SET08114 Coursework Report

Rebecca McGowan
40289316@live.napier.ac.uk
Edinburgh Napier University - Mobile Applications Development (SET08114)

1   Introduction

The aim of this assignment was to create an android application that would accept user input and provide some sort of functionality. I decided to create an app that could be used as a companion to those currently on the Slimming World diet.

My decision to go with this idea is due to the current selection of related apps. Slimming World have their own official app but any helpful functionality is reserved exclusively for those who have a paid Slimming World membership, and many people follow the diet without subscribing, leaving them unable to use the app fully.

There are also many prominent food diary/health fitness apps currently on the market, such as MyFitnessPal or Fitbit, but these often focus specifically on tracking macro and micro intake or exercise regimes.

And whilst there are a few lesser-known apps that have been designed with Slimming World in mind, the features on these apps are limited and finding one that combines a weight loss tracker with a food diary can be difficult.

In addition, I always choose dark themes for any apps on my own phone as I find they strain my eyes less and this is something I haven't been able to find in current Slimming World related apps.

To determine what features the app should have, I made a list of everything I found useful in my physical food diary, plus any other features specific to apps that I thought could be useful. I then prioritised this list to decide what functionality to start implementing first.

My final feature list was as follows:

- Store daily logs of food drink that detail diet-specific allowances (i.e. point count)
- Store weight entries in a chronological list and show progress
- Store body measurements in a chronological list and show progress
- Have optional notification reminders for users to use the app daily/weekly
- Have colour scheme options
- Consider allowing the user to specify their own daily point limit
- Have some form of recipe storage

- Have a form of database to hold the food items that have been provided already

The most important functionality is the ability to track what food is eaten each day and to record diet-specific details for each item to determine how they fit in to the daily allowances. This meant that I needed to be able to have an overview of each day as well as being able to manage each food item individually. The second most important feature is to be able to record weight and calculate the loss/gain.

2   Software Design

When modelling my app concept, I started by focusing on the two key functions I had previously identified: food tracking and weight tracking.
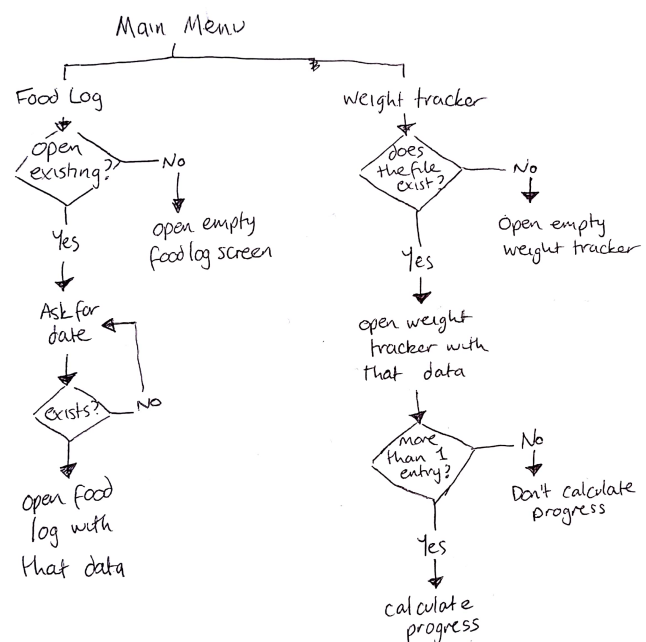

Figure 1: **App activities** - Accessing food logs weight tracker
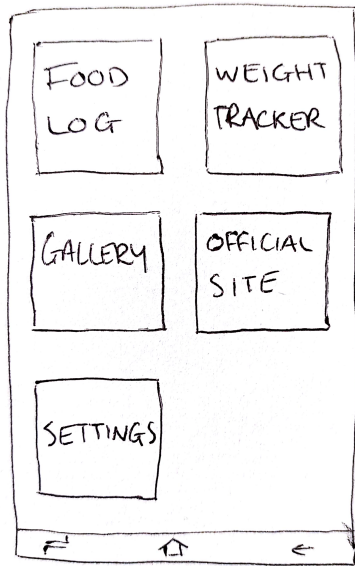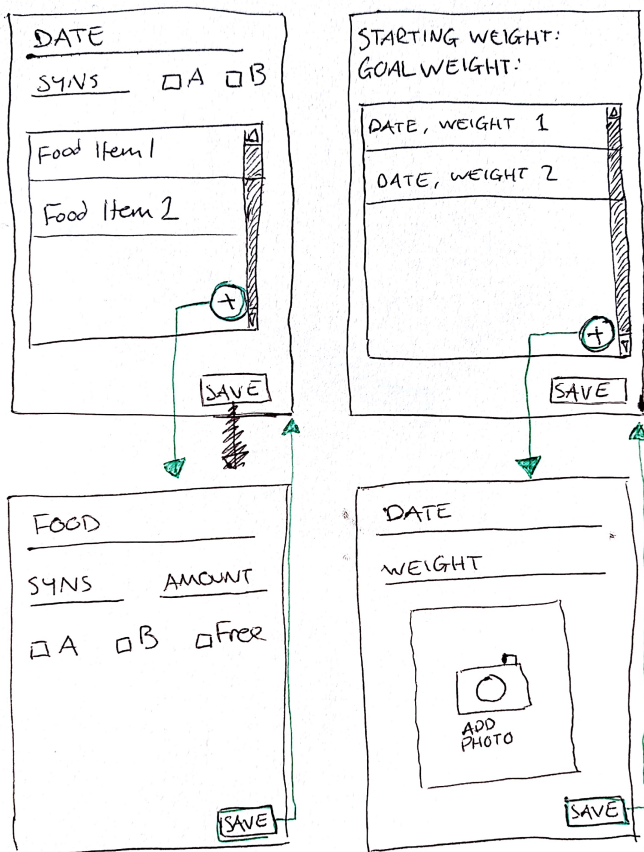
Figure 2: **Homescreen concept**

Figure 3: **Food weight concept screens** - Add/view food logs and weight tracker

3   Application Implementation

The actual implementation of my app remained largely faithful to my original concept with the exception of the main menu where square custom buttons were replaced with wide standard buttons. This is down to my lack of artistic ability and the fact I was uncertain how best to arrange the date field for existing logs. I plan to change this in future so that the main menu better resembles my original idea. In some instances, I removed fields from my original sketches that I felt did not add value.

From the homescreen [Figure 4] , the user can choose to create a new daily food log, open an existing log, or access the weight tracker. Creating a new food log or accessing an existing food log [Figure 5] provides the same functionality - add/amend the date, view/add/delete/amend food [Figure 6], and see the total number of points (syns) used that day based on the food that has been added. The weight tracker [Figure 7] opens a list of weight entries, defaulting the list box to show the most recent, and showing the progress made since the first weight entry. From here, the user can add/amend/delete weight entries [Figure 8]. Additionally, the user can access the official Slimming World website from the app.
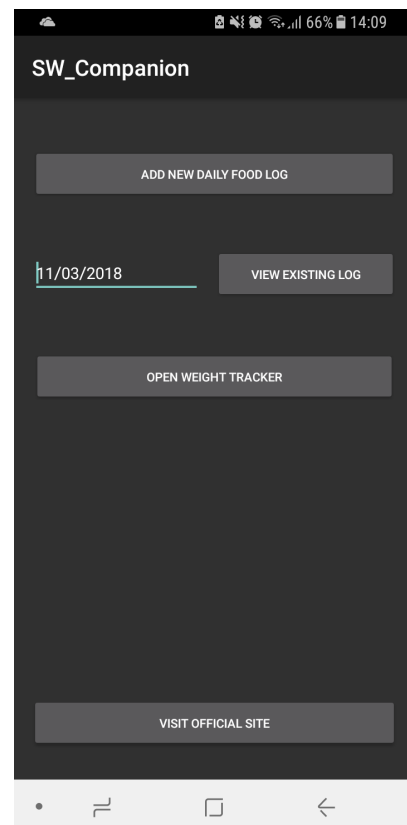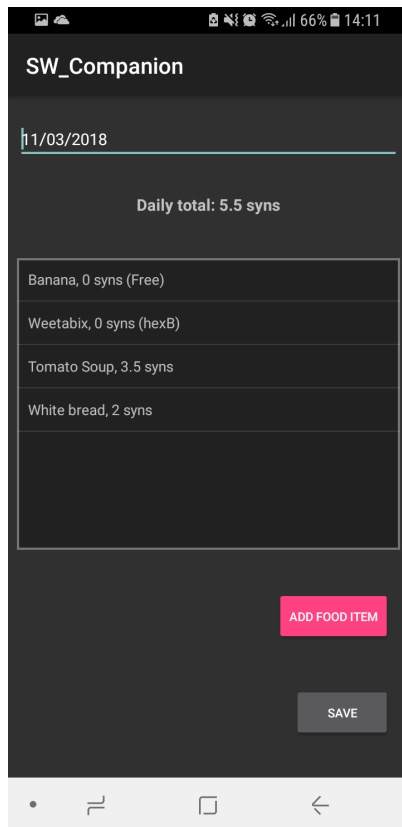

Figure 4: **Homepage** - Home screen

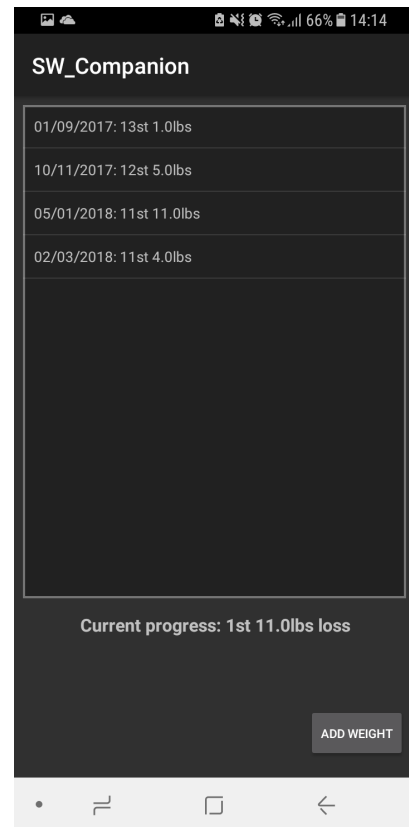Figure 5: **Food Log** - View/Add Daily Food Log

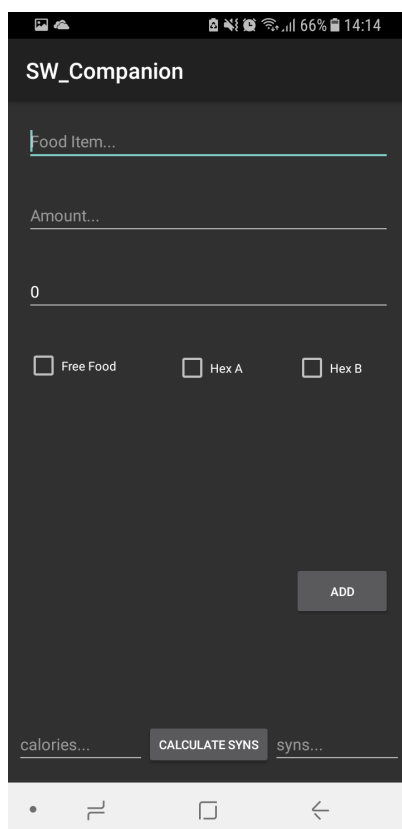Figure 7: **WeightTracker** - Weight Tracker
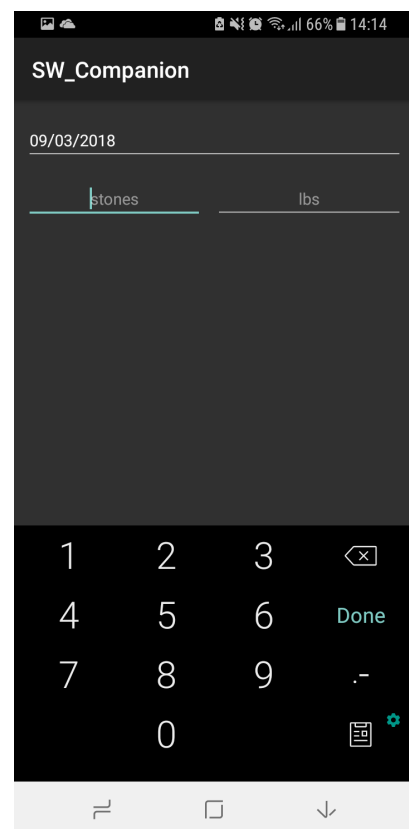

Figure 6: **Add Food** - Add/Amend Food Item

Figure 8: **Add Weight** - Add/Amend Weight Entry

4 Critical Evaluation

Original Concept Due to time constraints, my app is currently a very basic version of the concept I had in mind. Many of the features I had planned haven't been implemented and the original dashboard design I had has not been implemented as I reconsidered the decision to have the user enter the date of an existing log in a new window, which impacted the main design. The screens for the weight tracker and food log are almost exactly as imagined, although I stripped out some fields that turned out to be unnecessary in the food log.

Other Applications I had 4 particular apps in mind when designing my own: the Slimming World official app, My-FitnessPal, Slimming Tracker, and Food Counter (previously Syn Counter).

Slimming World official app: My app definitely doesn't offer the amount of functionality or the attractive design of this app for users who have a membership. However, my app does provide basic functionality free of charge to any user which Slimming World does not, and I believe this is more beneficial. I didn't expect to be able to measure up to this app in terms of design or overall functionality for those who have membership, but I did draw on this app for inspiration of what functionality is useful to those on the diet.

MyFitnessPal: This app was mostly inspiration for what I did not want to do with my own app. Having used this app for a long time in the past, I do feel it is a great app for those who aren't on a trademarked diet plan, but it not useful for those who are. I wanted to create an app that was as information-rich as this one but with the flexibility that I feel is needed for my own diet. As my app currently stands, it isn't as informative as MyFitnessPal but it does offer the flexibility needed by those on the Slimming World diet.

Slimming Tracker: I actually discovered this app when researching existing Slimming World apps after my initial concept and it is effectively the app I hope to create myself. It contains most of the features I would like to implement myself and some others that I had not considered. This app is my biggest inspiration with regards to the actual implementation of my app although I will be utilising a dark colour scheme instead.

Food Counter: There are quite a few apps similar to this one currently on the app store which makes it clear that people like having a tracker to count how many diet points they've used that day. I therefore made it a priority to offer the same point counting functionality so users can see how many points have been used as soon as they input food/drink or give them the ability to look back at previous days. Given that this functionality works, I would say that my app is currently more helpful this this one as it provides more context for each day but it wouldn't suit those that don't have an interest in keeping track of the specific food items and would rather have a quick and simple counter.

User Feedback As my app currently stands, it provides more functionality than a portion of the already available apps but there is still more functionality needed before it's suitable for more people (with different needs and requirements). In order to get tangible feedback, I had someone else currently on the Slimming World diet try the app out and received the following feedback:

Positives:

- The ListView in the food log makes it easy to see where various daily allowances have been used

- The syn calculator on the "add food item" page is very helpful and easy to use

- Like being able to amend food items/weight entries rather than having to delete erroneous ones and redo them

- It's helpful having a link to the official site

Negatives:

- Too easy to overwrite data if you've forgotten you already created a daily log for a specific date and create a new one again. There should be a warning or an automatic check to stop this happening

- Dark colour schemes don't suit everyone - there should be other options

- A measurements tracker would be beneficial in addition to the weight tracker

- Weekly notifications reminding you to add your weight would be useful

- The syn calculator on the "add food item" page is easy to overlook

- It would be useful to store frequently eaten foods so they can be added quicker in future

Improvements The most obvious improvement I could make to my app would be to improve it visually (through icons, buttons, images, etc), as right now it is not very visually appealing.

In addition to this, I would like to complete the original list of features (amending the priority of features to reflect the user feedback):

- Ensure users can't accidentally overwrite existing data

- Provide other colour scheme options

- Give the ability to store body measurements and view the progress

- Keep a database of frequently used food items

- Give users the option to toggle weekly notifications reminding them to input data to the app

Another feature I would like to implement is the ability to store photos internally with each weight entry that can also be viewed in a gallery within the app to provide a 'Before After' aspect. I also think it would be beneficial to make the app secure, through something like a PIN, so that users don't need to worry about others seeing their weight data or photos, given that many people do not readily share this information with others.

I also feel that it would be helpful to tweak the app so that it can be used by others on different trademarked diet plans, not just those on Slimming World.

5 Personal Evaluation

The biggest challenge I faced was saving and loading data from the app. I knew that I wanted to store files in the internal storage and that I would need to be able to serialize and deserialize objects but I was struggling to find resources to explain how to do that in a clear way. I finally found a clear explanation [1] which allowed me to progress with the app once I could save and load data.

The second biggest challenge was dealing with ListViews, again due to the information available. I came across many tutorials initially that complicated the process by requiring me to create custom adapters instead of explaining that I could use the in-built adapter provided I was only needing to display to toString() method for each object. Again, once I finally found a helpful resource that explained this [2], I was able to implement the ListView with the help of a tutorial [3] and continue making progress as I was now able to view what data I was saving.

I also had some challenges when dealing with dates and had to take some time to understand when and how I use Date-Format to deal with manipulating dates between Strings, Date fields, and text boxes. However, after getting this working once for saving and loading dates, this became much easier and the code was effectively reused throughout the app.

Another minor challenge was deciding how to deal with weight entries. Ideally, I would like to give users to option to choose between metric or imperial but for the sake of getting the app working, I decided to go with imperial as this is my own preference and I plan to use the app myself. The problem with imperial is that stones and pounds needed to be stored separately and calculating the difference between two weights is less straightforward. In the end however, I was happy with the solution I came up with which was to calculate the total weight in pounds, work out the difference, and convert the difference back in stones and pounds (using "loss" or "gain" in place of showing a negative value):

Listing 1: Weight difference calculation

```
1  private String getProgress() {
2      WeightEntry weight = weightFile.weightList.get(weightFile
       .weightList.size() − 1);
3      double lastWeight = (weight.stones * 14) + weight.lbs;
4      weight = weightFile.weightList.get(0);
5      double firstWeight = (weight.stones * 14) + weight.lbs;
6      double delta = lastWeight − firstWeight;
7      String movement = "";
8      if (delta < 0) {
9          movement = "loss";
10     } else if (delta > 0) {
11         movement = "gain";
12     }
13     double deltalbs = delta % 14;
14     double deltast = (delta − deltalbs) / 14;
15     String progress = "Current progress: " + (int)Math.abs(
       deltast) + "st " + Math.abs(deltalbs) + "lbs " +
       movement;
16     return progress;
17 }
```

I feel my app has turned out to be more basic than anticipated as I hadn't expected to have so much trouble with data storage or data display and I'm disappointed with this. Having used Java before, I hadn't expected to come across much trouble but Android Development was different enough from coding in Java in Visual Studio that there was a bigger learning curve for me to overcome than I expected. However, I do feel I learned enough about android development during the course of the project to give me enough confidence to continue developing apps, particularly this app, as I would like to get it to the finished result I had originally anticipated.

References

[1] Unknown, "How to read and write Java object to a file." https://www.mkyong.com/java/how-to-read-and-write-java-object-to-a-file/, 2016. [Online; accessed 16-February-2018].

[2] Unknown, "Using an ArrayAdapter with ListView." https://guides.codepath.com/android/Using-an-ArrayAdapter-with-ListView/, Year unknown. [Online; accessed 21-February-2018].

[3] A. Lim, "A Simple Android ListView Example." http://windrealm.org/tutorials/android/android-listview.php/, Year unknown. [Online; accessed 21-February-2018].