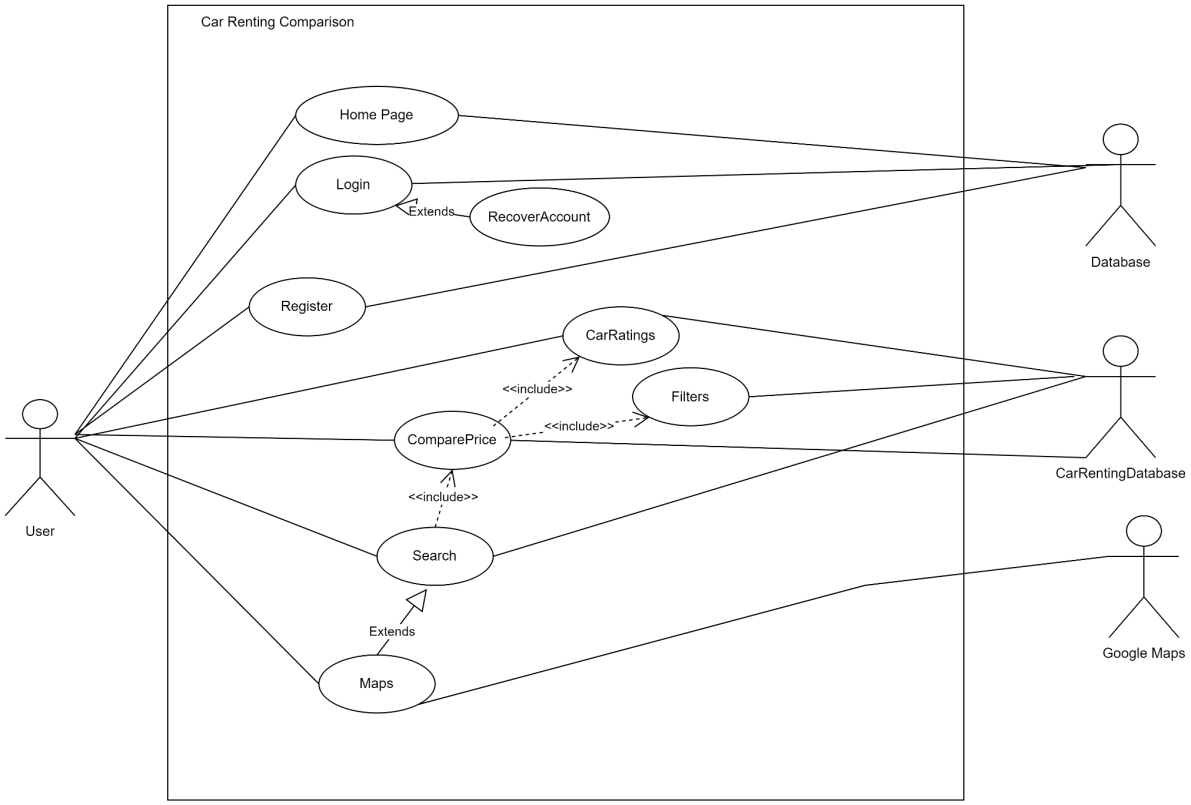
Lab 2 Deliverables

1. Refined Use Case Models

1.1. Refined Use Case Diagram



1.2. Refined Use Case Description

Use Case ID:	001		
Use Case Name:	Register		
Created By:	Lim Ke En	Last Updated By:	Lim Ke En
Date Created:	29 January 2023	Date Last Updated:	7 February 2023

Actor:	User (Initiating Actor), Database
Description:	First time users can register for an account to create their account.
Preconditions:	1. The user must be connected to the internet 2. The user has do not have an account prior to registration 3. The user has navigated to the Login Interface
Postconditions:	1. The user has successfully registered an account for the application with a unique username and password and their account is added into the system database. Or 2. The user is notified of the reason(s) why the registration of the account is unsuccessful.
Priority:	High
Frequency of Use:	20 times per day
Flow of Events:	1. At the homepage of the website, the user can click onto the “Login” button and the system will redirect the user to the login page. 2. Since the user is a new user, he would need to register for a new account. 3. The user can click on “Register as a new user” which is situated at the bottom of the login page and he will be redirected to the registration page. 4. The user would then need to input a valid email address, username, birth date, and a valid mobile phone. 5. The user will need to input a valid password that contains at least 8 characters which includes an upper-case, lower-case letter, numerical digits and a special character. 6. The user will need to input the password once more to confirm his password. 7. At the end of the registration form, the user would need to check the checkbox of “I agree to the Terms of Use and Privacy Policy”. 8. The user will click on the “Sign up” button to register their account. 9. The system validates if there is an identical email existing in the system

	10. The system will verify if the password satisfies all requirements 11. For further verification, the system will automatically generate a One-Time Password (OTP) in which the user will need to input the OTP (Verification function). 12. The system will verify if the user inputs the correct OTP. 13. Upon verification, the system will store all information in the database securely. 14. Once registration is successful, the system will automatically help the user to log into his account.
Alternative Flows:	<u>AF-S5: The user inputs a password that does not satisfy all the requirements set.</u> 1. The system displays the message "Password does not meet all requirements, please try again!" under the password field. 2. The system returns to Step 5 and waits for inputs from the user. <u>AF-S6: The user inputs a mismatched password.</u> 1. The system displays the message "Passwords do not match, please try again!" below the password field. 2. The system returns to Step 5 and waits for inputs from the user. <u>AF-S7: The user did not check the checkbox of "I agree to the Terms of Use and Privacy Policy."</u> 1. Upon clicking on the "Sign up" button, the system will display the message "Please check the checkbox for acknowledging the Terms of Use and Privacy Policy!" at the top of the registration form. 2. The system will return to Step 4 and wait for input from the user. <u>AF-S8: The user did not complete all of the input fields.</u> 1. Upon clicking on the "Sign up" button, the system will display the message "Please check that all fields are filled up!" at the top of the registration form 2. The system will return to Step 4 and wait for user to complete all inputs. <u>AF-S9: The user inputs an email address that had already been registered.</u> 1. The system displays the message "Email address has already been registered, please input another email address!" under the email address field. 2. The system returns to Step 4 and waits for input from the user.

	AF-S12: The user inputs an incorrect OTP 1. The system displays the message “Incorrect OTP! Please try again!” at the top of the registration form 2. The system will re-generate a new OTP and sent it to their email address. 3. The system returns to Step 4 and waits for the user to input again.
Exceptions:	EX-1: The user did not receive the OTP in his email. 1. Users can click on the “Resend OTP” button that is made available after 60 seconds. 2. Once the user clicks on the “Resend OTP” button, a new OTP will be generated by the system and sent to the email address. 3. The system will return to Step 4 and wait for the user to input again. EX-2: The user request for more than three OTP request. 1. On the fourth time the user requests to generate a new OTP, the system will generate the message “Please try again with a different email” at the top of the registration form. 2. The system returns to Step 4 and waits for input from the user.
Includes:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	002		
Use Case Name:	Login		
Created By:	Lim Ke En	Last Updated By:	Lim Ke En
Date Created:	29 January 2023	Date Last Updated:	7 February 2023

Actor:	User (Initiating Actors), Database
Description:	The user can login to his/her account with the correct credentials that are inputted when users register.
Preconditions:	1. The user must be connected to the Internet. 2. The user has a registered account.
Postconditions:	1. The User has successfully logged into his/her own account. OR 2. The user is notified of the reason(s) why he is unable to login into his account. Eg. You have input the wrong email address/password
Priority:	High
Frequency of Use:	More than 20 times a day
Flow of Events:	1. At the homepage of the website, the user can click onto the “Login” button and the system will redirect the user to the login page. 2. At the login page, the system requests the input of both the email address and password. 3. The user inputs his registered email address and his password. 4. The user clicks on the “Login” button 5. The system verifies the credentials (Email and Password) provided with the database *VerifyCredentials* function 6. If the Email and Password are correct and verified, the user will be directed into his account.
Alternative Flows:	<u>AF-S4: The user left the input field blank.</u> 1. Upon clicking onto the “Login” button, the system displays the following message “Please ensure that all fields are filled up!”. 2. The system will prompt the user to fill up all the fields. 3. The system returns to Step 3 and waits for the user to fill in all the fields. <u>AF-S5: The user inputs an incorrect email address or password.</u> 1. Upon clicking onto the “Login” button, the system displays the following message: “Invalid email address and/or password!” at the top of the login page using the *LoginError* function.

	2. The system returns to Step 3 and waits for the user to fill in the fields again.
Exceptions:	EX-1: The user inputs incorrect email address and password for more than five times 1. After five attempts, the system will display the message "More than five tries, please try again after 10 minutes or click on forget your password" 2. The system will only accepts registration hence returning to Step 3 after 10 minutes EX-2: The user forgets his username and/or password 1. The user clicks on "Forget Password" that is situated below the login button. 2. The user can recover his/her account using the extended use case *RecoverAccount*
Includes:	
Extends	*RecoverAccount*
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	003		
Use Case Name:	RecoverAccount		
Created By:	Lim Ke En	Last Updated By:	Lim Ke En
Date Created:	29 January 2023	Date Last Updated:	15 February 2023

Actor:	User (Initiating Actors), Database
Description:	The user can request for help if he forgets his password and this use case can help to recover the account.
Preconditions:	1. The user must be connected to the internet 2. The user has forgotten his/her login credentials (Password of his account)
Postconditions:	1. The user has successfully recovered his/her account by changing his password. OR 2. The user has sought further assistance and support from the customer service.
Priority:	Medium
Frequency of Use:	Approximately 5 times a day
Flow of Events:	1. The user clicks on "Forget your password" on the login page 2. The system displays the recover account page 3. User will input the registered email address and clicks on "Recover Account"

	4. The system will verify the email address to ensure that the email address exist and sends an email to the email address that is given 5. The system then automatically generates a One-time Password (OTP) and emails it into the user's inbox. 6. User will input the OTP. 7. Upon successful inputs of the OTP, the system will redirect users to a page where users can type in their new password with certain requirements. 8. The system then validates and verifies the new password and updates into the database. 9. Once the password is changed successfully in the database, users will be informed of the change in password and be redirected back to the login page.
Alternative Flows:	AF-S4: The user inputs the incorrect email address and email address is not registered 1. The system will display the following message: "Not a registered email, Please try again!" at the top of the page. 2. The system will returns to Step 3 and waits for user to input the registered email address AF-S7: The user inputs an incorrect OTP 1. The system displays the message "Incorrect OTP! Please try again!" at the top of the registration form 2. The system will re-generate a new OTP and send it to their email address. 3. The system returns to Step 4 and waits for the user to input again. AF-S8: The user inputs a password that does not satisfy all the requirements set. 1. The system displays the message "Password does not meet all requirements, please try again!" under the password field. 2. The system returns to Step 7 and waits for inputs from the user. AF-S8: The user inputs a mismatched password. 1. The system displays the message "Passwords do not match, please try again!" below the password field. 2. The system returns to Step 7 and waits for inputs from the user.
Exceptions:	EX-1: The user forgets his registered email 1. The user clicks on "Contact Support" button 2. The system will show the support email address

	3. The user will then contact the support via email to recover his/her email address EX-2: The user did not receive the OTP in his email. 1. Users can click on the “Resend OTP” button that is made available after 60 seconds. 2. Once the user clicks on the “Resend OTP” button, a new OTP will be generated by the system and sent to the email address. 3. The system will return to Step 4 and wait for the user to input again. EX-3: The user request for more than three OTP request. 1. On the fourth time the user requests to generate a new OTP, the system will generate the message “Please try again with a different email” at the top of the registration form. 2. The system returns to Step 4 and waits for input from the user. EX-4: The user did not receive any email from the website despite multiple tries. 1. Users can click onto the “Help” button and seek help from the customer service. 2. User then waits for the customer service to attend to his problem. 3. The system returns to Step 4 and waits for further input from the user.
Includes:	
Extends	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	004		
Use Case Name:	Search		
Created By:	Lim Ke En	Last Updated By:	Lim Ke En
Date Created:	30 January 2023	Date Last Updated:	15 February 2023

Actor:	User (Initiating Actor), CarRentingData (System)
Description:	Users will be able to search for the location they want to rent the car, pick up time, drop off time as well as filter the type of car that the users prefer
Preconditions:	1. Users must have logged in

	2. Users must be connected to an internet
Postconditions:	Users will obtain a list of car renting services that is sorted from the lowest price to the highest price.
Priority:	High
Frequency of Use:	20 times a day
Flow of Events:	1. Users can navigate through the navigation bar and click onto the "Start Searching" button that is situated on the navigation bar to start their searching process 2. Users can first type in their postal code of the current location 3. After that, users can enter the date of renting the car, start time of renting, drop off time, type of car (e.g. 5 seater car, electric car) depending on their preference 4. The system will retrieve the information of the preferred choice. 5. The system will display the results of the car renting services sorted from the lowest price to the highest price using the ComparePrice use case.
Alternative Flows:	<u>AF-S2: Users key in invalid postal code of their location</u> 1. The system will generate the following message: "Please enter a valid postal code" at the bottom of the text box for the location 2. The system will return back to step 2 and wait for further input from the user. <u>AF-S3: Users did not field in all the blanks of input</u> 1. The system will generate the following message: "Please input all the fields" at the bottom of the text box 2. The system will return back to step 2 and waits for further input from the user <u>AF-S3: Users do not know the amount of time they need in renting the car</u> 1. Users can click on the "Need estimate time?" button below the drop off time section. 2. System will then redirect the users to the Maps page where users can input the starting location and their destination of travel using the extend *Maps* use case
Exceptions:	<u>EX-1: The website loses internet connection while searching</u> 1. A page which shows the following message "Internet connection is lost, please try again later." which informs the user of the problem. 2. The system will return to step 1 for users to key in the inputs.
Includes:	ComparePrice
Extends	Maps

Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	005		
Use Case Name:	Maps		
Created By:	Lim Ke En	Last Updated By:	Lim Ke En
Date Created:	30 January 2023	Date Last Updated:	15 February 2023

Actor:	User (Initiating Actor), Google Maps
Description:	Users will use the google maps API to input the locations they are travelling to and the system will calculate the total estimated travelling time users require.
Preconditions:	1. Users are logged into the account 2. Users must be connected to an Internet 3. Users must have searched for the nearest renting location and set it as the starting point. 4. Google Maps API needs to be set up
Postconditions:	Users will obtain the total estimated travelling time and are able to estimate the total amount of time needed to rent the car.
Priority:	Medium
Frequency of Use:	10 times a day
Flow of Events:	1. After searching or knowing the start location, users will input the number of locations they will be travelling to. 2. Users will then input all the addresses of the location. 3. The system will then compute the total estimated travelling time using the CalculateTime function, this will allow users to estimate the amount of time they require to rent the car. 4. The system will then display the total amount of estimated travelling time.
Alternative Flows:	<u>AF-S2: User inputs an invalid address</u> 1. The system will display the following message “Please enter a valid address” under the invalid address 2. The system will return to step 2 and wait for further inputs from the user.
Exceptions:	NIL
Includes:	
Extends	

Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	006		
Use Case Name:	ComparePrice		
Created By:	Lim Ke En	Last Updated By:	Lim Ke En
Date Created:	30 January 2023	Date Last Updated:	7 February 2023

Actor:	User (Initiating Actor), CarRentingData
Description:	Users will be provided with the results that is sorted from the lowest to the highest price
Preconditions:	1. Users is connected to the internet 2. Users have login to their registered account 3. Users have used the search use case to search and filter about their preference 4. System shall be able to calculate the renting prices from all three companies
Postconditions:	Users will be provided with the list of cheapest rental service and users can make informed decision
Priority:	High
Frequency of Use:	20 times a day
Flow of Events:	1. Users will be directed to a page where the system will display the information according to user's preference that is sorted from lowest to highest price 2. The system will display information such as location of the car renting place, total rental amount, colour of car and type of car 3. The system will also show the rating of the car where users can refer to those ratings that is shown in the CarRating use case 4. At the side panel, users can filter more information such as the inclusivity of fuel using the filter use case
Alternative Flows:	<u>AF-S4: User can choose not to select any filter.</u> 1. The system will show all results sorted from the lowest to the highest price without applying any filter.
Exceptions:	
Includes:	*Filters, CarRatings*
Extends	
Special Requirements:	
Assumptions:	

Notes and Issues:	

Use Case ID:	007		
Use Case Name:	CarRatings		
Created By:	Lim Ke En	Last Updated By:	Lim Ke En
Date Created:	30 January 2023	Date Last Updated:	7 February 2023

Actor:	Users (Initiating Actor), Database
Description:	Users will be able to read reviews from other users about the different car renting companies and users can add reviews if they want
Preconditions:	1. Users is connected to the internet 2. Users have login to the registered account
Postconditions:	1. Users have read about the reviews from other users which can help them in making more informed choices OR 2. Users have added their reviews and ratings for the different car renting companies.
Priority:	Medium
Frequency of Use:	10 times a day
Flow of Events:	Reading Reviews 1. System will display the top 5 reviews 2. Users can read about the reviews and see the rating of the car renting company 3. If users wants to read more about the company, the link to the car renting company will be included in the reviews page Adding Reviews 1. Below the reviews section, the system will display a comment bar where users can add in their review of the car renting company 2. Users can add in their reviews and add their rating towards the company 3. After users are ready to add their review and rating, users can press the “Submit” button 4. The system will store the reviews and ratings in the database
Alternative Flows:	<u>AF-S2 (Adding Reviews): Users left the fields blank and press the “Submit” button</u> 1. The system will generate the following message: “Please enter text into the fields” at the bottom of the page

	2. The system will return to step 2 and wait for further inputs from the users.
Exceptions:	EX-1: No reviews about the car renting company 1. System will show the following message: “No reviews added” at the reviews section
Includes:	
Extends	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	008		
Use Case Name:	Filter		
Created By:	Lim Ke En	Last Updated By:	Lim Ke En
Date Created:	31 January 2023	Date Last Updated:	7 February 2023

Actor:	Users (Initiating Actor), Database
Description:	Include filters that filter choices such as inclusivity of fuel as well as if you can travel to malaysia.
Preconditions:	1. Users is connected to the internet 2. Users have login to the registered account 3. Users have used the search use case to search 4. Users have view the list of prices in the comparePrices use case
Postconditions:	Users will be able to filter their choices according to their preference for a more filtered list of suggestion
Priority:	Medium
Frequency of Use:	10 times a day
Flow of Events:	1. At the side panel, users can tick onto the check box that are applicable to their choices 2. Users are able to filter by the following selection: Fuel type and car condition 3. The system will regenerate the choices as follows 4. The system will generate a new list of choices according to the filtered choice 5. Users can now view the more detailed list according to their own preference
Alternative Flows:	
Exceptions:	
Includes:	

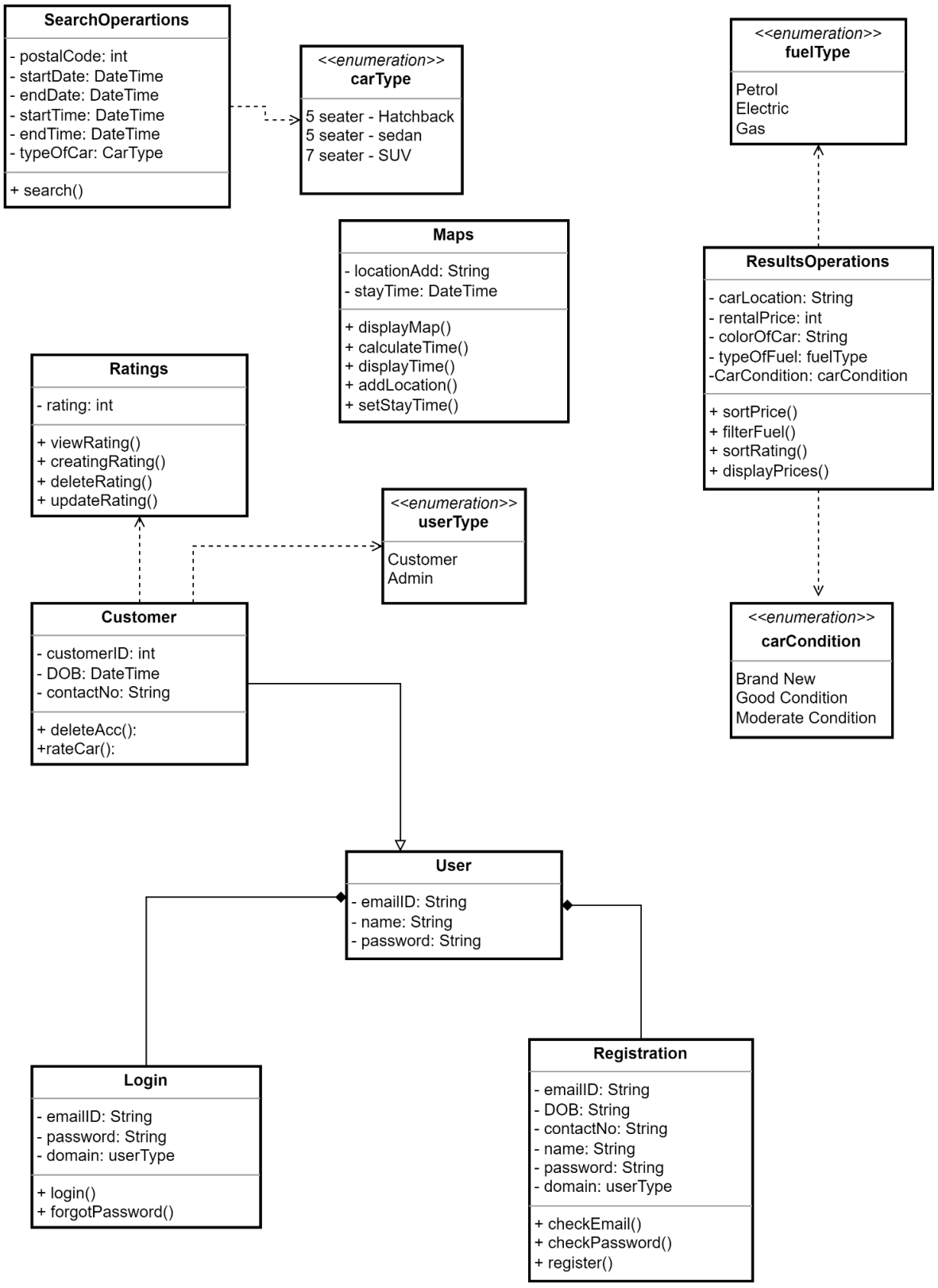
Extends	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	009		
Use Case Name:	ViewInformation		
Created By:	Lim Ke En	Last Updated By:	Lim Ke En
Date Created:	12 February 2023	Date Last Updated:	16 February 2023

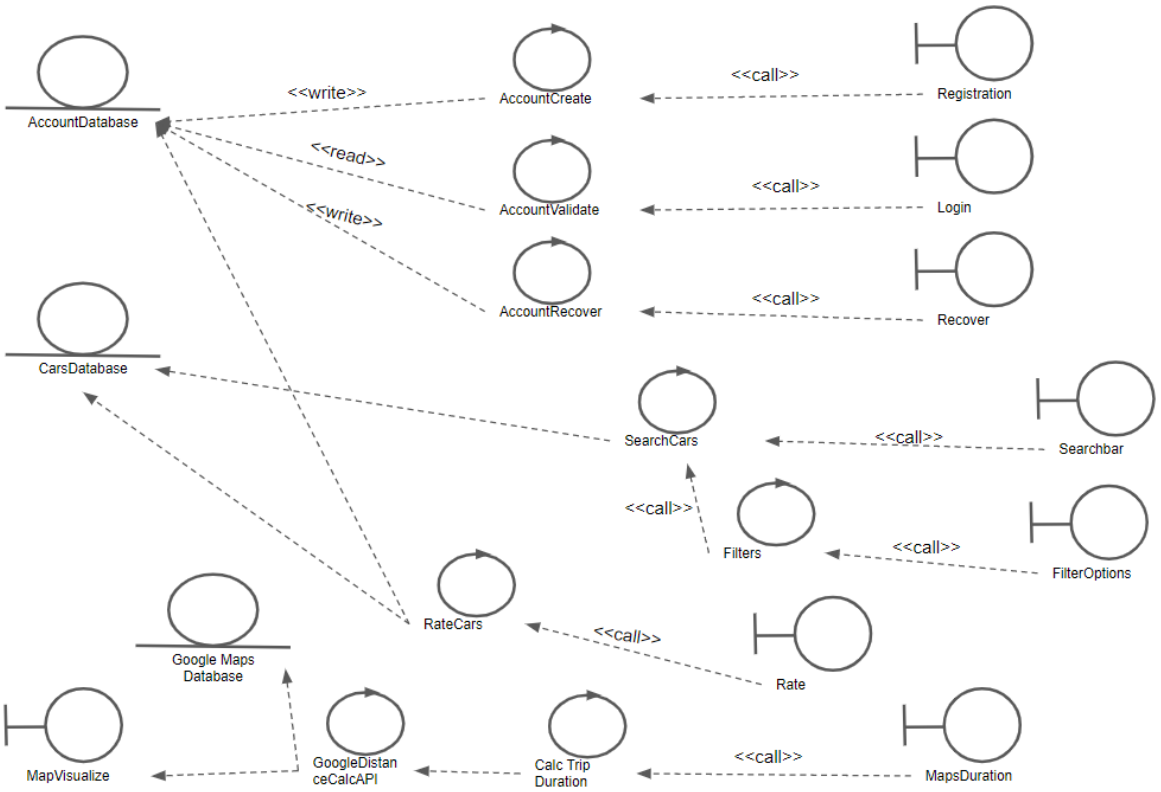
Actor:	Users (Initiating Actor), Database
Description:	Users can view more information about the specific car rental upon clicking on the view button
Preconditions:	1. A detailed list of car rental have been generated for comparison that is arranged from the lowest price to the highest price
Postconditions:	1. Users will be able to view more information about the car such as the terms and conditions 2. Users can be redirected to the car renting page if they choose to book the specific car rental service
Priority:	Medium
Frequency of Use:	10 to 15 times
Flow of Events:	1. Upon wanting to know more information, users can click on the view button. 2. After clicking the view button, the detailed information about the car will be shown. 3. Full detailed information about the car would be listed in that page. 4. Information such as the terms and condition of the car renting service and some (Frequently Ask Question) FAQ will be listed for the user 5. If users wants to book the certain car renting service, they can be redirected to the specific website or application
Alternative Flows:	
Exceptions:	
Includes:	
Extends	
Special Requirements:	
Assumptions:	

Notes and Issues:	

2. Class Diagram of entity classes

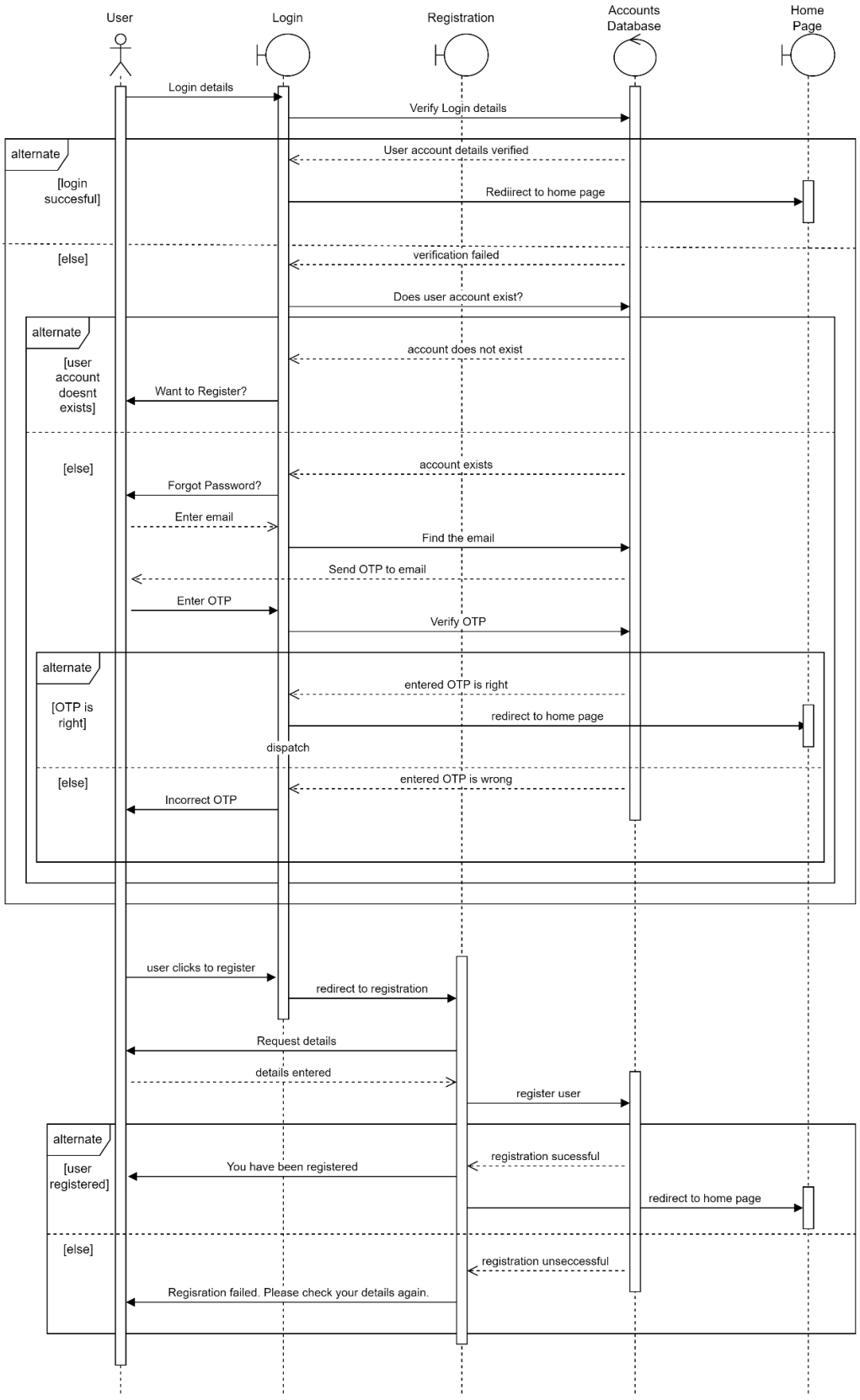


3. Key boundary classes and control classes

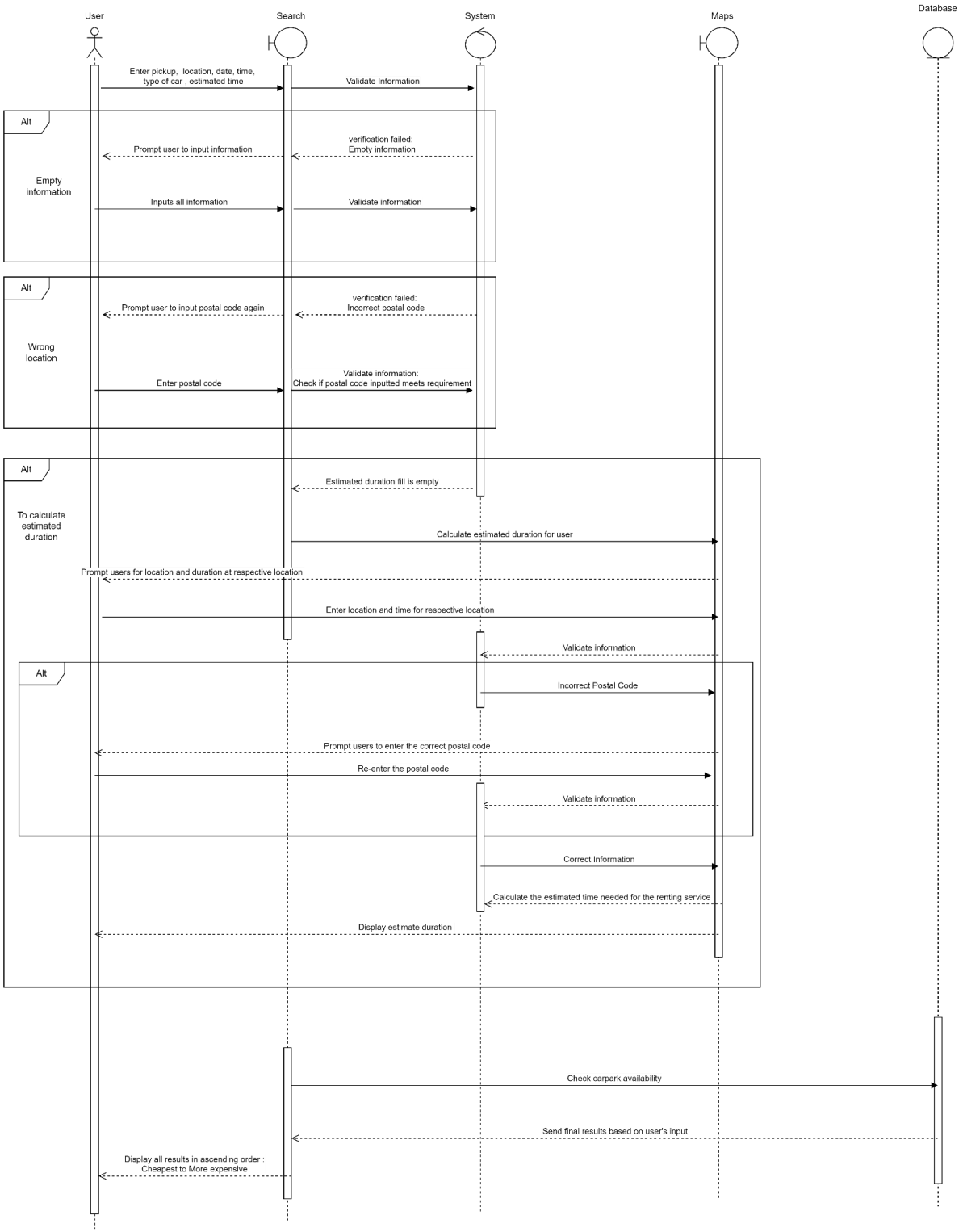


4. Sequence diagrams of some use cases

A. Login and Register Sequence Diagram



B. Search Sequence Diagram



5. Initial Dialogue map (State Machine Diagram)