

CarSoul: A modern car renting application connecting lessor with lessee on the same platform

Final Year Project

Session 2017-2021

A project submitted in partial fulfilment of the
COMSATS University Degree
of
BS in Computer Science / Software Engineering (CUI)



Department of Computer Science
COMSATS University Islamabad, Lahore Campus

28 December 2020

Project Detail

Type (Nature of project)	<input checked="" type="checkbox"/> Development <input type="checkbox"/> Research <input type="checkbox"/> R&D			
Area of specialization	Mobile Application			
Project Group Members				
Sr.#	Reg. #	Student Name	Email ID	*Signature
(i)	SP17-BSE-110	Syed Faizan Ali (Group Leader)	Syedzada411@gmail.com	<i>Faizan</i>
(ii)	SP17-BSE-074	Muhammad Hamad Awan	SP17-BSE-074@cuilahore.edu.pk	<i>Hamad</i>
(iii)	SP17-BSE-160	Ariba Ashraf	SP17-BSE-160@cuilahore.edu.pk	<i>Ariba</i>

*The candidates confirm that the work submitted is their own and appropriate credit has been given where reference has been made to work of others

Plagiarism Free Certificate

This is to certify that, I am **Syed Faizan Ali** S/D/o **Syed Imran Ali**, group leader of FYP under registration no **CIIT/SP17-BSE-110/LHR** at Computer Science Department, COMSATS Institute of Information Technology, Lahore. I declare that my FYP proposal is checked by my supervisor and the similarity index is **2%** that is less than 20%, an acceptable limit by HEC. Report is attached herewith as Appendix A.

Date: 28/12/2020 Name of Group Leader: Syed Faizan Ali Signature: *Faizan*

Name of Supervisor: **Madam Sana Rizwan**

Co-Supervisor (if any): **Madam Sobia Usman**

Designation: **Assistant Professor**

Designation: **Assistant Professor**

Signature: *Sana Rizwan*

Signature: *Sobia Usman*

HoD: **Dr. Zulfiqar Habib**

Signature: _____

Project Abstract

This project is aimed to propose a mobile application which will allow a person (lessor) to rent out their car through the application and other person (lessee) to hire that car for rent on the same application/platform. The idea is innovative and unique itself that allow lessee to check a list of cars available in a specific area matching his/her requirements and needs. He can compare cars based on rating provided by other users and prices and other factors as well.

After using the service, the user (lessee) can then provide his/her review and ratings for the car and the lessor which will rank the car and lessor on the application by a ranking algorithm on the application. The project is aimed to make lessor earn money by their vehicle and the lessee to find good quality, reliable and affordable cars with ease.

Considering the security situation and environment of our country. The application will allow lessor to track the location of his car in real time on a map using the GPS services from the lessee's mobile application. Once the car is allotted to someone. The lessor can view the current location of his car anytime from his app. This feature is what makes this application to stand out from other traditional models of car renting application. Moreover, the lessor can decide whether to post his car for rent or not by calculating the expected rent of his car/variant in that location. This feature will certainly help him to decide if he sees an attractive offer.

Table of Contents

CarSoul: A modern car renting application connecting lessor with lessee on the same platform	1
Final Year Project.....	1
Session 2017-2021	1
Project Detail	2
Plagiarism Free Certificate.....	2
List of Tables	16
1 CHAPTER:1 INTRODUCTION.....	19
1.1 Introduction.....	19
1.2 Lessor and Lessee can maintain their profile.....	19
1.3 Objectives	20
1.4 Problem Statement	20
1.4.1 Assumptions.....	20
1.4.2 Constraints	20
1.5 Project Scope.....	20
2 Chapter: 2 Requirements Analysis	21
2.1 Literature Review	21
2.2 Stakeholder's List	22
2.3 Requirement Elicitation	22
2.3.1 Functional Requirements	22
FR01: Car Search.....	22
FR02: Advance Filters	22
FR03: View Car Details for Lessee	23
FR04: Report Car Listing	23
FR05: Checkout.....	24
FR06: View Activity	24

FR07: Add to Favourite.....	24
FR08: Feedback.....	25
FR09: List your Car	25
FR10: Payment withdraw request.....	27
FR11: View Posted Cars.....	28
FR12: Track your Car	28
FR13: Deactivate Car Posting	29
FR14: View Trip History	29
FR15: View Active Trips.....	30
FR16: Messages	31
FR17: View Profile	31
FR18: Complete Profile.....	31
FR19: Edit Profile.....	32
FR20: View Account details	33
FR21: Sign Up.....	33
FR22: Login.....	34
FR23: Logout	34
FR24: Forgot Password	34
FR25: Cancel Order	35
2.3.2 Non-Functional Requirements:.....	35
2.3.3 Requirement Traceability Matrix.....	37
2.4 Use Case Descriptions:.....	39
2.4.1 Car Search.....	39
2.4.2 Advance Filter	40
2.4.3 View Car Details	41
2.4.4 Report Car Listing	42
2.4.5 Checkout:.....	43

2.4.6	View Activity:.....	44
2.4.7	Add to Favourite.....	45
2.4.8	Feedback.....	47
2.4.9	List Your Car	48
2.4.10	Payment Withdraw Request.....	50
2.4.11	View Posted Car	52
2.4.12	Track Your Car.....	53
2.4.13	Deactivate Car Posting	54
2.4.14	View Trip History	55
2.4.15	View Active Trips.....	56
2.4.16	Messages	57
2.4.17	View Profile	58
2.4.18	Complete Profile.....	59
2.4.19	Edit Profile.....	61
2.4.20	View Account Details	63
2.4.21	Signup	64
2.4.22	Login.....	65
2.4.23	Logout	66
2.4.24	Forgot Password	67
2.4.25	Cancel Order	69
2.5	Use Case Design's:	70
2.5.1	Use Case Diagram: Car Search	70
2.5.2	Use Case Diagram: Advance Filters	71
2.5.3	Use Case Diagram: Car Details for Lessee	72
2.5.4	Use Case Diagram: Report Car Listing	72
2.5.5	Use Case Diagram: Checkout.....	74
2.5.6	Use Case Diagram: View Activity	75

2.5.7	Use Case Diagram: Add to Favourite.....	75
2.5.8	Use Case Diagram: Feedback.....	76
2.5.9	Use Case Diagram: List Your Car	77
2.5.10	Use Case Diagram: Payment Withdraw Request.....	78
2.5.11	Use Case Diagram: View Posted Cars.....	79
2.5.12	Use Case Diagram: Track Your Car.....	80
2.5.13	Use Case Diagram: Deactivate Car Posting	80
2.5.14	Use Case Diagram: View Trip History.....	81
2.5.15	Use Case Diagram: View Active Trip.....	81
2.5.16	Use Case Diagram: Messages	82
2.5.17	Use Case Diagram: View Profile	83
2.5.18	Use Case Diagram: Complete Profile.....	84
2.5.19	Use Case Diagram: Edit Profile.....	85
2.5.20	Use Case Diagram: View Account Details	86
2.5.21	Use Case Diagram: Signup	86
2.5.22	Use Case Diagram: Login.....	87
2.5.23	Use Case Diagram: Logout.....	88
2.5.24	Use Case Diagram: Forgot Password	88
2.5.25	Use Case Diagram: Cancel Order	89
3	CHAPTER:3 System Design	90
3.1	Work Breakdown Structure:	90
3.2	Activity Diagrams:.....	91
3.2.1	Car Search.....	91
3.2.2	Advance Filters.....	92
3.2.3	View Car Details for Lessee.....	93
3.2.4	Report Car Listing	94
3.2.5	Checkout.....	95

3.2.6	View Activity	96
3.2.7	Add to Favourite.....	97
3.2.8	Feedback.....	98
3.2.9	List Your Car	99
3.2.10	Payment Withdraw Request.....	100
3.2.11	View Posted Cars.....	101
3.2.12	Track Your Car.....	102
3.2.13	Deactivate Car Posting	103
3.2.14	View Trip History	104
3.2.15	View Active Trips.....	105
3.2.16	Messages	106
3.2.17	View Profile	107
3.2.18	Complete Profile.....	108
3.2.19	Edit Profile.....	109
3.2.20	View Account Details	110
3.2.21	Signup	111
3.2.22	Login.....	112
3.2.23	Logout	113
3.2.24	Forgot Password	114
3.2.25	Cancel Order	115
3.3	Sequence Diagrams:.....	116
3.3.1	Car Search:	116
3.3.2	Advance Filter:	116
3.3.3	View Car Details:.....	117
3.3.4	Report Car Listing:	118
3.3.5	Checkout:	119
3.3.6	View Activity:	120

3.3.7	Add to Favourite:	121
3.3.8	Feedback:.....	122
3.3.9	List Your Car:	123
3.3.10	Payment Withdraw Request:.....	124
3.3.11	View Posted Car:.....	126
3.3.12	Track Your Car:.....	126
3.3.13	Deactivate Car Posting:.....	127
3.3.14	View Trip History:.....	127
3.3.15	View Active Trips:.....	128
3.3.16	Messages:	128
3.3.17	View Profile:	129
3.3.18	Complete Profile:.....	129
3.3.19	Edit Profile:	130
3.3.20	View Account Details:.....	130
3.3.21	Sign Up:.....	131
3.3.22	Login:.....	131
3.3.23	Logout:	132
3.3.24	Forgot Password:	132
3.3.25	Cancel Order:	133
3.4	Software Architecture:.....	134
3.5	Class Diagram:.....	135
3.6	Database Diagram:.....	136
3.7	Network Diagram:	136
3.8	Collaboration Diagrams:	137
3.8.1	Car Search:	137
3.8.2	Advance Filter:	137
3.8.3	View Car Details:.....	137

3.8.4	Report Car Listing:	138
3.8.5	Checkout:	138
3.8.6	View Activity:	139
3.8.7	Add to Favourite:	140
3.8.8	Feedback:	140
3.8.9	List Your Car:	141
3.8.10	Payment Withdraw Request:	141
3.8.11	View Posted Cars:	142
3.8.12	Track Your Car:	142
3.8.13	Deactivate Car Posting:	143
3.8.14	View Trip History:	143
3.8.15	View Active Trips:	144
3.8.16	Messages:	145
3.8.17	View Profile:	146
3.8.18	Complete Profile:	146
3.8.19	Edit Profile:	147
3.8.20	View Account Details:	147
3.8.21	Signup:	148
3.8.22	Login:	149
3.8.23	Logout:	149
3.8.24	Forgot Password:	150
3.8.25	Cancel Order:	150
4	CHAPTER:4 System Testing	151
4.1.1	Car Search:	151
4.1.2	Advance Filters:	152
4.1.3	View Car Details	153
4.1.4	Report Car Listing	154

4.1.5	Checkout.....	155
4.1.6	View Activity	156
4.1.7	Add to Favourite.....	157
4.1.8	Feedback.....	158
4.1.9	List Your Car	159
4.1.10	Payment Withdraw Request.....	162
4.1.11	View Posted Cars.....	164
4.1.12	Track Your Car.....	165
4.1.13	Deactivate Car Posting	166
4.1.14	View Trip History.....	167
4.1.15	View Active Trips.....	167
4.1.16	Messages	168
4.1.17	View Profile	169
4.1.18	Complete Profile.....	170
4.1.19	Edit Profile.....	172
4.1.20	View Account Details	174
4.1.21	Signup	175
4.1.22	Login.....	176
4.1.23	Logout	177
4.1.24	Forgot Password	178
4.1.25	Cancel Order	180
4.2.1	Unit Testing:	181
4.2.2	Integration Testing:.....	181
4.2.3	System Testing:	182
4.2.4	Acceptance Testing:.....	182
5	Conclusion.....	182
5.1.1	Problem faced and Lessons learned:	182

5.1.2	Project Summary:	182
5.1.1	Future Work:	182
6	Prototype	183
6.1	Car Search:.....	183
6.2	Advance Filter:	183
6.3	View Car Details for Lessee:	184
6.4	Report Car Listing:.....	184
6.5	Add to Favorite:.....	185
6.6	Feedback:.....	185
6.7	List Your Car:.....	186
6.8	Payment Withdraw Request:.....	187
6.9	View Posted Car:	188
6.10	Track Your Car:.....	188
6.11	View Active Trips:	189
6.12	Messages:	190
6.13	View Profile:	190
6.14	Complete Profile:	191
6.15	Edit Profile:	191
6.16	View Account Details:	192
6.17	Signup:	192
6.18	Login:.....	193
6.19	Forget Password:.....	193
6.20	Cancel Order:	194
7	References	194
8	Turnitin report.....	195

List of Figures

Figure 1: Use case diagram for Car search	70
Figure 2: Use case diagram for Advance filters	71
Figure 3: Use case diagram for Car details for Lessee.....	72
Figure 4: Use case diagram for Report car listing.....	73
Figure 5: Use case diagram for Checkout	74
Figure 6: Use case diagram for View Activity.....	75
Figure 7: Use case diagram forAdd to favorite	76
Figure 8: Use case diagram for Feedback	76
Figure 9: Use case diagram for List your car	78
Figure 10: Use case diagram for Payment withdraw request.....	78
Figure 11: Use case diagram for View Posted cars.....	79
Figure 12: Use case diagram for Track your car	80
Figure 13: Use case diagram for Deactivate car posting.....	80
Figure 14: Use case diagram for View Trip History	81
Figure 15: Use case diagram for View Active Trip	82
Figure 16: Use case diagram for Messages	82
Figure 17: Use case diagram for View Profile	83
Figure 18: Use case diagram for Complete Profile	84
Figure 19: Use case diagram for Edit Profile	85
Figure 20: Use case diagram for View Account Details	86
Figure 21: Use case diagram for Signup	87
Figure 22: Use case diagram for Login	87
Figure 23: Use case diagram for Logout	88
Figure 24: Use case diagram forForgot Password	89
Figure 25: Use case diagram for Cancel Order	89
Figure 26: Work Breakdown Structure	90
Figure 27: Activity diagram for Car Search.....	91
Figure 28: Activity diagram for Advance Filter.....	92
Figure 29: Activity diagram for View Car details.....	93
Figure 30: Activity diagram for Report Car Listing	94
Figure 31: Activity diagram for Checkout	96
Figure 32: Activity diagram for View Activity.....	96
Figure 33: Activity diagram for Add to Favorite	97
Figure 34: Activity diagram for Feedback	98
Figure 35: Activity diagram for List Your Car	100
Figure 36: Activity diagram forPayment Withdraw Request.....	101
Figure 37: Activity diagram for View Posted Cars	101
Figure 38: Activity diagram for Track Your Car	102
Figure 39: Activity diagram for Deactivate Car Posting.....	104
Figure 40: Activity diagram for View Trip History	104
Figure 41: Activity diagram for View Active Trips.....	105
Figure 42: Activity diagram for Messages	106

Figure 43: Activity diagram for View Profile	107
Figure 44: Activity diagram for Complete Profile	109
Figure 45: Activity diagram for Edit Profile	110
Figure 46: Activity diagram forView Account Details	110
Figure 47: Activity diagram for Signup	112
Figure 48: Activity diagram for Login	112
Figure 49: Activity diagram for Logout	113
Figure 50: Activity diagram for Forgot password.....	115
Figure 51: Activity diagram for Cancel Order	115
Figure 52: Sequence diagram for Car Search.....	116
Figure 53: Sequence diagram forAdvance Filters	117
Figure 54: Sequence diagram for View Car Details	118
Figure 55: Sequence diagram for Report Car Listing	119
Figure 56: Sequence diagram for Checkout	120
Figure 57: Sequence diagram for View Activity	121
Figure 58: Sequence diagram for Add to Favorite	122
Figure 59: Sequence diagram for Feedback	123
Figure 60: Sequence diagram for List Your Car	124
Figure 61: Sequence diagram for Payment withdraw request.....	125
Figure 62: Sequence diagram for View Posted Cars.....	126
Figure 63: Sequence diagram for Track Your Car	127
Figure 64: Sequence diagram for Deactivate car posting	127
Figure 65: Sequence diagram for View Trip History.....	128
Figure 66: Sequence diagram for View Active Trips	128
Figure 67: Sequence diagram for Messages	128
Figure 68: Sequence diagram for View Profile.....	129
Figure 69: Sequence diagram for Complete Profile	130
Figure 70: Sequence diagram for Edit Profile.....	130
Figure 71: Sequence diagram for View Account Details.....	130
Figure 72: Sequence diagram for Signup	131
Figure 73: Sequence diagram for Login.....	131
Figure 74: Sequence diagram for Logout	132
Figure 75: Sequence diagram for Forgot Password	133
Figure 76: Sequence diagram for Cancel Order	133
Figure 77: Software Architecture	134
Figure 78: Class Diagram.....	135
Figure 79: Database Diagram.....	136
Figure 80: Network Diagram	137
Figure 81: Collaboration diagram for Car Search	137
Figure 82: Collaboration diagram for Advance Filters	137
Figure 83: Collaboration diagram for View Car Details	138
Figure 84: Collaboration diagram for Report Car Listing.....	138
Figure 85: Collaboration diagram for Checkout	139
Figure 86: Collaboration diagram for View Activity	139

Figure 87: Collaboration diagram for Add to Favorite	140
Figure 88: Collaboration diagram for Feedback	141
Figure 89: Collaboration diagram for List Your Car	141
Figure 90: Collaboration diagram for Payment Withdraw Request.....	142
Figure 91: Collaboration diagram for View Posted Car	142
Figure 92: Collaboration diagram for Track Your Car	143
Figure 93: Collaboration diagram for Deactivate Car Posting.....	143
Figure 94: Collaboration diagram for View Trip History	144
Figure 95: Collaboration diagram for View Active Trips.....	145
Figure 96: Collaboration diagram for Messages	146
Figure 97: Collaboration diagram for View Profile	146
Figure 98: Collaboration diagram for Complete Profile	147
Figure 99: Collaboration diagram for Edit Profile	147
Figure 100: Collaboration diagram for View Account Details	148
Figure 101: Collaboration diagram for Signup	148
Figure 102: Collaboration diagram for Login.....	149
Figure 103: Collaboration diagram for Logout	149
Figure 104: Collaboration diagram for Forgot Password.....	150
Figure 105: Collaboration diagram for Cancel Order	151
Figure 106: Screen shot Car Search	183
Figure 107: Screen shot Advance Search.....	183
Figure 108: Screen shot Car Details Lessee	184
Figure 109: Screen shot Report Car Listing.....	184
Figure 110: Screen shot Add to favorite	185
Figure 111: Screen shot Feedback	185
Figure 112: Screen shot List your car.....	187
Figure 113: Payment withdraw request	187
Figure 114: Screen shot View Posted Car	188
Figure 115: Screen shot Track your car.....	188
Figure 116: Screen shot View Active Trips.....	189
Figure 117: Screen shot Messages.....	190
Figure 118: Screen shot View Profile	190
Figure 119: Screen shot Complete Profile	191
Figure 120: Screen shot Edit Profile.....	191
Figure 121: Screen shot View Account Details	192
Figure 122: Screen shot Signup.....	192
Figure 123: Screen shot Login	193
Figure 124: Screen shot Forgot password	193
Figure 125: Screen shot Cancel order	194

List of Tables

Table 1: Use case for Car search.....	40
---------------------------------------	----

Table 2: Use case for Advance Filters	41
Table 3: Use case for View car details for Lessee	42
Table 4: Use case for Report car listing.....	43
Table 5: Use case for View Activity.....	44
Table 6: Use case for View Activity.....	45
Table 7: Use case Add to favorite.....	47
Table 8: Use case for Feedback	48
Table 9: Use case for List your car	50
Table 10: Use case for Payment withdraw request.....	51
Table 11: Use case for View posted cars	53
Table 12: Use case for Track your car	54
Table 13: Use case for Deactivate car posting.....	55
Table 14: Use case for View trip history	56
Table 15: Use case for View active trips	57
Table 16: Use case for Messages	58
Table 17: Use case for View profile	59
Table 18: Use case for Complete profile	61
Table 19: Use case for Edit profile	63
Table 20: Use case for View account details	64
Table 21: Use case for Signup	65
Table 22: Use case for Login.....	66
Table 23: Use case for Logout.....	67
Table 24: Use case for Forgot password.....	69
Table 25: Use case for Cancel order	70
Table 26: Test case for Car search.....	152
Table 27: Test case for Advance filters	153
Table 28: Test case for View car details	154
Table 29: Test case for Report car listing	155
Table 30: Test case for Checkout.....	156
Table 31: Test case for View Activity	157
Table 32: Test case for Add to favorite	158
Table 33: Test case for Feedback.....	159
Table 34: Test case for List your car	162
Table 35: Test case for Payment withdraw request	164
Table 36: Test case for View posted cars	165
Table 37: Test case for track your car.....	166
Table 38: Test case for Deactivate car posting	167
Table 39: Test case for View trip history.....	167
Table 40: Test case for View active trips.....	168
Table 41: Test case for Messages	169
Table 42: Test case for View profile.....	170
Table 43: Test case for Complete profile.....	172
Table 44: Test case for Edit profile.....	174
Table 45: Test case for View account details	174
Table 46: Test case for Signup.....	176
Table 47: Test case for Login	177

Table 48: Test case for Logout	178
Table 49: Test case for Forgot password	180
Table 50: Test case for Cancel order	181

1 CHAPTER:1 INTRODUCTION

1.1 Introduction

Have you ever tried to hire a car on rent? The struggle and long process makes it so difficult that the user chooses taxis and other ride hailing apps for daily commute. The user has to go through the individual websites or mobile apps to search for a good and affordable car. Although there are few companies out there providing a user-friendly website or a mobile app to get things done.

The next option comes of visiting the car rental showrooms by own. How you are going to check the car is in good condition and safe to drive on road. Not everyone is a mechanic, right? And we all know the condition of rented cars. Here comes the role of CarSoul which will ensure the quality of car to customer by security checks and standards to maintain.

Moreover, the monopoly of the car renting industry makes renting prices so high that are not affordable for a middle or a lower-class man. And there are no proper maintenance and security of the car.

The lessor will be able to list his car on the application. The lessor can provide the hours in which the car will be available for availability. The lessor can also apply restrictions whether the car will only be available for a 2-hour drive or the entire week. He/She can offer rates of his/her choice considering the rate provided by his/her competitors on the application. The lessor will receive requests from the lessee to rent his/her cars to which he can respond.

The lessee will be able to view the cars by category, brand, or a model/variant. He can check for the availability of the car and request to rent the car by filling out a form.

The rating system will help the lessee in ensuring the lessor and the car can be trusted and have a good reputation on the application. The lessee can also read reviews provided by other lessees which will help him/her to make better decision.

1.2 Lessor and Lessee can maintain their profile

- Lessor can create his profile where he can upload his car pictures, specifications, details, pricing and other factors. He can track his car live on map for security. He can check his past earnings from his cars.
- Lessee can create his profile where he can search for different cars and hire them for rent. He can view his past spending's list of cars he used before.

1.3 Objectives

Provide users the power of peer to peer car renting and eliminating the monopoly of car renting companies.

A platform where a lessee can get rent a suitable car within budget in couple of minutes.

A platform which allow lessor to post his/her details car on the application and start earning through renting his/her car.

1.4 Problem Statement

In Pakistan, there are no website or mobile application that allow a common person to hire a car on rent with ease. The long process makes it so difficult that the user chooses taxis and other ride hailing apps for daily commute. The person has to visit the car rental showrooms by own. How you are going to check the car is in good condition and safe to drive on road but not everyone is a mechanic. There is also not a single first-rate online platform in Pakistan that allow the lessor to post the car details for renting.

Moreover, the monopoly of the car renting industry makes difficult to the middle-class man to afford the renting car due to the high price.

1.4.1 Assumptions

We have made the following assumptions in the scope of our project.

- The application will run smoothly on android operating system available on different mobile brands.
- The end users own a mobile phone.
- The mobile phone has a stable internet connection.
- The end users are familiar with interface of other mobile applications like CarSoul.

1.4.2 Constraints

Following are the constraints in the scope of our project.

- The application is only available in English language.
- The time frame for project development was less than a year.
- Encouraging lessors to rent out their car and ensuring the security of their vehicle.

1.5 Project Scope

CarSoul mobile application allow the user (lessee) to get a car on rent within minutes by providing some essential information. This platform provides the user to get suitable, within budget and best quality car ensure by CarSoul. This mobile application also provides the user (lessor) to list his/her car for renting by providing car details. The lessor can apply restrictions whether the car will only be available for a 2-hour drive or the entire week.

When the lessee selects suitable car than he/she can pay the bill through credit/debit card. After deduction of the CarSoul charges remaining amount will send to the lessor account.

2 Chapter: 2 Requirements Analysis

2.1 Literature Review

1) Rentmycar:

Rentmycar.pk is a website which allows users to hire a car for rent owned by local owners. They are also you to rent out your car and earn good money from it. They currently operating in Karachi and Islamabad. [1]

2) Zipcar:

Zipcar is an American based company working on the principle of car sharing. It allows users to instantly select a car, provide trip details and let you know the estimated charges. It can be used to hire a car for both short and long term. It is one of the most successful companies of USA.[2]

3) Zoomcar:

Zoomcar is an Indian based company working on the same principle as zipcar. Users can rent out a car based on their choice for both short term and long term. All the cars are owned by the company itself.[3]

4) Getaround:

Getaround is an American peer-to-peer car sharing company currently operating in some of the states of USA. Private car owners can rent out their car to other people where they can use them by paying some specified charges by the car owner.[4]

5) Turo:

Turo is another American based peer-to-peer car sharing company currently operating in USA and recently expanded to UK as well. The turo works on the same principle as getaround app where people can earn money from their cars and people who are not able to afford a car can use a decent car in minimal charges.[5]

Looking at the traffic conditions of Pakistan. We came up with the idea of CarSoul which works on the principle of peer-to-peer car sharing but unlike western countries, unfortunately the security conditions here in Pakistan are not so good. Also, people lack trust when it comes to hand over an asset of worth 2-5 million to a total stranger. The CarSoul is different in a way that the primary focus of the application is to break the monopoly of car renting industry and off course providing an affordable solution to commute in cities. Companies like zoomcar which owns all its the cars does not give freedom to the consumer. Considering the demographics and attitude of people in Pakistan. CarSoul will provide safety assurance to the car owners by offering features like live car tracking and some other. This type of features is missing in other applications. Moreover, the company will provide stable car insurance in case of any accident or mishap.

2.2 Stakeholder's List

- Lessor.
- Lessee.
- Developers.
- Project Supervisor.
- Car renting Industry.

2.3 Requirement Elicitation

2.3.1 Functional Requirements

FR01: Car Search

Req. No.	Functional Requirements
FR01-01	The system shall enable the lessee to enter name of location to search for cars.
FR01-02	The system shall enable the lessee to choose current location by activating GPS of device.
FR01-03	The system shall enable the lessee to select starting date for trip.
FR01-04	The system shall enable the lessee to select ending date for trip.
FR01-05	The system shall enable the lessee to press the search button.

FR02: Advance Filters

Req. No.	Functional Requirements
FR02-01	The system shall enable the lessee to filter the results of cars
FR02-02	The system shall enable the lessee to sort the car list by pricing.
FR02-03	The system shall enable the lessee to filter the car list by Vehicle type.
FR02-04	The system shall enable the lessee to filter the car list by Model.
FR02-05	The system shall enable the lessee to filter the car list by Make.

FR02-06	The system shall enable the lessee to filter the car list by Year of manufacture.
FR02-07	The system shall enable the lessee to filter the car list by Color.

FR03: View Car Details for Lessee

Req. No.	Functional Requirements
FR03-01	The system shall enable the lessee to view Car name.
FR03-02	The system shall enable the lessee to view Car model.
FR03-03	The system shall enable the lessee to view name of Car host.
FR03-04	The system shall enable the lessee to view ratings of Car.
FR03-05	The system shall enable the lessee to view the pricing of Car per hour/per day.
FR03-06	The system shall enable the lessee to change the trip dates.
FR03-07	The system shall enable the lessee to change the pickup and drop off location.
FR03-08	The system shall enable the lessee to view specifications of car.

FR04: Report Car Listing

Req. No.	Functional Requirements
FR04-01	The system shall enable the lessee to report the car listed.
FR04-02	The system shall enable the lessee to choose the reason for which car is being reported.
FR04-03	The system shall enable the lessee to submit the report.

FR05: Checkout

Req. No.	Functional Requirements
FR05-01	The system shall enable the lessee to view amount of total bill.
FR05-02	The system shall enable the lessee to view car details.
FR05-03	The system shall enable the lessee to view trip dates.
FR05-04	The system shall enable the lessee to view trip location.
FR05-05	The system shall enable the lessee to add payment info if not added in signup.
FR05-06	The system shall enable the lessee to send a custom message to lessor.
FR05-07	The system shall enable the lessee to choose the payment method.
FR05-08	The system shall enable the lessee to click on confirm checkout button.

FR06: View Activity

Req. No.	Functional Requirements
FR06-01	The system shall enable the lessee to view history of cars he/she has booked in the past.
FR06-02	The system shall enable the lessee to see the recent notifications.

FR07: Add to Favourite

Req. No.	Functional Requirements
FR07-01	The System shall enable the lessee to add any car to his/her favorites list.
FR07-02	The System shall enable the lessee to view car details.

FR07-03	The System shall enable the lessee to view the favorite icon.
FR07-04	The System shall enable the lessee to press on the favorite icon.
FR07-05	The System shall save the car details in database as favorite.

FR08: Feedback

Req. No.	Functional Requirements
FR08-01	The System shall enable the lessee to give feedback to lessee.
FR08-02	The System shall enable the lessee to give feedback to lessee's car.
FR08-03	The System shall enable the lessor to choose the star ratings for lessee car.
FR08-04	The System shall enable the lessor to choose the star ratings for lessee's car.
FR08-05	The System shall enable the lessor to enter feedback message.
FR08-06	The System shall save the feedback information in database.
FR08-07	The System shall enable the lessor to view successful feedback message.

FR09: List your Car

Req. No.	Functional Requirements
FR09-01	The System shall enable the lessor to list his/her car in the application.
FR09-02	The System shall enable the lessor to enter car's Model Year.
FR09-03	The System shall enable the lessor to enter car's Maker company name.
FR09-04	The System shall enable the lessor to enter car's Model name.
FR09-05	The System shall enable the lessor to enter car's Mileage.
FR09-06	The System shall enable the lessor to enter car's Transmission (Auto or

	Manual).
FR09-07	The System shall enable the lessor to press the Next button to move onto the next form.
FR09-08	The system shall validate the entered information and moved to the next form.
FR09-09	The System shall enable the lessor to upload profile picture if not available.
FR09-10	The System shall enable the lessor to enter mobile number.
FR09-11	The System shall enable the lessor to press the send code button.
FR09-12	The System shall allow the lessor to enter the code.
FR09-13	The System shall enable the lessor to press the Next button to move on the next form.
FR09-14	The system shall validate the entered information and moved to next form.
FR09-15	The System shall enable the lessor to select driving license issuing country.
FR09-16	The System shall enable the lessor to enter the driving license number.
FR09-17	The System shall enable the lessor to enter first, middle and last name as it is written on the driving license.
FR09-18	The System shall enable the lessor to select date of birth.
FR09-19	The System shall enable the lessor to press the Next button to move on the next form.
FR09-20	The system shall validate the entered information and moved to next form.
FR09-21	The System shall enable the lessor to enter advance notice for his/her car.
FR09-22	The System shall enable the lessor to enter Minimum trip duration
FR09-23	The System shall enable the lessor to press the Next button to move on the next form.
FR09-24	The system shall validate the entered information and moved to next form.

FR09-25	The System should enable the lessor to enter car license number.
FR09-26	The System should enable the lessor to check available features of the cars
FR09-27	The System shall enable the lessor to enter car description.
FR09-28	The System shall enable the lessor to upload car pictures.
FR09-29	The System shall enable the lessor to press the Next button to move onto the next form.
FR09-30	The system shall validate the entered information and moved to the next form.
FR09-31	The System shall enable the lessor to press the Agree with terms button.
FR09-32	The System shall enable the lessor to press the publish button.
FR09-33	The system shall save the complete record in database.

FR10: Payment withdraw request

Req. No.	Functional Requirements
FR10-01	The System shall enable the lessor to send payment withdraw request.
FR10-02	The System shall enable the lessor to enter his/her amount which he/she want to withdraw from CarSoul account.
FR10-03	The System shall enable the lessor to enter his/her IBAN number.
FR10-04	The System shall enable the lessor to enter his/her CNIC number.
FR10-05	The System shall enable the lessor to enter his/her Full name.
FR10-06	The System shall enable the lessor to enter his/her date of birth.
FR10-07	The System shall enable the lessor to enter his/her area postal code.
FR10-08	The System shall enable the lessor to enter his/her address.
FR10-09	The System shall enable the lessor to click check box button to accept the

	terms and conditions.
FR10-10	The System shall enable the lessor to press the send request button.
FR10-11	The System shall validate all the information.
FR10-12	The System shall save the withdraw payment request record in the database.

FR11: View Posted Cars

Req. No.	Functional Requirements
FR11-01	The System shall enable the lessor to view his/her cars details.
FR11-02	The System shall enable the lessor to view car's name.
FR11-03	The System shall enable the lessor to view car's modal year.
FR11-04	The System shall enable the lessor to view car's ratings.
FR11-05	The System shall enable the lessor to view the status (active or deactivated) of the car.
FR11-06	The System shall enable the lessor to view the deactivate button.
FR11-07	The System shall enable the lessor to view the pictures of the cars.
FR11-08	The System shall enable the lessor to scroll and view all the cars details.

FR12: Track your Car

Req. No.	Functional Requirements
FR12-01	The System shall enable the lessor to track car's live location on google maps.
FR12-02	The System shall enable lessee to turn on the GPS of his/her mobile phone.

FR12-03	The System shall send location information to the server.
FR12-04	The System shall save location information in the database.
FR12-05	The System shall send location information to the lessee mobile phone.
FR12-06	The System shall enable lessor to view the his/her car's live location on google maps.

FR13: Deactivate Car Posting

Req. No.	Functional Requirements
FR13-01	The System shall enable the lessor to deactivate his/her car from listing.
FR13-02	The System shall enable lessee to click on the deactivate button.
FR13-03	The System shall enable the lessor to click on the confirm button.
FR13-04	The System shall update the database and change the car status.
FR13-05	The System shall show the message that car has been deactivated.

FR14: View Trip History

Req. No.	Functional Requirements
FR14-01	The System shall enable the lessor to view all trip details in a list.
FR14-02	The System shall enable the lessor to view car name.
FR14-03	The System shall enable the lessor to view car Model.
FR14-04	The System shall enable the lessor to view car rating
FR14-05	The System shall enable the lessor to view trip charges.
FR14-06	The System shall enable the lessor to view trip duration.

FR14-07	The System shall enable the lessor to view trip start date.
FR14-08	The System shall enable the lessor to view trip end date.
FR14-09	The System shall enable the lessor to view lessee name.
FR14-10	The System shall enable the lessor to view lessee rating.
FR14-11	The System shall enable the lessor to view message button to send message to lessee.
FR14-12	The System shall enable the lessor to view profile button to view the lessee profile.

FR15: View Active Trips

Req. No.	Functional Requirements
FR15-01	The System shall enable the lessor to view all active trips details in a list.
FR15-02	The System shall enable the lessor to view car name.
FR15-03	The System shall enable the lessor to view car Model.
FR15-04	The System shall enable the lessor to view car rating.
FR15-05	The System shall enable the lessor to view trip charges.
FR15-06	The System shall enable the lessor to view trip duration.
FR15-07	The System shall enable the lessor to view trip start date.
FR15-08	The System shall enable the lessor to view trip end date.
FR15-09	The System shall enable the lessor to view lessee name.
FR15-10	The System shall enable the lessor to view lessee rating.
FR15-11	The System shall enable the lessor to view message button to send message to lessee.
FR15-12	The System shall enable the lessor to view profile button to view the lessee profile.
FR15-13	The System shall enable the lessor to view cancel trip button.

FR16: Messages

Req. No.	Functional Requirements
FR16-01	The System shall enable the lessor and lessee to send and receive messages.
FR16-02	The System shall enable the lessor and lessee to type the text in text field.
FR16-03	The System shall enable the lessor and lessee to select emoji from collection.
FR16-04	The System shall enable the lessor to press the send button to send message.
FR16-05	The System shall enable the lessor to view the message sent symbol.

FR17: View Profile

Req. No.	Functional Requirements
FR17-01	The System shall enable the lessor to view his/her name.
FR17-02	The System shall enable the lessor to view date of joining.
FR17-03	The System shall enable the lessor to view email.
FR17-04	The System shall enable the lessor to view his/her picture.
FR17-05	The System shall enable the lessor to view his/her phone number.
FR17-06	The System shall enable the lessor to view his/her reviews from other lessees.

FR18: Complete Profile

Req. No.	Functional Requirements
FR18-01	The System shall enable the lessor to complete his/her profile.
FR18-02	The System shall enable the lessor to view his/her email.
FR18-03	The System shall enable the lessor to upload his/her picture.
FR18-04	The System shall enable the lessor to enter the date of birth.
FR18-05	The System shall enable the lessor to enter his/her phone number.
FR18-06	The System shall enable the lessor to enter the code.
FR18-07	The System shall validate all information.
FR18-08	The System shall save all the information in the database.

FR19: Edit Profile

Req. No.	Functional Requirements
FR19-01	The System shall enable the lessor to edit his/her profile by clicking on edit profile button.
FR19-02	The System shall enable the lessor to edit his/her name.
FR19-03	The System shall enable the lessor to upload his/her picture.
FR19-04	The System shall enable the lessor to enter his/her phone number.
FR19-05	The System shall enable the lessor to receive OTP code.
FR19-06	The System shall enable the user to enter the OTP code.
FR19-07	The system shall validate the OTP code.
FR19-08	The system shall enable the lessor to press the save changes button.
FR19-09	The System shall update information in the database.

FR19-10	The System shall show confirmation of profile edited successfully.
---------	--

FR20: View Account details

Req. No.	Functional Requirements
FR20-01	The System shall enable the lessor to view his/her account details.
FR20-02	The System shall enable the lessor to view total available balance.
FR20-03	The System shall enable the lessor to view the Transactions history.
FR20-04	The System shall enable the lessor to view the withdraw payment request button.
FR20-05	The System shall enable the lessor to view the close account button.

FR21: Sign Up

Req. No.	Functional Requirements
FR21-01	The System shall enable the lessor to register an account.
FR21-02	The System shall enable the lessor to enter his/her email.
FR21-03	The System shall enable the lessor to enter his/her name.
FR21-04	The System shall enable the lessor to enter password.
FR21-05	The System shall enable the lessor to press the signup Button.
FR21-06	The System shall validate all the information.
FR21-07	The System shall save the record in the database.
FR21-08	The System shall enable the lessor to view the success message.

FR22: Login

Req. No.	Functional Requirements
FR22-01	The System shall enable the lessor to login into the app.
FR22-02	The System shall enable the lessor to enter the email.
FR22-03	The System shall enable the lessor to enter password.
FR22-04	The System shall enable the lessor to press the login button.
FR22-05	The System shall enable the lessor to press the forget password button if he/she does not remember his/her password.

FR23: Logout

Req. No.	Functional Requirements
FR23-01	The System shall enable the lessor to logout from the app.
FR23-02	The System shall enable the lessor to press the logout button.
FR23-03	The System shall enable the lessor to press the confirm button.
FR23-04	The System shall save the current state of the system in database.
FR23-05	The System shall enable the lessor to view successful logout message.

FR24: Forgot Password

Req. No.	Functional Requirements
-----------------	--------------------------------

FR24-01	The System shall enable the lessee and lessor to select the forgot password option.
FR24-02	The System shall enable the lessee and lessor to send the password reset email to him/her.
FR24-03	The System shall enable the lessee and lessor to enter the OTP code received on his/her email.
FR24-04	The System shall validate the OTP code.
FR24-05	The System shall enable the lessee and lessor to enter the new password.
FR24-06	The System shall update the information in the database.
FR24-07	The System shall enable the lessee and lessor to view the password changed success message.

FR25: Cancel Order

Req. No.	Functional Requirements
FR25-01	The System shall enable the lessee and lessor to cancel the order he/she made.
FR25-02	The System shall enable the lessee and lessor to press the cancel order button.
FR25-03	The System shall enable the lessee and lessor to enter reason of cancellation.
FR25-04	The System shall delete the order from the database.
FR25-05	The System shall enable the lessee and lessor to view the success message.

2.3.2 Non-Functional Requirements:

NFR01: Security

NFR01-01	The System must only allow users to view their profile after authenticating the correct credentials.
NFR01-02	Users must be forced to create a strong password with a combination of small letters, capital letters and numbers.
NFR01-03	The System must ensure that the lesser earnings should only be visible to authorized users.
NFR01-04	The System must ensure that the passwords are not visible by default at login and signup screens.
NFR01-05	The users must be notified by an email when a change in password happens.

NFR02: Performance

NFR02-01	The average loading time of the home screen should not be more than 10 seconds.
NFR02-02	The average System response time should not be more than 5 seconds.
NFR02-03	The System must successfully run on a 4g network or equivalent.
NFR02-04	The System must successfully run on a device with 2 GB of RAM or above.

NFR03: Usability

NFR03-01	The user must be able to use the System without any training.
NFR03-02	A new user must be able to fluently use the application after 30 minutes of usage.
NFR03-03	The average mistakes made by the users must not be more than 5 per day.

NFR04: Availability

NFR04-01	The System must be available for use between 5am to 12am for users.
----------	---

NFR04-02	The payment withdraw option should be available 99% of the time for the user irrespective of the time.
NFR04-03	In terms of System updates the system must not be unavailable for more than 5 minutes 95% of the time.

NFR05: Safety

NFR05-01	The System must not allow any users to hire cars between 12am to 5am.
NFR05-02	The System must not allow users under 18 years to hire cars for rent.

2.3.3 Requirement Traceability Matrix

ID	Associate ID	Requirement Description	Use case ID	Test Case ID	Activity Diagram ID	Collaboration Diagram ID	Sequence Diagram ID	Actor
1	FR01	Car Search	UC ID 01	TC ID 01	AD ID 01	CD ID 01	SD ID 01	Lessee
2	FR02	Advance Filters	UC ID 02	TC ID 02	AD ID 02	CD ID 02	SD ID 02	Lessee
3	FR03	View Car Details for Lessee	UC ID 03	TC ID 03	AD ID 03	CD ID 03	SD ID 03	Lessee
4	FR04	Report Car Listing	UC ID 04	TC ID 04	AD ID 04	CD ID 04	SD ID 04	Lessee
5	FR05	Checkout	UC ID 05	TC ID 05	AD ID 05	CD ID 05	SD ID 05	Lessee
6	FR06	View Activity	UC ID 06	TC ID 06	AD ID 06	CD ID 06	SD ID 06	Lessee/Lessor
7	FR07	Add to Favorite	UC ID 07	TC ID 07	AD ID 07	CD ID 07	SD ID 07	Lessee
8	FR08	Feedback	UC ID 08	TC ID 08	AD ID 08	CD ID 08	SD ID 08	Lessee
9	FR09	List your Car	UC ID 09	TC ID 09	AD ID 09	CD ID 09	SD ID 09	Lessor

10	FR10	Payment withdraw request	UC ID 10	TC ID 10	AD ID 10	CD ID 10	SD ID 10	Lessor
11	FR11	View Posted Cars	UC ID 11	TC ID 11	AD ID 11	CD ID 11	SD ID 11	Lessor
12	FR12	Track your Car	UC ID 12	TC ID 12	AD ID 12	CD ID 12	SD ID 12	Lessor
13	FR13	Deactivate car posting	UC ID 13	TC ID 13	AD ID 13	CD ID 13	SD ID 13	Lessor
14	FR14	View Trip History	UC ID 14	TC ID 14	AD ID 14	CD ID 14	SD ID 14	Lessee/Lessor
15	FR15	View Active Trips	UC ID 15	TC ID 15	AD ID 15	CD ID 15	SD ID 15	Lessee/Lessor
16	FR16	Messages	UC ID 16	TC ID 16	AD ID 16	CD ID 16	SD ID 16	Lessee/Lessor
17	FR17	View Profile	UC ID 17	TC ID 17	AD ID 17	CD ID 17	SD ID 17	Lessee/Lessor
18	FR18	complete profile	UC ID 18	TC ID 18	AD ID 18	CD ID 18	SD ID 18	Lessee/Lessor
19	FR19	Edit Profile	UC ID 19	TC ID 19	AD ID 19	CD ID 19	SD ID 19	Lessee/Lessor
20	FR20	View Account details	UC ID 20	TC ID 20	AD ID 20	CD ID 20	SD ID 20	Lessor
21	FR21	Signup	UC ID 21	TC ID 21	AD ID 21	CD ID 21	SD ID 21	Lessee/Lessor
22	FR22	Login	UC ID 22	TC ID 22	AD ID 22	CD ID 22	SD ID 22	Lessee/Lessor
23	FR23	Logout	UC ID 23	TC ID 23	AD ID 23	CD ID 23	SD ID 23	Lessee/Lessor
24	FR24	Forgot Password	UC ID 24	TC ID 24	AD ID 24	CD ID 24	SD ID 24	Lessee/Lessor
25	FR25	Cancel Order	UC ID 25	TC ID 25	AD ID 25	CD ID 25	SD ID 25	Lessee/Lessor

2.4 Use Case Descriptions:

2.4.1 Car Search

Use caseID:001	Use case Name: Car Search
Priority: High	
Actors: Lessee	
Use Case Summary	Car Search allows lessee to search a car in a specific city by providing starting and ending dates for trip.
Pre-condition:	1. The information of car was previously saved in the system and the lessee wants to search the car.
Normal Flow of Events	Alternative Path
1. The Use case starts when the lessee wants to search for a car to hire it for rent.	1a. If the lessee does not confirm car search, the information will not be displayed.
2. The lessee enters city name or select the current city by activating GPS.	
3. The lessee enters starting and ending dates for his/her trip.	
4. The user presses the search button.	4b. Cancel Action
5. The system responds with list of cars.	
Exception 1	
6. This Use case ends	
Exceptions / Alerts	
1. The system cannot display a car which is not saved in the database.	
Post Conditions	

Step#	Description
1:	A list of cars will be displayed.
Use Case Cross References	
Includes	None
Extends	None

Table 1: Use case for Car search

2.4.2 Advance Filter

Use caseID: 002	Use case Name: Advance Filters
Priority: Medium	
Actors: Lessee	
Use Case Summary	Advance filters allow lessee to filter out results of car search which are not relevant to him/her. It allows him/her to narrow down the results of car search.
Pre-condition	1. The lessee searched for car and a list of cars was displayed by the system.
Normal Flow of Events	Alternative Path
1. The Use case starts when the lessee wants to apply different filters on car search results. Filters such as sort by price, search by vehicle type, Model, Make, Year of manufacture and color.	1a. If the lessee does not apply the filters, the system will not change the car search results.
2. The lessee selects one, some or all these filters.	
3. The system responds with updating the search results.	
Exception 1	
4. This Use case ends.	
Exceptions / Alerts	
1. The system will not display any result if none of the car match those filter criteria.	

Post Conditions	
Step#	Description
1.	An updated list of cars will be displayed.
Use Case Cross References	
Includes	None
Extends	None

Table 2: Use case for Advance Filters

2.4.3 View Car Details

Use caseID: 003	Use case Name: View Car Details
Priority: High	
Actors: Lessee	
Use Case Summary	View car details allow lessee to view details of any car.
Pre-condition	1. The information of car was previously saved in the system and the lessee wants to view it.
Normal Flow of Events	
1. The Use case starts when the user clicks on any car picture to view its details like its name, model, host, ratings, pricing and other specifications of car.	1a. If the lessee does not click on the car picture, the system will not display the details of car.
2. The lessee clicks on the car picture.	
Exception 1	
3. The system responds with details of car.	
4. This Use case ends.	
Exceptions / Alerts	
1. The system will not allow lessee to view the details of car whose information is not available.	

Post Conditions	
Step#	Description
1.	The details of cars will be displayed.
Use Case Cross References	
Includes	None
Extends	None

Table 3: Use case for View car details for Lessee

2.4.4 Report Car Listing

Use caseID: 004	Use case Name: Report Car Listing
Priority: Low	
Actors: Lessee	
Use Case Summary	Report car listing allows lessee to report a car which is violating the application terms and conditions to get it reviewed.
Pre-condition	1. The information of car was previously saved in the system and the lessee wants to report it.
Normal Flow of Events	Alternative Path
1. The Use case starts when the lessee wants to report a car listing.	1a. If the lessee does not report the car. It will not be reviewed.
2. The lessee view a car listing and finds it that it is violating the terms and conditions of CarSoul application.	
Exception 1	
3. The lessee selects the report option.	3a. Cancel Action
4. The lessee selects the reason for reporting the car listing.	
Exception 2	

5. The system responds the user by saying that your report has been submitted and will be reviewed soon. Thanks for reporting it.	
6. This Use case ends.	
Exceptions / Alerts	
1. The system will not allow the lessee to report the same car listing twice. 2. The system will not allow the lessee to choose the reason for report that is not available.	
Post Conditions	
Step#	Description
1.	The system will remove the reported car listing from list of cars shown to lessee.
Use Case Cross References	
Includes	Car Details for Lessee
Extends	None

Table 4: Use case for Report car listing

2.4.5 Checkout:

Use caseID: 005	Use case Name: Checkout
Priority: High	
Actors: Lessee	
Use Case Summary	The checkout allows the lessee to confirm a car to hire for rent by making its payment.
Pre-condition	1. The car was available to hire for rent.
Normal Flow of Events	Alternative Path
1. The Use case starts when the lessee wants to hire a car for rent and make payment for it.	1a. If the lessee does not want to hire the car. The system will not display the checkout details.

2. The lessee clicks on the checkout option. Exception 1	2a. Cancel Action.
3. The lessee review the details of checkout like total bill, trip dates and location, payment method.	3a. If the lessee has not added the payment method and its details in signup. The system will allow him/her to add the payment method.
4. The lessee can write a custom message for the owner of the car.	
5. The lessee will click on the confirm order option.	5a. Cancel Action.
6. The system will respond user by saying your order has been placed successfully.	
7. This Use case ends.	
Exceptions / Alerts	
1. The system will not allow lessee to checkout a car which is not available for hiring.	
Post Conditions	
Step#	Description
1.	The order list of lessee will be updated.
Use Case Cross References	
Includes	None

Table 5: Use case for View Activity

Use caseID006	Use case Name: View Activity
Priority: Medium	
Actors: Lessee	

Use Case Summary	View Activity allow the lessee to view all the details of the activities (s)he has performed. Lessee can view the history of cars (s)he booked in the past and any important notification available.
Pre-condition:	1. Lessee must be logged in to view his/her Activity information.
Normal Flow of Events	Alternative Path
1. This use case start when lessee wants to view his/her activities and click on activity button.	1.aif lessor does not click on activity button to view activity then system will not show activity.
2.System responds to Lessee by displaying his/her Activity information. Exception 1	
3 This use case end.	
Exceptions / Alerts	
1 The system shall not allow Lessee to view his/her Activity detail if the user's logged In session is expired.	
Post Conditions	
Step#	Description
1:	Information will be Successfully displayed.
Use Case Cross References	
Includes	Login
Extends	None

Table 6: Use case for View Activity

2.4.7 Add to Favourite

Use caseID007

Use case Name: Add to Favorite

Priority: Medium	
Actors: Lessee	
Use Case Summary	Add to favorite allow the lessee to save car as its favorite car. The lessee click on the heart icon of the car post than this car post save as favorite car in database.
Pre-condition:	1. Lessee must be logged In to saved his/her car as favorite.
Normal Flow of Events	Alternative Path
1. This use case start when lessee wants to save a car post as favorite car.	1.a if lessor does not confirm to save the car as favorite than system will not save it.
2.Lessee click on the heart icon which displaying on the car post.	
3 System will responds to lessee by filling the heart icon with color.	3.aif post already saved as favorite than it will unsaved it. System will show transparent heart icon.
4 This use case end.	
Exceptions / Alerts	
None	
Post Conditions	
Step#	Description
1:	Heart icon will be filled with color.
Use Case Cross References	
Includes	None
Extends	None

Table 7: Use case Add to favorite

2.4.8 Feedback

Use caseID008		Use case Name: Feedback
Priority: Medium		
Actors: Lessee		
Use Case Summary	Feedback allow the lessee to give the feedback about the car. System allow the lessee to choose stars ratings for car and add text for comments.	
Pre-condition:	<ol style="list-style-type: none"> 1. Lessee must be logged In. 2. Lessee must has booked a car. 	
Normal Flow of Events		Alternative Path
1. This use case start when lessee wants to give feedback about the car by click on feedback button.		1.a if lessee does not click on feedback button system will not show feedback form.
2. Lessee click on the stars icons to give the stars rating.		
3 System will responds to lessee by filling the star Icons with color.		
4 Lessee will add text in the text box as feedback of the car.		
5 Lessee will click on submit to post feedback.		5.a Lessee can click on Cancel button.
6 The use case end.		
Exceptions / Alerts		
None		
Post Conditions		

Step#	Description
1:	Star icons will be filled with color.
Use Case Cross References	
Includes	None
Extends	None

Table 8: Use case for Feedback

2.4.9 List Your Car

Use caseID: 009	Use case Name: List Your Car
Priority: Medium	
Actors: Lessor	
Use Case Summary	List Your Car allow the lessor to post his/her car details on to the application so lessee can see card details and booked car for rent.
Pre-condition	1. Lessee must be logged In.
Normal Flow of Events	Alternative Path
1. This use case start when lessor wants to post his/her car details on application.	1.a if lessor does not select car list option system will not show the car post form.
2. Lessor will click on the List your car button.	
3. System will respond and show a form to lessor.	
4. Lessor will add details of the car, Model name, Model year, Maker company, Mileage, and transmission (Auto/ Manual),	
5. Lessor will click on the next button.	
6. System will respond and show the next form to the lessor. Exception 1:	6.b if any wrong information provided than system will highlight that information.

7. Lessor will upload profile picture, add mobile number,	
8. Lessor will click on the send code button.	
9. Lessor will enter opt code.	
10. Lessor will click on the next button.	
11. System will respond and show the next form to Lessor.	11.c if any wrong information provided than system will highlight that information.
12. Lessor will select driving license issuing country, enter driving license number, enter first, middle, last name. and select date of birth.	
13. Lessor will click on the next button.	
14. System will respond and show the next form to Lessor.	14.d if any wrong information provided than system will highlight that information.
15. Lessor will add advance notice time, minimum trip time and maximum trip time.	
16. Lessor will click on the next button.	
17. System will respond and show the next form to Lessor.	17.e if any wrong information provided than system will highlight that information.
18. Lessor will add car license number, check all available feature of the car, add car description and upload cars pictures.	
19. Lessor will click on the next button.	
20. System will respond and show the next form to Lessor.	17.f if any wrong information provided than system will highlight that information.
21. Lessor will click on check box button to accept the terms and conditions.	

22. Lessor will click on publish button.	
23. System will respond and show a dialog box that your car has been listed successfully.	23.g if any error occur than System will show error details and highlight the wrong information.
Exceptions / Alerts	
1. If wrong information is provided than system will not show next form.	
Post Conditions	
Step#	Description
1.	Lessor can see his/her car details in MyCar Menu.
Use Case Cross References	
Includes	None.
Extends	None.

Table 9: Use case for List your car

2.4.10 Payment Withdraw Request

Use caseID: 010	Use case Name: Payment Withdraw Request
Priority: Medium	
Actors: Lessor	
Use Case Summary	Payment Withdraw request use case allow the lessor to send payment withdraw request to CarSoul, which he/she has earned from his/her car.
Pre-condition	<ol style="list-style-type: none"> 1. Lessor must be logged In. 2. Lessor must have 20\$ in the account.
Normal Flow of Events	Alternative Path
1.This use case start when lessor want to send payment withdraw request.	1.a System will not show payment withdraw form if lessor does not confirm

	to withdraw payment.
2. Lessor will click on the payment withdraw button.	
3. System will respond and show a form to the lessor.	
4. Lessor will enter amount how much he/she want to withdraw. Exception 1:	
5. Lessor will enter IBAN number, CNIC number, name, Date of birth, area postal code, address,	
6. Lessor will click on the check box button to accept terms and conditions.	
7. Lessor will click on the send request button.	7.b Cancel Action
8. System will respond and show dialog box that your request has been submitted successfully. Exception 2:	
Exceptions / Alerts	
1. The system will send exception if lessor want to enter amount more than he/she earned. 2. The system will send exception if IBAN or CNIC are not correct.	
Post Conditions	
Step#	Description
1.	Lessor can see his/her withdraw request in the account menu.
Use Case Cross References	
Includes	None.
Extends	None.

Table 10: Use case for Payment withdraw request

2.4.11 View Posted Car

Use caseID: 011	Use case Name: View Posted Cars
Priority: Medium	
Actors: Lessor	
Use Case Summary	View Posted Cars allow the Lessor to view the details of the cars that he/she has posted or listed on the application.
Pre-condition	<ol style="list-style-type: none"> 1. Lessor must be logged In and 2. Lessor must has posted at least one car.
Normal Flow of Events	Alternative Path
1. This use case start when lessor want to view the his/her cars name, model year, ratings, status, pictures.	1. If lessor does not confirm the view car details then system will not show car details.
2. System will responds to lessor and show all details of the cars in list form. Exception 1:	
3. The use case end	
Exceptions / Alerts	
1. The system will not show car details If lessor is not logged in or lessor has not posted any car details.	
Post Conditions	
Step#	Description
1.	Lessor can see all of his/her cars details.
Use Case Cross References	
Includes	None.
Extends	None.

Table 11: Use case for View posted cars

2.4.12 Track Your Car

Use caseID: 012		Use case Name: Track Your Car
Priority: 12		
Actors: Lessor		
Use Case Summary	Track Your Car allow the lessor to see his/her car location live on map. Lessor will click on the car location button and can see car icon on map.	
Pre-condition	<ol style="list-style-type: none"> 1. Lessor must be logged In and 2. Lessor must has rented out his/her car. 	
Normal Flow of Events		Alternative Path
1. This use case start when lessor want to see his/her car live location on map.		1.a if lessor does not confirm to track his/her car. System will not show live car location.
2. Lessor will click on car location button. Exception 1:		
3. System will responds to lessor and show car icon on the map.		
Exceptions / Alerts		
2. The system will not show car on map if lessor car is not on trip.		
Post Conditions		
Step#		Description
1.		Lessor can see car icon on the map.
Use Case Cross References		
Includes	None.	

Extends	None.
----------------	-------

Table 12: Use case for Track your car

2.4.13 Deactivate Car Posting

Use caseID: 013	Use case Name: Deactivate Car Posting
Priority: Medium	
Actors: Lessor	
Use Case Summary	Deactivate Car Posting allow the Lessor to change the status of a posted car enable to disable. So, Lessee cannot view that particular car details.
Pre-condition	<ol style="list-style-type: none"> 1. Lessor must be logged In and 2. Lessor must has posted car details.
Normal Flow of Events	Alternative Path
1. This use case start when lessor want to deactivate a car posted.	1.a if lessor does not confirm the deactivate the car post then system will not deactivate the car post.
2. Lessor will click on deactivate button.	
3. System will respond to lessor and show a confirmation dialog.	
4. Lessor will click on the confirm button.	4.b Lessor can cancel that action by clicking on cancel button.
5. System will respond to Lessor and show that car has been deactivated successfully.	
Exceptions / Alerts	
None	
Post Conditions	
Step#	Description

1.	Lessor can view the activate button on car details.
Use Case Cross References	
Includes	None.
Extends	None.

Table 13: Use case for Deactivate car posting

2.4.14 View Trip History

Use caseID: 014	Use case Name: View Trip History
Priority: Medium	
Actors: Lessor, Lessee	
Use Case Summary	Track Your Car allow the lessor to see all details of the trips of his/her cars in the past.
Pre-condition	1. Lessor/lessee must be logged In.
Normal Flow of Events	Alternative Path
1. This use case start when lessor/lessee want to view the trips history of his/her cars.	1a if lessor/lessee does not confirm to view the trips history then system will not show the trips history details.
2. Lessor/lessee will click on car tips history button.	
3. System will respond to lessor/lessee and show all details of the trips his/her cars in the past in the list form.	
Exception 1:	
Exceptions / Alerts	
3. The system will not show trips details if his/her has no trip in the past.	
Post Conditions	

Step#	Description
1.	Lessor can view the details of the trips.
Use Case Cross References	
Includes	None.
Extends	None.

Table 14: Use case for View trip history

2.4.15 View Active Trips

Use caseID: 015	Use case Name: View Active Trips
Priority: Medium	
Actors: Lessor, Lessee	
Use Case Summary	View Active Trips allow the lessor/Lessee to view all the current trips of his/her cars. Lessor can see the time left for the trip and other details.
Pre-condition	<ol style="list-style-type: none"> 1. Lessor/Lessee must be logged In. 2. Lessor/Lessee has at least one of his/her car of trip.
Normal Flow of Events	Alternative Path
<ol style="list-style-type: none"> 1. This use case start when lessor want to view current trip of his/her car. 	1a if lessor/lessee does not confirm to view the active trips then system will not show the active trips details.
<ol style="list-style-type: none"> 2. Lessor will click on Active trip button. Exception 1: 	
<ol style="list-style-type: none"> 3. System will respond to lessor and show the trips details which are currently active. 	
Exceptions / Alerts	
<ol style="list-style-type: none"> 4. The system will not show any details if lessor car is not on trip. 	

Post Conditions	
Step#	Description
1.	Lessor can view the details of the active trip.
Use Case Cross References	
Includes	None.
Extends	None.

Table 15: Use case for View active trips

2.4.16 Messages

Use caseID: 016	Use case Name: Messages
Priority:	Low
Actors:	Lessor, Lessee
Use Case Summary	Messages allow lessee and lessor to have a conversation and discuss further details for the order. For example, pickup point and drop-off points.
Pre-condition	1. The order was made by the lessee and a new conversation thread was created in the system.
Normal Flow of Events	Alternative Path
1. The Use case starts when the lessee or lessor wants to send or receive a message.	1a. If the lessee/lessor does not send any text message. The system will not update it in the database.
2. The lessee/lessor opens up the messages option.	
3. The lessee/lessor types the message in the text field. He/she chooses the emoji (if want to).	
Exception 1	
4. The lessee/lessor presses the sent message button.	4a. Cancel Action
Exception 2	

5. The system responds by sending the message to the lessee/lessor.	
6. The system saves the message in the database.	
7. This Use case ends.	
Exceptions / Alerts	
1. The system will not allow lessee/lessor to send a message more than 255 characters long. 2. The system will not allow the lessee/lessor to send an empty message.	
Post Conditions	
Step#	Description
1.	The messages will be updated in the database.
Use Case Cross References	
Includes	None
Extends	None

Table 16: Use case for Messages

2.4.17 View Profile

Use caseID: 017	Use case Name: View Profile
Priority: Medium	
Actors: Lessee, Lessor	
Use Case Summary	View profile allow lessee and lessor to see details of his profile.
Pre-condition	1. The information of lessee/lessor was previously saved in the system and the lessee/lessor wants to view it.
Normal Flow of Events	Alternative Path
1. The Use case starts when the lessee/lessor wants to view his profile.	1a. If the user does not click on the view profile option. The system will not display the information.

2. The lessee/lessor clicks on the view profile option to view information like his/her name, date of joining, email, display picture, phone number, reviews and payment details.	
3. The system responds by displaying the information.	
4. This Use case ends.	
Exceptions / Alerts	
1.	
Post Conditions	
Step#	Description
1.	
Use Case Cross References	
Includes	None
Extends	None

Table 17: Use case for View profile

2.4.18 Complete Profile

Use caseID: 018	Use case Name: Complete Profile
Priority: High	
Actors: Lessee, Lessor	
Use Case Summary	Complete profile allows the lessee/lessor to enter and save the information which was left in the signup.
Pre-condition	1. The profile was created by the lessee/lessor and the information was saved in the system. He/she wants to store the information which was not stored before.
Normal Flow of Events	Alternative Path

1. The Use case starts when the lessee/lessor wants to complete the profile.	1a. If the lessee/lessor does not select the complete profile option. The system will not add the information in the database.
2. The lessee/lessor clicks on the complete profile option to add information like his/her profile picture, date of birth, phone number and payment method.	
3. The lessee/lessor selects the profile picture from his/her mobile phone gallery.	
4. The lessee/lessor adds his/her date of birth. Exception 1	
5. The lessee/lessor adds the phone number.	
6. The lessee/lessor clicks on the verify phone number option. Exception 2	
7. The system responds by sending a OTP code to the lessee/lessor mobile number.	
8. The lessee/lessor enters the OTP code.	
9. The system validates the OTP code.	
10. The system responds by saying the OTP code is verified. Exception 3	
11. The lessee/lessor chooses the payment method. Exception 4	
12. The lessee/lessor enters the payment details.	
13. The lessee/lessor selects the verify option.	
14. The system responds by saying payment method verified. Exception 5	
15. The lessee/lessor selects the save option.	11a. Cancel Action.
16. The system responds by saying the information is added.	

17. The system adds the information in the database.	
18. This Use case ends.	
Exceptions / Alerts	
<p>1. The system will not allow the lessee/lessor to select DOB less than 18 years.</p> <p>2. The system will not allow the lessee/lessor to add number greater or smaller than 11 digits.</p> <p>3. The system will not validate if the OTP code is wrong.</p> <p>4. The system will not allow lessee/lessor to select the payment method which is not listed.</p> <p>5. The system will not verify the payment method if the payment details are not correct.</p>	
Post Conditions	
Step#	Description
1.	The profile information will be added in the database.
Use Case Cross References	
Includes	None
Extends	None

Table 18: Use case for Complete profile

2.4.19 Edit Profile

Use caseID: 019	Use case Name: Edit Profile
Priority: Medium	
Actors: Lessee, Lessor	
Use Case Summary	Edit profile allows lessee/lessor to edit information which he/she stored before.
Pre-condition	1. The information was previously saved in the system and the lessee/lessor wants to change it now.
Normal Flow of Events	Alternative Path

1. The Use case starts when the user wants to edit his information like name, email, password, profile picture, phone number and payment detail.	1a. If the lessee/lessor does not select the edit profile option. The system will not update the information in the database.
2. The lessee/lessor clicks on edit profile option.	
3. The lessee/lessor edits his/her name.	3a. If the lessee/lessor does not edit his/her name. the system will not change it in the database.
4. The lessee/lessor changes his/her email and password.	4a. If the lessee/lessor does not change his/her email and password. The system will not change it in the database.
5. The lessee/lessor edits his/her profile picture.	5a. If the lessee/lessor does not edit his/her profile picture. The system will not change it in the database.
6. The lessee/lessor edits his/her payment details.	6a. If the lessee/lessor does not edit his/her payment details. The system will not change it in the database.
7. The lessee/lessor selects save button option.	7a. Cancel Action.
8. The system responds with a confirmation message. Exception 1	
9. The system updates the information in the database.	
10. This Use case ends.	

Exceptions / Alerts

- 1.** The system will not show any confirmation message if the lessee/lessor has not changed any information.

Post Conditions

Step#	Description
1.	The information will be updated in the database.

Use Case Cross References

Includes	None
Extends	None.

Table 19: Use case for Edit profile

2.4.20 View Account Details

Use caseID: 020		Use case Name: View Account Details
Priority: Medium		
Actors: Lessor		
Use Case Summary	View Account Details allow the Lessor to view all account details.	
Pre-condition	1. Lessor must be logged In .	
Normal Flow of Events		Alternative Path
1.This use case start when lessor want to view account details.		1a if lessor/lessee does not confirm to view the account details then system will not show the account details.
2. Lessor will click on the Account button. Exception 1:		2.a Cancel Action
3. System will respond to lessor and show the details of the account. Details include lessor name, email, current balance, and history of transactions.		
4 The use case end		
Exceptions / Alerts		
None		
Post Conditions		
Step#		Description
1.		Lessor can view the details of the account.
Use Case Cross References		

Includes	None.
Extends	None.

Table 20: Use case for View account details

2.4.21 Signup

Use caseID: 021 Use case Name: Sign Up	
Priority: Medium	
Actors: Lessor, Lessee	
Use Case Summary	Sign up allow the Lessor/Lessee to create an account.
Pre-condition	1. Lessor/Lessee must have a valid email.
Normal Flow of Events	
1. This use case start when Lessor/Lessee want to create an account on CarSoul.	1a if lessor/lessee does not confirm to view signup then system will not show signup details.
2. Lessor/Lessee will enter email address.	
3. Lessor/Lessee will enter his/her name.	
4. Lessor/Lessee will enter a password.	
5. Lessor/Lessee will click on the Signup button.	5.a Cancel Action
6 System will validate all fields and show OPT code field.	
Exception 1:	
7. Lessor/Lessee will enter OTP code.	
8. Lessor/Lessee will click on verify button.	
9. System will validate and show OTP code is verified message.	
Exception 2:	

10. System will respond to Lessor/Lessee that your account has been created successfully.	
Exceptions / Alerts	
<ol style="list-style-type: none"> 1. If any wrong information provided than system will highlight that information 2. If opt code is wrong than system will show error message. 	
Post Conditions	
Step#	Description
1.	Lessor/Lessee will redirect to the application dashboard.
Use Case Cross References	
Includes	None.
Extends	None.

Table 21: Use case for Signup

2.4.22 Login

Use caseID: 022	Use case Name: Login
Priority: Medium	
Actors: Lessee, Lessor	
Use Case Summary	Login allows lessee/lessor to create a new session in the application.
Pre-condition	<ol style="list-style-type: none"> 1. The email and password of lessee/lessor was stored in the database when he/she signed up. 2. The lessee/lessor is not already logged in.
Normal Flow of Events	Alternative Path
<ol style="list-style-type: none"> 1. The Use case starts when lessee/lessor wants to login. 2. The lessee/lessor enters his/her email. 	<ol style="list-style-type: none"> 1a. If the lessee/lessor does not log in. The system will not create a new session.

3. The lessee/lessor enters his/her password.	3a. If the lessee/lessor does not remember his/her password. He/she will choose forgot password option.
4. The lessee/lessor presses the login button. Exception 1	4a. Cancel Action.
5. The system validates the credentials.	
6. The system displays a confirmation message for successful login.	6a. The system displays an error message for wrong username or password.
7. This Use case ends.	
Exceptions / Alerts	
1. The system will allow the lessee/lessor to press login button if the username and password are not entered.	
Post Conditions	
Step#	Description
1.	A new session will be created for the lessee/lessor.
Use Case Cross References	
Includes	None
Extends	None

Table 22: Use case for Login

2.4.23 Logout

Use caseID: 023	Use case Name: Logout
Priority: Medium	
Actors: Lessee, Lessor	
Use Case Summary	Logout allows lessee/lessor to destroy the session which was created at the time of login.

Pre-condition	1. The lessee/lessor is already logged in.
Normal Flow of Events	Alternative Path
1. The Use case starts when lessee/lessor wants to logout.	1a. If the lessee/lessor does not select the logout option. The system will not destroy the session.
2. The lessee/lessor presses the logout button.	
3. The system responds with confirmation message.	
4. The system destroys the session.	
5. This Use case ends.	
Exceptions / Alerts	
1.	
Post Conditions	
Step#	Description
1.	The current user session will be destroyed.
Use Case Cross References	
Includes	None
Extends	None

Table 23: Use case for Logout

2.4.24 Forgot Password

Use caseID: 024	Use case Name: Forgot Password
Priority: Medium	
Actors: Lessee, Lessor	
Use Case Summary	Forgot password allows the lessee/lessor to create a new password for his/her account.

Pre-condition	<p>1. The email and password of lessee/lessor was stored in the database when the he/she signed up.</p> <p>2. The lessee/lessor is not already logged in.</p>
Normal Flow of Events	Alternative Path
1. The Use case starts when the lessee/lessor wants to create a new password for login.	1a. If the lessee/lessor does not select the forgot password option. The system will not create a new password.
2. The lessee/lessor clicks on forgot password option.	
3. The lessee/lessor clicks on send reset email option.	3a. Cancel Action.
4. The system responds by sending a reset password email containing an OTO code.	
5. The lessee/lessor enters the OTP code. Exception 1	
6. The lessee/lessor presses the verify button.	
7. The system validates the OTP code. Exception 2	
8. The lessee/lessor enters a new password.	
9. The lessee/lessor clicks on change password option.	9a. Cancel Action
10. The system responds with a confirmation message.	
11. The system updates the information the database.	
12. This Use case ends.	
Exceptions / Alerts	
<p>1. The system will not allow the lessee/lessor to enter the OTP code after 30 seconds.</p> <p>2. The system will not validate the code if it is wrong.</p>	
Post Conditions	
Step#	Description

1.	The password is updated in the database.
Use Case Cross References	
Includes	None
Extends	Node

Table 24: Use case for Forgot password

2.4.25 Cancel Order

Use caseID: 025	Use case Name: Cancel Order
Priority: Medium	
Actors: Lessee	
Use Case Summary	Cancel order allows lessee/lessor to cancel order that they made earlier.
Pre-condition	1. The order was made earlier by the lessee and saved in the database.
Normal Flow of Events	Alternative Path
1. The Use case starts when the lessee wants to cancel the order.	1a. If the lessee does not cancel the order. The system will not delete the order from database.
2. The lessee clicks on cancel order option.	
3. The system asks lessee for confirmation.	
4. The lessee clicks on Ok.	4a. Cancel Action.
5. System responds with success message.	
6. System deletes the order from the database.	
7. This Use case ends.	
Exceptions / Alerts	
1.	

Post Conditions	
Step#	Description
1.	The order will be deleted from database.
Use Case Cross References	
Includes	None
Extends	None

Table 25: Use case for Cancel order

2.5 Use Case Design's:

2.5.1 Use Case Diagram: Car Search

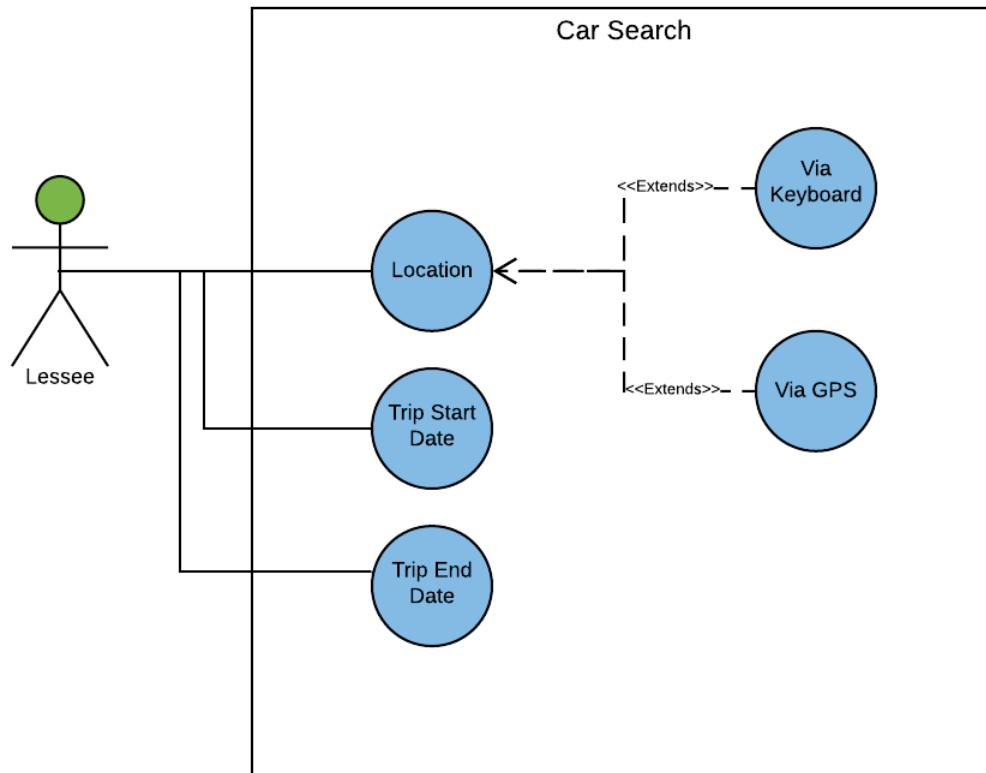


Figure 1: Use case diagram for Car search

2.5.2 Use Case Diagram: Advance Filters

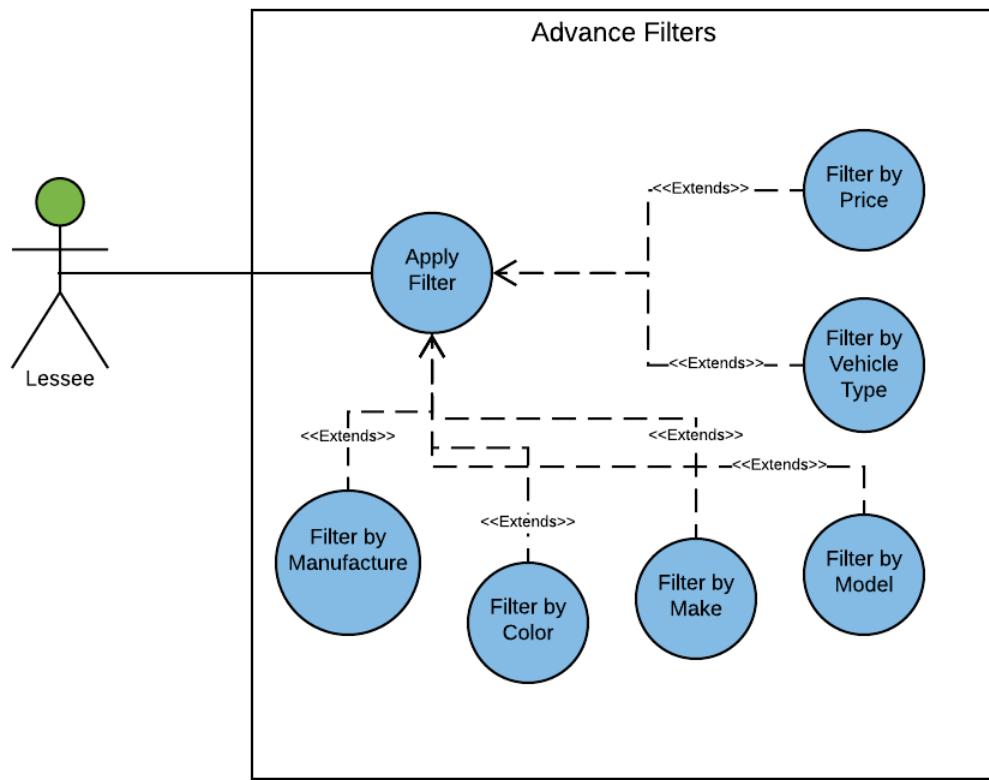


Figure 2: Use case diagram for Advance filters

2.5.3 Use Case Diagram: Car Details for Lessee

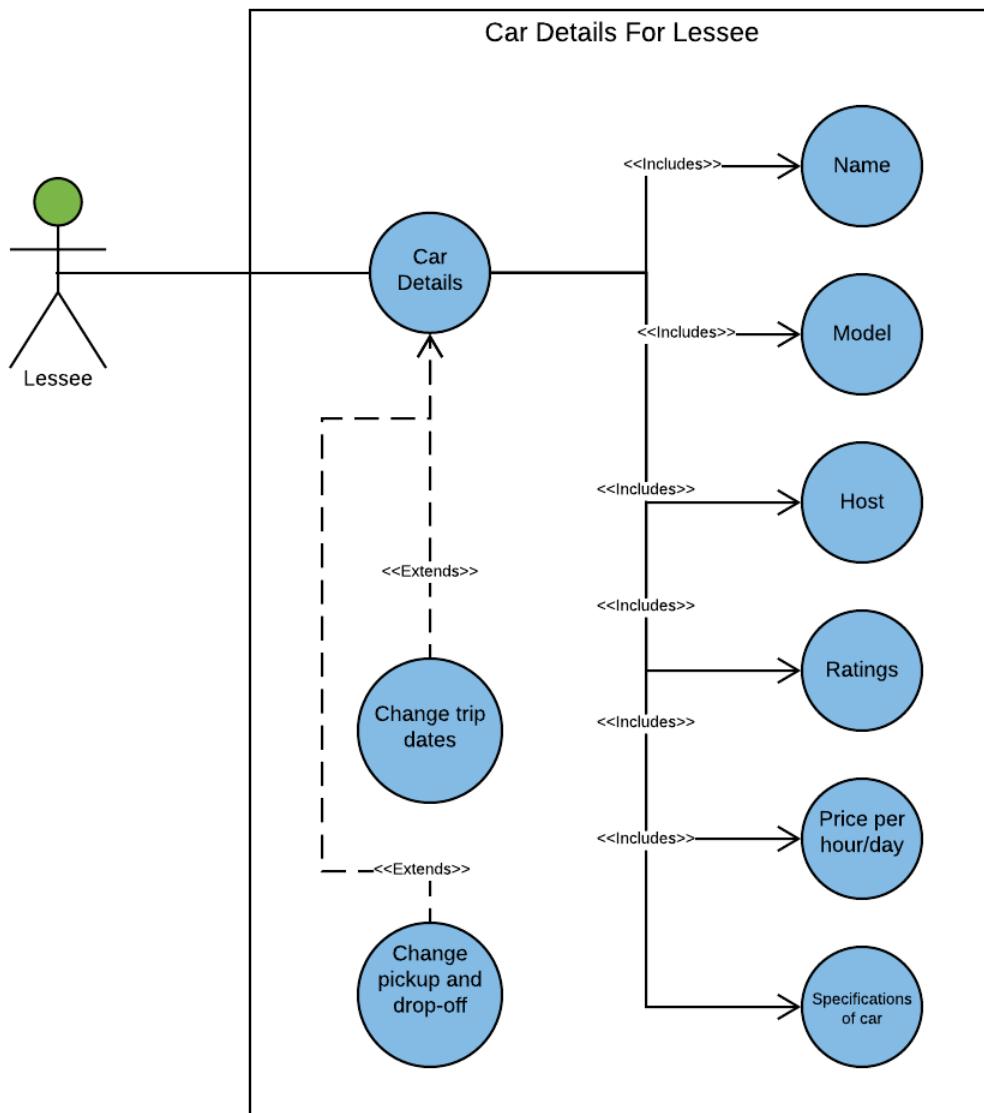


Figure 3: Use case diagram for Car details for Lessee

2.5.4 Use Case Diagram: Report Car Listing

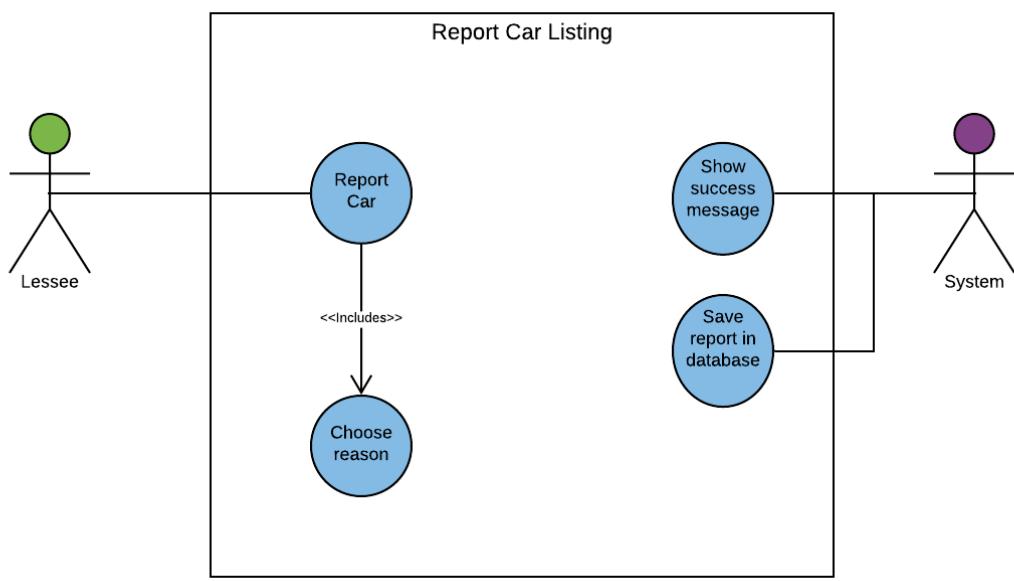


Figure 4: Use case diagram for Report car listing

2.5.5 Use Case Diagram: Checkout

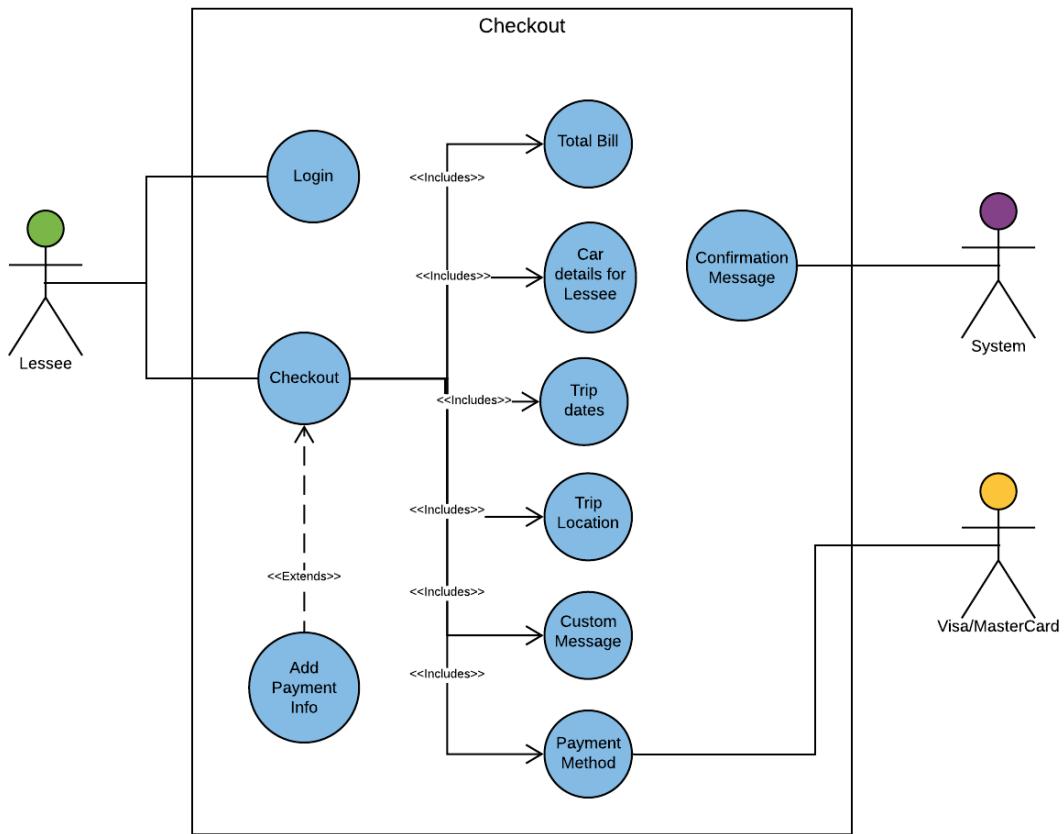


Figure 5: Use case diagram for Checkout

2.5.6 Use Case Diagram: View Activity

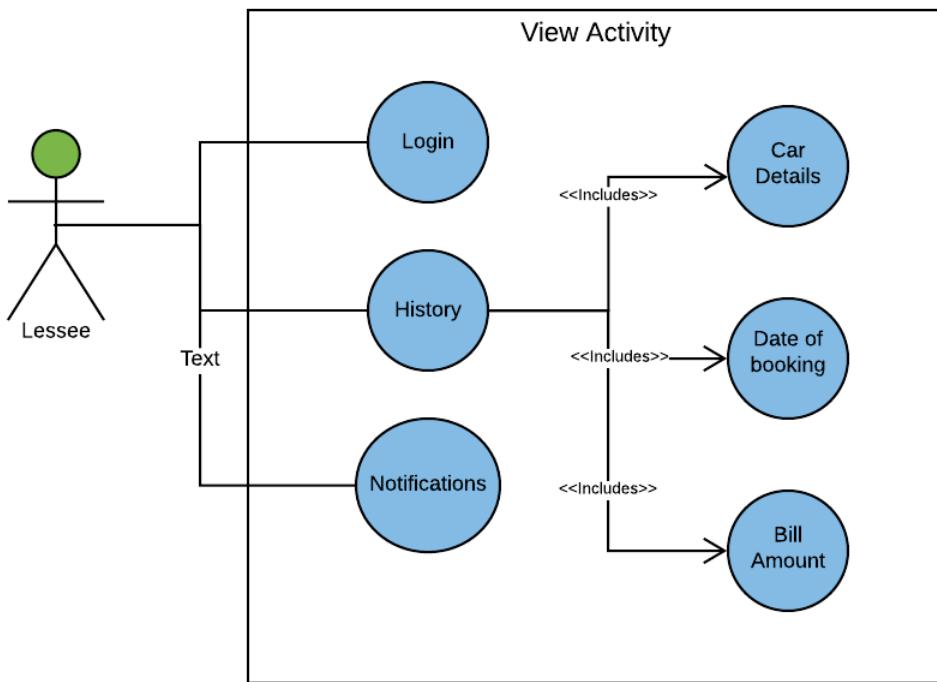


Figure 6: Use case diagram for View Activity

2.5.7 Use Case Diagram: Add to Favourite

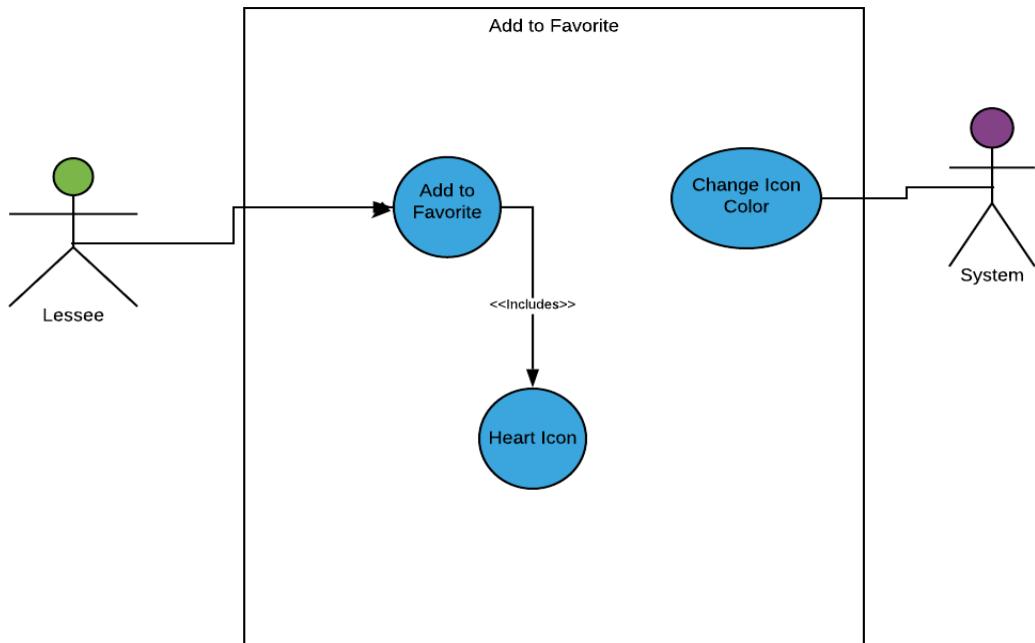


Figure 7: Use case diagram forAdd to favorite

2.5.8 Use Case Diagram: Feedback

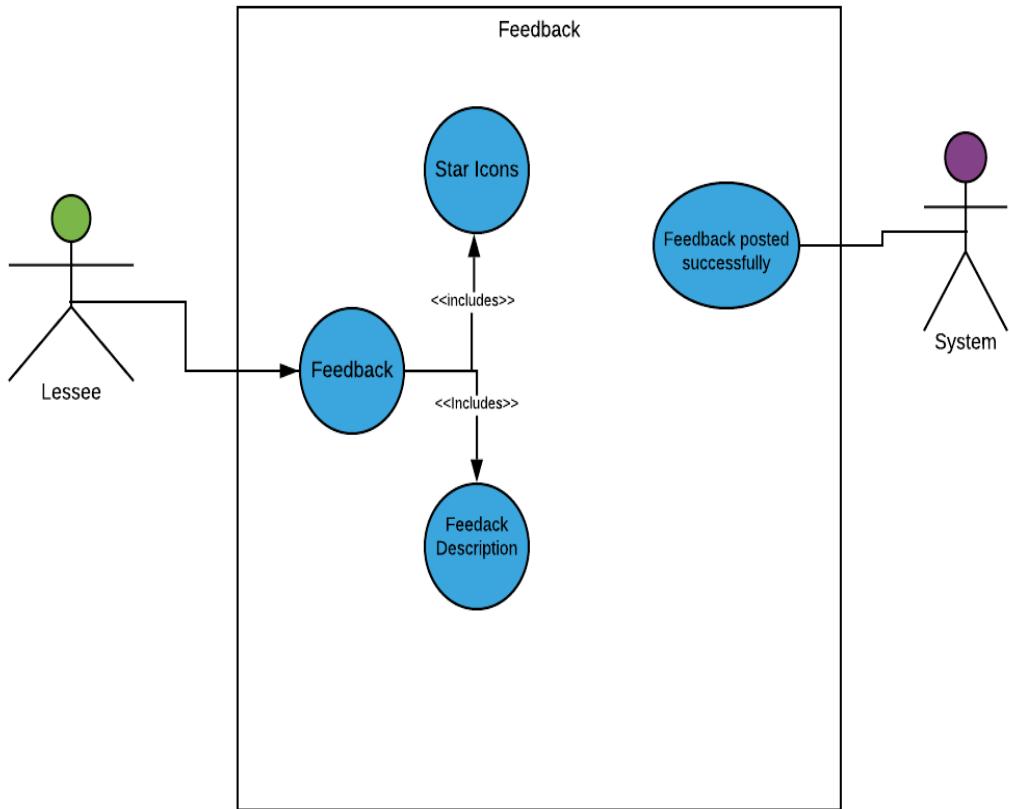


Figure 8: Use case diagram for Feedback

2.5.9 Use Case Diagram: List Your Car

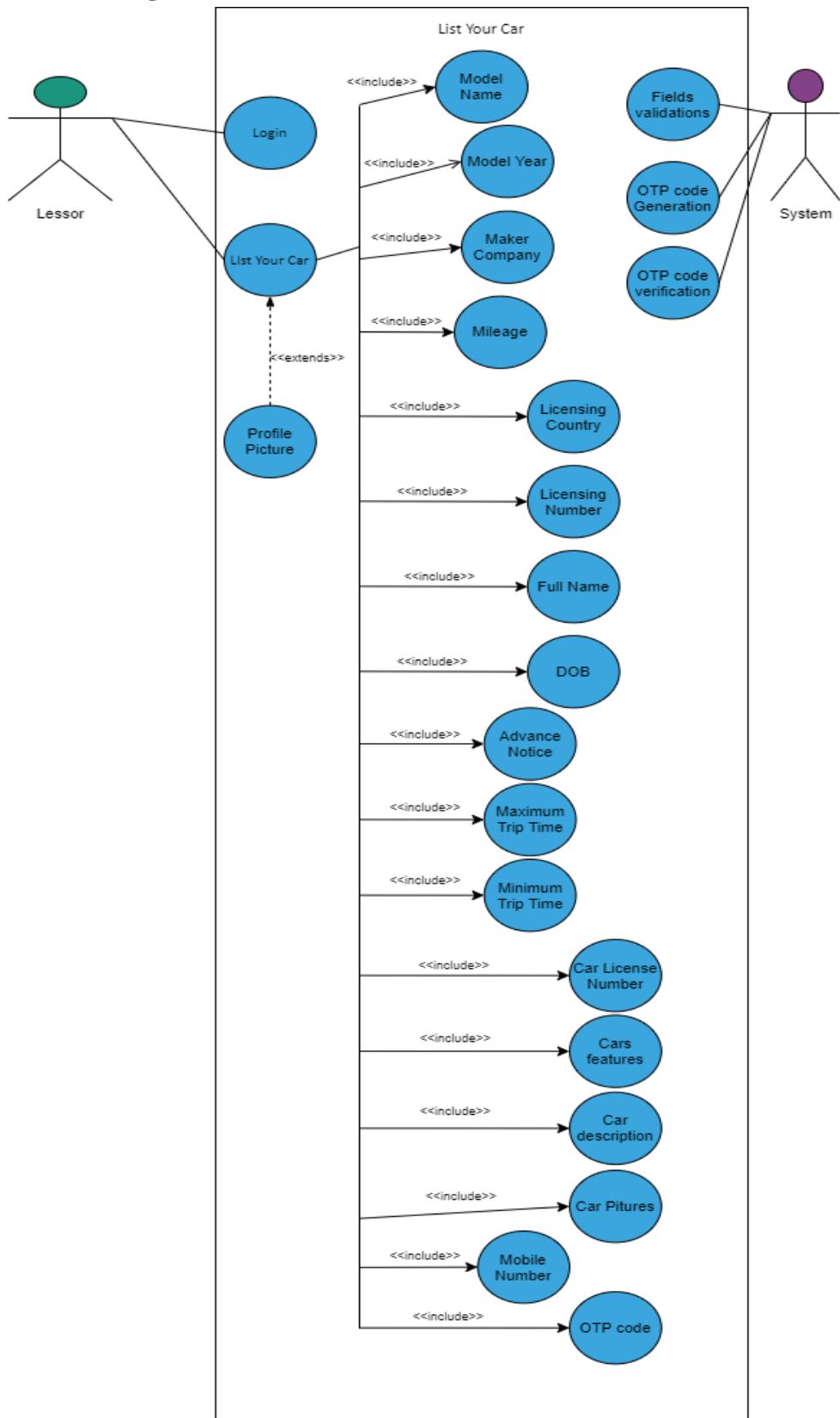


Figure 9: Use case diagram for List your car

2.5.10 Use Case Diagram: Payment Withdraw Request

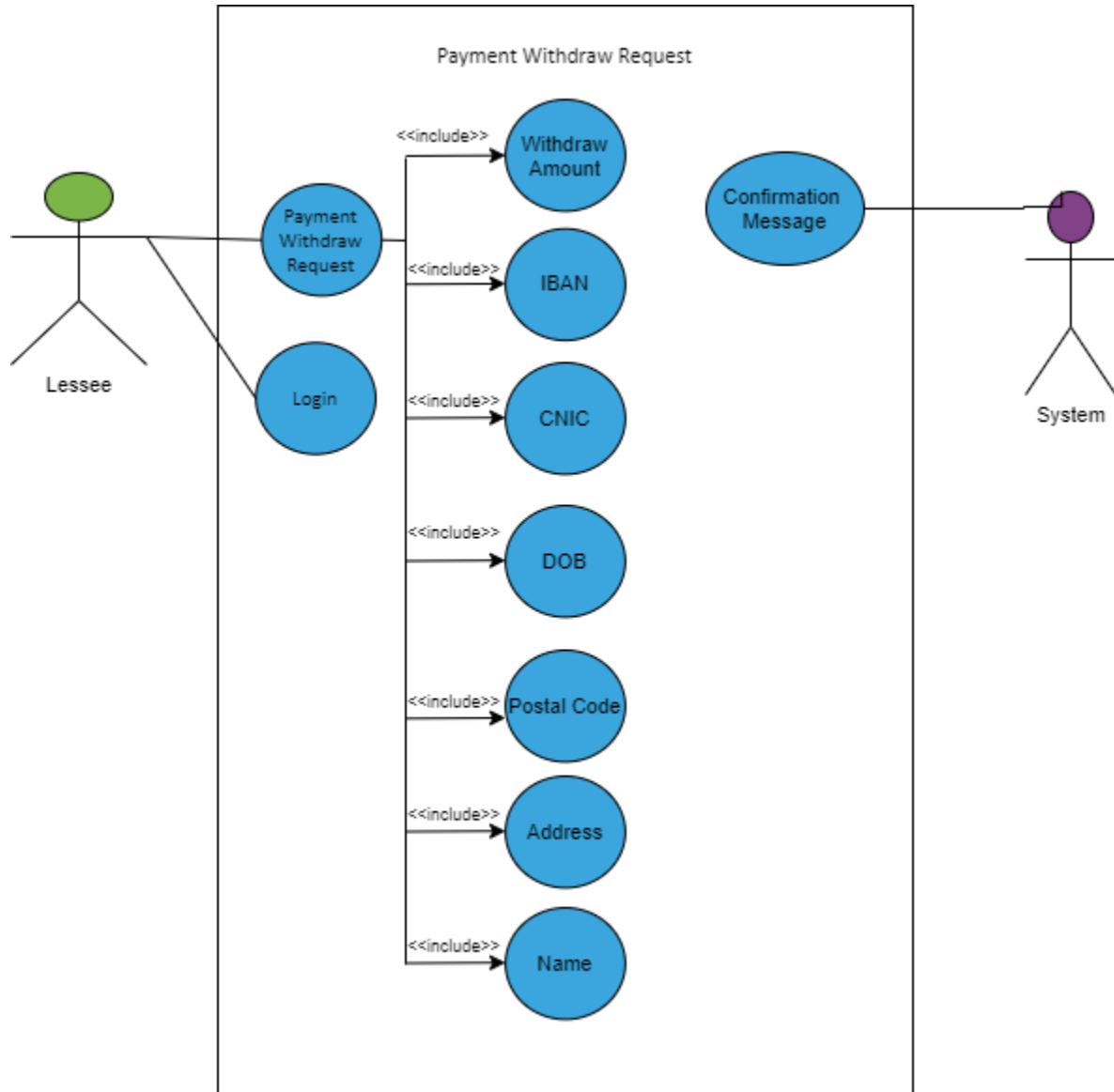


Figure 10: Use case diagram for Payment withdraw request

2.5.11 Use Case Diagram: View Posted Cars

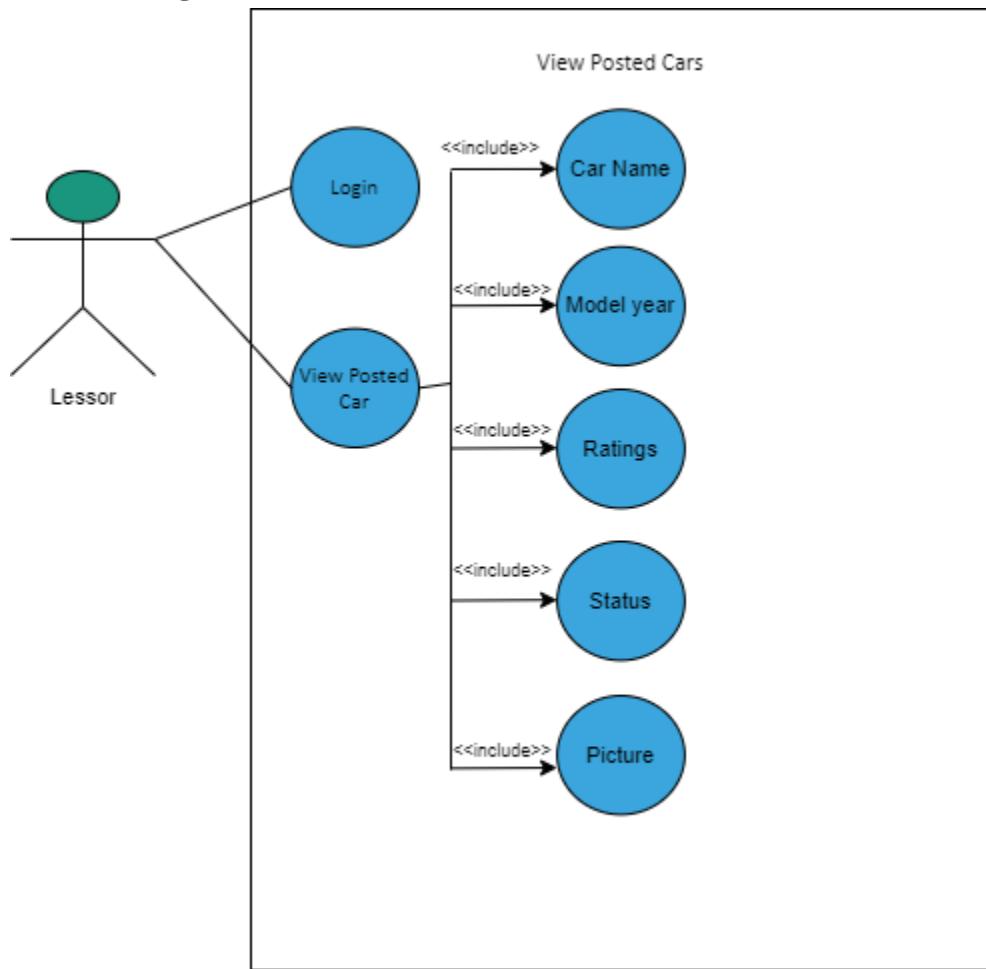


Figure 11: Use case diagram for View Posted cars

2.5.12 Use Case Diagram: Track Your Car

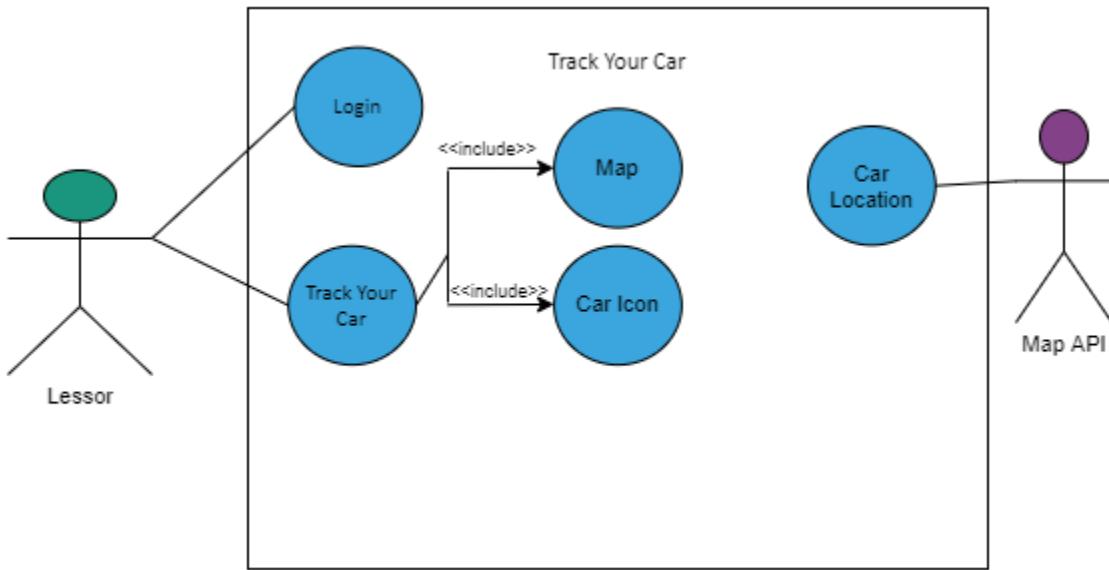


Figure 12: Use case diagram for Track your car

2.5.13 Use Case Diagram: Deactivate Car Posting

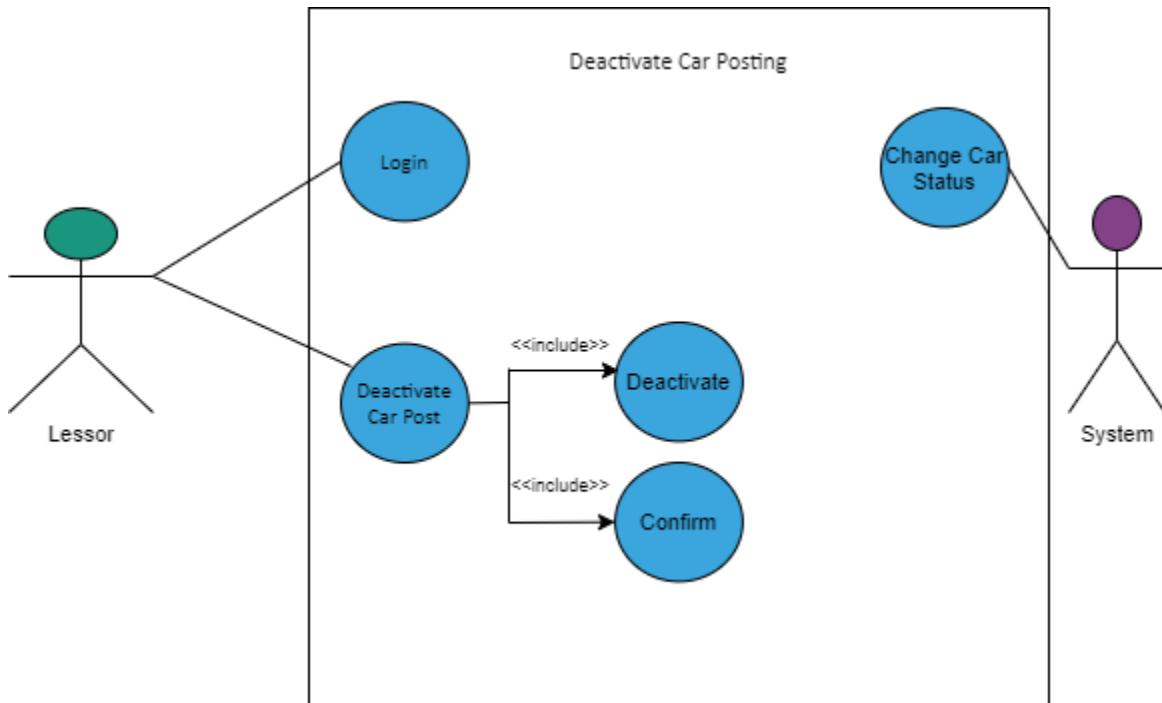


Figure 13: Use case diagram for Deactivate car posting

2.5.14 Use Case Diagram: View Trip History

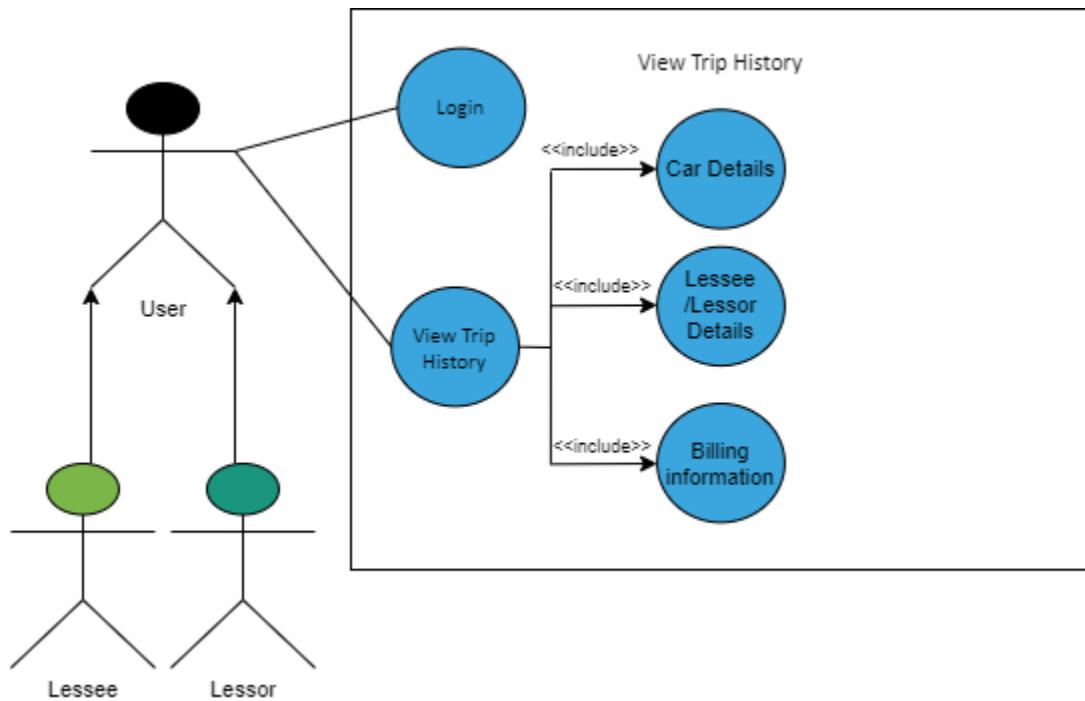


Figure 14: Use case diagram for View Trip History

2.5.15 Use Case Diagram: View Active Trip

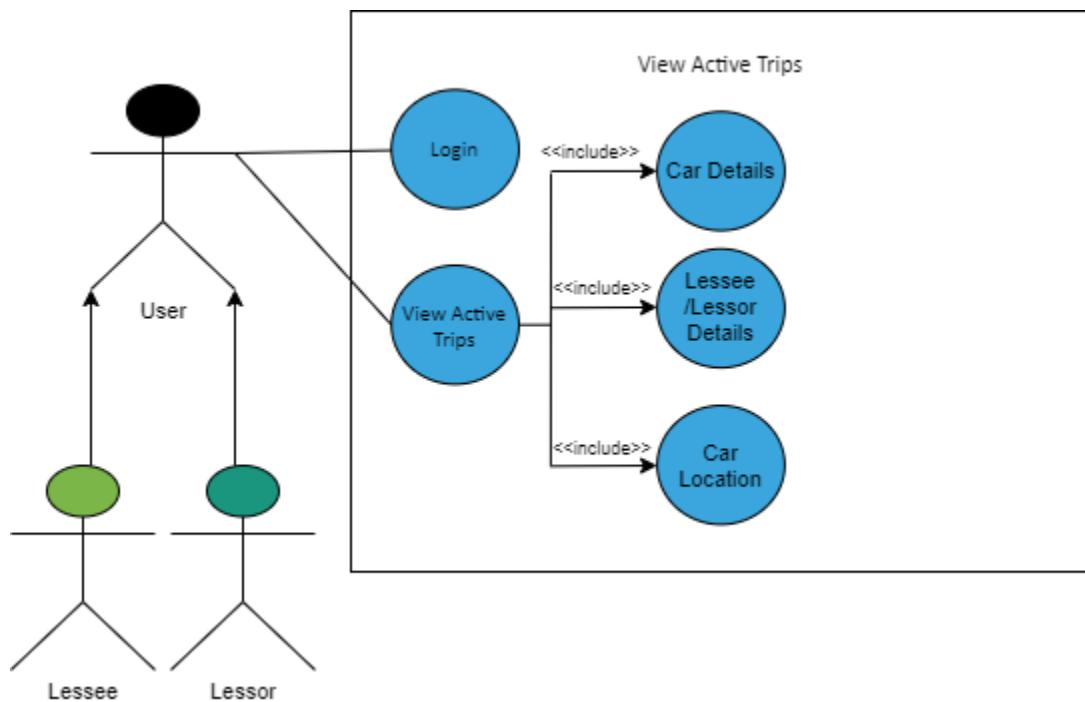


Figure 15: Use case diagram for View Active Trip

2.5.16 Use Case Diagram: Messages

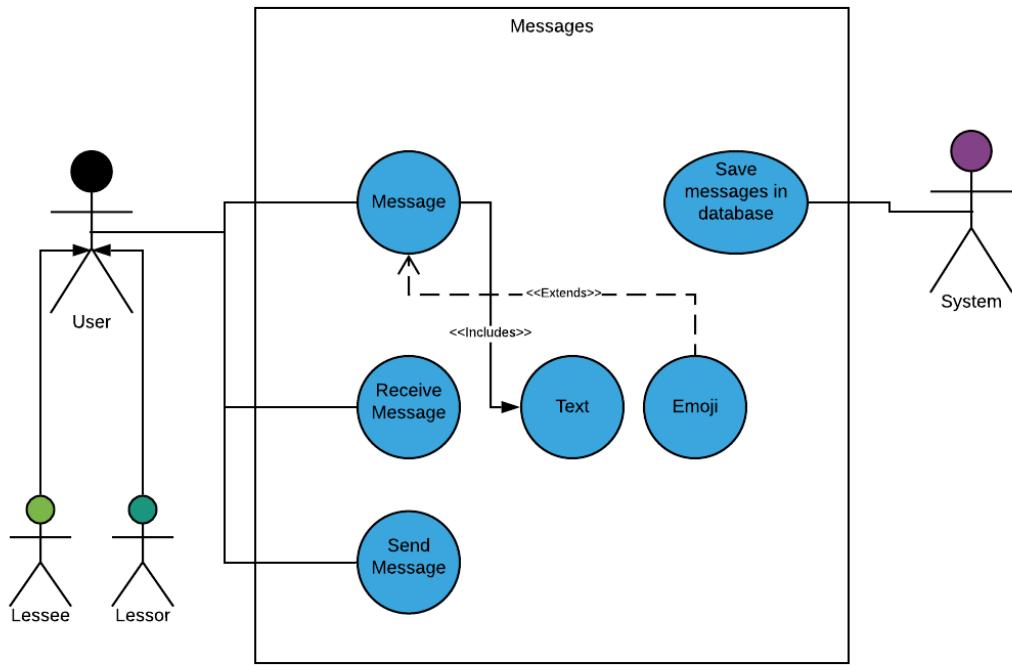


Figure 16: Use case diagram for Messages

2.5.17 Use Case Diagram: View Profile

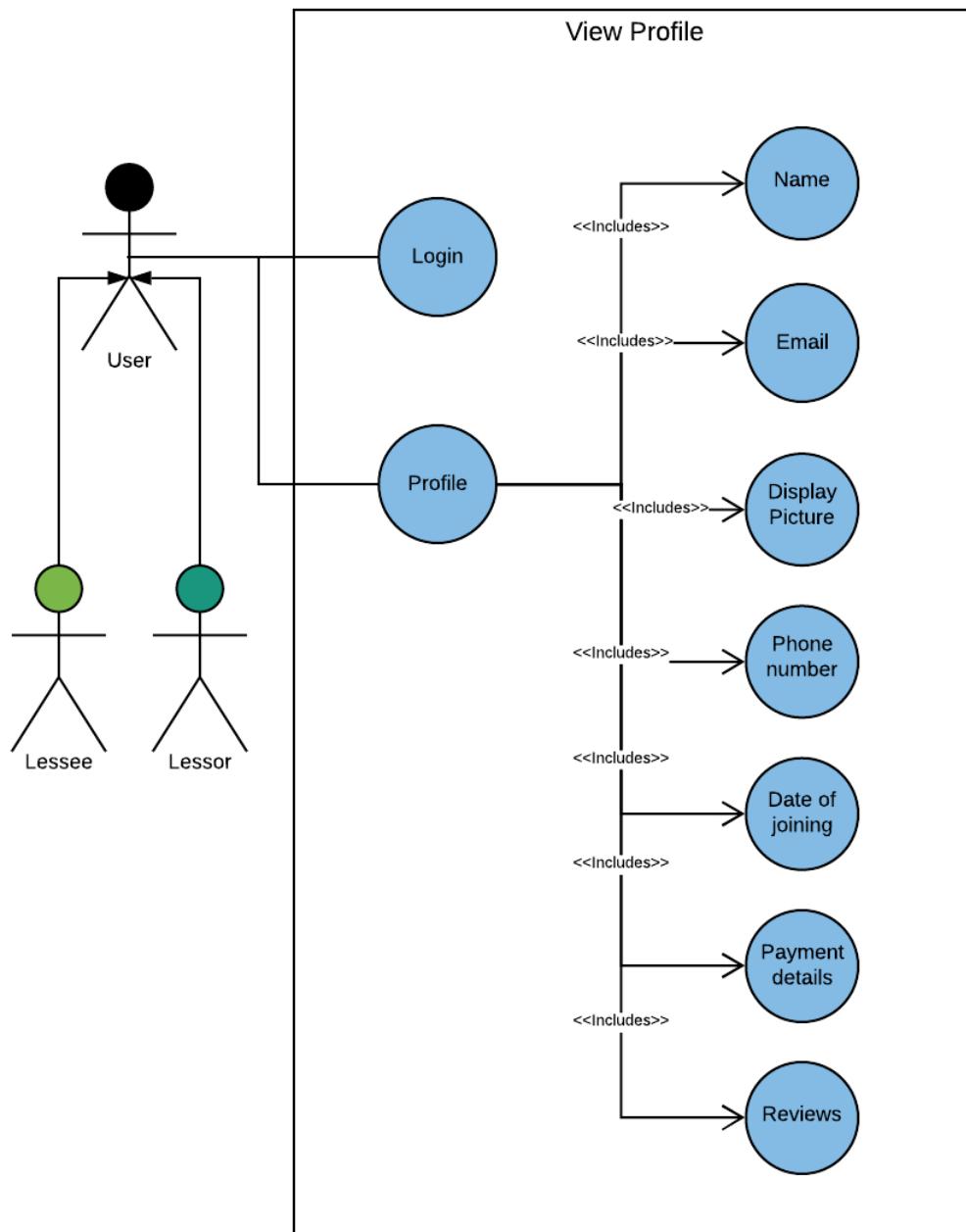


Figure 17: Use case diagram for View Profile

2.5.18 Use Case Diagram: Complete Profile

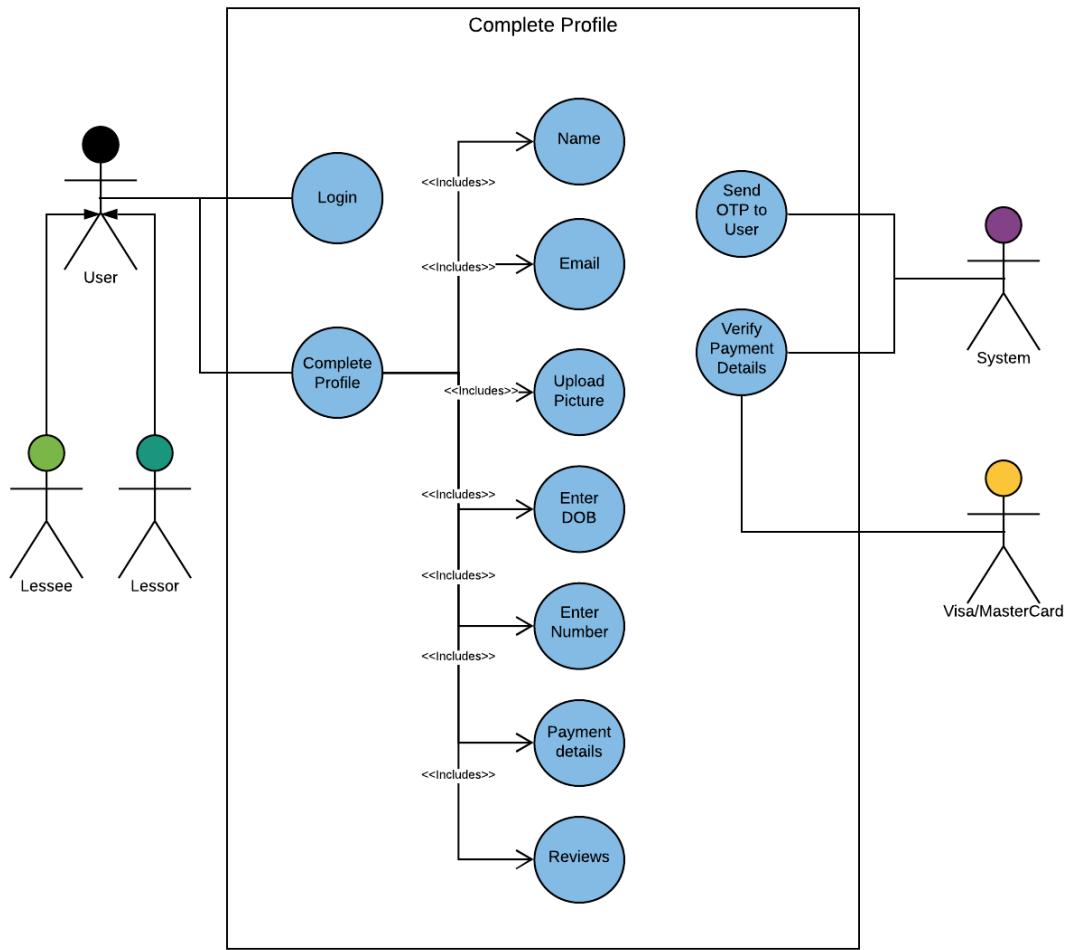


Figure 18: Use case diagram for Complete Profile

2.5.19 Use Case Diagram: Edit Profile

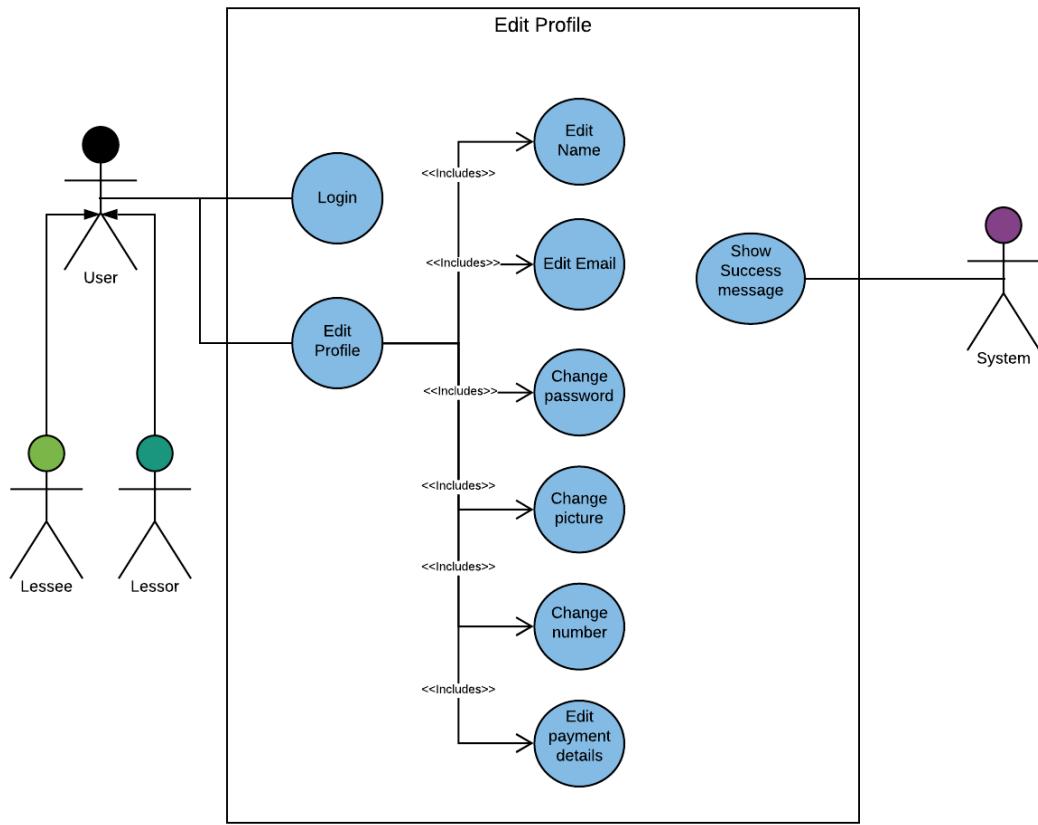


Figure 19: Use case diagram for Edit Profile

2.5.20 Use Case Diagram: View Account Details

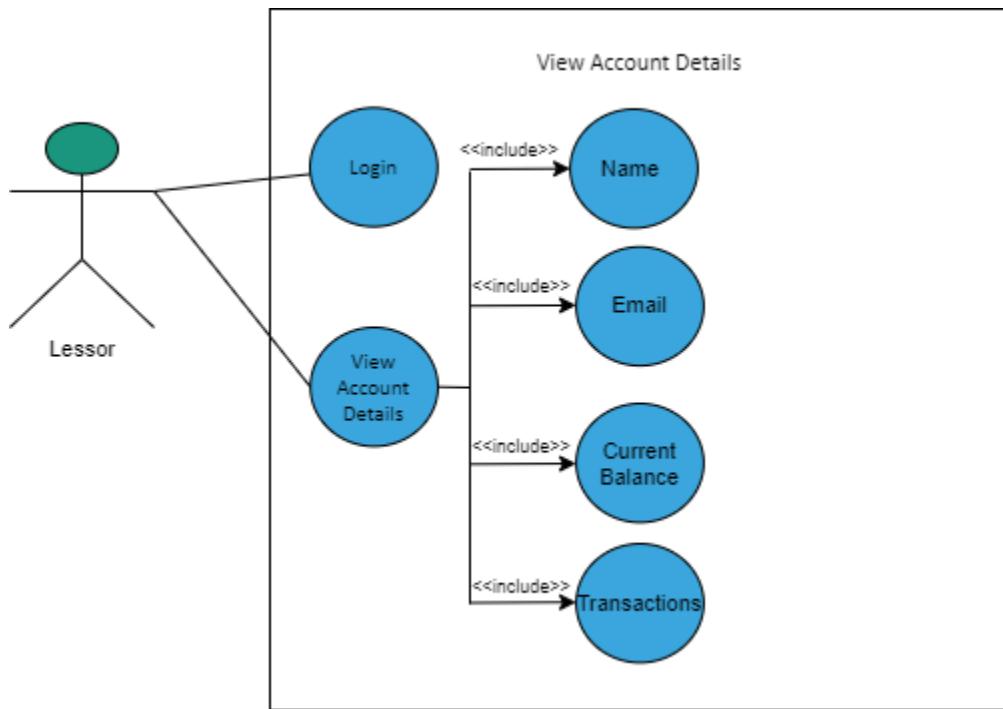


Figure 20: Use case diagram for View Account Details

2.5.21 Use Case Diagram: Signup

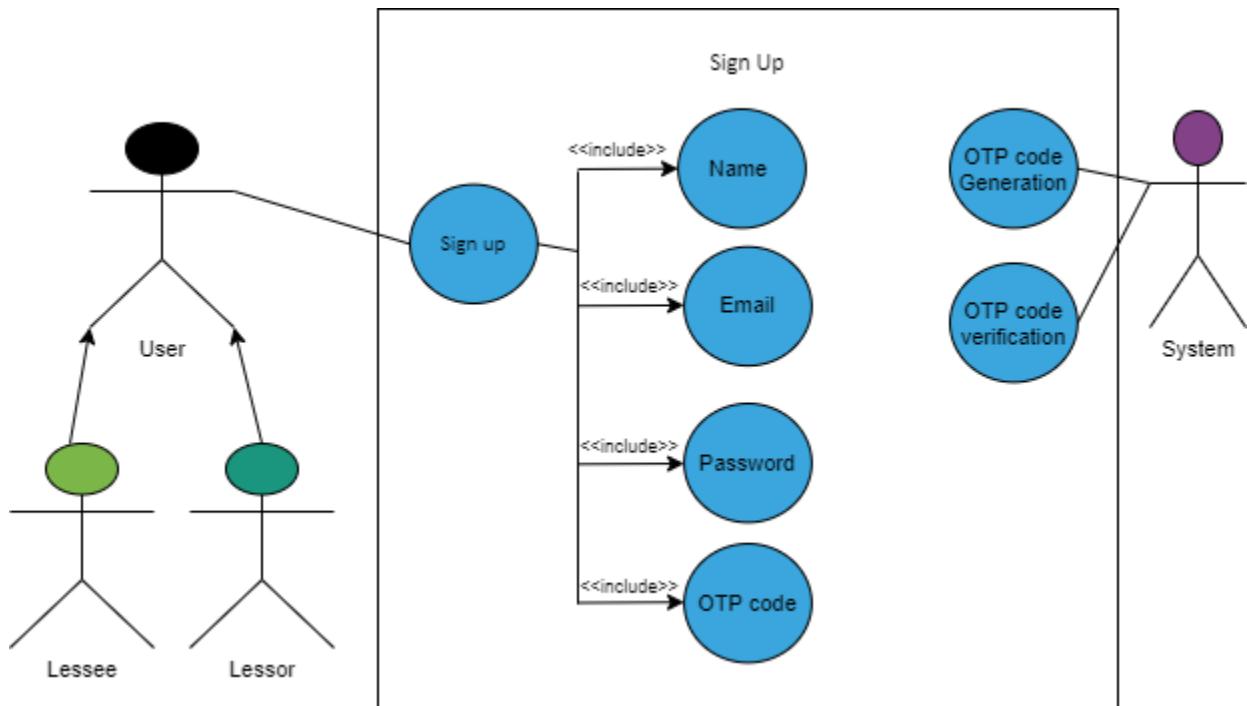


Figure 21: Use case diagram for Signup

2.5.22 Use Case Diagram: Login

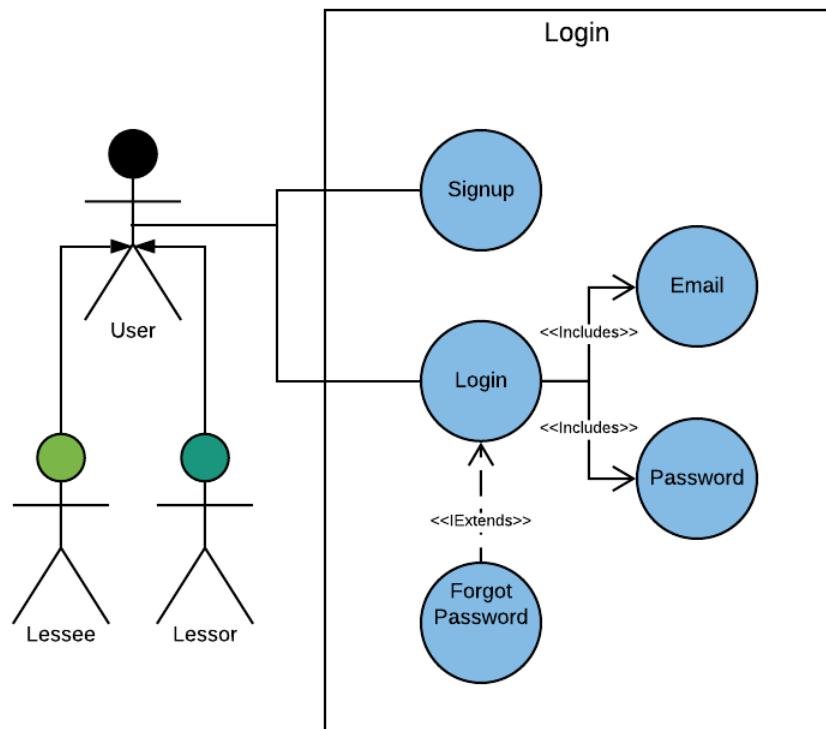


Figure 22: Use case diagram for Login

2.5.23 Use Case Diagram: Logout

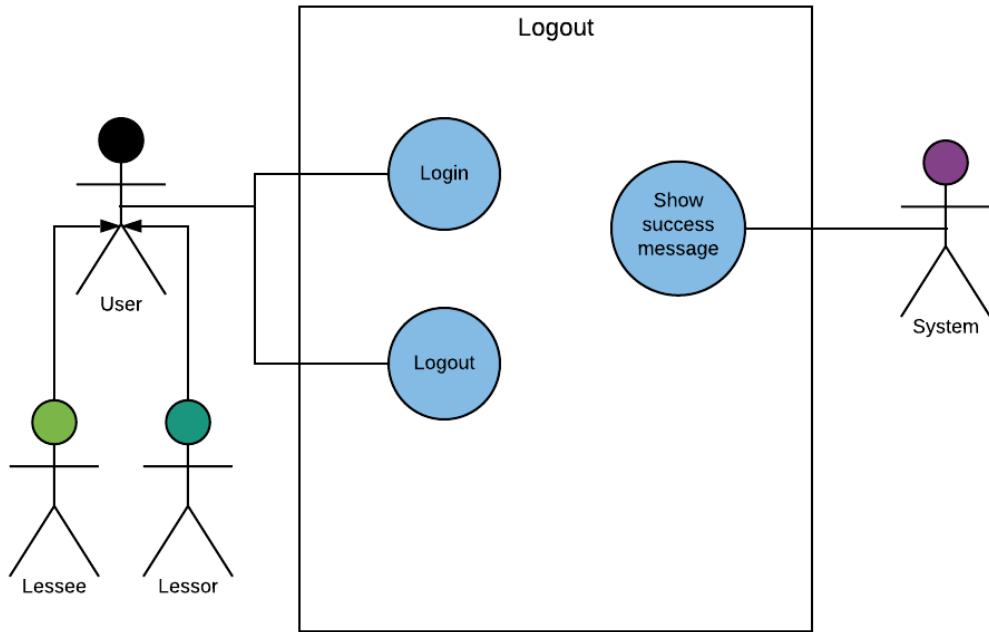


Figure 23: Use case diagram for Logout

2.5.24 Use Case Diagram: Forgot Password

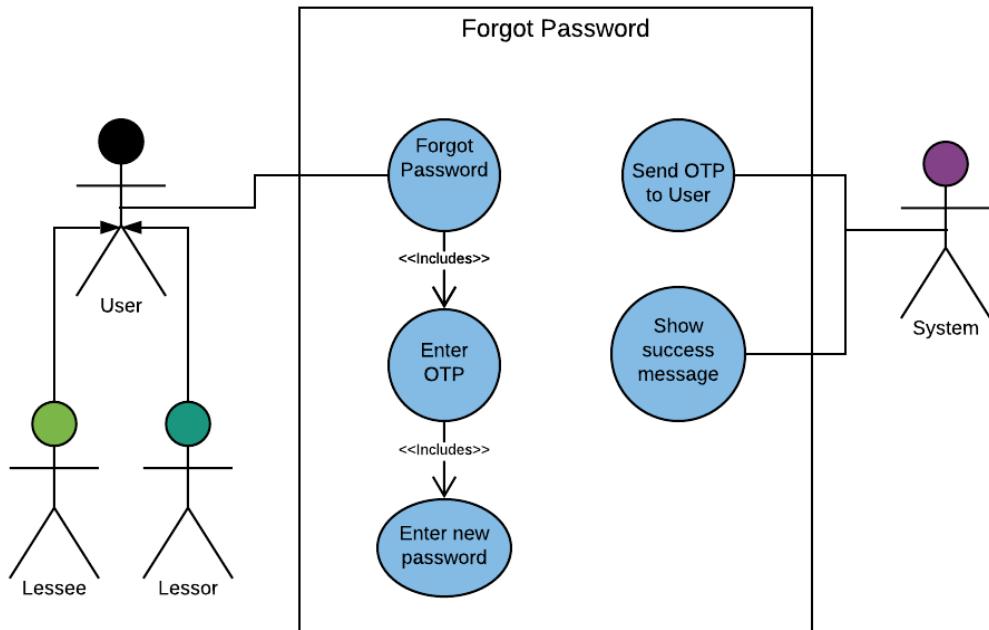


Figure 24: Use case diagram forForgot Password

2.5.25 Use Case Diagram: Cancel Order

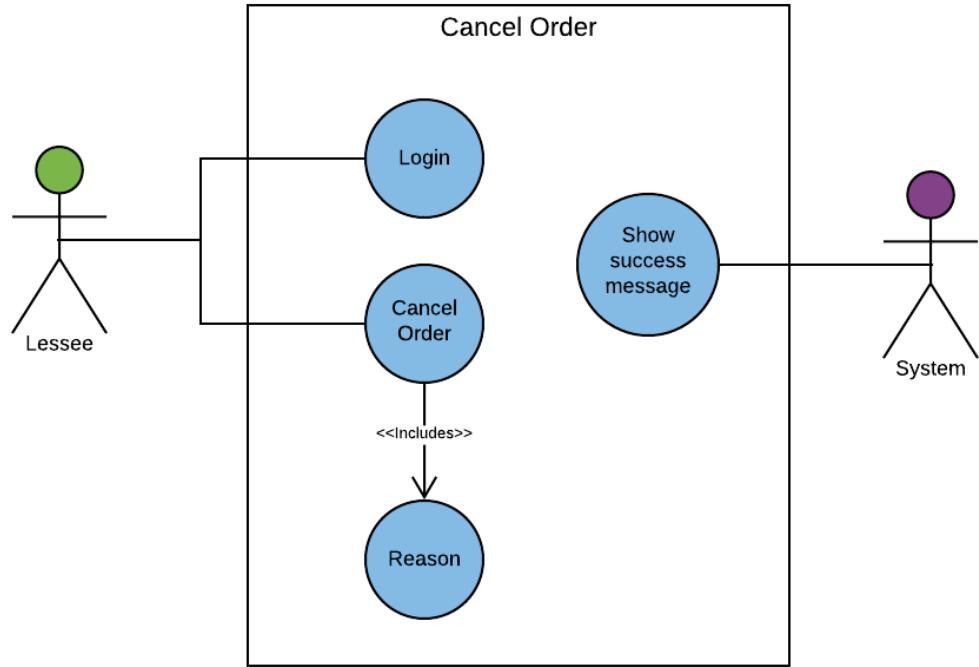


Figure 25: Use case diagram for Cancel Order

3 CHAPTER:3 System Design

3.1 Work Breakdown Structure:

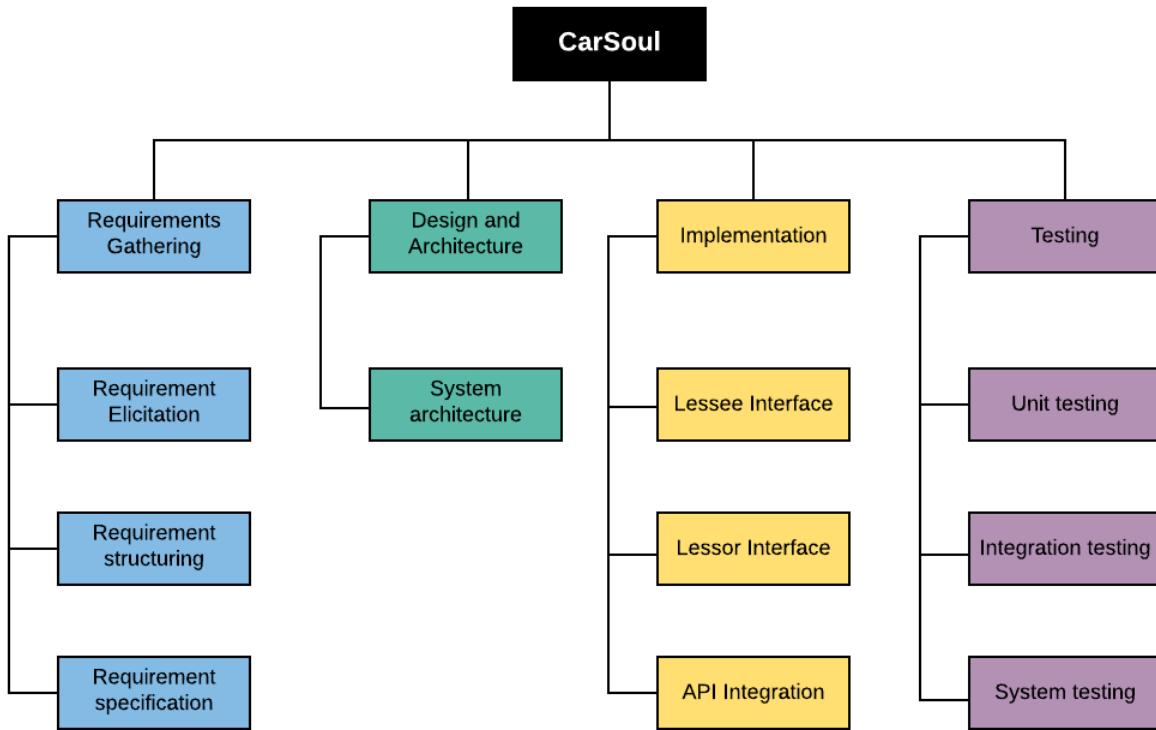


Figure 26: Work Breakdown Structure

3.2 Activity Diagrams:

3.2.1 Car Search

Car Search

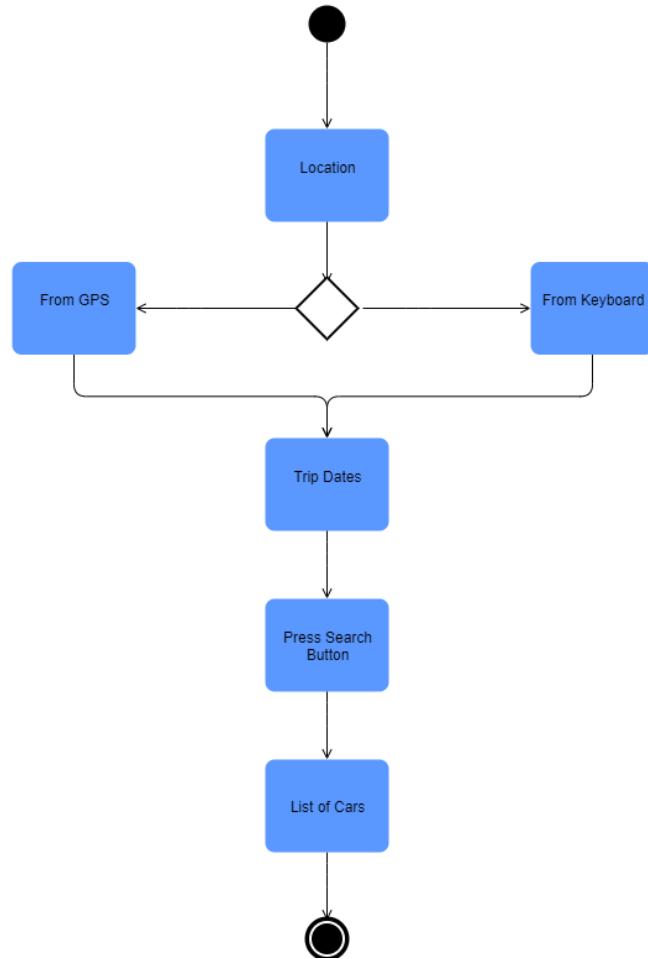


Figure 27: Activity diagram for Car Search

3.2.2 Advance Filters

Advance Filters

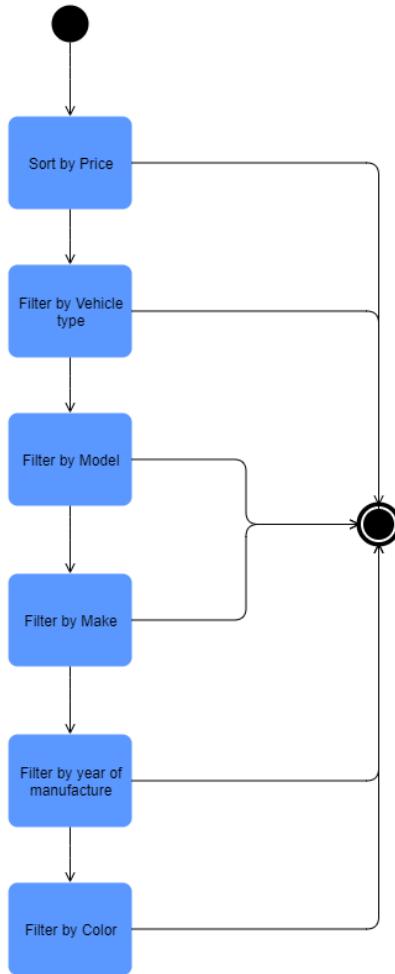


Figure 28: Activity diagram for Advance Filter

3.2.3 View Car Details for Lessee

View car details for Lessee

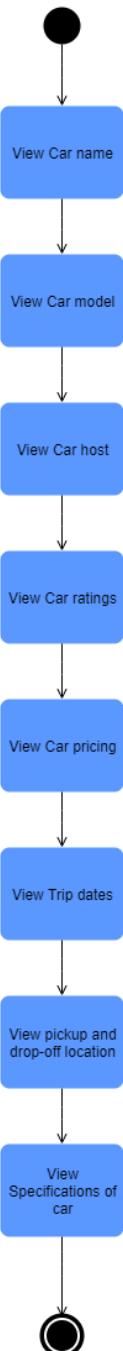


Figure 29: Activity diagram for View Car details

3.2.4 Report Car Listing

Report car listing

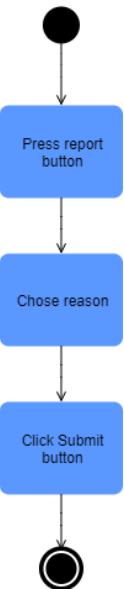


Figure 30: Activity diagram for Report Car Listing

3.2.5 Checkout

Checkout

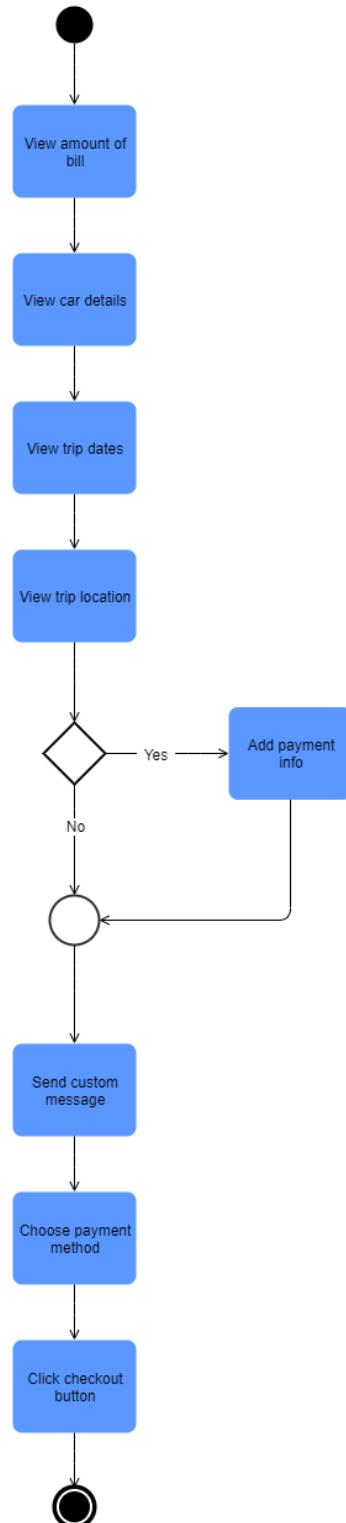


Figure 31: Activity diagram for Checkout

3.2.6 View Activity

View Activity

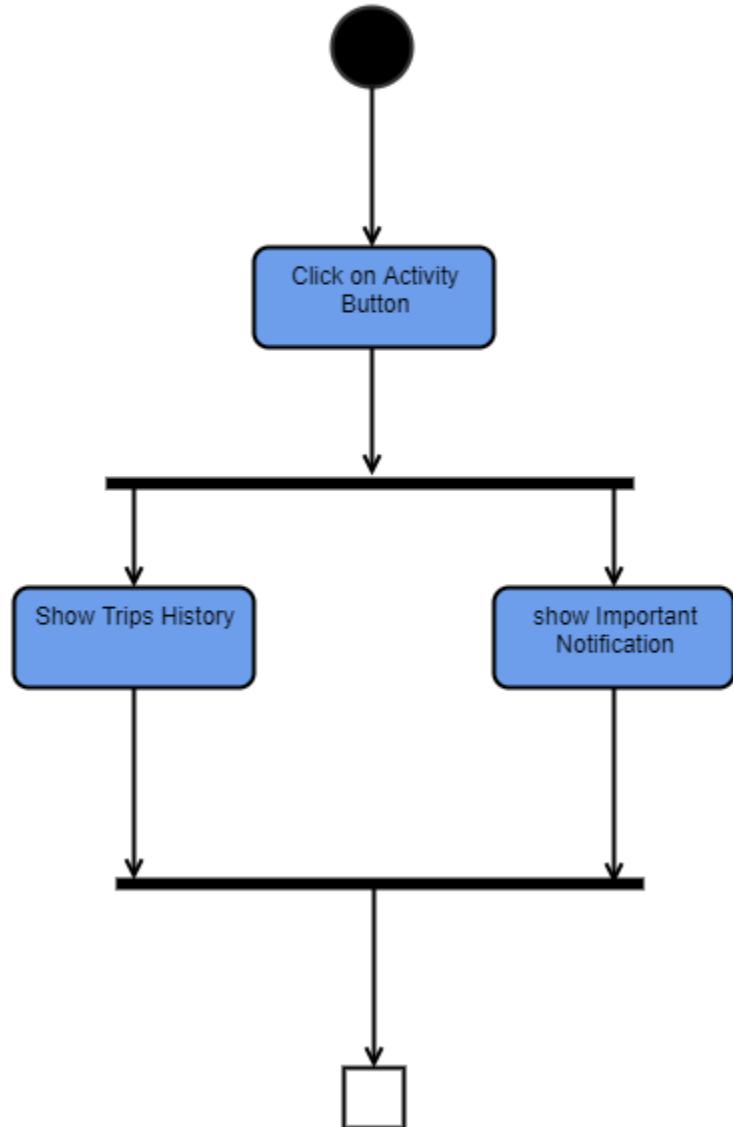


Figure 32: Activity diagram for View Activity

3.2.7 Add to Favourite

Add to Favorite

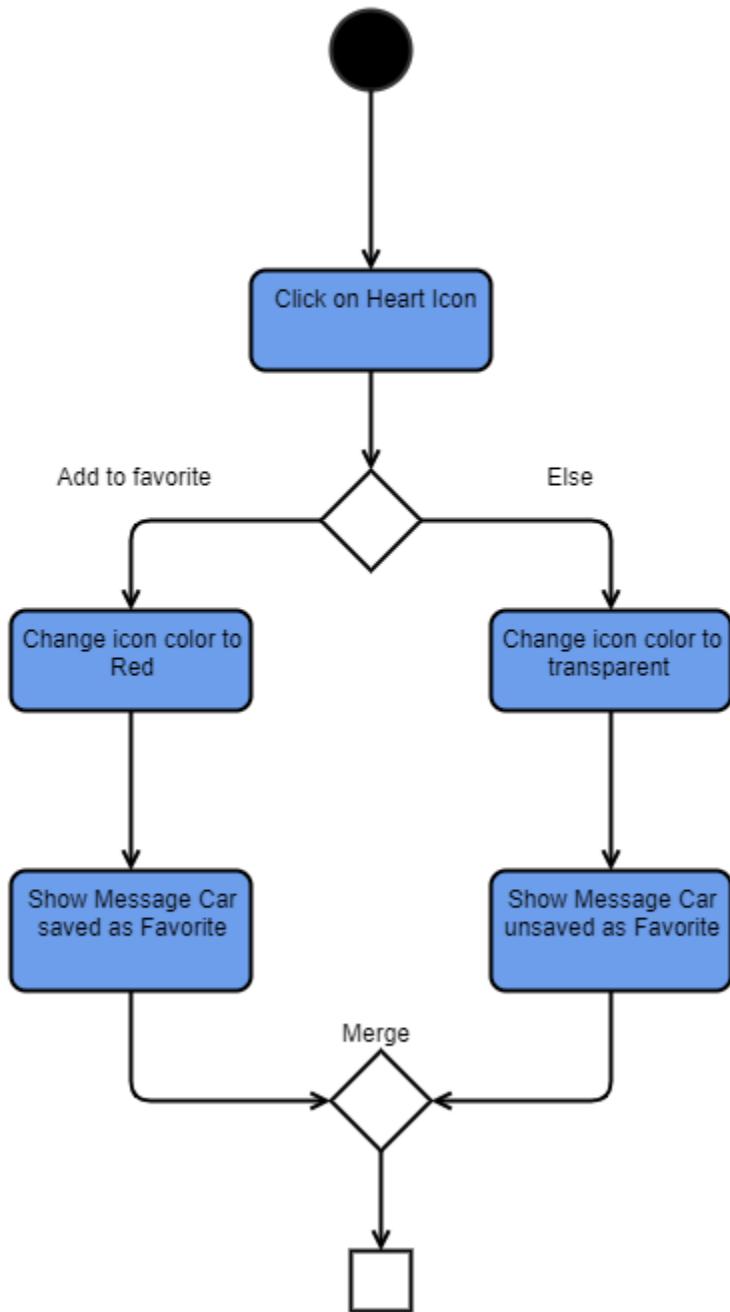


Figure 33: Activity diagram for Add to Favorite

3.2.8 Feedback

Feedback

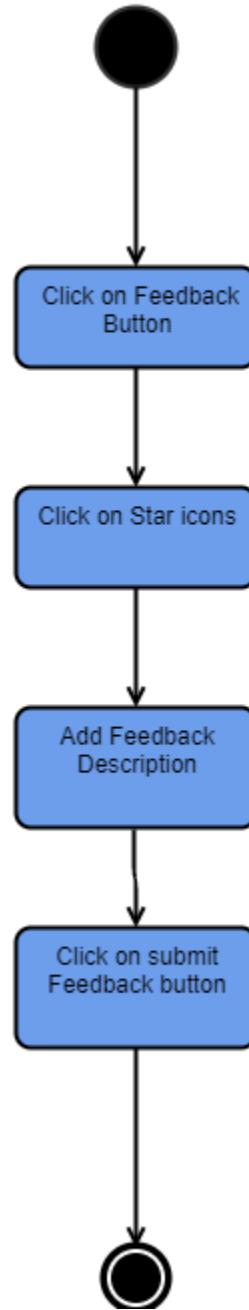


Figure 34: Activity diagram for Feedback

3.2.9 List Your Car

List Your Car

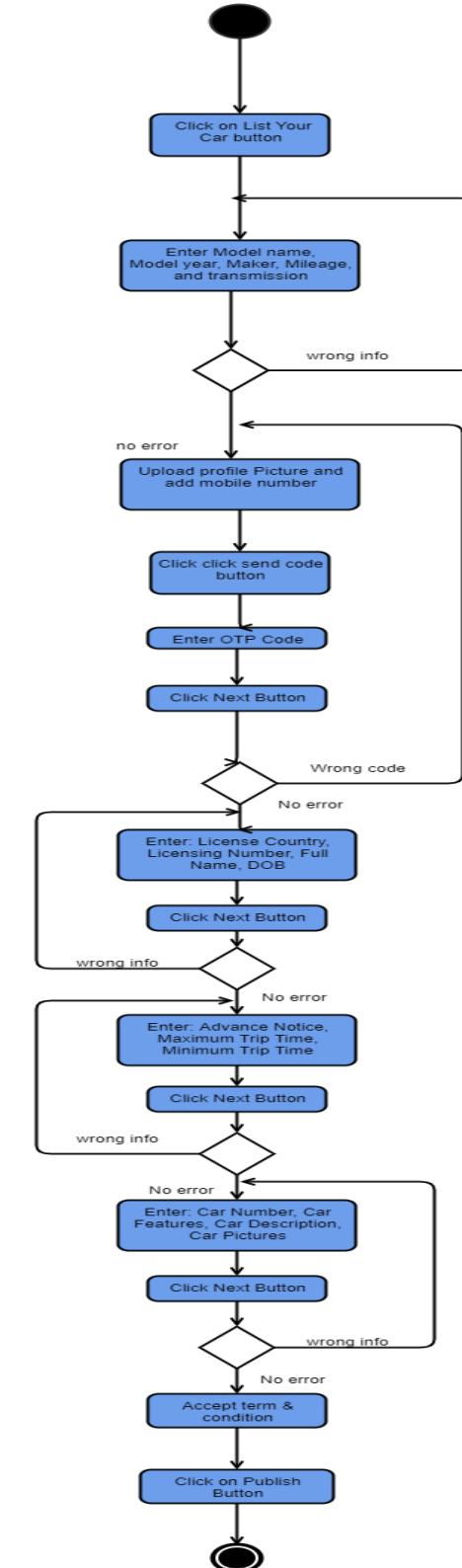


Figure 35: Activity diagram for List Your Car

3.2.10 Payment Withdraw Request

Payment Withdraw Request

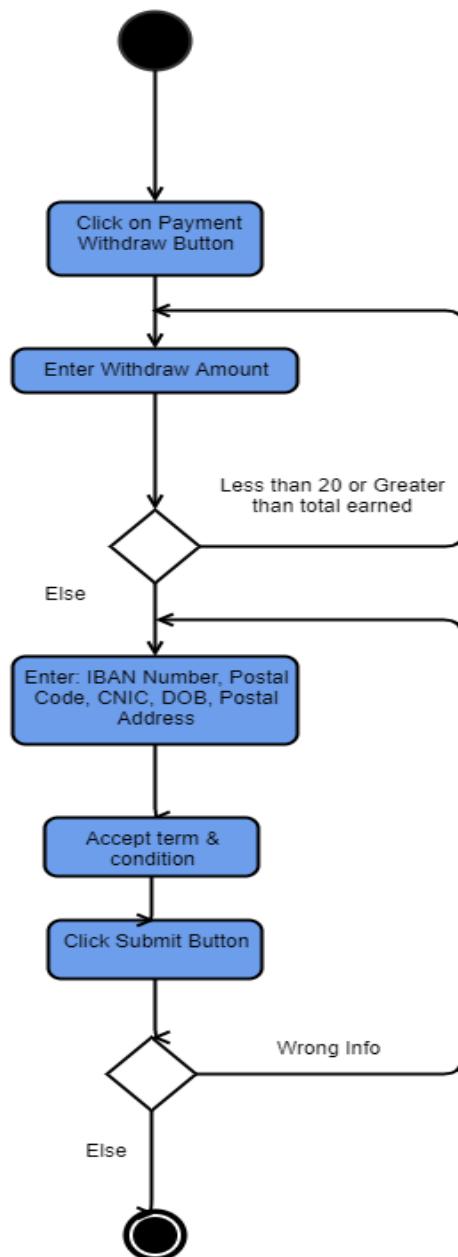


Figure 36: Activity diagram for Payment Withdraw Request

3.2.11 View Posted Cars

View Posted Cars

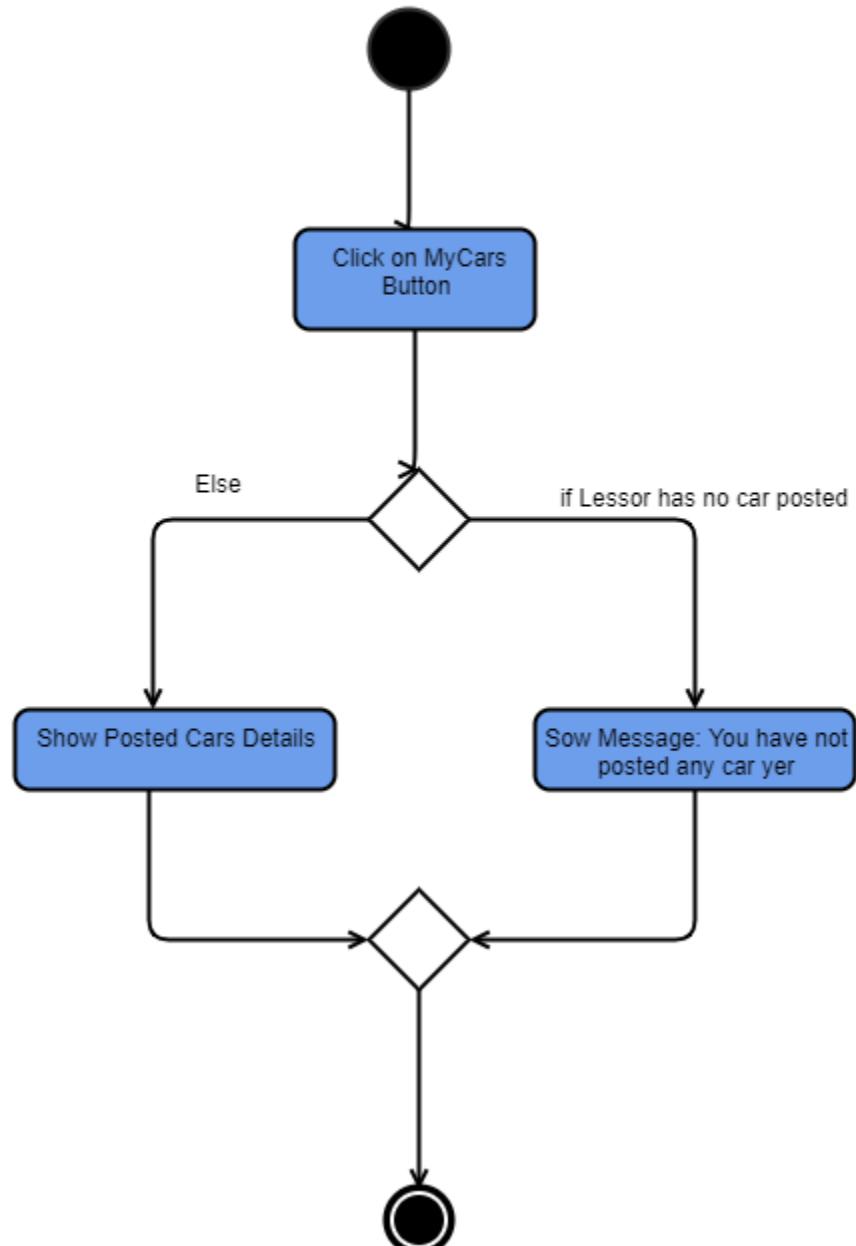


Figure 37: Activity diagram for View Posted Cars

3.2.12 Track Your Car

Track Your Car

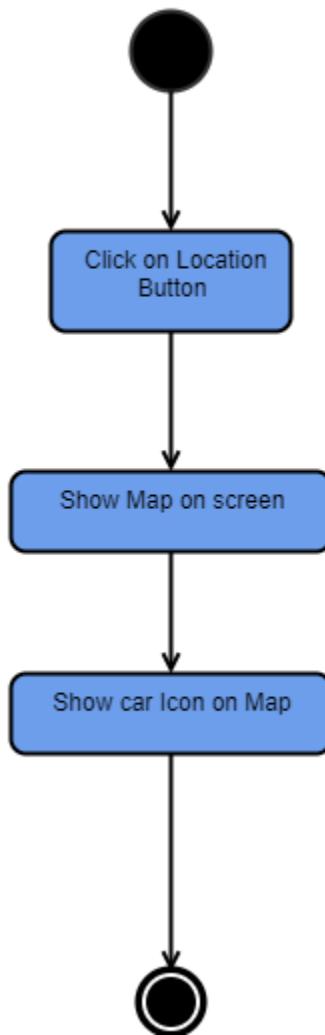


Figure 38: Activity diagram for Track Your Car

3.2.13 Deactivate Car Posting

Deactivate car Posting

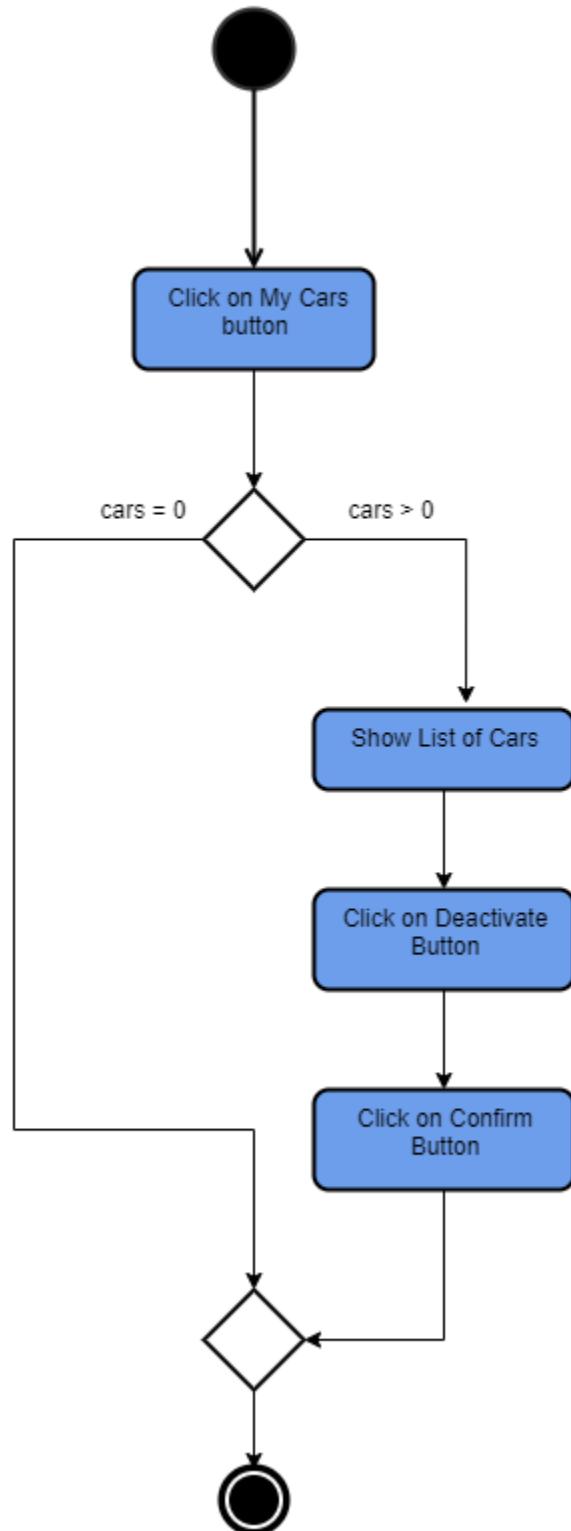


Figure 39: Activity diagram for Deactivate Car Posting

3.2.14 View Trip History

View Trip History

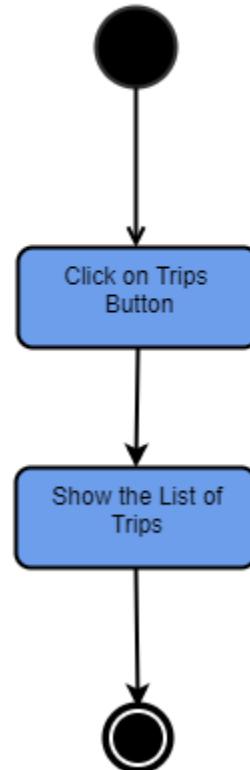


Figure 40: Activity diagram for View Trip History

3.2.15 View Active Trips

View Active Trips



Figure 41: Activity diagram for View Active Trips

3.2.16 Messages

Messages

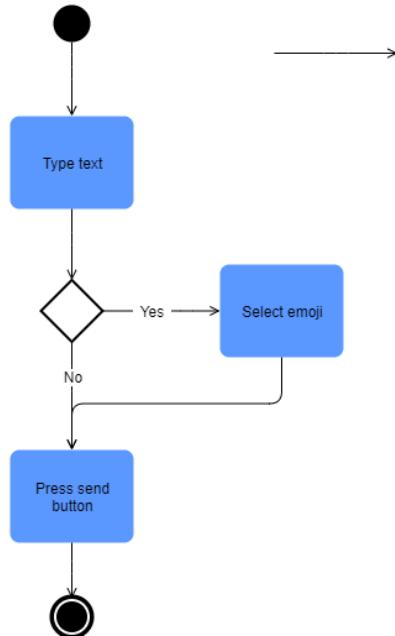


Figure 42: Activity diagram for Messages

3.2.17 View Profile

View Profile

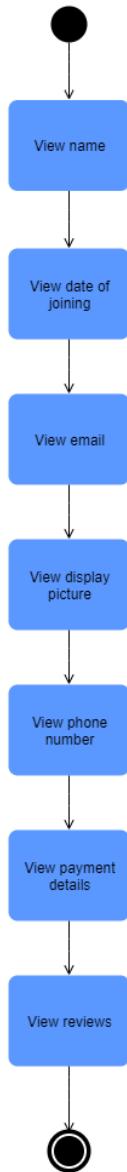


Figure 43: Activity diagram for View Profile

3.2.18 Complete Profile

Complete Profile

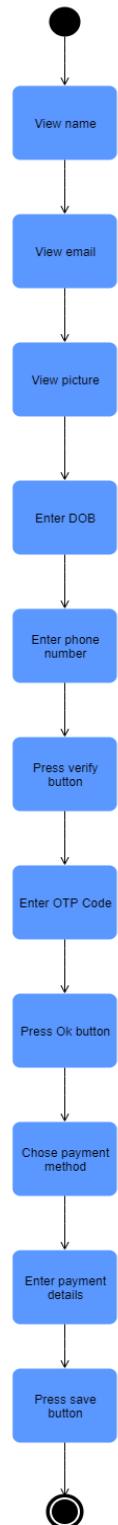


Figure 44: Activity diagram for Complete Profile

3.2.19 Edit Profile

Edit Profile

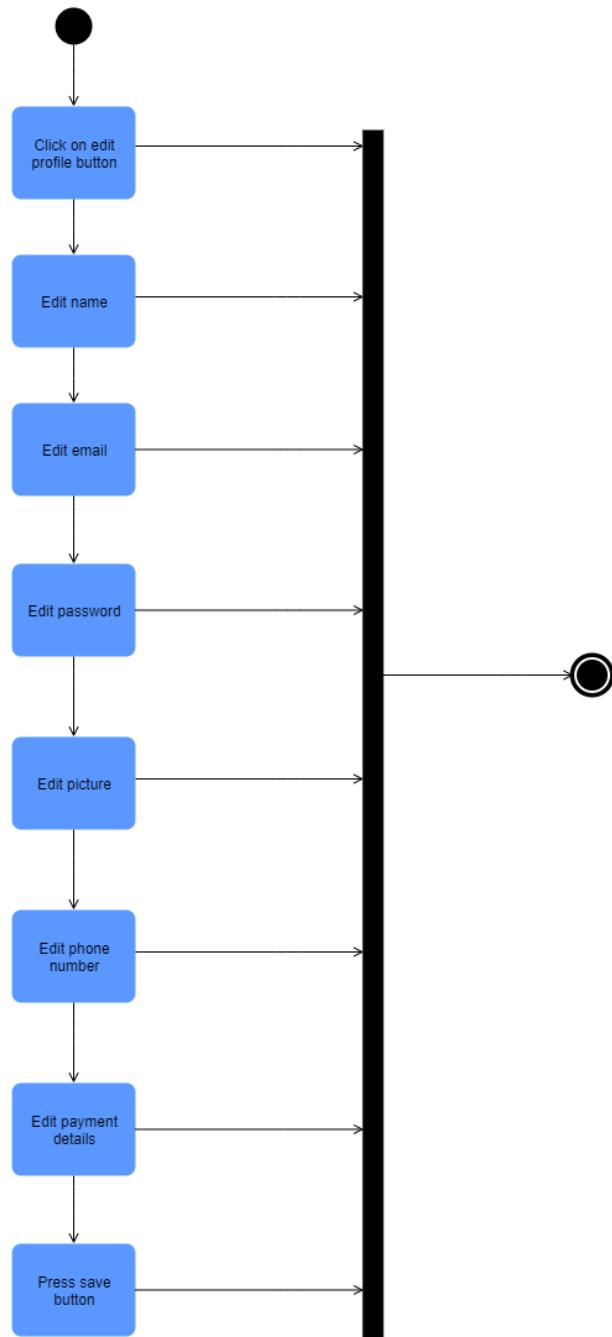


Figure 45: Activity diagram for Edit Profile

3.2.20 View Account Details

View Account Details

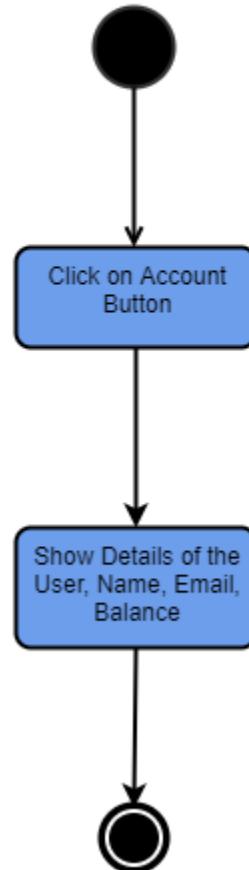


Figure 46: Activity diagram for View Account Details

3.2.21 Signup

Sign Up

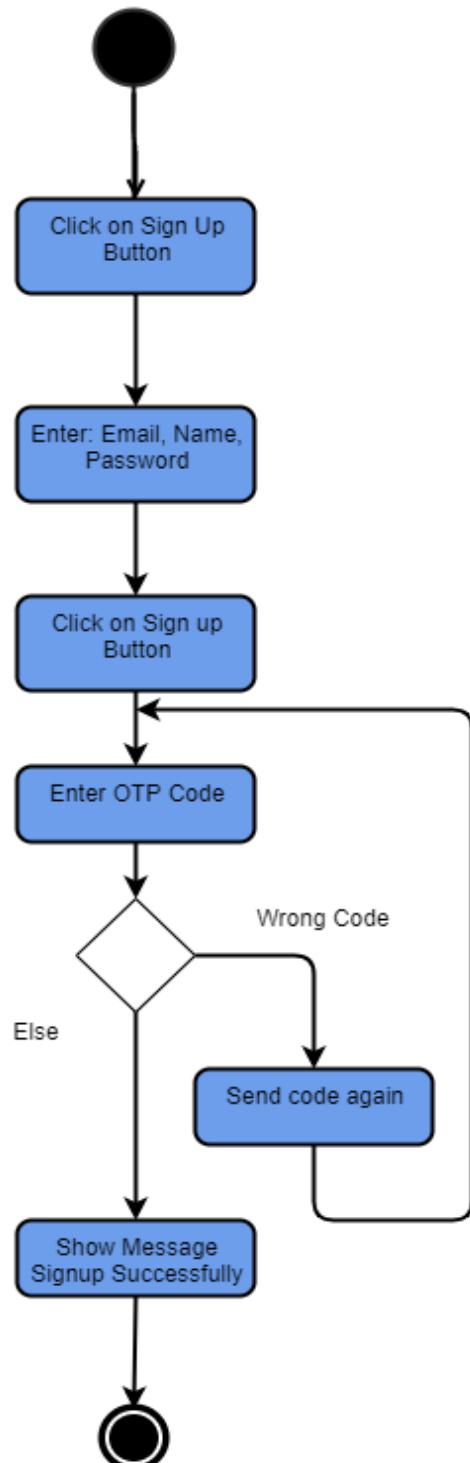


Figure 47: Activity diagram for Signup

3.2.22 Login

Login

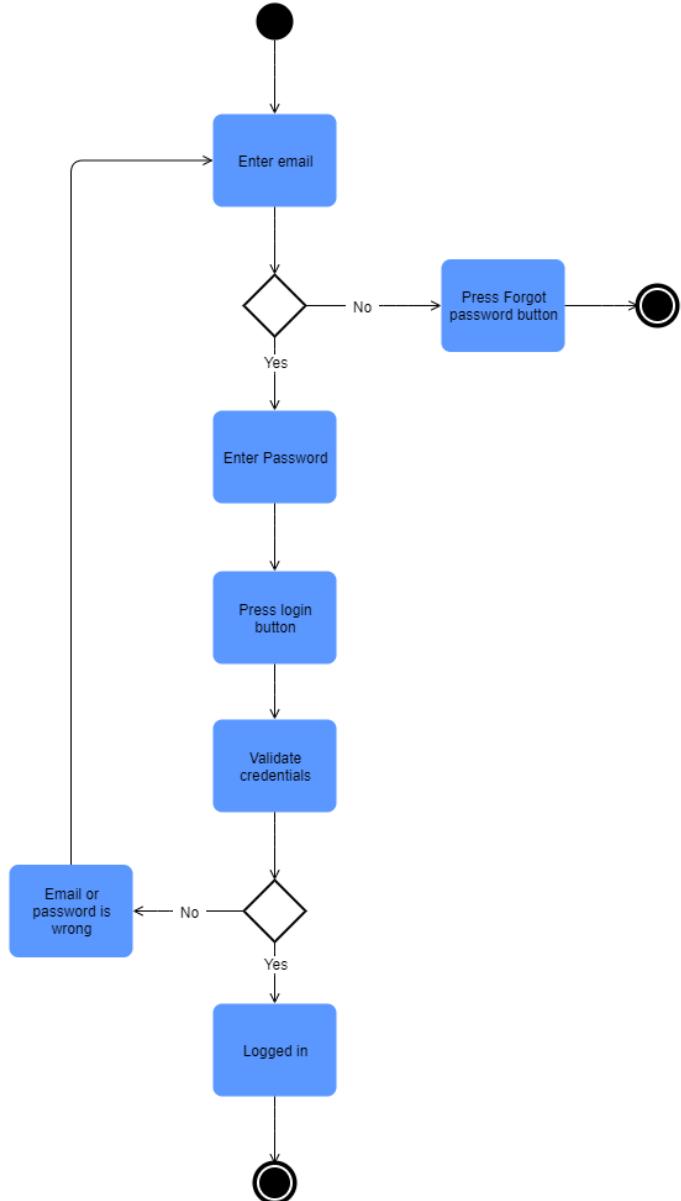


Figure 48: Activity diagram for Login

3.2.23 Logout

Logout

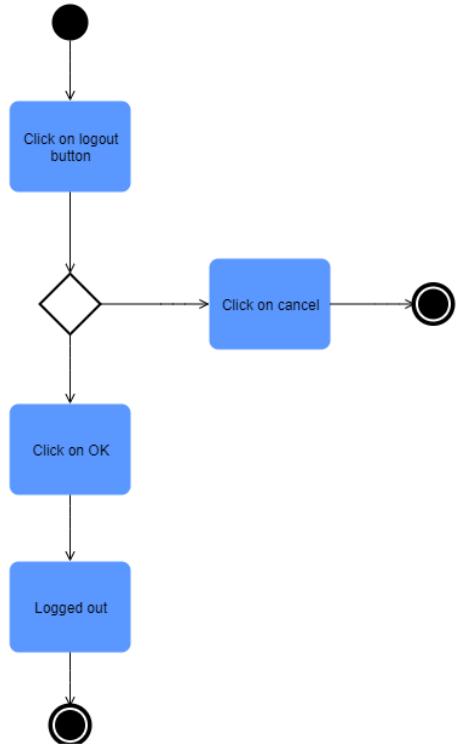


Figure 49: Activity diagram for Logout

3.2.24 Forgot Password

Forgot Password

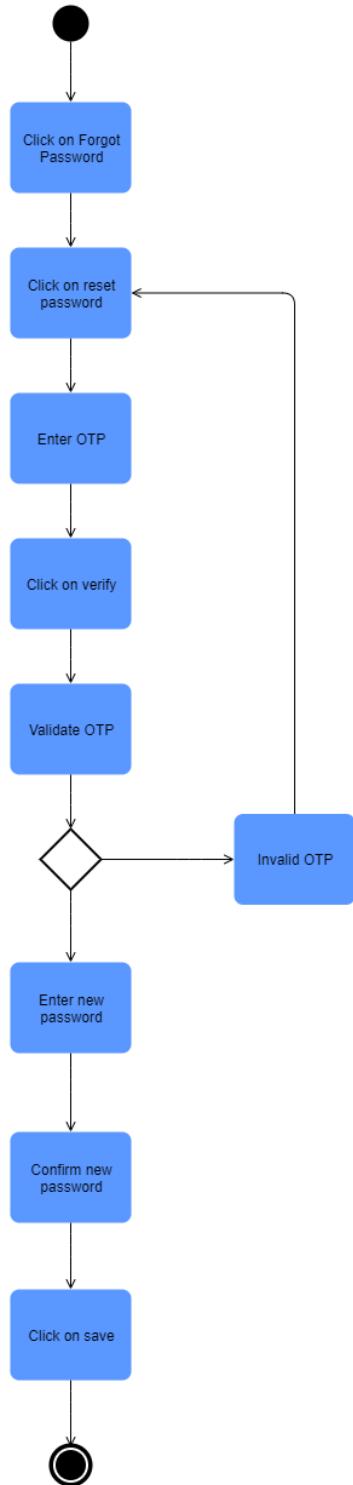


Figure 50: Activity diagram for Forgot password

3.2.25 Cancel Order

Cancel Order



Figure 51: Activity diagram for Cancel Order

3.3 Sequence Diagrams:

3.3.1 Car Search:

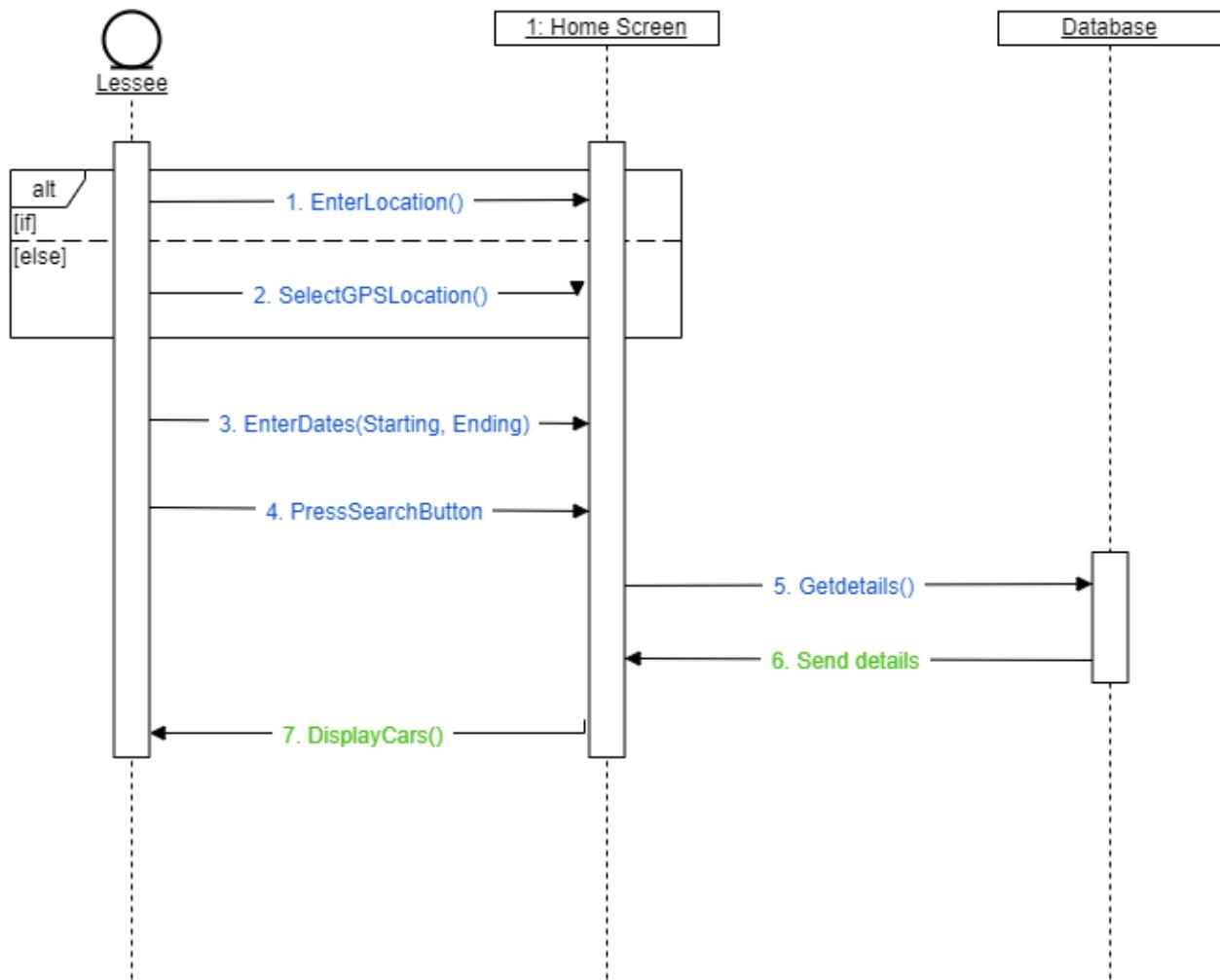


Figure 52: Sequence diagram for Car Search

3.3.2 Advance Filter:

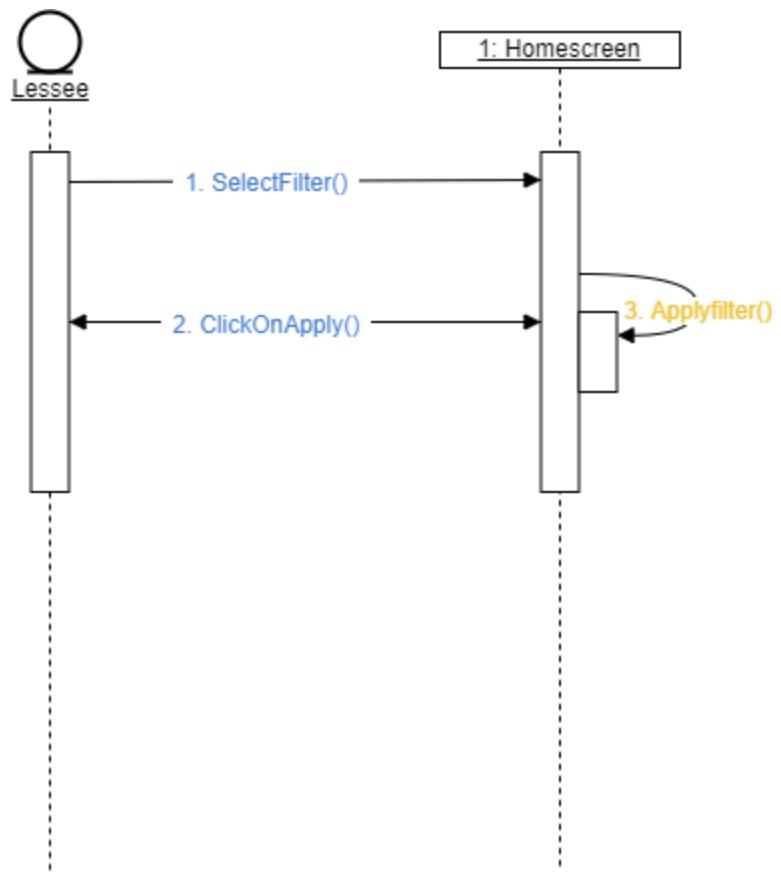


Figure 53: Sequence diagram for Advance Filters

3.3.3 View Car Details:

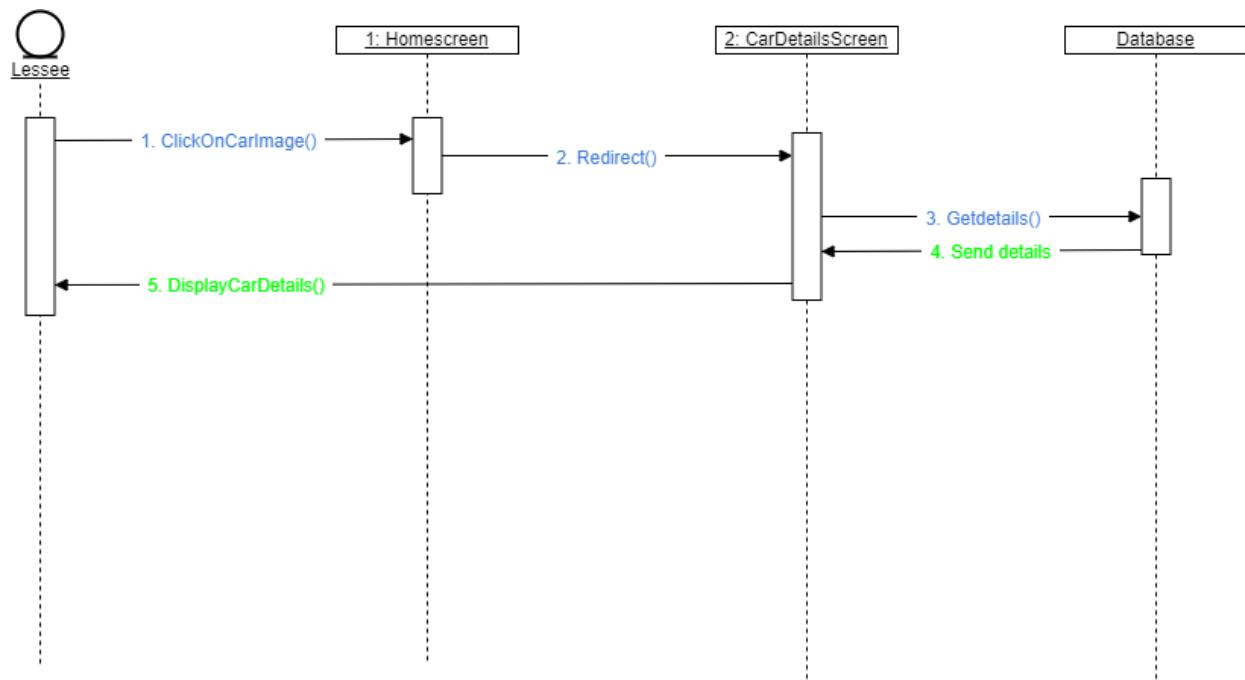


Figure 54: Sequence diagram for View Car Details

3.3.4 Report Car Listing:

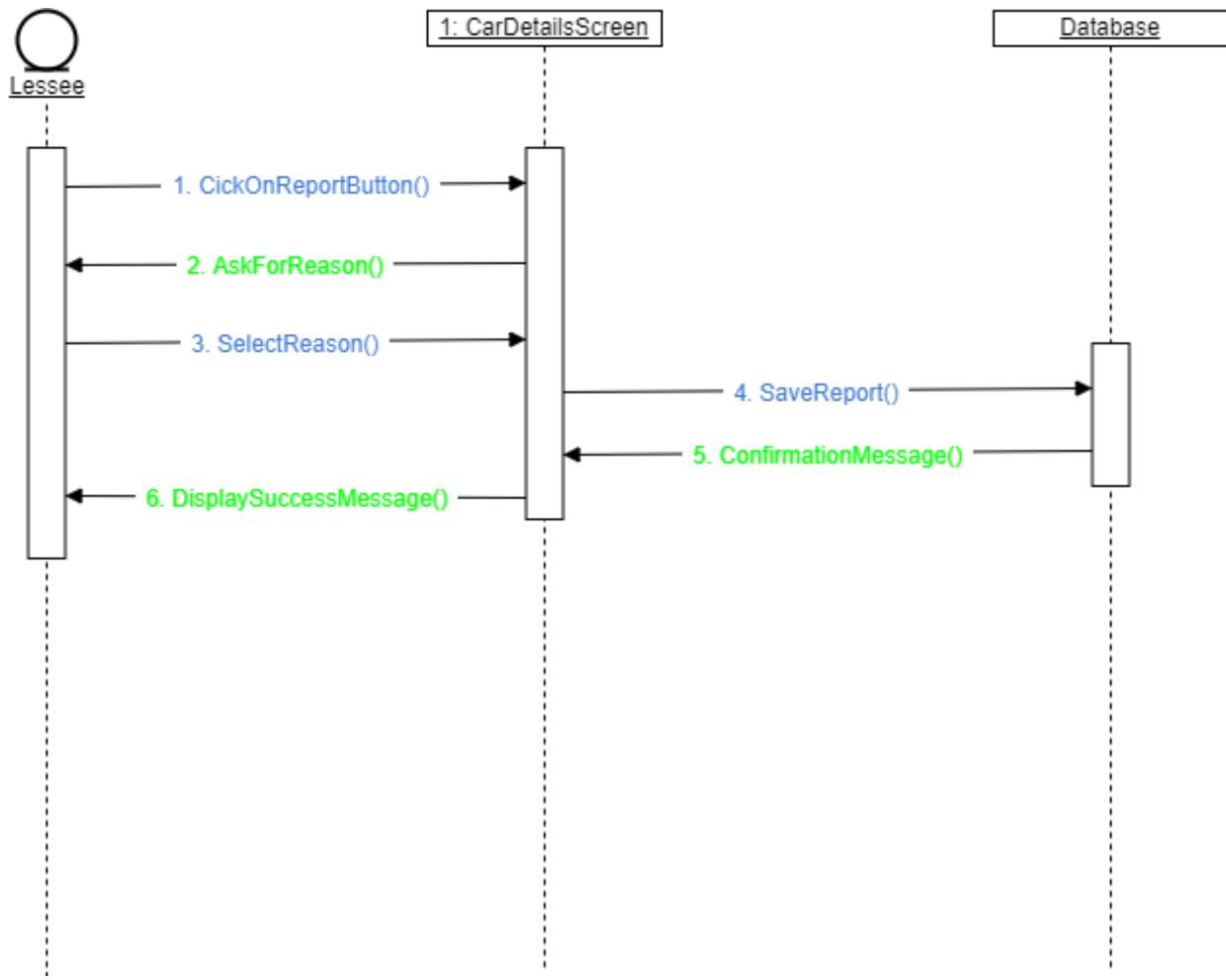


Figure 55: Sequence diagram for Report Car Listing

3.3.5 Checkout:

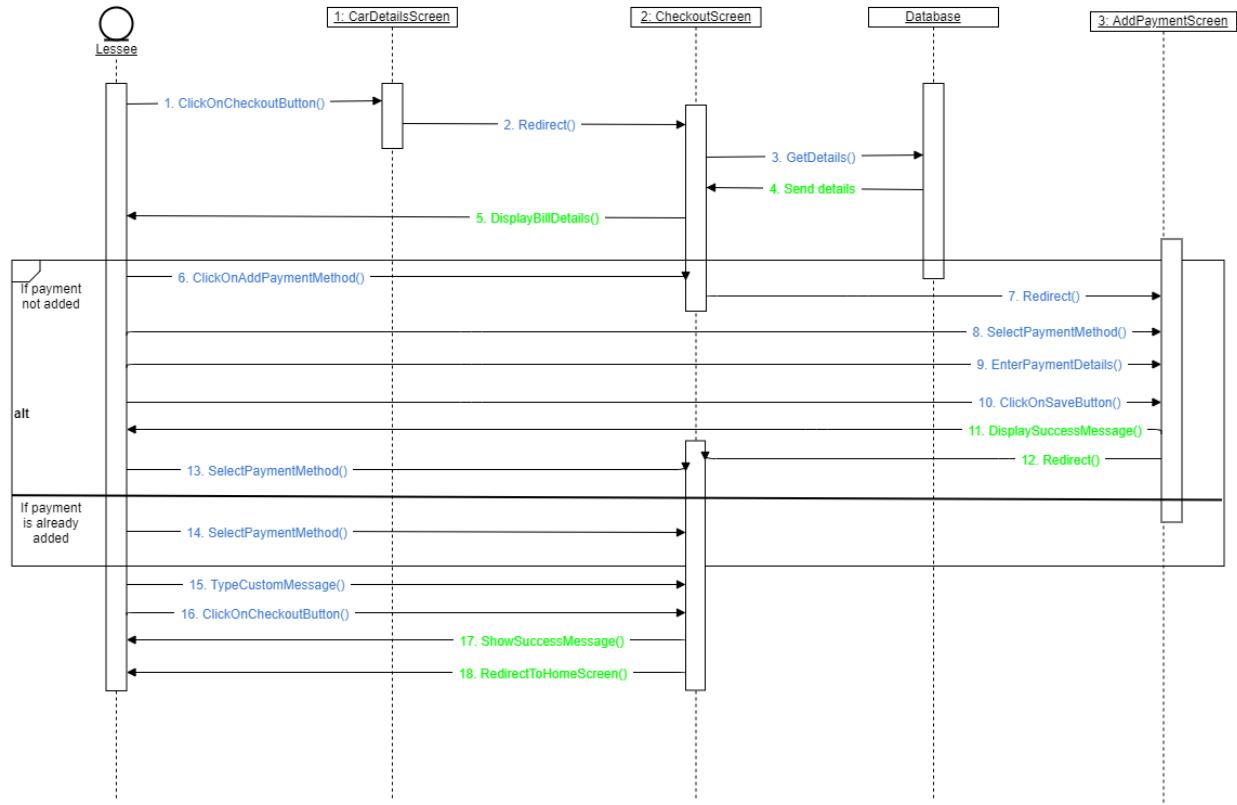


Figure 56: Sequence diagram for Checkout

3.3.6 View Activity:

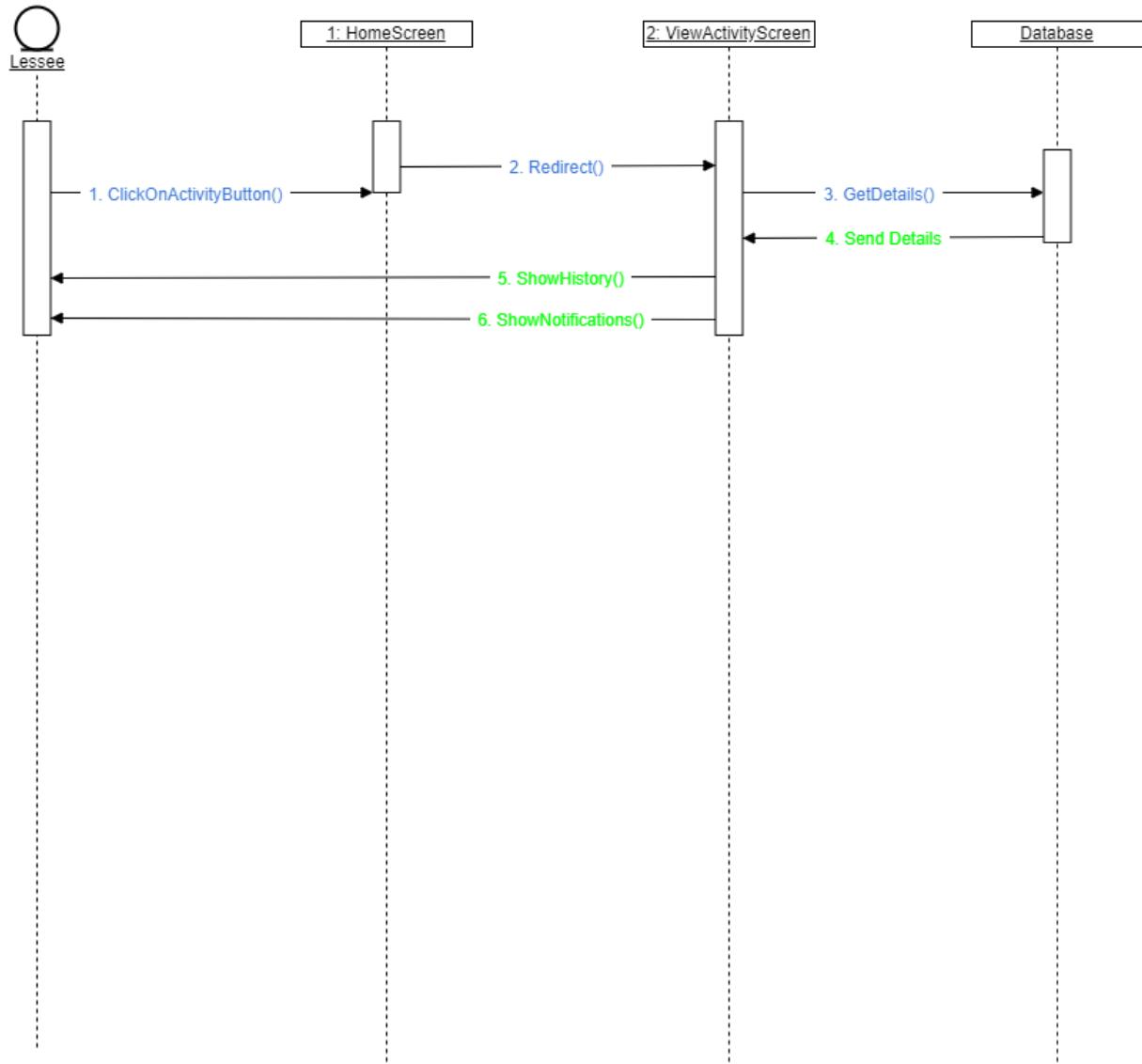


Figure 57: Sequence diagram for View Activity

3.3.7 Add to Favourite:

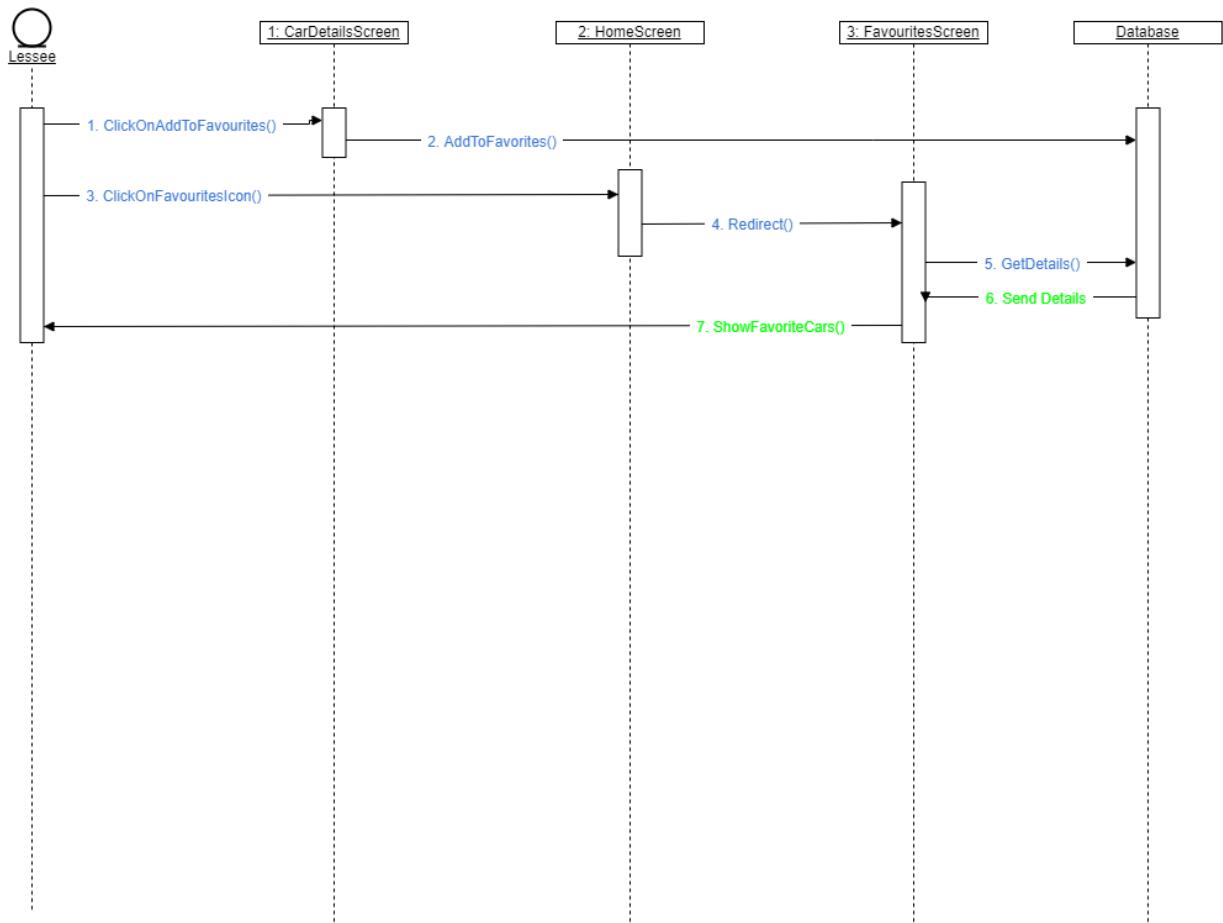


Figure 58: Sequence diagram for Add to Favorite

3.3.8 Feedback:

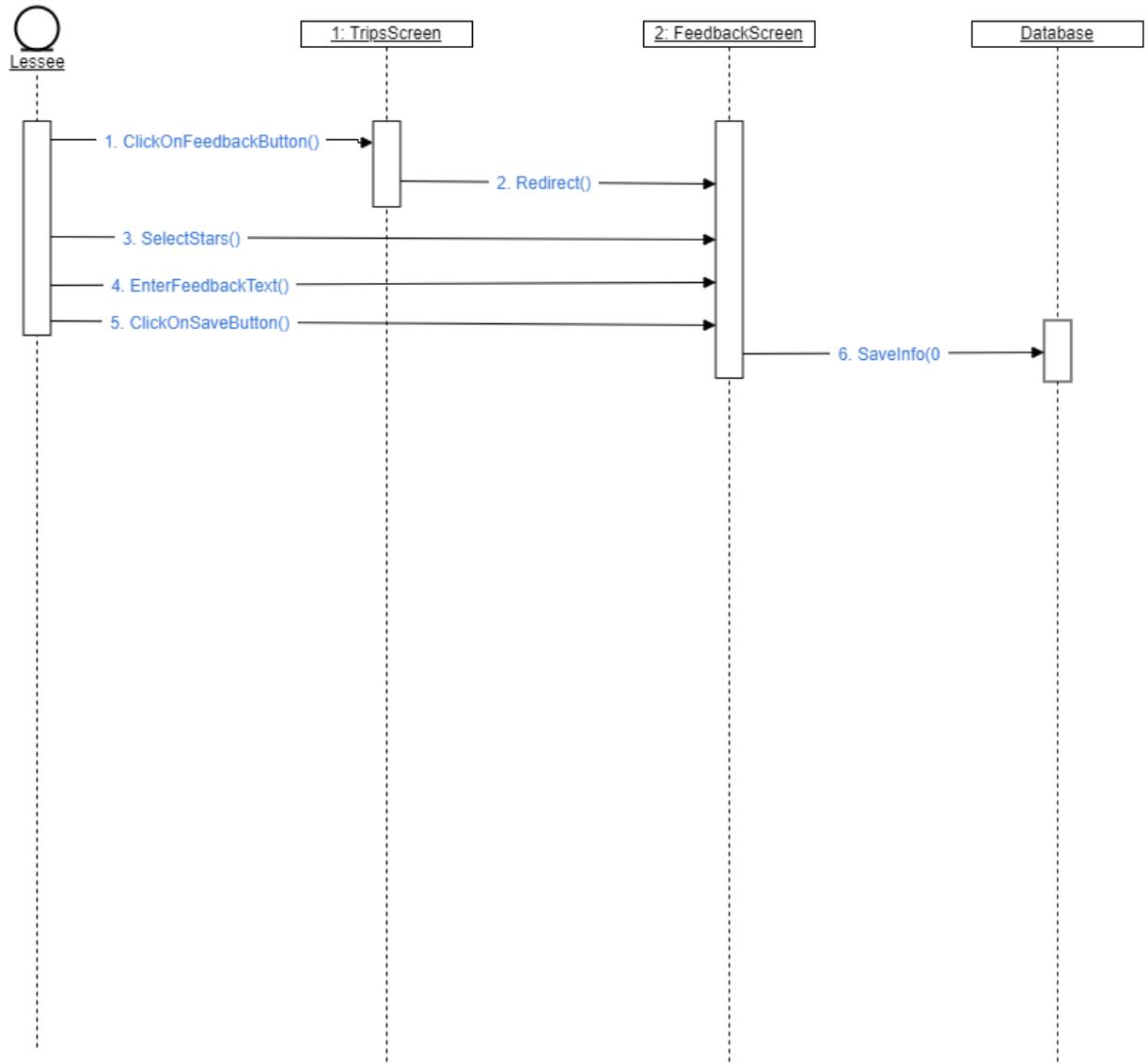


Figure 59: Sequence diagram for Feedback

3.3.9 List Your Car:

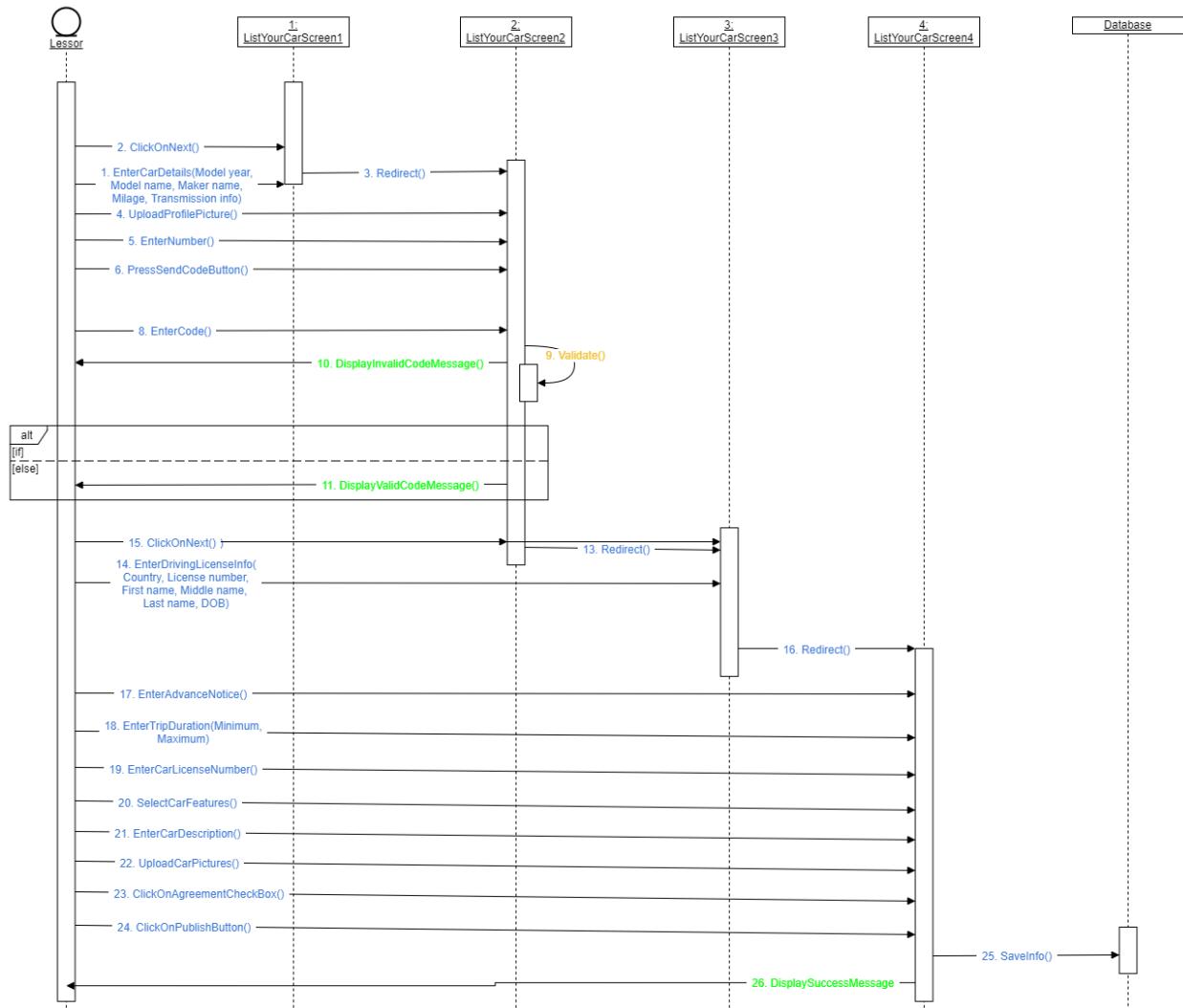


Figure 60: Sequence diagram for List Your Car

3.3.10 Payment Withdraw Request:

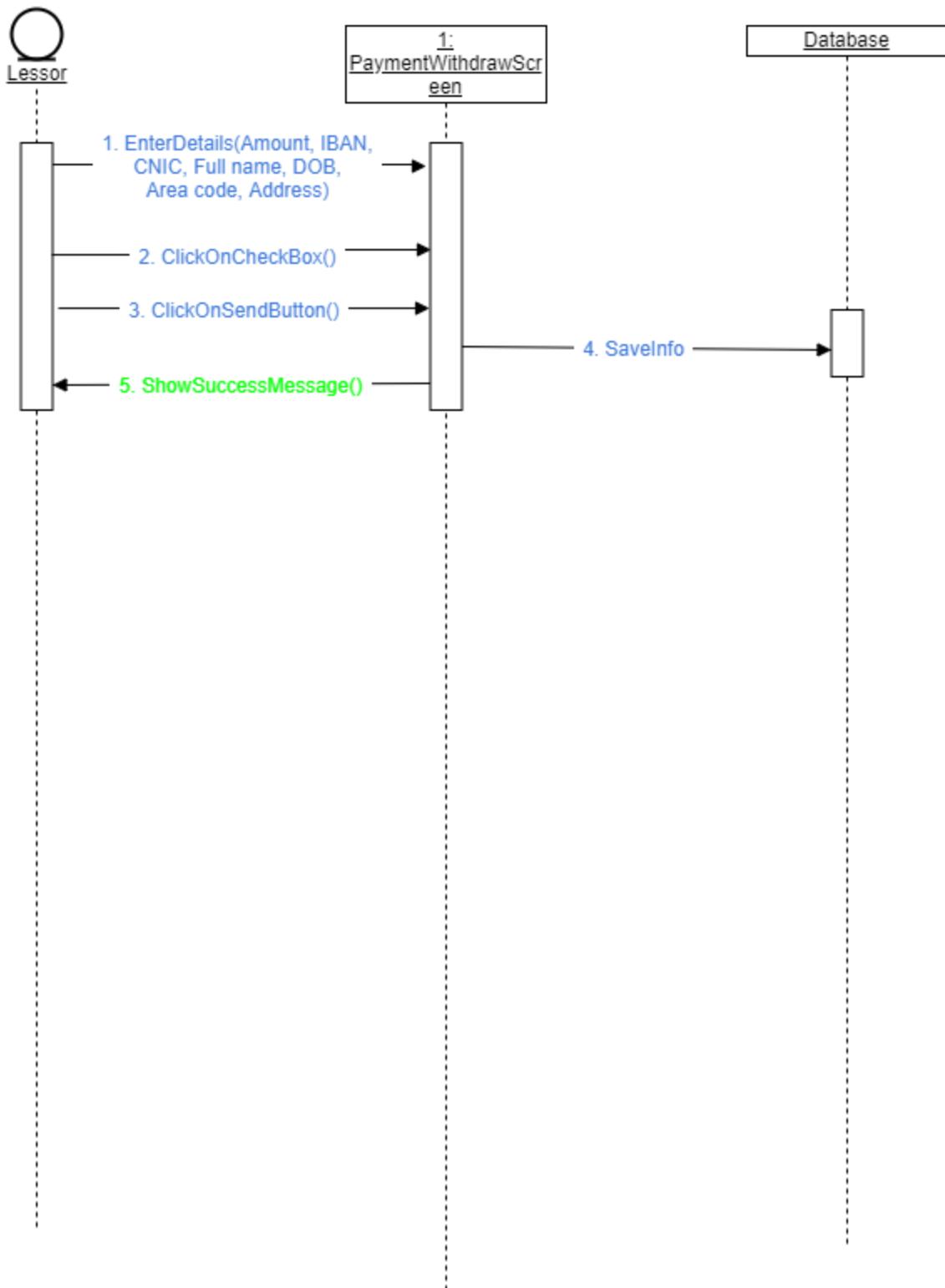


Figure 61: Sequence diagram for Payment withdraw request

3.3.11 View Posted Car:

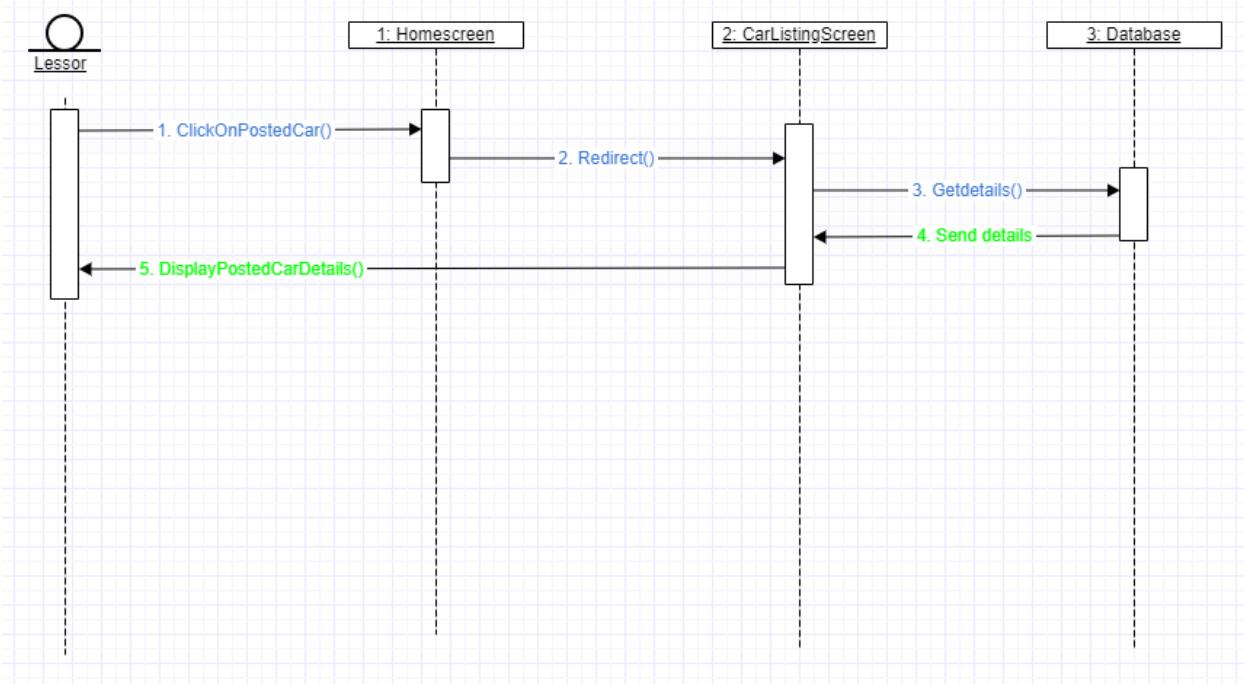


Figure 62: Sequence diagram for View Posted Cars

3.3.12 Track Your Car:

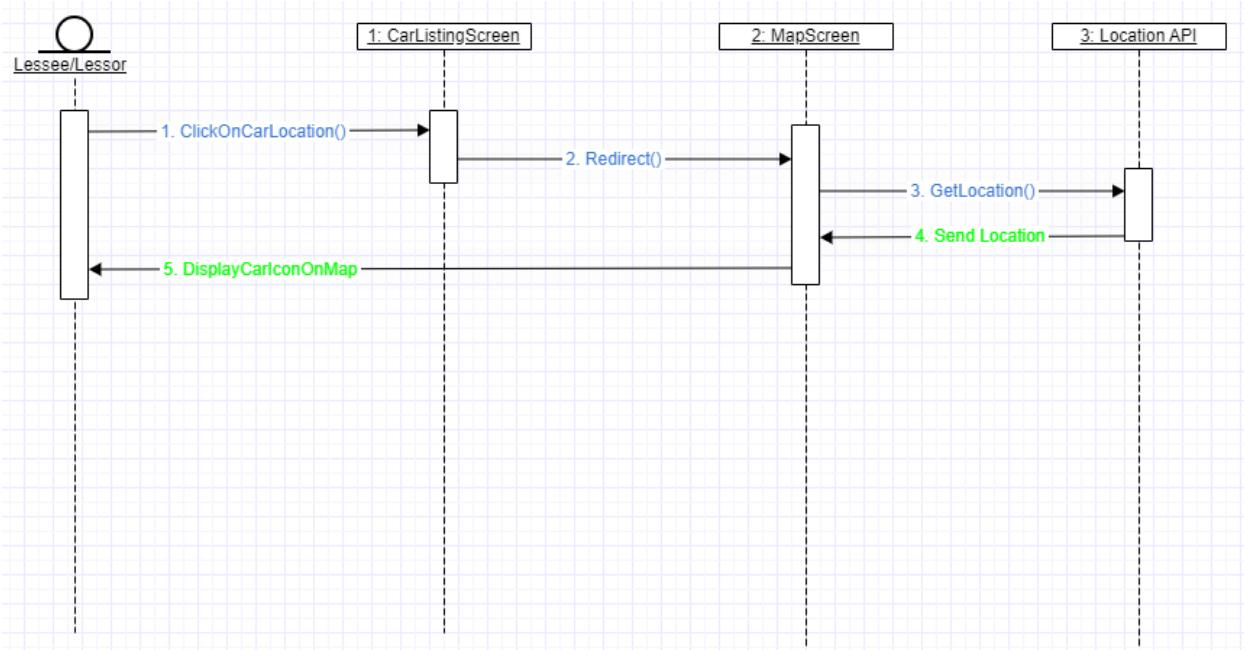


Figure 63: Sequence diagram for Track Your Car

3.3.13 Deactivate Car Posting:

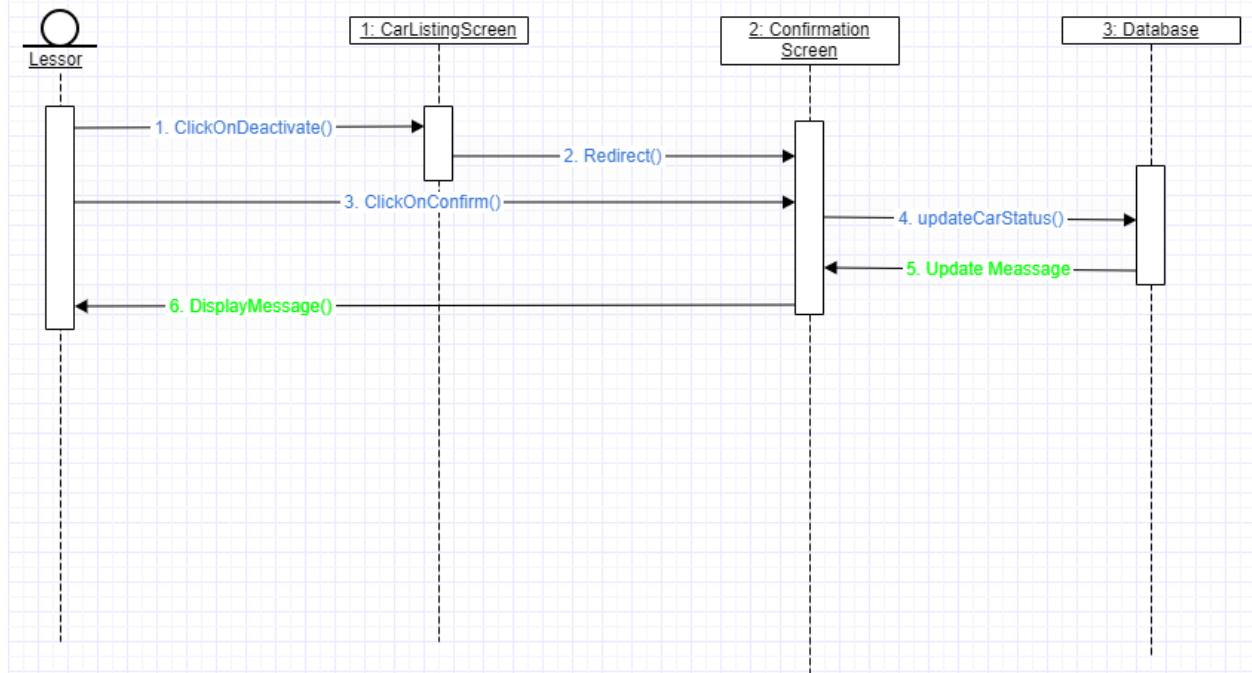


Figure 64: Sequence diagram for Deactivate car posting

3.3.14 View Trip History:

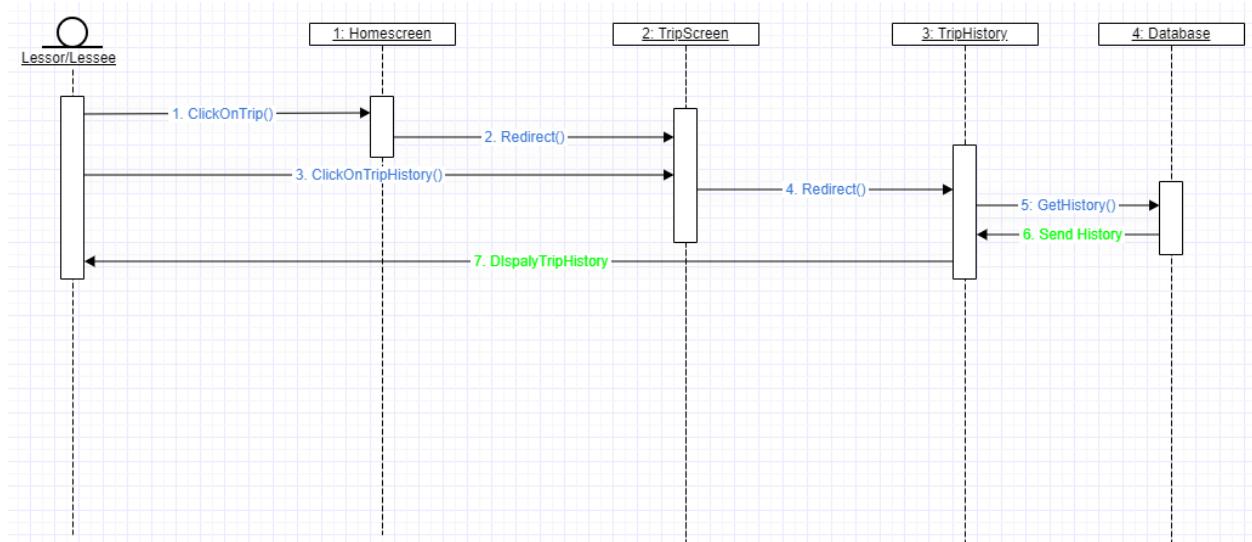


Figure 65: Sequence diagram for View Trip History

3.3.15 View Active Trips:

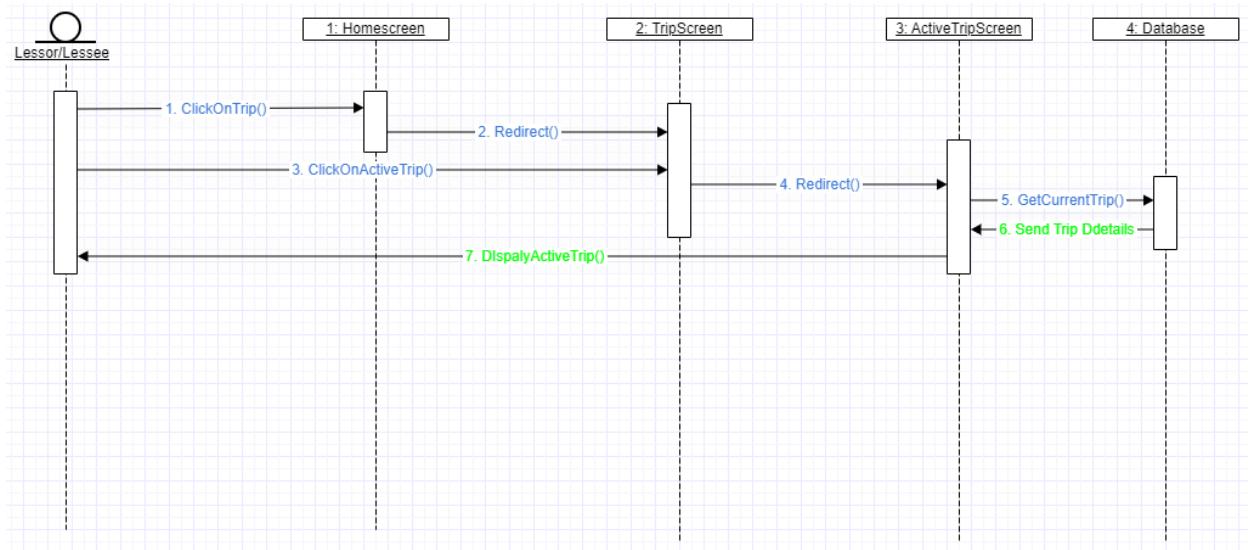


Figure 66: Sequence diagram for View Active Trips

3.3.16 Messages:

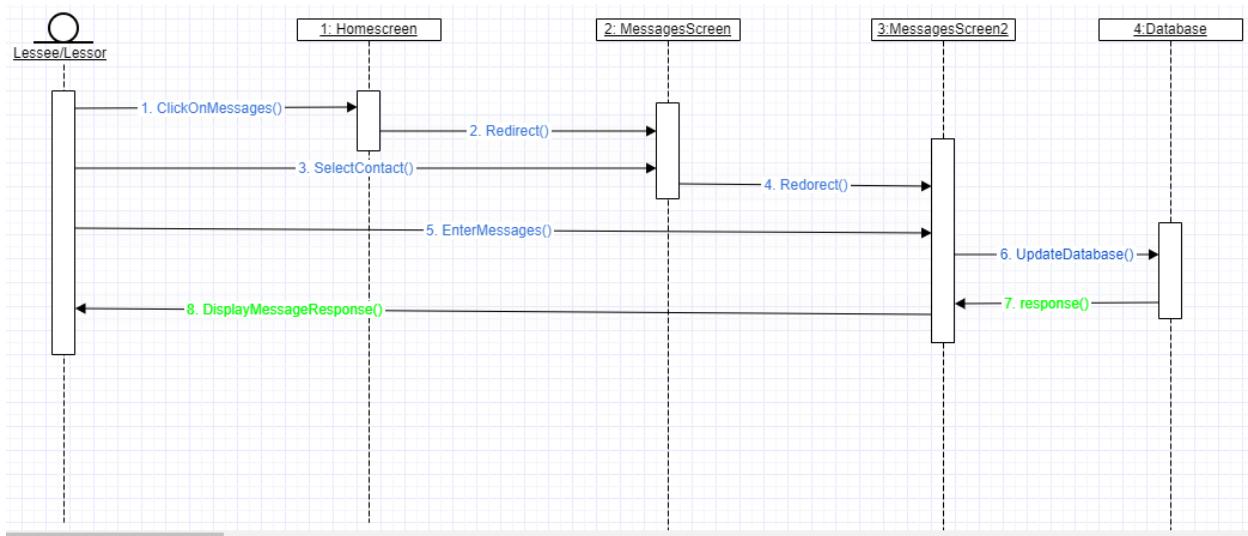


Figure 67: Sequence diagram for Messages

3.3.17 View Profile:

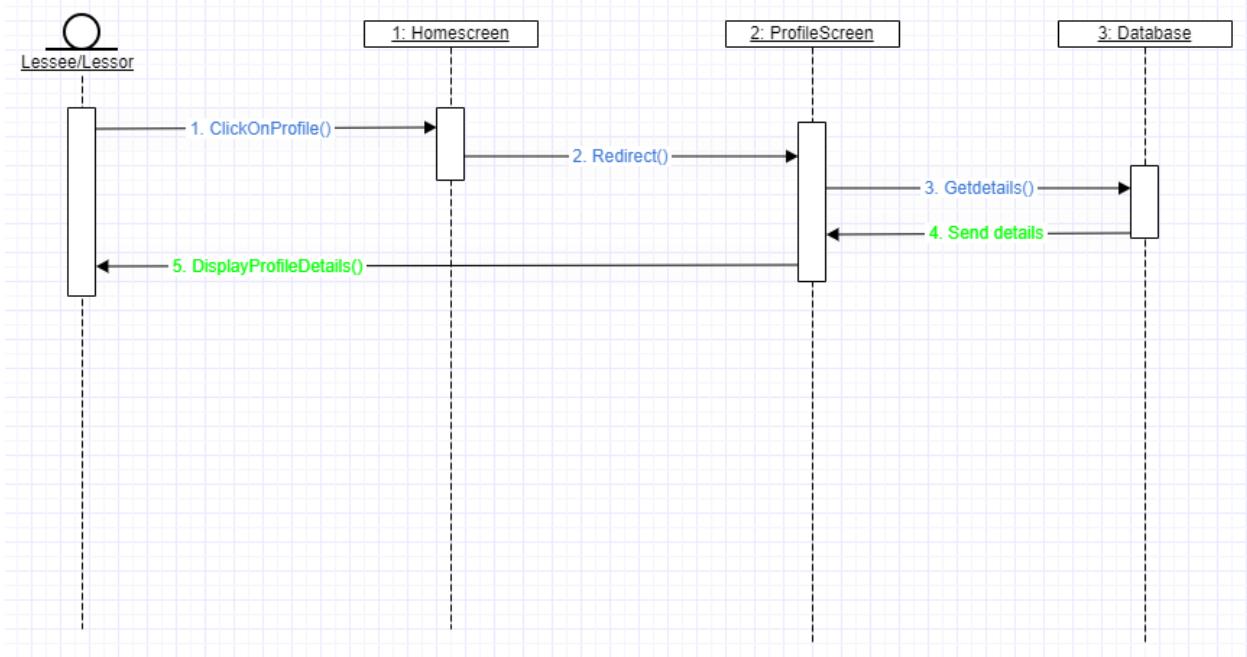


Figure 68: Sequence diagram for View Profile

3.3.18 Complete Profile:

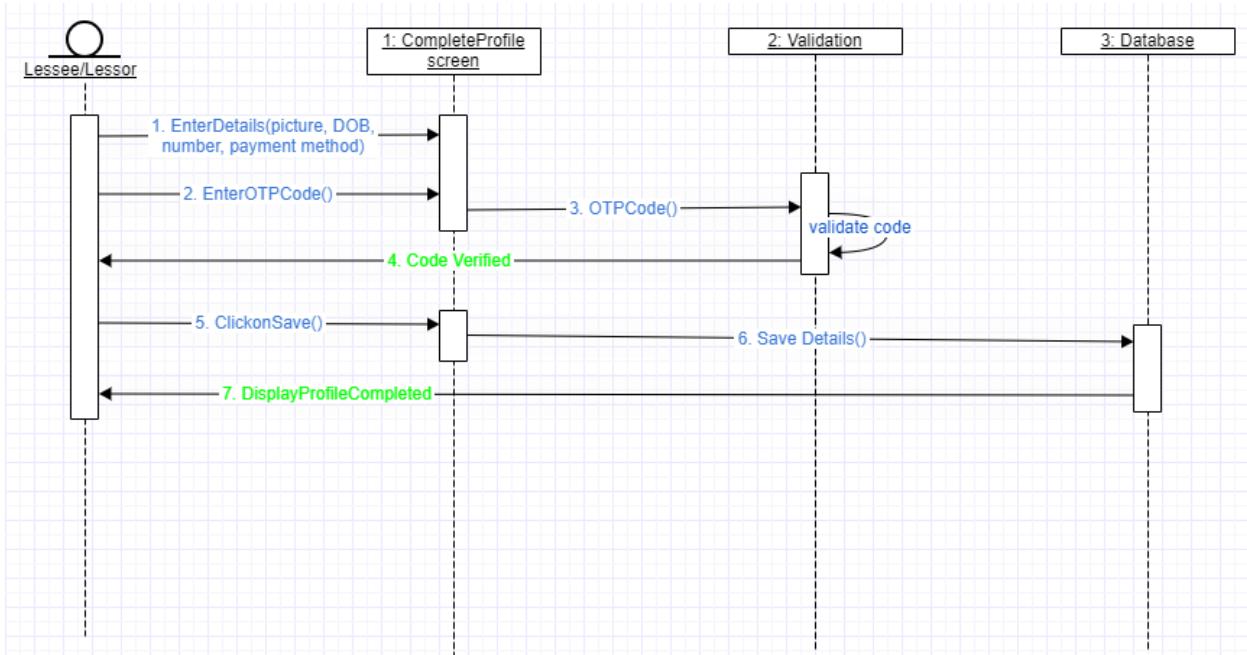


Figure 69: Sequence diagram for Complete Profile

3.3.19 Edit Profile:

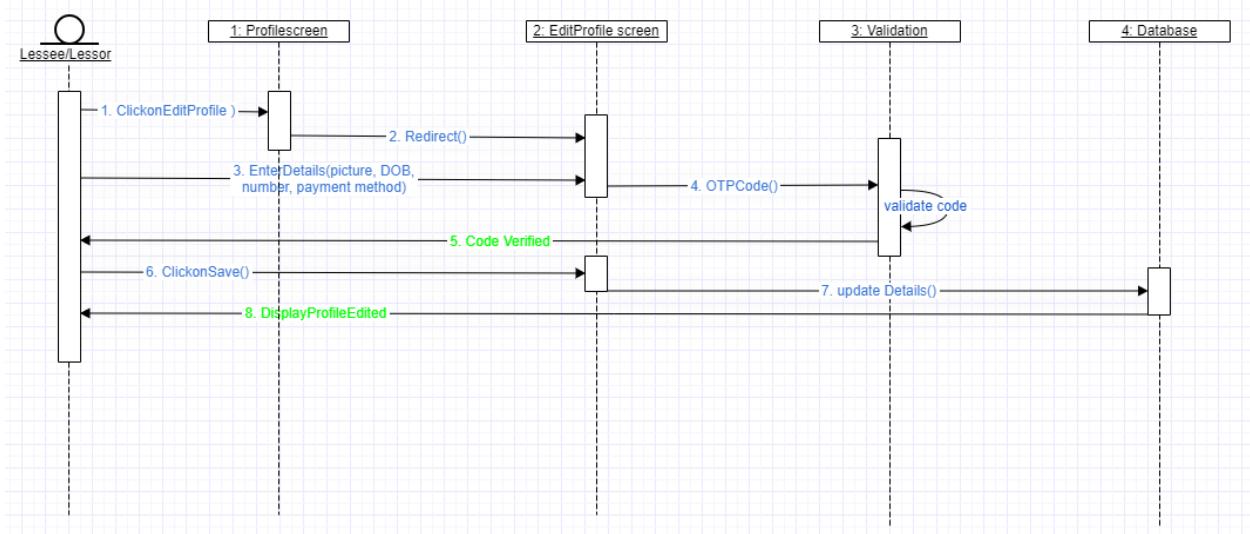


Figure 70: Sequence diagram for Edit Profile

3.3.20 View Account Details:

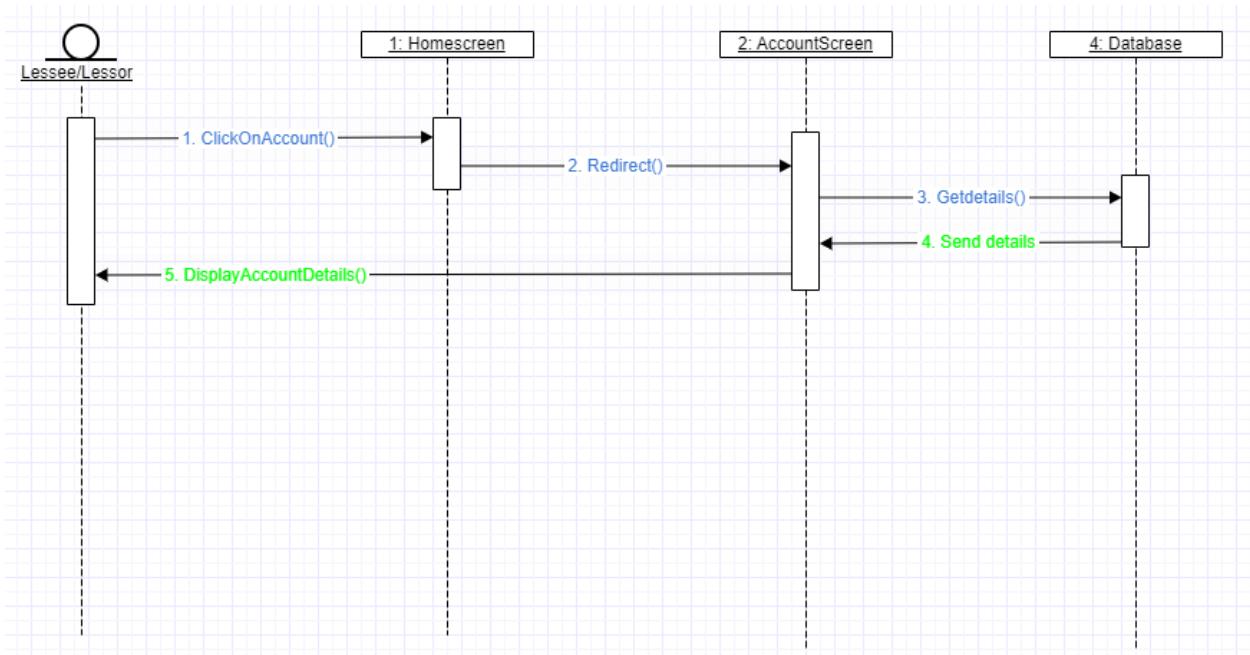


Figure 71: Sequence diagram for View Account Details

3.3.21 Sign Up:

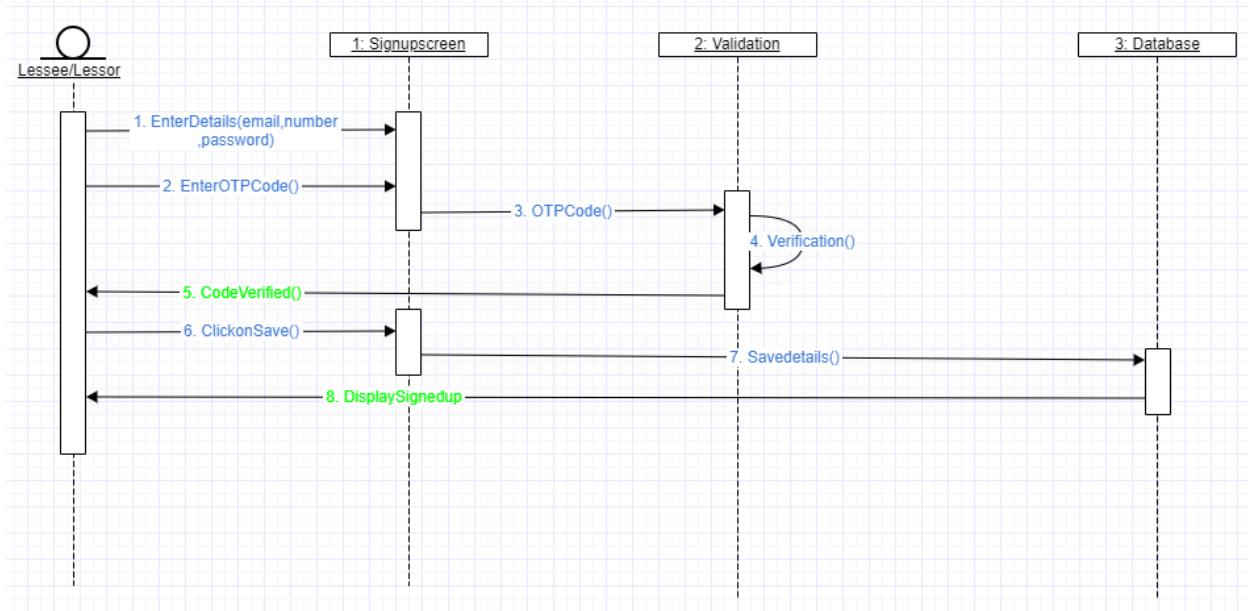


Figure 72: Sequence diagram for Signup

3.3.22 Login:

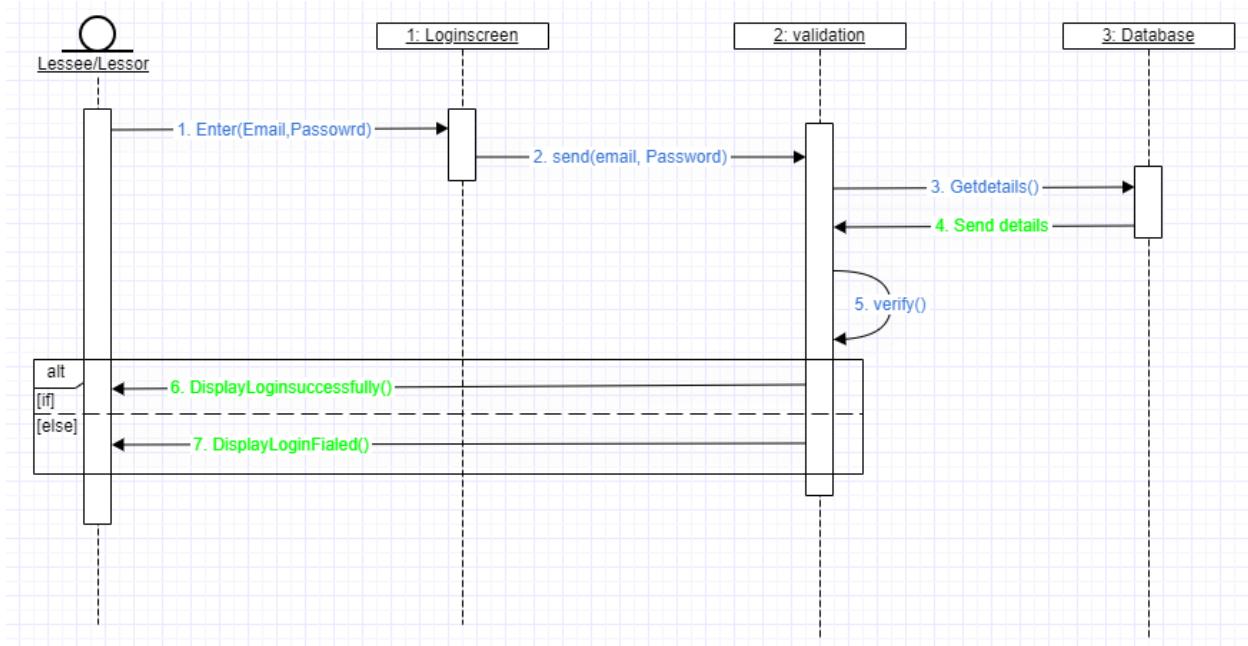


Figure 73: Sequence diagram for Login

3.3.23 Logout:

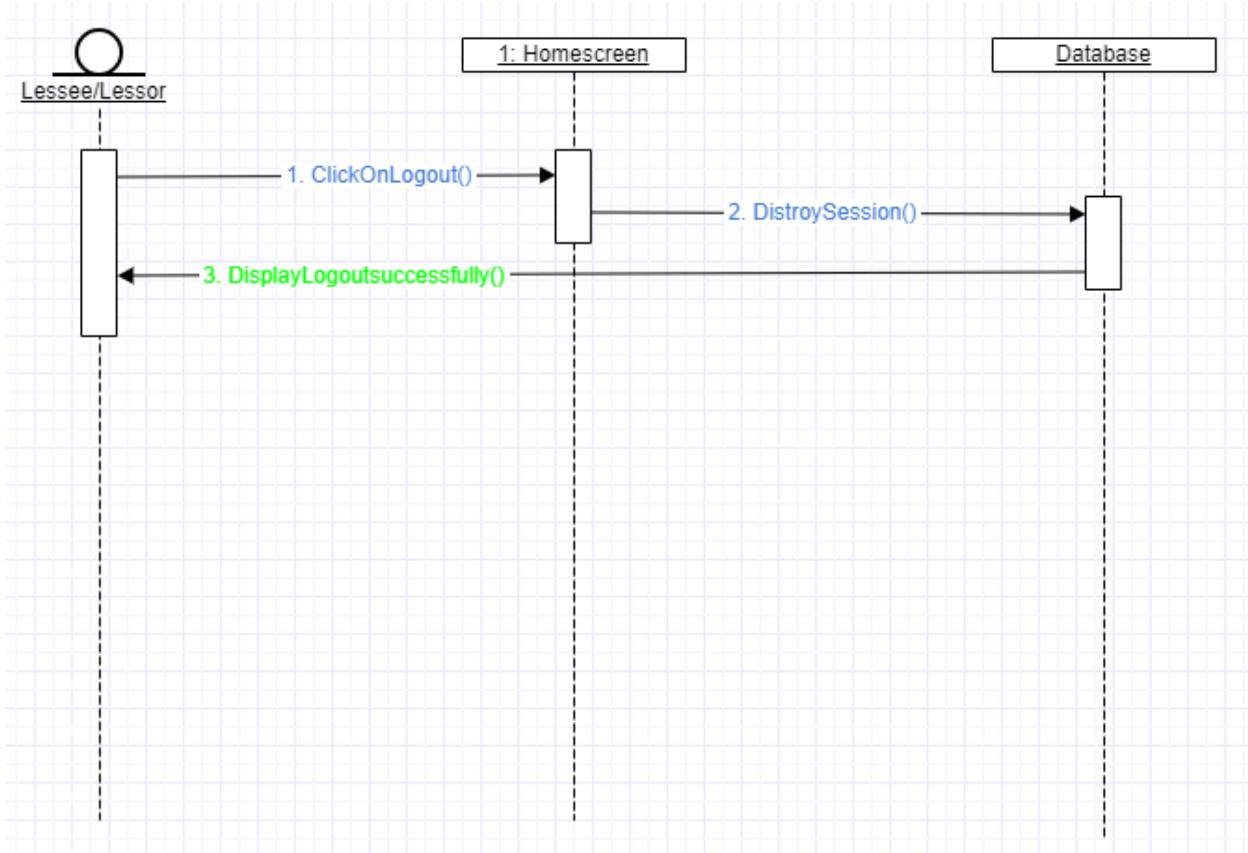


Figure 74: Sequence diagram for Logout

3.3.24 Forgot Password:

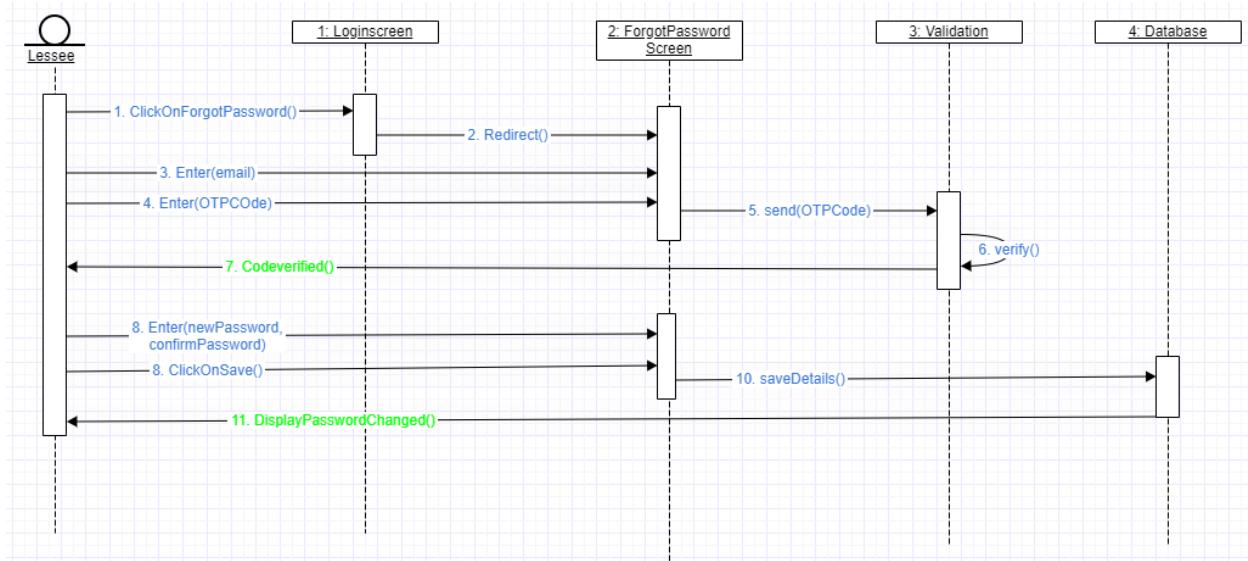


Figure 75: Sequence diagram for Forgot Password

3.3.25 Cancel Order:

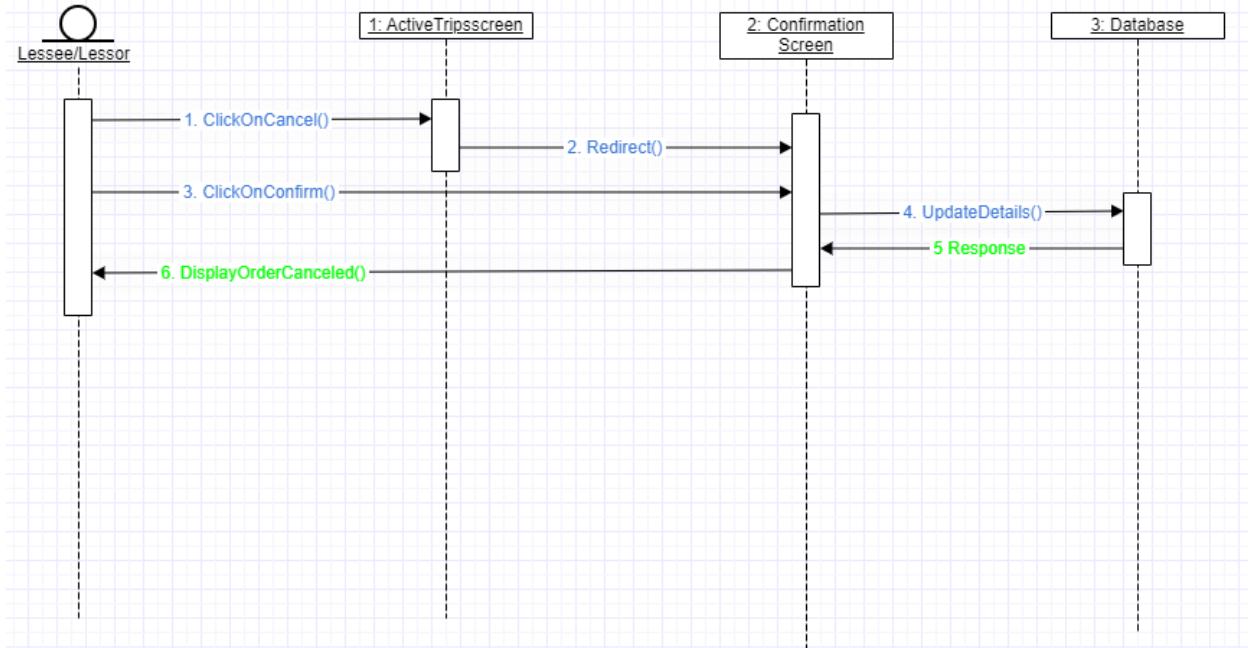


Figure 76: Sequence diagram for Cancel Order

3.4 Software Architecture:

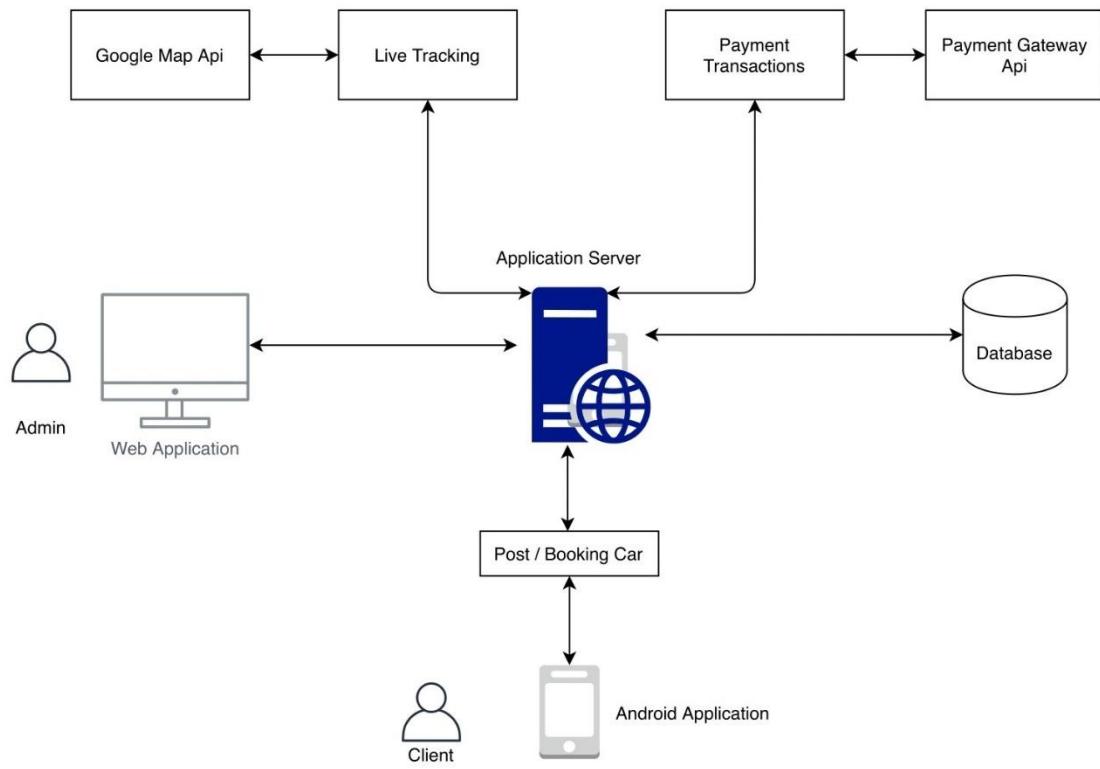


Figure 77: Software Architecture

3.5 Class Diagram:

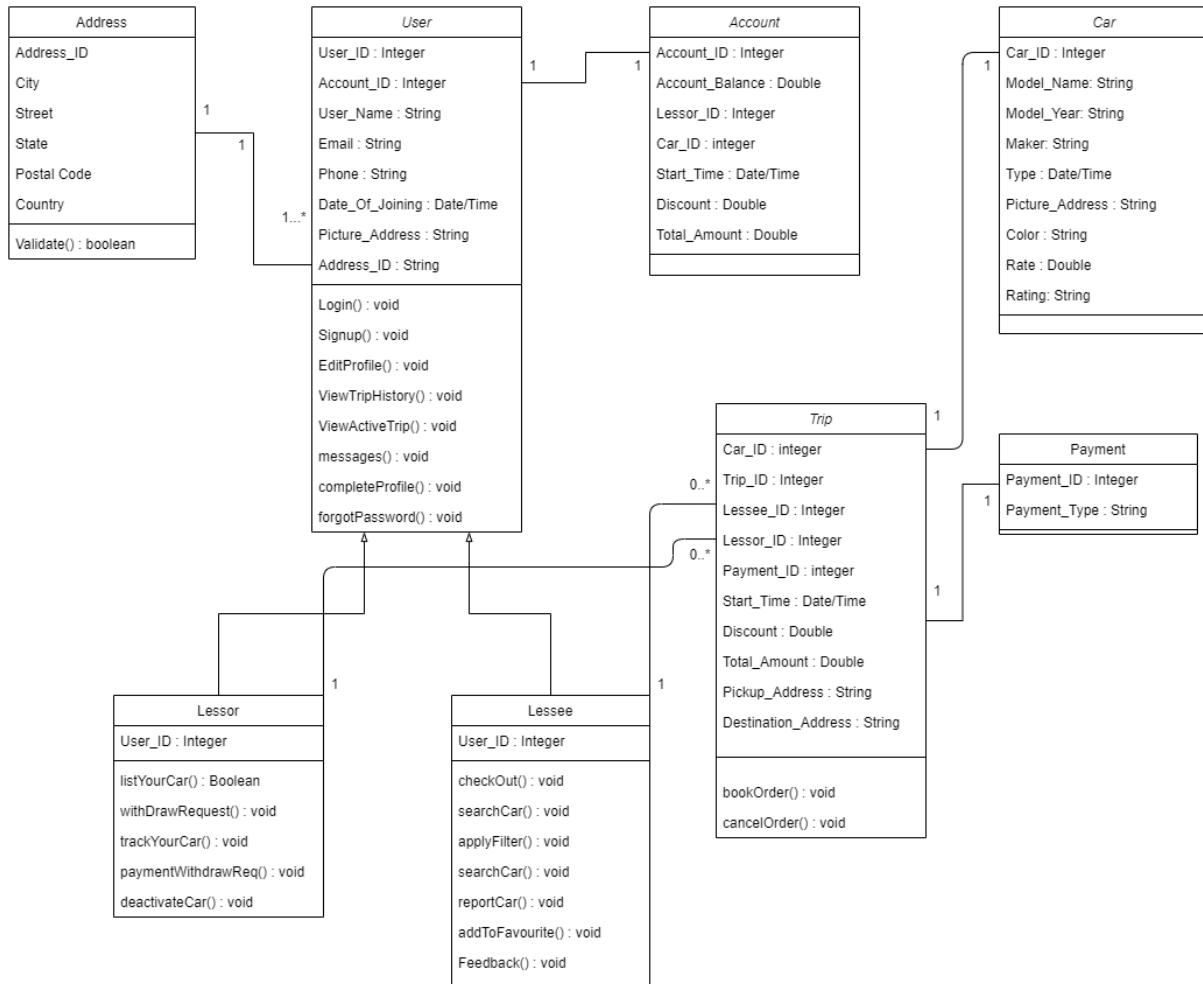


Figure 78: Class Diagram

3.6 Database Diagram:

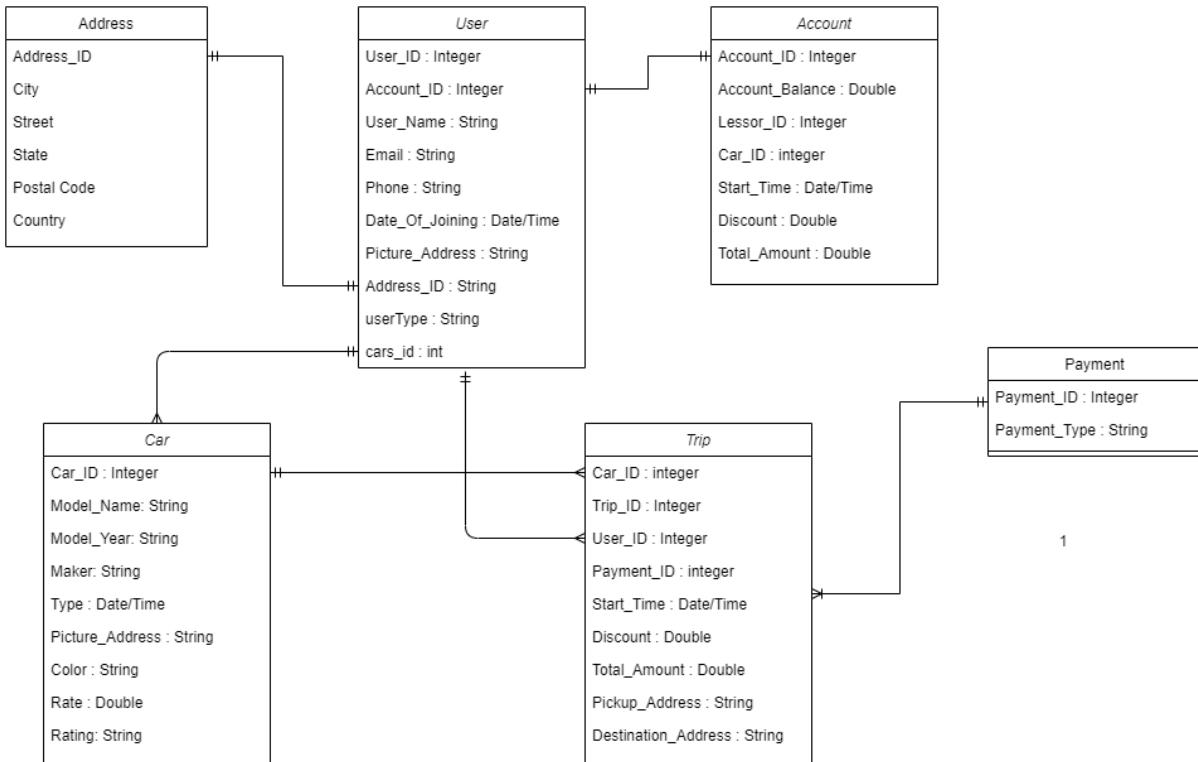


Figure 79: Database Diagram

3.7 Network Diagram:

Task	Assigned To	Start Date	End Date	Duration	Feb	March	April	May	June	July	Aug	Sept	Oct	Nov	Dec
Research	Ariba	2/10/2020	2/28/2020	18											
Planning	Ariba	3/1/2020	3/15/2020	14											
Requirements Analysis	Hammad	3/16/2020	3/31/2020	15											
UI Design	Faizan	4/1/2020	4/30/2020	29											
Backend (Lessee Interface)	Faizan	5/1/2020	6/15/2020	45											
Backend (Lessor Interface)	Hammad	6/16/2020	8/31/2020	76											
API Integration + Database	Ariba	9/1/2020	10/31/2020	60											
Testing and Debugging	Faizan	11/1/2020	11/10/2020	9											
Deployment	Hammad	11/11/2020	11/15/2020	4											
Documentation	Faizan & Hammad	11/16/2020	12/31/2020	45											

Figure 80: Network Diagram

3.8 Collaboration Diagrams:

3.8.1 Car Search:

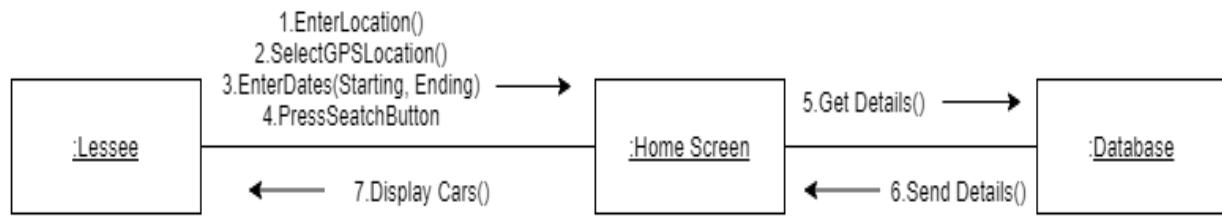


Figure 81: Collaboration diagram for Car Search

3.8.2 Advance Filter:

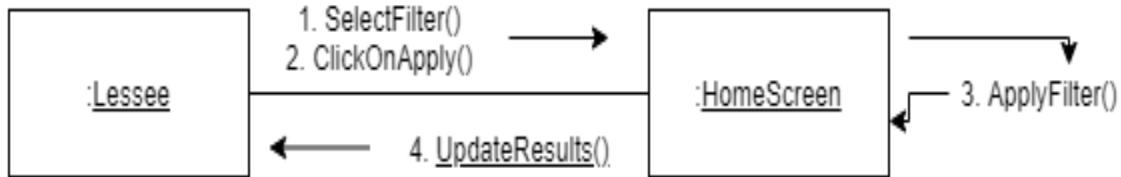


Figure 82: Collaboration diagram for Advance Filters

3.8.3 View Car Details:

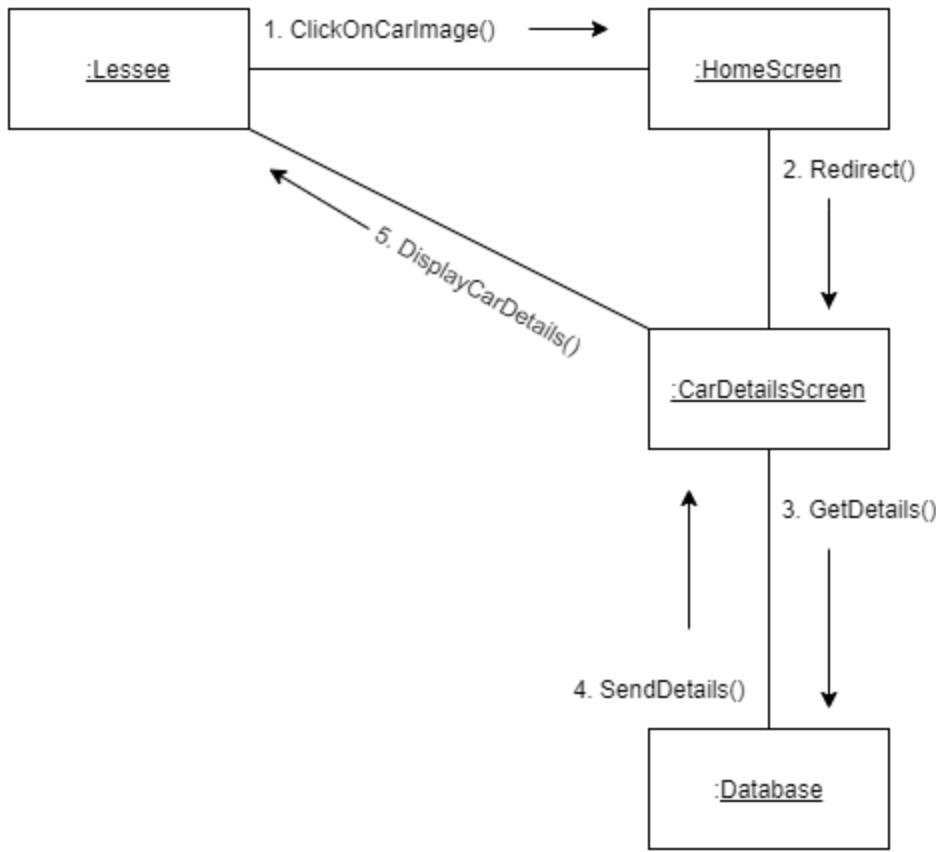


Figure 83: Collaboration diagram for View Car Details

3.8.4 Report Car Listing:



Figure 84: Collaboration diagram for Report Car Listing

3.8.5 Checkout:

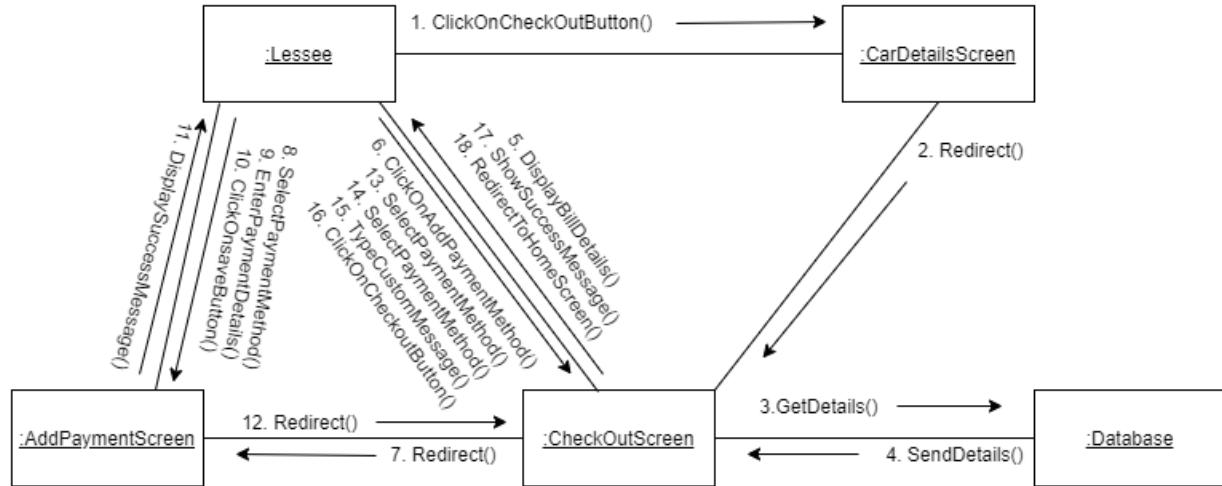


Figure 85: Collaboration diagram for Checkout

3.8.6 View Activity:

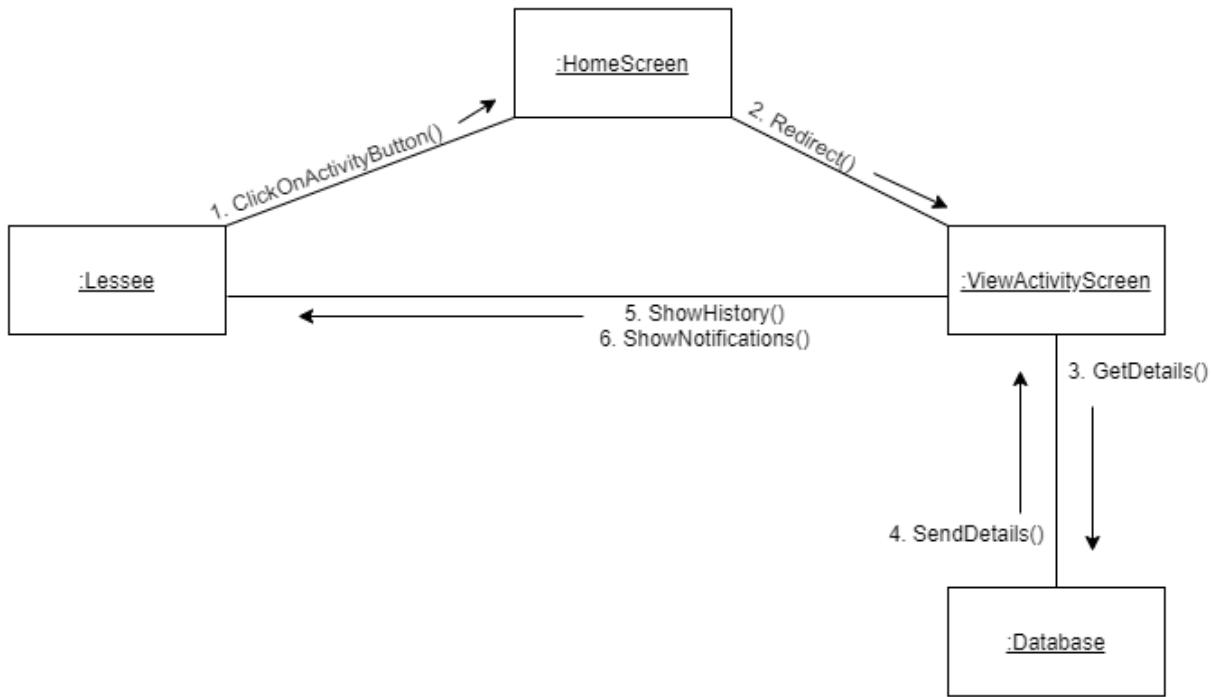


Figure 86: Collaboration diagram for View Activity

3.8.7 Add to Favourite:

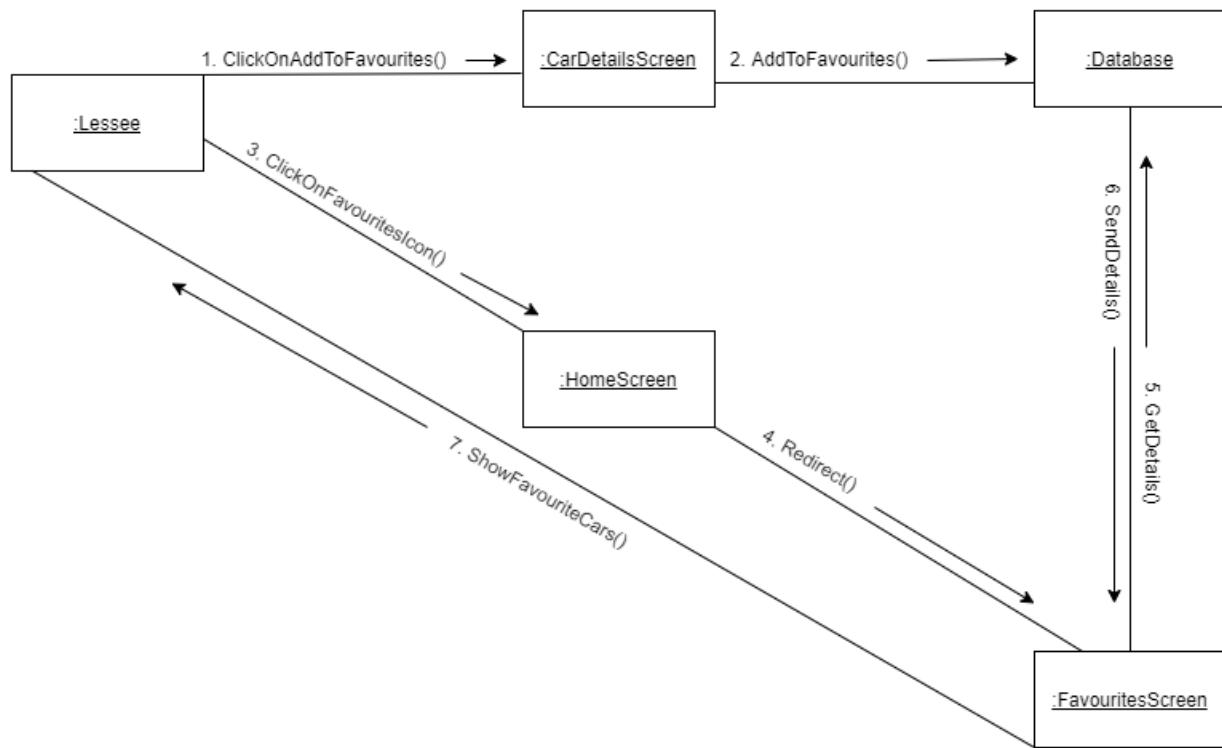


Figure 87: Collaboration diagram for Add to Favorite

3.8.8 Feedback:

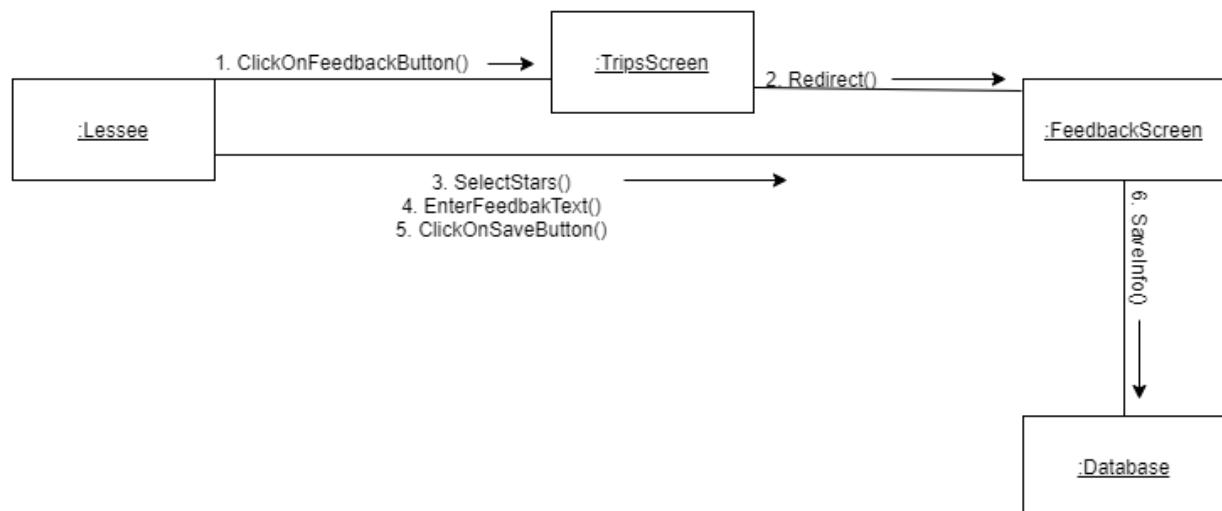


Figure 88: Collaboration diagram for Feedback

3.8.9 List Your Car:

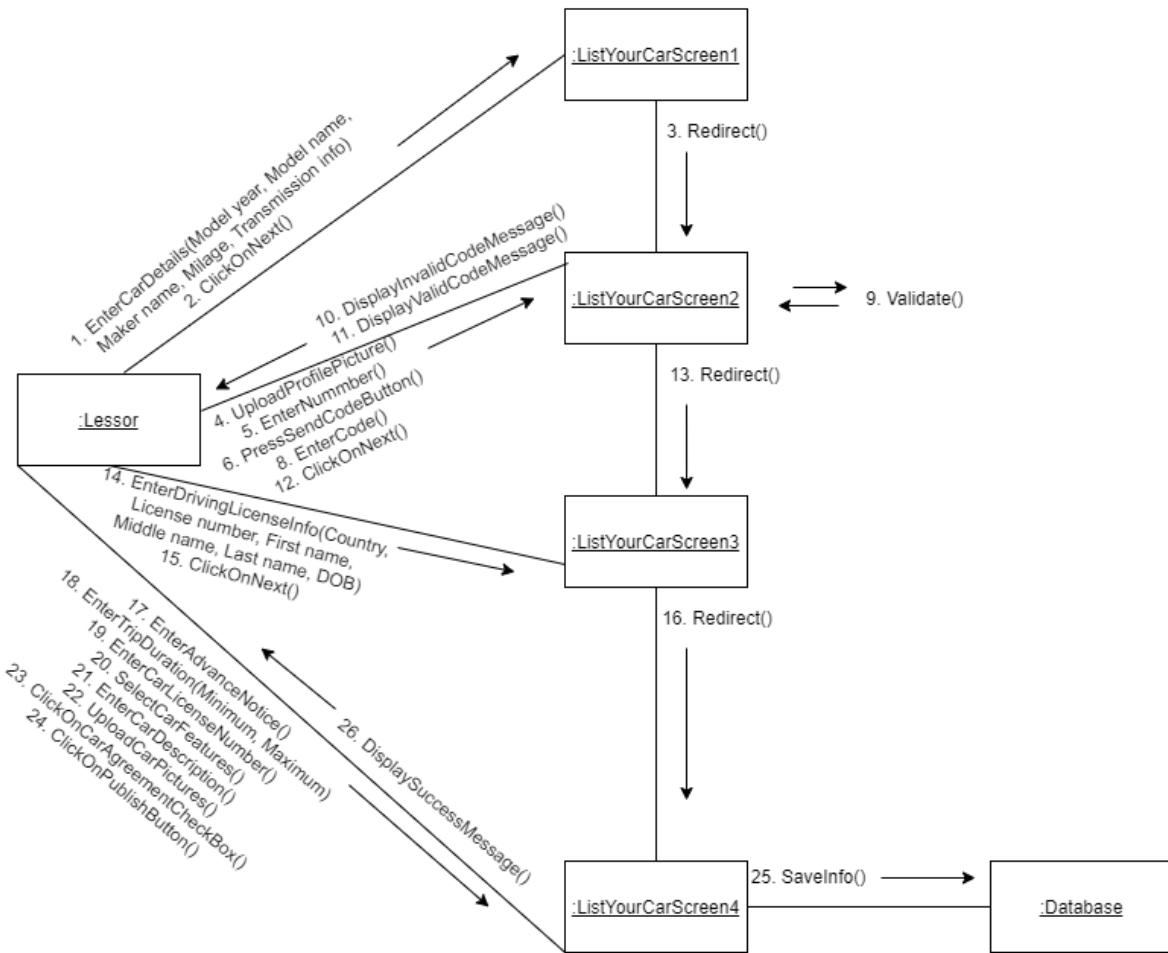


Figure 89: Collaboration diagram for List Your Car

3.8.10 Payment Withdraw Request:

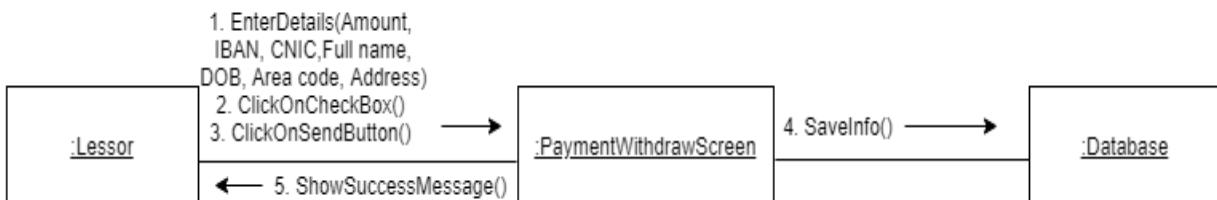


Figure 90: Collaboration diagram for Payment Withdraw Request

3.8.11 View Posted Cars:

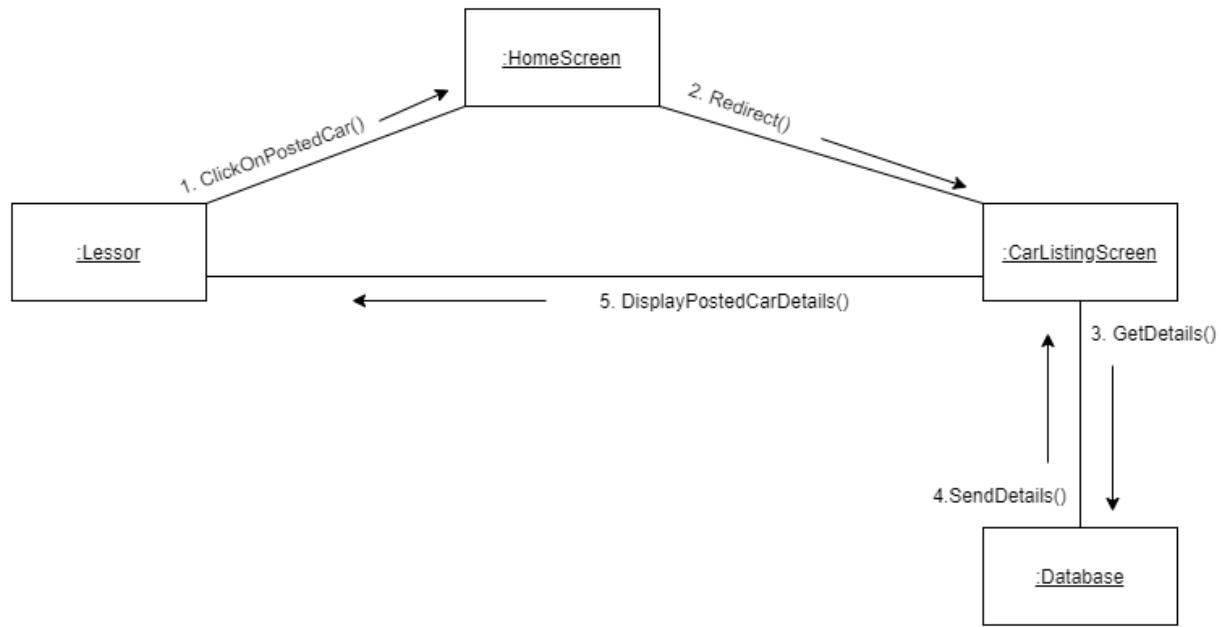


Figure 91: Collaboration diagram for View Posted Car

3.8.12 Track Your Car:

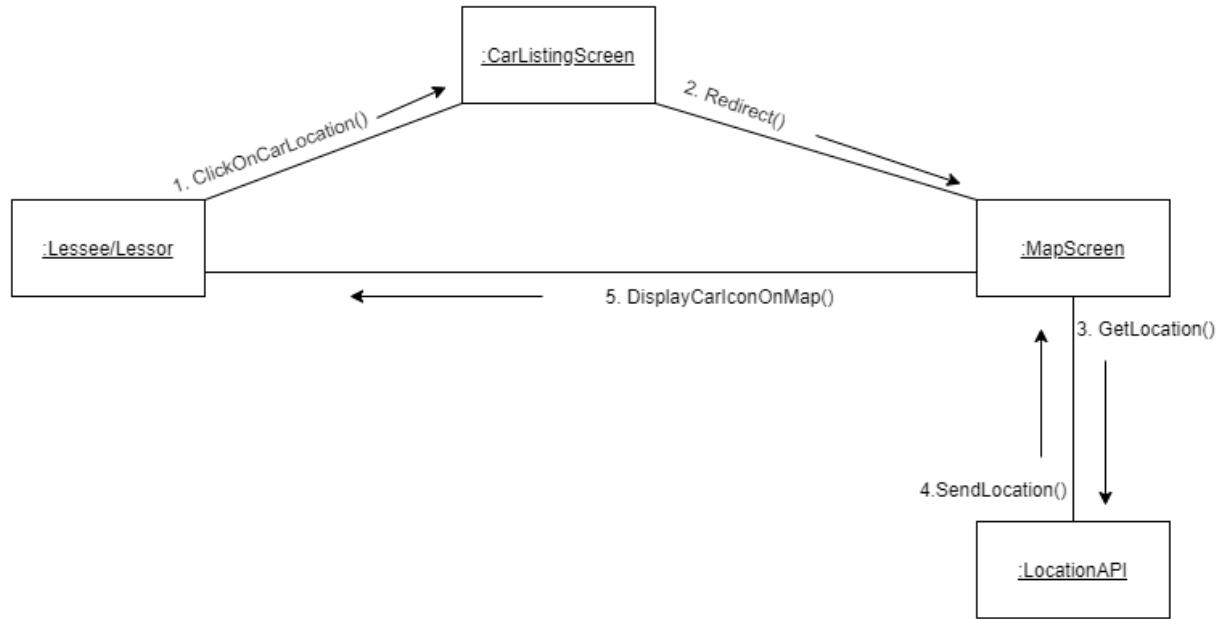


Figure 92: Collaboration diagram for Track Your Car

3.8.13 Deactivate Car Posting:

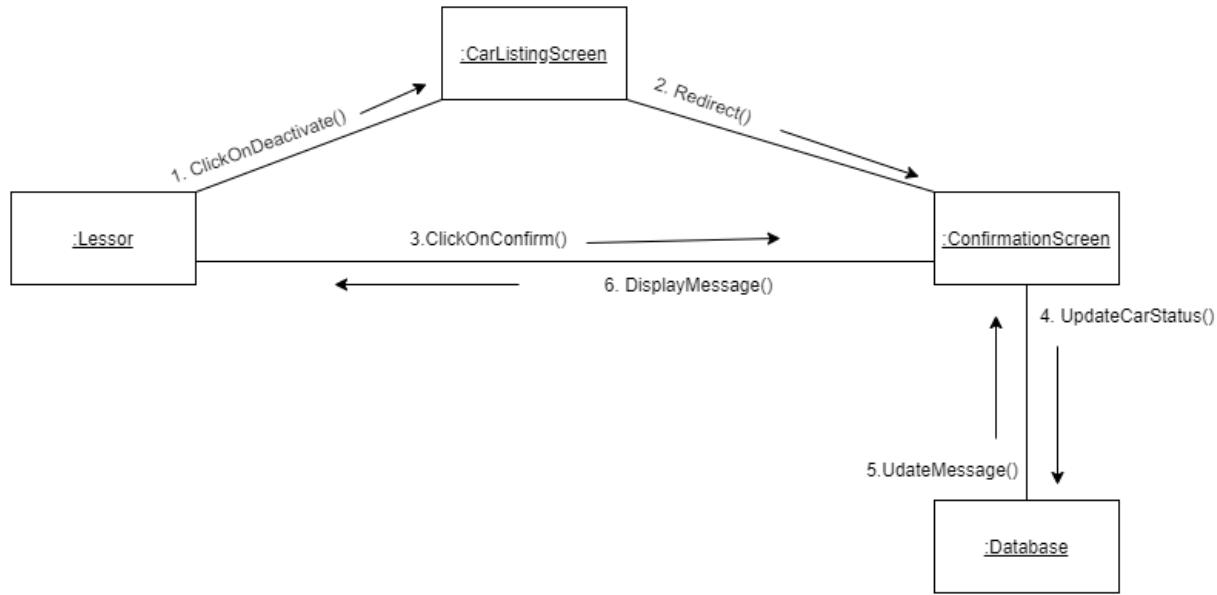


Figure 93: Collaboration diagram for Deactivate Car Posting

3.8.14 View Trip History:

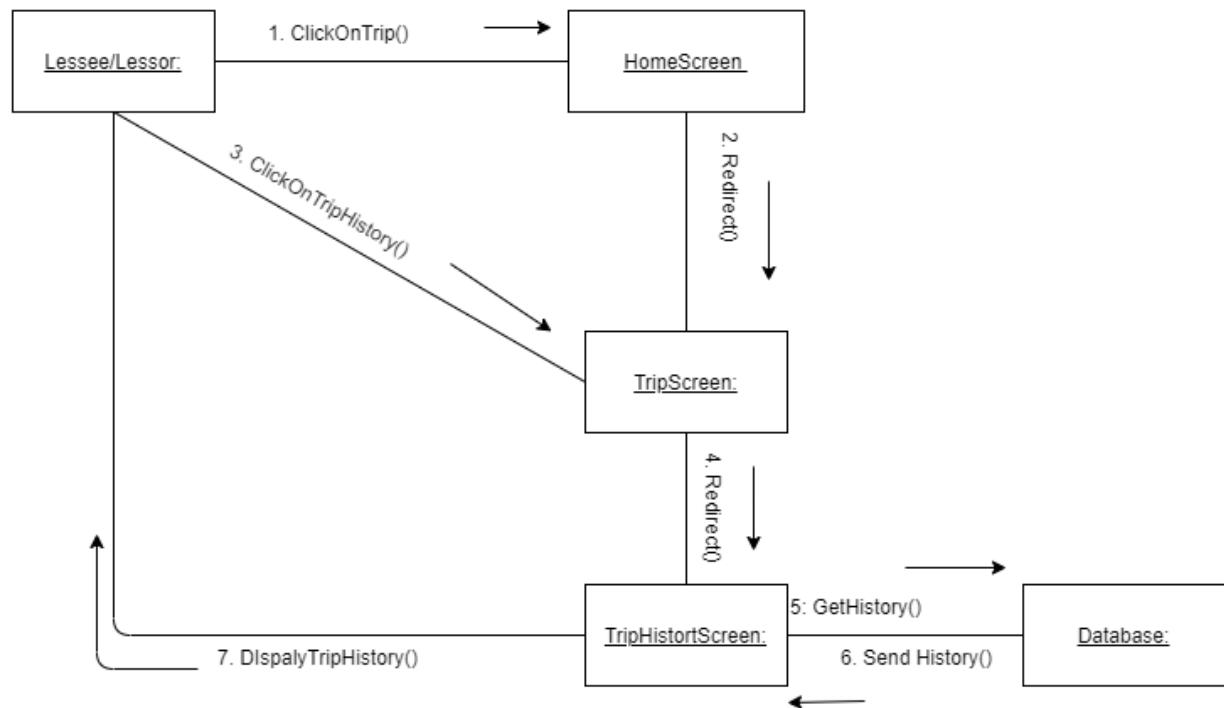


Figure 94: Collaboration diagram for View Trip History

3.8.15 View Active Trips:

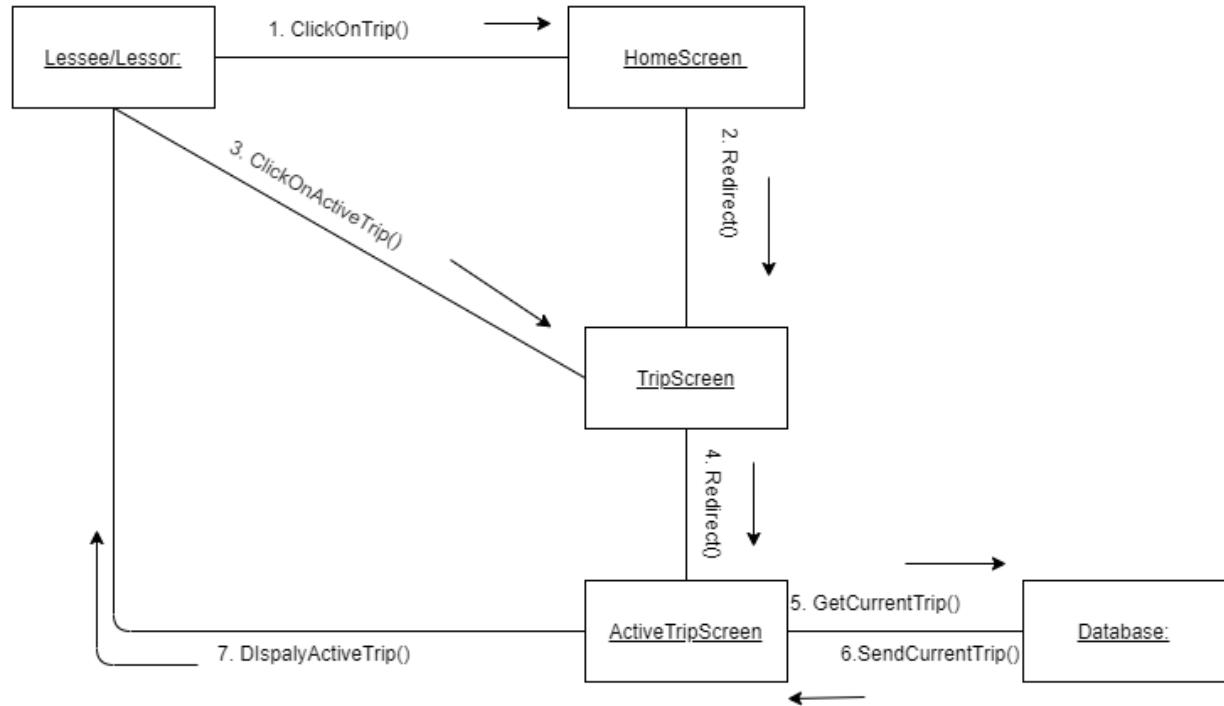


Figure 95: Collaboration diagram for View Active Trips

3.8.16 Messages:

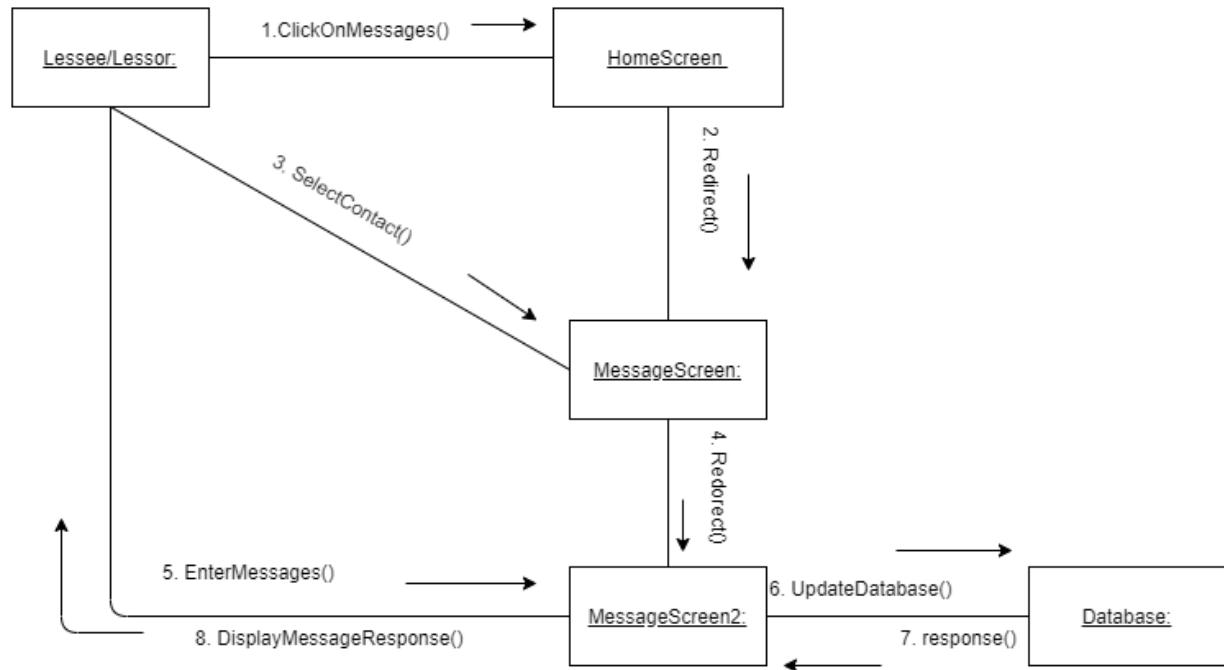


Figure 96: Collaboration diagram for Messages

3.8.17 View Profile:

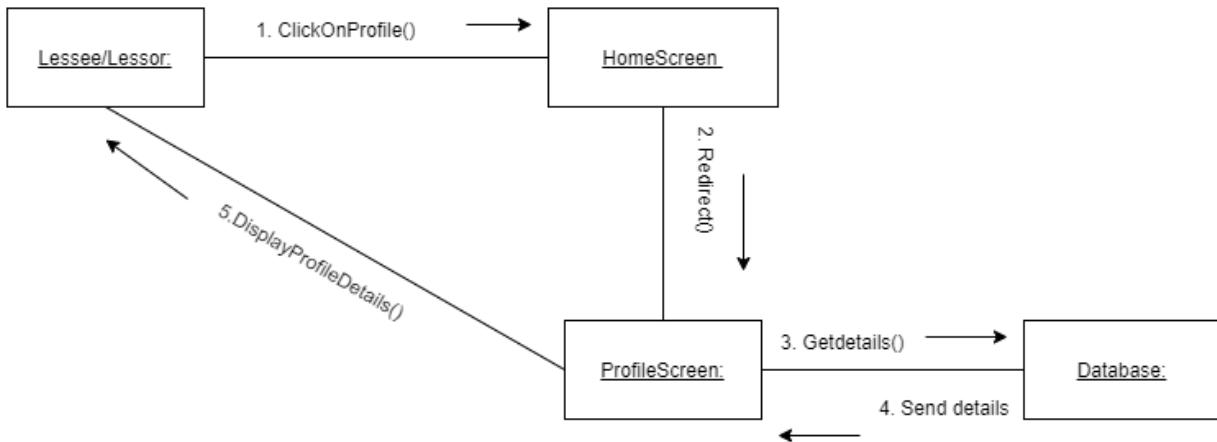


Figure 97: Collaboration diagram for View Profile

3.8.18 Complete Profile:

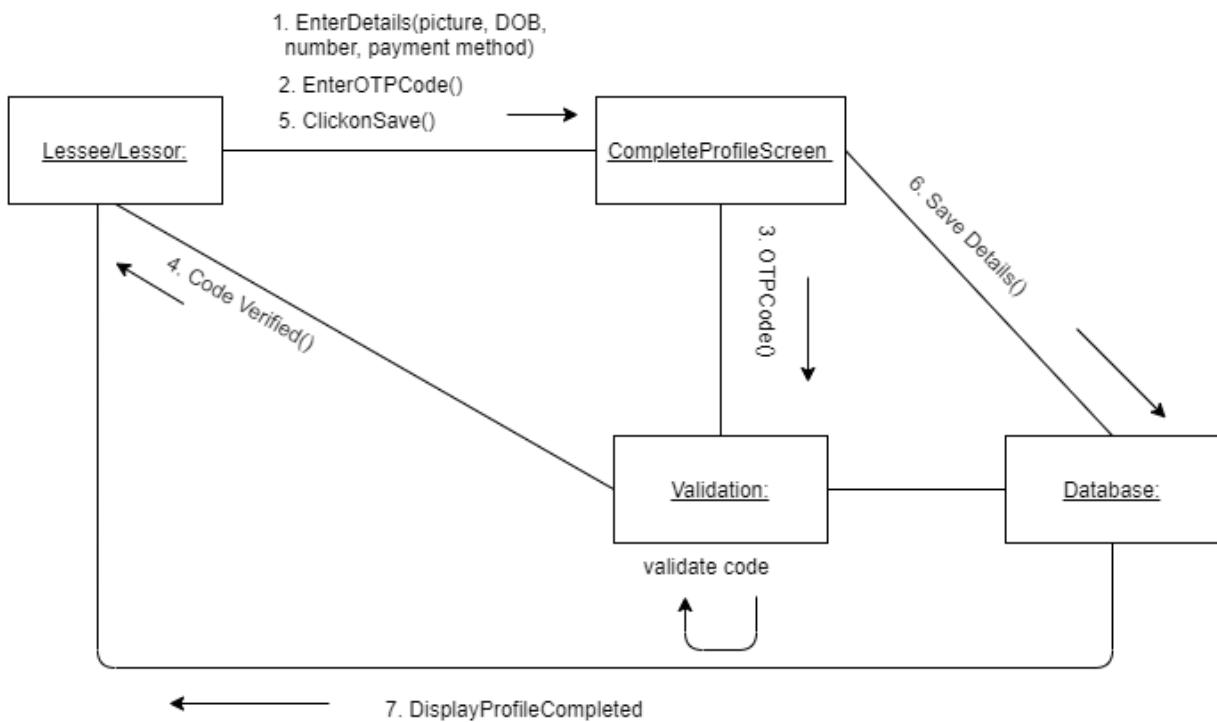


Figure 98: Collaboration diagram for Complete Profile

3.8.19 Edit Profile:

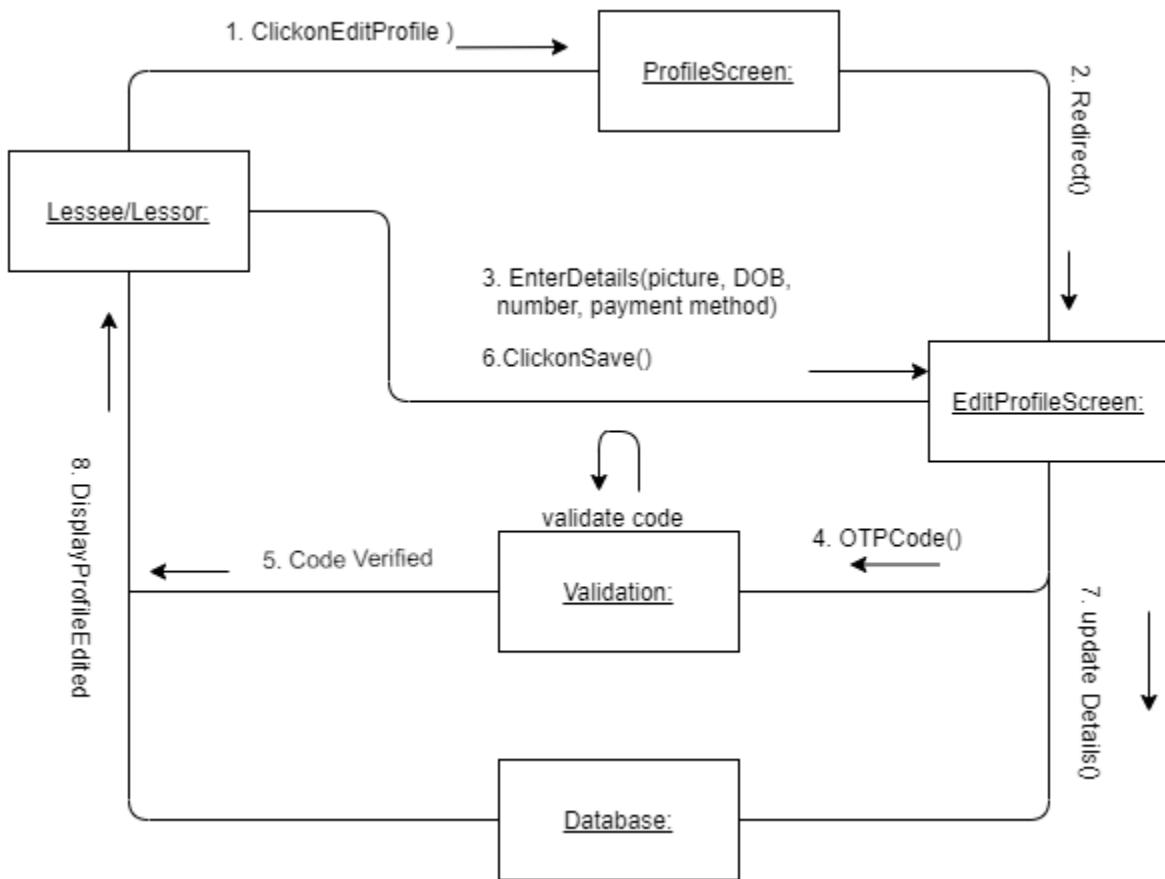


Figure 99: Collaboration diagram for Edit Profile

3.8.20 View Account Details:

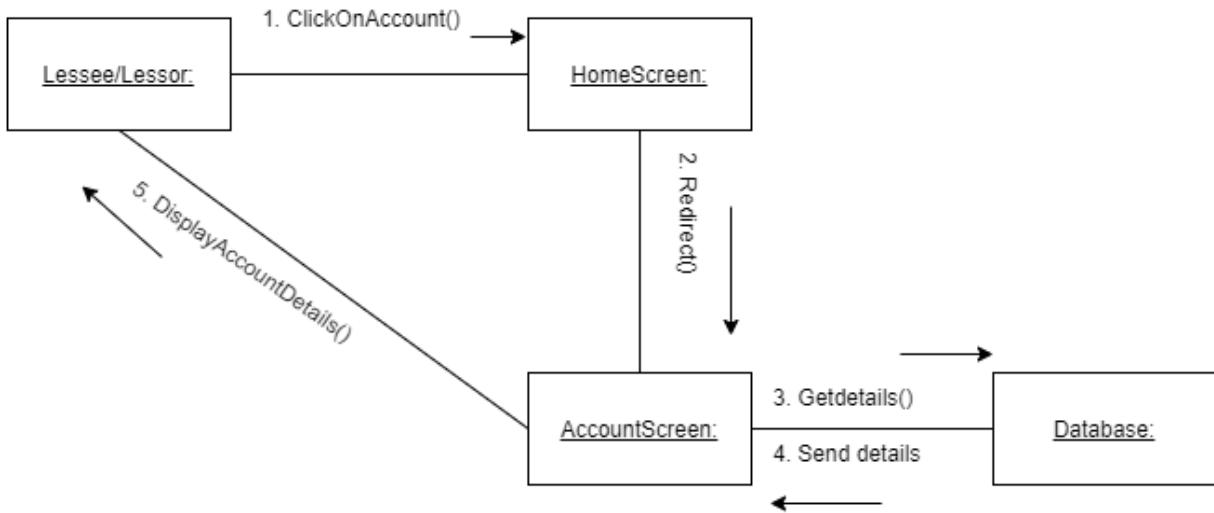


Figure 100: Collaboration diagram for View Account Details

3.8.21 Signup:

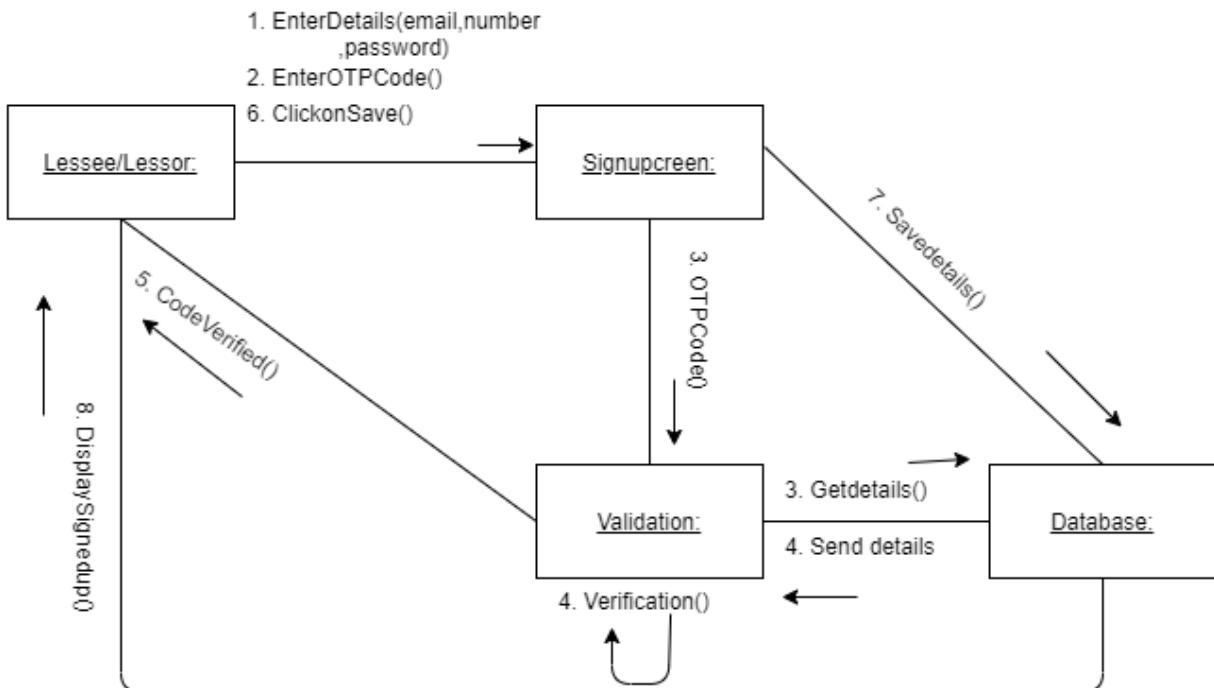


Figure 101: Collaboration diagram for Signup

3.8.22 Login:

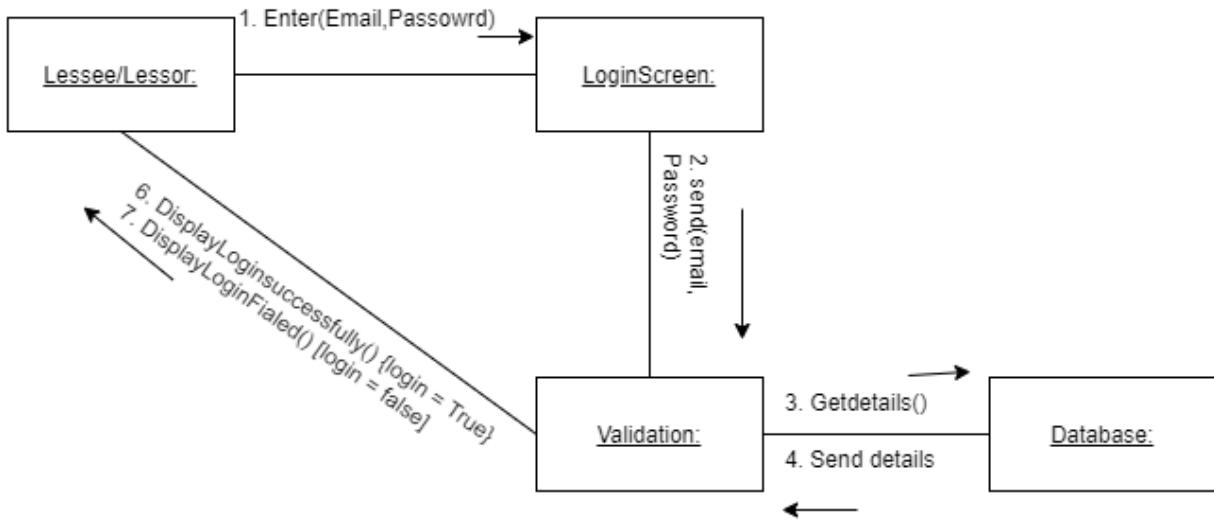


Figure 102: Collaboration diagram for Login

3.8.23 Logout:

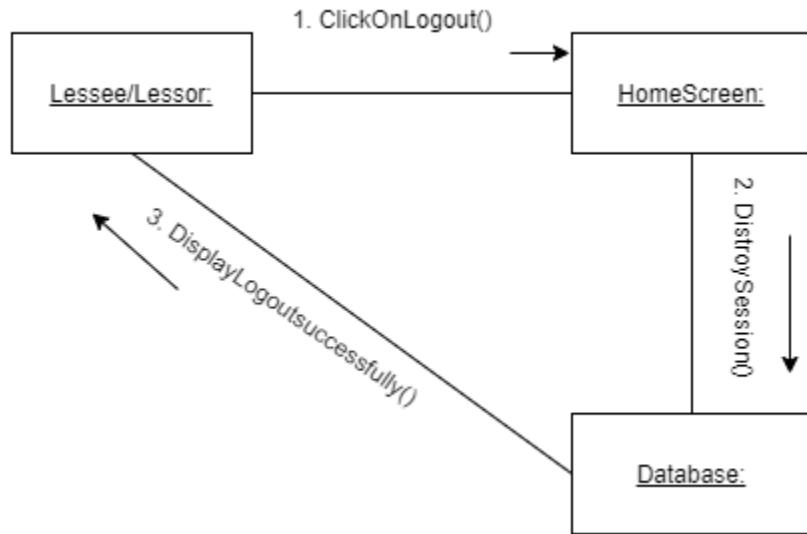


Figure 103: Collaboration diagram for Logout

3.8.24 Forgot Password:

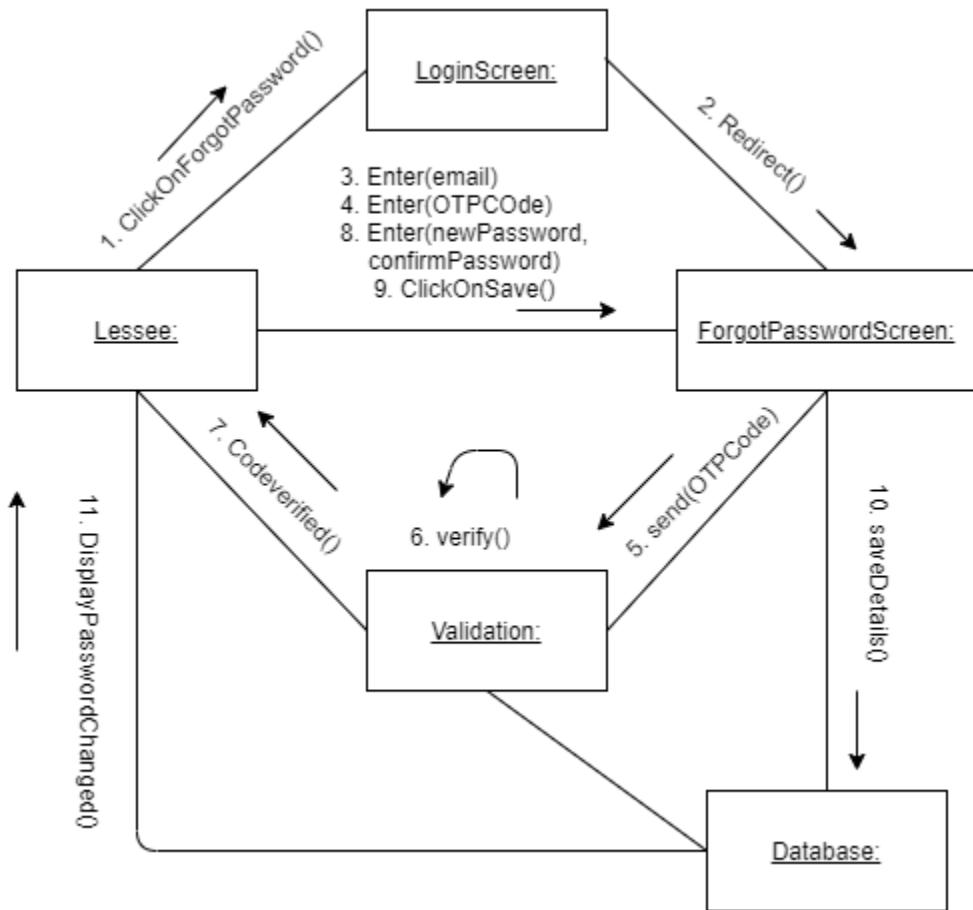


Figure 104: Collaboration diagram for Forgot Password

3.8.25 Cancel Order:

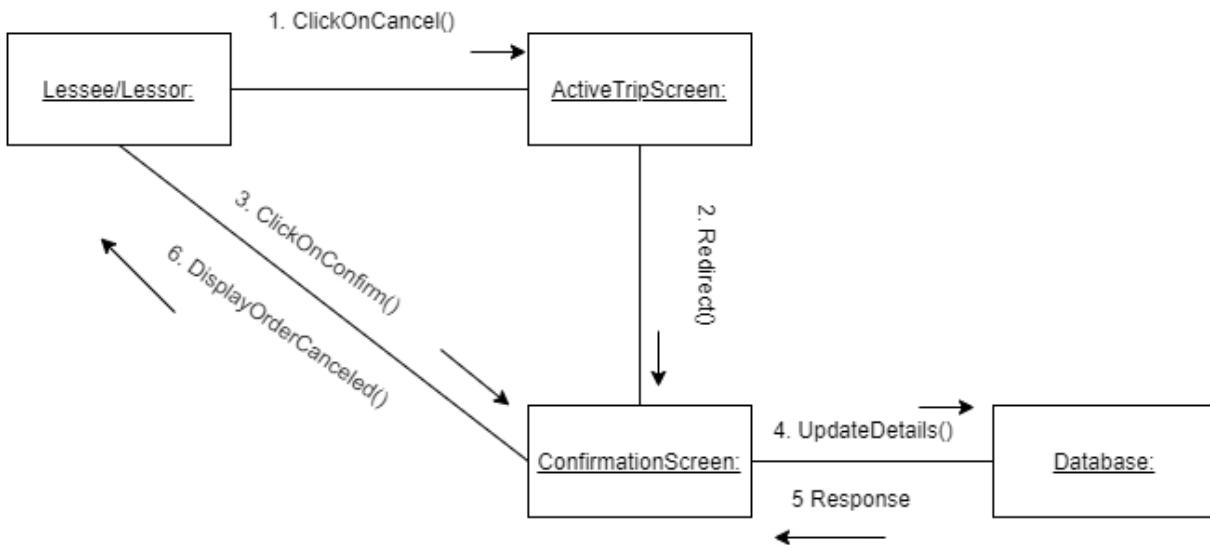


Figure 105: Collaboration diagram for Cancel Order

4 CHAPTER:4 System Testing

4.1 Test Cases:

4.1.1 Car Search:

Test Case #:	01	Test Case Name:	Car Search			
System:	CarSoul	Subsystem:	Home Screen, Database			
Designed By:	Syed Faizan Ali	Design Date:	25/07/2020			
Executed By:	Hammad Awan	Execute Date:	10/12/2020			
Short Description:	This Test case tests the car search functionality. Car Search allows lessee to search a car in a specific city by providing starting and ending dates for trip.					
Pre-Conditions						
The information of car was previously saved in the system and the lessee wants to search the car.						
Step:	Action	Input Data	Expected System Response	Actual Response	Pass/ Fail	Comment

1	Lessees access the Home screen.		The system shows the Home screen.	As Expected	Pass	
2	The lessee enters city name or select the current city by activating GPS.	Lahore,	The System show the textual visualization for lessee entering the text.	As Expected	Pass	
3	The lessee enters starting and ending dates for his/her trip.	Starting = 25/07/202, Ending = 30/07/202	The system shows the textual visualization for lessee entering the data.	As Expected	Pass	
4	The user presses the search button.		The system redirects the lessee to the car List screen and show the list of the car details.	As Expected	Pass	
Post-Conditions:						
1. A list of cars will be displayed.						

Table 26: Test case for Car search

4.1.2 Advance Filters:

Test Case #:	02	Test Case Name:	Advance Filter			
System:	CarSoul	Subsystem:	Car List Screen			
Designed By:	Syed Faizan Ali	Design Date:	25/07/2020			
Executed By:	Hammad Awan	Execute Date:	10/12/2020			
Short Description:	This Test case tests the Advance Filter functionality. Advance filters allow lessee to filter out results of car search which are not relevant to his/her needs and choice. It allows him/her to narrow down the results of car search.					
Pre-Conditions						
The lessee searched for car and a list of cars was displayed by the system for lessee to view.						

Step:	Action	Input Data	Expected System Response	Actual Response	Pass/ Fail	Comment
1	Lessees access the car list screen.		The system shows the car list.	As Expected	Pass	
2	The lessee selects one or more filter by clicking on the filters button. Price, date, model etc.	Price = 1000, Date= 25/07/202, Mode= abs,	The System shows the updated car list according to the filter.	As Expected	Pass	
Post-Conditions:						
1. An updated list of cars will be displayed.						

Table 27: Test case for Advance filters

4.1.3 View Car Details

Test Case #:	03	Test Case Name:	View Car Details
System:	CarSoul	Subsystem:	Car Details Screen, Database
Designed By:	Syed Faizan Ali	Design Date:	25/07/2020
Executed By:	Hammad Awan	Execute Date:	10/12/2020
Short Description:	This Test case tests the View Car functionality. View car details allow lessee to view details of any car.		

Pre-Conditions

1. The information of car was previously saved in the system and the lessee wants to view it.

Step:	Action	Input Data	Expected System Response	Actual Response	Pass/ Fail	Comment
1	Lessees access the car list screen.		The system shows the car list screen and show list of car cards.	As Expected	Pass	

2	Lessees click on the one of the car cards.		The System redirects the lessee to the car details screen and show the details of the car.	As Expected	Pass	
Post-Conditions:						
1. The details of cars will be displayed.						

Table 28: Test case for View car details

4.1.4 Report Car Listing

Test Case #:	04	Test Case Name:	Report Car Listing													
System:	CarSoul	Subsystem:	Car Listing and Report screen, Database													
Designed By:	Syed Faizan Ali	Design Date:	25/07/2020													
Executed By:	Hammad Awan	Execute Date:	11/12/2020													
Short Description:	This Test case tests the Report Car Listing functionality. Report car listing allows lessee to report a car which is violating the application terms and conditions to get it reviewed.															
Pre-Conditions																
1. The information of car was previously saved in the system and the lessee wants to report it.																
Step:	Action	Input Data	Expected System Response	Actual Response	Pass/ Fail	Comment										
1	Lessees access the car listing screen.		The system shows the car listing screen and show list of car cards.	As Expected	Pass											

2	Lessees click on the Report button.		The System redirects to Report screen.	As Expected	Pass	
3	Lessee select the reason of for reporting the car listing.	Report text = text abc	The system shows the textual visualization that lessee is entering the text.	As Expected	Pass	
4	Lessee click on submit button to submit the report.		The System shows the message that your report has been submitted.	As Expected	Pass	
Post-Conditions:						
1. The system will remove the reported car listing from list of cars shown to lessee.						

Table 29: Test case for Report car listing

4.1.5 Checkout

Test Case #:	05	Test Case Name:	Checkout
System:	CarSoul	Subsystem:	Checkout and database.
Designed By:	Syed Faizan Ali	Design Date:	25/07/2020
Executed By:	Hammad Awan	Execute Date:	11/12/2020
Short Description:	The checkout allows the lessee to confirm a car to hire for rent by making its payment. This Test case tests the Checkout functionality.		

Pre-Conditions

1. The car was available to hire for rent.

Step:	Action	Input Data	Expected System	Actual	Pass/ Fail	Comment

			Response	Response		
1	Lessees access the car details screen.		The system shows the car details screen.	As Expected	Pass	
2	Lessees click on the checkout button.		The System redirects to checkout screen and shows the details about the order.	As Expected	Pass	
3	Lessees click on the order button.		The system shows the message that your order has been placed.	As Expected	Pass	
Post-Conditions:						
1. The order list of lessees will be updated.						

Table 30: Test case for Checkout

4.1.6 View Activity

Test Case #:	06	Test Case Name:	View Activity			
System:	CarSoul	Subsystem:	View Activity screen, Database			
Designed By:	Syed Faizan Ali	Design Date:	25/07/2020			
Executed By:	Hammad Awan	Execute Date:	11/12/2020			
Short Description:	This Test case tests the View Activity functionality. View Activity allow the lessee to view all the details of the activities (s)he has performed. Lessee can view the history of cars (s)he booked in the past and any important notification available.					
Pre-Conditions						
1. Lessee must be logged in to view his/her Activity information.						

Step:	Action	Input Data	Expected System Response	Actual Response	Pass/ Fail	Comment
1	Lessees access the home screen.		The system shows the Home screen.	As Expected	Pass	
2	Lessees access the view activity screen by clicking activity button.		The system redirects to view activity screen and show the activity details.	As Expected	Pass	
Post-Conditions:						
1. Information will be Successfully displayed.						

Table 31: Test case for View Activity

4.1.7 Add to Favourite

Test Case #:	07	Test Case Name:	Add to Favourite													
System:	CarSoul	Subsystem:	Car details screen, database													
Designed By:	Syed Faizan Ali	Design Date:	25/07/2020													
Executed By:	Hammad Awan	Execute Date:	12/12/2020													
Short Description:	This Test case tests the Add to Favourite functionality. Add to favorite allow the lessee to save car as its favorite car. The lessee clicks on the heart icon of the car post than this car post saves as favorite car in database.															
Pre-Conditions																
1. The car was available to hire for rent.																
Step:	Action	Input Data	Expected System Response	Actual Response	Pass/ Fail	Comment										
1	Lessees access the car details screen.		The system shows the car details screen.	As Expected	Pass											

2	Lessees click on the heart icon.		The system show visualization that hear icon fill with the red color and show message car added to favorite.	As Expected	Pass	
3	Lessees click on the heart icon again.		The system shows the visualization that heart icon remove color and become transparent and message show that car hi removed from the favorite car list.	As Expected	Pass	
Post-Conditions:						
1. Heart icon will be filled with color.						

Table 32: Test case for Add to favorite

4.1.8 Feedback

Test Case #:	08	Test Case Name:	Feedback			
System:	CarSoul	Subsystem:	Feedback screen, Database			
Designed By:	Syed Faizan Ali	Design Date:	25/07/2020			
Executed By:	Hammad Awan	Execute Date:	13/12/2020			
Short Description:	Feedback allow the lessee to give the feedback about the car. System allow the lessee to choose stars ratings for car and add text for comments. This Test case tests the Feedback functionality.					
Pre-Conditions						
1. Lessee must be logged In. 2. Lessee must has booked a car.						
Step:	Action	Input Data	Expected System Response	Actual Response	Pass/ Fail	Comment

1	Lessees access the car details screen.		The system shows the car details screen.	As Expected	Pass	
2	Lessees click on the feedback.		The system redirects to the feedback screen.	As Expected	Pass	
3	Lessees click on the star's icons.		The system shows the visualization that stars icon filling with yellow colors.	As Expected	Pass	
4	Lessees enter the feedback comments to the text area.	Comments = good services.	The system shows the visualization that lessee is entering text.	As Expected	Pass	
5	Lessees click on the submit button.		The system shows the message that your feedback has been submitted successfully.	As Expected	Pass	
Post-Conditions:						
1. Star icons will be filled with color.						

Table 33: Test case for Feedback

4.1.9 List Your Car

Test Case #:	09	Test Case Name:	List Your Car.
System:	CarSoul	Subsystem:	List Your Car screens, Database
Designed By:	Syed Faizan Ali	Design Date:	25/07/2020
Executed By:	Hammad Awan	Execute Date:	13/12/2020

Short Description:		List Your Car allow the lessor to post his/her car details on to the application so lessee can see card details and booked car for rent. This Test case tests the List Your Car functionality.									
Pre-Conditions											
1. Lessee must be logged In.											
Step:	Action	Input Data	Expected System Response	Actual Response	Pass/ Fail	Comment					
1	Lessors click on the List your car button.		The system shows the List your car screen.	As Expected	Pass						
2	Lessees enter the car details, name, model, company, Mileage.	Name = civic, Model = GLI, Company= Toyota, Mileage = 100	The system shows the visualization that lessor filling the form.	As Expected	Pass						
3	Lessors click on the next button.		The system redirects to the other car listing screen for more information.	As Expected	Pass						
4	Lessors enter mobile number.	Number = 123456789	The system shows the visualization that lessee is entering text.	As Expected	Pass						
5	Lessors click on the send code button.		The system shows the message that code has been sent.	As Expected	Pass						

6	Lessors enter the OTP code.	Code = 1234	The system shows the visualization that lessor entering the code. verify the code and show message phone number verified.	As Expected	Pass	
4	Lessors click on the next button.		The system redirects to the other car listing screen for more information.	As Expected	Pass	
5	Lessors enter more details, driving license, license country, name, and date.	License = 123455, Country = Pakistan, Name = Faizan, Date = 12/02/2010	The system shows the visualization that lessor filling the form.	As Expected	Pass	
6	Lessors click on the next button.		The system redirects to the other car listing screen for more information.	As Expected	Pass	
7	Lessors add advance notice time, minimum trip time and maximum trip time.	Notice = 2 days, Minimum = 3 days= Maximum = 30 days	The system shows the visualization that lessor filling the form.	As Expected	Pass	
8	Lessors enter the car description, car features, car license number and pictures.	License = 12344567, Description = abc, Features = air bags,	The system shows the visualization that lessor filling the form.	As Expected	Pass	

9	Lessors click on the next button.		The system redirects to the other car listing screen for more information.	As Expected	Pass	
10	Lessors click on the check box.		The system shows the visualization that lessor clicked on the check box and shows the message you have accept the term and condition.	As Expected	Pass	
11	Lessors click on the publish button.		The system shows the message that your car has been listed successfully.	As Expected	Pass	
Post-Conditions:						
1. Lessor can see his/her car details in Listing car screen.						

Table 34: Test case for List your car

4.1.10 Payment Withdraw Request

Test Case #:	10	Test Case Name:	Payment Withdraw Request
System:	CarSoul	Subsystem:	Payment Withdraw screen
Designed By:	Syed Faizan Ali	Design Date:	25/07/2020
Executed By:	Hammad Awan	Execute Date:	14/12/2020

Short Description:		This Test case tests the Payment Withdraw Request functionality. Payment Withdraw request use case allow the lessor to send payment withdraw request to CarSoul, which he/she has earned from his/her car.									
Pre-Conditions											
1 Lessor must be logged In.											
Step:	Action	Input Data	Expected System Response	Actual Response	Pass/ Fail	Comment					
1	Lessors access the payment withdraw screen.		The system shows the payment withdraw screen.	As Expected	Pass						
2	Lessors click on the withdraw button.		The system redirects to the payment withdraw request screen.	As Expected	Pass						
3	Lessors enter amount.	Amount = 20\$	The system shows the visualization that lessor entering amount.	As Expected	Pass						
4	Lessors enter IBAN number, CNIC, Name, date of birth, postal code, and address	IBAN= 122345, CNIC= 35202021238896, DOB= 15/12/1997 Code = 12345, Address = 123 Street ABC Town Lahore	The system shows the visualization that lessor is filling form.	As Expected	Pass						
5	Lessors click on the check box to accept the terms and conditions.		The system shows the visualization that lessor clicking on the check box and show message you have accepted the terms and conditions.	As Expected	Pass						

5	Lessees click on the send request button.		The system shows the message that your request has been submitted successfully.	As Expected	Pass	
Post-Conditions:						
1. Lessor can see his/her withdraw request in the account menu.						

Table 35: Test case for Payment withdraw request

4.1.11 View Posted Cars

Test Case #:	11	Test Case Name:	View posted cars
System:	CarSoul	Subsystem:	Listing cars screen. Database
Designed By:	Syed Faizan Ali	Design Date:	25/07/2020
Executed By:	Hammad Awan	Execute Date:	14/12/2020
Short Description:	This Test case tests the View Posted Cars functionality. View Posted Cars allow the Lessor to view the details of the cars that he/she has posted or listed on the application.		

Pre-Conditions

- 1 Lessor must be logged In and
- 2 Lessor must has posted at least one car

Step:	Action	Input Data	Expected System Response	Actual Response	Pass/ Fail	Comment
1	Lessors access the Listing Car screen.		The system shows the Listing car screen.	As Expected	Pass	
2	Lessors click on view listed cars button.		The system shows the listed cars cards.	As Expected	Pass	

Post-Conditions:

1. Lessor can see his/her withdraw request in the account menu.

Table 36: Test case for View posted cars

4.1.12 Track Your Car

Test Case #:	12	Test Case Name:	Track Your Car
System:	CarSoul	Subsystem:	Track your car and listing car screen, database
Designed By:	Syed Faizan Ali	Design Date:	25/07/2020
Executed By:	Hammad Awan	Execute Date:	15/12/2020
Short Description:	This Test case test the Track Your Car functionality. Track Your Car allow the lessor to see his/her car location live on map. Lessor will click on the car location button and can see car icon on map.		

Pre-Conditions

1 Lessor must be logged In.

2 Lessor must have 20\$ in the account.

Step:	Action	Input Data	Expected System Response	Actual Response	Pass/ Fail	Comment
1	Lessors access the listing car screen.		The system shows the Listing car screen.	As Expected	Pass	
2	Lessors click on the Track button on the car cad.		The system redirects to the Track your car screen and show the map with car location.	As Expected	Pass	

Post-Conditions:

1. Lessor can see car icon on the map.

Table 37: Test case for track your car

4.1.13 Deactivate Car Posting

Test Case #:	13	Test Case Name:	Deactivate Car Posting													
System:	CarSoul	Subsystem:	Listing car screen, database													
Designed By:	Hammad Awan	Design Date:	25/07/2020													
Executed By:	Syed Faizan Ali	Execute Date:	16/12/2020													
Short Description:	This Test case test the Deactivate Car posting functionality. Deactivate Car Posting allow the Lessor to change the status of a posted car enable to disable. So, Lessee cannot view that car details.															
Pre-Conditions																
1 Lessor must be logged In and 2 Lessor must has posted car details.																
Step:	Action	Input Data	Expected System Response	Actual Response	Pass/ Fail	Comment										
1	Lessors access the listing car screen.		The system shows the Listing car screen.	As Expected	Pass											
2	Lessors click on the Deactivate button on the car card.		The system shows the confirmation dialog box.	As Expected	Pass											
3	Lessors click on the confirm button.		The system shows the message that your car has been deactivated.	As Expected	Pass											
Post-Conditions:																
1. Lessor can view the activate button on car details.																

Table 38: Test case for Deactivate car posting

4.1.14 View Trip History

Test Case #:	14	Test Case Name:	View Trip History
System:	CarSoul	Subsystem:	Trip and Trip history screen, database
Designed By:	Hammad Awan	Design Date:	25/07/2020
Executed By:	Syed Faizan Ali	Execute Date:	16/12/2020
Short Description:	This Test case test the View Trip History functionality. View Trip History allow the lessor to see all details of the trips of his/her cars in the past.		

Pre-Conditions

- 1 Lessor must be logged In and
- 2 Lessor must has posted car details.

Step:	Action	Input Data	Expected System Response	Actual Response	Pass/ Fail	Comment
1	Lessors access the Trip screen.		The system shows the Trip screen.	As Expected	Pass	
2	Lessors click on the trip history button.		The system redirects to the trip history screen and shows all previous trips.	As Expected	Pass	

Post-Conditions:

1. The system will not show trips details if he/she has no trip in the past.

Table 39: Test case for View trip history

4.1.15 View Active Trips

Test Case #:	15	Test Case Name:	View Active Trips										
System:	CarSoul	Subsystem:	Trip and Active Trip screen, database										
Designed By:	Hammad Awan	Design Date:	25/07/2020										
Executed By:	Syed Faizan Ali	Execute Date:	16/12/2020										
Short Description:	View Active Trips allow the lessor/Lessee to view all the current trips of his/her cars. Lessor can see the time left for the trip and other details. This Test case test the View Active Trips functionality.												
Pre-Conditions													
1 Lessor/Lessee must be logged In. 2 Lessor/Lessee has at least one of his/her car of trip.													
Step:	Action	Input Data	Expected System Response	Actual Response	Pass/ Fail	Comment							
1	Lessors/Lessees access the Trip screen.		The system shows the Trip screen.	As Expected	Pass								
2	Lessors/Lessees click on the Active Trips button.		The system redirects to the Active Trip screen and shows all currently active trips.	As Expected	Pass								
Post-Conditions:													
1. Lessor can view the details of the active trip.													

Table 40: Test case for View active trips

4.1.16 Messages

Test Case #:	16	Test Case Name:	Messages
System:	CarSoul	Subsystem:	Message screen, database

Designed By:	Hammad Awan	Design Date:	25/07/2020			
Executed By:	Syed Faizan Ali	Execute Date:	17/12/2020			
Short Description:	Messages allow lessee and lessor to have a conversation and discuss further details for the order. For example, pickup point and drop-off points. This Test case test the messages functionality.					
Pre-Conditions						
1 The order was made by the lessee and a new conversation thread was created in the system.						
Step:	Action	Input Data	Expected System Response	Actual Response	Pass/ Fail	Comment
1	Lessors/Lessees access the Message screen.		The system shows the messages screen.	As Expected	Pass	
2	Lessors/Lessees enter text message.		The system shows the textual visualization that user entering the text.	As Expected	Pass	
3	Lessors/Lessees click on the send button.		The system shows a tick mark on the message.	As Expected	Pass	
Post-Conditions:				1. Lessor can view the details of the active trip.		

Table 41: Test case for Messages

4.1.17 View Profile

Test Case #:	17	Test Case Name:	View Profile
System:	CarSoul	Subsystem:	Home and Profile screen, Database

Designed By:	Hammad Awan	Design Date:	25/07/2020			
Executed By:	Syed Faizan Ali	Execute Date:	17/12/2020			
Short Description:	This Test case test the View Profile functionality. View profile allow lessee and lessor to see details of his profile.					
Pre-Conditions						
1 The information of lessee/lessor was previously saved in the system and the lessee/lessor wants to view it.						
Step:	Action	Input Data	Expected System Response	Actual Response	Pass/ Fail	Comment
1	Lessors/Lessees access the Home screen.		The system shows the Home screen.	As Expected	Pass	
2	Lessors/Lessees click on the Profile Button.		The system redirects to the Profile screen and show the details of the profile.	As Expected	Pass	
Post-Conditions:						
1. Profile details will be displayed.						

Table 42: Test case for View profile

4.1.18 Complete Profile

Test Case #:	17	Test Case Name:	Complete Profile
System:	CarSoul	Subsystem:	Profile and complete Profile screen, Database
Designed By:	Hammad Awan	Design Date:	25/07/2020
Executed By:	Syed Faizan Ali	Execute Date:	18/12/2020
Short Description:	This Test case test the Complete Profile functionality. Complete profile allows the lessee/lessor to enter and save the information which was left in the signup.		

Pre-Conditions

1 The profile was created by the lessee/lessor and the information was saved in the system. He/she wants to store the information which was not stored before.

Step:	Action	Input Data	Expected System Response	Actual Response	Pass/ Fail	Comment
1	Lessors/Lessees access the Profile screen.		The system shows the Profile screen.	As Expected	Pass	
2	Lessors/Lessees click on the Complete Profile Button.		The system redirects to the complete Profile.	As Expected	Pass	
3	Lessor/Lessee upload his/her profile picture.	Upload = picture	The system shows the dialog box to select the picture.	As Expected	Pass	
4	Lessor/Lessee add Date of birth, phone number and select payment method.	Date = 12/12/1997, Number = 123456, Payment = cash	The system shows the visualization the user filling the form.	As Expected	Pass	
5	Lessor/lessee click on the send code button.		The system shows the message that code has been sent.	As Expected	Pass	
6	Lessor/Lessee enter the OTP Code.	Code = 1234	The system shows the textual visualization that user entering text.	As Expected	Pass	

7	Lessor/lessee click on the verify button.		The system shows the message that code has been verified.	As Expected	Pass	
8	Lessor/Lessee click on the save button		The system shows the message that information has been saved.	As Expected	Pass	
Post-Conditions:						
1. The profile information will be added in the database.						

Table 43: Test case for Complete profile

4.1.19 Edit Profile

Test Case #:	17	Test Case Name:	Edit Profile
System:	CarSoul	Subsystem:	Profile and Edit Profile screen, Database
Designed By:	Hammad Awan	Design Date:	25/07/2020
Executed By:	Syed Faizan Ali	Execute Date:	18/12/2020
Short Description:	This Test case test the Edit Profile functionality. Edit profile allows lessee/lessor to edit information which he/she stored before.		

Pre-Conditions

- 1 The information was previously saved in the system and the lessee/lessor wants to change it now.

Step:	Action	Input Data	Expected System Response	Actual Response	Pass/ Fail	Comment
1	Lessors/Lessees access the Profile screen.		The system shows the Profile screen.	As Expected	Pass	

2	Lessors/Lessees click on the Edit Profile Button.		The system redirects to the Edit Profile.	As Expected	Pass	
3	Lessor/Lessee upload his/her profile picture.	Upload = picture	The system shows the dialog box to select the picture.	As Expected	Pass	
4	Lessor/Lessee add Date of birth, phone number and select payment method.	Date = 12/12/1997, Number = 123456, Payment = cash	The system shows the visualization the user filling the form.	As Expected	Pass	
5	Lessor/lessee click on the send code button.		The system shows the message that code has been sent.	As Expected	Pass	
6	Lessor/Lessee enter the OTP Code.	Code = 1234	The system shows the textual visualization that user entering text.	As Expected	Pass	
7	Lessor/lessee click on the verify button.		The system shows the message that code has been verified.	As Expected	Pass	
8	Lessor/Lessee click on the save button		The system shows the message that information has been saved.	As Expected	Pass	

Post-Conditions:

1. The profile information will be updated in the database.

*Table 44: Test case for Edit profile***4.1.20 View Account Details**

Test Case #:	17	Test Case Name:	View Account Details
System:	CarSoul	Subsystem:	Account screen, Database
Designed By:	Hammad Awan	Design Date:	25/07/2020
Executed By:	Syed Faizan Ali	Execute Date:	19/12/2020
Short Description:	This Test case test the View Account Details functionality. View Account Details allow the Lessor to view all account details.		

Pre-Conditions

- 1 Lessor must be logged In..

Step:	Action	Input Data	Expected System Response	Actual Response	Pass/ Fail	Comment
1	Lessors/Lessees access the Home screen.		The system shows the Home screen.	As Expected	Pass	
2	Lessors/Lessees click on the Account Button.		The system redirects to the Account screen and show the details of the Account.	As Expected	Pass	

Post-Conditions:

1. Lessor can view the details of the account.

Table 45: Test case for View account details

4.1.21 Signup

Test Case #:	17	Test Case Name:	Signup
System:	CarSoul	Subsystem:	Signup screen, Database
Designed By:	Hammad Awan	Design Date:	25/07/2020
Executed By:	Syed Faizan Ali	Execute Date:	19/12/2020
Short Description:	Sign up allow the Lessor/Lessee to create an account. Lessor/Lessee enter his/her email, number, and password to get register. This Test case test the Signup functionality.		

Pre-Conditions

1 Lessor/Lessee must have a valid email.

Step:	Action	Input Data	Expected System Response	Actual Response	Pass/ Fail	Comment
1	Lessors/Lessees access the Login screen.		The system shows the login screen.	As Expected	Pass	
2	Lessors/Lessees click on the Signup Button.		The system redirect to the Signup screen.	As Expected	Pass	
3	Lessor/lessee enter email, number, and password.	Email = hammad@gmail.com , Number = 12345678 Password = 1234abcd	The system shows textual visualization that user entering text.	As Expected	Pass	
4	Lessor/lessee click on the send code button.		The system shows the message that code has been sent.	As Expected	Pass	

5	Lessor/Lessee enter the OTP Code.	Code = 1234	The system shows the textual visualization that user entering text.	As Expected	Pass	
6	Lessor/lessee click on the verify button.		The system shows the message that code has been verified.	As Expected	Pass	
7	Lessor/Lessee click on the signup button		The system shows the message that signup successfully.	As Expected	Pass	
Post-Conditions:						
1. Lessor can view the details of the account.						

Table 46: Test case for Signup

4.1.22 Login

Test Case #:	022	Test Case Name:	Login
System:	CarSoul	Subsystem:	Login Screen, Database
Designed By:	Hammad Awan	Design Date:	25/07/2020
Executed By:	Syed Faizan Ali	Execute Date:	19/12/2020
Short Description:	This Test case test the Login functionality. Login allows lessee/lessor to create a new session in the application.		

Pre-Conditions

1. The email and password of lessee/lessor was stored in the database when he/she signed up.
2. The lessee/lessor is not already logged in.

Step:	Action	Input Data	Expected	Actual	Pass/	Comment
-------	--------	------------	----------	--------	-------	---------

			System Response	Response	Fail	
1	Lessee/Lessor access the login screen.		The system shows the login screen.	As Expected	Pass	
2	Lessee/Lessor enter valid email and password.	hammadawan@gmail.com 12345678	The system show visualization the user filing the form.	As Expected	Pass	
3	The lessee/lessor presses the login button.		The system redirect to the Home screen and show message logged in successfully.	As Expected	Pass	
4	Repeat the steps 1, 2 and enter invalid email and password.	hammadawan@mail.com 1234567	The system show login fail message and redirect to login screen again.	As Expected	Pass	
Post-Conditions:						
<ol style="list-style-type: none"> 1. Redirect to the Home screen. 2. Show the error message if invalid credentials 						

Table 47: Test case for Login

4.1.23 Logout

Test Case #:	17	Test Case Name:	Logout
System:	CarSoul	Subsystem:	Home and Login screen, Database
Designed By:	Hammad Awan	Design Date:	25/07/2020
Executed By:	Syed Faizan Ali	Execute Date:	20/12/2020
Short Description:	This Test case test the Logout functionality. Logout allows lessee/lessor to		

	destroy the session which was created at the time of login.											
Pre-Conditions												
1 The lessee/lessor is already logged in.												
Step:	Action	Input Data	Expected System Response	Actual Response	Pass/ Fail	Comment						
1	Lessors/Lessees access the Home screen.		The system shows the Home screen.	As Expected	Pass							
2	Lessors/Lessees click on the logout Button.		The system redirects to the Login screen.	As Expected	Pass							
Post-Conditions:												
1. The current user session will be destroyed.												

Table 48: Test case for Logout

4.1.24 Forgot Password

Test Case #:	024	Test Case Name:	Forgot password			
System:	CarSoul	Subsystem:	Login Screen, Database			
Designed By:	Hammad Awan	Design Date:	25/07/2020			
Executed By:	Syed Faizan Ali	Execute Date:	20/12/2020			
Short Description:	This Test case test the Forgot Password functionality. Forgot password allows the lessee/lessor to create a new password for his/her account.					
Pre-Conditions						
1. The email and password of lessee/lessor was stored in the database when he/she signed up. 2. The lessee/lessor is not already logged in.						
Step:	Action	Input Data	Expected System	Actual	Pass/	Comment

			Response	Response	Fail	
1	Lessee/Lessor access the login screen.		The system shows the login screen.	As Expected	Pass	
2	Lessee/Lessor click on the forgot password button.		The system shows a dialog to enter email.	As Expected	Pass	
3	The lessee/lessor enter the email.	hammadawan@gmail.com	The System show the textual visualization that lessee/lessor entering the text.	As Expected	Pass	
4	Lessee/Lessor click on the send code button.		The system show message has been sent and open a dialog box to enter OTP code.	As Expected	Pass	
5	Lessee/Lessor enter the OTP code and press verify button.	Code = 789809	The system redirects to the forgot password screen.	As Expected	Pass	
6	Lessee/Lessor enter the new password.	New password= 12345678	The System show the textual visualization that lessee/lessor entering the text.	As Expected	Pass	
7	Lessee/Lessor click on change password button,		The system shows message that password has been changed.	As Expected	Pass	
8	Repeat the step 1 to 5 with wrong OTP code	Wrong code = 786786	The system shows message that OTP is incorrect and redirect to the login screen.	As Expected	Pass	

Post-Conditions:

1. Lessee/Lessor can login again with new password.

Table 49: Test case for Forgot password

4.1.25 Cancel Order

Test Case #:	17	Test Case Name:	Cancel Order
System:	CarSoul	Subsystem:	Home and Login screen, Database
Designed By:	Hammad Awan	Design Date:	25/07/2020
Executed By:	Syed Faizan Ali	Execute Date:	20/12/2020
Short Description:	Cancel order allows lessee/lessor to cancel order that they made earlier. This Test case test the Cancel Order functionality.		

Pre-Conditions

- 1 The order was made earlier by the lessee and saved in the database.

Step:	Action	Input Data	Expected System Response	Actual Response	Pass/ Fail	Comment
1	Lessors/Lessees access the Active Trips screen.		The system shows the Active Trip screen.	As Expected	Pass	
2	Lessors/Lessees click on the Cancel Button.		The system shows the confirmation dialog box.	As Expected	Pass	
3	Lessor/Lessee click on the confirm button		The system shows the message that order has been canceled.	As Expected	Pass	

Post-Conditions:

1. The order will be deleted from database.

Table 50: Test case for Cancel order

4.2 Unit/ Integration/ Acceptance Testing:

4.2.1 Unit Testing:

Unit testing is one of the levels of software testing in which software is tested as individual units/components. The purpose of unit testing is to check that all the units of software are working as they were designed to do. There are many tools available in market to perform unit testing.

In this project, following units were tested using manual testing technique.

Login	Posting
Signup	Advance Filter
Forgot Password	Add to Favorite
Logout	Checkout
Edit Profile	Cancel order
View Profile	View Activity
Complete Profile	View Active Trips
List your car	View Trip History
View Posted Car	Messaging
Track Your Car	Feedback
Car Search	Report Car Listing
View Car Details	Payment withdraw request
Deactivate Car	

4.2.2 Integration Testing:

Integration testing is another level of software testing in which the individual units of the software are tested in a form of a group by linking them with each other. The purpose of integration testing is check whether the interaction between these units is causing any fault or not. The test drivers and test stubs are used in integration testing of any unit/component is not yet developed but is needed to perform the integration testing.

In our project, following units are going to be tested

- Profile unit.
- Car Search unit.
- Trip management unit.

4.2.3 System Testing:

System testing is a type of black box testing in which the software is tested as a whole. All of its units and components are well integrated, and the system is complete. The purpose of performing system testing is to check whether the system is meeting its specified requirements or not. It helps testers to identify how much the software is satisfying its consumer needs before its release.

4.2.4 Acceptance Testing:

Acceptance testing as the name implies is used to test the software whether it is acceptable for delivery or not. The purpose of acceptance testing is to check if the system is compliant to our business requirements. There are different types of acceptance testing. In our project we aim to perform beta testing on our product.

5 Conclusion

5.1.1 Problem faced and Lessons learned:

We faced many problems in this project such as combining functional requirements of two applications/portals in one. This application is intended to be used by two different users (lessee and lessor) with completely different requirements. The main challenge was to entertain these two users in a way that the one user is not distracted by the features provided to the other user. Moreover, Covid-19 made this project difficult in many ways and restricted our collaboration, but we managed to overcome those issues and successfully complete this project.

In this project, we learned the core of Android. We learned to use to cloud database i.e. Firebase. This project also taught us how to gather requirements efficiently from a user perspective, implement them, and test them extensively to satisfy the user needs.

5.1.2 Project Summary:

Commuting from one place to another is the most common thing we humans do on daily basis. It is not a big deal unless the public transport system is good in your area. We all are aware of the public transport system here in Pakistan. The next option left is to buy a car and the way car prices are going up day by day, making it impossible for the middle-class man to own a car. We tried to solve this problem and provide a viable solution for commuting. Also, eliminating the monopoly of car industry and the hassle to rent a car at the same time.

5.1.1 Future Work:

We plan to develop an IOS and Web app for this project in future. Making it available to a larger audience. We also intend to convert this project into a product and launching it on the national level in Pakistan.

6 Prototype

6.1 Car Search:

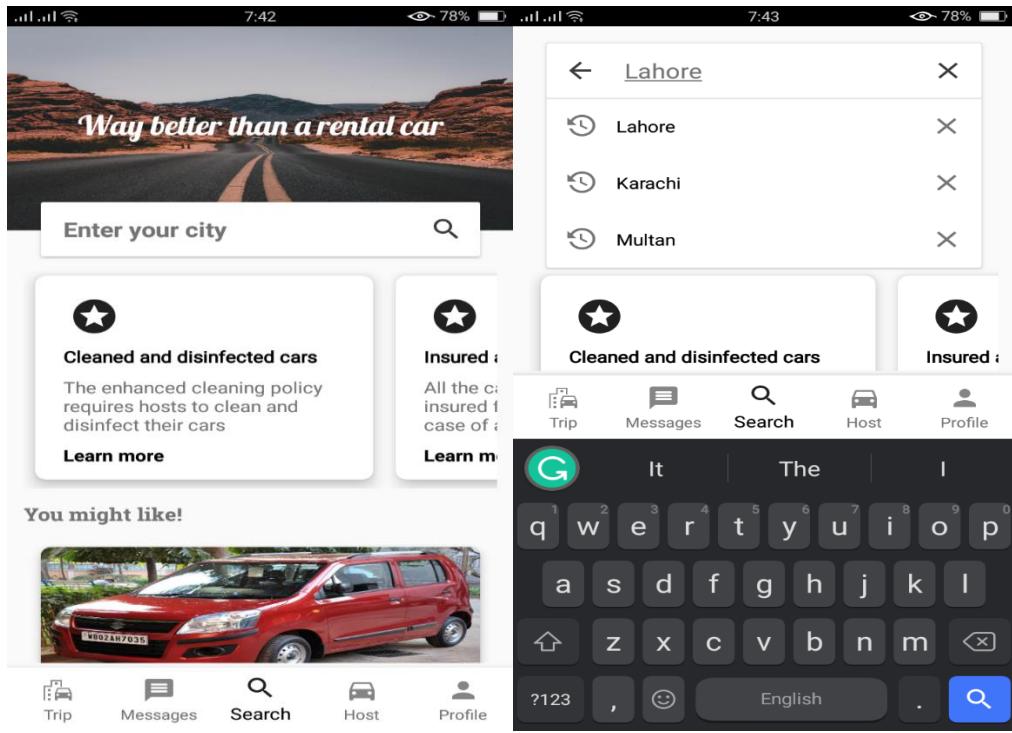


Figure 106: Screen shot Car Search

6.2 Advance Filter:

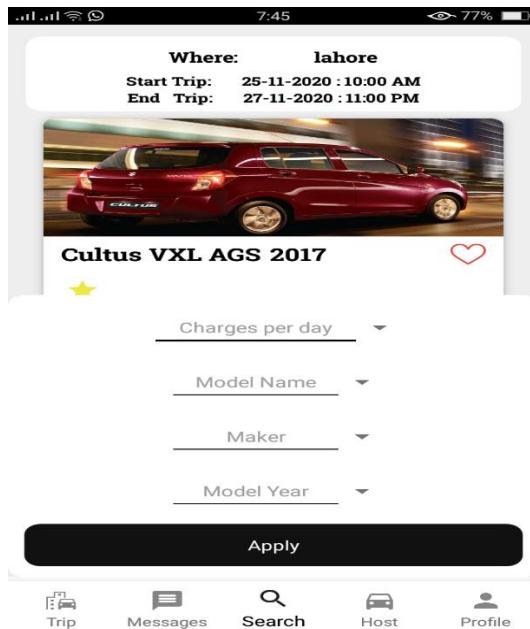


Figure 107: Screen shot Advance Search

6.3 View Car Details for Lessee:

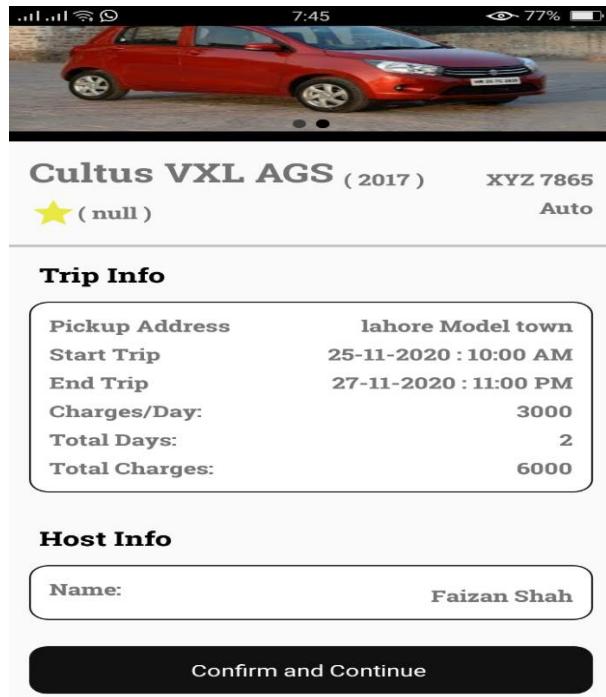


Figure 108: Screen shot Car Details Lessee

6.4 Report Car Listing:

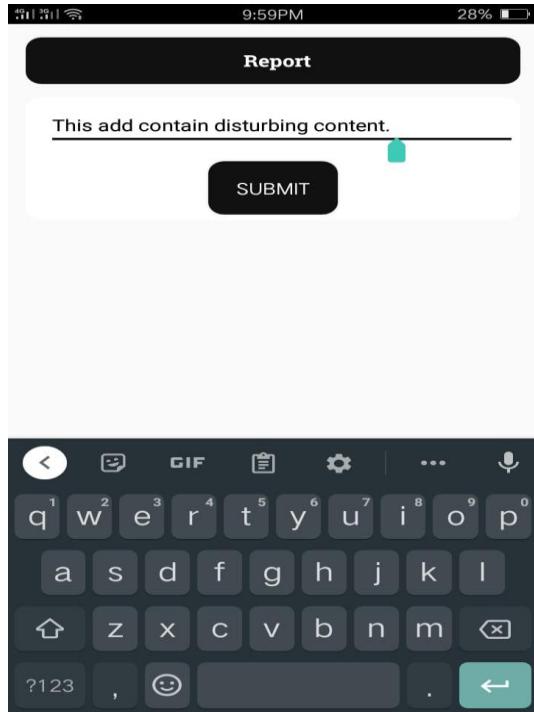


Figure 109: Screen shot Report Car Listing

6.5 Add to Favorite:



Figure 110: Screen shot Add to favorite

6.6 Feedback:

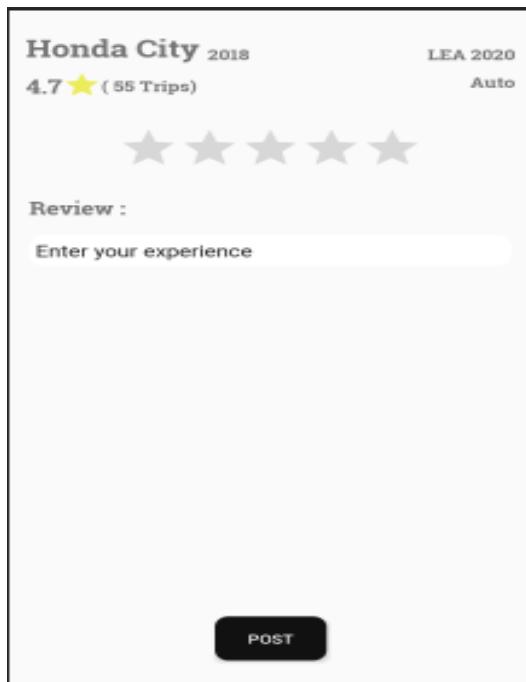


Figure 111: Screen shot Feedback

6.7 List Your Car:

Trip Business Time

- Advance Notice Duration (Days): 1
- Minimum Trip Duration (Days): 1
- Maximum Trip Duration (Days): 3
- Charges per Day (PKR): 4000

General Car Info!

- Maker: Honda
- Model Name: Civic
- Model Year: 2017
- Transmission: Auto

Driver Info!

- Driving License Issue Country: Pakistan
- Driving License Number: 4463285324489076
- Driver Full Name: Sheikh Faizan
- Date of Birth: 3-8-1993

Car Details

- Car License Number: LEO 5477
- Add Features +
- 1 Air Bags
- 2 Power Steering
- 3 Power Windows
- 4 Central Locking
- 5 Extra Wheel
- 6 Puncture equipment

Tell us about your Car:

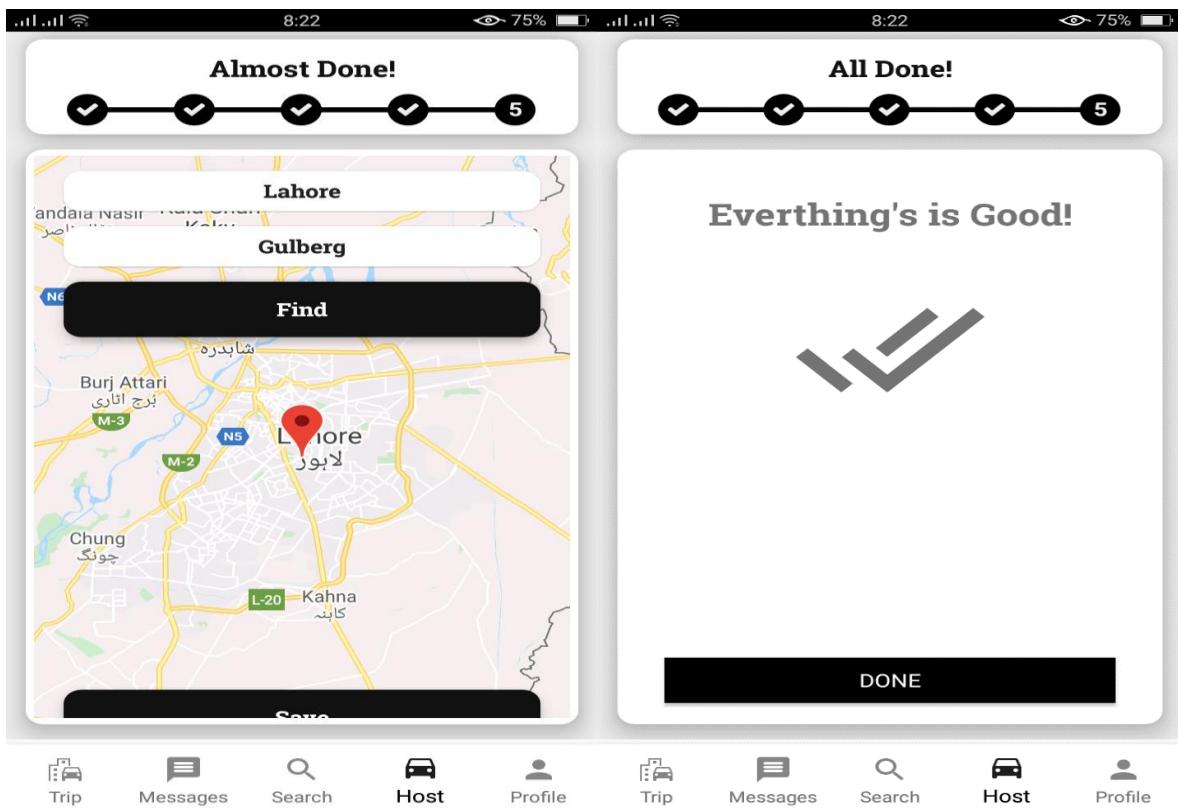


Figure 112: Screen shot List your car

6.8 Payment Withdraw Request:

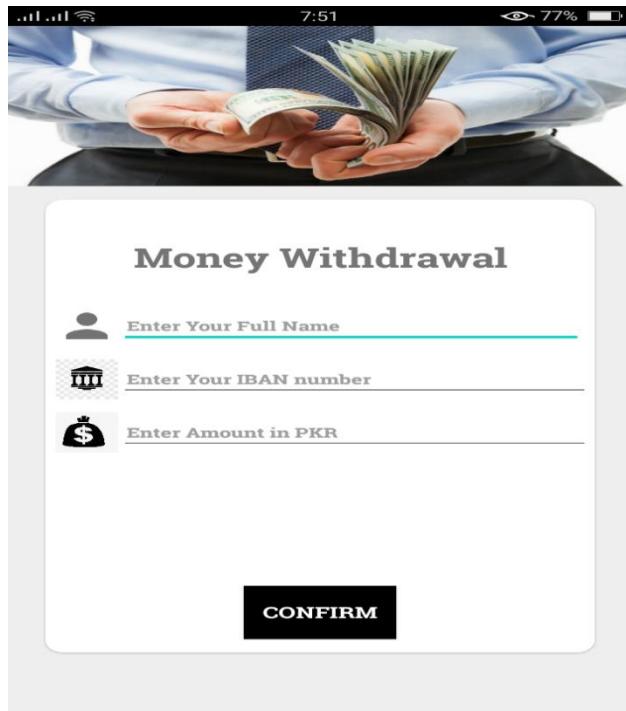


Figure 113: Payment withdraw request

6.9 View Posted Car:

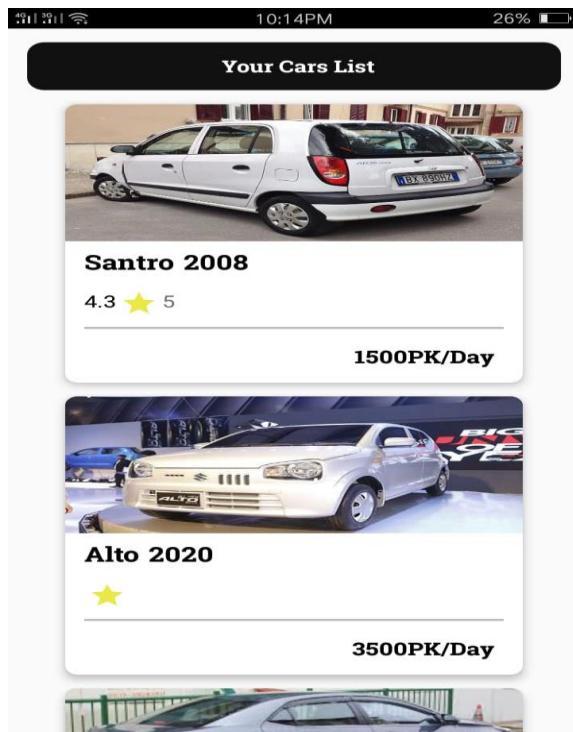


Figure 114: Screen shot View Posted Car

6.10 Track Your Car:

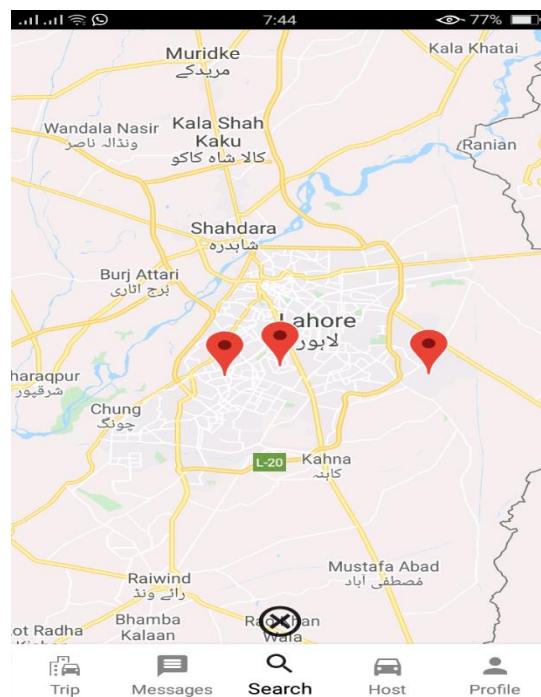


Figure 115: Screen shot Track your car

6.11 View Active Trips:

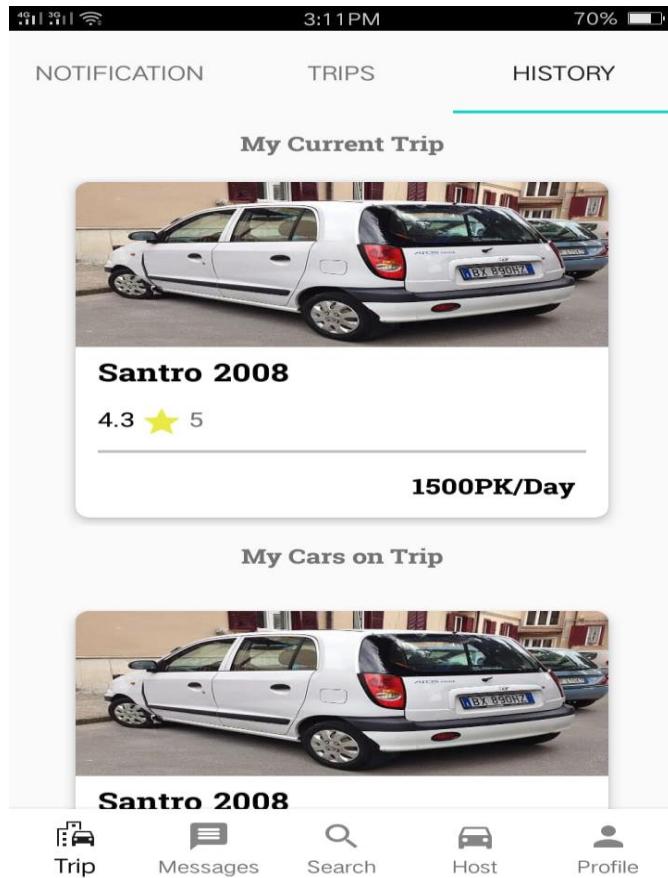


Figure 116: Screen shot View Active Trips

6.12 Messages:

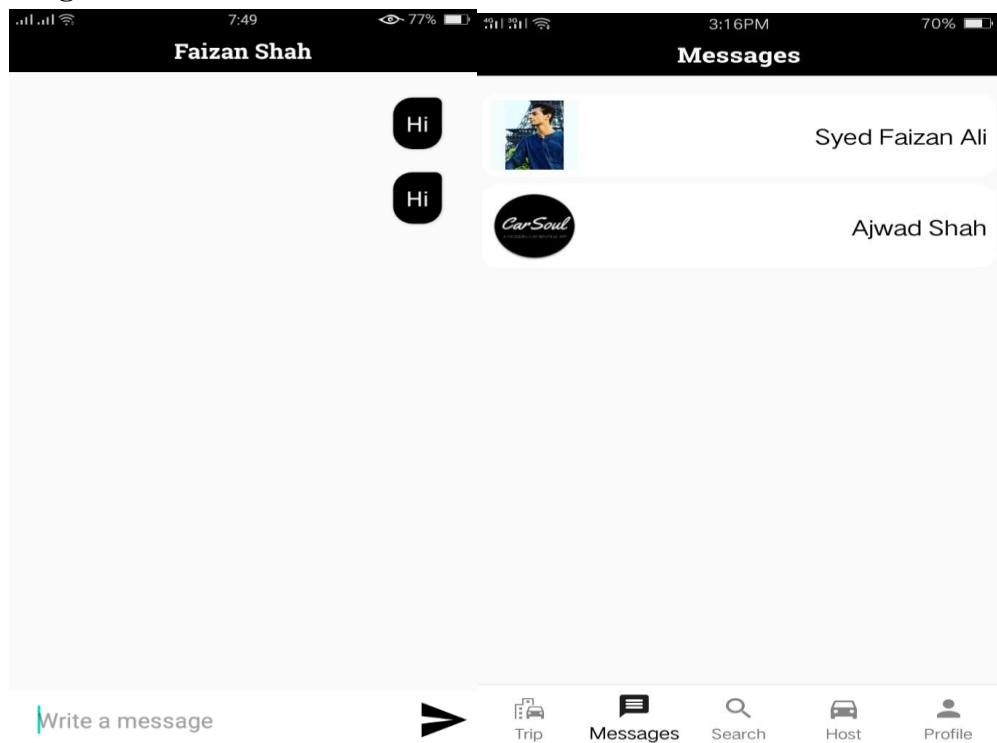


Figure 117: Screen shot Messages

6.13 View Profile:

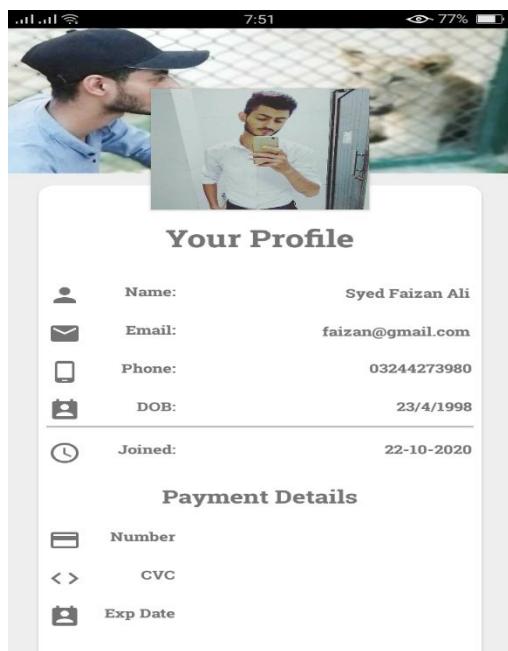


Figure 118: Screen shot View Profile

6.14 Complete Profile:

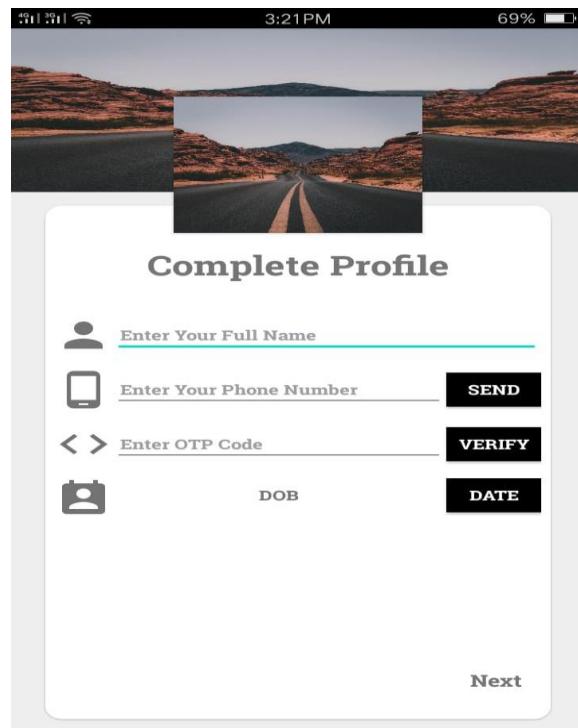


Figure 119: Screen shot Complete Profile

6.15 Edit Profile:

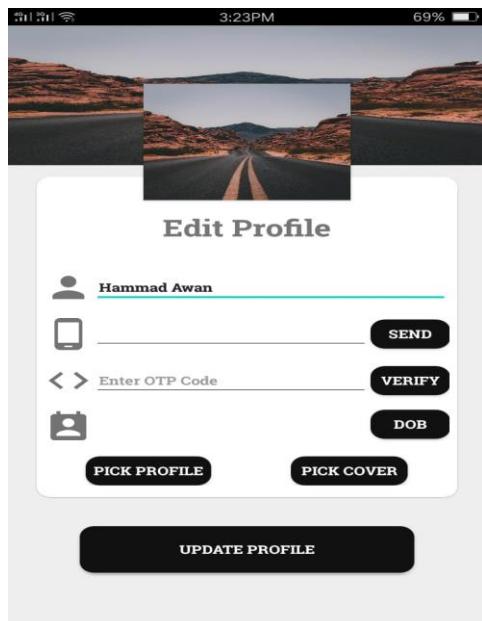


Figure 120: Screen shot Edit Profile

6.16 View Account Details:



Figure 121: Screen shot View Account Details

6.17 Signup:

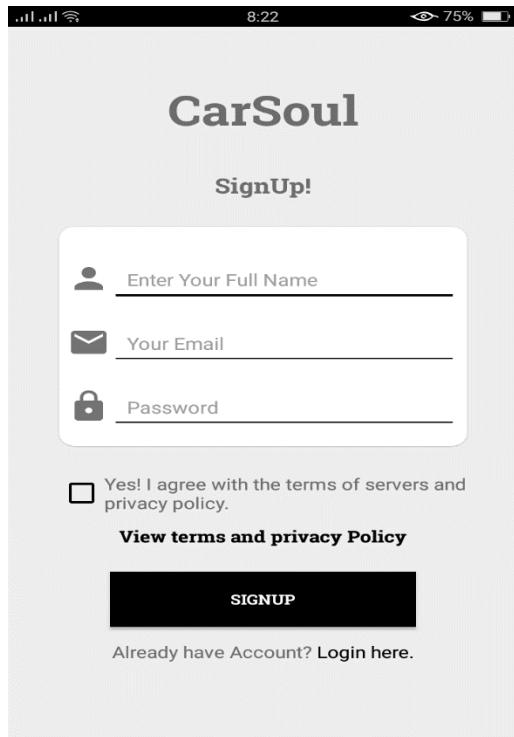


Figure 122: Screen shot Signup

6.18 Login:

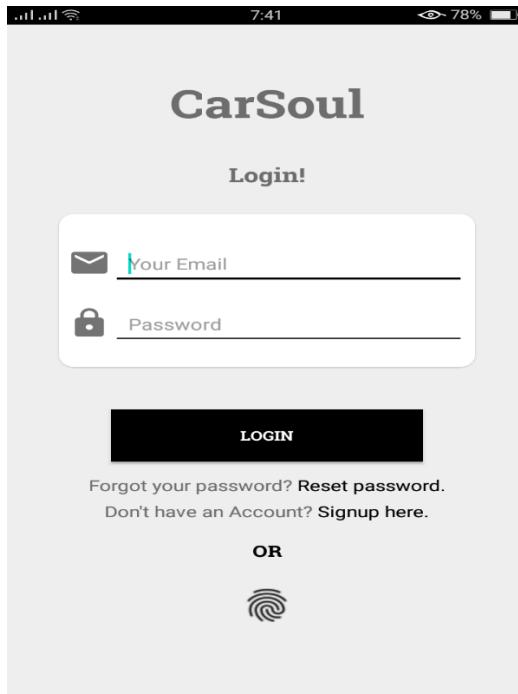


Figure 123: Screen shot Login

6.19 Forget Password:

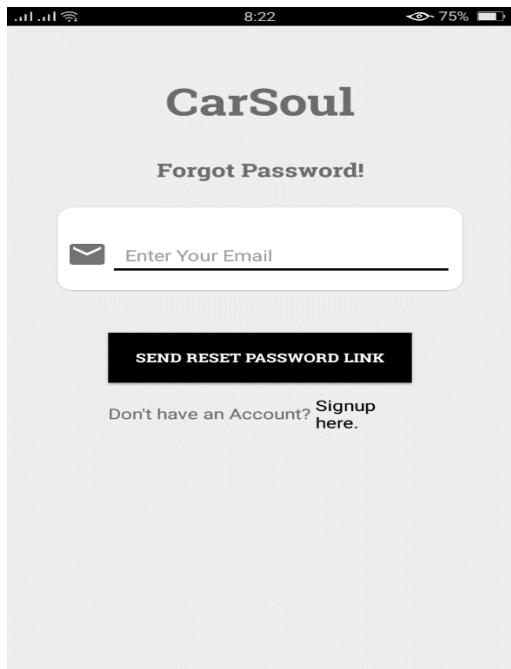


Figure 124: Screen shot Forgot password

6.20 Cancel Order:

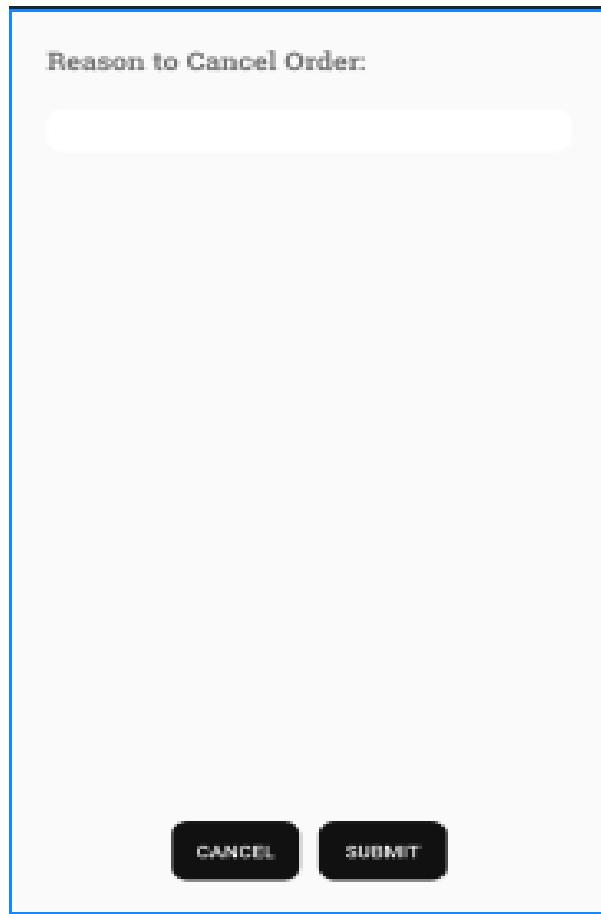


Figure 125: Screen shot Cancel order

7 References

- [1]. <https://rentmycar.pk/> (website)
- [2]. <https://www.zipcar.com/> (website)
- [3]. <https://www.zoomcar.com/> (website)
- [4]. <https://www.getaround.com/> (website)
- [5]. <https://turo.com/> (website)

8 Turnitin report

Submission date: 28-Dec-2020 03:45PM (UTC+0500)

Submission ID: 1481652698

File name: lication_connecting_lessor_with_lessee_on_the_same_platform.docx (6.43M)

Word count: 20182

Character count: 100491

SP20SE06_CarSoul A modern car renting application connecting lessor with lessee on the same platform

ORIGINALITY REPORT

2 %	2 %	1 %	%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	utpedia.utp.edu.my Internet Source	1 %
2	www.slideshare.net Internet Source	<1 %
3	www.coursehero.com Internet Source	<1 %
4	Preeti Gulia, Jyoti Chugh. "Comparative Analysis of Traditional and Object-Oriented Software Testing", ACM SIGSOFT Software Engineering Notes, 2015 Publication	<1 %

5	passenger-57.ctr.co.uk Internet Source	<1 %
6	www.qatouch.com Internet Source	<1 %
7	www.medicine.manchester.ac.uk Internet Source	<1 %
Chefira, Rakrak. "Enhancing Decision Making		
8	through Combined Classification Techniques and Probabilistic Data Analysis for Ubiquitous Healthcare Anomaly Monitoring", Applied Sciences, 2019 Publication	<1 %
9	www.alltechtricks.in Internet Source	<1 %
10	D.W. Bustard, Z. He, F.G. Wilkie. "Linking soft systems and use-case modelling through scenarios", Interacting with Computers, 2000 Publication	<1 %
11	hdl.handle.net Internet Source	<1 %
12	atozic.blogspot.com Internet Source	<1 %

13

www.ques10.com

Internet Source

<1 %
