**Use Case Description Template:**

| User Case ID | 01 | | |
| --- | --- | --- | --- |
| Use Case Name |  | | |
| Created By |  | Last Updated by |  |
| Date Created |  | Date Last Updated |  |

| Actor |  |
| --- | --- |
| Description |  |
| Preconditions |  |
| Postconditions |  |
| Priority |  |
| Frequency of Use |  |
| Flow of Events |  |
| Alternative Flows |  |
| Exceptions |  |
| Includes |  |
| Special Requirements |  |
| Assumptions |  |
| Notes and Issues |  |

| User Case ID | 01 | | |
| --- | --- | --- | --- |
| Use Case Name | Search & Filter | | |
| Created By | Chee Han | Last Updated by | Chee Han |
| Date Created | 03/09/2023 | Date Last Updated | 08/09/2023 |

| Actor | User, Google Map API |
| --- | --- |
| Description | This use case involves searching for carparks based on location, distance, and availability. |
| Preconditions | * Location services are enabled * User has network connectivity |
| Postconditions | * Carpark results displayed * Map display updated |
| Priority | High |
| Frequency of Use | High |
| Flow of Events | 1. User opens app 2. System displays map with preferred search radius 3. System initiates search with either user current location or user keyed location 4. Google Map API checks for list of carparks within search radius 5. System displays results on map as markers |
| Alternative Flows | NIL |
| Exceptions | User turned off GPS location services   1. System will prompt user to turn on location services   Invalid Carpark API response   1. System displays “Carpark information currently unavailable” |
| Includes | View Map, Search Radius, Obtain Carpark Details, Obtain Favourites |
| Special Requirements | Results be displayed in a way that users can find available parking in no more than 2 steps |
| Assumptions | User has a stable network connection |
| Notes and Issues | NIL |

| User Case ID | 02 | | |
| --- | --- | --- | --- |
| Use Case Name | Manual Location Input | | |
| Created By | Chee Han | Last Updated by | Michael Santoso |
| Date Created | 03/09/2023 | Date Last Updated | 11/11/2023 |

| Actor | User, Google Maps API |
| --- | --- |
| Description | This use case allows the user to manually input a location |
| Preconditions | * User presses the search bar |
| Postconditions | * User presses enter |
| Priority | Medium |
| Frequency of Use | Medium |
| Flow of Events | 1. User press on search bar 2. User keys in location 3. Google Map API gives list of possible intended locations (Auto-complete) 4. User enters desired location 5. System initiates Search and Filter use case |
| Alternative Flows | User keys in an unknown location   1. System displays “Invalid Address (Not a town in Singapore)”   User keys in unexpected information (symbols and numbers)   1. System displays “Input Errors” |
| Exceptions | Invalid Google Map API response   1. System displays “Map information currently unavailable” |
| Includes | NIL |
| Special Requirements | NIL |
| Assumptions | User has a stable network connection |
| Notes and Issues | NIL |

| User Case ID | 03 | | |
| --- | --- | --- | --- |
| Use Case Name | Filter Preferences | | |
| Created By | Chee Han | Last Updated by | Chee Han |
| Date Created | 03/09/2023 | Date Last Updated | 12/09/2023 |

| Actor | User, Database |
| --- | --- |
| Description | Users can set a preferred search radius (e.g. 1 km, 2 km, 5 km) for nearby carparks and in terms of their carpark availability (e.g. fully occupied, semi-occupied, available). |
| Preconditions | * User presses filter icon on search bar |
| Postconditions | * New filter preferences has been set |
| Priority | High |
| Frequency of Use | Low |
| Flow of Events | 1. System displays results based on previous filters 2. User press filter icon 3. User updates filter preferences 4. System updates filter preferences in the Database 5. System filters current results according to new preferences |
| Alternative Flows | Default Preferences (if preferences are not set yet):   1. From step 1, system sets the carpark availability preferences to show all carparks (fully occupied, semi-occupied, available) and sets the radius of search to be 2 km 2. System filters current results according to the default preferences |
| Exceptions | NIL |
| Includes | NIL |
| Special Requirements | * User must be able to select search radius from the list * User must be able to filter the carpark availability to the preferred result |
| Assumptions | User has a stable network connection |
| Notes and Issues | NIL |

| User Case ID | 04 | | |
| --- | --- | --- | --- |
| Use Case Name | Pinpoint Car Location on Map | | |
| Created By | Chee Han | Last Updated by | Chee Han |
| Date Created | 03/09/2023 | Date Last Updated | 08/09/2023 |

| Actor | User, Database, Google Maps API |
| --- | --- |
| Description | Users can pinpoint their car’s location on the map. Afterwards, users can remove the pinpoint. |
| Preconditions | * User goes to modal bottom sheet and press pinpoint button |
| Postconditions | * Car pinpoint location displayed on map * Car pinpoint location unpinned by user |
| Priority | Low |
| Frequency of Use | Medium |
| Flow of Events | 1. User reaches carpark 2. User presses carpark bubble on map 3. User presses pin parked button on modal bottom sheet 4. System prompts for carpark slot details 5. User enters parking slot details 6. System saves entered information to the Database 7. System marks car location at user current location (assume user has just parked to show exact position in map) 8. System shows slot details when pinpoint in map is clicked |
| Alternative Flows | If user already has a parked location, the modal bottom sheet in the carpark bubble will not contain the pin parked button   1. System will display parked location button on the map interface 2. User will press parked location button on the top AppBar 3. System prompts user to delete parked location 4. User press yes to delete parked location 5. System will delete parked location from Database 6. System will then save the parked history to Database |
| Exceptions | NIL |
| Includes | NIL |
| Special Requirements | Size of pin-point marker should be small and visible enough to indicate the correct carpark when there are multiple carparks in the same vicinity |
| Assumptions | User has a stable network connection |
| Notes and Issues | NIL |

| User Case ID | 05 | | |
| --- | --- | --- | --- |
| Use Case Name | Mark Favorite Carparks | | |
| Created By | Chee Han | Last Updated by | Chee Han |
| Date Created | 08/09/2023 | Date Last Updated | 08/09/2023 |

| Actor | User, Database |
| --- | --- |
| Description | Users can bookmark their handpicked carparks |
| Preconditions | * Users pressed favourite icon (star) on modal bottom sheet of specific carparks |
| Postconditions | * Carparks marked as favourite (if not favourite initially) * Carpars removed as favourite (if favourite initially) |
| Priority | Low |
| Frequency of Use | Low |
| Flow of Events | 1. User press on carpark bubble 2. User press favourite icon (star) on modal bottom sheet 3. System sends request to database 4. Database saves carpark to user’s favourites 5. System updates the favourite icon (star) to be filled in yellow |
| Alternative Flows | If the carpark is already a favourite of the user   1. After step 4, Database remove carpark from user’s favourites 2. System updates the favourite icon (star) to be a blank star |
| Exceptions | NIL |
| Includes | Obtain Favourite |
| Special Requirements | When carparks are favourites, they have to be updated in favourites page (Obtain Favourites Use Case). |
| Assumptions | User has a stable network connection |
| Notes and Issues | NIL |

| User Case ID | 06 | | |
| --- | --- | --- | --- |
| Use Case Name | Obtain Favourite | | |
| Created By | Chee Han | Last Updated by | Michael Santoso |
| Date Created | 08/09/2023 | Date Last Updated | 11/11/2023 |

| Actor | User, Database |
| --- | --- |
| Description | Users can view their favourite carparks |
| Preconditions | * User presses on favourite page in the drawer |
| Postconditions | * Favourites list displayed * Favourites list cleared |
| Priority | Low |
| Frequency of Use | Low |
| Flow of Events | 1. System request user’s favourites from Database 2. Database sends user’s favourites   If user has no favourites   1. System displays “No favourites available”   Else   1. System displays favourites list |
| Alternative Flows | If favourites list is displayed,   1. User can click star icon to delete a particular favourite carpark 2. System request deletion of user’s favourites from database 3. Database removes user’s favourites   If favourites list is displayed,   1. User can click navigate button to direct the map to the specified favourite carpark area   If favourites list is displayed,   1. User can click “Clear Favourites” to delete all favourites 2. System request deletion of all user’s favourites from database 3. Database clears all user’s favourite |
| Exceptions | NIL |
| Includes | Mark Favourite |
| Special Requirements | When a carpark favourites is deleted/cleared out, they have to be updated in the modal bottom sheet on specific carpark bubble (Mark Favorite Carparks Use Case) |
| Assumptions | User has a stable network connection |
| Notes and Issues | NIL |

| User Case ID | 07 | | |
| --- | --- | --- | --- |
| Use Case Name | Obtain History | | |
| Created By | Chee Han | Last Updated by | Parashar Pranav |
| Date Created | 08/09/2023 | Date Last Updated | 10/11/2023 |

| Actor | User, Database |
| --- | --- |
| Description | User can view their parking history after pinpointing car. |
| Preconditions | * User navigates to history page |
| Postconditions | * User’s parking history is displayed * User’s parking history is cleared |
| Priority | Low |
| Frequency of Use | Low |
| Flow of Events | 1. User goes to history page 2. System requests database for user’s parking history 3. Database returns user’s parking history 4. System display user parking time and place |
| Alternative Flows | If user does not have a history yet   1. Database returns empty response 2. System displays “No Parking History”   If user has a history   1. After step 4, user can click on “Clear History” button to remove all history 2. System request deletion of all user’s history from database 3. Database clears all user’s history |
| Exceptions | NIL |
| Includes | Clear History |
| Special Requirements | Database needs to save user history from past 3 months |
| Assumptions | User has a stable network connection |
| Notes and Issues | NIL |

| User Case ID | 08 | | |
| --- | --- | --- | --- |
| Use Case Name | View Map | | |
| Created By | Arjun | Last Updated by | Arjun |
| Date Created | 03/09/2023 | Date Last Updated | 03/09/2023 |

| Actor | User, Google Maps API |
| --- | --- |
| Description | Display of carpark locations and availability of carpark slots within the map. |
| Preconditions | User launches app |
| Postconditions | * Show the nearest carparks to the user’s location * Show the availability of carpark slots in those nearby carparks |
| Priority | High |
| Frequency of Use | High |
| Flow of Events | 1. User launches app 2. System initiates Search & Filter 3. System display carparks bubbles near the user on map |
| Alternative Flows | NIL |
| Exceptions | Invalid Google Map API response   1. System displays loading state until API is working   Invalid Carpark API response   1. System displays “Carpark information currently unavailable”   User GPS not currently activated   1. System prompts the user to activate GPS 2. System displays loading state until GPS is activated |
| Includes | NIL |
| Special Requirements | * Map view must be compatible with a variety of devices and screen sizes * Icons on the map must be visible |
| Assumptions | User has a stable network connection |
| Notes and Issues | NIL |

| User Case ID | 09 | | |
| --- | --- | --- | --- |
| Use Case Name | Obtain Directions | | |
| Created By | Arjun | Last Updated by | Chee Han |
| Date Created | 03/09/2023 | Date Last Updated | 08/09/2023 |

| Actor | Google Maps API |
| --- | --- |
| Description | Uses the google maps API to obtain the directions to the carpark that the user has chosen. |
| Preconditions | User clicks the navigate icon on the modal bottom sheet of specified carpark |
| Postconditions | The map shows the directions from the user’s current location to the carpark that the user has chosen |
| Priority | Medium |
| Frequency of Use | Medium |
| Flow of Events | 1. User clicks on carpark bubble to open modal bottom sheet 2. User press navigate icon 3. System request Google Maps API for polyline direction 4. Google Maps API returns the directions from the user to the specified carpark location 5. System shows directions from current location to the specified carpark on the map interface |
| Alternative Flows | NIL |
| Exceptions | User has turned off GPS location services   1. System will prompt user to “Turn on location services”   Invalid Google Maps API response   1. System displays “Directions currently unavailable” |
| Includes | NIL |
| Special Requirements | NIL |
| Assumptions | User has a stable network connection |
| Notes and Issues | NIL |

| User Case ID | 10 | | |
| --- | --- | --- | --- |
| Use Case Name | Obtain Carpark Details | | |
| Created By | Arjun | Last Updated by | Parashar Pranav |
| Date Created | 03/09/2023 | Date Last Updated | 10/11/2023 |

| Actor | User, Carpark APIs |
| --- | --- |
| Description | The carpark details are retrieved from the carpark APIs in the database and show the details on the bottom modal sheet. |
| Preconditions | User clicks on the carpark bubble |
| Postconditions | The carpark details are shown in bottom modal sheet |
| Priority | High |
| Frequency of Use | High |
| Flow of Events | 1. On initialising the app and thereafter every 60 seconds, system requests the carpark details from the database 2. Database fetches carpark details from API and returns the details to the system 3. System displays carpark bubble on map with different colour to indicate carpark availability 4. User clicks carpark bubble to show modal bottom sheet 5. System displays the specific carpark details (Area, Development, Type, Distance, Carpark ID, Slots available) |
| Alternative Flows | NIL |
| Exceptions | Invalid Carpark API response   1. System displays “Carpark information currently unavailable” |
| Includes | NIL |
| Special Requirements | To be initiated every 60 seconds |
| Assumptions | User has a stable network connection |
| Notes and Issues | NIL |

| User Case ID | 11 | | |
| --- | --- | --- | --- |
| Use Case Name | Current Location | | |
| Created By | Pranav | Last Updated by | Pranav |
| Date Created | 16/09/2023 | Date Last Updated | 16/09/2023 |

| Actor | User |
| --- | --- |
| Description | On clicking the current location button, the map view is reset to show the current location of the user. |
| Preconditions | The user has moved around the map to view different location, and click the current location button. |
| Postconditions | The map shows the current location of the user. |
| Priority | High |
| Frequency of Use | Medium |
| Flow of Events | 1. User clicks “Move to current location” button 2. System moves the camera back to the user current location |
| Alternative Flows | NIL |
| Exceptions | NIL |
| Includes | View Map |
| Special Requirements | NIL |
| Assumptions | User has stable network connection and has location services turned on. |
| Notes and Issues | NIL |

| User Case ID | 12 | | |
| --- | --- | --- | --- |
| Use Case Name | Clear History | | |
| Created By | Pranav Parshar | Last Updated by | Pranav Parashar |
| Date Created | 16/09/2023 | Date Last Updated | 16/09/2023 |

| Actor | User, Database |
| --- | --- |
| Description | The app clears history data of more than 3 months or the user can manually choose to do the same before that. |
| Preconditions | History is stored in the database. |
| Postconditions | History is cleared. |
| Priority | Medium |
| Frequency of Use | Low |
| Flow of Events | 1. User navigates to history page in the drawer 2. User clicks the “Clear History” button. 3. System request deletion of all user’s history from database 4. Database clears all user’s history 5. System displays “No Parking History” |
| Alternative Flows | If history is of more than 3 months,   1. Database clears all parking history 2. System displays “No Parking History:   If there is no history   1. System displays “No Parking History” in the history page |
| Exceptions | NIL |
| Includes | NIL |
| Special Requirements | NIL |
| Assumptions | User has a stable internet connection. |
| Notes and Issues | NIL |

| User Case ID | 13 | | |
| --- | --- | --- | --- |
| Use Case Name | CarPark Database Update | | |
| Created By | Parashar Pranav | Last Updated by | Parashar Pranav |
| Date Created | 10/11/2023 | Date Last Updated | 10/11/2023 |

| Actor | Carpark API, Database |
| --- | --- |
| Description | Update the local carpark database using carpark api |
| Preconditions | CarPark table is created |
| Postconditions | CarPark table is populated using appropriate values |
| Priority | High |
| Frequency of Use | High |
| Flow of Events | 1. Evert 60 second, database gets carpark data from carpark API 2. In the database, check if any changes in the data, and update each record, if the new data has some car parks missing, set the available slots in those car parks to 0. |
| Alternative Flows | NIL |
| Exceptions | NIL |
| Includes | NIL |
| Special Requirements | Executed every 60 seconds |
| Assumptions | User has a stable internet connection. |
| Notes and Issues | NIL |