

# UHire

## Peer to Peer Car Rental System

Prepared for: UOW School of Computing and Information Technology

Prepared By: Group 2

Name	Contribution	Signature
Tianming	Contributed	
Reece	Contributed	
Josh	Contributed	
Vu	Contributed	
Wenjuan	Contributed	

# Table of Contents

<b>Table of Contents</b>	<b>2</b>
<b>User Stories</b>	<b>3</b>
<b>Prototypes</b>	<b>10</b>
Database Design	10
User Interface	11
<b>Taiga Management</b>	<b>16</b>
Previous Documentation	16
Sprint Plans	18
<b>Meeting Records</b>	<b>26</b>
<b>Analysis &amp; Design</b>	<b>46</b>
Use Case Diagram	46
Use Case Descriptions	47
Class Diagram	61
Sequence Diagrams	63
<b>Testing</b>	<b>72</b>
Test Cases	72

## User Stories

*Manage memberships including types of users, user profiles, identity verification, and credit check, and their reviews/ratings.*

**#1. As a user, I want to sign up to the website, so that I can have an account to login into the system with an account number and password. (C)(R)(V)**

- Decide fields required for registration
- Design register page UI
- Implement register page UI
- Implement and connect register to backend
- Test register page

**#2. As a user, I want to log in the system via account number and password so that I can rent cars in the system (be a car renter). (C) (R)(V)**

- Design login page UI
- Implement login page UI
- Connect login page to backend
- Test login page

**#3. As a user (either a car owner or a car renter), I want to register my credit card's information to the system so that the system can use it to perform transactions.**

- Design credit card page UI
- Implement credit card page UI
- Implement and connect the page to backend
- Test credit card page

**#4. As a user, I want to keep and modify all my information on a profile, including my personal details so that I don't have to introduce myself every time a user questions me and I can check my system credits easily. (C) (R) (V)**

- Design profile UI
- Implement profile UI
- Test profile UI by display all user details, including rating
- Implement user profile creation code
- Implement modify profile
- Display users listings
- Test profile page

**#5. As a user, I would like to view other users' profiles to gather more information about the user and gain confidence about conducting a transaction with them. (E)**

- Design profile in read-only mode, for when viewing another profile
- Implement the alternative profile UI
- Connect the read-only profile UI to appropriate back-end code.
- Test the read-only profile, including listings

**#6. As a car owner, I want to have a verification on my identity, my credit history checked and my cars verified so that all other users can trust me. (C) (R) (E)**

- Design upload identity and upload credit (include car registration) UI
- Implement upload identity and verify credit UI
- Implement the verification back-end code
- Implement communication with verification server
- Test the verification process

**#7. As a user, I want to be able to check everyone's (including mine) reviews/ratings so that I know who to trust and what to pay more attention to. (C) (R) (V)**

- Design UI elements so that the reviews and ratings can be displayed on the profile. (Both for my profile and read-only mod profile)
- Add UI elements in the front-end code.
- Test the review/ratings section.

**#8. As a renter, I want to give a review and rating to the car owner after the renting is finalised so that other people can all read the feedbacks to the car owner and help other people to make better decisions. (C) (R) (V)**

- Design feedback box
- Implement Feedback UI and connect it to the booking list pages (It appears once a renting is finalized)
- Connect feedback UI to backend
- Test feedback UI

***Manage car listings (e.g. description of cars such as year, model, transmission, and odometer and available dates, etc.) for car owners.***

**#9. As a car owner, I want to add a new car to my car listing as well as all its description (e.g. year, model, transmission, odometer and available dates etc.) so that the new car information is posted and potential renter can see my new post. (C) (R) (V)**

- Decide what data to store for each car
- Design a UI for managing personal car listings
- Implement UI
- Implement add a car to the listing
- Test the adding functionality.

**#10. As a car owner, I want to view my current car listing so that I can update information about my cars. (C) (R) (V)**

- Design UI elements to display cars in a list.
- Add update function to the code.
- Implement the UI elements.
- Connect to the back-end code.
- Test the car listings.

**#11. As a car owner, I want to delete a posted car in my car listing so that I can make the car unavailable. (C) (R) (V)**

- Add a UI element to the indicating car deletion.
- Implement deletion functionality
- Test the car deletion functionality.

***Manage car searching (based on location, car preferences, etc.) and booking requests.***

**#12. As a renter, I want to view all the cars available based on search criteria such as location and car preference etc. so that I can find the best options. (C) (R) (V)**

- Decide available criteria to browse by
- Design car listing to display
- Implement GUI
- Code the search box
- Connect to the back-end code
- Test the search functionality

**#13. As a renter, I want to book requests for cars that I intend to rent and make the payment (rental fees and the bonds) to the system. (C) (R) (E)**

(This user story is related to rejecting booking and finalizing renting)

- Implement UI elements that can let renter request booking when viewing car details
- Implement the back-end code (so that it extracts money from the renter's bank card and hold it in the system)
- Connect the UI elements to the back-end code
- Test the request booking process

**#14. As a car owner, I want to see all the booking requests for my cars so that I can manage them. (V)**

- Design booking requests manager GUI (should display a list of current booking and renting)
- Implement Booking requests manager GUI
- Implement view booking requests
- Test the request manager.

**#15. As a car owner, I want to confirm or reject booking requests so that I can decide which renters to have my car. (V)**

- Implement confirm tickets
- Implement reject tickets
- Test the confirmation process
- Test the rejection process (including the money goes back to renter's account.)

**#16. As a user, I want to finalize a renting so that I confirm a success of the whole renting and the rental fee held in the system can go to the car owner's account and the bonds can be returned to my account. (E)**

- Implement UI elements to finalize a renting
- Implement back-end code so that when the renting is finalized by car owner and renter, the rental fees goes to car owner's account and bond is returned to renter's account.
- Test the functionality of finalizing renting process

*Manage private messaging channel between users before booking and accepting reservations (e.g. for questions, handover time and location, etc.).*

**#17. As a user, I want to message car owners/renters before accepting/creating booking requests in a private messaging channel so that I can know more details like handover time, location and discuss any concerns. (C) (R) (V)**

- Design a message system GUI
- Implement message system GUI
- Connect message system with database query system
- Implement chat features send message
- Test the message system

**#18. As a user, I want to see all my current messaging in a list so that I can continue the messaging with someone. (C) (R) (V)**

- Design a message system history page GUI
- Implement message history GUI
- Implement display message history
- Test the functionality

*Manage payment transactions (including bond, rental fees, etc.) between car owners and renters, and between the system's owner and car owners.*

**#19. As a renter, I want to make transaction within the system through within the system so that I can pay the car owner easily. (C) (R) (E)**

- Implement UI elements for users to include their linked bank accounts and bank cards for sending and receiving money in profile.
- Implement the back-end code (update user's bank information accordingly and link users to their bank accounts or bank card)
- Connect the UI elements to the back-end code
- Test the functionality

**#20. As a potential renter, I want the rental fees to be held in the system until the renting is finalised so that the rental fee is kept securely during my renting. (C) (R) (V)**

- Design a transaction subsystem that holds all money from renters (money goes from renter's bank card)
- Connect the subsystem to 'Request booking' button
- Test the functionality

**#21. As a car owner, I want to receive the rental fees after the renter has finished renting so that the whole process is finalised.**

**#22. As a car owner, I want renters to provide a bond within the system so that if they cause damages to the car, they need to pay for this. (C) (R) (V)**

-Implement a automatic process for a car renter to pay the bond when they request a booking

-Test the functionality

**#23. As the system owner, I want payments to be made as a commission, direct to my account. (C) (E)**

-Implement UI elements such that the system admin can update the commission rate in terms and conditions

-Implement the back-end so that each transaction consists of a percentage commission fee that will go to the system owner's account automatically when the renting is finalized by both parties (renter and car owner)

-Test the functionality

**#24. As the system owner, I want to update my bank account via a system admin so that all commision fees can go to the updated bank account. (C) (E)**

-Implement UI elements such that the system admin can update the system owner's bank account in terms and conditions

-Link the system owner's bank account to each transactions between car renters and car owners.

-Test the functionality

Manage links to social network accounts (e.g. Facebook).

**#25. As a user, I want to link my account to my social network accounts, so that people can find more information on my social network account (e.g. Facebook). (C) (R) (V)**

-Implement a widget to display a social media link in view self profile

-Implement modification of social media link in my profile

-Implement a widget to display a social media link in view other profile

-Test if all the widget works functionally

**#26. As a user, I want to sign up via my social network accounts so that I do not have to fill in my personal details manually. (C) (R) (V)**

- Design GUI widget to be added to registration page
- Implement GUI widget to be added to registration page
- Integrate registration with common social media accounts

-Test if a user can sign up purely via social network account

*Other.*

**#27. As a user, I want to view the guidelines of the website so that I know all the policies and terms and conditions. (C) (R) (V)**

- Design a guidelines page GUI
- Implement a guidelines page GUI
- Test if guidelines can be displayed properly

**#28. As a system admin, I want to be upload the latest version of policy and terms and conditions, and guidelines. (C) (R) (V)**

- Design administration settings page GUI
- Implement administration setting pages GUI
- Implement policy modifier
- Implement terms and conditions modifier
- Implement guidelines modifier

-Test the functionality

**#29. As a system admin, I want to delete users so that I can clean up those bad-behaviour users. (C) (R) (V)**

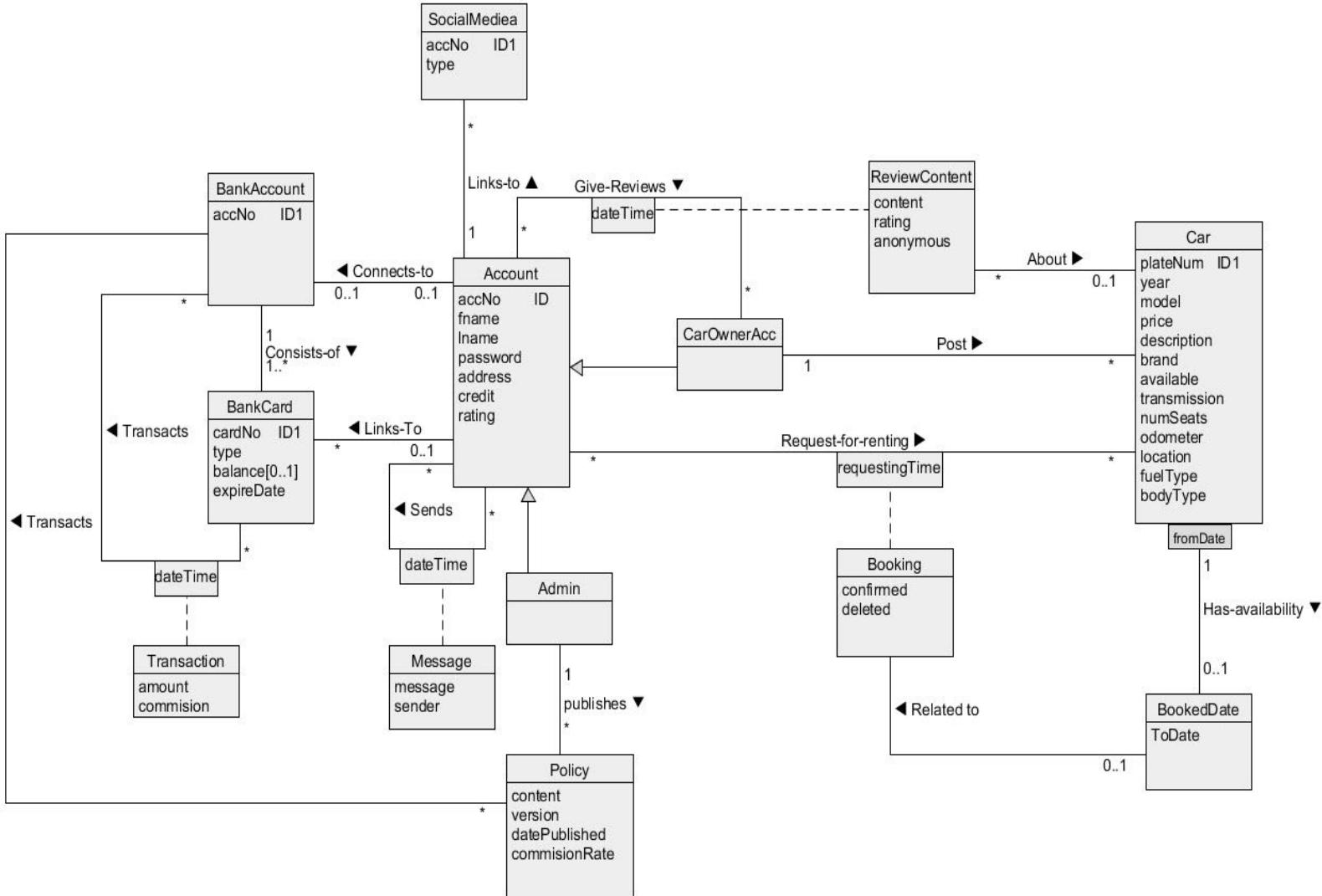
- Add a delete button widget on profile only available to admins
- Implement delete button

-Connect the button to the back-end code and databases

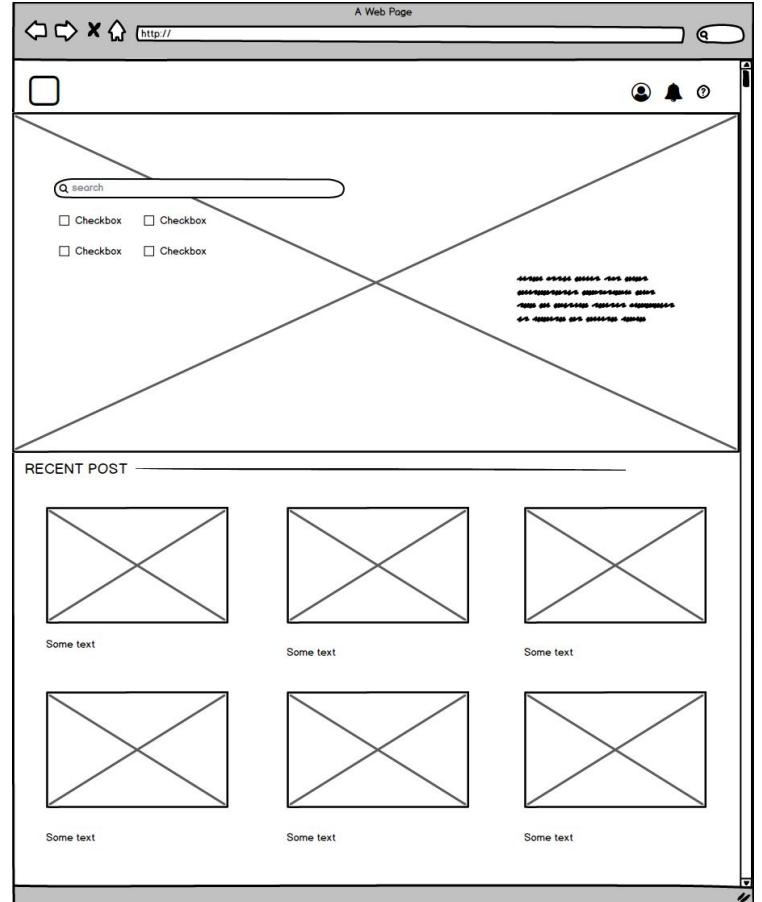
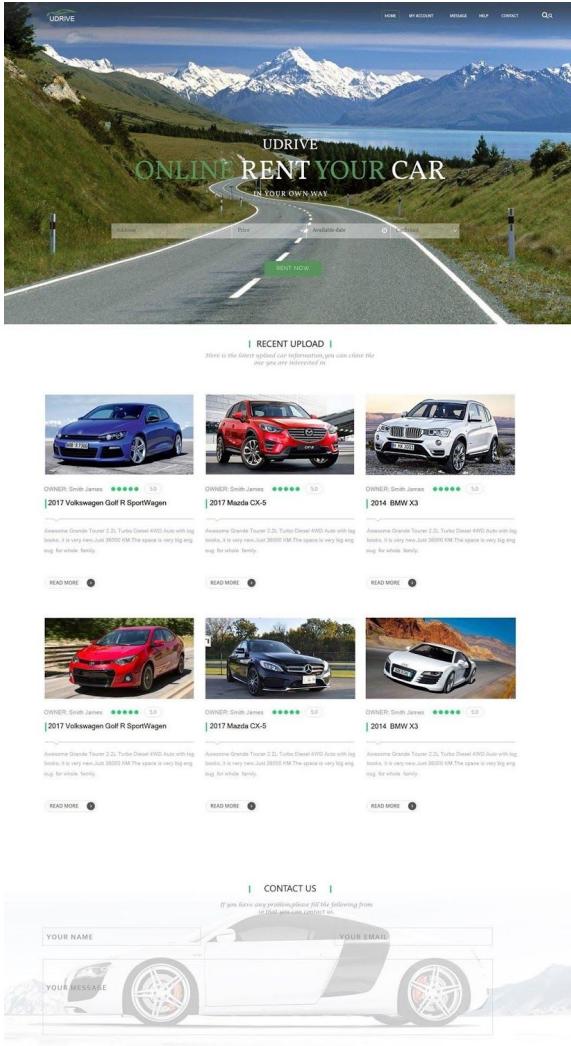
-Test the deletion functionality

# Prototypes

## Database Design



## User Interface



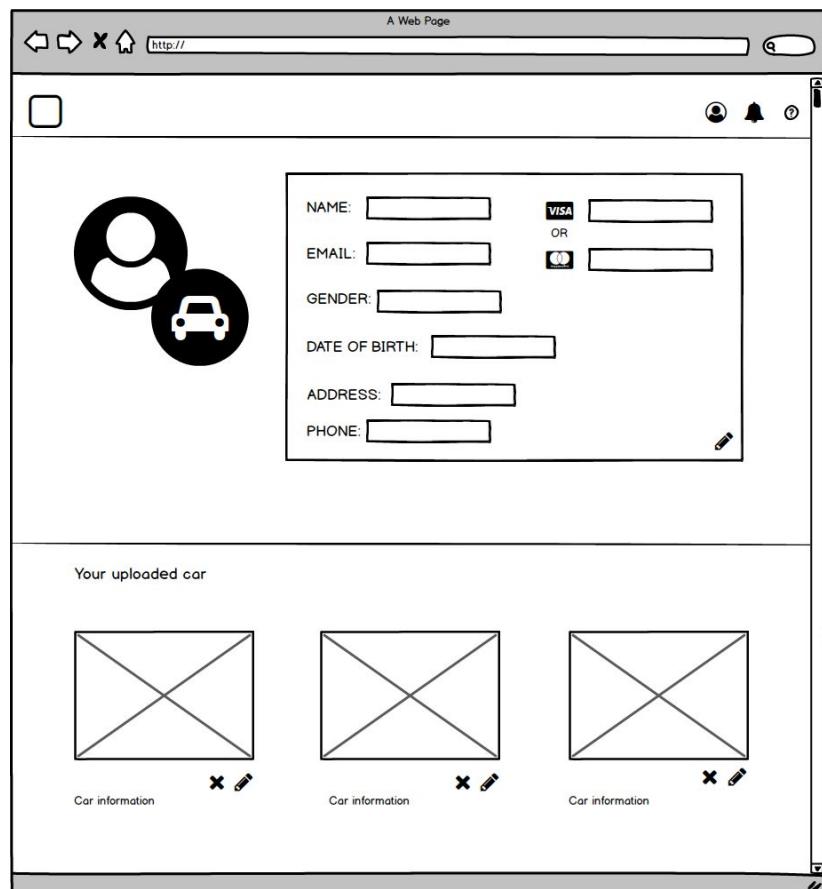
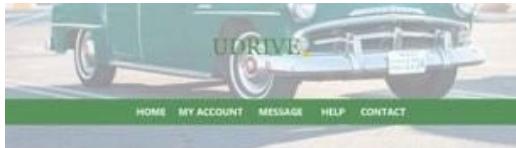
## Main Page

The majority of the layout remained consistent throughout the prototype development, the initial prototype displayed an effective way of displaying some featured vehicles, and providing functionality for users to search for vehicles matching a more specific criterion. The search functionalities became a single bar to allow for a less cluttered user experience and a cleaner appearance. A contact us segment was added to the bottom of the page to allow users to provide feedback or seek immediate help with any issues when using the UHire platform.

From the main page, users are able to navigate to the car details page of their choosing from the available choices on the page.

## Home Bar

The Home bar provides users to navigate to the main page, personal account page, messages page, help, & contact page from anywhere on the website.



## Account Page

The final prototype design for the account page largely remained consistent with the initial prototype, providing functionality for users to view information about other users, and users to edit their own information if necessary. During [SPRINT NUMBER HERE] the development team decided users will find importance in being able to view a users' review history, so all past user reviews were added to the page. As seen by the high-fidelity page prototype, top and bottom banners were added to all the pages to give users a consistent experience throughout the application, and allow for easier navigation.

From the account page, users are able to navigate to upload their own vehicle listing to the site, or edit previously submitted information.



**UPLOAD**

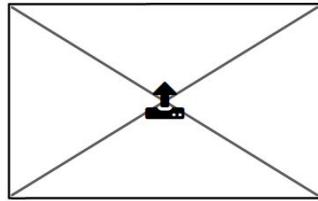


**UPLOAD**

CAR BRAND:

RENT PRICE:

LOCATION:



<< >>

**Complete**

CAR BRAND:

RENT PRICE:

LOCATION:  **COMPLETE**

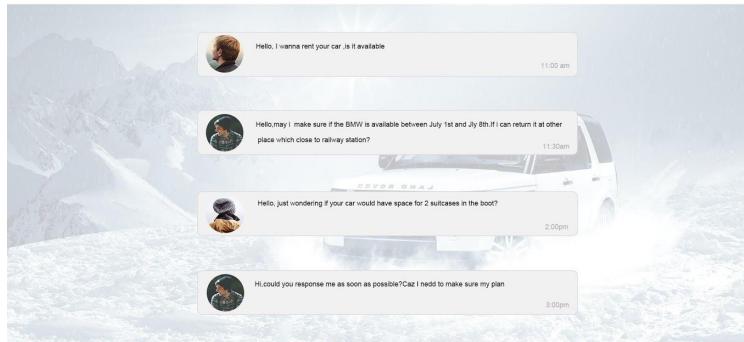
### Upload Vehicle Page

The page where vehicle owners are able to upload their vehicle to the site, the layout remained the same during development. The page was developed to be consistent with the remainder of the website theme, and allows users to provide the required information to upload their own vehicle to their page on the system.

Once on the upload page, users can opt to upload their inputted information to the website, and while they can navigate the remainder of the website as usual, they cannot reach any further pages into the upload segment.



## MESSAGE



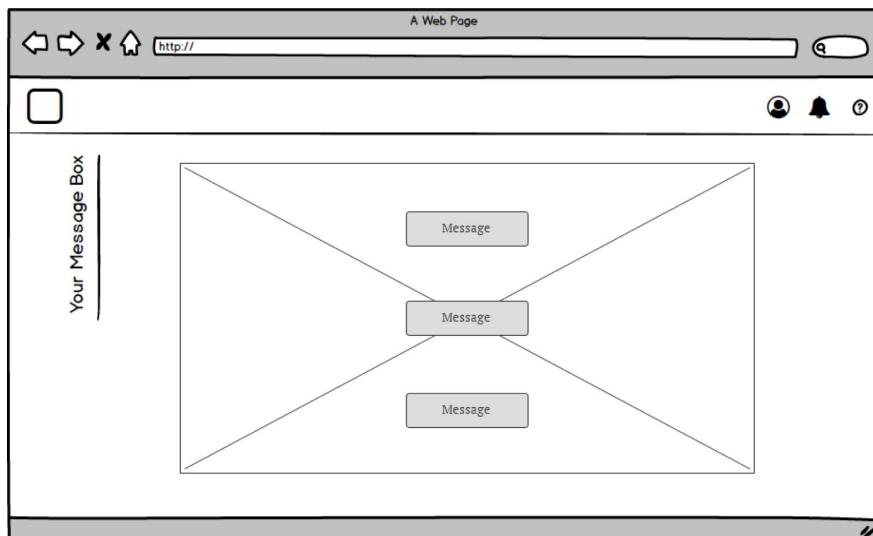
• • •

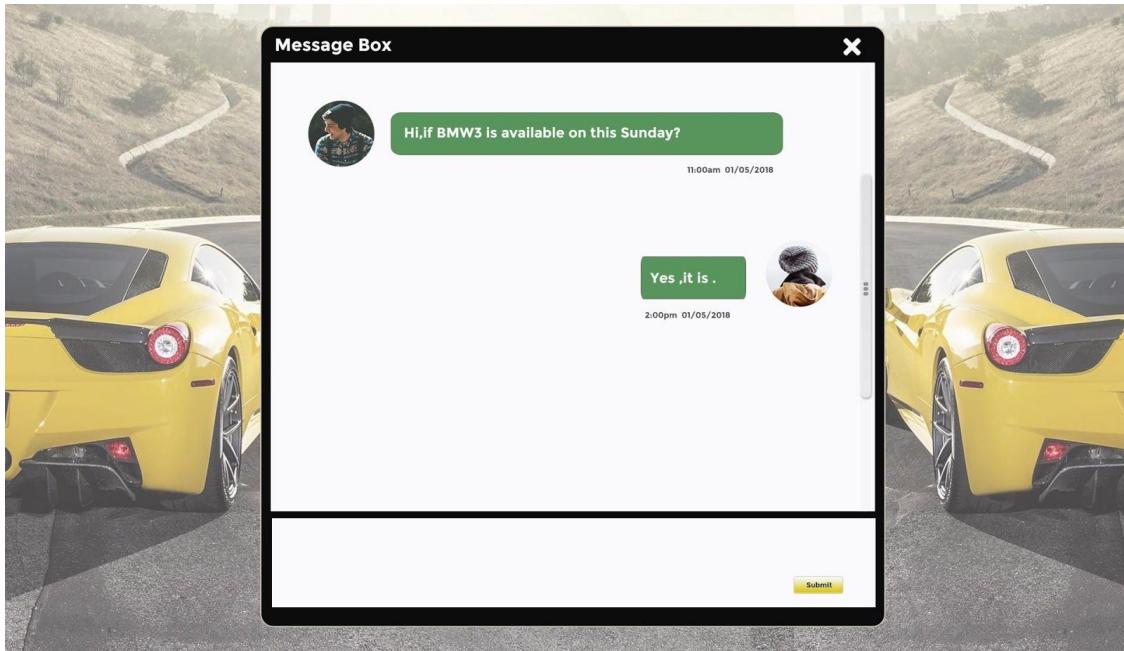


### Messages Page:

The only changes made to the messages page during development is the positioning of the title as to coincide with all the other webpages on the website. As the top and bottom banners are included on the page, the title was moved to better suit the longer-vertical page, as well as improve the readability. As another note, the page navigation (green dots under message box), was removed as infinite scrolling was implemented to allow an easier experience for users to search through their messages.

From the messages page users are able to open up specific message links, to retrieve a more in-depth conversation history from the Chat Box page.

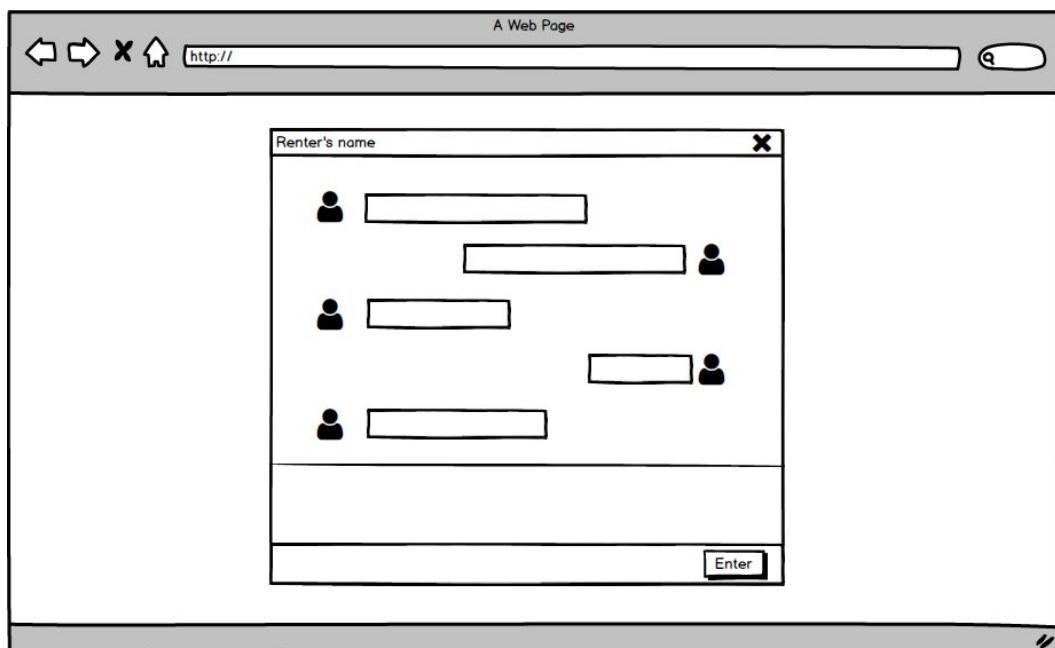




### Chat Box

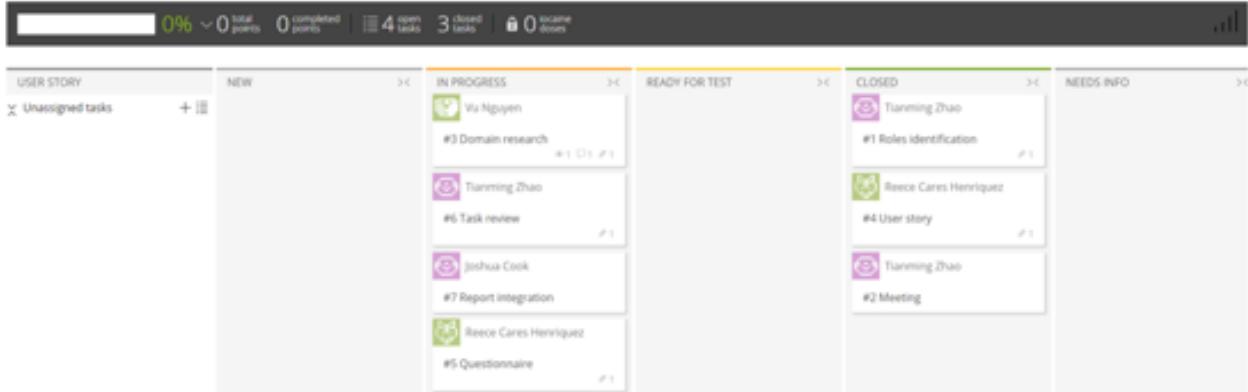
The chat box is a simple design, usually fairly consistent throughout multiple chat-based platforms that can be found online. A user can view messages sent from another user on one side of the screen, see messages that they have sent on the opposite side. The page also provides functionality for a user to write out a response and send it in the space provided.

From the chat-box page, users are able to navigate to another user's profile page to view public information about them.

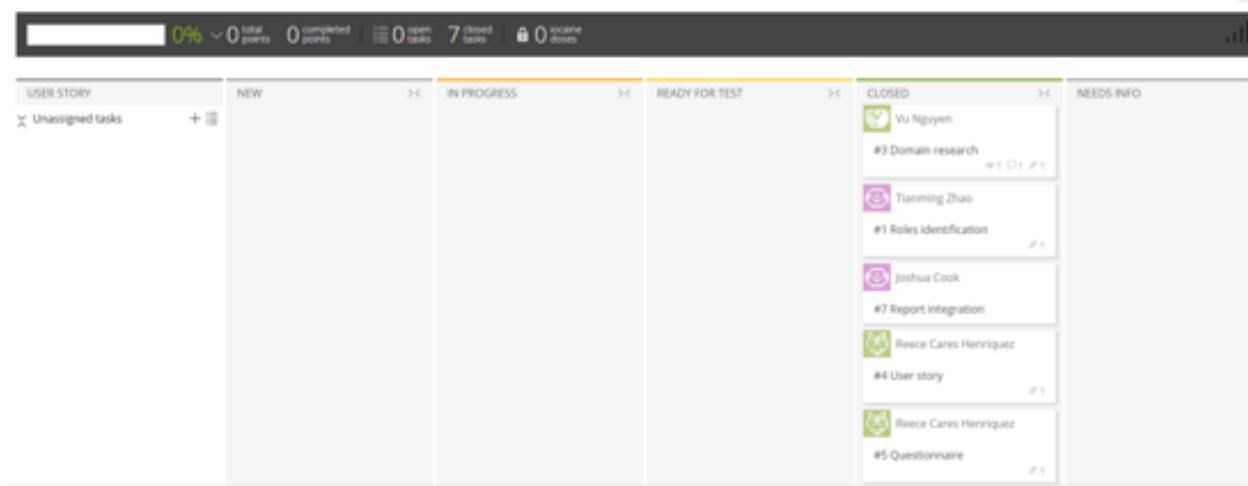


# Taiga Management

## Previous Documentation



GROUP 2 PEER-TO-PEER CAR RENTING SYSTEM REQUIREMENTS ANALYSIS AND DESIGN OF THE SYSTEM 05 MAR 2018-11 MAR 2018



## GROUP 2 PEER-TO-PEER CAR RENTING SYSTEM COMPLETE REQUIREMENTS AND ARCHITECTURE OF THE SYSTEMS 12 MAR 2018-18 MAR 2018

0% ✓ 0 total points 0 completed points | 5 open tasks 1 closed tasks | 🔒 0 secure doses

USER STORY	NEW	IN PROGRESS	READY FOR TEST	CLOSED	NEEDS INFO
X Unassigned tasks +		<p>wenjuan sun #32 Initial User Interface Design</p> <p>Vu Nguyen #31 Creating Persona</p> <p>Tianming Zhao #33 Database Design</p> <p>Reece Cares Henriquez #34 Complete User Stories</p> <p>Joshua Cook #35 Requirements Elicitation</p>		<p>Reece Cares Henriquez #36 Programming</p>	

## GROUP 2 PEER-TO-PEER CAR RENTING SYSTEM COMPLETE REQUIREMENTS AND ARCHITECTURE OF THE SYSTEMS 12 MAR 2018-18 MAR 2018

0% ✓ 0 total points 0 completed points | 1 open tasks 5 closed tasks | 🔒 0 secure doses

USER STORY	NEW	IN PROGRESS	READY FOR TEST	CLOSED	NEEDS INFO
X Unassigned tasks +		<p>Reece Cares Henriquez #34 Complete User Stories</p>	<p>Joshua Cook #35 Requirements Elicitation</p>	<p>Reece Cares Henriquez #36 Programming</p> <p>Tianming Zhao #33 Database Design</p> <p>Vu Nguyen #31 Creating Persona</p> <p>wenjuan sun #32 Initial User Interface Design</p>	

0% ✓ 467.5 total points 0 completed points | 35 open tasks 0 closed tasks | 🔒 0 secure doses

USER STORY	NEW	IN PROGRESS	READY FOR TEST	CLOSED	NEEDS INFO
#142 Membership [100points]	<p>+ Not assigned</p> <p>#148 Implement modify profile</p> <p>+ Not assigned</p> <p>#142 Implement user profile creation code</p> <p>+ Not assigned</p> <p>#145 Implement profile UI</p> <p>+ Not assigned</p> <p>#146 Test profile UI by displaying all user details, including rating</p> <p>+ Not assigned</p> <p>#144 Design profile UI</p> <p>+ Not assigned</p> <p>#143 Display users ratings</p>				

## Sprint Plans

### Sprint 3: 01-13 April

GROUP 2 PEER-TO-PEER CAR RE... SPRINT 3 01 APR 2018-13 APR 2018

100% ✓ 139 total points 139 completed points | 0 open tasks 8 closed tasks 0 locale doses

USER STORY	NEW	<>	<>	CLOSED	<>	NEEDS INFO	<>	Unassigned tasks
✗ #81 As a user, I want to sign up to the website, so that I can have an account to login into the system with an account number and password. (#1)				#83 Design register page UI #84 Implement register page UI #85 Implement and connect register to backend #86 Test register page				
✗ #87 As a user, I want to log in to the system via account number and password so that I can rent cars in the system (be a car renter). (#2)				#88 Design login page UI #89 Implement login page UI #90 Connect login page to backend #91 Test login page				
✗ Unassigned tasks	+ ⋮							

This sprint focuses on functions related to users signing up and creating accounts to use the service, using an account number and a password.

## Sprint 4: 14-22 April

GROUP 2 PEER-TO-PEER CAR RE... SPRINT 4 14 APR 2018-22 APR 2018

100% ✓ 645 total points 645 completed points | 0 open tasks 63 closed tasks | 0 locale doses

USER STORY	NEW	CLOSED	NEEDS INFO
<p>✗ #96 As a user (either a car owner or a car renter), I want to register my credit card's information to the system so that the system can use it to perform transactions. (#3) ● 33points</p>		<ul style="list-style-type: none"> <li>#97 Add a button to user profile, labelled "Become an Owner", link to Add a Car page</li> <li>#98 Implement the back-end system so that it marks the user as a car owner once the user adds a car successfully and give all privileges of car owners to the user.</li> <li>#99 Test the functionality (including testing if all privileges are given correctly)</li> </ul>	
<p>✗ #142 As a user, I want to keep and modify all my information on a profile, including my personal details so that I don't have to introduce myself every time a user questions me and I can check my system credits easily. (#4) ● 100 points</p>		<ul style="list-style-type: none"> <li>#148 Implement modify profile</li> <li>#147 Implement user profile creation code</li> <li>#145 Implement profile UI</li> <li>#150 Test profile page</li> <li>#146 Test profile UI by displaying all user details, including rating</li> </ul>	

GROUP 2 PEER-TO-PEER CAR RE... SPRINT 4 14 APR 2018-22 APR 2018

100% ✓ 645 total points 645 completed points | 0 open tasks 63 closed tasks | 0 locale doses

USER STORY	NEW	CLOSED	NEEDS INFO
<p>100points</p>		<ul style="list-style-type: none"> <li>#146 Test profile UI by displaying all user details, including rating</li> <li>#82 Decide fields required for registration</li> <li>#149 Display users ratings</li> <li>#144 Design profile UI</li> </ul>	
<p>✗ #155 As a car owner, I want to have a verification on my identity, my credit history checked and my cars verified so that all other users can trust me. (#6) ● 50 points</p>		<ul style="list-style-type: none"> <li>#157 Design upload identity and upload credit (include car registration) UI</li> <li>#158 Implement upload identity and verify credit UI</li> <li>#159 Implement the verification back-end code</li> <li>#160 Implement communication with verification server</li> <li>#161 Test the verification process</li> <li>#133 Decide what data to</li> </ul>	
<p>✗ #132 As a car owner, I want</p>			

Help Vu Nguyen

GROUP 2 PEER-TO-PEER CAR RE... SPRINT 4 14 APR 2018-22 APR 2018

100% 645 total points 645 completed points 0 open tasks 63 closed tasks 0 locale doses

USER STORY	NEW	CLOSED	NEEDS INFO
X #132 As a car owner, I want to add a new car to my car listing as well as all its description (e.g. year, model, transmission, odometer and available dates etc.) so that the new car information is posted and potential renter can see my new post. (#9) ● 35points		#133 Decide what data to store for each car #134 Design a UI for managing personal car listings #135 Implement UI #136 Implement add a car to the listing #137 Test the adding functionality	
X #12 As a car owner, I want to view my current car listing so that I know all the status of posted cars should there be some data that needs to update. (#10) ● 25.5points		#40 Design UI elements to display cars in a list #41 Implement the UI elements #80 Test the car listings #79 Connect to the back-end code	
X #193 As a car owner, I want to delete a posted car in my		#196 Add a UI element to the indicating car deletion	

Help Vu Nguyen

GROUP 2 PEER-TO-PEER CAR RE... SPRINT 4 14 APR 2018-22 APR 2018

100% 645 total points 645 completed points 0 open tasks 63 closed tasks 0 locale doses

USER STORY	NEW	CLOSED	NEEDS INFO
X #193 As a car owner, I want to delete a posted car in my car listing so that I can make the car unavailable. (#11) points		#196 Add a UI element to the indicating car deletion #197 Implement deletion functionality #198 Test the car deletion functionality	
X #13 As a renter, I want to be able to view all the cars available based on search criteria such as location and car preference etc. so that I can find the best options. (#12) ● 33points		#42 Decide available criteria to browse by #43 Design listings browsing page #44 Implement GUI #45 Code the search box #76 Connect to the back-end code #77 Test the search functionality	
X #14 As a renter, I want to book requests for cars that I intend to rent so that if the		#48 Implement UI elements that can let renter request booking when viewing car details	

Help Vu Nguyen

GROUP 2 PEER-TO-PEER CAR RE... SPRINT 4 14 APR 2018-22 APR 2018

100% ✓ 645 total points 645 completed points | 0 open tasks 63 closed tasks | 0 localene doses

USER STORY	NEW	CLOSED	NEEDS INFO
✗ #14 As a renter, I want to book requests for cars that I intend to rent so that if the car owner confirms it, I can make payment transactions and get the car. (#13) ● 31 points		#48 Implement UI elements that can let renter request booking when viewing car details #53 Implement the back-end code (so that it extracts money from the renter's bank card and hold it in the system) #78 Connect the UI elements to the back-end code #167 Test the request booking process	
✗ #124 As a car owner, I want to see all the booking requests for my cars so that I can manage them. (#14) ● 30 points		#125 Design booking requests manager GUI #126 Implement Booking requests manager GUI #127 Implement view booking requests #128 Test the request manager	
✗ #129 As a car owner, I want to confirm or reject booking requests so that I can decide which renters to have my car. (#15) ● 50 points		#130 Implementing confirm tickets #131 Implement reject tickets #165 Test the confirmation process #166 Test the rejection process	
✗ #195 As a user, I want to finalize a renting so that I confirm a success of the whole renting and the rental fee held in the system can go to the car owner's account and the bonds can be returned to my account. (#16) points		#199 Implement UI elements to finalize a renting #200 Implement back-end code so that when the renting is finalized by car owner and renter, the rental fees goes to car owner's account and bond is returned to renter's account #201 Test the functionality of finalizing renting process	
✗ #19 As a renter, I want to make transaction within the system through my system		#51 Implement UI elements for users to include their linked bank accounts and bank cards for sending and	

Help Vu Nguyen

GROUP 2 PEER-TO-PEER CAR RE... SPRINT 4 14 APR 2018-22 APR 2018

100% ✓ 645 total points 645 completed points | 0 open tasks 63 closed tasks | 0 localene doses

USER STORY	NEW	CLOSED	NEEDS INFO
✗ #129 As a car owner, I want to confirm or reject booking requests so that I can decide which renters to have my car. (#15) ● 50 points		#130 Implementing confirm tickets #131 Implement reject tickets #165 Test the confirmation process #166 Test the rejection process	
✗ #195 As a user, I want to finalize a renting so that I confirm a success of the whole renting and the rental fee held in the system can go to the car owner's account and the bonds can be returned to my account. (#16) points		#199 Implement UI elements to finalize a renting #200 Implement back-end code so that when the renting is finalized by car owner and renter, the rental fees goes to car owner's account and bond is returned to renter's account #201 Test the functionality of finalizing renting process	
✗ #19 As a renter, I want to make transaction within the system through my system		#51 Implement UI elements for users to include their linked bank accounts and bank cards for sending and	

GROUP 2 PEER-TO-PEER CAR RE... SPRINT 4 14 APR 2018-22 APR 2018

100% ✓ 645 total points 645 completed points | 0 open tasks 63 closed tasks | 0 locale doses

USER STORY	NEW	CLOSED	NEEDS INFO
✗ #19 As a renter, I want to make transaction within the system through my system account so that I can pay the car owner easily. (#19) 65points		#51 Implement UI elements for users to include their linked bank accounts and bank cards for sending and receiving money in profile #112 Connect the UI elements to the back-end code #52 Implement the back-end code (update user's bank information accordingly and link users to their bank accounts or bank card) #113 Test the functionality	
✗ #29 As a potential renter, I want the rental fees to be held in the system until the renting is finalized so that the rental fee is kept securely during my renting. (#20) 56.5points		#178 Design a transaction subsystem that holds all money from renters (money goes from renter's bank card and will go to car owner's account) #179 Connect the subsystem to 'Request booking' button #180 Test the functionality	

GROUP 2 PEER-TO-PEER CAR RE... SPRINT 4 14 APR 2018-22 APR 2018

100% ✓ 645 total points 645 completed points | 0 open tasks 63 closed tasks | 0 locale doses

USER STORY	NEW	CLOSED	NEEDS INFO
✗ #194 As a car owner, I want to receive the rental fees after the renter has finished renting so that the whole process is finalized. (#21) points		#202 Add functionality rental fee transfer #203 Test the functionality	
✗ #28 As a car owner, I want renters to provide a bond within the system so that if they cause damages to the car, they need to pay for this. (#22) 75points		#176 Implement an automatic process for a car renter to pay the bond when they request a booking #177 Test the functionality	
✗ #21 As the system owner, I want payments to be made as a commission, direct to my account. (#23) 61points		#173 Implement UI elements such that the system admin can update the commission rate in terms and conditions #174 Implement the back-end so that each transaction consists of a percentage commission fee that will go to the system owner's account	

In this sprint we finished all user stories in relation to membership, develop functions for listing and searching cars, and the process of payment transactions.

## Sprint 5: 23-29 April

The screenshot shows a Jira board for 'GROUP 2 PEER-TO-PEER CAR RE...' during 'SPRINT 5 23 APR 2018-29 APR 2018'. The board has four columns: 'USER STORY', 'NEW', 'CLOSED', and 'NEEDS INFO'. There are three user stories in the 'USER STORY' column:

- #15 As a user, I want to message car owners/renters before accepting/creating booking requests in a private messaging channel so that I can know more details like handover time, location and discuss any concerns. (#17) ● 51 points
- #17 As a user, I want to see all my current messaging in a list so that I can continue the messaging with someone. (#18) ● 34 points
- Unassigned tasks

In the 'CLOSED' column, there are six tasks:

- #49 Design a message system GUI
- #50 Implement message system GUI
- #61 Connect message system with database query system
- #62 Implement chat features send message
- #63 Test the message system
- #64 Design a message system history page GUI
- #65 Implement message history GUI
- #66 Implement display message history
- #169 Test the functionality

The 'NEEDS INFO' column is currently empty.

In this sprint functions about messaging between users are developed.

## Sprint 6: 30 April - 06 May

The screenshot shows a Jira board for 'GROUP 2 PEER-TO-PEER CAR RE...' during 'SPRINT 6 30 APR 2018-06 MAY 2018'. The board has four columns: 'USER STORY', 'NEW', 'CLOSED', and 'NEEDS INFO'. There are three user stories in the 'USER STORY' column:

- #100 As a renter, I want to give a review and rating to the car owner after the renting is finalized so that other people can all read the feedbacks to the car owner and help other people to make better decisions. (#8) ● 45 Points
- #156 As a user, I want to be able to check everyone's (including mine) reviews/ratings so that I know who to trust and what to pay more attention to. (#7) ● 24 points
- #143 As a user, I would like to view other users' profiles to gather more information about the user and gain confidence about conducting a transaction with them. (#5) ● 76 points

In the 'CLOSED' column, there are four tasks in each of the first two rows:

- #101 Design feedback box integrated into finalised transaction
- #102 Implement feedback UI and connect it to the booking lists page
- #103 Connect feedback UI to backend
- #104 Test feedback UI
- #162 Design UI elements so that the reviews and ratings can be displayed on the profile
- #163 Add UI elements in the front-end code
- #164 Test the review/ratings section

In the third row, there are four tasks:

- #151 Design profile in read-only mode, for when viewing another profile
- #152 Implement the alternative profile UI
- #153 Connect the read-only profile UI to appropriate back-end code
- #154 Test the read-only profile, including ratings

The 'NEEDS INFO' column is currently empty.

In this sprint we develop functions related to leaving and viewing reviews.

## Sprint 7: 07 - 13 May

**GROUP 2 PEER-TO-PEER CAR RE... SPRINT 7 07 MAY 2018-13 MAY 2018**

100% ✓ 169 total points | 169 completed points | 0 open tasks | 14 closed tasks | 0 cocaine doses

USER STORY	NEW	CLOSED	NEEDS INFO
✗ #108 As a user, I want to sign up via my social network accounts so that I do not have to fill in my personal details manually. (#26) ● 36 points		#181 Test if all the widget works functionally	#109 Design GUI widget to be added to registration page #110 Implement GUI widget to be added to registration page #111 Integrate registration with common social media accounts #182 Test if a user can sign up purely via social network account
✗ #24 As a user, I want to view the guidelines of the website so that I know all the policies and terms and conditions. (#27) ● 21 points			#121 Design a guideline page GUI #122 Implement a guideline page GUI #123 Test if guidelines are displayed properly

**GROUP 2 PEER-TO-PEER CAR RE... SPRINT 7 07 MAY 2018-13 MAY 2018**

100% ✓ 169 total points | 169 completed points | 0 open tasks | 14 closed tasks | 0 cocaine doses

USER STORY	NEW	CLOSED	NEEDS INFO
✗ #20 As the system owner, I want to update my bank account via a system admin so that all commission fees can go to the updated bank account. (#24) ● 75 points		#170 Implement UI elements such that the system admin can update the system owner's bank account in terms and conditions #171 Link the system owner's bank account to each transactions between car renters and car owners #172 Test the functionality	
✗ #22 As a user, I want to link my account to my social network accounts, so that people can find more information on my social network account. (#25) ● 37 points			#105 Implement a widget to display a social media link in view self profile #106 Implement modification of social media link in my profile #107 Implement a widget to display a social media link in view other profile #181 Test if all the widget works functionally

Sprint 7 focused on functions that enable users to connect their UHire account to their social media accounts and view the T&C/guidelines of the system. Also, we develop functions for system admin to receive commission fees from finalised transactions between users.

## Sprint 8: 14-20 May

USER STORY	NEW	CLOSED	NEEDS INFO
#27 As a system admin, I want to be upload the latest version of policy and terms and conditions, and guidelines. (#28) 55points		<ul style="list-style-type: none"> <li>#114 Design administration settings page GUI</li> <li>#115 Implement administration setting pages GUI</li> <li>#185 Test the functionality</li> <li>#118 Implement guidelines modifier</li> <li>#117 Implement terms and conditions modifier</li> <li>#116 Implement policy modifier</li> </ul>	
#25 As a system admin, I want to delete users so that I can clean up those bad-behavior users. (#29) 21 points		<ul style="list-style-type: none"> <li>#120 Implement delete button</li> <li>#183 Connect delete button to the back-end code and database</li> <li>#184 Test the deletion functionality</li> <li>#119 Add a delete button widget on profile only available to admins</li> </ul>	

In sprint 8 we develop functions that enable system admin to upload T&C to the system and to delete users' accounts if they have bad behaviours.

# Meeting Records

## MEETING AGENDA – INITIAL PLANS

---

### MEETING INFORMATION

**Objective:** Discussion on the roles, initial plan for the whole project, technology used, first task and deliverable for each team member

<b>Date:</b>	09/03/2018	<b>Location:</b>	Building 3 Room 128
<b>Time:</b>	11.00 AM	<b>Meeting Type:</b>	In-person
<b>Team leader:</b>	Tianming Zhao		
<b>Timekeeper:</b>	Tianming Zhao	<b>Note Taker:</b>	Reece
<b>Attendees:</b>	Tianming Zhao, Wenjuan Sun,		

---

### PREPARATION FOR MEETING

**Please Read:** Roles.docs, week2 Task.docx

**Please Bring:** All outcomes or deliverable

---

### ACTION ITEMS FROM PREVIOUS MEETING

None. This is the very first meeting

---

### AGENDA ITEMS

Item/Presenter/Time Allotted

1. Initial user story / Reece / 10 mins
2. Initial plan / Tianming / 5 mins
3. Research presentation / Vu / 15 mins
4. How to improve the user experience / 10 mins
5. Q&A / All team members / 5 - 10 mins

---

### NEW ACTION ITEMS

Item/Responsible/Due Date

1. Questionnaire / Reece / 11/03/2018
2. Getting familiar with technology / Reece / 18/03/2018
3. Detailed plan with WBS / Tianming / 13/03/2018
4. Persona / Vu / 18/03/2018

# Initial Meeting

9/03/18

11:00AM

Building 3, Rm128

<b>Meeting called by:</b>	Tianming Zhao	<b>Type of meeting:</b>	In Person, Planning
<b>Timekeeper:</b>	Joshua Cook	<b>Note taker:</b>	Joshua Cook

**Attendees:** Tianming Zhao, Wenjuan(Chloe) Sun, Joshua Cook, Vu Hoang

**Please read:** Roles.docx, Week 2 Tasks.docx

**Please bring:** All outcomes or deliverables

## Minutes

**Agenda item:** Initial Plan

**Presenter:** Tianming Zhao

**Discussion:**

Tianming presented to the group a rough timeline of how we should progress through the project, including gathering requirements and starting development.

**Conclusions:**

Requirements were discussed and agreed upon by the team, involving account validation, risks of the system, and payment methods. A questionnaire will need to be developed to help the team have a better understanding of requirements.

**Action items**

- |                        | <b>Person responsible</b> | <b>Deadline</b> |
|------------------------|---------------------------|-----------------|
| ✓ Create Questionnaire | Reece Henriquez           | 11/03/18        |
| ✓ Create detailed WBS  | Tianming Zhao             | 13/03/18        |

**Agenda item:** Research Presentation

**Presenter:** Vu Hoang

**Discussion:**

Vu Presented an initial domain analysis of the current market.

**Conclusions:**

Team decided what aspects of other similar products could be improved on and changed.

**Action items**

- |                            | <b>Person responsible</b> | <b>Deadline</b> |
|----------------------------|---------------------------|-----------------|
| ✓ Requirements Elicitation | Joshua Cook               | 16/03/18        |

# MEETING AGENDA – TUTORIAL INTERVIEW

---

## MEETING INFORMATION

**Objective:** Interview Kerri about requirements of the system

<b>Date:</b>	13/03/2018	<b>Location:</b>	Building 3, Room 127
<b>Time:</b>	4:30PM	<b>Meeting Type:</b>	Interview
<b>Called By:</b>	Tianming Zhao	<b>Note Taker:</b>	Reece Henriquez
<b>Timekeeper:</b>	Reece Henriquez		
<b>Attendees:</b>	Tianming Zhao, Reece Henriquez, Dr Kerri		

---

## PREPARATION FOR MEETING

**Please Read:** *Questionnaire.docx*

**Please Bring:** *Questionnaire.docx*

---

## ACTION ITEMS FROM PREVIOUS MEETING

None

---

## AGENDA ITEMS

Interview Dr Kerri the following questions

1. Are social media accounts linked for logging in purposes, or are users able to view each other's social media accounts on their profiles, or both?
2. It is mentioned that the application needs to manage payment transactions between the car owner, and the system owner. How is the owner paid? (Commission? fixed rate? monthly debit?)
3. How will the system prevent users from organising a booking agreement between themselves, bypassing the system fees and agreements?
4. Should there be punishments for users with low ratings/ users that fail credit checks?
5. Should there be ongoing credit checks or simply once at registration?
6. What is an appropriate length of time for a "cool-off" period where users will not be penalised for cancelling a booking?
7. How will user identification be validated? i.e. ensuring that a driver's licence is valid, not expired, not a fake etc.
8. How should dishonest practices be handled?

# Tutorial Interview

13.03.18  
4:30PM

Building 3, Room 127

<b>Meeting called by:</b>	Tianming Zhao	<b>Type of meeting:</b>	Interview
<b>Timekeeper:</b>	Reece Henriquez	<b>Note taker:</b>	Reece Henriquez

**Attendees:** Tianming Zhao, Reece Henriquez

**Please read:** Questionnaire.docx

**Please bring:** Questionnaire.docx

## Minutes

**Agenda item:** Interview Tutor

**Presenter:** Tianming, Reece

### Discussion:

Reece and Tianming asked Dr Kerri questions from Questionnaire.docx.

### Conclusions:

Answers given included:

1. Are social media accounts linked for logging in purposes, or are users able to view each other's social media accounts on their profiles, or both?
  - a. Social media should be able to be linked for registration purposes but aren't used further than that.
2. It is mentioned that the application needs to manage payment transactions between the car owner, and the system owner. How is the owner paid? (Commission? fixed rate? monthly debit?)
  - a. The owner of the system will be paid through commission.
3. How will the system prevent users from organising a booking agreement between themselves, bypassing the system fees and agreements?
  - a. If the car owner & renter arrange a private deal outside of the system before submitting a ticket, then the car will be uninsured.
4. Should there be punishments for users with low ratings/ users that fail credit checks?
  - a. No, punishment is up to the discretion of the renters trust in the user.
5. Should there be ongoing credit checks or simply once at registration?
  - a. It would be good to have ongoing checks.
6. How will user identification be validated? I.e. ensuring that a driver's licence is valid, not expired, not a fake etc.
  - a. An external dummy system provides validation.

Action items	Person responsible	Deadline
✓ Update Requirements Elicitation with new Info	Joshua Cook	15/03/18

# MEETING AGENDA – REQUIREMENTS AND ARCHITECTURE OF THE SYSTEM

---

## MEETING INFORMATION

**Objective:** *Discussion on the current requirements, based on the requirements, initially create a draft of system architecture*

<b>Date:</b>	14/03/2018	<b>Location:</b>	Building 3 Room 128
<b>Time:</b>	12.40 AM	<b>Meeting Type:</b>	In-person
<b>Team leader:</b>	Tianming Zhao		
<b>Timekeeper:</b>	Josh Cook	<b>Note Taker:</b>	Reece
<b>Attendees:</b>	All team members		

---

## PREPARATION FOR MEETING

**Please Read:** *User stories.txt, Week 3 tasks*

**Please Bring:** *All outcomes or deliverable*

---

## ACTION ITEMS FROM PREVIOUS MEETING

Partially develop a user stories list.

Conduct domain analysis

---

## AGENDA ITEMS

Item/Presenter/Time Allotted

1. User story revisited / Reece / 7 mins
2. Adds to user stories / All team members / 5 mins
3. Persona / Vu / 10 mins
4. Structure of the system / Tianming / 5 mins
5. Architecture of the system / Tianming, Reece / 20 mins
6. User interface / Tianming, Reece, Chloe / 10 mins
7. Q&A / All team members / 5 - 10 mins

---

## NEW ACTION ITEMS

Item/Responsible/Due Date

1. Complete domain analysis / Vu / 14/03/2018
2. Integration of requirements / Josh / 16/03/2018
3. Database design / Tianming / 18/03/2018
4. Back-end design / Tianming, Reece / 18/03/2018
5. Persona / Vu / 17/03/2018
6. Low-fi design / Wenjuan / 18/03/2018
7. Programming tasks for Reece / Reece / 18/03/2018

# Requirements of the System

14/03/18

12:40PM

Reece's House

<b>Meeting called by:</b>	Tianming Zhao	<b>Type of meeting:</b>	In-person
<b>Timekeeper:</b>	Joshua Cook	<b>Note taker:</b>	Reece Henriquez

**Attendees:** Tianming, Chloe, Reece, Joshua, Vu

**Please read:** User Stories.txt, Week 3 Tasks.docx

**Please bring:** All outcomes or deliverables.

## Minutes

**Agenda item:** User Stories

**Presenter:** Whole Team

**Discussion:**

User Stories document was revisited and discussed, suggesting changes to how the system bank account should be handled, and dishonest users.

**Conclusions:**

Team decided to have an admin/ system account which will be linked to an external bank account, and provide a FAQ for complications to negative actions.

**Action items**

**Person responsible**

**Deadline**

**Agenda item:** User Interface

**Presenter:** Tianming, Chloe

**Discussion:**

The team discussed how the user should navigate through the pages of the system, and the basic layout/ design of each page. All initial pages were drawn up on a whiteboard, with changes being made iteratively.

**Conclusions:**

An idea of how the messaging system between users would look and work was established, and all the main page layouts were sketched on a whiteboard.

**Action items**

**Person responsible**

**Deadline**

✓ Low Fidelity Sketches

Chloe Sun

18/03/18

**Special Notes:** User interface took much longer than anticipated and meeting was adjourned once complete.

# MEETING AGENDA – UI AND DATABASES REVISITED

---

## MEETING INFORMATION

**Objective:** Review the UI created by Wenjuan Sun, and started to think about how to implement it.

<b>Date:</b>	21/03/2018	<b>Location:</b>	Building 3-122
<b>Time:</b>	13.30-14.30 PM	<b>Meeting Type:</b>	In-person
<b>Called By:</b>	Tianming		
<b>Timekeeper:</b>	Josh	<b>Note Taker:</b>	Reece
<b>Attendees:</b>	All team members		

---

## PREPARATION FOR MEETING

**Please Read:** Low-Fi UI designs, database diagram

**Please Bring:** User stories (will be needed for next week submission)

---

## ACTION ITEMS FROM PREVIOUS MEETING

Item/Responsible/Due Date

1. Databases diagram / Tianming / 18/03/2018
2. Back-end design / Tianming, Reece / 18/03/2018
3. Persona / Vu / 17/03/2018
4. Low-fi design / Wenjuan / 18/03/2018

---

## AGENDA ITEMS

Item/Presenter/Time Allotted

1. Discussion database design / Tianming Zhao / 10 minutes
2. Discussion on UI / Wenjuan Sun / 20 minutes
3. Discussion on UI implementation / All team member / 15 minutes
4. Q & A / All team members / 15 minutes

---

## NEW ACTION ITEMS

Item/Responsible/Due Date

1. Refine database design / Tianming / 25/03/2018
2. Refine UI design / Wenjuan Sun / 25/03/2018
3. Start to implement the web page / All team member / 30/03/2018

---

## OTHER NOTES OR INFORMATION

None.

# Meeting Minutes - Week 4

21-03-18

1:00pm

Building 3, Rm122

<b>Meeting called by:</b>	Tianming Zhao	<b>Type of meeting:</b>	Weekly Progress Meeting
<b>Timekeeper:</b>	Joshua Cook	<b>Note taker:</b>	Joshua Cook

**Attendees:** Tianming Zhao, Chloe Sun, Vu Hoang, Joshua Cook

**Please read:** Database Design v1.bmp, UI Sketches Folder

**Please bring:** User Stories

## Minutes

**Agenda item:** Review Database Architecture

**Presenter:** Tianming Zhao

**Discussion:**

Review of the database design, investigate use cases and determine if there are any aspects missing to the schema.

**Conclusions:**

Changes to the user credit card and bank account were made, moving bank account information to another class in order for admin and normal users to have it. Commission rate attribute was also added to transaction.

**Action items**

**Person responsible**

**Deadline**

✓ Develop Database Design v2.bmp

Tianming Zhao

30/03/18

**Agenda item:** Review User Interface Sketches

**Presenter:** Tianming, Chloe

**Discussion:**

Whole team discussed various use cases, verifying that all user cases can be handled in the current user interface design.

**Conclusions:**

Josh suggested some changes to the messaging page, extending a panel for aesthetic reasons and to include additional information about the conversation. Also booking confirmations needed to be handled through the user interface, it was decided to include in the messaging page, distinguishing between normal conversations and booking requests that can be accepted or declined. Other little details will be added to the design such as additional information for listing a vehicle, reviews on user account pages, changing the reviews on the vehicle page to be specific to the vehicle, not the user in general.

**Action items**

**Person responsible**

**Deadline**

✓ Next iteration of user interface designs

Chloe Sun

31/03/18

# MEETING AGENDA – MID-PROJECT REPORT

## MEETING INFORMATION

**Objective:** Distribute tasks for each team member to finish the progress report due in week 6

<b>Date:</b>	28/03/2018	<b>Location:</b>	3-122
<b>Time:</b>	2.00-2.40 p.m.	<b>Meeting Type:</b>	In-person
<b>Called By:</b>	Tianming		
<b>Timekeeper:</b>	Josh	<b>Note Taker:</b>	Josh, Tianming, Wenjuan
<b>Attendees:</b>	Tianming		

## PREPARATION FOR MEETING

**Please Read:** Refined UI design, Refined database design, Assignment specification

**Please Bring:** User stories.

## ACTION ITEMS FROM PREVIOUS MEETING

Item/Responsible/Due Date

1. Refined database design / Tianming / 25/03/2018
2. Refined UI design / Tianming / 25/03/2018

## AGENDA ITEMS

Item/Presenter/Time Allotted

1. Review of user stories / All team members / 10 minutes
2. Discussion on assigning tasks to team member for finishing mid-project report / All team members / 30 minutes
3. Implementation of the web / All team members / 10 minutes
4. Q & A / All team members / 5 minutes

## NEW ACTION ITEMS

Item/Responsible/Due Date

1. A list of tasks for each user story / Reece, Vu / 01/04/2018 (strict)
2. A use case diagram / Tianming / 29/03/2018 (strict)
3. Use case descriptions / All team members / 01/04/2018 (strict)
4. Evidence for following Scrum / Vu / 28/02/2018
5. Preliminary design / Tianming (Database, class diagram), Wenjuan (UI Design), Reece (class diagram)
6. Meeting minutes and meeting record / Josh / 01/03/2018 (strict)
7. Plan for remaining tasks / Tianming / 01/03/2018 (strict)
8. Signature for every team member's contribution / All team members / 03/03/2018 before class(strict)
9. Integration of the report / Josh / 02/03/2018

**OTHER NOTES OR INFORMATION**

None.

# Week 5 - Progress Meeting

28/03/18

2:00PM

Building 3, Room 122

<b>Meeting called by:</b>	Tianming Zhao	<b>Type of meeting:</b>	In-Person Progress Update
<b>Timekeeper:</b>	Joshua Cook	<b>Note taker:</b>	Reece Henriquez

**Attendees:** Tianming Zhao, Chloe Sun, Vu Hoang, Reece Henriquez, Joshua Cook

**Please read:** Refined UI Design, Refined database design, Assignment Specification

**Please bring:** User Stories

## Minutes

**Agenda item:** Review User Stories

**Presenter:** Group

**Discussion:**

All team members went through user stories to make sure all aspects of the system were covered and altered to suit changes requested by week 4 feedback.

**Conclusions:**

Various user stories to be fleshed out and altered

Action items	Person responsible	Deadline
✓ Update User Stories	Tianming, Reece	31/03/18
✓ Update Taiga	Vu Hoang	1/04/18

**Agenda item:** Discussion on assigning tasks to members

**Presenter:** Group

**Discussion:**

The group listed all requirements for mid-project deliverables that had not yet been completed, and distributed the work appropriately between the group.

**Conclusions:**

Various new action items were created.

Action items	Person responsible	Deadline
✓ Tasks for each user story	Reece, Vu	01/04/18
✓ Use Case Diagram	Tianming	29/03/18
✓ Use Case Descriptions	All Team Members	01/04/18
✓ Plan for remaining tasks	Tianming	01/04/18
✓ Integration of Report	Joshua	02/04/18

# MEETING AGENDA – [WEEK 7]

## MEETING INFORMATION

**Objective:** *Discuss what changes need to be made to the project documentation, assign new tasks to team members.*

**Date:** 11/04/2018  
**Time:** 11:00 AM  
**Timekeeper:** Reece Henriquez  
**Attendees:** Entire Team

**Location:** Online  
**Meeting Type:** Video Conference  
**Note Taker:** Joshua Cook

## PREPARATION FOR MEETING

**Please Read:** *Mid-Project Report*

**Please Bring:** N/A

## AGENDA ITEMS

Item/Presenter/Time Allotted

1. Discussion about creating High Fidelity Designs / Chloe Sun / 20minutes
2. Discussion about future iterations / Tianming Zhao / 20minutes

## NEW ACTION ITEMS

Item/Responsible/Due Date

1. Begin Creating Test Data / Vu Hoang / 30/04/2018
2. High Fidelity Prototype Designs / Chloe Sun / 18/04/2018

# Team Meeting Week 7

Date: 11/04/2018

Time: 11:00am

Location: Online Video Conference

<b>Meeting called by:</b>	Tianming Zhao	<b>Type of meeting:</b>	In Person
<b>Timekeeper:</b>	Reece Henriquez	<b>Note taker:</b>	Joshua Cook

**Attendees:** Tianming Zhao, Chloe Sun, Reece Henriquez, Joshua Cook, Vu Hoang

**Please read:** Mid-Project Report

**Please bring:**

## Minutes

**Agenda item:** Discuss High Fidelity Designs

**Presenter:** Tianming Zhao

**Discussion:**

Tianming started by stating that High Fidelity Designs need to be designed before the team begins to create the web-pages, the team went through the current design and determined if any changes needed to be made.

**Conclusions:**

Some aspects of the low-fidelity designs were slightly altered to be implemented in the high fidelity designs.

More Specific information and page content was determined by the team.

Action items	Person responsible	Deadline
✓ Create High Fidelity Designs	Chloe Sun	18/04/2018

**Agenda item:** Discussion about future iterations

**Presenter:** Tianming Zhao

**Discussion:**

Tianming outlined the timeline for developing the final system, html pages needed to be developed to match the upcoming UI designs, and the system database needed to be developed.

**Conclusions:**

As chloe finished high fidelity UI designs, the team could work on implementing them.

Tianming would begin developing the database, and Vu is to begin creating test data for the final system.

Action items	Person responsible	Deadline
✓ Develop HTML pages	Tianming, Reece, Josh	25/04/2018
✓ Create Test Data	Vu Hoang	30/04/2018

# MEETING AGENDA – MID-PROJECT FEEDBACK

---

## MEETING INFORMATION

**Objective:** *Discuss the feedback received for the mid-project report, report on progress*

<b>Date:</b>	23/04/2018	<b>Location:</b>	Reece's House
<b>Time:</b>	3:00 PM	<b>Meeting Type:</b>	In Person
<b>Called By:</b>	Tianming Zhao		
<b>Timekeeper:</b>	Vu Hoang	<b>Note Taker:</b>	Joshua Cook
<b>Attendees:</b>	Entire Group		

---

## PREPARATION FOR MEETING

**Please Read:** *Mid-Project Report Feedback*

**Please Bring:** *Mid-Project Report Feedback*

---

## AGENDA ITEMS

Item/Presenter/Time Allotted

1. Discuss points specified in the feedback document / Tianming Zhao, Joshua Cook, Reece Henriquez / 20 minutes
2. Specify what changes need to be made to documentation / Tianming Zhao / 15 minutes
3. Update on system design progress / Tianming, Reece / 10 minutes

---

## NEW ACTION ITEMS

Item/Responsible/Due Date

1. Update Use Case Descriptions / Vu Hoang / 30/04/2018
2. Update User Stories / Reece Henriquez, Joshua Cook / 03/05/2018
3. Update Taiga Use Case Descriptions / Vu Hoang / 04/05/2018

# Team Meeting - Week 8

Date 23/04/2018

Time: 3:00PM

Location: Reece's House

<b>Meeting called by:</b>	Tianming Zhao	<b>Type of meeting:</b>	In-Person
<b>Timekeeper:</b>	Vu Hoang	<b>Note taker:</b>	Joshua Cook

**Attendees:** Tianming Zhao, Reece Henriquez, Vu Hoang, Joshua Cook

**Please read:** Mid-Project Report Feedback

**Please bring:** Mid-Project Report Feedback

## Minutes

**Agenda item:** Discuss Feedback/ Changes

**Presenter:** Tianming, Reece, Josh

### Discussion:

The team went through the feedback given, pulling up the user stories, use case descriptions and Taiga pages specified in the feedback. The team collectively contributed as to what changes to make if not clearly specified, and decided who is available to make the changes.

### Conclusions:

The user stories were brought up a lot in the feedback therefore needed a lot of attention. All other aspects would be worked on and updated.

### Action items

	<b>Person responsible</b>	<b>Deadline</b>
✓ Update Use Case Descriptions	Vu Hoang	30/04/2018
✓ Update User Stories	Reece, Joshua	03/05/2018
✓ Update Taiga Documentation	Vu Hoang	04/05/2018

**Agenda item:** Update on System progress

**Presenter:** Tianming Zhao, Reece

### Discussion:

Tianming and Reece showcased the current development of the HTML pages, and what functionality is yet to be implemented

### Conclusions:

The web-pages were making good progress and reflected the High Fidelity designs well. PHP implementation needed to be implemented to provide functionality of the system.

# MEETING AGENDA – FUNCTIONALITY REPORT

---

## MEETING INFORMATION

---

**Objective:** *Preparation for the upcoming Functionality report*

**Date:** 30/05/2018

**Location:** Online

**Time:** 3:00 PM

**Meeting Type:** Video Conference

**Called By:** Tianming Zhao

**Timekeeper:** Joshua Cook

**Note Taker:** Joshua Cook

**Attendees:** Entire Team

---

## PREPARATION FOR MEETING

---

**Please Read:** N/A

**Please Bring:** N/A

---

## AGENDA ITEMS

---

Item/Presenter/Time Allotted

1. Discuss Current Functionality / Tianming Zhao / 20 minutes
2. Discuss Requirements for Report / Tianming Zhao / 10 minutes

---

## NEW ACTION ITEMS

---

Item/Responsible/Due Date

1. WebPage PHP Integration / Tianming Zhao / 03/05/2018
2. Functionality Report / Joshua Cook / 03/05/2018

# Team Meeting - Week 9

Date: 30/05/2018

Time: 3:00PM

Location: Online

<b>Meeting called by:</b>	Tianming Zhao	<b>Type of meeting:</b>	Video Conference
<b>Timekeeper:</b>	Joshua Cook	<b>Note taker:</b>	Joshua Cook

**Attendees:** Tianming Zhao, Chloe Sun, Joshua Cook, Reece Henriquez, Vu Hoang

**Please read:** N/A

**Please bring:** N/A

## Minutes

**Agenda item:** Discuss Functionality                           **Presenter:** Tianming Zhao

**Discussion:**

Tianming displayed the current functionality the system has, which is mainly using javascript.

**Conclusions:**

The system had to integrate PHP in order to connect to the database.

Action items	Person responsible	Deadline
✓ WebPage PHP Integration	Tianming Zhao	03/05/2018

**Agenda item:** Discuss Report Requirements                           **Presenter:** Tianming Zhao

**Discussion:**

Tianming addressed what functionalities he expects the team to implement by the functionality report due date, and what needs to be completed afterwards.

**Conclusions:**

The aim is to have basic functionality, allowing users to navigate and view the web pages with information from the database, and using the search engine to find vehicles.

Action items	Person responsible	Deadline
✓ Functionality Progress Report	Joshua Cook	03/05/2018

# MEETING AGENDA – WEEK 10

---

## MEETING INFORMATION

---

**Objective:**

**Date:** 10/05/2018      **Location:** Online  
**Time:** 12:00 PM      **Meeting Type:** Text Conversation  
**Called By:** Tianming Zhao  
**Timekeeper:** Reece Henriquez      **Note Taker:** Joshua Cook  
**Attendees:** Entire Group

---

## PREPARATION FOR MEETING

---

**Please Read:** *Functionality Progress Report*

**Please Bring:** N/A

---

## AGENDA ITEMS

---

### Item/Presenter/Time Allotted

1. Discuss Required Documentation for final submission / Tianming Zhao / 15 minutes
2. Discuss plan for remaining front / back end implementation / Tianming Zhao / 10 minutes
3. Discuss plan for testing the system / Tianming Zhao / 10 minutes

---

## NEW ACTION ITEMS

---

### Item/Responsible/Due Date

1. Class Diagram / Vu Hoang / 12/05/2018
2. Sequence Diagram / Josh, Vu / 13/05/2018
3. Test Data / Reece, Josh / 16/05/2018

# Team Meeting - Week 10

Date 10/05/2018

Time: 12:00PM

Location: Online

<b>Meeting called by:</b>	Tianming Zhao	<b>Type of meeting:</b>	Online
<b>Timekeeper:</b>	Joshua Cook	<b>Note taker:</b>	Joshua Cook

**Attendees:** Tianming Zhao, Reece Henriquez, Vu Hoang, Joshua Cook

**Please read:** Functionality Progress Report

## Minutes

**Agenda item:** Discuss Documentation for Submission      **Presenter:** Tianming

**Discussion:**

Tianming went through the project specification, outlining what is yet to be completed.

**Conclusions:**

Class and Sequence Diagrams needed to be developed, Taiga needed to be updated, and the documentation needed to be integrated into the final report.

Action items	Person responsible	Deadline
✓ Class Diagram	Vu Hoang	12/05/2018
✓ Sequence Diagrams	Vu, Joshua	13/05/2018
✓ Update Taiga Documentation	Vu Hoang	15/05/2018
✓ Integrate Documentation into Report	Joshua Cook	20/05/2018

**Agenda item:** Discuss Plan on Front/Back end      **Presenter:** Tianming Zhao

**Discussion:**

The team discussed what functionality is still yet to be implemented into the system, and an outline of when they needed to be completed.

**Conclusions:**

A timetable of requirement due dates was created in-order to develop the system on time.

**Agenda item:** Discuss Plan for Testing      **Presenter:** Tianming Zhao

**Discussion:**

The team discussed suitable testing methods for the system to ensure it is functioning to standard.

**Conclusions:**

A comprehensive testing plan with data is to be developed to enable the team to thoroughly test the system.

Action items	Person responsible	Deadline
✓ Test Data	Reece, Josh	16/05/2018

# MEETING AGENDA – WEEK 11

---

## MEETING INFORMATION

**Objective:** *Ensure project is on schedule and ready for final submission*

**Date:** 16/05/2018                   **Location:** Online  
**Time:** 5:00 PM                      **Meeting Type:** Text Conference  
**Called By:** Tianming Zhao  
**Timekeeper:** Joshua Cook           **Note Taker:** Joshua Cook  
**Attendees:** Tianming Zhao, Reece Henriquez, Vu Hoang, Joshua Cook, Chloe Sun

---

## AGENDA ITEMS

Item/Presenter/Time Allotted

1. Review Current Documentation / Tianming, Joshua / 15 minutes
2. Review Current System / Tianming / 10 minutes

---

## NEW ACTION ITEMS

Item/Responsible/Due Date

1. Final Report / Joshua Cook / 21/05/2018

# Team Meeting - Week 11

Date 16/05/2018

Time: 5:00PM

Location: Online

<b>Meeting called by:</b>	Tianming Zhao	<b>Type of meeting:</b>	Online
<b>Timekeeper:</b>	Joshua Cook	<b>Note taker:</b>	Joshua Cook

**Attendees:** Tianming Zhao, Reece Henriquez, Vu Hoang, Joshua Cook

**Please read:** N/A

**Please bring:** N/A

## Minutes

**Agenda item:** Review Current Documentation

**Presenter:** Tianming, Josh

**Discussion:**

Tianming and Josh went through the current state of the report documents, making a note of what needed to be refined, and what to / not include in the final report.

**Conclusions:**

All the required work to be completed was noted and Tianming distributed the tasks to the team.

Action items	Person responsible	Deadline
✓ Final Refine Database Diagram	Tianming Zhao	20/05/2018
✓ Sequence Diagrams Refine	Joshua Cook	20/05/2018
✓ Refine and Document Testing Output	Reece Henriquez	20/05/2018
✓ Complete Final Report	Joshua Cook	21/05/2018

**Agenda item:** Review Current System

**Presenter:** Tianming Zhao

**Discussion:**

Tianming outlined the remaining tasks left for the system, and what the highest priorities for the system are.

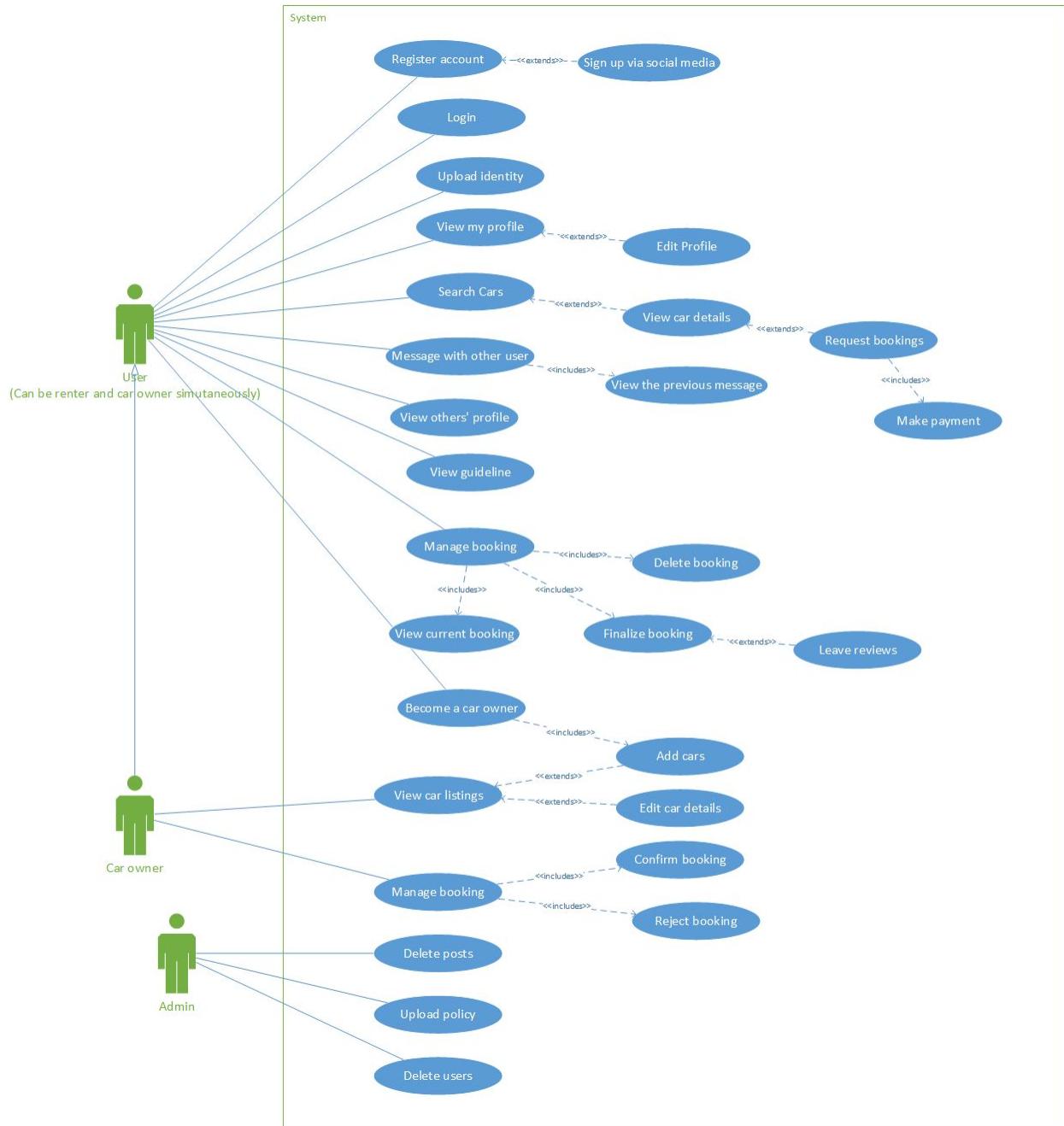
**Conclusions:**

The team noted what needed attention and continued developing the system.

Action items	Person responsible	Deadline
✓ Implement Payment Functionality	Tianming Zhao	18/05/2018
✓ Finalise Chat System	Reece Henriquez	18/05/2018
✓ Continue Testing	Reece Henriquez	20/05/2018

# Analysis & Design

## Use Case Diagram



## Use Case Descriptions

<b>Name:</b> Search cars <b>ID:</b> 1
<b>Stakeholders and goals:</b> Renter - wants to search for a car that they would like to rent
<b>Description:</b> Renter wants to search all cars that meets certain criterias to rent.
<b>Actors:</b> Renter
<b>Trigger:</b> Renters fill in the criteria form and then click search button
<p><b>Normal flow:</b></p> <ol style="list-style-type: none"> <li>1. Renter fills the criteria in the search form</li> <li>2. System returns a list of car information and displays them on the screen</li> <li>3. Renter clicks one of the car picture: The ‘View car detail’ use case is performed</li> <li>4. Repeat steps 3 for each car</li> <li>5. Repeat step 1-4</li> <li>6. End</li> </ol>
<p><b>Sub-flows:</b> Renter fills the criteria in the search form:</p> <ol style="list-style-type: none"> <li>1. (optional) Renter enters the locations of the car</li> <li>2. (optional) Renter enters the price range</li> <li>3. (optional) Renter enters other restrictions</li> </ol>
<p><b>Alternative/Exceptional flows:</b></p> <p>1a Invalid input: 1a1 - System displays which inputs are invalid.      1b Cannot connect to the database: 1b1 - System displays ‘Cannot connect to the database’.      1c Renter does not fill in the criteria and view the default displays of cars. Just have step 3,4</p>

<b>Name:</b> View car detail <b>ID:</b> 2
<b>Stakeholders and goals:</b> Renter - wants to check the details of a car before booking. Car owner - wants to see their posts in view mode. Admin - wants to see if the car information is false or invalid or spam. (to delete)
<b>Description:</b> User selects a vehicle post to display, vehicle page shows all information about the specified vehicle.
<b>Actors:</b> Renter, Car owner, admin
<b>Trigger:</b> A post of a car is clicked.
<b>Normal flow:</b> 1. A user clicks a post of a car. 2. The system returns the detailed information of the car and displays them on the screen. 3. (optional) A user clicks 'Book this car' button : 'Book request' use case is performed. 4. End
<b>Sub-flows:</b> None
<b>Alternative/Exceptional flows:</b>

<b>Name:</b> Request booking <b>ID:</b> 3
<b>Stakeholders and goals:</b> Renter - wants to book a car to rent
<b>Description:</b> Renters want to rent a car. They need to first book it, the request will be sent to the car owner, if the car owner accepted it, the renter successfully get the car.
<b>Actors:</b> Renter
<b>Trigger:</b> Renter clicks book button when viewing a car listing in detail.
<b>Normal flow:</b> 1. Renter clicks 'Book this car'. 2. System sends a booking request to the car owner who can choose to accept or reject 3. End.
<b>Sub-flows:</b> None
<b>Alternative/Exceptional flows:</b> 3a. The car has become unavailable before the user sends the booking request 3a1. The user receives a message displaying "Vehicle not available for selected dates"

<b>Name:</b> Manage Booking <b>ID:</b> 4
<b>Stakeholders and goals:</b> Car owner - wants to manage a list of bookings from potential renters
<b>Description:</b> When an owner views the booking list, they can confirm, reject, or finalise bookings.
<b>Actors:</b> Car owner
<b>Trigger:</b> A car owner clicks a button 'Booking list' on the top of the page
<p><b>Normal flow:</b></p> <ol style="list-style-type: none"> <li>1. A car owner clicks the button 'Booking list' on the profile page.</li> <li>2. The system returns a list of bookings for their cars and displays them on the screen.</li> <li>3. (optional) The car owner clicks 'Reject' for one of the booking request: Use case 'Reject booking' is performed.</li> <li>4. (optional) The car owner clicks 'Confirm' for one of the booking request: Use case 'Confirm booking' is performed.</li> <li>5. (optional) The car owner clicks 'Finalise' for one of the booking (at status: ready to close): Use case 'Finalise booking' is performed.</li> <li>6. Repeat 3 - 5</li> <li>7. End.</li> </ol>
<b>Sub-flows:</b>
<p><b>Alternative/Exceptional flows:</b></p> <p>4a No booking currently: 4a1 - System displays 'Currently no booking'.</p>

<b>Name:</b> Reject Booking <b>ID:</b> 5
<b>Stakeholders and goals:</b> Car owner - Car owner rejects one of the booking request
<b>Description:</b> Car owner is viewing all the booking details, he wants to reject one of the booking requests.
<b>Actors:</b> Car owner
<b>Trigger:</b> The car owner is viewing the list of bookings and he clicks 'Reject' for one of the booking.
<p><b>Normal flow:</b></p> <ol style="list-style-type: none"> <li>1. A car owner clicks 'Reject' for one of the booking.</li> <li>2. The system asks the user to confirm the action.</li> <li>3. A car owner confirms the action.</li> <li>4. The system removes the request and returns a message to the potential renter (who requests it)</li> <li>5. End.</li> </ol>
<b>Sub-flows:</b> None
<p><b>Alternative/Exceptional flows:</b></p> <p>5a The car owner does not confirm the action, he/she cancels the action: to 5 End.</p>

<b>Name:</b> Confirm booking <b>ID:</b> 6
<b>Stakeholders and goals:</b> Car owner - Car owner confirm a booking request.
<b>Description:</b> Car owner is viewing all the booking details, he wants to confirm one of the booking requests.
<b>Actors:</b> Car owner
<b>Trigger:</b> The car owner clicks ‘Confirm booking’ for one of the booking on the list.
<p><b>Normal flow:</b></p> <ol style="list-style-type: none"> <li>1. Car owner clicks ‘Confirm Booking’ for a desired booking</li> <li>2. Car owner confirms the ‘Confirm Booking’ action when prompted.</li> <li>3. End.</li> </ol>
<b>Sub-flows:</b> None
<p><b>Alternative/Exceptional flows:</b></p> <p>2a The car owner does not confirm the action, he/she rejects the action: go to reject booking 5 End.</p>

<b>Name:</b> Message Another User <b>ID:</b> 7
<b>Stakeholders and Goals:</b> Car Renter or Car Owner: To message another user
<b>Description:</b> A user wants to message another user of UHire, to respond to a message or discuss details of an existing or future booking.
<b>Actors:</b> Car Renter or Car Owner (Users)
<b>Trigger:</b> User would like to enquire information about existing or potential booking.
<p><b>Normal Flow:</b></p> <ol style="list-style-type: none"> <li>1. Car Renter is currently viewing a vehicle which they are interested in.</li> <li>2. Click on message button to directly message user</li> <li>3. System opens message box to send to requested user</li> <li>4. Fill out message field with desired message</li> <li>5. Click “Send” button to send private message to user</li> <li>6. System sends message and notification to other user</li> <li>7. END</li> </ol>
<b>Sub-Flow:</b>
<p><b>Alternative/ Exceptional Flows:</b></p> <p>1a1. User clicks on Messages icon in the menu bar (accessible from anywhere)      1a2. User selects conversation with the desired recipient      1a3. User fills out message field and clicks “Send” as normal flow.      2a. User selects profile page of desired user, proceeds to selects “Message User” from profile page.</p>

<b>Name:</b> View Previous Messages <b>ID:</b> 8
<b>Stakeholders and Goals:</b> Car Renter or Car Owner (Users) – View previously sent and received messages
<b>Description:</b> A user would like to see a message that was previously sent, typically to recall discussed booking details
<b>Actors:</b> Users
<b>Trigger:</b> User would like to see a response to their message
<b>Normal Flow</b> 1. User selects Messages icon on menu bar 2. System loads messages associated with account 3. User selects conversation they would like to view 4. System loads message history with that conversation 5. User sees desired message, or scrolls through conversation history until found 6. END
<b>Sub-Flow:</b>
<b>Alternative/ Exceptional Flows:</b>

<b>NAME:</b> Finalise Booking <b>ID:</b> 9
<b>Stakeholders and Goals:</b> Owner - Confirm that the booking has concluded and receive the payment being held by the system. Renter - Confirmed that the vehicle was returned
<b>Description:</b> Owner needs to confirm that the renting process has concluded
<b>Actors:</b> Owner
<b>Trigger:</b> Renter returns the vehicle to the owner, owner ensures that the vehicle is in order and finalises the booking to receive payment.
<b>Normal Flow</b> 1. Renter returns car to specified return point on return date. 2. Owner navigates to booking page, clicks on icon to finalise booking, confirming vehicle was returned. 3. System updates that vehicle was returned, and prompts user for a review 4. Owner proceeds to leave feedback for the renter. (Use case Provide Feedback) 5. END
<b>Sub-Flow:</b> 4a. Renter will also leave feedback for owner, use case Provide Feedback.
<b>Alternative/ Exceptional Flows:</b> 1a. Renter is unsatisfied with vehicle, clicks on “report issue” on booking page before booking is finalised.

1b. Owner tries to finalise booking before return date
• Owner receives error “Booking has not reached end date”
2a. Owner is unsatisfied with the return of vehicle, clicks on “report an issue” on the booking page.
4a. Owner doesn’t wish to leave feedback until later, leaves feedback via the booking page.

<b>Name:</b> View my profile <b>ID:</b> 10
<b>Stakeholders and Goals:</b> User -To be able to view their own profile, to view their personal information and listings
<b>Description:</b> A user can view their own profile to review their personal information and quickly access their listings
<b>Actors:</b> User
<b>Trigger:</b> A user has an account and wants to review their own details.
<b>Normal Flow:</b>
1. A user wants to review their profile, so they click their profile page 2. System opens and displays their profile page 3. User scrolls through the page to view desired information 4. END
<b>Sub-Flows</b> 2a can click on view listings calls use case View Listings
<b>Alternative/Exceptional Flows:</b>

<b>Name:</b> Edit profile <b>ID:</b> 11
<b>Stakeholders and Goals:</b> User – To be able to easily modify their personal details so other users can view their most recent details
<b>Description:</b> A user wants to edit their profile because their details are out of date or don’t exist. So other users can view their details.
<b>Actors:</b> User
<b>Trigger:</b> A user decides that their details are out of date or need to be added
<b>Normal Flow:</b>
1. A user decides that their details are out of date 2. They click their profile link, to view their profile 3. System opens the profile page 4. They then click modify and can edit any of the displayed fields that represent their profile 5. The user clicks save, to finalise 6. System updates the provided information and provides confirmation 7. End
<b>Sub-Flows</b>
<b>Alternative/Exceptional Flows:</b>
1a. A user starts using the app and creates account for first time and is prompted to edit their profile

<b>Name: Edit Listed Vehicles ID: 12</b>
<b>Stakeholders and Goals:</b>
Users (Owner) – To be able to modify listings so then information is most up to date
<b>Description:</b> A user wants to edit current listings, this may include editing existing information, or removing the vehicle from the server.
<b>Actors:</b> User (Owner)
<b>Trigger:</b> An owner wants to change outdated information about their vehicle, or remove the vehicle from their account.
<b>Normal Flow:</b>
<ol style="list-style-type: none"> <li>1. Owner decides to update vehicle information.</li> <li>2. Owner navigates to personal profile page.</li> <li>3. Owner clicks view vehicles to see all currently listed vehicles.</li> <li>4. System provides all vehicles owned by the user</li> <li>5. Clicks button to edit vehicles.</li> <li>6. Re-Enters information in appropriate fields, and saves the updated information.</li> <li>7. System updates new information and provides confirmation</li> <li>8. End.</li> </ol>
<b>Sub-Flows</b>
<b>Alternative/Exceptional Flows:</b>
5a Owner may select to delete the entire vehicle instead.

<b>Name: Add Vehicle ID: 13</b>
<b>Stakeholders and Goals:</b>
User – Upload a new vehicle to their account, so it will be available for renting by other users.
<b>Description:</b> A user uploads a vehicle to their account to create a listing visible to other users, once a user uploads a car, they become a Vehicle Owner User.
<b>Actors:</b> User
<b>Trigger:</b> A owner wants to upload a vehicle to their account
<b>Normal Flow:</b>
<ol style="list-style-type: none"> <li>1. User goes to their profile</li> <li>2. Selects the option to upload a vehicle</li> <li>3. User provides required information &amp; identification, and proof of vehicle</li> <li>4. System verifies the user, vehicle, and provides confirmation that the upload was complete</li> <li>5. End.</li> </ol>
<b>Sub-Flows</b>
<b>Alternative/Exceptional Flows:</b>
4a. Insufficient or invalid information is provided, system reports error to user in which case they can cancel or retry.

<b>Name: Upload Identity ID: 14</b>
<b>Stakeholders and Goals:</b> Users (Owners) – To be able to upload their identity and credit to their accounts to be verified so other users can see that they have been can be verified and checked
<b>Description:</b> A user wants to be able to rent and rent out their cars. They must upload their identity, so it can be verified.
<b>Actors:</b> Users
<b>Trigger:</b> A user wants to become a renter or an owner
<b>Normal Flow:</b> <ol style="list-style-type: none"> <li>1. User creates an account (ID: 22)</li> <li>2. User wishes to verify account</li> <li>3. System prompts user to provide appropriate information</li> <li>4. User provides personal information, and uploads required images</li> <li>5. System checks that all the provided information was sufficient, returns confirmation</li> <li>6. END</li> </ol>
<b>Sub-Flows</b>
<b>Alternative/Exceptional Flows:</b> 4a user verification is negative return to 2

<b>Name: View Guideline Information ID: 15</b>
<b>Stakeholders and Goals:</b> Users – To have an informational area, where users can gain an understanding of system processes
<b>Description:</b> A user needs information about the system and on how aspects of transactions and marketing processes work
<b>Actors:</b> Users
<b>Trigger:</b> A user needs information
<b>Normal Flow:</b> <ol style="list-style-type: none"> <li>1. A user clicks the information tab area</li> <li>2. System provides the user with guideline information</li> <li>3. A user can scroll and read through the various related topics</li> <li>4. END</li> </ol>
<b>Sub-Flows</b>
<b>Alternative/Exceptional Flows:</b>

<b>Name: Provide Reviews ID: 16</b>
<b>Stakeholders and Goals:</b>
Users - Leave a review and rating on their booking experience with another user.
<b>Description:</b> Once a booking has finished, users may leave feedback on each other to provide a public review of their experience.
<b>Actors:</b> Users
<b>Trigger:</b> A booking has concluded and a user would like to leave feedback
<b>Normal Flow:</b>
<ol style="list-style-type: none"> <li>1. When the booking is complete a user has an option to leave feedback</li> <li>2. A user navigates to the booking page, and clicks the “Leave Review” button.</li> <li>3. System prompts the user to enter a detailed review</li> <li>4. The user fills in required information, such as rating and comments.</li> <li>5. User selects “finished” and sends the review to the other user.</li> <li>6. System checks whether a sufficient response has been received, then adds this review to the user/ vehicle profile</li> <li>7. END</li> </ol>
<b>Sub-Flows</b>
<b>Alternative/Exceptional Flows:</b>
2a Once an owner has finalised a booking they are prompted to leave a review.

<b>Name: Delete Users ID: 17</b>
<b>Stakeholders and Goals:</b>
Admin – Admins can delete users who are behaving inappropriately or in an unprofessional manner, in order to maintain the systems integrity.
Users – Will get deleted if they break the terms of service.
<b>Description:</b> Admin discovers a user who has repeatedly been causing issues, or behaves in a manner that deserves immediate termination from the service.
<b>Actors:</b> Admin
<b>Trigger:</b> An admin discovers a user breaking terms of service
<b>Normal Flow:</b>
<ol style="list-style-type: none"> <li>1. Admin is monitoring listings/ reports</li> <li>2. Admin discovers a user that is acting inappropriately</li> <li>3. Admin clicks on user to view profile</li> <li>4. Admin clicks terminate account.</li> <li>5. System removes the account information from the database</li> <li>6. END</li> </ol>
<b>Sub-Flows</b>
<b>Alternative/Exceptional Flows:</b>

<b>Name: Delete Post ID: 18</b>
<b>Stakeholders and Goals:</b>
Admin – Admins can delete posts which are inappropriate or misleading.
Owner - Vehicle will get removed from the service
<b>Description:</b> Admin discovers a post that is inappropriate, misleading, or has received numerous complaints from users.
<b>Actors:</b> Admin
<b>Trigger:</b> An admin discovers a post that breaks terms of service.
<b>Normal Flow:</b>
<ol style="list-style-type: none"> <li>1. Admin is monitoring listings/ reports</li> <li>2. Admin discovers a listing that is breaking the terms of service.</li> <li>3. Admin clicks “remove post”</li> <li>4. End.</li> </ol>
<b>Sub-Flows</b>
<b>Alternative/Exceptional Flows:</b>

<b>Name: View others profile ID: 19</b>
<b>Stakeholders and Goals:</b>
Car Renters - Can view other users' profiles who are associated to car listings, so they can get better information about other users
<b>Description:</b> A user is browsing through listings and sees a candidate vehicle so they open the listings or directly click the users name to view their profile for information on the user.
An owner receives a booking request from a user, then proceeds to open their profile page for more information on the renter.
<b>Actors:</b> Car Renters, Car Owner
<b>Trigger:</b> A user would like to receive more information about another specific user.
<b>Normal Flow:</b>
<ol style="list-style-type: none"> <li>1. User is browsing vehicle listings</li> <li>2. User clicks a vehicle owners' username for a vehicle of interest</li> <li>3. System retrieves the owners' account and displays it</li> <li>4. User scrolls through the profile page to view information including rating to make a judgement whether to follow through with a booking.</li> <li>5. END</li> </ol>
<b>Sub-Flows</b>
<b>Alternative/Exceptional Flows:</b>
1a Car Owner would instead click a user profile from the booking page, continue step 3
2a A user may instead access the users profile from the listings rather than the browsing page

<b>Name: Delete Booking ID: 20</b>
<b>Stakeholders and Goals:</b>
Renter - no longer wants a booking they made so they can delete it, receiving a refund.
<b>Description:</b> A renter has made a booking but no longer requires it. They can delete the booking
<b>Actors:</b> Renter
<b>Trigger:</b> User no longer wants a booking
<b>Normal Flow:</b>
<ol style="list-style-type: none"> <li>1. User makes a booking (ID: 3)</li> <li>2. User does not need booking anymore</li> <li>3. User goes to manage the booking (ID: 4)</li> <li>4. User selects to delete the booking</li> <li>5. System removes the booking, and makes the vehicle available again</li> <li>6. END</li> </ol>
<b>Sub-Flows</b>
<b>Alternative/Exceptional Flows:</b>
4a. User tries to cancel booking outside the “cooling-off” period, user confirms there will be a fee involved with cancelling.

<b>Name: View Car Listings ID:21</b>
<b>Stakeholders and Goals:</b>
Owner - Views their own car listings, to check all information is up to date
<b>Description:</b> An owner wants to view all their current listing to check that information is up to date.
<b>Actors:</b> Owner
<b>Trigger:</b> An owner decides their listings information are out of date
<b>Normal Flow:</b>
<ol style="list-style-type: none"> <li>1. Owner opens their profile</li> <li>2. Scrolls to see listings</li> <li>3. Can click on each to view</li> </ol>
<b>Sub-Flows</b>
<b>Alternative/Exceptional Flows:</b>

<b>Name: Register Account ID: 22</b>
<b>Stakeholders and Goals:</b>
Renters - To be able to create an account to be apart of the system, either through normal means or social media
<b>Description:</b> A user wants to begin using the system so they create an account so they can either rent or rent out their own cars

<b>Trigger:</b> A user wants to use the system
<b>Normal Flow:</b>
1. User opens website 2. Clicks on register 3. System prompts user to fill in required information 4. Fills in details and confirms submit 5. System verifies information, provides confirmation of created account. 6. END
<b>Sub-Flows</b>
<b>Alternative/Exceptional Flows:</b> 3a Creates account with social media, go to ID:23 5a System returns invalid information, go to flow 3

<b>Name: Register via social media ID:23</b>
<b>Stakeholders and Goals:</b> Renters - To be able to create an account to be apart of the system by using social media account
<b>Description:</b> A user wants to begin using the system so they create an account so they can either rent or rent out their own cars. Their account is created with an already existing social media account.
<b>Actors:</b> Renter
<b>Trigger:</b> A user wants to use the system
<b>Normal Flow:</b>
1. User opens website 2. Clicks on register 3. Clicks connect with social media account 4. System prompts login from social media platform 5. Logins in to social media and approves integration 6. System verifies login was success, creates account 7. END
<b>Sub-Flows</b>
<b>Alternative/Exceptional Flows:</b>

<b>Name: Login to account ID:24</b>
<b>Stakeholders and Goals:</b> Renters - To login into their account to view all that has occurred involving their account
<b>Description:</b> A user wants to continue using the system with their last activities so they login into the system with their username and password that they generated
<b>Actors:</b> Renter, Owners
<b>Trigger:</b> A user wants to use the system

<b>Normal Flow:</b>
1. User opens website
2. Clicks on login
3. Enters details, clicks login
4. System provides confirmation that the user details were correct
5. END
<b>Sub-Flows</b>
<b>Alternative/Exceptional Flows:</b>
4a. User details are wrong, System reports error loops back to enter details

<b>Name:</b> Manage Booking <b>ID:</b> 25
<b>Stakeholders and Goals:</b>
Owner/Renters - To be able to manage bookings that have been requested, confirmed or in progress
<b>Description:</b> An owner or renter wants to manage bookings that have been requested, confirmed or in progress
<b>Actors:</b> Owners, Renters
<b>Trigger:</b> Owner/Renter want to check information on a booking, or begin the finalization process
<b>Normal Flow:</b>
1. User selects booking manager at the top of the screen
2. System retrieves existing bookings and requests associated with the account
3. (Optional) Owner can confirm or reject booking: Do use case confirm book or reject booking
4. (Optional) Renter can finalize a booking: Do use case finalise booking
5. (Optional) Renger can delete existing booking: Do use case delete booking
6. END
<b>Sub-Flows</b>
<b>Alternative/Exceptional Flows:</b>

<b>Name:</b> View Current Bookings <b>ID:</b> 26
<b>Stakeholders and Goals:</b>
Owner/Renters - To view bookings that have been made, confirmed and in process.
<b>Description:</b> A user wants to view bookings that are currently a point of interest either that they have been confirmed, pending approval, or in the midst of occurring
<b>Actors:</b> Renter, Owners
<b>Trigger:</b> A user wants to view current bookings that have been made
<b>Normal Flow:</b>
1. User opens booking manager (ID: 25)
2. Sees all available booking
3. (Optional) Finalise, Confirm, Reject, Delete each goes to their corresponding use case
4. END

<b>Sub-Flows</b>
<b>Alternative/Exceptional Flows:</b>

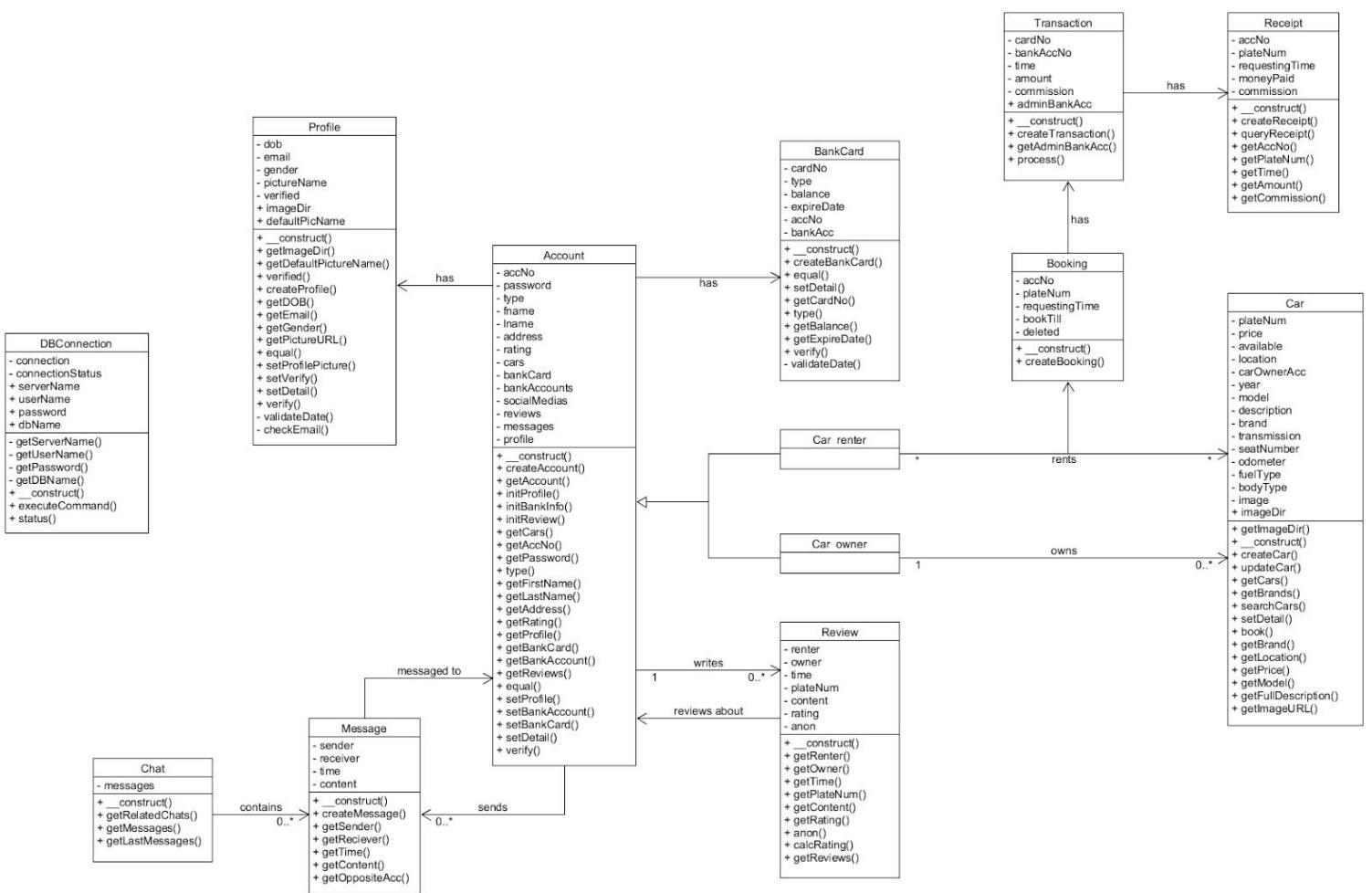
<b>Name:</b> Confirm Bookings <b>ID:</b> 27
<b>Stakeholders and Goals:</b>
Owner - To confirm a requested booking
<b>Description:</b> A renter wants to rent a car so they send a booking request ticket to the owner, the owner sees this in the booking manager and can choose to accept or reject.
<b>Actors:</b> Renter, Owners
<b>Trigger:</b> A owner sees a booking
<b>Normal Flow:</b>
<ol style="list-style-type: none"> <li>1. Owner opens booking manager (ID: 25)</li> <li>2. Sees all requested bookings</li> <li>3. Clicks confirm</li> <li>4. System updates booking status to response</li> <li>5. END</li> </ol>
<b>Sub-Flows</b>
<b>Alternative/Exceptional Flows:</b>
3a. Owners would not like to go through with the booking, selects reject booking instead

<b>Name:</b> Upload policy <b>ID:</b> 28
<b>Stakeholders and goals</b>
System admin – wants to upload system policy
<b>Description</b>
System admin posting a document regarding terms and conditions of using the system, – information such as rules, policies, commission rate, ...etc.
<b>Actors</b>
System admin
<b>Trigger</b>
System admin wants users to have basic information of how the system works, what they can do and should not do. Also this document is to protect the business from expensive lawsuits.
<b>Normal flow</b>
<ol style="list-style-type: none"> <li>1. System admin consults with the team to have a clear understanding of what needs to be included in the T&amp;C.</li> <li>2. System admin writes T&amp;C.</li> <li>3. The T&amp;C is reviewed by professional lawyers and modified where needed.</li> <li>4. System admin upload the T&amp;C to the system.</li> </ol>
<b>Sub-flows</b>
None
<b>Alternative/Exceptional flows</b>

- |   |
|---|
| 1a. System user sets system variables including commission rate, and deposit accounts |
| 2a: System admin cannot write T&C on his own: hires lawyers to write T&C themselves.  |

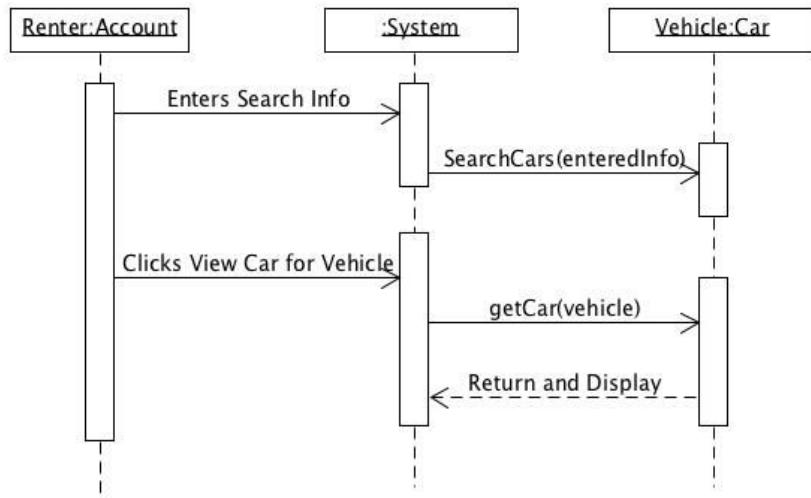
<b>Name:</b> Become a car owner <b>ID:</b> 29
<b>Stakeholders and goals</b> Renter - To upgrade privileges to car owner
<b>Description</b> A user wants to become an owner to profit on vehicles that they rent out. A user makes an account and begins as a renter. To become a car owner they create a car listing with documents proving the existence and correct ownership of vehicle
<b>Actors</b> Renter
<b>Trigger</b> A user wants to rent out cars
<b>Normal flow</b> 1. Registers account 2. Goes to profile 3. Clicks "Become a car owner" 4. Uploads a car's details and registration papers 5. END
<b>Sub-flows</b>
<b>Alternative/Exceptional flows</b>

## Class Diagram

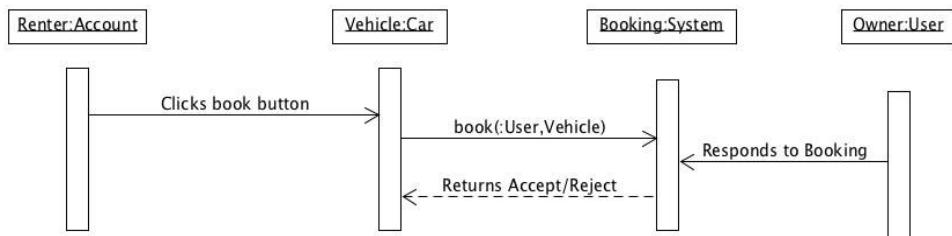


## Sequence Diagrams

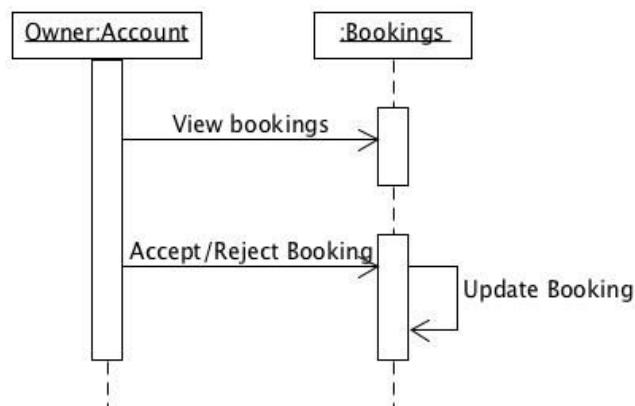
### Use Case 1-2

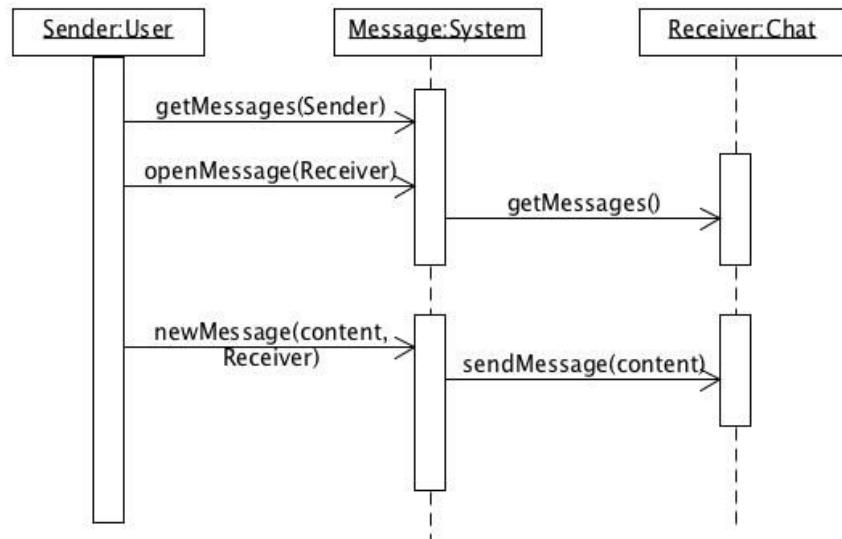
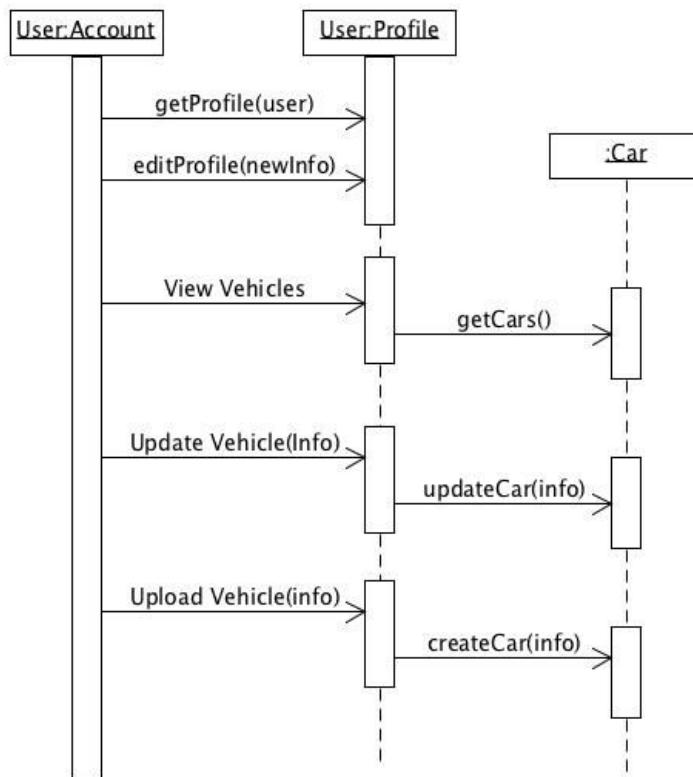


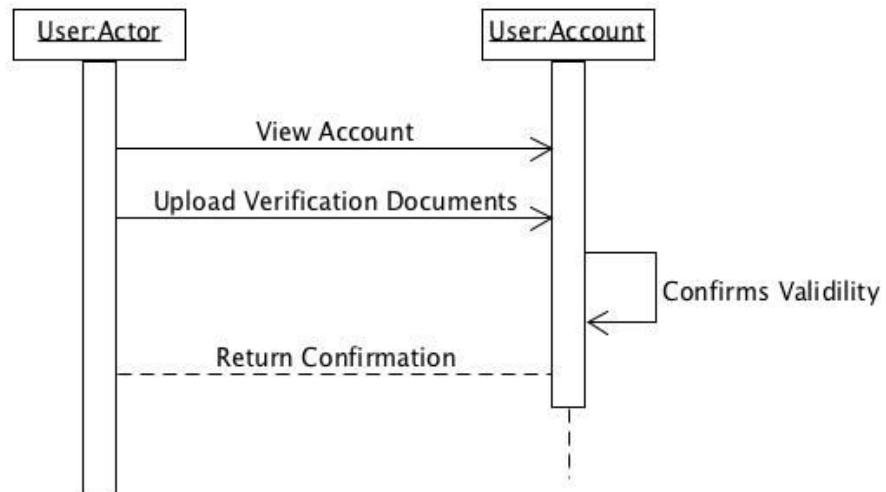
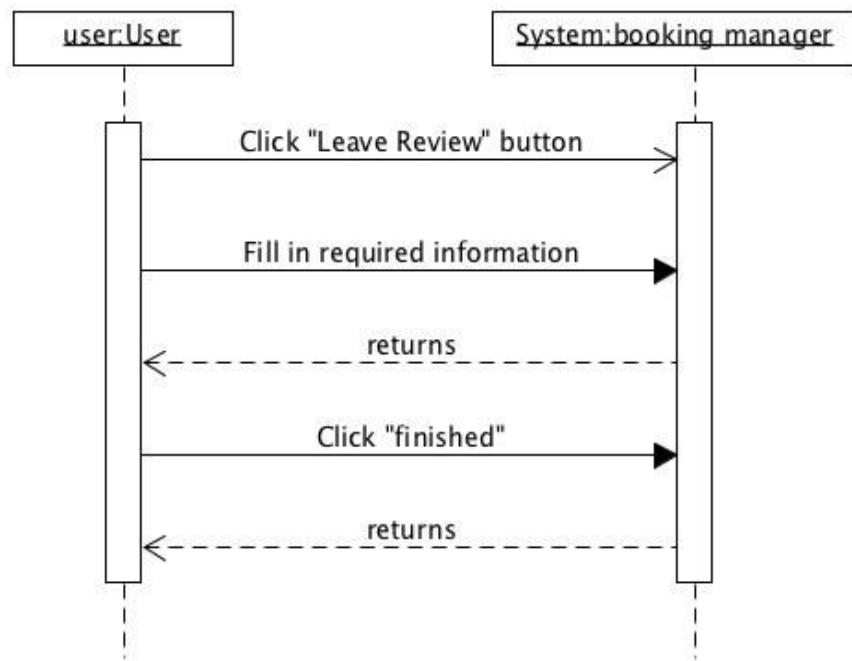
### Use Case 3



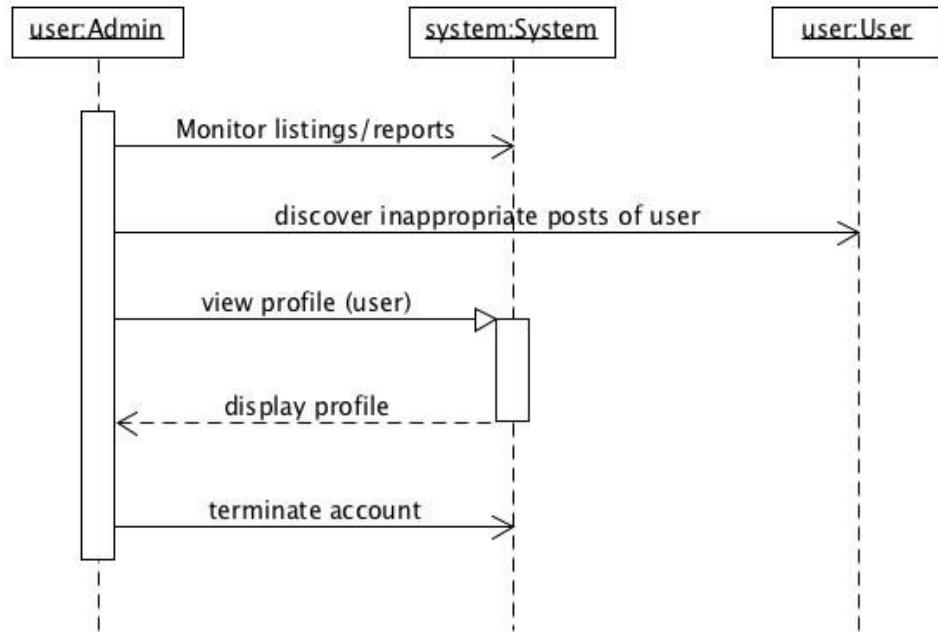
### Use Case 4-6



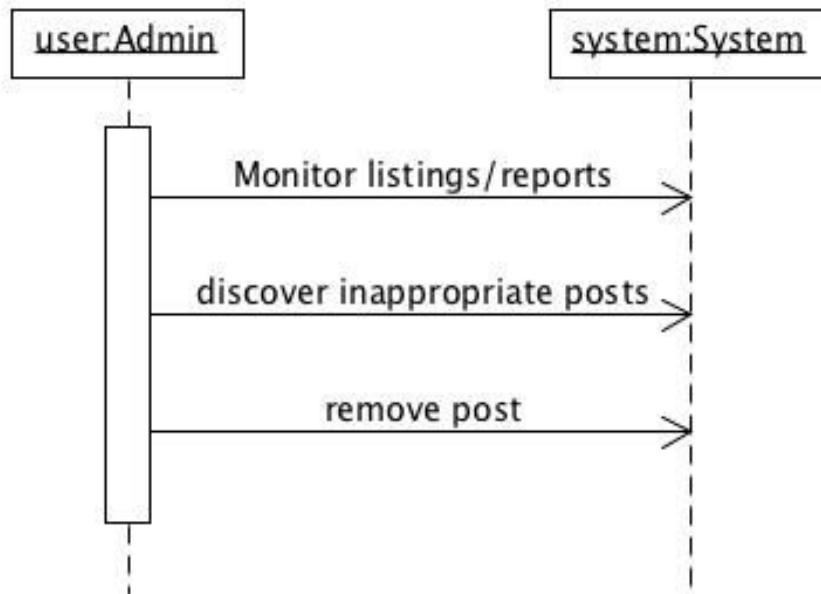
**Use Case 7-9****Use Case 10-13**

**Use Case 14****Use Case 16**

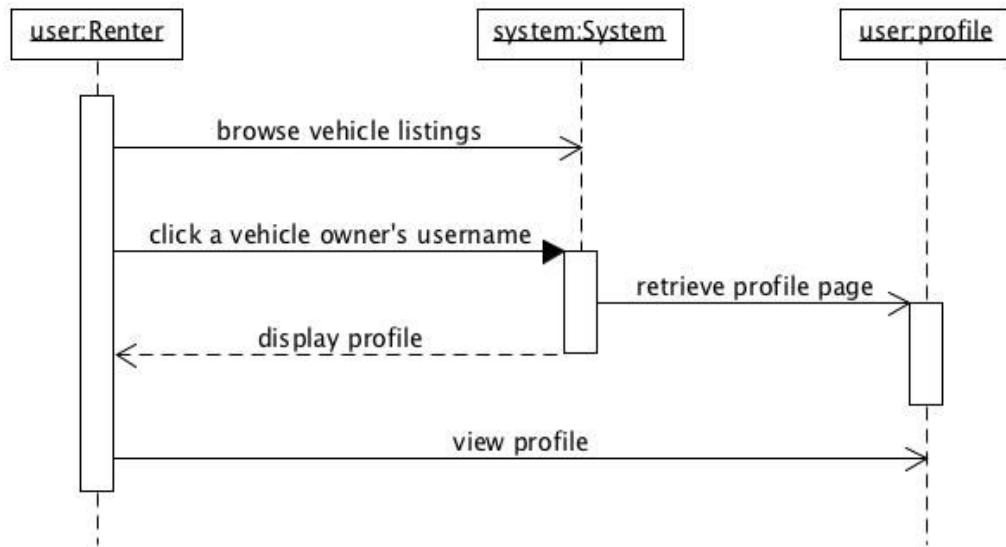
### Use Case 17



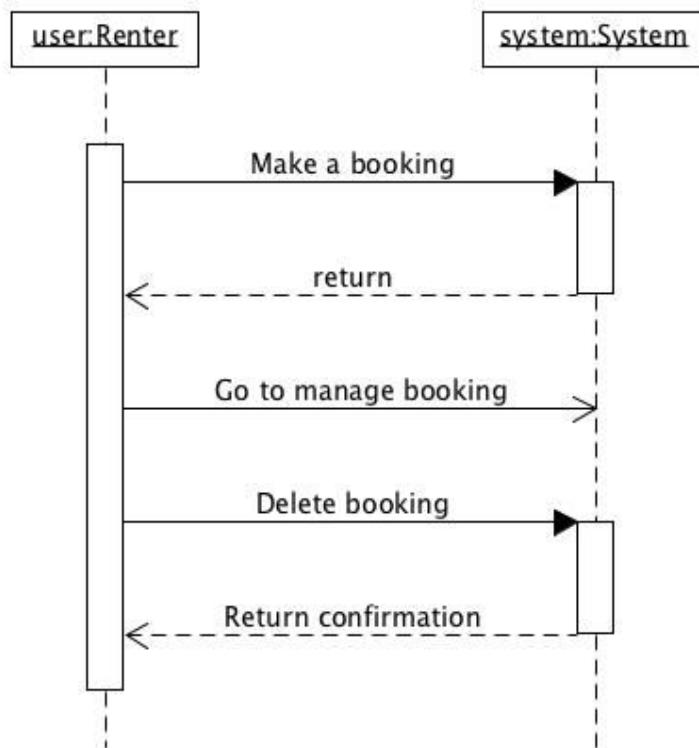
### Use Case 18

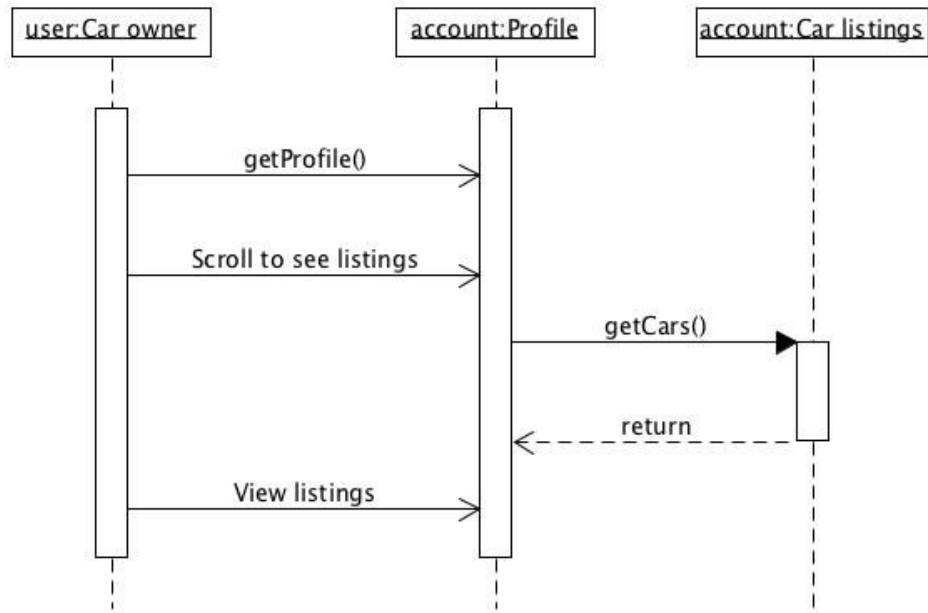
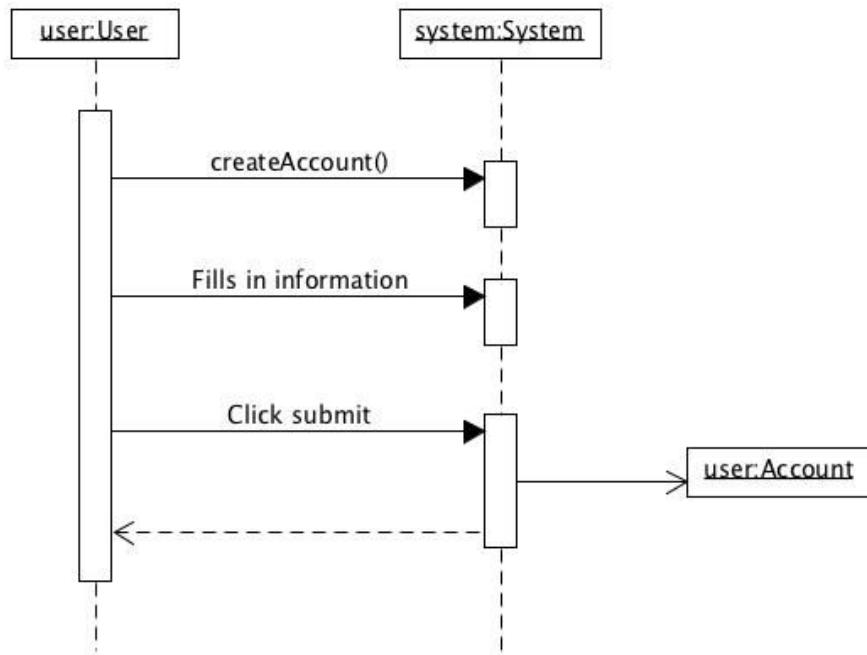


### Use Case 19

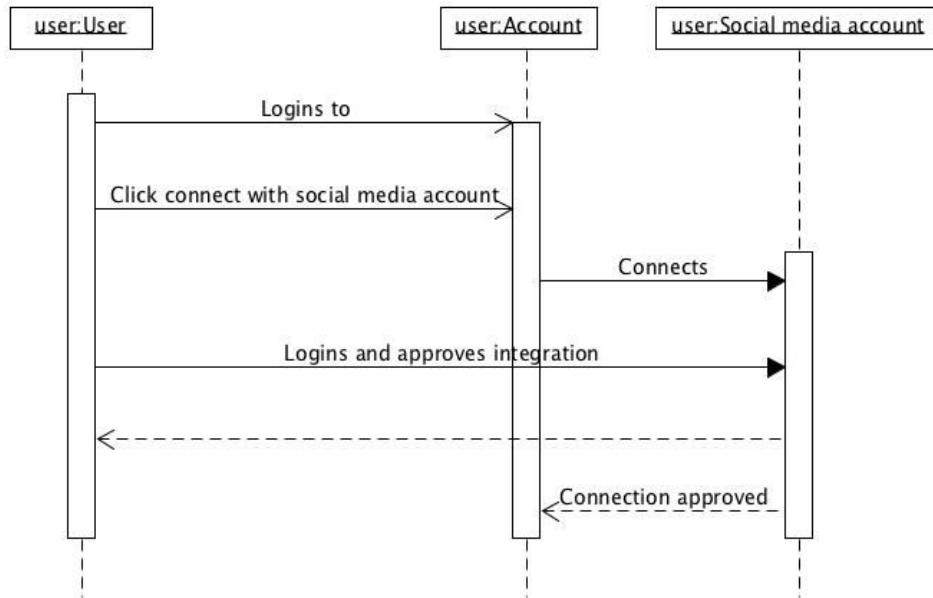


### Use Case 20

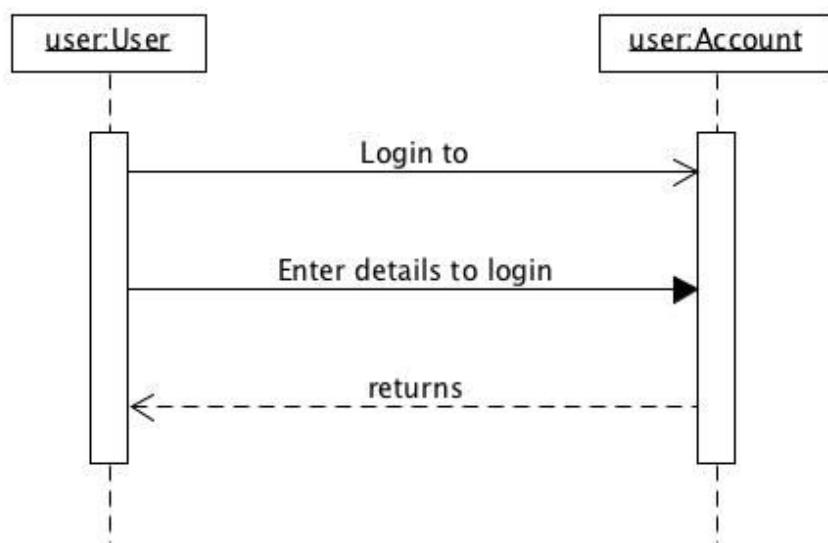


Use Case 21Use Case 22

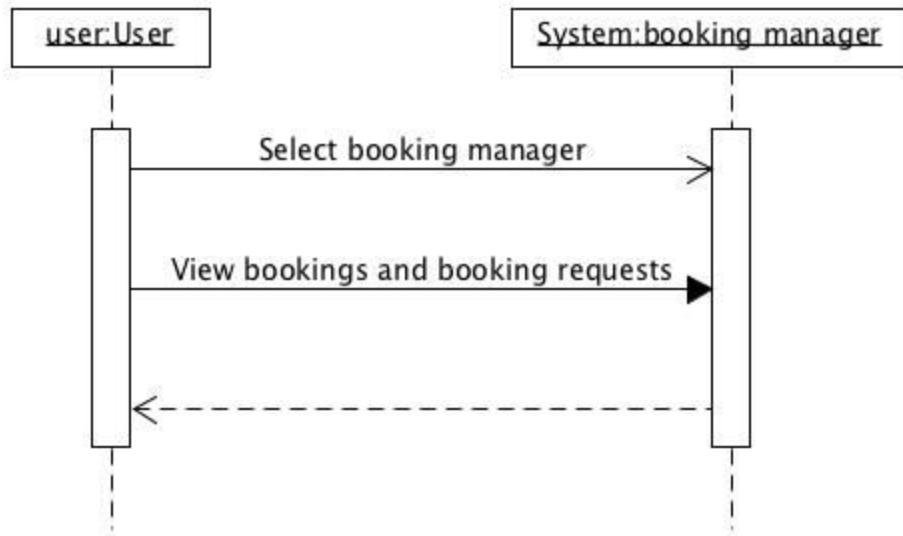
### Use Case 23



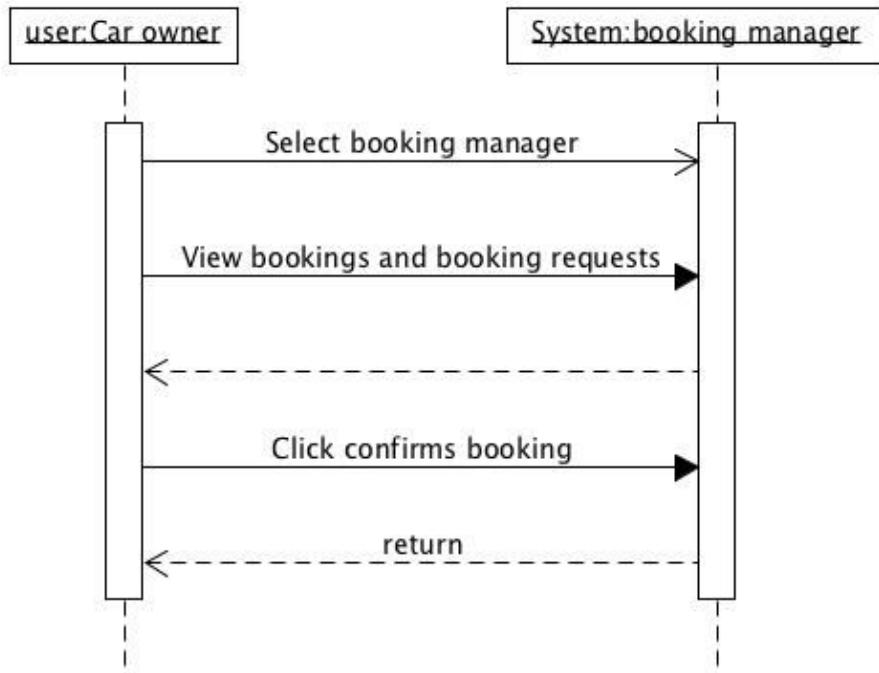
### Use Case 24



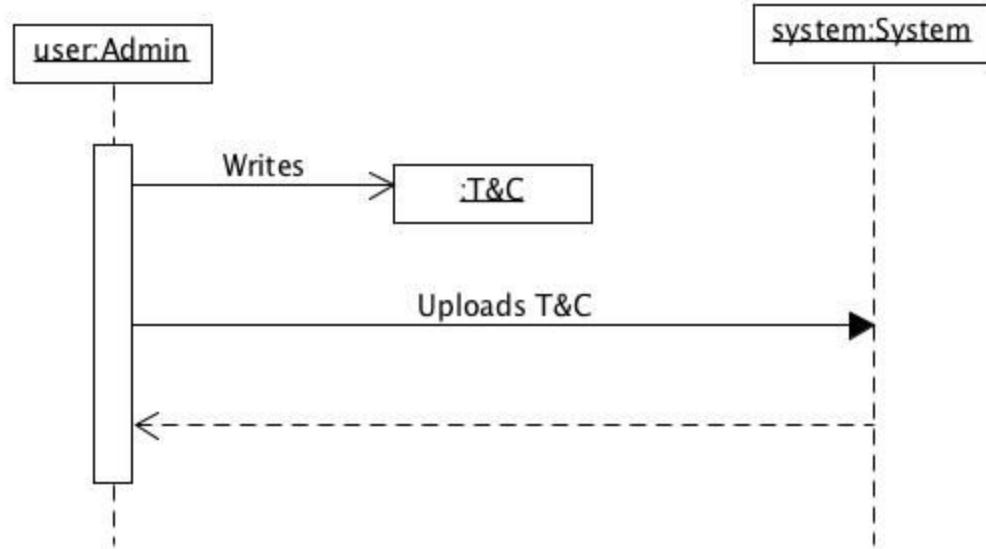
### Use Case 25



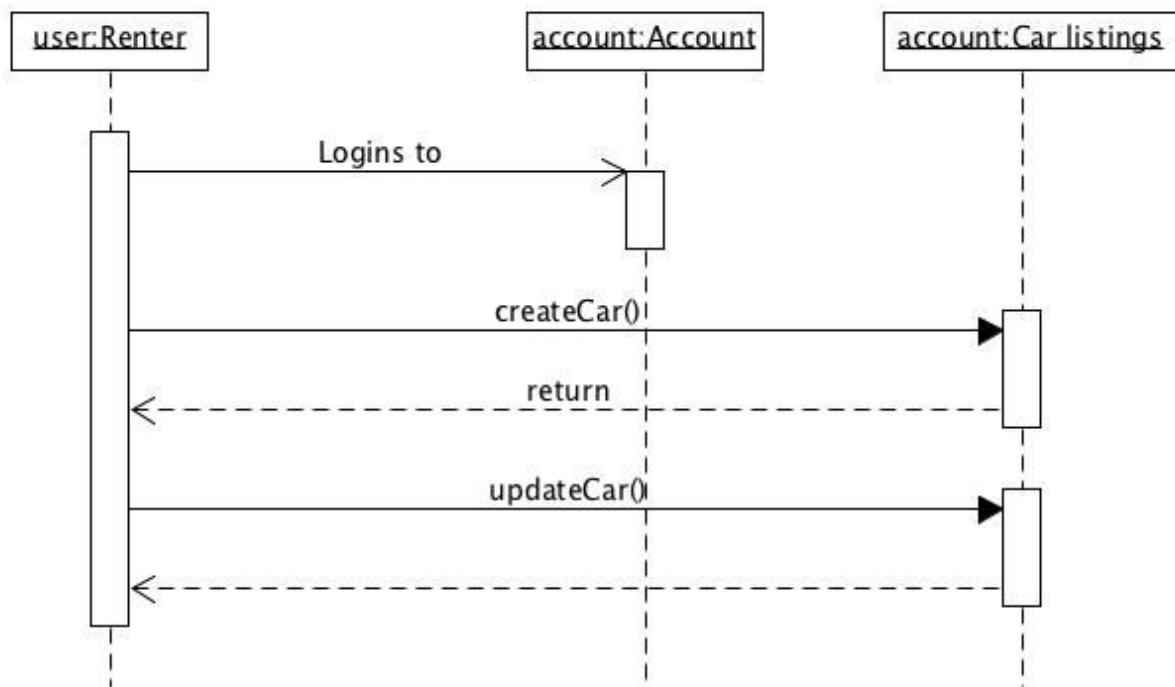
### Use Case 27



### Use Case 28



### Use Case 29



## Testing

Test Data was created using a program to generate over 7000 insert statements for the database, containing user accounts and vehicles. Test data was generated to ensure that every possible field / functionality was supplied to the program.

Once the test data had been generated, use case descriptions were analysed and referenced to create test cases the system needed to be tested against. These test cases are crucial to ensuring that the system will provide sufficient functionality to the users.

Once the output from the test cases was returning positive with no failures or noted problems, the test case could be closed and moved onto the next case.

### Test Cases

24 Login	
Actions	<ul style="list-style-type: none"> <li>a. On Page load</li> <li>b. Login with correct User Details</li> <li>c. Login with incorrect User Details</li> </ul>
Expected Result	<ul style="list-style-type: none"> <li>a. The page will display correctly</li> <li>b. The website will take you to the main page</li> <li>c. The website will display incorrect details</li> </ul>
Pass/Fail	Pass
Observations (if any)	

14 Verify Identity/Upload	
Actions	<ul style="list-style-type: none"> <li>a. A user enters all details, saves and verifies</li> <li>b. A user enters no or partial details saves and verifies</li> </ul>
Expected Result	<ul style="list-style-type: none"> <li>a. Details get saved, verification is okay</li> <li>b. Details get saved, verification is not okay</li> </ul>
Pass/Fail	Pass
Observations (if any)	

10 View my Profile	
Actions	a. Click my account
Expected Result	a. Displays user details
Pass/Fail	Pass
Observations (if any)	

11 Edit my profile	
Actions	a. Clicks edit details
Expected Result	a. Opens up the editing page
Pass/Fail	Pass
Observations (if any)	

1 Search Cars	
Actions	a. Input only 1 value b. Input all values c. Input no values
Expected Result	a. Searches based on a single criteria b. Searches based on all values c. Displays in standard order
Pass/Fail	Pass
Observations (if any)	

2 View Car Details	
Actions	a. Displays information on a selected vehicle
Expected Result	a. Correct information is displayed
Pass/Fail	Pass
Observations (if any)	

3 Request bookings	
Actions	<ul style="list-style-type: none"> <li>a. Make a booking, and enter in days to book for</li> <li>b. Make a booking and enter junk</li> </ul>
Expected Result	<ul style="list-style-type: none"> <li>a. Vehicle will be booked for x days</li> <li>b. Error will display, and you re-input</li> </ul>
Pass/Fail	Pass
Observations (if any)	

19 View others profile	
Actions	<ul style="list-style-type: none"> <li>a. Click person profile head</li> </ul>
Expected Result	<ul style="list-style-type: none"> <li>a. Opens up their profile for viewing</li> </ul>
Pass/Fail	pass
Observations (if any)	

7 Message others	
Actions	<ul style="list-style-type: none"> <li>a. Press message me and send message</li> <li>b. Open messages and message someone</li> <li>c. Send message</li> </ul>
Expected Result	<ul style="list-style-type: none"> <li>a. Opens messages, and can click someone to message</li> <li>b. Opens a new message chat, and can message</li> <li>c. Message is displayed in chat box so other party can view and respond</li> </ul>
Pass/Fail	Pass
Observations (if any)	

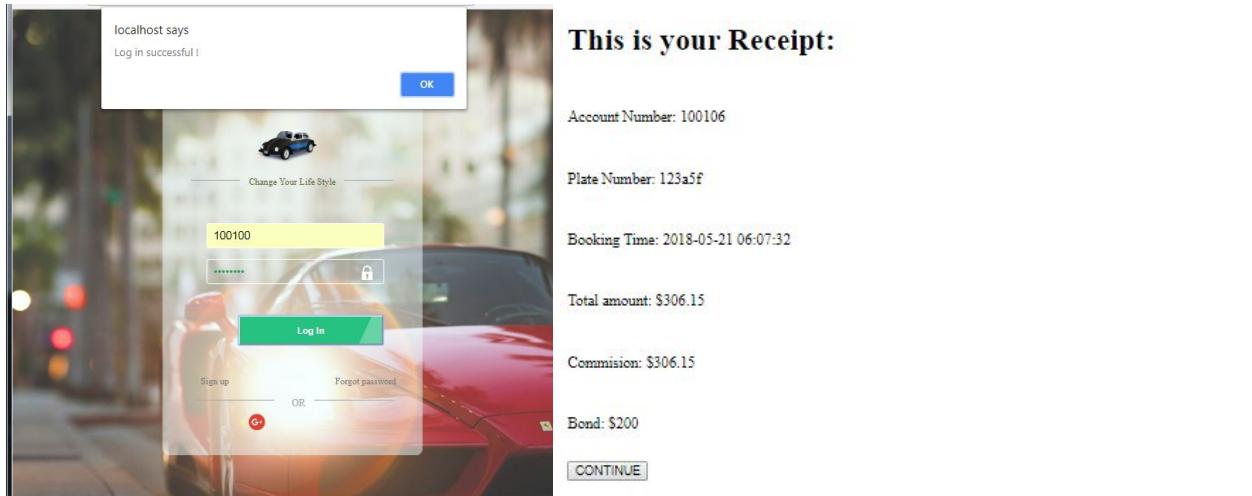
<b>8 View previous messages</b>	
Actions	a. Open Message system and click on a previous message to view
Expected Result	b. A message and past messages can be viewed
Pass/Fail	pass
Observations (if any)	

<b>13 Add Cars</b>	
Actions	a. Click upload car to add a new car
Expected Result	a. Opens upload page and you can add
Pass/Fail	Pass
Observations (if any)	

<b>Edit Cars</b>	
Actions	a. Select car to update and click update button
Expected Result	a. Opens up existing car to be modified
Pass/Fail	Pass
Observations (if any)	

16 Leave reviews	
Actions	a. Leave a review on users profile page
Expected Result	a. A review will be permanent in the database
Pass/Fail	Pass
Observations (if any)	

## Results



This is your Receipt:

Account Number: 100106

Plate Number: 123a5f

Booking Time: 2018-05-21 06:07:32

Total amount: \$306.15

Commission: \$306.15

Bond: \$200

CONTINUE

localhost says  
Log in successful !

OK

Change Your Life Style

100100

.....

Log In

Sign up

Forgot password

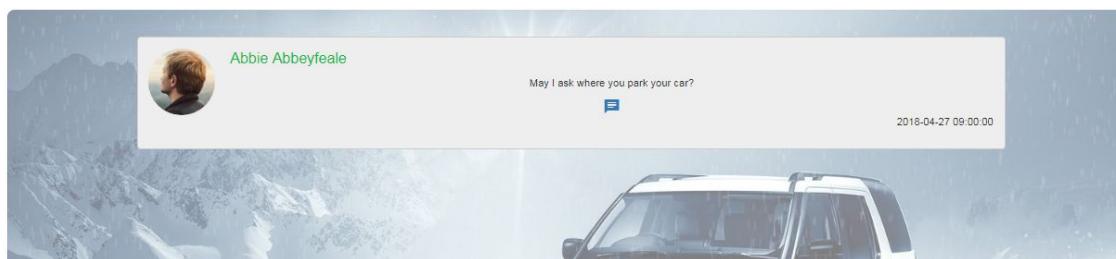
OR

G

UDRIVE

Home My Account Messages Help Contact

.MESSAGES.





**"CAR OWNER"**

PERSONAL INFORMATION

Full Name: Abbi Abbeydale

D.O.B: 1934-01-10

E-Mail: Abbi\_Abbeydale@generic.com

Gender: FEMALE

Address: 28 Abbotsford Place

You haven't verified your account. To verify:

VERIFY

CREDIT CARD INFORMATION

BANK ACCOUNT: 9754 1006

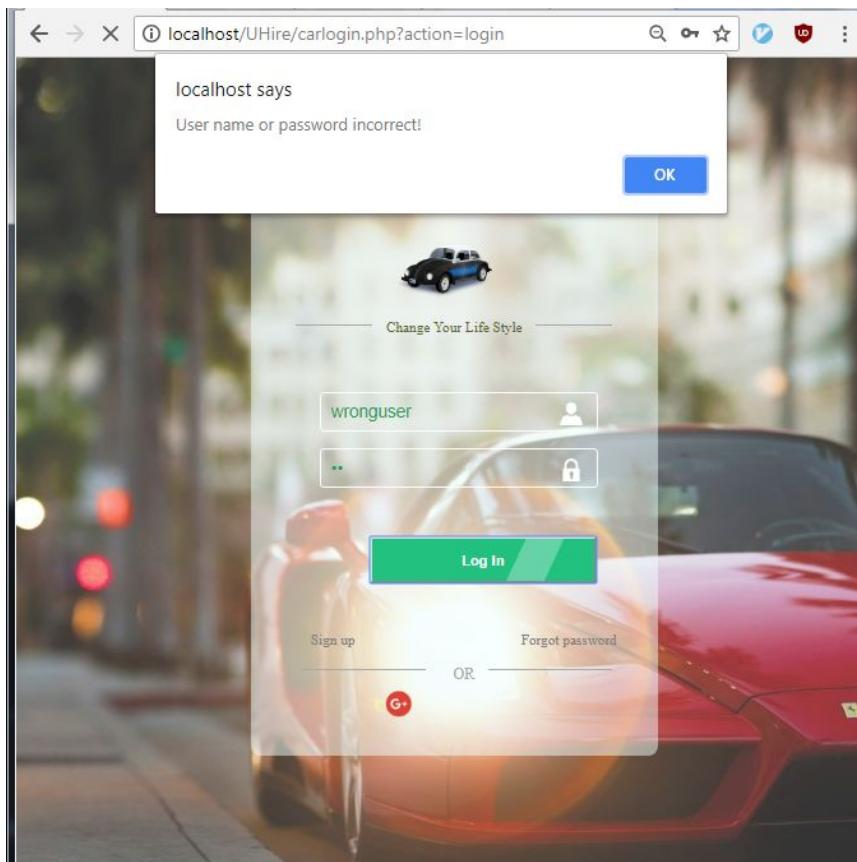
BANK CARD: VISA MASTERCARD AMEX

CARD NO.: 147483653

TYPE: credit card

BALANCE: \$4493.00

EXPIRE DATE: 2020-12-31



```

2825 INSERT INTO CAR
2826 VALUES
2827 ('10003f', '99.7', '1', 'A.J. Hare Avenue', '100100', '2019-1-3', '2002-01-01', 'Ecopower', 'Ford', 'auto', '7', '10002', 'Electric', 'SUV'),
2828 ('10003f', '76.9', '0', 'Abarsco Boulevard', '100102', '2018-10-25', '1998-01-01', 'Yaris', 'The radio in this car is working and comes with a drink holder', 'Toyota', 'auto', '4', '5156', 'Gasoline', 'Sedan'),
2829 ('1007de', '17.6', '1', 'Abberton Lane', '100104', '2019-0-25', '2002-01-01', 'Civic', 'This car comes with a turbo', 'Honda', 'manual', '2', '10556', 'Diesel', 'Hatchback'),
2830 ('100bc0', '76.1', '1', 'Abbott Crescent', '100106', '2020-9-14', '2013-01-01', 'Astra', 'Has a new engine!!!', 'Volkswagen', 'manual', '4', '20896', 'Autogas', 'Sportscar'),
2831 ('1013ab', '28.8', '1', 'Abbottford Drive', '100108', '2019-8-16', '2014-01-01', 'Accord', 'Has a large boot', 'Nissan', 'auto', '2', '8525', 'Biogiesel', 'Coupe'),
2832 ('1013ab', '75.9', '1', 'Abbotsbury Highway', '100110', '2020-10-27', '2006-01-01', 'Candy', 'The radio in this car is working and comes with a drink holder', 'Mazda', 'manual', '5', '22324', 'Ethanol', 'Crossover'),
2833 ('10179a', '71.6', '0', 'Abbotsford Place Street', '100112', '2019-9-13', '2006-01-01', 'Corolla', 'The radio in this car is working and comes with a drink holder', 'Mercedes', 'auto', '5', '7339', 'Unleaded 95', 'Conver'),
2834 ('10179a', '71.6', '0', 'Abbotsford Place Street', '100114', '2019-10-10', '2014-01-01', 'Golf', 'Has a large boot', 'Volkswagen', 'manual', '5', '10002', 'Unleaded', 'Pickup'),
2835 ('1017f8', '89.7', '1', 'Abbots May', '100116', '2020-4-17', '2002-01-01', 'Tucson', 'Has a large boot', 'Renault', 'manual', '3', '12848', 'Diesel', 'Microcar'),
2836 ('10236f', '35.1', '0', 'Abbots Road', '100118', '2020-5-27', '2002-01-01', 'Tucson', 'Has a new engine!!!', 'Lamborghini', 'manual', '3', '5409', 'Autogas', 'Station Wagon'),
2837 ('10275b', '53.8', '1', 'Acromobile Crescent', '100120', '2018-5-21', '2013-01-01', 'Elantra', 'Has a new engine!!!', 'Tesla', 'auto', '2', '28157', 'Super Unleaded', 'Roadster'),
2838 ('102d45', '96.8', '0', 'Abel Lane', '100122', '2020-11-13', '1999-01-01', 'HiLux', 'This is a good car, with 4 wheels', 'Ferrari', 'manual', '2', '38433', 'Autogas', 'Minivan'),
2839 ('102f34', '54.8', '0', 'Abercombe Drive', '100124', '2018-10-8', '2006-01-01', 'Focus', 'Has a new engine!!!', 'Porsche', 'auto', '4', '28843', 'Biogiesel', 'CUV'),
2840 ('10305c', '99.1', '1', 'Abercraigie Boulevard', '100126', '2019-10-17', '2006-01-01', 'Rogue', 'Has a new engine!!!', 'Bentley', 'auto', '3', '10450', 'Gasoline', 'AWD'),
2841 ('103712', '21.5', '0', 'Abercrobie Way', '100127', '2020-6-25', '1991-01-01', 'Qashqai', 'The radio in this car is working and comes with a drink holder', 'Bugatti', 'manual', '3', '10450', 'Gasoline', 'AMO'),
2842 ('103b01', '21.8', '0', 'Aberdeen Way', '100130', '2018-7-15', '2011-01-01', 'CX-5', 'This car comes with a turbo', 'BMW', 'auto', '5', '13463', 'Super Unleaded', 'Coupe'),
2843 ('103ef0', '55.4', '0', 'Aberfeldy Crescent', '100132', '2020-7-13', '2011-01-01', 'Accord', 'Has a new engine!!!', 'Jeep', 'auto', '2', '27894', 'Autogas', 'Station Wagon'),
2844 ('10420f', '12.2', '0', 'Abergeldie Highway', '100134', '2018-6-17', '2009-01-01', 'Sportage', 'Has a new engine!!!', 'Maserati', 'manual', '7', '26752', 'Autogas', 'AMO'),
2845 ('1046cc', '96.8', '0', 'Abermen Way', '100136', '2019-7-19', '2001-01-01', '5-Class', 'This car comes with a turbo', 'Mitsubishi', 'manual', '7', '1047', 'Biogiesel', 'Crossover'),
2846 ('1046cc', '96.8', '0', 'Abermen Way', '100137', '2019-7-19', '2001-01-01', '5-Class', 'This car comes with a turbo', 'Mitsubishi', 'manual', '7', '17392', 'Biogiesel', 'Crossover'),
2847 ('1048e0', '63.1', '1', 'Abergill Road', '100140', '2019-9-18', '2002-01-01', 'Elantra', 'This car is working and comes with a drink holder', 'Honda', 'manual', '5', '17184', 'Electric', 'Minivan'),
2848 ('10529n', '97.8', '0', 'Abergill Road', '100141', '2019-9-18', '2002-01-01', 'Elantra', 'This car is working and comes with a drink holder', 'Honda', 'manual', '5', '13947', 'Autogas', 'Roadster'),
2849 ('10558b', '61.8', '1', 'Abell Street', '100144', '2018-9-20', '2017-01-01', 'Aventador', 'This car comes with a turbo', 'BMW', 'auto', '6', '2658', 'Ethanol', 'Minivan'),
2850 ('10559j', '98.1', '1', 'Aboukir Boulevard', '100146', '2020-2-25', '2007-01-01', 'GLC', 'Has a new engine!!!', 'Mercedes', 'auto', '3', '16598', 'Biogiesel', 'AMO'),
2851 ('10560b', '88.7', '1', 'Aburac Plaza', '100148', '2019-6-17', '2011-01-01', 'Aventador', 'Has a new engine!!!', 'Nissan', 'manual', '4', '14093', 'Autogas', 'Station Wagon'),
2852 ('106257', '58.8', '0', 'Abuklea Highway', '100150', '2019-6-17', '2006-01-01', 'CX-5', 'This car comes with a turbo', 'Mazda', 'auto', '5', '12658', 'Autogas', 'Hatchback'),
2853 ('106257', '58.8', '0', 'Abuklea Highway', '100151', '2019-6-17', '2006-01-01', 'Candy', 'The radio in this car is working and comes with a drink holder', 'Mitsubishi', 'auto', '6', '4655', 'Autogas', 'Station Wagon'),
2854 ('106353', '19.1', '1', 'Access Road', '100154', '2019-8-26', '1995-01-01', 'Slyphy', 'Has a new engine!!!', 'Honda', 'auto', '3', '23489', 'Biogiesel', 'Hatchback'),
2855 ('106624', '97.8', '0', 'Achillies Lane', '100156', '2019-10-23', '1991-01-01', '5-Class', 'This is a good car, with 4 wheels', 'Honda', 'auto', '4', '28998', 'Diesel', 'Crossover'),
2856 ('107213', '98.3', '1', 'Ackling Highway', '100158', '2019-12-26', '2016-01-01', 'A4', 'This is a good car, with 4 wheels', 'Nissan', 'manual', '5', '2422', 'Ethanol', 'Commercial'),
2857 ('107602', '87.3', '0', 'Acorn Lane', '100160', '2018-5-11', '2011-01-01', 'Slyphy', 'This car comes with a turbo', 'Porsche', 'manual', '7', '25271', 'Diesel', 'AMO'),
2858 ('109191', '73.4', '0', 'Acorn Lane', '100162', '2019-1-17', '2006-01-01', 'Focus', 'This is a good car, with 4 wheels', 'Honda', 'manual', '5', '23393', 'Autogas', 'Pickup'),
2859 ('109191', '73.4', '0', 'Acorn Lane', '100163', '2019-1-17', '2006-01-01', 'Elantra', 'This car comes with a turbo', 'Mercedes', 'manual', '3', '16598', 'Biogiesel', 'Commercial'),
2860 ('109191', '73.4', '0', 'Acorn Lane', '100165', '2019-4-6', '1992-01-01', 'Civic', 'The radio in this car is working and comes with a drink holder', 'Lamborghini', 'auto', '6', '20536', 'Ethanol', 'Coupe'),
2861 ('10958c', '58.3', '1', 'Acorn Lane', '100168', '2019-9-26', '1994-01-01', 'Camry', 'Has a large boot', 'Ford', 'auto', '5', '2610', 'Autogas', 'Hatchback'),
2862 ('10989d', '43.8', '0', 'Acropolis Way', '100170', '2020-7-51', '1998-01-01', 'Focus', 'The radio in this car is working and comes with a drink holder', 'Mitsubishi', 'auto', '6', '17034', 'Biogiesel', 'SUV'),
2863 ('10989c', '44.7', '0', 'Actinotus Crescent', '100172', '2019-1-28', '2014-01-01', 'GLC', 'This car comes with a turbo', 'Volvo', 'manual', '7', '22422', 'Ethanol', 'Hatchback'),
2864 ('10918b', '53.3', '0', 'Action Avenue', '100174', '2019-3-22', '1998-01-01', 'CR-V', 'Has a new engine!!!', 'Nissan', 'auto', '3', '28489', 'Electric', 'Microcar'),
2865 ('10918b', '53.3', '0', 'Action Avenue', '100176', '2019-3-22', '1998-01-01', 'CR-V', 'Has a new engine!!!', 'Honda', 'manual', '5', '13947', 'Autogas', 'Microcar'),
2866 ('10996b', '55.2', '0', 'Adda Road', '100177', '2018-9-18', '2002-01-01', 'A4', 'This car comes with a turbo', 'Mitsubishi', 'auto', '5', '26147', 'Ethanol', 'Commercial'),
2867 ('10955b', '16.9', '0', 'Adda Highway', '100178', '2019-1-5', '1997-01-01', 'HiLux', 'This car comes with a turbo', 'Mitsubishi', 'auto', '5', '6051', 'Unleaded', 'Crossover'),
2868 ('109147', '15.2', '0', 'Adelma Lane', '100182', '2019-7-20', '1996-01-01', 'Slyphy', 'This car comes with a turbo', 'Mazda', 'manual', '2', '28634', 'Biogiesel', 'AMO'),
2869 ('10953b', '61.2', '0', 'Adam Highway', '100184', '2019-8-27', '2016-01-01', 'Accord', 'This is a good car, with 4 wheels', 'Nissan', 'auto', '4', '2716', 'Biogiesel', 'Microcar'),
2870 ('10952b', '85.4', '1', 'Adamnaby Street', '100186', '2020-11-10', '2006-01-01', 'GLC', 'This is a good car, with 4 wheels', 'Mazda', 'auto', '7', '4432', 'Gasoline', 'Hatchback'),
2871 ('10952b', '85.4', '1', 'Adamnaby Street', '100187', '2020-11-10', '2006-01-01', 'GLC', 'This is a good car, with 4 wheels', 'Mazda', 'auto', '7', '11711', 'Gasoline', 'Minivan'),
2872 ('109183', '51.9', '0', 'Adamson Highway', '100189', '2019-12-15', '1996-01-01', 'Accord', 'This car comes with a drink holder', 'Lamborghini', 'manual', '7', '22807', 'Gasoline'),
2873 ('109472', '28.5', '0', 'Aderstone Place', '100192', '2019-10-27', '2003-01-01', 'Accord', 'This car comes with a turbo', 'Nissan', 'manual', '3', '13944', 'Autogas', 'SUV'),
2874 ('1098e1', '6.6', '1', 'Adderton Avenue', '100194', '2018-4-20', '2016-01-01', 'Slyphy', 'The radio in this car is working and comes with a drink holder', 'Mitsubishi', 'manual', '6', '16095', 'Autogas', 'Hatchback'),
2875 ('10bc0b', '21.1', '1', 'Addington Highway', '100196', '2019-2-24', '2017-01-01', 'Aventador', 'Has a new engine!!!', 'Ferrari', 'auto', '7', '6051', 'Unleaded', 'Crossover'),
2876 ('10cb0f', '4.2', '1', 'Addiscombe Crescent', '100198', '2028-8-18', '2005-01-01', '5-Class', 'Has a large boot', 'Toyota', 'auto', '4', '886', 'Electric', 'Sedan'),
2877 ('1094eb', '88.1', '0', 'Addiscombe Crescent', '100199', '2019-1-24', '2019-01-01', 'Central', 'This car comes with a turbo', 'BMW', 'auto', '5', '19593', 'Diesel', 'Sedan'),
2878 ('1094eb', '59.1', '0', 'Addiscombe Lane', '100200', '2019-9-5', '1995-01-01', 'Central', 'This car comes with a turbo', 'BMW', 'auto', '3', '21226', 'Biogiesel', 'Station Wagon'),
2879 ('10cc8c', '59.3', '0', 'Addy Road', '100204', '2018-11-26', '2012-01-01', 'Accord', 'This car comes with a turbo', 'Ferrari', 'auto', '6', '1531', 'Electric', 'Convertible'),
2880 ('10d97b', '76.0', '1', 'Addy Place', '100206', '2018-8-21', '1991-01-01', 'Cordoba', 'The radio in this car is working and comes with a drink holder', 'Nissan', 'auto', '6', '32454', 'Diesel', 'CUV'),
2881 ('10d46a', '3.1', '1', 'Adeline Avenue', '100208', '2019-11-28', '1998-01-01', 'Astra', 'This is a good car, with 4 wheels', 'Nissan', 'manual', '3', '19348', 'Ethanol', 'Commercial'),
2882 ('10dc48', '23.8', '1', 'Adeline Avenue', '100209', '2019-1-28', '2004-01-01', 'Accord', 'This is a good car, with 4 wheels', 'Ferrari', 'auto', '7', '1381', 'Diesel', 'SUV'),
2883 ('10dc48', '23.8', '1', 'Adeline Avenue', '100210', '2019-1-28', '2004-01-01', 'Accord', 'This is a good car, with 4 wheels', 'Ferrari', 'auto', '7', '1381', 'Diesel', 'SUV'),
2884 ('10dc48', '23.8', '1', 'Adeline Avenue', '100211', '2019-1-28', '2004-01-01', 'Accord', 'This is a good car, with 4 wheels', 'Nissan', 'manual', '3', '14059', 'Ethanol', 'Microcar'),
2885 ('10dc48', '57.1', '0', 'Adder Crescent', '100215', '2019-1-12', '1996-01-01', 'Focus', 'The radio in this car is working and comes with a drink holder', 'Mazda', 'manual', '5', '18566', 'Biogiesel', 'Microcar'),
2886 ('10e815', '82.1', '0', 'Adhera Avenue', '100218', '2018-4-6', '1994-01-01', 'Corolla', 'The radio in this car is working and comes with a drink holder', 'Volkswagen', 'auto', '6', '15748', 'Autogas', 'Microcar'),
2887 ('10ec04', '56.3', '0', 'Adina Road', '100220', '2018-8-27', '2000-01-01', 'CX-5', 'This car comes with a turbo', 'Volvo', 'auto', '5', '22184', 'Ethanol', 'AMO'),

```

Snippet of the test data used, too much test data exists to appropriately fit into this report.