Application Name: Park your Car (@group12005)
Target User: Drivers and Motorists
Application type: Telegram Bot

Functional Requirements

1. The system must be able to provide a brief introduction to the Bot upon launching.
    1.1. The system must provide instructions to the users on how to use the Bot.
    1.2. The system must provide a help command should the user require any help.

2. Users must be able to query the system.
    2.1. Users must send current location to begin.
        2.1.1. The query must be passed to the Bot via an on screen button.
            2.1.1.1. The on screen button must hide the user's keyboard
        2.1.2. The query must show the map of the user's current location.
            2.1.2.1. A pin shall be marked on the map to show user's location
        2.1.3. The query format must be in Latitude and Longitude for every query irregardless of device type.
    2.2. Users shall query with only one request at any one time.
        2.2.1. Users shall be able to clear this request upon completion to enter a new request.
        2.2.2. The user must be able to view their search history.

3. The System must be able to convert the user's location from Latitude and Longitude to X and Y coordinates.
    3.1. The conversion must use all the decimal points.

4. The query shall return data depending on the choice made by the user.
    4.1. The system must show the top 5 car park locations that are nearest to the user's location.
        4.1.1. The system shall list down the details of the top 5 car parks that are closest.
            4.1.1.1. The details must include the car park number, total lots and total lots available.
    4.2. The system must be able to receive a keyword for the destination address from the user.
        4.2.1. The system shall only return 1 selected carpark if there exists an address that completely matches the user's input.
        4.2.2. The system shall list down the details of 5 selected carparks whose address partially matches the user's input.
            4.2.2.1. The details must include the carpark number, total lots and total lots available.
        4.2.3. The system must report an error if the previous keyword does not match any existing car park.
            4.2.3.1. The system must prompt the user to enter a new keyword.

5. The system must be able to calculate the parking price based on the user's inputs, and the current date and time.
    5.1. The parking price returned must be rounded to 2 decimal places.
        5.1.1. The price returned must be in Singapore Dollars.
    5.2. The system must report an error if the time range entered is invalid.

6. The system must refresh periodically to retrieve the current number of available parking lots.
    6.1. The system must retrieve the updated number of available parking lots upon the user querying query.

7. The system must be able to calculate the distance between the user's current location and the user's selected car park.
    7.1. The system shall display the distance in kilometers.

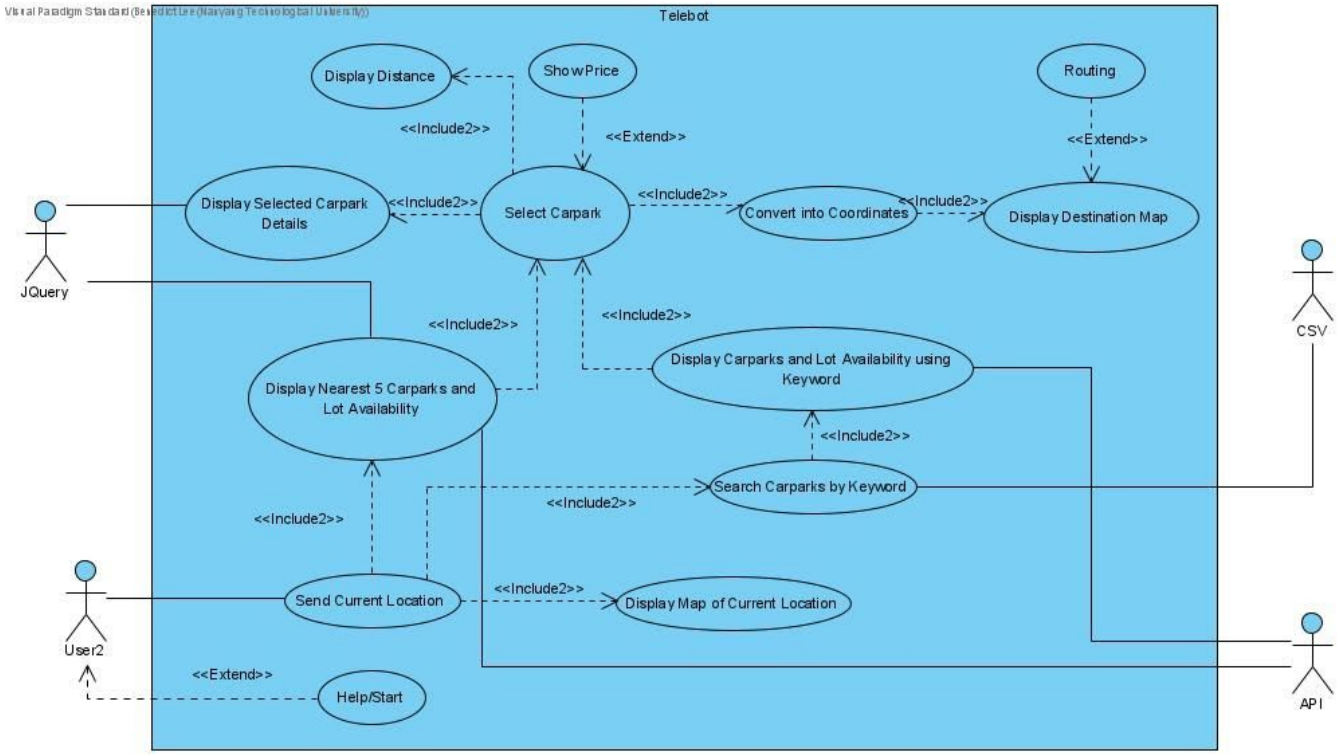
Non Functional Requirements

1. Security:
    1.1. The data sent from the user, and the data received must not be stored.
    1.2. The hosting application must be secured using HTTPS encryption and contain a valid security certificate.
2. Reliability:
    2.1. The website hosting the application must be available for use 24/7.
    2.2. The query response must provide accurate, updated and consistent information regarding the location of the nearest, or nearby car parks.
3. Performance:
    3.1. A query for the nearest carpark should not take longer than 8 seconds to process before returning the results in the chat.
    3.2. The system must restart within 10minutes of downtime.
4. Usability:
    4.1. The user must not need to spend more than 2 minutes using the application to find a carpark.
    4.2. There will be different bubble text boxes and help messages to guide the user.
5. Scalability:
    5.1. Both the database and the website hosting the telegram bot must be able to simultaneously handle up to 60 queries at any one time.
    5.2. The telegram bot must be able to handle at least 60 users at any one time.
6. User Friendliness:
    6.1. The telegram bot must remain user friendly and readable on different mobile and tablet screen sizes.
    6.2. The current location shall be marked out with an indicator.
    6.3. The applications shall provide the general direction to the destination for the user to navigate.

Data Dictionary

Term	**Definition**
HDB	Housing Development Board, Singapore's public housing authority and statutory board under the Ministry of National Development.
Carpark	An area or multi story building where cars or other vehicles may be left temporarily.
Car Park Deck	Number of storeys
Chat Bot	A software application used to conduct an on-line chat conversation via text or text-to-speech, in lieu of providing direct contact with a live human agent.
Gantry	A barrier located at the entry and exit points off a car park to register incoming vehicles and deducting parking fees from outgoing vehicles.
Parking Lot	Also known as a car lot, is a cleared area that is intended for parking vehicles.
Basement Carpark	A carpark located below ground level.
Surface Car Park	Parking which is not enclosed or created by a structure and is allocated an area on ground level
Multi-Story Carpark	Building designed for car parking where there are a number of floors or levels on which parking takes place.
Electronic Parking System	A cashless system that wirelessly reads an In-vehicle unit containing a cashcard at the entry and exit points of a carpark.
PH	Parking is available at this location on a public holiday.
BLK	An apartment block.
Night Parking	Short-Term parking is available from 10.30pm to 7am on the following day at this location.
Car Park No.	A number designated to the carpark, which can be used to identify it easily.
Telegram	A messaging application which is able to interact with our bot.
Database	An organized collection of data, generally stored and accessed electronically from a computer system.

Use Case Diagram:



Use cases:

1) Send Current Location
2) Help/Start
3) Display Map of Current Location
4) Search Carparks by Keyword
5) Display Nearest 5 Carparks and Lot Availability
6) Display Carparks and Lot Availability using Keyword
7) Select Carpark
8) Display Selected Carpark Details
9) Display Distance
10) Show Price
11) Convert into Coordinates
12) Display Destination Map
13) Routing

Use Case Description

Use Case ID:	01		
Use Case Name:	Send Current Location		
Created By:	Jun Han	**Last Updated By:**	Jun Han
Date Created	4/11/20	**Date Last Updated**	4/11/20
Actor:	User		
Description:	Prompt for the user to send current location using GPS.		
Preconditions:	The user must have location services enabled and an Internet connection		
Postconditions:	The user will be able to use the various functions with relation to the current location		
Priority:	High.		
Frequency of Use:	Often.		
Flow of Events:	1. The user accesses the Telegram Bot and invokes the command to start. 2. Bot prompts the user to send his/her current address 3. User sends his/her current address using the prompt		
Alternative Flows:			
Exceptions:	1. User's location service is not enabled or unavailable. 2. System will display the message "An error has occurred" 3. Return to Use Case 2.		
Includes:	-		
Special Requirements:	-		
Assumptions:	User has an internet connection. User's device has location services turned on.		
Notes and Issues:	-		

Use Case ID:	02		
Use Case Name:	Help/Start		
Created By:	Benedict	**Last Updated By:**	Benedict
Date Created	4/11/20	**Date Last Updated**	4/11/20
Actor:	User		
Description:	To help user to begin using the Bot		
Preconditions:	The user must have location services enabled and an Internet connection		
Postconditions:	The user will be able to use the GPS capability of their device to send their current location		
Priority:	High.		
Frequency of Use:	Often.		
Flow of Events:	1. The user accesses the Telegram Bot and invokes the command to start		
Alternative Flows:			
Exceptions:			
Includes:	-		
Special Requirements:	-		
Assumptions:	User has an internet connection. User's device has location services turned on.		
Notes and Issues:	-		

Use Case ID:	03		
Use Case Name:	Display Map of Current Location		
Created By:	Zheng Xuan	**Last Updated By:**	Zheng Xuan
Date Created	4/11/20	**Date Last Updated**	4/11/20
Actor:	User		
Description:	A map of the user's current location will be displayed		
Preconditions:	User must have sent their current location		
Postconditions:	The user will be able to see the surrounding area of the location he/she has sent on a displayed map.		
Priority:	High.		
Frequency of Use:	Frequent.		
Flow of Events:	1. A map is shown with a pin on the address provided.		
Alternative Flows:	-		
Exceptions:	-		
Includes:	-		
Special Requirements:	-		
Assumptions:	The user's current location has been sent. User has an internet connection. User's device has location services turned on.		
Notes and Issues:	-		

Use Case ID:	04		
Use Case Name:	Search Car Parks by Keyword		
Created By:	Sheng kai	**Last Updated By:**	Sheng Kai
Date Created	4/11/20	**Date Last Updated**	4/11/20
Actor:	User, CSV		
Description:	User input keywords to match the addresses of the car parks in the database		
Preconditions:	-		
Postconditions:	Car parks matching the keyword will be generated		
Priority:	Medium.		
Frequency of Use:	Medium.		
Flow of Events:	1. The user selects the option to enter the keyword 2. The bot prompts the user for the keyword 3. The user enters the keyword 4. The system searches for potential matches with the database. 5. A list of 5 car parks matching the keyword is generated.		
Alternative Flows:	1. The user selects the option to enter the keyword 2. The bot prompts the user for the keyword 3. The user enters a keyword that matches the full address of a car park stored in the database. 4. That particular car park's number will be generated.		
Exceptions:	1. The user enters an invalid address. 2. The user enters an address that is not found in the database. 3. An error message is generated. 4. Return to Use Case 4.		
Includes:	-		
Special Requirements:	-		
Assumptions:	User has an internet connection. User's device has location services turned on.		
Notes and Issues:	-		

Use Case ID:	05		
Use Case Name:	Display Nearest 5 Car Parks and Lot Availability		
Created By:	Max	**Last Updated By:**	Max
Date Created	4/11/20	**Date Last Updated**	4/11/20
Actor:	User, API, JQuery		
Description:	Display a list of the 5 nearest car parks and their Lot Availability		
Preconditions:	User must have sent their current location		
Postconditions:	Display a list with 5 of the nearest car parks and their lot availability. The user will be able to select the car park for more functions.		
Priority:	Medium.		
Frequency of Use:	Medium		
Flow of Events:	1) The bot retrieves the data from JQuery and finds the nearest 5 car parks from the User's current location. 2) The bot retrieves the Lot Availability from API using the data from JQuery. 3) Lists the nearest 5 carparks and their Lot Availability to the User.		
Alternative Flows:	-		
Exceptions:	-		
Includes:	-		
Special Requirements:	-		
Assumptions:	The user's current location has been sent. User has an internet connection. User's device has location services turned on.		
Notes and Issues:	-		

Use Case ID:	06		
Use Case Name:	Display Car Parks and Lot Availability using Keyword		
Created By:	Jun Han	**Last Updated By:**	Sheng Kai
Date Created	4/11/20	**Date Last Updated**	4/11/20
Actor:	User, API		
Description:	Display a list of the car parks that matches the user's keyword and their Lot Availability		
Preconditions:	User must have input keywords		
Postconditions:	Display a list of the carparks from the CSV data and their lot availability. The user will be able to select the car park for more functions.		
Priority:	Medium.		
Frequency of Use:	Medium		
Flow of Events:	1. The bot retrieves the Lot Availability from API using the data from CSV 2. Display a list of the carparks from the CSV data and their lot availability to the User		
Alternative Flows:			
Exceptions:	-		
Includes:	-		
Special Requirements:	-		
Assumptions:	The user must have input keywords User has an internet connection. User's device has location services turned on.		
Notes and Issues:	-		

Use Case ID:	07		
Use Case Name:	Select Carpark		
Created By:	Max	**Last Updated By:**	Max
Date Created	4/11/20	**Date Last Updated**	4/11/20
Actor:	User		
Description:	Allows the user to select a specific car park from the list.		
Preconditions:	The user's selected the option to show nearest car parks. OR The user must have input keywords.		
Postconditions:	The user will be able to select the car park for more details..		
Priority:	Medium.		
Frequency of Use:	Often.		
Flow of Events:	1. The bot shows a list of car parks with its Lot Availability to the user. 2. The user selects a carpark number from the provided list.		
Alternative Flows:	-		
Exceptions:			
Includes:	-		
Special Requirements:	-		
Assumptions:	The user's current location has been sent. User has an internet connection. User's device has location services turned on.		
Notes and Issues:	-		

Use Case ID:	08		
Use Case Name:	Display Selected Carpark Details		
Created By:	Benedict	**Last Updated By:**	Benedict
Date Created	4/11/20	**Date Last Updated**	4/11/20
Actor:	User, JQuery		
Description:	Display more details about the selected carpark. Address, x coordinates, y coordinates, carpark type, type of parking system, short term parking, free parking, night parking, number of carpark decks, gantry height and carpark basement information will be shown.		
Preconditions:	The user's current location has been sent. A carpark has been selected from the list shown		
Postconditions:	The detailed information of the car park will be shown to the User		
Priority:	Medium.		
Frequency of Use:	Often.		
Flow of Events:	1. The user selects a carpark from the list. 2. The bot retrieves information from JQuery using the selected carpark information. 3. The bot displays the detailed information of the carpark to the user.		
Alternative Flows:	-		
Exceptions:	-		
Includes:	-		
Special Requirements:	-		
Assumptions:	The user's current location has been sent. User has an internet connection. User's device has location services turned on.		
Notes and Issues:	-		

Use Case ID:	09		
Use Case Name:	Display Distance		
Created By:	Max	**Last Updated By:**	Max
Date Created	4/11/20	**Date Last Updated**	4/11/20
Actor:	User		
Description:	Shows the distance to the selected carpark, from the User		
Preconditions:	The user's current location has been sent. A carpark has been selected from the list shown		
Postconditions:	The distance from the selected carpark to the User's current location will be shown in kilometres		
Priority:	Medium.		
Frequency of Use:	Seldom.		
Flow of Events:	1. The user selects a carpark from the list 2. The user's location is converted to X and Y coordinate. 3. The bot uses the selected carpark's X and Y coordinate to calculate the distance. 4. The bot displays the distance of the selected carpark to the user's current location.		
Alternative Flows:	-		
Exceptions:	-		
Includes:	-		
Special Requirements:	-		
Assumptions:	The user's current location has been sent. The user must have selected a carpark. User has an internet connection. User's device has location services turned on.		
Notes and Issues:	-		

Use Case ID:	10		
Use Case Name:	Show Price		
Created By:	Zheng Xuan	**Last Updated By:**	Zheng Xuan
Date Created	4/11/20	**Date Last Updated**	4/11/20
Actor:	User		
Description:	Provides the cost of parking given a starting date & time and ending date & time.		
Preconditions:	Users must have provided a valid range for the start date and time, and end date and time.		
Postconditions:	The cost of parking at the selected carpark within that selected timeframe will be displayed. OR The user receives an explanation from the Bot why the parking rate cost is unavailable.		
Priority:	Medium.		
Frequency of Use:	Often.		
Flow of Events:	1. The user selects the show price function 2. The bot prompts the user for the starting date & time and ending date & time of parking 3. The user enters the starting date & time and ending date & time of parking 4. The bot calculates the cost using the input from the user 5. The bot displays the cost to the user		
Alternative Flows:	-		
Exceptions:	1. The timeframe given by the user is invalid. 2. The bot will display an error message 3. The bot prompts the user for the starting date & time and ending date & time of parking again		
Includes:	-		
Special Requirements:	-		
Assumptions:	The user's current location has been sent. The user must have selected a car park. User has an internet connection.		

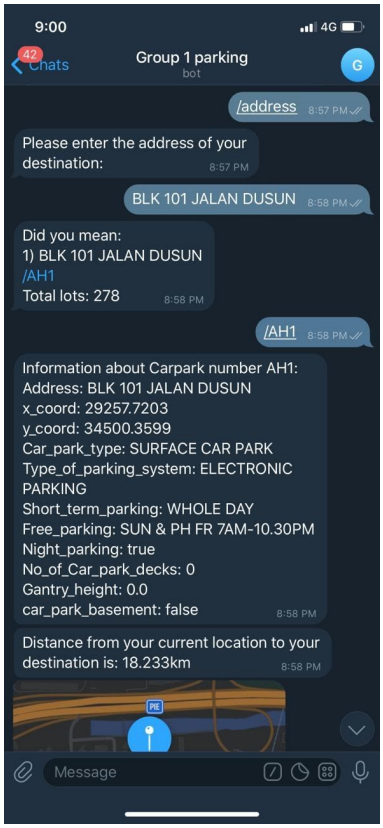
	User's device has location services turned on.
Notes and Issues:	-

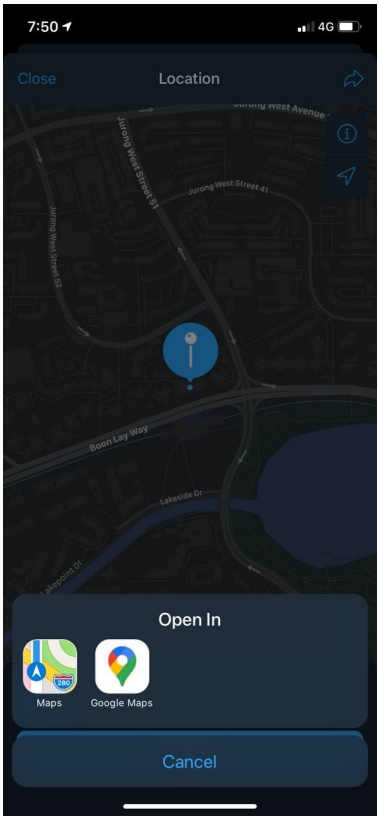
Use Case ID:	11		
Use Case Name:	Convert to coordinates		
Created By:	Sheng Kai	**Last Updated By:**	Sheng Kai
Date Created	4/11/20	**Date Last Updated**	4/11/20
Actor:	-		
Description:	Convert the user's selected carpark 'x' and 'y' coordinates into a set of latitude and longitude coordinates.		
Preconditions:	User must have selected a carpark		
Postconditions:	The latitude and longitude of the user's entered address will be generated.		
Priority:	High.		
Frequency of Use:	Often.		
Flow of Events:	1. The input carpark's 'x' and 'y' coordinates are converted into latitude and longitude coordinates.		
Alternative Flows:	-		
Exceptions:	-		
Includes:	-		
Special Requirements:	-		
Assumptions:	The user's current location has been sent. The user must have selected a car park. User has an internet connection. User's device has location services turned on.		
Notes and Issues:	-		

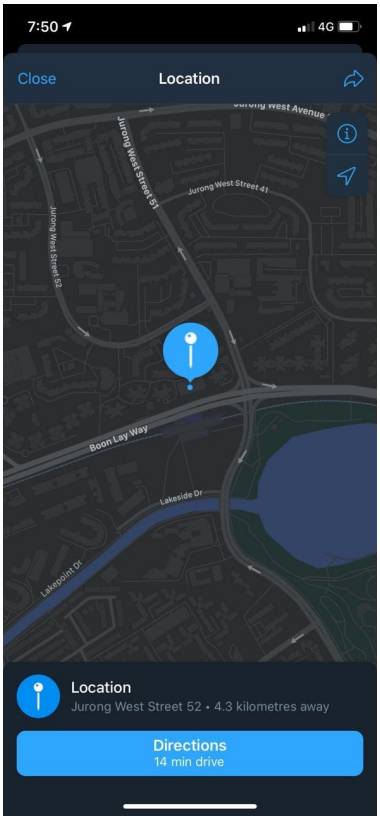
Use Case ID:	12		
Use Case Name:	Display destination map		
Created By:	Jun Han	**Last Updated By:**	Jun Han
Date Created	4/11/20	**Date Last Updated**	4/11/20
Actor:	User		
Description:	Displays the location around the user's selected carpark through a map.		
Preconditions:	User must have selected a carpark		
Postconditions:	A map showing the location around the user's selected carpark will be shown in the Telegram chat window.		
Priority:	Medium		
Frequency of Use:	Often		
Flow of Events:	1. The user selects a carpark from the list. 2. A map of the area around the selected carpark is generated by the bot. 3. The user clicks on the map. 4. The map is displayed to the user.		
Alternative Flows:	-		
Exceptions:	-		
Includes:	-		
Special Requirements:	-		
Assumptions:	The user's current location has been sent. The user must have selected a car park. User has an internet connection. User's device has location services turned on.		
Notes and Issues:	-		

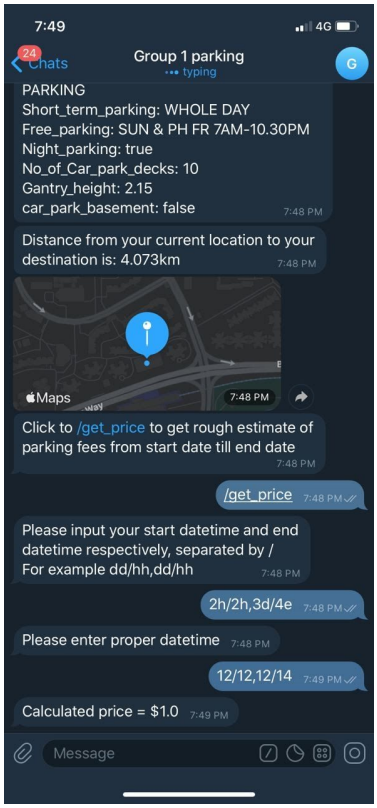
Use Case ID:	13		
Use Case Name:	Routing		
Created By:	Benedict	**Last Updated By:**	Benedict
Date Created	4/11/20	**Date Last Updated**	4/11/20
Use Case Name:			
Actor:	User		
Description:	Shows the user the approximate route on the map to reach their selected carpark.		
Preconditions:	The user must have selected a carpark. The user must have clicked on the map		
Postconditions:	A route is shown on the user's preferred map application, from the user's current location to the user's selected carpark		
Priority:	Medium		
Frequency of Use:	Often		
Flow of Events:	1. User clicks on the map 2. User clicks on "directions" 3. The user's preferred map application opens and displays a route from the user's current location to the user's selected carpark		
Alternative Flows:	-		
Exceptions:	-		
Includes:	-		
Special Requirements:	-		
Assumptions:	The user's current location has been sent. OR The user must have input keywords User has an internet connection. User's device has location services turned on. User has a compatible map application		
Notes and Issues:	-		

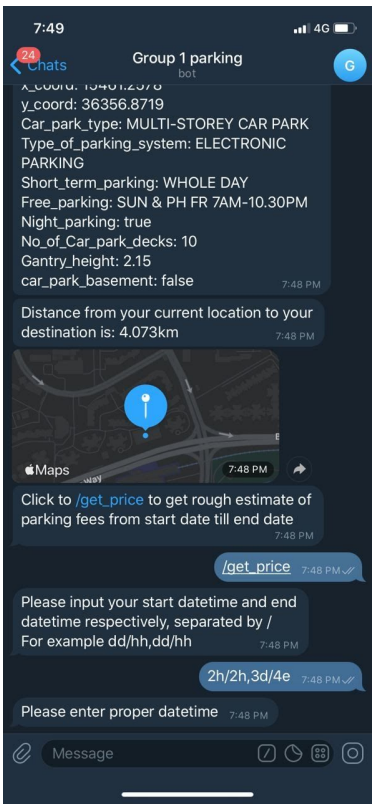
User Interface

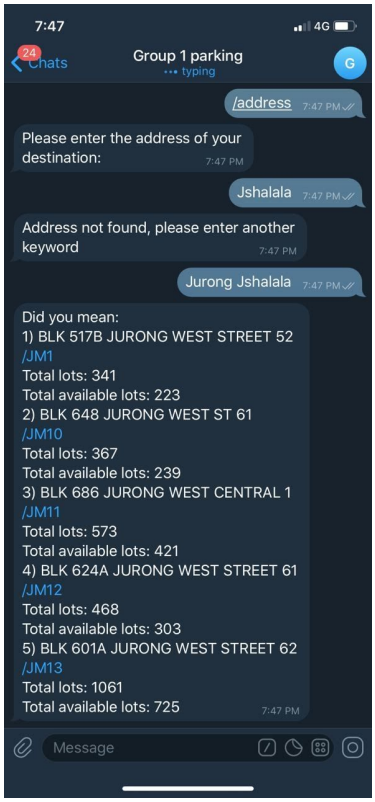


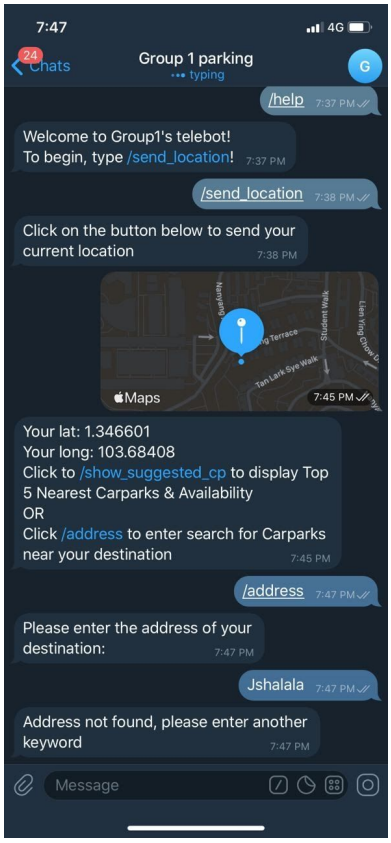


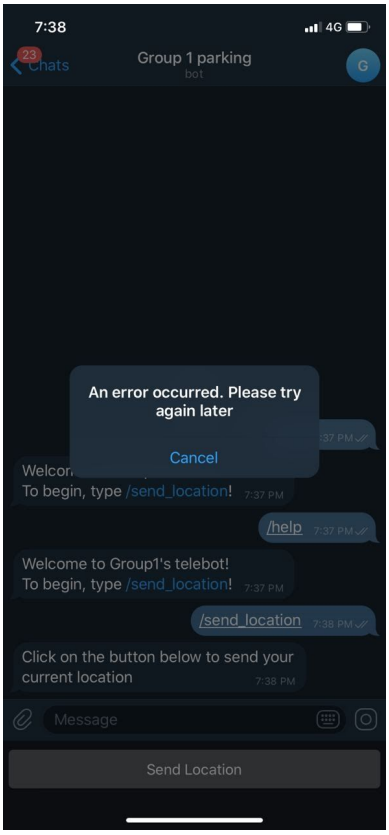


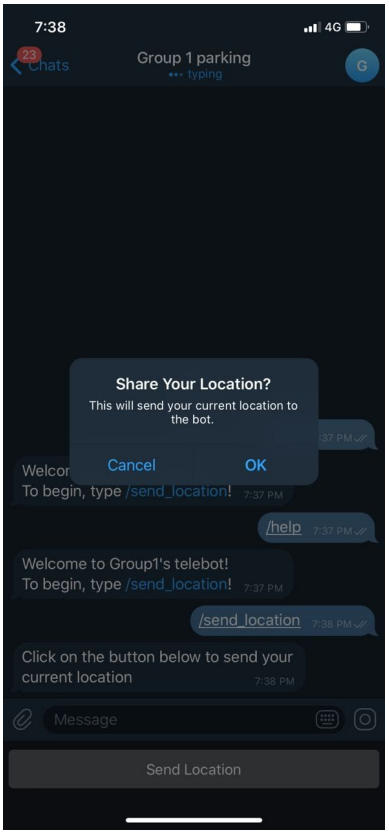


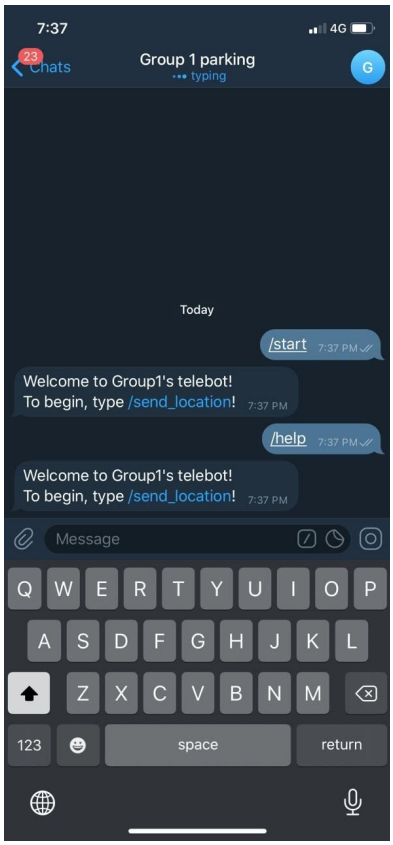





7:47
4G
24 Chats
Group 1 parking
typing
G
/help 7:37 PM
Welcome to Group1's telebot!
To begin, type /send_location! 7:37 PM
/send_location 7:38 PM
Click on the button below to send your current location 7:38 PM
Maps 7:45 PM
Your lat: 1.346601
Your long: 103.68408
Click to /show_suggested_cp to display Top 5 Nearest Carparks & Availability
OR
Click /address to enter search for Carparks near your destination 7:45 PM
/address 7:47 PM
Please enter the address of your destination: 7:47 PM
Jshalala 7:47 PM
Address not found, please enter another keyword 7:47 PM
Message


7:38
4G
23 Chats
Group 1 parking
bot
G
An error occurred. Please try again later
Cancel
To begin, type /send_location! 7:37 PM
/help 7:37 PM
Welcome to Group1's telebot!
To begin, type /send_location! 7:37 PM
/send_location 7:38 PM
Click on the button below to send your current location 7:38 PM
Message
Send Location


7:38
4G
23 Chats
Group 1 parking
typing
G
Share Your Location?
This will send your current location to the bot.
Cancel
OK
To begin, type /send_location! 7:37 PM
/help 7:37 PM
Welcome to Group1's telebot!
To begin, type /send_location! 7:37 PM
/send_location 7:38 PM
Click on the button below to send your current location 7:38 PM
Message
Send Location


7:37
4G
23 Chats
Group 1 parking
typing
G
Today
/start 7:37 PM
Welcome to Group1's telebot!
To begin, type /send_location! 7:37 PM
/help 7:37 PM
Welcome to Group1's telebot!
To begin, type /send_location! 7:37 PM
Message
Q W E R T Y U I O P
A S D F G H J K L
Z X C V B N M
123
space
return

Identified Actors:

1. User. (Initiating actor.)
2. JQuery.
3. CSV Database.
4. Telegram API.

Identified Entity Classes (Information being tracked by the system.) - Data Management.

1. HDB carpark JQuery database.
2. HDB carpark CSV database.
3. Carpark availability API.

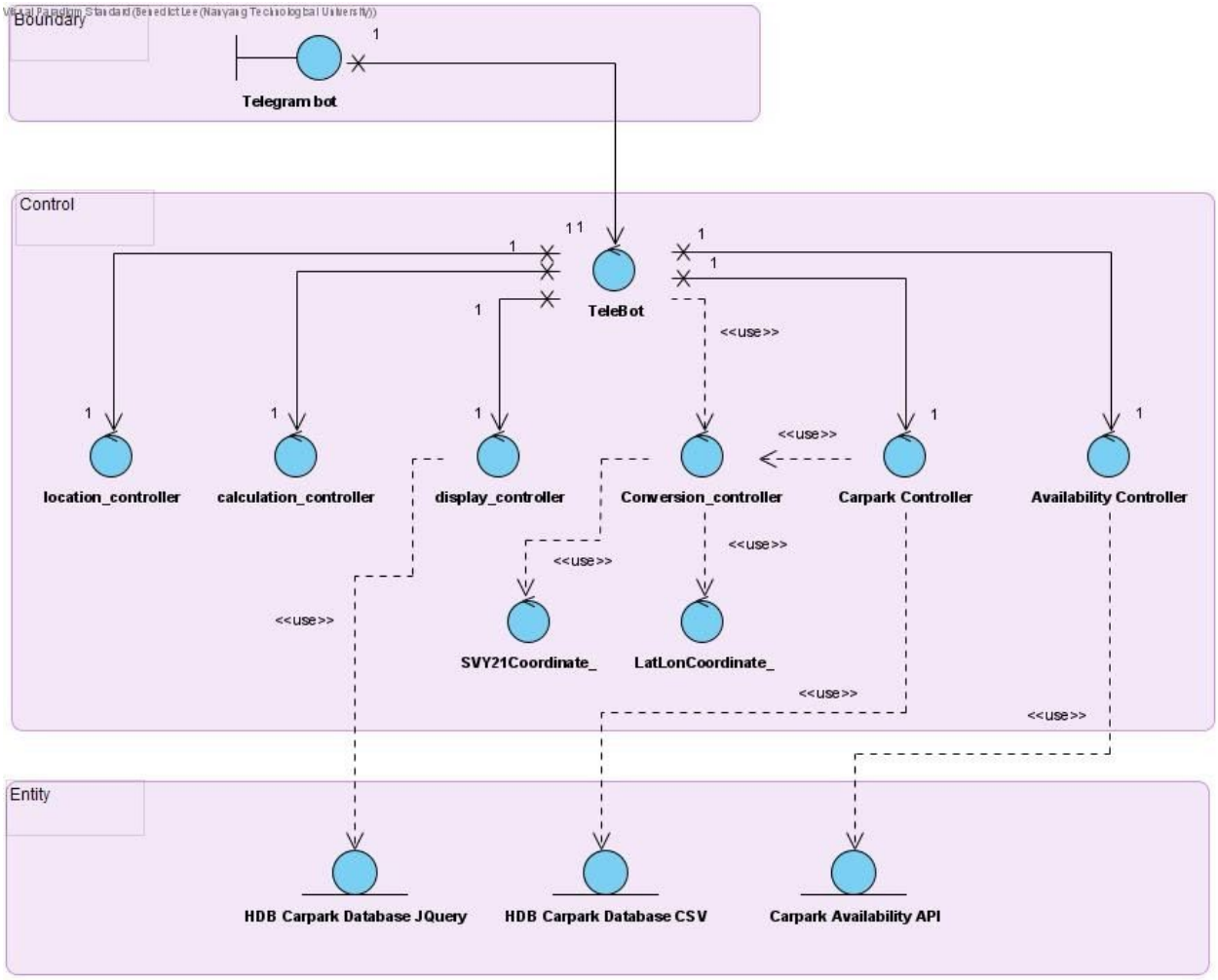
Identified Boundary Classes (Interaction between actor and system.) - Interface.

1. Telegram bot.

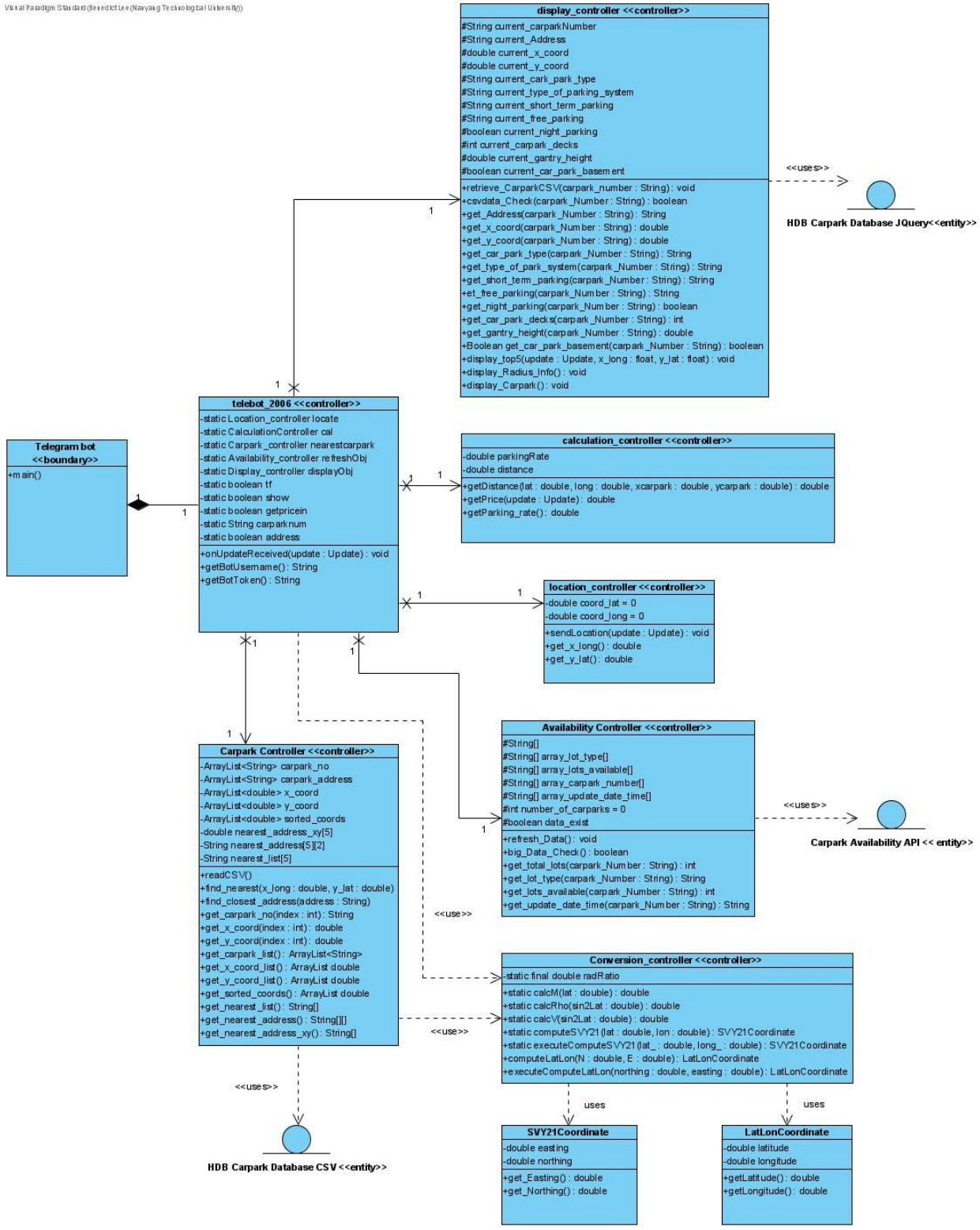
Identified Control Classes (Logic to realize the use case.) - Functions.

1. Telebot.
2. Display controller.
3. Calculation controller.
4. Location controller.
5. Conversion controller.
6. Carpark controller.
7. Availability controller.
8. SVY21 coordinate converter.
9. Latitude Longitude coordinate converter.

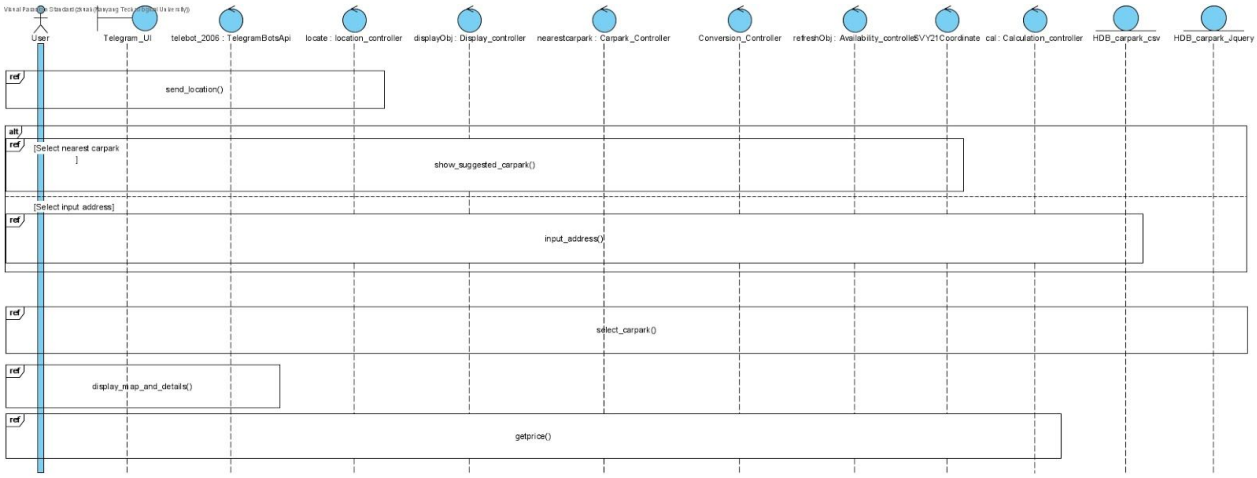
Conceptual Model Diagram



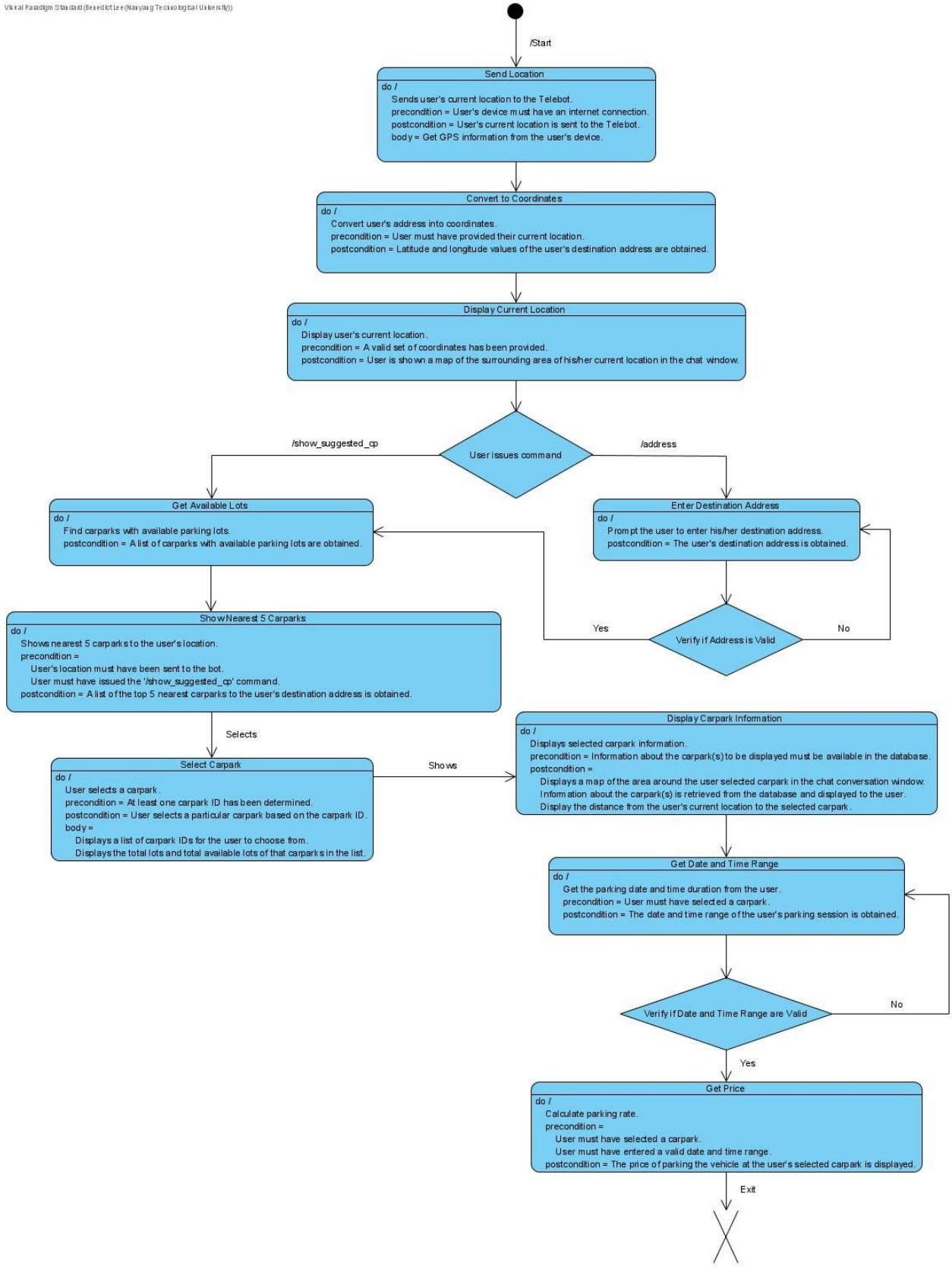
Class Diagram



Sequence Diagram (full)



Initial Dialog Map



Test Cases:

1. **Location services not turned on**
    - Telegram displays an error message stating that the user's location could not be sent.

2. **User selects Address function but enters an invalid address**
    - Bot returns an error message and prompts the user to enter the address again.

3. **User enters a partial address.**
    - Bot returns relevant data to complete the partial address

4. **User enters a full address**
    - Bot returns only that specific address that matches.

5. **Devices's GPS/location setting is disabled.**
    - Error message when sending location.
    - User's device will prompt the user to enable GPS/location.
    - Test: Disable GPS/location services on phone and attempt to send location to the bot.

6. **Date or time entered in a wrong format.**
    - Display an error message and prompt the user to re-enter.

7. **Show routing**
    - Display a prompt to open an external map application such as Google Maps for real time navigation.

More test cases:

No.	Test case	Expected Result	Actual Result
1	/help	Prompt message to guide user	Prompt message to guide user
2	Device's GPS setting is disabled upon pressing /send_location	Error message prompt	Error message prompt
3	"Jshalala" in input address	Error message prompt and prompt user to reenter	Error message prompt and prompt user to reenter
4	"Jurong Jshalala" in input address	Search from partial string of "Jurong"	Search from partial string of "Jurong"
5	Exact address in input address	Shows only 1 exact match of carpark	Shows only 1 exact match of carpark
6	Press into maps provided	Prompt message for user to open map	Prompt message for user to open map
7	/get_price before selecting carpark	Error message prompt	Error message prompt
8	"20/10,20/1e" for datetime format	Error message prompt and prompt user to reenter	Error message prompt and prompt user to reenter
9	"20/10,2e/14" for datetime format	Error message prompt and prompt user to reenter	Error message prompt and prompt user to reenter
10	"20/1e,20/14" for datetime format	Error message prompt and prompt user to reenter	Error message prompt and prompt user to reenter
11	"2e/10,20/14" for datetime format	Error message prompt and prompt user to reenter	Error message prompt and prompt user to reenter
12	"20/10,2e/14" for datetime format	Error message prompt and prompt user to reenter	Error message prompt and prompt user to reenter
13	"20/14,20/10" for datetime format (end date first)	Error message prompt and prompt user to reenter	Error message prompt and prompt user to reenter
14	"32/10,20/14" for datetime format (start date > 31)	Error message prompt and prompt user to reenter	Error message prompt and prompt user to reenter
15	"20/25,20/14" for datetime format (start hour > 24)	Error message prompt and prompt user to reenter	Error message prompt and prompt user to reenter
16	"20/14,32/14" for datetime format (end date> 31)	Error message prompt and prompt user to reenter	Error message prompt and prompt user to reenter
17	"20/14,20/25" for datetime format (end hour>25)	Error message prompt and prompt user to reenter	Error message prompt and prompt user to reenter
18	"randomwords" for datetime format	Error message prompt and prompt user to reenter	Error message prompt and prompt user to reenter