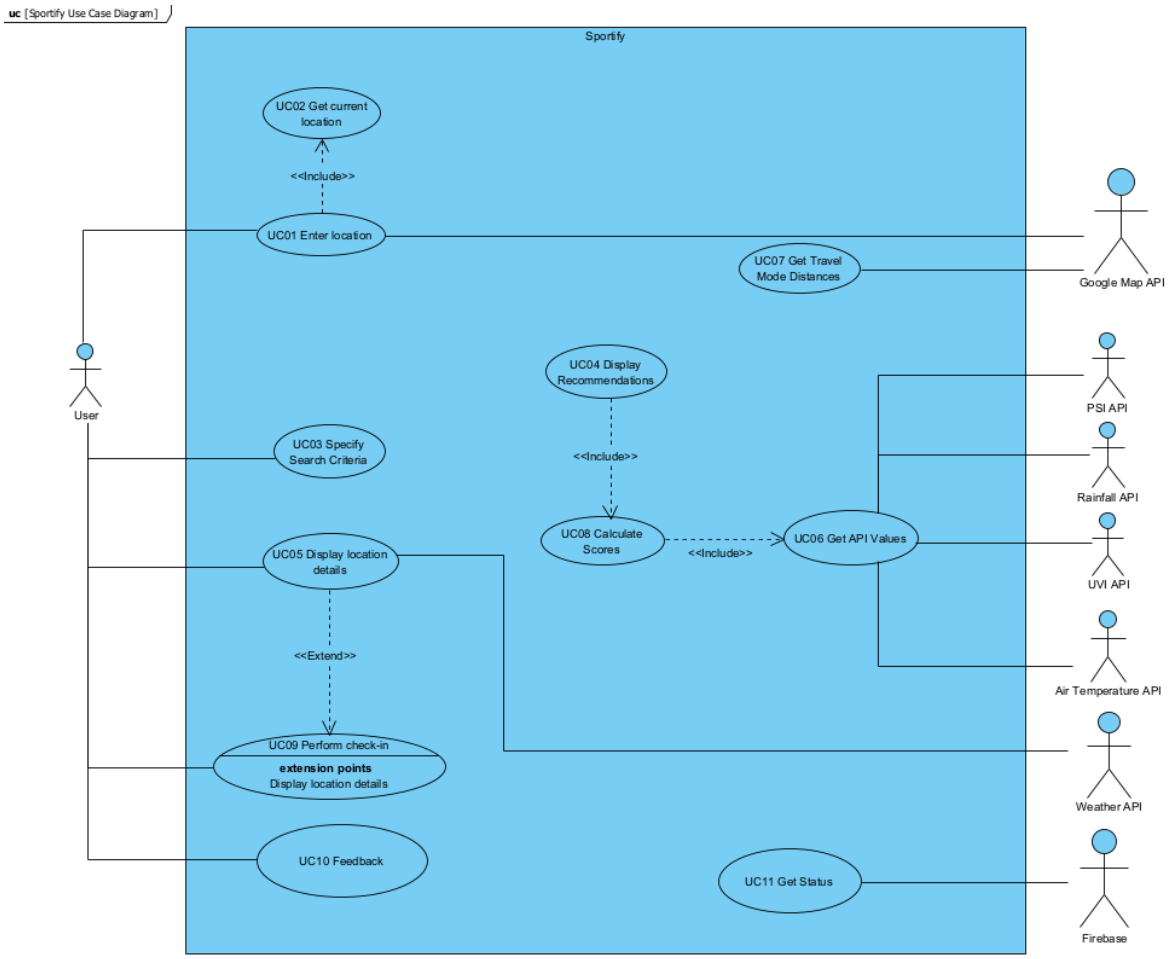
Use Case Diagram



Use Case Description

Use Case ID:	UC01		
Use Case Name:	Enter location		
Created By:	Ding Ren	Last Updated By:	Ding Ren
Date Created:	8/2/24	Date Last Updated:	8/2/24

Actor:	User & Google Map API
Description:	User can set specific location used as departure location and basis of search radius
Preconditions:	1. User has not specified a location
Postconditions:	1. Map reflects marker on specified location
Priority:	
Frequency of Use:	1. Everytime a new search for a new location is made.
Flow of Events:	1. User selects search bar 2. User enters name of location 3. User can select location from a dropdown list of recommendations obtained from Google Map API.
Alternative Flows:	AF-S3: 1. User can select the current location button to use the included use case “Get current location” to use the user's current location by GPS as specified user’s location.
Exceptions:	EX1: Location specified not within Singapore 1. Sportify return empty location dropdown list
Includes:	UC02
Special Requirements:	API response time should be 0.1s which can be counted as immediate response where users would not feel any interruption.
Assumptions:	
Notes and Issues:	

Use Case ID:	UC02		
Use Case Name:	Get current location		
Created By:	Ding Ren	Last Updated By:	Ding Ren
Date Created:	8/2/24	Date Last Updated:	8/2/24

Actor:	
Description:	Obtains the current location of the user to use as the user's specified location.
Preconditions:	
Postconditions:	1. Sportify now selects user’s current location as departure location. 2. Sportify marks the current location of the user on the map.
Priority:	
Frequency of Use:	1. Everytime user wishes to use their current location as departure location.
Flow of Events:	3. Sportify prompts users to allow use of the device's GPS to know current location. 4. Sportify returns the current location of user.
Alternative Flows:	AF-S1: User rejects use of GPS 1. “Current Location” button no longer works. 2. User can only select address through the search bar.
Exceptions:	
Includes:	
Special Requirements:	User is in a location where GPS signal is available.
Assumptions:	
Notes and Issues:	

Use Case ID:	UC03		
Use Case Name:	Specify Search Criteria		
Created By:	Randall	Last Updated By:	Randall
Date Created:	8/2/24	Date Last Updated:	8/2/24

Actor:	User
Description:	User specifies the search radius and preferred modes of transport.
Preconditions:	1. User has specified a location.
Postconditions:	
Priority:	
Frequency of Use:	1. Everytime user wishes to change the search radius and modes of transport.
Flow of Events:	2. Upon entering location or getting current location, user can select the radius button to adjust the search radius. 3. Users can shift the slider to desired radius. 4. Upon selecting the search radius button, the user can select a preferred mode of travel. 5. Users must select at least one mode of travel before proceeding to submit a query.
Alternative Flows:	
Exceptions:	EX1: User didn't specify search criteria 1. Sportify prompt user to set search criteria. EX2: User didn't specify mode of transport 1. Sportify prompts user to select mode of transport. EX3: User didn't specify location before specifying search criteria 1. Sportify prompts user to specify a departure location.
Includes:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	UC04		
Use Case Name:	Display Recommendations		
Created By:	Steven	Last Updated By:	Steven
Date Created:	8/2/24	Date Last Updated:	8/2/24

Actor:	
Description:	Displays recommended sports locations ranked according to their overall scores.
Preconditions:	
Postconditions:	
Priority:	
Frequency of Use:	1. Everytime new query is made.
Flow of Events:	1. Sportify uses the included use case “Calculate Scores” to get overall score for all sports facilities within the search radius. 2. Sportify ranks them in descending order. 3. Sportify displays them according to their ranks.Spotify shows all available locations in the search radius including: a. Picture of location b. Name of sport location c. Address of sport location d. Type of activity available in sport location e. Distance of sport location from user specified location f. Location fees/charges
Alternative Flows:	
Exceptions:	EX1: No locations found in search radius 1. Sportify prompts user with message “No location!” at home page.
Includes:	UC08
Special Requirements:	Requires data from other use cases.
Assumptions:	All previous use cases are correct and provide accurate information.
Notes and Issues:	

Use Case ID:	UC05		
Use Case Name:	Display location details		
Created By:	Randall	Last Updated By:	Randall
Date Created:	8/2/24	Date Last Updated:	8/2/24

Actor:	User
Description:	User specifies the search radius.
Preconditions:	1. User has submitted a search 2. Sportify displays recommendations. 3. User has selected the “Find Out More” button for a sports location from the result page.
Postconditions:	
Priority:	
Frequency of Use:	1. Everytime user selects “Find out more” button of a sports location
Flow of Events:	1. Sportify gets weather condition of sport location from Weather API. 2. User presses find out more button of sport location to show: a. Playing count b. Check-In count c. Pre-check-in count d. Air temperature e. Weather condition f. Rain g. PSI h. UVI
Alternative Flows:	
Exceptions:	
Includes:	
Special Requirements:	API response time should be 0.1s which can be counted as immediate response where users would not feel any interruption.
Assumptions:	
Notes and Issues:	

Use Case ID:	UC06		
Use Case Name:	Get API values		
Created By:	Ding Ren	Last Updated By:	Ding Ren
Date Created:	8/2/24	Date Last Updated:	8/2/24

Actor:	PSI API
Description:	Obtains and collates PSI values for the different areas of Singapore.
Preconditions:	1. User has specified location. 2. User has specified search-radius. 3. User has specified modes of travel. 4. User has selected search button to show recommendations result.
Postconditions:	
Priority:	
Frequency of Use:	1. Everytime a new search is made.
Flow of Events:	1. Sportify queries the PSI API, Air Temperature API, Weather API, UVI API and Rainfall API to get their respective values from areas across Singapore. 2. Sportify finds out the area where values are taken from at which the location is closest to. 3. Sportify returns the values from the identified area to take reference from for location.
Alternative Flows:	
Exceptions:	EX1: If any of the APIs does not return a result 1. Replace any of the weather quantities with NaN if that API does not return a result.
Includes:	
Special Requirements:	API response time should be 0.1s which can be counted as immediate response where users would not feel any interruption.
Assumptions:	
Notes and Issues:	

Use Case ID:	UC07		
Use Case Name:	Get Travel Mode		
Created By:	Ding Ren	Last Updated By:	Ding Ren
Date Created:	8/2/24	Date Last Updated:	8/2/24

Actor:	Google Map API
Description:	Obtain and collate the travel distance to locations within specified search radius from specified location based on the user selected mode of travel.
Preconditions:	1. User has specified location. 2. User has specified search-radius. 3. User has specified mode of travel. 4. User has selected search button to show recommendations result.
Postconditions:	
Priority:	
Frequency of Use:	1. Everytime a new search is made.
Flow of Events:	1. Sportify queries the Google Map API to get the most optimized routes to location based on the user's selected mode of travel. 2. The Google Map API returns the routes of location in search radius from user’s specified location.
Alternative Flows:	
Exceptions:	EX1: If any of the APIs does not return a result. 1. Replace any distance not returned with NaN.
Includes:	
Special Requirements:	API response time should be 0.1s which can be counted as immediate response where users would not feel any interruption.
Assumptions:	
Notes and Issues:	

Use Case ID:	UC08		
Use Case Name:	Calculate Scores		
Created By:	Ding Ren	Last Updated By:	Ding Ren
Date Created:	8/2/24	Date Last Updated:	8/2/24

Actor:	
Description:	Calculates score for every sport location located within the specified search radius for Sportify to rank.
Preconditions:	1. Sport locations within search radius have been filtered. 2. User has specified a departure location, search radius and mode of transport. 3. Use has selected the search button to show recommendation results.
Postconditions:	
Priority:	
Frequency of Use:	1. Everytime a new search is made.
Flow of Events:	1. Sportify uses the included use case “Get API Values” to get the PSI, rainfall, UVI, air temperature and distance by specified travel mode of a location within the search radius. 2. Sportify uses the retrieved values to generate an overall score out of 100 for each sport location within the search radius.
Alternative Flows:	
Exceptions:	
Includes:	UC06
Special Requirements:	API response time should be 0.1s which can be counted as immediate response where users would not feel any interruption.
Assumptions:	
Notes and Issues:	

Use Case ID:	UC09		
Use Case Name:	Perform check-in		
Created By:	Ding Ren	Last Updated By:	Ding Ren
Date Created:	19/2/24	Date Last Updated:	19/2/24

Actor:	User
Description:	Allows user to inform others that they have arrived and is currently at the sport facility. This adds on to a live-count of the number of people present at the sport facility but has yet started playing or has yet to find a playing group or has yet to find a playing partner.
Preconditions:	1. Status of user must be “Pre-checked-in” 2. User not in any of the following statuses: “Default”, “Checked-in”, “Playing”
Postconditions:	1. Changes user status to “Checked-in” 2. Increases the counter for Check-in at that sport facility 3. Decreases the counter for Pre-check-in at that sport facility
Priority:	
Frequency of Use:	Everytime the “Check-in” button on the subpage of a sport facility is pressed
Flow of Events:	1. User presses “Pre Check-in” button 2. Sportify checks if the status of user is “Pre-checked-in” 3. If no, Sportify changes user status to “Pre checked-in” 4. Sportify increments the counter for pre-check-in at that sport facility by 1 5. User presses “Check-in” button 6. Sportify checks if the status of user is “Pre Checked-in” 1. If yes, Sportify changes user status to “Checked-in” 2. Sportify decrements counter for pre-check-in at sport location by 1 and increment counter for check-in at sport location by 1 3. User presses “Playing” button 4. Spotify changes user status to “Playing”. 5. Sportify starts timer of 180 minutes for user. 6. User can press “Check-out” button manually while status is checked-in to change user status to checked-out

	7. Sportify decrements counter for check-in by 1
Alternative Flows:	AF-S2: User status is not “pre-checked-in” 1. Sportify displays error message “ Please perform pre-check-in first!” 1. Sportify prompts user to restart app. AF-S6: User presses “Check-out” button 1. Sportify changes user status to “Default” 2. Sportify decrements the counter for “Checked-in” at that sport facility by 1 AF-S6: Timer of 180 minutes is up 1. Sporify displays error message “ Check-in has expired! Playing time is up!” 2. Sportify changes user status to “Default” 3. Sportify decrements the counter for “Checked-in” at that sport facility by 1
Exceptions:	
Includes:	
Special Requirements:	
Assumptions:	All previous use cases are correct and provide accurate information.
Notes and Issues:	

Use Case ID:	UC10		
Use Case Name:	Feedback		
Created By:	Randall	Last Updated By:	Randall
Date Created:	8/2/24	Date Last Updated:	8/2/24

Actor:	User
Description:	User is able to provide feedback on Sportify.
Preconditions:	
Postconditions:	
Priority:	
Frequency of Use:	1. Everytime user wishes to leave a feedback.
Flow of Events:	1. User clicks on “Help” button on top navigation bar. 2. User keys in email. 3. User keys in content of feedback. 4. User clicks on “Send Feedback” 5. User is prompted that feedback has been successfully sent. 6. Feedback is sent to Sportify’s external Firebase server where it can be read.
Alternative Flows:	
Exceptions:	EX1: User leaves either email or feedback content blank upon sending feedback. 1. Sportify prevents sending of feedback. 2. Sportify prompts user to fill up both contents.
Includes:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	UC11		
Use Case Name:	Get Status		
Created By:	Randall	Last Updated By:	Randall
Date Created:	8/2/24	Date Last Updated:	8/2/24

Actor:	Firebase
Description:	Sportify fetches feedback and user's status from Firebase server.
Preconditions:	1. User has submitted a search. 2. Sportify displays recommendations. 3. User has selected the "Find Out More" button for a sports location from the result page. 4. User has clicked on "Pre Check-in", "Pre Check-out", "Check-in", "Check-out" and "Playing" buttons and trigger a change in user status.
Postconditions:	
Priority:	
Frequency of Use:	1. Everytime user has triggered a change in user status.
Flow of Events:	1. Sportify fetches user's status from external Firebase server. 2. Sportify changes user status and sends new status back to external Firebase server.
Alternative Flows:	
Exceptions:	
Includes:	
Special Requirements:	
Assumptions:	
Notes and Issues:	