**Project Mission Statement**

The rm-rf/\* team will develop a mobile app that benefits drivers from reducing the time taken to locate a nearby car park and also easy access to parking space by utilizing the data provided by SG government such as HDB carpark information and car park availability API. This project will be considered complete after the app is fully tested by our developers.

**Functional Requirements**

1. The application must allow the user to sign up for an account with email or social network applications such as Facebook and Google account.

1.1. Upon signing up, the application must allow users to login and logout.

1.2. Once login, the application must keep the login status of the current user.

2. The application must request the permission of the user's location from the mobile phone system.

2.1. If permission is denied by the user, the application must allow the user to enter an address manually.

2.2. If permission is granted, the application shall show a map consisting of the nearest carpark to the user.

2.2.1. The user must be able to select the radius of the region to show all the carparks based on his/her current location.

3. The application must allow users to search for the car parks near the selected location or current location:

3.1. The application must allow users to search car parks by full address name.

3.1.1. The application must automatically display the nearest search result under the drop bar.

3.2. The application must allow users to search the nearby car parks by postal code

3.3. The application must allow users to search the nearby car parks of the selected/current location by type (eg. Surface, Multistorey and basement)

4. The application must give users an internal locus of the control of the carpark’s icon

4.1. The user can drag the map around Singapore to see the car parks.

4.2. The user can zoom in and zoom out of the map to see the car parks.

5. The application must allow the users to get the sign and status of the carpark’s icon

5.1. The icon must display the current available slots of the car park

5.2. The car park icon must display the current car park status by colours.

5.3. The car park icons’ status must be updated by every 30 seconds from the APIcaller.

6. The application must display more detailed information of a car park when the user taps the carpark’s icon

6.1. The application must display the car park full address name.

6.2. The application must display the number of available parking lots of the selected car park.

6.3. The application must display the total capacity of the selected car park.

6.4. The application must display the available parking hours of the car park.

6.5. The application must display the parking fee of the car parks based on different time slots.

6.6. The application must display the type of the car park (Multistorey, Basement, Surface).

6.7. The application must display the review of the current carpark.

7. The application must help user to navigate to the destination car park

7.1. The application must use google map service for help to navigate to the carpark.

8. The application must allow the user to add the review of the carpark with rating.

8.1. The review must include the cost, convenience and security.

8.2. The application must accumulate the total ratings and calculate the weighted average from the total users.

8.3. The rating is from one star to five stars.

**Non-Functional Requirements**

1. The application must be compatible with phones running across various platforms such as Android and IOS.
2. The application must be able to maintain the log in status even if the user exits the app.
3. The application must accumulate the total ratings and calculate the weighted average from the total users
4. The application must be fully operated with 24x7 daily.
5. The application must display park data details within 2s when a user taps into any car park.
   1. If the data fetching fails, the application must retry up to 3 times before displaying an informative error message.
6. The application must record down the daily active user on the cloud database.
7. The application must design in OOP language with strict MVC design structure.