**HEALTH:**

**- find buddies func(scan people within a radius that has the app to chat)**

**- plan time slots func in the app (to see how many people also going around that time?)**

**- comments/rate**

**- consider weather and advice if outdoor sports is good?**

**(.kml)**

**https://data.gov.sg/dataset/healthier-caterers**

**https://data.gov.sg/dataset/healthier-eateries**

**Vigorous: https://data.gov.sg/dataset/gymssg**

**Light: https://data.gov.sg/dataset/parkssg**

**https://data.gov.sg/dataset/sportsfieldssg**

**https://data.gov.sg/dataset/wateractivitiessg**

**(extras)**

**https://data.gov.sg/dataset/hcs-product-list**

**Elicit and Document requirements:**

1. **Determine the target users of your application. Elicit functional and non-functional requirements of the application. ~~Note that in doing this you may wish to have some of the team members act as customers and the rest of the team be the project development team. This is an artificial way of defining the requirements but within the constraints of this course project it is a practical way to proceed.~~**
2. **Document the requirements in appropriate technical format. The requirements should clearly state who performs what system functionality, taking what input and producing what output.**
3. **Atomise the requirements such that they are verifiable and traceable. ~~See page 126~130 of Fox for requirements specification heuristics.~~ ~~You will be writing test cases in Lab #4 to verify the requirements.~~ You will also be asked to demonstrate traceability from requirements to the final product.**
4. **Document important terms of your application (e.g., user, device, input, output) in a data dictionary. Explain each term with a brief description. Identify attributes of each term and relationships between terms.**

**Target Users:**

* People who exercise / eat healthily regularly
* People who are looking to start exercising / eating healthily
* People who want to exercise / eat healthily, but lack/don’t know the proper ways and methods
* Mainly between the age of 18-60
  + Below 18 - stranger danger
  + Above 60 - Bad with Technology?? (Can be deleted)

**Functional Requirements:**

1. **User** must be able to sign up for an **Account**.
2. **User** must be able to enter Create Account page to register a new Account in the Database. 
   1. **System** must be able to check if the User's **Account Username** is duplicated.
   2. **System** must be able to check if User’s **Password** fulfills requirements such as:
      1. At least 1 Uppercase letter
      2. At least 1 Lowercase letter
      3. At least 1 number
3. **User** must be able to reset his/her **Password**.
   1. **User** must be allowed to click on *“Forgot your password”* button to change their **Password**.
4. **User** must be able to log in to the **Webapp** by entering **Username** and **Password.**
   1. If **User** enters the wrong **Username**, **Webapp** will respond with an error message stating Wrong / Invalid **Username**.
   2. If **User** enters the wrong **Password**, **Webapp** will respond with an error message stating Wrong / Invalid **Password**.
   3. **System** will redirect **User** to **Explore** page.
5. **User** must be able to query the **System** for **locations** on exercising.
   1. **User** must be able to access the **Exercise Activity** page.
      1. When User enters a location in Search bar, User must receive a map of nearby(20-50m) exercising areas..
      2. **User** can click on the areas to find more information:
         1. Location Name
         2. Address
         3. Image of location
         4. Type of Facilities
6. **User** must be able to query the **System** for **locations** on healthy food.
   1. **User** must be able to access the **Healthy Food Locations** page.
      1. When User enters a location in Search bar on healthy food, User must receive a list of results stating nearby(20-50m) locations on healthy food.
7. **User** must be able to select if the **Webapp** has access to **User’s Location.**
8. **User** must be able to add **filters** to improve search results.
   1. Filters for exercise can include:
      1. Type of exercises:
         1. Running / Jogging
         2. Gym things
         3. Pilates / Yoga
         4. Exercise Corners / Fitness Corners
         5. Idk what else ppl do for exercising i dont exercise
      2. Type of food:
         1. Different Cuisines
         2. Restaurant / Food Court / Hawker Centre
         3. Fast Food (Healthy!) / Regular Food
         4. Etc
      3. **User** must be able to filter the **Location** based on rating of **Location** .
      4. **User** must be able to filter **Location** based on alphabetical order.
9. **Filters** must include searching for Food only or Exercise Only or both (For example).
   1. I think we can delete this cos we having 2 separate pages for exercising and food right?
10. **User** must be able to match up and communicate with other **Users** in a nearby location.
    1. **User** must be able to access the **Find A Partner** page
       1. **System** will show a list of **other Users** nearby based on **User’s** current location.
11. **User** must be able to leave comments / rates about the **locations/activities**.
    1. **User** can be prompted on comments / 0-5 star review of locations.
12. **User** must be able to mark their favourite & dislike locations and filter the search results based on their preferences.

**Non-Functional Requirements:**

* **Webapp** must load within 3 seconds.
* **Webapp** must be able to detect other users within a certain specified radius. (20-50m)
* User’s Account must be secure.
  + Strong / Weak password?
  + After a certain number of login attempts, System can lock the Account to protect User’s information.
* **Webapp** must load appropriately on different devices
  + CSS must account for devices of different screen sizes.
* **Webapp** must be available in different languages to cater for Users of all races / age.
* DB
* CMS
* API
* Security

**Data Dictionary {**

**User :** Person using **Webapp**

**Webapp :** User Interface to access data

**Account :** Personal account unique to each user

**Username:** A string of values unique to each user, used to access the User's **Account**.

**Password :** A string of values unique and private, used to access the User’s **Account**

**System :** Database that stores query results

**Location : Data** stored in **System** about priority locations (Healthy Food / Exercising Areas)

**Activity :** Types of exercise available at Location

**Filters :** Keywords available for a User to edit / manipulate results to User’s preference.

**}**

**Data Dictionary**

|  |  |
| --- | --- |
| **User** | Person using **Webapp** |
| **WebApp** | User Interface to access data |
| **Account** | Personal account unique to each user |
| **Username** | A string of values unique to each user, used to access the User's **Account**. |
| **Password** | A string of values unique and private, used to access the User’s **Account** |
| **System** | Database that stores query results |
| **Location** | **Data** stored in **System** about priority locations (Healthy Food / Exercising Areas) |
| **Activity** | Types of exercise available at Location |
| **Filters** | Keywords available for a User to edit / manipulate results to User’s preference. |
| **Find A Partner** | Page that allows user to find users nearby based on user’s current location |
| **Exercise Activity** | Page that allows user to find nearby exercising locations based on the location the user input. |
| **Healthy Food Location** | Page that allows user to find nearby healthy food locations based on the location the user input. |
| **Favorites** | Page that allows user to add their favorite location for easier access |