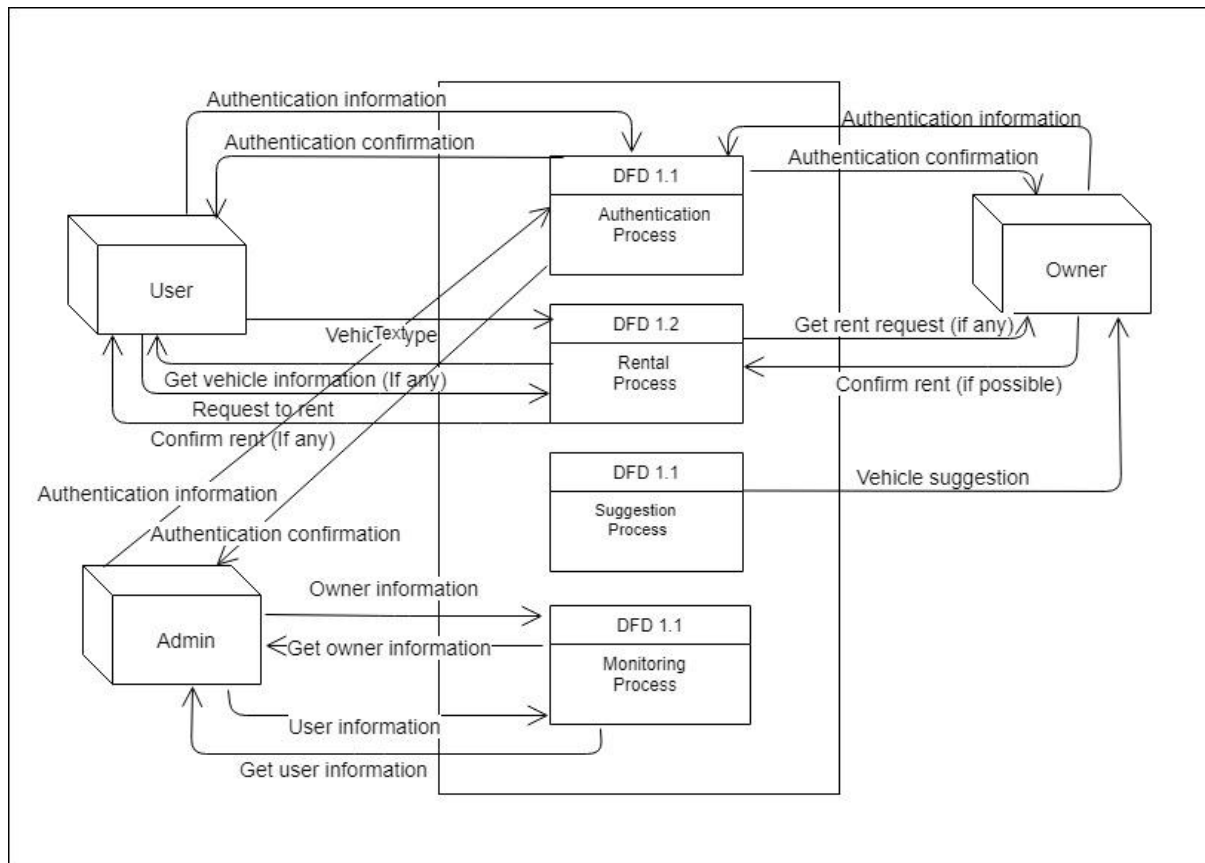
**Figure 4.3** ER Diagram

Entity relationship diagram shows the relationships of entity sets stored in a database. Entities have attributes that define its properties.

#### 4.4 Data Flow Diagram



**Figure 4.4.1** DFD level 0



**Figure 4.4.2 DFD level 1**

Data flow diagrams show the traditional visual representations of the information that streaming within the system. An ideal DFD depicts the whole requirements for the system visually and graphically.

# CHAPTER 5

## SYSTEM TESTING

### 5.1 Unit Testing

Software testing is used to evaluate the functionality of the system. Unit testing is a level of software testing where individual units or components of a software are tested. The purpose of unit testing is to validate that each unit of software performs as it was designed to perform. Units are the tiniest testable parts of any software. Unit testing helps us finding any defects or bugs in the system that is being tested. In this case, I'm following PHP unit testing procedure. I'll go through every functionality of this system that needs to be tested.

#### Features that are tested in this case:

- Authentication
- Adding vehicle
- Ordering vehicle
- Customer feedback

| Test Name: Authentication |                              |                                         |        |
|---------------------------|------------------------------|-----------------------------------------|--------|
| Test Procedure            | PHP Unit Test                |                                         |        |
| Test Id                   | ES001                        |                                         |        |
| Step                      | Operator actions             | Expected result and evaluation criteria | Result |
| 01                        | Form validation check        | Minimum length six, required all field. | Ok     |
| 02                        | User existing (registration) | Duplicate mail check.                   | Ok     |
| 03                        | Valid credential             | Create user session.                    | Ok     |

**Table 5.1** Unit Test Case for “Authentication”

| <b>Test Name: Add vehicle</b> |                         |                                                |               |
|-------------------------------|-------------------------|------------------------------------------------|---------------|
| <b>Test Procedure</b>         | PHP Unit Test           |                                                |               |
| <b>Test Id</b>                | <b>ES002</b>            |                                                |               |
| <b>Step</b>                   | <b>Operator actions</b> | <b>Expected result and evaluation criteria</b> | <b>Result</b> |
| 01                            | Form validation check   | All input field is required                    | Ok            |
| 02                            | Image upload            | Extension correct and size less than 2mb.      | Ok            |

**Table 5.2** Unit Test Case for “Adding vehicles”

| <b>Test Name: Order</b> |                         |                                                |               |
|-------------------------|-------------------------|------------------------------------------------|---------------|
| <b>Test Procedure</b>   | PHP Unit Test           |                                                |               |
| <b>Test Id</b>          | <b>ES003</b>            |                                                |               |
| <b>Step</b>             | <b>Operator actions</b> | <b>Expected result and evaluation criteria</b> | <b>Result</b> |
| 01                      | Form validation check   | All input field is required                    | Ok            |
| 02                      | Phone number            | Eleven length and correct format               | Ok            |

**Table 5.3** Unit Test Case for “Ordering vehicles”

| <b>Test Name: Feedback</b> |                         |                                                |               |
|----------------------------|-------------------------|------------------------------------------------|---------------|
| <b>Test Procedure</b>      | PHP Unit Test           |                                                |               |
| <b>Test Id</b>             | <b>ES004</b>            |                                                |               |
| <b>Step</b>                | <b>Operator actions</b> | <b>Expected result and evaluation criteria</b> | <b>Result</b> |
| 01                         | Form validation check   | All input field is required                    | Ok            |
| 02                         | Email                   | Valid email                                    | Ok            |

**Table 5.4** Unit Test Case for “Customer’s feedback”

## CHAPTAR 6

### USER MANUAL

#### Step 1:

Visiting the home page.

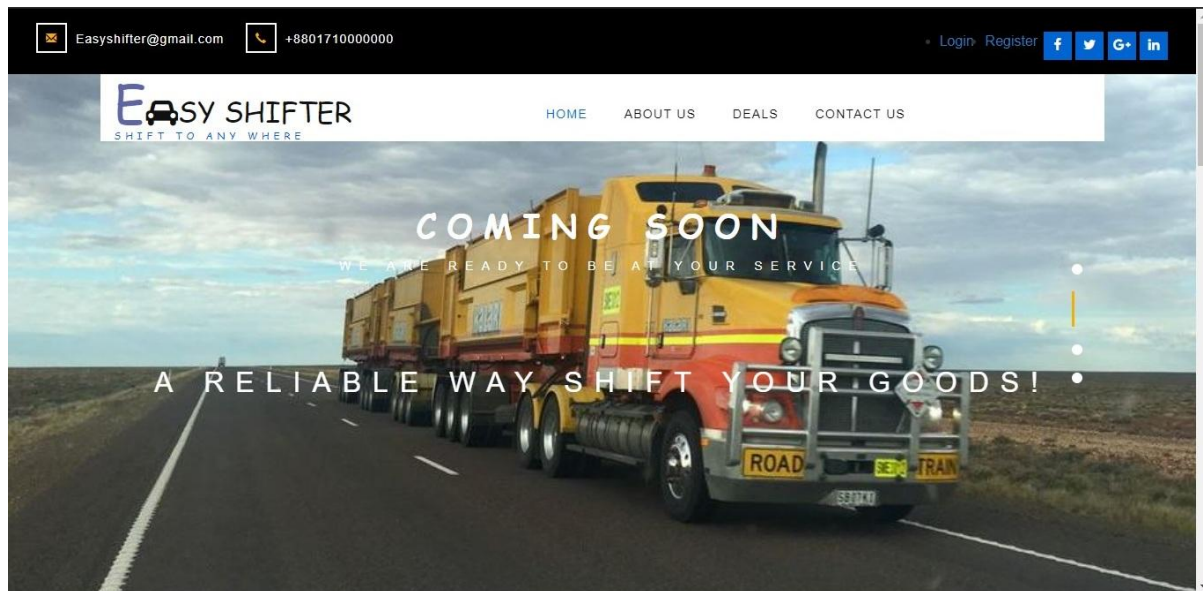


Figure 6.1 Home page

#### Step 2:

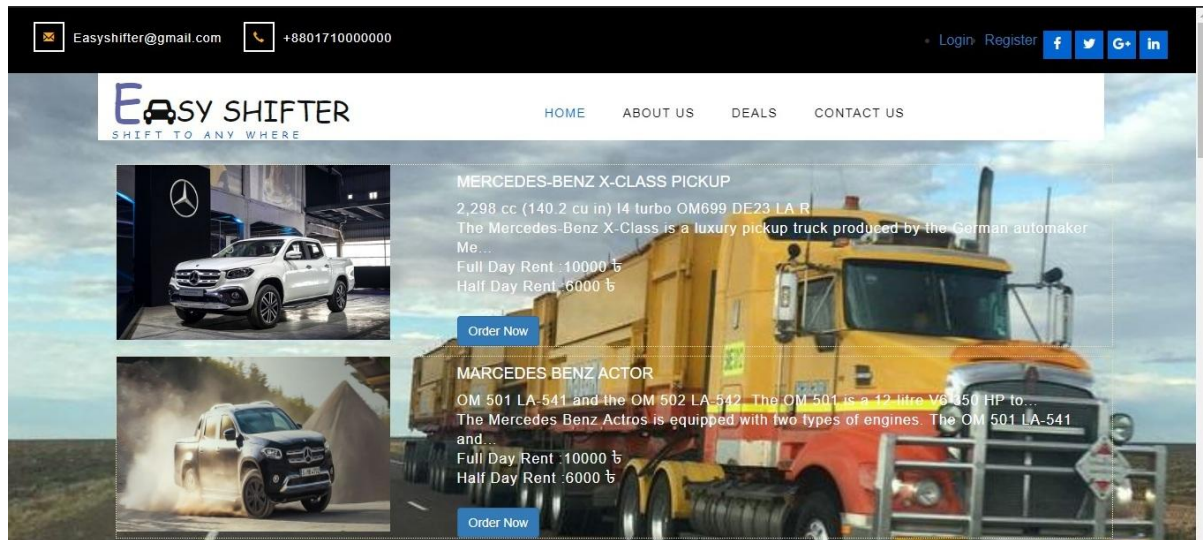
After clicking on “deals”, this will appear –



Figure 6.2 Deals

### **Step 3:**

Customers can click on any specific type they want and existing vehicles of that particular type will appear.



**Figure 6.3** Available Vehicles

### **Step 4:**

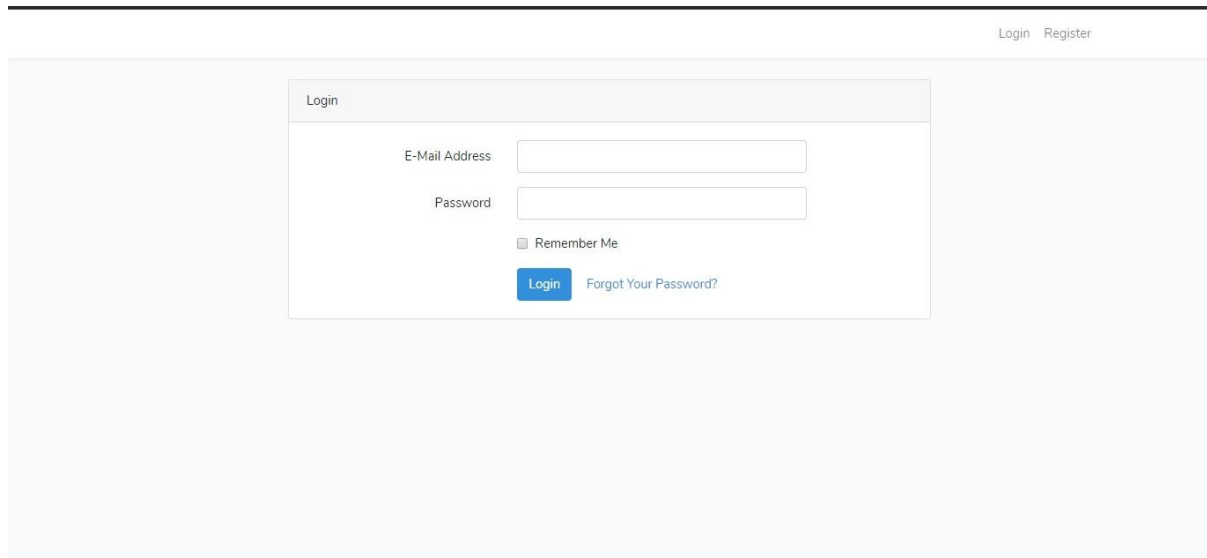
Customers and owners will have to register in the system.

The screenshot displays a 'Register' form. It includes input fields for 'Name', 'E-Mail Address', 'Password', and 'Confirm Password'. Below these fields, there are radio buttons for 'User Type', with 'Owner' selected and 'Customer' as an option. A blue 'Register' button is positioned at the bottom of the form.

**Figure 6.4** Registration

### **Step 5:**

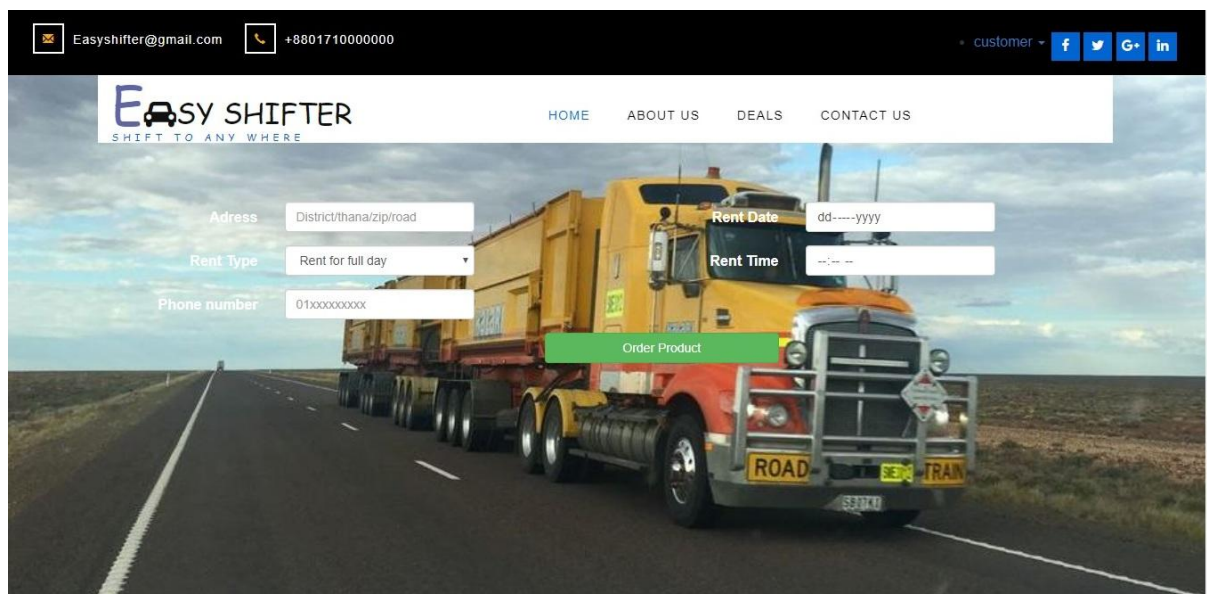
Customers and owners will have to login in the system to order or post for any vehicle.



**Figure 6.5 Login**

### **Step 6:**

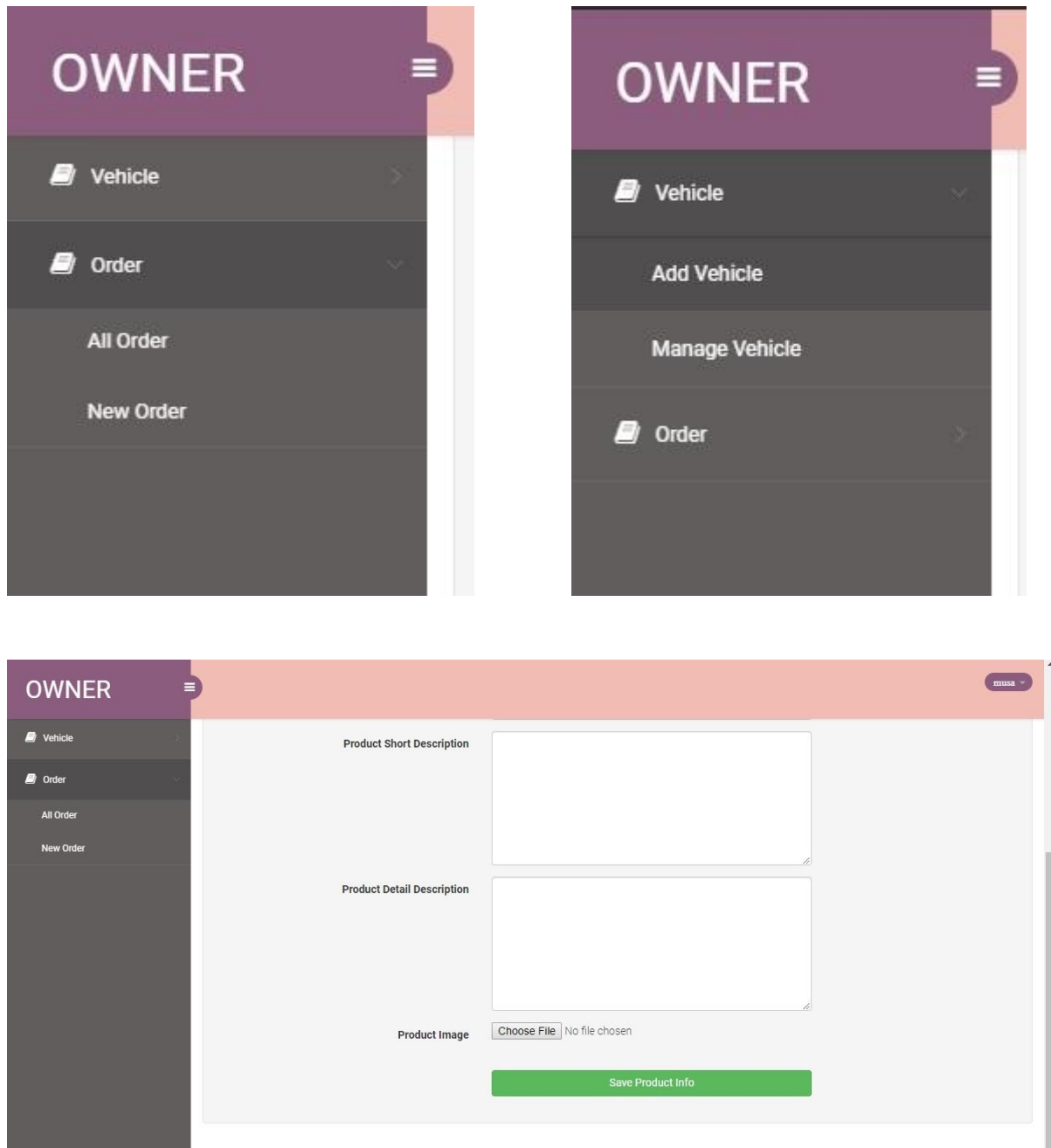
After login and choosing a particular vehicle, customers will have to fill this form up to order a vehicle.



**Figure 6.6 Ordering Vehicle**

### Step 7:

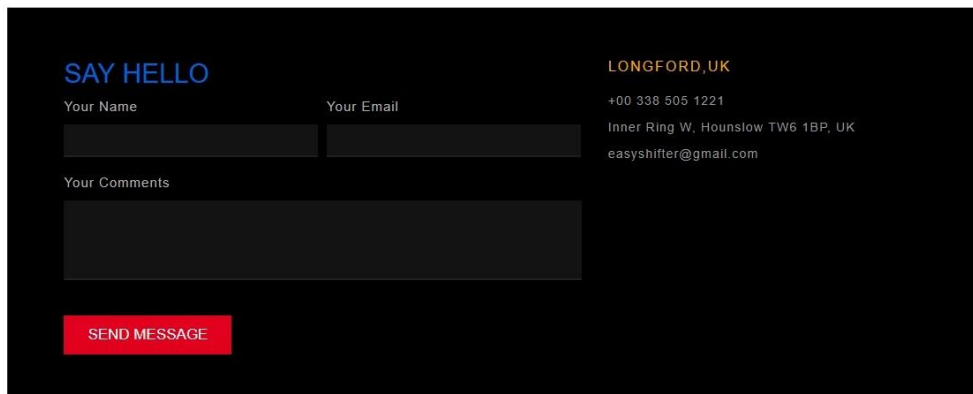
After a successful login and after clicking on the “vehicles” option on the navbar, owners can view their order, manage their vehicle or post for a new vehicle.



**Figure 6.7** Owner Panel

**Step 8:**

Customers and owners can give their opinion through the “feedback” option.



The image shows a feedback form titled "SAY HELLO" in blue text. The form is set against a dark background. It includes three input fields: "Your Name", "Your Email", and "Your Comments". The "Your Name" and "Your Email" fields are side-by-side, while "Your Comments" is a larger text area below them. A red button labeled "SEND MESSAGE" is positioned at the bottom left of the form. To the right of the form, contact information is displayed in orange and white text: "LONGFORD, UK", "+00 338 505 1221", "Inner Ring W, Hounslow TW6 1BP, UK", and "easyshifter@gmail.com".

IT'S TIME TO SHIFT EASILY

**Figure 6.8** Feedback

## CHAPTER 7

### CONCLUSION

My only focus during this project was to make a platform, where people could find their expected vehicle for shifting their goods and furniture at a reasonable price without less effort. This project was about to give people a better choice. I know there are lots of shortcomings of this project. But I'm about to make this a better platform in near future. There are lots of future planning and scopes I have figured out for this system.

#### 7.1 Future Scope

- I'm planning to develop a chat box for better experience.
- In the upcoming days, I want to build an android version of this system.
- In near future, I want to come out with an online payment system for this platform.

#### 7.2 Limitations

- I couldn't build a virtual chat box in this short time.
- There's no option for online payment.
- This is not a single page application.

#### 7.3 GitHub Link

<https://github.com/Wowmusa/Easy-Shifter>

## REFERENCES

1. <https://trucklagbe.com/>
2. <https://gari.com.bd/>
3. Thomas T. Barker, Writing Software Documentation, Second Edition , 1998
4. <https://laravel.com/docs/5.7>
5. Craig Larman, Applying UML and patterns, 2004

