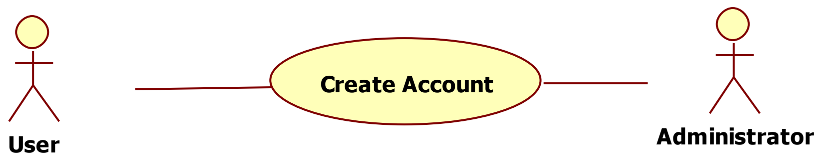
Technical Architecture

Carl Hand- X00091174

1. Use Cases:



1.1 Title:

Create account

1.2 Primary Actor:

User

1.3 Scope:

Organization

1.4 Level:

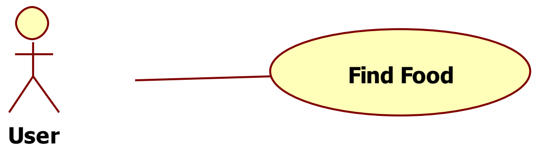
User Goal

1.5 Story:

The user selects to register on the app. The user enters in their account details and selects register. The system checks that the details entered by the user are valid and adhere to the guidelines in place by the administrator. The user's account has now been created and confirmation message is displayed.

Alternative:

- User's details do not adhere to the guidelines in place by the administrator. Error message displayed to the user.
- User's details such as username are already registered under an account on the system. Error message displayed to the user.
- User selects cancel before completion and is returned to start page.



1.1 Title:

Find Food

1.2 Primary Actor:

User

1.3 Scope:

Organization

1.4 Level:

User Goal

1.5 Story:

The user enters the food locator section on the app. They select which type of unique food they are searching for and click search. The system locates all stores nearby stocking the selected food item and displays the results to the user. User selects store to find out more information about it.

Alternative:

- User clicks search without selecting food they are looking for.
- System times out while searching for user's location. Notification message displayed to user. User returned to search page to try again.

User
Search For Training Programs

1.1 Title:

Search For Training Programs

1.2 Primary Actor:

User

1.3 Scope:

1.4 Level:

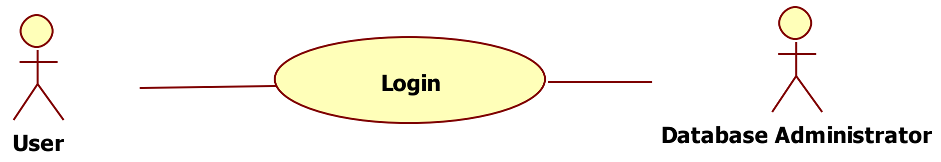
User Goal

1.5 Story:

The user selects which sport they play, their training goal, specific type of training they would like, i.e. aerobic or anaerobic (optional), the intensity of the training (optional) and clicks search. The system returns all programs matching the user’s requirements. User views programs.

Alternative:

- User does not select required details such as the sport they play. Error message displayed to the user.



1.1 Title:

Login

1.2 Primary Actor:

User

1.3 Scope:

Organization

1.4 Level:

Sub-function

1.5 Story:

User enters in login details such as username and password and selects login. The database is searched to verify that the account exists. User s greeted with welcome message. The user is navigated to the home page.

Alternative:

- User login details are incorrect and are addressed with an error message.
- User Account does not exist and are greeted with an error message.

User
Submit Training Results

1.1 Title:

Submit Training Results

1.2 Primary Actor:

User

1.3 Scope:

Organization

1.4 Level:

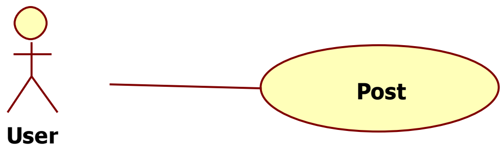
User Goal

1.5 Story:

User enters in results of training test before starting new training program. User submits details. System checks information and saves it to the users profile in the database. Stats are displayed to the user.

Alternative:

- User enters invalid details such as entering in a letter instead of a number for their 1 Rep Max for the squat. Error message displayed to user.



1.1 Title:

Post

1.2 Primary Actor:

User

1.3 Scope:

Organization

1.4 Level:

User Goal

1.5 Story:

The user selects to post a comment. The user submits a post about a program they have tried, something to do with the food locator, or any other queries they might have. The system posts the comment under the related topic. Post is displayed for the user to see.

Alternative:

- User clicks to post the comment before writing anything into the text box provided. The system displays an error message to the user.

2. Technical Architecture

2.1 Software Components:

For my project I will be making use of Microsoft's Azure to create the database for my system that will hold the user information when they register and also all the training programs and related material. I will be hosting the backend of my application on Google App Engine as this will allow me to really focus on the cloud applications operations and not fret about other areas such as reliability and scalability and also administration. Android studio is convenient in the sense that when a backend is created, Android studio provides the required libraries and a sample backend after it has generated the new App Engine application. My app will run on android platforms with a maximum API level 20 and a minimum API level of 8 as this will support 95% of people using android phones.

2.2 Platform libraries:

This application will be written in java and I will make use of the android APIs and Google APIs for the android platform as well as the Google location services API, which is part of the Google Play Services, for the food locator section of the application.

2.3 Distribution and Deployment:

I will use the Microsoft SQL server JDBC Driver in order to connect to the Azure SQL database. I will also deploy a cloud backend for my app which will back up all the user data and training programs. For the security side of my application I have chosen to use the Android Application Sandbox. This feature of Android will allow me to separate each of my application's data and code execution. Thus this will allow me to increase my security and also prevent malware. In addition to the Android Application sandbox, Android has many other features built in to provide security which I will take full advantage of to ensure my application is up to the highest possible degree of security. The data saved on the device will be stored on internal storage and backed up on the cloud. This will ensure that all the data is only accessible to my app. In order to guarantee the security of user data I will try to minimize the use of APIs that can access confidential data about the user. Also I will not store the user's vital data such as username and password on the device locally. Instead I will use the username and password provided by the user to carry out initial authentication and follow this by using a short-lived, service-specific authorization token. This will reduce the chances of security attacks such as phishing from occurring.

2.4 Risks:

Several risks are present that may affect the delivery of my project. Firstly I am using one programming language, java, to implement the various aspects of my application and I am depending on solely one way to approach the functional aspects of the application. In addition to this is the introduction of d3 and tableau which is completely new to me. These will be used to generate the graphs showing the user statistics and other information and so I must guarantee that I get to grips with this language so as the graphs are of the greatest standard that the user can easily view and understand. Ensuring that the application's performance does not decelerate and affect the user's experience is another risk associated when displaying these graphs. User's location is another risk at present for me since user location can vary in terms of accuracy if the user is their phone outdoors or indoors. Since a large part of my project is based on information about training programs, I must reassure the user that these programs being returned are reliable and are from creditable sources. Otherwise the user will be reluctant to use the app and search for training programs as they will feel the programs are not trustworthy.

3 Prototype

3.1 Prototype Deliverable for week 8

Register and login. User will first create an account/register on the app and then they will try to log in. Will test User register success and failure and user login success and failure.

3.2 Prototype Deliverable for week 11

I hope to have the Search For Training Programs use case delivered by week 11. Although the database will not be entirely full I aim to have a subsequent amount of data in it for the user to search through. The user will select sport and their desired training goal and click search. Desired training programs will be returned to the user. I will test for success where the user enters in all desired search details. I will also test for failure by omitting certain desired search details.