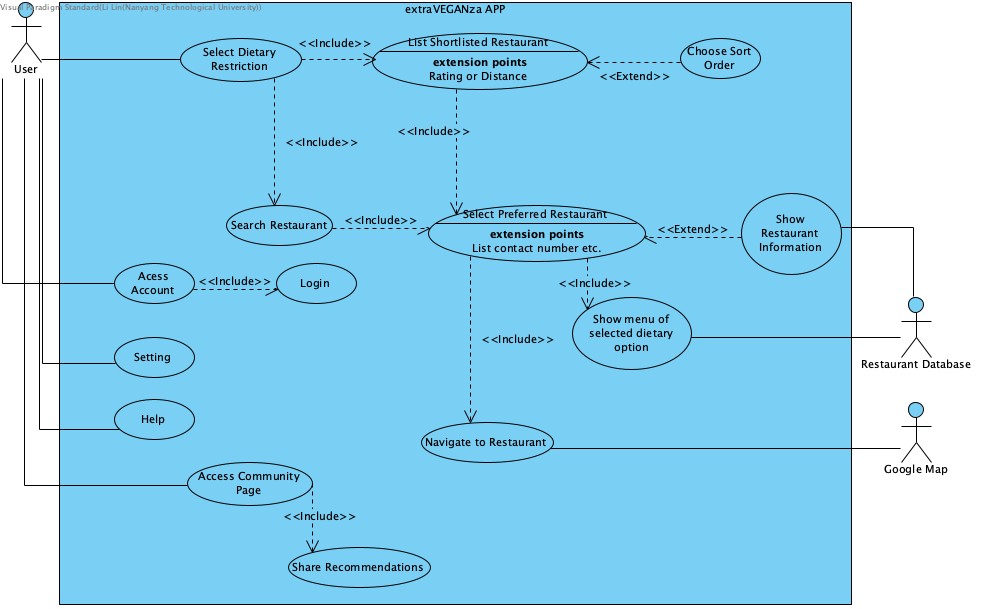
# Use Case Diagram

****

# Use Case Description

1. **Select Dietary Restrictions**
   1. **Description:**

This use case allows users to select their own dietary restrictions.

* 1. **Anticipating Actors:** 
     1. User (Initiating actor)
  2. **Flow of events:**
     1. The user selects one or more preferred dietary options available, located on the left sidebar of the application.
     2. The app displays a list of shortlisted restaurants based on the selected dietary options provided by the user.

1. **List Shortlisted Restaurants**
   1. **Description:**

This use case allows users to obtain a list of restaurants based on predetermined criteria.

* 1. **Anticipating Actors:** 
     1. User
  2. **Flow of events:** 
     1. The application searches the application’s database for restaurants that matched the predetermined criteria inputted by the user, such as dietary restrictions, search query, and filters.
     2. The application displays a list of shortlisted restaurants based on the predetermined criteria inputted by the user.

1. **Choose Sort Order**
   1. **Description:**

This use case allows users to choose the sorting order of the listing of restaurants.

* 1. **Anticipating Actors**:
     1. User
  2. **Flow of events:**
     1. The user chooses sorting order, either by alphabetical order, by distance, or by rating.
     2. The application reorders the list of restaurants according to the user’s sorting choice.

1. **Search Restaurant**
   1. **Description:**

This use case allows users to search a restaurant from the restaurant database.

* 1. **Anticipating Actors:** 
     1. User
     2. Restaurant Database
  2. **Flow of events:**
     1. The user clicks on the search box.
     2. The user enters the name of the restaurant.
     3. The APP searches the name of the restaurant in the database.
     4. The APP shows the result.

1. **Select Preferred Restaurants**
   1. **Description:**

This use case allows users to select a restaurant from the listing of restaurants.

* 1. **Anticipating Actors:** 
     1. User
  2. **Flow of events:**
     1. The user clicks on the preferred restaurant from the list of restaurants displayed on the application.
     2. The user confirms to view the restaurant.
  3. **Alternative Flows:**
     1. If the user likes the restaurant, she can bookmark it by clicking the “❤️”.

1. **Show Restaurant Information**
   1. **Description:**

This use case allows users to view the information of a restaurant after being selected.

* 1. **Anticipating Actors:** 
     1. User
     2. Restaurant Database
  2. **Flow of events:**
     1. The application retrieves the restaurant’s information, including location, contact number and operation hours.
     2. The application displays the information retrieved from the database to the user.

1. **Show Restaurant Menu**
   1. **Description:**

This use case allows users to view a restaurant’s menu with only selected dietary options.

* 1. **Anticipating Actors:** 
     1. User
     2. Restaurant Database
  2. **Flow of events**:
     1. The application shows the restaurant’s menu based on the user’s selected dietary options.
     2. The application shows other users’ ratings and recommendations.
     3. The user clicks on the food option.
     4. The application shows the food’s information, including ingredients, calories and allergies.

1. **Navigate to Restaurant**
   1. **Description:**

This use case allows users to navigate to the restaurant selected via Google Map.

* 1. **Anticipating Actors:** 
     1. User
     2. Google Map
  2. **Flow of events:**
     1. The user chooses the restaurant they want to go to.
     2. The application will show a navigation guide to the restaurant via an in-frame Google Map view.
     3. The user can further change navigation settings through Google Map.

1. **Access Community Page**
   1. **Description:**

This use case allows users to access the application’s community page.

* 1. **Anticipating Actors:** 
     1. User (Initiating actor)
  2. **Flow of events:**
     1. The user must be logged into their account to be granted access to the community page.
     2. The user views other users’ recommendations and ratings.

1. **Share Recommendations**
   1. **Description:**

This use case allows users to share their recommendation of food / restaurants to the community page.

* 1. **Anticipating Actors:** 
     1. User
  2. **Flow of events:**
     1. The user must be logged into their account to proceed, otherwise recommendation cannot be proceeded.
     2. The user selects food or restaurant she wants to recommend.
     3. The user writes her recommendation in the text box shown.
     4. The user sends out the recommendation.

1. **Login**
   1. **Description:**

This use case allows users to login into / signup for their application account.

* 1. **Anticipating Actors:**
     1. User
  2. **Flow of events:**
     1. The user will be prompted to enter their username and password for their account.
     2. If authentication of the account is successful, the user will be able to login to their account.
     3. The user will be guided back to the home page.
  3. **Alternate flow:**
     1. (After step i) If the authentication of the account is unsuccessful, the user will be prompted for their username and password again (repeat from step 1).

1. **Access Settings**
   1. **Description:**

This use case allows users to manage the settings of the application.

* 1. **Anticipating Actors:** 
     1. User
  2. **Flow of events:** 
     1. The user is shown a list of settings that can be managed.
     2. The user chooses which settings to be changed, and changes it according to the user’s preferences.

1. **Access Help**
   1. **Description:**

This use case allows users to access the help page of the application.

* 1. **Anticipating Actors:**
     1. User
  2. **Flow of events:**
     1. The user is shown a help page for the application.

1. **Access Account**
   1. **Description:**

This use case allows users to manage their account’s settings.

* 1. **Anticipating Actors:**
     1. User
  2. **Flow of events:**
     1. The user is shown a list of account’s settings that can be managed.
     2. The user chooses which settings to be changed, and changes it according to the user’s preferences.