

Use Case ID:	1		
Use Case Name:	Loading The Website		
Created By:		Last Updated By:	
Date Created:	03/02/2020	Date Last Updated:	

Actor:	User
Description:	Application is initiated
Preconditions:	<ol style="list-style-type: none"> 1. The user must have access to the internet. 2. The user must allow the browser to access their location
Postconditions:	<ol style="list-style-type: none"> 1. Brings user to the login page
Priority:	High
Frequency of Use:	Twice a day
Flow of Events:	<ol style="list-style-type: none"> 1. User type the website on the browser's search bar. 2. Application launched and proceeds to landing page
Alternative Flows:	-
Exceptions:	EX1: User fails to launch application <ol style="list-style-type: none"> 1. Webpage displays that an error has occurred.
Includes:	-
Special Requirements:	-
Assumptions:	-
Notes and Issues:	-

Use Case ID:	2		
Use Case Name:	Login		
Created By:		Last Updated By:	
Date Created:		Date Last Updated:	

Actor:	User, Database, APIs
Description:	Executes login procedure to connect the account with the database
Preconditions:	<ol style="list-style-type: none"> 1. Application must be running. 2. The user has an email account registered in the database.
Postconditions:	<ol style="list-style-type: none"> 1. User is logged in and moved to the "Navigation" page. 2. Favourite locations are retrieved. 3. User profile is retrieved.
Priority:	High
Frequency of Use:	Once every instance of loading the application
Flow of Events:	<ol style="list-style-type: none"> 1. When the application is loaded, User will be directed to the login page. 2. User will enter the email account and password. 3. When login information is validated, the account will load the main page of the application, the "Navigation" page.
Alternative Flows:	<p>AF1: User has already logged in when starting the application and information has been cached in browser information.</p> <ol style="list-style-type: none"> 1. User will be directed to the "Navigation" page. <p>AF2: Account does not exist in the database.</p> <ol style="list-style-type: none"> 1. Alert the user of "Incorrect Username/Password" at the top of the login page 2. Application returns to step 2. <p>AF3: User enters the wrong data input.</p> <ol style="list-style-type: none"> 1. Alert the user of "Incorrect Username/Password" at the top of the login page 2. Application returns to step 2.
Exceptions:	<p>EX1: Failure to retrieve information from the Database.</p> <ol style="list-style-type: none"> 1. Webpage displays that an error has occurred.

Includes:	NIL
Special Requirements:	The application must be able to retrieve login information and favourites within 5 seconds.
Assumptions:	NIL
Notes and Issues:	NIL

Use Case ID:	3		
Use Case Name:	Sign Up		
Created By:		Last Updated By:	
Date Created:		Date Last Updated:	

Actor:	User, Database, APIs
Description:	User wants to create an account.
Preconditions:	<ol style="list-style-type: none"> 1. Application must be running. 2. The user has an email account registered in the database. 3. User clicked "Sign Up" in the Login Page.
Postconditions:	<ol style="list-style-type: none"> 1. User is logged in and moved to the "Navigation" page. 2. User profile is saved in the database.
Priority:	High
Frequency of Use:	The first time the user uses the application.
Flow of Events:	<ol style="list-style-type: none"> 1. User shall enter the email account, password and email. 2. Application verifies if data inputs are in the correct format. 3. User click "Sign Up" 4. Application store user data successfully into the Database. 5. Application shall automatically log in and bring the user to the navigation page.
Alternative Flows:	<p>AF1: Email addressed exist in the database.</p> <ol style="list-style-type: none"> 1. After step 3, the application will redirect the user back to the "Sign Up" page and display the message "Email Already Used" at the top. 2. Application returns to step 1. <p>AF2: User key in invalid inputs.</p> <ol style="list-style-type: none"> 1. After step 3, the application brings scroll to the particular input field with incorrect input format, with a message "Enter the Correct Format" above the input field. 2. User input data in the right format. 3. Application returns to step 3
Exceptions:	<p>EX1: Failure to store data to the Database.</p> <ol style="list-style-type: none"> 1. Webpage displays that an error has occurred.

	EX2: User wants to log in into his existing account: <ol style="list-style-type: none"> 1. Click "Login". 2. Application redirects user to "Login" page.
Includes:	-
Special Requirements:	-
Assumptions:	-
Notes and Issues:	-

Use Case ID:	4		
Use Case Name:	Navigation		
Created By:		Last Updated By:	
Date Created:	03-02-2020	Date Last Updated:	

Actor:	User, Database and API
Description:	<p>User will be directed to the “Navigation” page, where they can set their desired start and end location.</p> <p>Concurrently, the car park prices and availability API will be retrieved from their respective APIs to guide the user to the destination based on the real-time car park price and availability.</p>
Preconditions:	1. User must be logged in.
Postconditions:	1. Destination options consisting of different carparks are generated
Priority:	High
Frequency of Use:	Twice a day
Flow of Events:	<ol style="list-style-type: none"> 1. Application displays a popup with the message “The website would like to use your current location.” with 2 options which are “Don’t Allow” and “Allow”. 2. User allows the browser to use the current location. 3. User enters Starting Point. (Option of “Your Location” will be displayed when clicked on search bar) 4. User enters destination location. 5. Application computes carparks around the destination, with their parking prices and distance from destination. 6. Application displays options of carparks and their respective parking price and distance from destination. 7. User chooses one of the carparks. 8. Application prompts the user to open google maps. 9. Google map is opened, with the starting point and destination (carpark) keyed in, ready to navigate the user.
Alternative Flows:	AF1: User clicked “Don’t Allow” after step 1 and manually enter

	<p>starting point.</p> <ol style="list-style-type: none"> 1. Application proceeds to step 4. <p>AF2: User clicked “Don’t Allow” after step 1 and enter the option of “Your Location” in step 3 when entering the starting point.</p> <ol style="list-style-type: none"> 1. Application returns to step 1.
Exceptions:	<p>EX1: API call fails and unable to retrieve the current location.</p> <ol style="list-style-type: none"> 1. Application display alert “Unable to retrieve location” 2. Application returns to step 3 and the user has to manually enter the location. <p>EX2: User did not enable location service for their browser and the user clicked “Allow” at step 1.</p> <ol style="list-style-type: none"> 1. Application display alert “ Please enable location service for your browser” with the option “Dismiss”. 2. Application proceeds to step 3 <p>EX3: User do not have Google Map in his/her phone</p> <ol style="list-style-type: none"> 1. After step 8, it redirects the user to the App Store to download Google Map
Includes:	-
Special Requirements:	<ol style="list-style-type: none"> 1. User must enable location service for their browser 2. User must have downloaded Google Map
Assumptions:	-
Notes and Issues:	-

Use Case ID:	5		
Use Case Name:	Store Favourite Locations		
Created By:		Last Updated By:	
Date Created:	03/02/20	Date Last Updated:	04/02/20

Actor:	User, Database, APIs
Description:	User adds a location to a list of "Favourites" locations.
Preconditions:	<ol style="list-style-type: none"> 1. User is logged in. 2. User's favourite locations are retrieved.
Postconditions:	<ol style="list-style-type: none"> 1. Current location successfully added to favourites list (or user declines to add the location)
Priority:	Medium
Frequency of Use:	Every time a location is searched for in the navigation page.
Flow of Events:	<ol style="list-style-type: none"> 1. User search for a location on the navigation page, either in the starting point field or destination field. 2. The location will appear with a white star icon beside the location. 3. User clicks the star icon. 4. Popup appears that prompts the user to enter the name of the location to be saved. 5. User click "Done" and popup closes. 6. Location is saved to the user's Favourites and the white star icon will now be yellow.
Alternative Flows:	-
Exceptions:	EX1: Unable to update "Favourites" from Database <ol style="list-style-type: none"> 1. Application displays alert "Unable to save location to Favourites" 2. Application returns to step 1.
Includes:	-
Special Requirements:	The application shall update favourites within 5 seconds.
Assumptions:	-
Notes and Issues:	-

Use Case ID:	6		
Use Case Name:	User removes a location to the list of “Favourites” locations.		
Created By:		Last Updated By:	
Date Created:	04/02/20	Date Last Updated:	04/02/20

Actor:	User, Database, APIs
Description:	User will be prompted to store their current destination.
Preconditions:	<ol style="list-style-type: none"> 1. User is logged in. 2. User’s favourite locations are retrieved.
Postconditions:	<ol style="list-style-type: none"> 1. Current location successfully remove to favourites list (or user declines to add the location)
Priority:	Medium
Frequency of Use:	Every time a location is searched for in the navigation page.
Flow of Events:	<ol style="list-style-type: none"> 1. User search for a location on the navigation page, either in the starting point field or destination field. 2. User clicks the list of favourite locations. 3. The application will display a list of locations with yellow stars beside each location. 4. User clicks on the yellow star for the location the user wants to remove. 5. A popup appears and a message “Are you sure you want to remove this location from Favourites?” with options “Confirm” and “Cancel”. 6. User clicks “Confirm”. 7. The location is removed from the list of favourites and the yellow star will now be white.
Alternative Flows:	AF1: User clicks the wrong star. <ol style="list-style-type: none"> 1. User click “Cancel” in step 5. 2. The application returns to step 3.
Exceptions:	EX1: Unable to update “Favourites” to Database <ol style="list-style-type: none"> 1. .Application display alert “Unable to remove location from Favourites” 3. Application returns to step 1.
Includes:	-
Special Requirements:	The application shall update favourites within 5 seconds.

Assumptions:	-
Notes and Issues:	-