SC2006 Software Engineering

NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Lab#1 Deliverables

Lab Group: SDAA

Group Name: Team 3

Group Members

Name	Matriculation No.
Ong Sheng Da	U2220668C
Shan Yi	U2222846C
Pala Tejaswi	U2221881D
Ng Zi Xuan	U2220246D
Ong Xin Ning, Trini	U2222202K

Purpose

Our mission is to simplify the urban commuting experience by helping drivers quickly find available parking spaces near their desired destination. We aim to alleviate the frustration and time wasted in searching for parking lots. This will make urban mobility more efficient, convenient, and pleasant for all drivers in Singapore.

Intended Audience

Given that our web application uses HDB API, we are only able to access HDB parking lots. Additionally, since there are vehicle size and weight restrictions, only certain vehicles are allowed into HDB car parks. Therefore, this web application is targeted at vehicle drivers, specifically those who drive Class 2 and Class 3 vehicles, e.g. motorcycles, cars, taxis, and vans.

Functional Requirements

1 GPS

1.1. GPS system will get the current physical position of the user device.

2 Car park information

2.1 The information shall display the following:

Address of car park.
Distance from current position obtained from the GPS and the car park.
The number of available lots.

2.2 At most 5 car park information within 2km radius from user's current position will be displayed to the user, sorted based on the nearest distance from the current position, with the nearest distance at the top, if the destination location is not indicated by the user.

2.3 At most 5 car park information within 2km radius from the destination location will be displayed to the user, sorted based on the distance from the destination, with the nearest distance at the top, if the destination is indicated by the user.

2.4 Distance will be calculated using Google Maps API.

2.5 If no car parks are found withing 2km radius, the system will throw in error message ""No available car park that meets your filter condition."

3 Filter

3.1 The filter will be an optional function for the users to set when searching for car parks.

3.2 The filter will accept user inputs from the following:

3.2.1 Tick boxes of types of car parks with multiple selections allowed:

Multi-Storey Car Parks
Basement Car Parks
Surface Car Parks

3.2.2 Tick boxes of types of parking systems with multiple selections allowed:

Coupon Parking
Electronic Parking System

3.2.3 Tick box of Night Parking Availability:

Night Parking

3.3 If the user did not use the filter function, the following will be the default filter:

All the tick boxes of types of car parks will be checked.
All the tick boxes of types of parking systems will be checked.
The tick box of "Night Parking" will NOT be checked.

4 Favourite

4.1 User will be able to indicate their favourite car parks.

4.2 Users shall only be able to remove their favourite car parks from their list of favourite car parks.

4.3 User will be able to navigate to their favourite car parks.

5 Map View

5.1 Users must be able to navigate to their chosen car park.

- 5.1.1 Map must show the shortest route to the car park using the Google Map API.
- 5.1.2 Map view will be displayed on an entirely new page.
- 5.1.3 Map view will be in full screen.

6 User accounts

6.1 Users must create unique individual accounts.

- 6.1.1 No more than 1 account shall share the same username.
  - 6.1.1.1 A username is initialised by the user when creating an account.
  - 6.1.1.2 The username cannot be changed once the account is initialised.

6.2 A password must be set when an account is created.

- 6.2.1 The user will be able to change his/her password.
  - 6.2.1.1 The user will be able to change his/her password before login.
    - 6.2.1.1.1 The User must enter a valid username before changing the password.
  - 6.2.1.2 The user will be able to change his/her password after successful login.
- 6.2.2 The user must enter a new password to change the password.
- 6.2.3 The user must re-enter the new password.
- 6.2.4 Once both entries of the new password match exactly, the new password will be changed successfully.

7 Report system

7.1 Users will be able to report any faults of the car park

- 7.1.1 Users must include their car park address in the report.
- 7.1.2 Users must provide a text description in the form of a string.

7.2 Admin will be able to view the reports submitted by users on the admin main page.

- 7.2.1 Admin will be able to view each case in detail with a click.
- 7.2.2 Admin will resolve each case manually by contacting the town council offline.
- 7.2.3 Admin shall click on the “resolved” button once they contact the town council offline.
- 7.2.4 Admin shall be able to click on a back button if they have not resolved the case.

8 Admin accounts

8.1 An admin account will be created by adding the username and password straight into the account database.

Non-Functional Requirements

1 Performance

1.1 The web application shall take within 3 seconds to load.
1.2 The login page shall take within 3 seconds to load.
1.3 The page after filtering shall take within 5 seconds to load.
1.4 The information on available car park slots shall take within 5 seconds to load.

2 Security

2.1 The password must have a minimum of 8 characters, with at least 1 uppercase letter, 1 lowercase letter, 1 digit and 1 special character.
2.2 When the user tries to log in or change passwords, the system will verify the username and password to prevent unauthorized access.
2.3 The information on users' accounts will be protected strictly.

3 Usability

3.1 The web application must be easy to use and navigate.
3.2 The favourite function will build a shortcut for the users to find the car park.

4 Maintainability

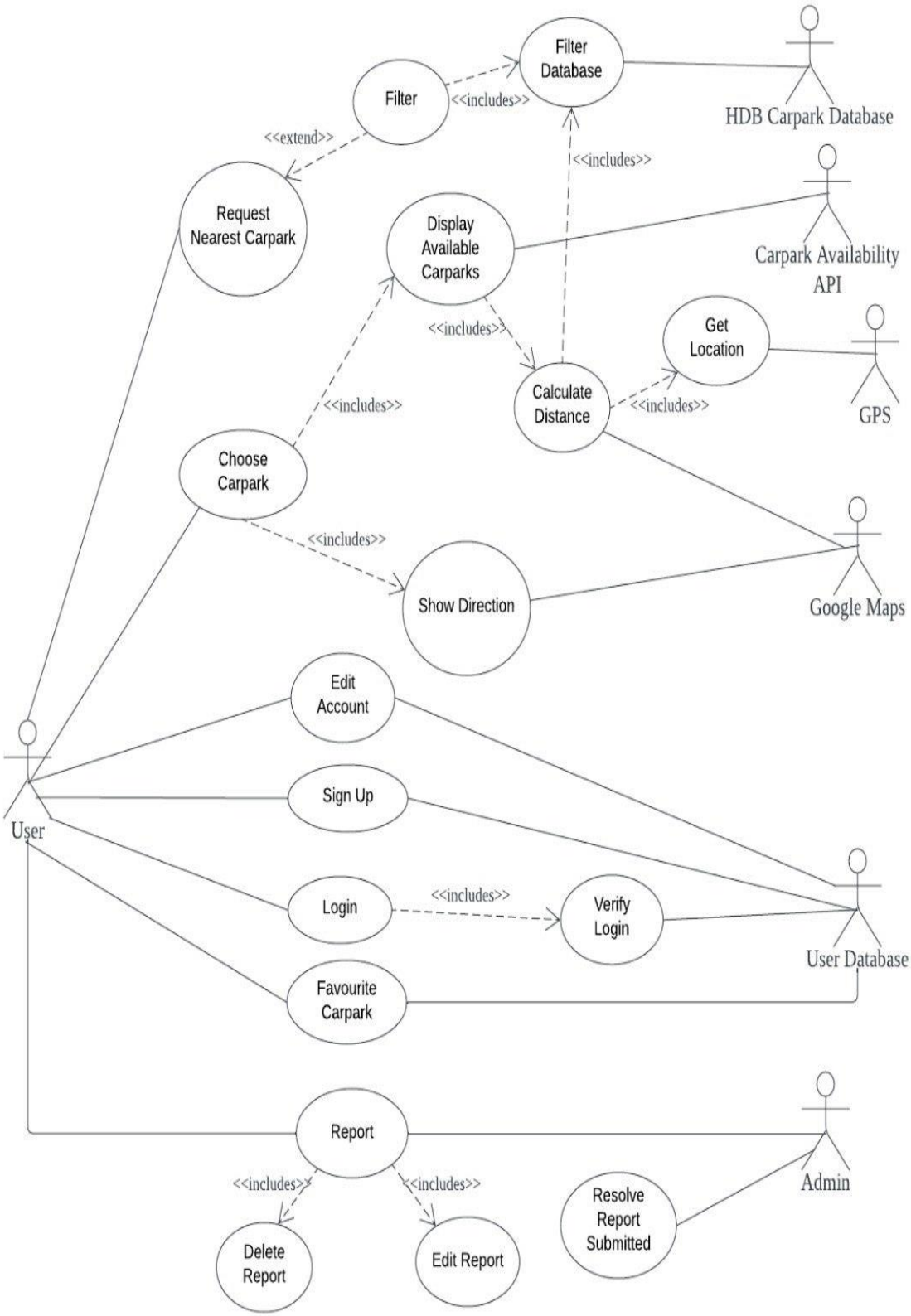
4.1 Our team will be responsible for the maintenance and upgrade of the website.
4.2 The structure should be well-documented and organized, making it easy for developers to understand and maintain.
4.3 The program should adhere to a consistent coding style and naming conventions across the entire codebase.

Data Dictionary

Term	Definition
User	A person who has a registered account and is using the parking app.
Carpark	Physical location where cars can be parked.
Availability	The status of a parking space, either available or occupied.
Opening hours	The hours during which a car park is open
Night Parking	The status in which the car park offers parking from 7pm - 7am
Electronic Parking System	A cashless system that uses EPS antenna to read the number of ERP In-Vehicle Unit (IU) at the entry and exit of the car park
Coupon Parking	Parking System in which drivers without coupons will have to pay through Parking.sg mobile app or display valid parking coupons if they do not have valid season parking.
Multi-Storey Car Park	Car park that has parking spaces allocated to multiple storeys
Basement Car Park	Car park that has parking spaces allocated underground
Surface Car Park	Car park that has parking spaces allocated on road level.
Destination	Location that the user will be going from his/her current position
Profile	A user's personal information, such as their name, email address, and phone number.
Username	A unique name that is associated with the user used for login.
Password	A string of characters (letters, numbers, special symbols) that allow the user to access their account
GPS	A utility system that provides users with positioning services.

Distance	The distance from the user to available carparks.
Report	A short description which allows users to talk about their experience with the car park application.
Case	Report submitted by the user which is view by the admin
Favourite	An indicator that helps the system to filter the lists of places that the user commonly goes to.
Filter	Different types and conditions of car parks can be chosen for the users for searching

Use Case Diagram



Use Case Description

Priority = High, Medium, Low

Use Case ID:	T301		
Use Case Name:	Sign Up		
Created By:	Zi Xuan	Last Updated By:	Zi Xuan
Date Created:	02/09/2023	Date Last Updated:	06/09/2023

Actor:	User, Admin, User Database
Description:	Users who do not have a pre-existing account will be able to initialise an account with his/her personal details, in order to access the features of the web application.
Preconditions:	1. User must not have a pre-existing account. a. Username has not been taken. b. Password must fulfill the following requirements: minimum 8 characters, 1 lowercase letter, 1 uppercase letter, 1 digit and 1 special character.
Postconditions:	1. System will display a message upon successful account creation. 2. System will redirect user to login page.
Priority:	High
Frequency of Use:	High
Flow of Events:	1. User will click on button “Create a new account” 2. User will key in email, username and password; password must be entered twice and match identically 3. System will validate information by searching for username and email in user database 4. System will add new entry for user information in user database 5. System redirects user to login page

Alternative Flows:	AF-S2: Password does not match requirements 1. System will display the message “Your password must be 8 digits and contain 1 uppercase, 1 lowercase letter, 1 digit and 1 special character!” 2. System will return to main flow Step 2 AF-S3: Username or email already exists 1. System will display the message “Your username or email is taken!” 2. System will return to main flow Step 2
Exceptions:	
Includes:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	T302		
Use Case Name:	Login		
Created By:	Zi Xuan	Last Updated By:	Zi Xuan
Date Created:	02/09/2023	Date Last Updated:	02/09/2023

Actor:	User, Admin, User Database
Description:	This feature will allow users (and admin) to enter the main page of the web application to access its main functions. The user must input a valid username and corresponding password to successfully login.
Preconditions:	User must have a pre-existing account created in the User Database
Postconditions:	User will be redirected to main page upon successful login
Priority:	High
Frequency of Use:	High
Flow of Events:	1. User enters username and password in the login form. 2. User clicks on the “Login” button to submit the form. 3. The system will match the username and password entered in the form, with the records in User Database. 4. System will redirect user to main page.
Alternative Flows:	AF-S2: Username is not found in User Database 1. System will print an error message “Username is not found, please enter a valid username!” 2. System returns to main flow Step 1 AF-S2: Username is found in User Database, but password does not match 1. System will print an error message “Incorrect password! Please try again.”

	2. System returns to main flow Step 1
Exceptions:	
Includes:	
Special Requirements:	
Assumptions:	User already has a pre-existing account created
Notes and Issues:	

Use Case ID:	T303		
Use Case Name:	Edit Account		
Created By:	Zi Xuan	Last Updated By:	Zi Xuan
Date Created:	02/09/2023	Date Last Updated:	02/09/2023

Actor:	User, User Database
Description:	This case will allow users to change the password associated with their current accounts, to a different password. Login will require the new password to be entered for subsequent login attempts.
Preconditions:	1. User must have a pre-existing account 2. User must have a pre-existing valid password
Postconditions:	1. System will send a success message “Your password has been successfully updated!” 2. User will be redirected to login page after successful password change
Priority:	Low
Frequency of Use:	Low
Flow of Events:	1. User clicks on “forgot password” button on login page 2. System redirects user to “Change Password” page 3. User enters username 4. System validates username entered with User Database 5. User enters new password twice 6. System validates if password matches requirements 7. System sends success message “Your password has been successfully updated!” and redirects user to login page
Alternative Flows:	AF-S4: User enters invalid username, i.e. username not found in User Database 1. System sends an error message “Incorrect username!” 2. System returns to main flow Step 3 AF-S6: User enters non-matching password when confirming password

	1. System sends an error message “Passwords do not match, please re-enter password!” 2. System returns to main flow Step 5
Exceptions:	
Includes:	
Special Requirements:	
Assumptions:	User will change old password which is non-identical to the new password
Notes and Issues:	

Use Case ID:	T304		
Use Case Name:	Favourite Car Park		
Created By:	Zi Xuan	Last Updated By:	Zi Xuan
Date Created:	02/09/2023	Date Last Updated:	02/09/2023

Actor:	User, User Database, HDB Carpark Database
Description:	This case will allow users to select their most frequented car parks and add them to a favourite list. Users will thereafter be able to select carparks from this favourite list without searching for their addresses.
Preconditions:	1. User must know the address of the carpark they wish to add 2. Carpark selected must not be added on the list previously
Postconditions:	1. Favourite list will be updated with newly added car park 2. System will redirect user back to “Add Favourite Car Park” page
Priority:	Medium
Frequency of Use:	Medium - High
Flow of Events:	1. User enters address of car park they wish to add to favourites 2. System will validate for address in HDB Carpark Database by finding corresponding car park name 3. System will send a pop-up asking user to confirm selection 4. User clicks on “confirm” button 5. System adds car park to favourite list and redirects user back to “Add Favourite Car Park” page
Alternative Flows:	AF-S2: User enters invalid address 1. System will display a message “No car parks found” 2. User must clear search field 3. System returns to main flow Step 1 AF-S2: User has previously added the car park to favourite list 1. System sends an error message “Carpark is already on favourite list!”

	2. System returns to main flow Step 1 AF-S3: User clicks “cancel” button 1. System returns to main flow Step 1
Exceptions:	
Includes:	
Special Requirements:	
Assumptions:	User has a pre-existing account, and has not previously added the car park.
Notes and Issues:	

Use Case ID:	T305		
Use Case Name:	Report		
Created By:	Zi Xuan	Last Updated By:	Zi Xuan
Date Created:	02/09/2023	Date Last Updated:	02/09/2023

Actor:	User, Admin
Description:	This case allows users to submit report to report any operational and infrastructural faults at HDB car parks. Users will be able to provide a short text description to explain the fault.
Preconditions:	1. User must know the address of the car park they wish to report
Postconditions:	1. System will show a success message “Your Report has been submitted!”
Priority:	Low
Frequency of Use:	Low
Flow of Events:	1. User enters car park address to report 2. System validates car park address 3. User enters text description 4. User clicks on “submit report” button 5. System prompts to “confirm” or “cancel” submission 6. User clicks “confirm” button 7. System will send this review to the admin, and display a success message “Your has been submitted!”
Alternative Flows:	AF-S2: Invalid or incorrect carpark address 1. System will display error message “No car parks found” 2. System returns to main flow Step 1 AF-S4: User does not input text 1. System will send an error message “Please enter text!” 2. System returns to main flow Step 3

	AF-S5: The user clicks on “cancel button” 1. System returns to main flow Step 4
Exceptions:	
Includes:	
Special Requirements:	
Assumptions:	User must know car park address
Notes and Issues:	

Use Case ID:	T306		
Use Case Name:	Delete Report		
Created By:	Zi Xuan	Last Updated By:	Zi Xuan
Date Created:	02/09/2023	Date Last Updated:	02/09/2023

Actor:	User, Admin
Description:	This case allows users to delete a previously submitted report from a list of report the user has submitted.
Preconditions:	1. User must have at least 1 or more pending report
Postconditions:	1. List of report is updated, and deleted report is no longer visible 2. System displays message “Your report has been deleted”
Priority:	High
Frequency of Use:	High
Flow of Events:	1. User selects report to be deleted from list of report 2. User clicks on “delete” button 3. System prompts user to confirm deletion with “confirm” or “cancel button” 4. User clicks on “confirm” button 5. System removes report from User and Admin end 6. System displays a success message “Your report has been successfully deleted!” and redirects user back to list of report
Alternative Flows:	AF-S3: User clicks on “cancel” button 1. System returns to main flow Step 1
Exceptions:	
Includes:	

Special Requirements:	
Assumptions:	User has at least 1 or more pending report
Notes and Issues:	

Use Case ID:	T307		
Use Case Name:	Edit Report		
Created By:	Zi Xuan	Last Updated By:	Zi Xuan
Date Created:	02/09/2023	Date Last Updated:	02/09/2023

Actor:	User, Admin
Description:	This case allows users to edit any pre-existing report. Users may edit the text description submitted. Both the user and admin will be able to see the changes reflected on the other end.
Preconditions:	1. User must have 1 or more pre-existing pending report
Postconditions:	1. User and Admin can see edited report with the changes made
Priority:	High
Frequency of Use:	High
Flow of Events:	1. User selects a report from list of report 2. User clicks on “edit button” 3. User edits text description 4. System validates form fields 5. System prompts user to “confirm” or “cancel” edits. 6. User clicks on “confirm” button 7. System reflect changes in user and admin end 8. System displays a success message “Your report has been updated!”
Alternative Flows:	AF-S4: Text field is empty 1. System displays error message “Fill in text fields!” 2. System returns to main flow Step 3 AF-S5: User clicks on “cancel” button 1. System returns to main flow Step 3

Exceptions:	
Includes:	
Special Requirements:	
Assumptions:	User only can edit pending report
Notes and Issues:	

Use Case ID:	T308		
Use Case Name:	Resolve Report Submitted		
Created By:	Zi Xuan	Last Updated By:	Zi Xuan
Date Created:	02/09/2023	Date Last Updated:	02/09/2023

Actor:	Admin, User
Description:	This case allows the admin to resolve and clear pending report by users, once necessary administrative actions have been executed to remedy the faults. The report resolved will subsequently be removed from the list of report for both admin and user.
Preconditions:	1. There is a pre-existing report submitted by user 2. Measures have been taken in response to user's report
Postconditions:	1. System displays a success message "Report has been resolved" 2. System redirects admin back to list of report
Priority:	High
Frequency of Use:	High
Flow of Events:	1. Admin selects report from the list of report submitted by user 2. Admin clicks on "resolve" button 3. System prompts user with "confirm" and "cancel" button 4. Admin clicks on "confirm" button 5. System removes report from both user and admin's list of report 6. System displays a success message "Report has been resolved" 7. System redirects admin back to list of report
Alternative Flows:	AS-S3: Admin clicks on "cancel" button 1. System returns to main flow Step 1
Exceptions:	

Includes:	
Special Requirements:	
Assumptions:	There is 1 or more pre-existing report submitted by users
Notes and Issues:	

Use Case ID:	T309		
Use Case Name:	Filter		
Created By:	Sheng Da	Last Updated By:	Sheng Da
Date Created:	02/09/2023	Date Last Updated:	05/09/2023

Actor:	User, HDB Car Park Database
Description:	User will be able to filter for types of car park they desire based on the given fields: type of car park, type of parking, night parking availability
Preconditions:	User must successfully login into the System
Postconditions:	User hit the small triangle button after selecting their filter
Priority:	High
Frequency of Use:	High
Flow of Events:	1. User will click on the triangle filter button 2. There will be several tick boxes for user to choose: a. Tick Boxes of “Multi-Storey Car Parks”, “Basement Car Parks” and “Surface Car Parks” under Types of Car Parks Label b. Tick Boxes of “Coupon Parking” and “Electronic Parking System” under Types of Parking System c. Tick Box of “Night Parking” under Night Parking Availability Label 3. Once the user finishes setting their filter, they will click on the triangle filter button again 4. The Conditions they set will be use to subset from the copy of the original database from the HDB Car Park DataBase 5. This subset of data will then be passed on to use case “Calculate Distance”(T312) for further processing
Alternative Flows:	AFS3: If the user did not choose to select any filter 1. The following will be the default filter for the system: a. All the tick boxes of Type of Car Parks will be ticked.

	b. All the tick boxes of Type of Parking System will be ticked. c. The Tick box of “Night Parking” WILL NOT be ticked.
Exceptions:	
Includes:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	T310		
Use Case Name:	Request Nearest Car Park		
Created By:	Sheng Da	Last Updated By:	Sheng Da
Date Created:	02/09/2023	Date Last Updated:	05/09/2023

Actor:	User
Description:	User will type the place that they wish to go. System will help to search available car park based on filter condition within 2km radius
Preconditions:	User Must Login into the System
Postconditions:	Display of 5 or less car parks will be shown that meet the user predefined filter.
Priority:	High
Frequency of Use:	High
Flow of Events:	1. User will type their destination location they wish to visit 2. User can choose to apply filter (T309) to choose the type of car park they desired 3. System will search for available car parks within 2km distance that meets the user filter conditions using the coordinates of the destination they type as the centre of the circle. 4. Top 5 or less available car parks that meet condition sorted based on distance from the destination location (with nearest on the top) will be displayed for user to choose in use case “Choose Car Park” (T313)
Alternative Flows:	AFS1: User did not type their destination when searching for the car park. 1. User hit search without typing their destination address will have their current position as the centre of the circle instead of the destination they type in step 3 2. System will search for available car parks within 2km distance that meets the user filter conditions

	3. Top 5 or less available car parks that meet condition sorted based on distance from current user position (with nearest on the top) will be displayed for user to choose in use case “Choose Car Park” (T313) AFS2: User did not choose to apply any filter. 1. User hit search without applying the filter. In the this scenario, default filter describe in Alternative Flow of T309 will be used
Exceptions:	
Includes:	
Special Requirements:	
Assumptions:	1. User type in valid Singapore location in the search box as their destination location. 2. Device supports GPS 3. Device supports Internet connection
Notes and Issues:	

Use Case ID:	T311		
Use Case Name:	Get Location		
Created By:	Sheng Da	Last Updated By:	Sheng Da
Date Created:	02/09/2023	Date Last Updated:	05/09/2023

Actor:	GPS
Description:	Get the current position of the user and/or the destination
Preconditions:	
Postconditions:	
Priority:	High
Frequency of Use:	High
Flow of Events:	1. The GPS will get the destination location in the search box ,car parks from the filtered database, and the current user position
Alternative Flows:	AFS1: If the user did not type the destination location in the search box in use case " Request for Nearest Car Park" 1. The GPS will get the current use location and the car parks from the filtered database.
Exceptions:	
Includes:	
Special Requirements:	
Assumptions:	1. Device support GPS 2. Device support Internet Connection

Notes and Issues:	

Use Case ID:	T312		
Use Case Name:	Calculate Distance		
Created By:	Sheng Da	Last Updated By:	Sheng Da
Date Created:	02/09/2023	Date Last Updated:	05/09/2023

Actor:	GPS, Google Maps
Description:	Calculate the distances based on the given coordinates.
Preconditions:	Inputs of the address of the destination location or the current user position
Postconditions:	Return the distances of given coordinates
Priority:	High
Frequency of Use:	High
Flow of Events:	1. The distance of the destination location and available car park that meets filter conditions will be calculated. 2. The top 5 nearest distance from the destination location or less will be picked out. 3. The distance of these 5 locations and the user current position will be calculated.
Alternative Flows:	AFS1: The user did not input any destination location in the search box in use case “Request Nearest Car Park” T310. 1. The distance of the user's current user position and the car parks that meet filter conditions will be calculated. 2. Top 5 nearest car park from the current user position will be calculated
Exceptions:	
Includes:	

Special Requirements:	
Assumptions:	1. Device support GPS 2. Device support Internet Connection
Notes and Issues:	

Use Case ID:	T313		
Use Case Name:	Choose Car Park		
Created By:	Sheng Da	Last Updated By:	Sheng Da
Date Created:	02/09/2023	Date Last Updated:	05/09/2023

Actor:	User
Description:	The user will choose 1 of the car parks that he/she wishes to go from the display of available car parks
Preconditions:	At most 5 of the nearest car parks that meets the condition will be shown
Postconditions:	The user chooses the car park
Priority:	High
Frequency of Use:	High
Flow of Events:	1. After the distance calculation in use case “Calculate Distance” (T312), at most 5 of car park will be shown to the user 2. Information in use case “Display Available Car Parks” (T315) will be shown to the user. 3. User will choose 1 of the car park that he/she wants to go 4. This input will feed the system with the information needed for the use case “Show Direction” (T314)
Alternative Flows:	AFS2: No car park shown in the available car parks display 1. The system will output “No available car park that meets your filter condition.”
Exceptions:	
Includes:	

Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	T314		
Use Case Name:	Show Direction		
Created By:	Sheng Da	Last Updated By:	Sheng Da
Date Created:	02/09/2023	Date Last Updated:	02/09/2023

Actor:	User, Google Map, GPS
Description:	The map will show the direction from the current user position to the destination car park of choice
Preconditions:	The user select 1 of the car park from the “Choose Car Park” use case
Postconditions:	The user reach destination
Priority:	High
Frequency of Use:	High
Flow of Events:	1. The map will then show the user the direction to the destination car park 2. Once the user parked their car, they will click on the button “X” at the top right of the web app. 3. After clicking the button, it will return to the homepage of the web app.
Alternative Flows:	
Exceptions:	
Includes:	
Special Requirements:	
Assumptions:	1. Device support GPS 2. Device support Internet Connectivity

Notes and Issues:	

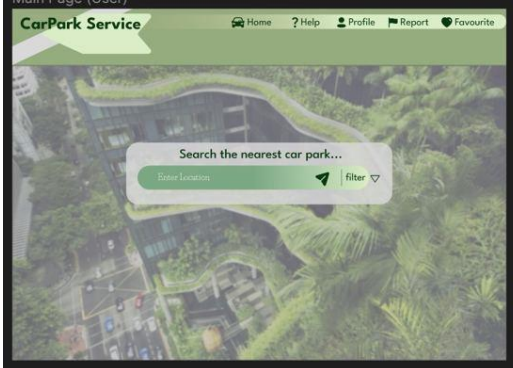
Use Case ID:	T315		
Use Case Name:	Display Available Car Parks		
Created By:	Sheng Da	Last Updated By:	Sheng Da
Date Created:	05/09/2023	Date Last Updated:	05/09/2023

Actor:	User
Description:	The display will show top 5 available car parks sorted based on the destination location the user input
Preconditions:	The user complete use case “Request Nearest Car” Parks” (T310)
Postconditions:	The user selected the car park he/she wishes to go
Priority:	High
Frequency of Use:	High
Flow of Events:	1. Information of the top 5 available car parks sorted based on the destination location of the user input in use case “Request Nearest Car Park” (T310) sorted top down, with nearest distance from the destination location on the top will be shown to the user. 2. Other information displayed to the user include Address of car park, Distance from current position and the number of available lots
Alternative Flows:	ASF1: User did not input destination location in T310. 1. Information of the top 5 available car parks sorted based on the current user position and the carpark sorted top down, with nearest distance from the user current position on the top will be shown to the user
Exceptions:	

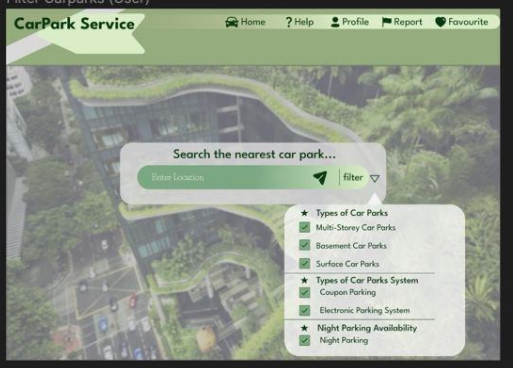
Includes:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

MOCK UP UI

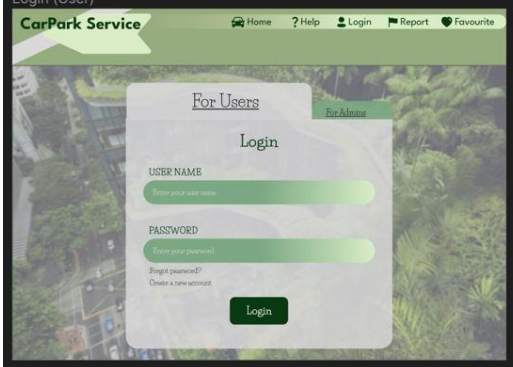
Main Page (User)



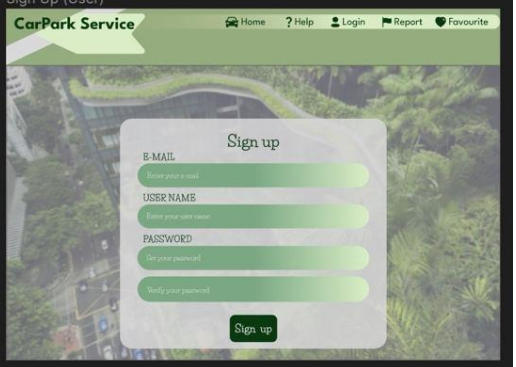
Filter Carparks (User)



Login (User)

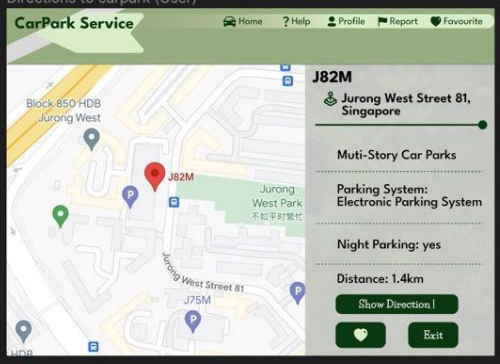


Sign Up (User)

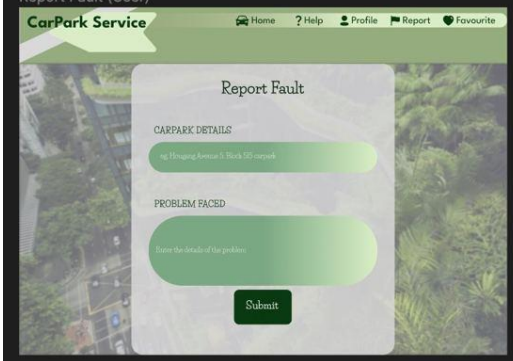


Carpark List (User)

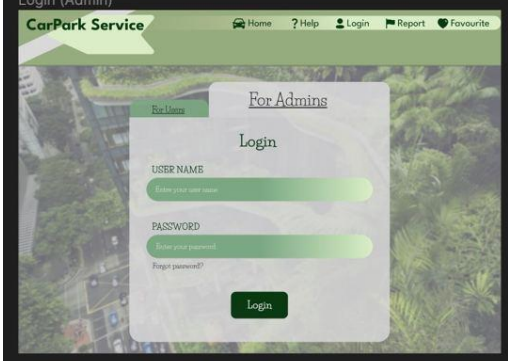
Directions to carpark (User)



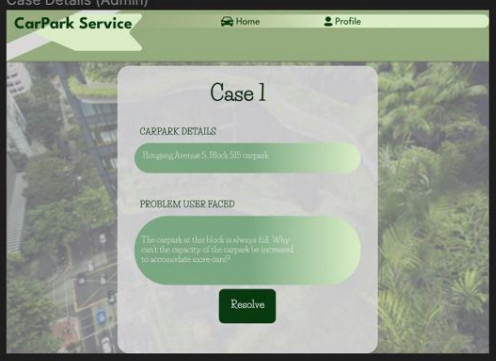
Report Fault (User)



Login (Admin)



Case Details (Admin)



Case List (Admin)