

Alex Page Final Project

by Alexandra Constance Page

FILE	17988_ALEXANDRA_CONSTANCE_PAGE_ALEX_PAGE_FINAL_PROJECT_78196_573573314.DOCX (13.98M)		
TIME SUBMITTED	21-APR-2016 02:46PM	WORD COUNT	23435
SUBMISSION ID	56207358	CHARACTER COUNT	150444

An investigation into the development of an android mobile application for recipe generation with the aim of tackling food waste reduction and tracking nutrition.

BSC BUSINESS INFORMATION TECHNOLOGY

Alexandra Page

4/21/2016

This document details the stages that will be executed during an investigation into the development of an android mobile application for food input and recipe generation which will aim to tackle food waste reduction and track nutrition information to aid a user in developing and maintaining a healthy lifestyle.

DECLARATION

I, ... declare that I am the sole author of this Project; that all references cited have been consulted; that I have conducted all work of which this is a record, and that the finished work lies within the prescribed word limits.

This work has not previously been accepted as part of any other degree submission.

Signed:

Date:

FORM OF CONSENT

I, ... hereby consent that my Project, submitted in candidature for the B.Sc.(Hons) Business Information Technology degree, if successful, may be made available for inter-library loan or photocopying (subject to the law of copyright), and that the title and abstract may be made available to outside organisations.

Signed:

Date:

Abstract

The following documentation is aimed at determining the need for tailored recipe generation software with food waste reduction functionalities. There is no current mobile application on the market that specifies a recipe index produced specifically using ingredients owned by a user.

The research approach instigated a combination of deductive and inductive viewpoints, with the deductive approach testing the hypothesis that the use of the application improves a persons' food waste management techniques. Primary research was collected via surveys and interviews, forming the basis of the systematic requirements.

Varied public opinions derived from data acquired via externalised sourcing and secondary research required an inductive observational approach. Experimental testing and survey evaluations preceded the development of the software, in order to support or contest the hypothesis.

The primary development paradigm appropriated to drive the application production was an amalgamation of Scrum and Extreme Programming practices, using short iterations and requirement prioritisation to optimize development.

Based on qualitative and quantitative findings derived from testing and feedback, it was concluded that there is a market for the application and a general consensus that it would be useful in every day practices. Therefore, further work in regards to the project would be beneficial to the industry.

Contents

Abstract	4
1.0 Introduction to the Proposed Research	11
2.0 Review of Literature	13
2.1 Introduction.....	13
2.2 What are Mobile Phones?	13
2.3 What are Mobile Applications?	14
2.3.1 The Platform Wars: Android vs. iOS	15
2.3.2 Mobile Health and Fitness Technologies.....	16
2.3.2.1 Mobile Health Applications	16
2.3.2.2 Wearable Technologies	24
2.3.2.3 Mobile Health and Hardware Integration	24
2.3.2.4 Current Mobile Health and Wearable Technology Market Trends	25
2.3.2.5 Mobile Health and Wearable Technology Advancements and Failures	25
2.4 Global Food Waste Statistics	26
2.4.1 Organisational Management and Solution Determination	27
2.4.2 Management Successes and Failures	27
2.5 Review of technical solutions: IOS.....	28
2.5.1 History and Market Penetration.....	28
2.5.2 Development Platforms Considered	28
2.5.3 Ease of Development.....	29
2.5.4 Adaptability	29
2.5.5 Hardware Devices.....	29
2.6 Review of technical solutions: Android	29
2.6.1 History and Market Penetration.....	29
2.6.2 Development Platforms Considered	30
2.6.3 Ease of Development.....	30
2.6.4 Adaptability	31
2.6.5 Hardware Devices.....	31
2.7 Review of Literature Conclusion.....	31
3.0 The Overarching Aim and Objectives	33
3.1 Aims	33
3.2 Objectives	33
4.0 Aims and Objectives Satisfaction Determination	34

5.0 Research Methodology.....	34
5.1 Data Accumulation	34
5.1.1 Secondary Research	35
5.1.2 Primary Research.....	35
5.2 Research Philosophy and Approach	35
5.3 Research Strategy and Time Horizon	37
5.4 Data Collection Method Including the Study Population and Sample	39
6.0 Ensuring Reliability and Validity	40
7.0 Development Methodology	41
7.1 Selected Development Platform	41
7.2 Review of Development Methodologies	41
7.2.1 Classic Methodologies	42
7.2.2 Agile Methodologies.....	44
7.3 Intended Development Methodology with Justification	47
7.3.1 Overcoming Disadvantages of the Approach	48
7.3.2 Taking Advantage of any Strengths of the Approach	48
8.0 Android Application Design	49
8.1 Introduction.....	49
8.2 Requirements Specification.....	49
8.3 Development Environment and Language	49
8.4 Database Design	50
8.4.1 Entity Relationship Diagram	50
8.5 Use Case Diagrams	51
8.5.1 Top Level Use Case	51
8.5.2 Generate Recipes Use Case	52
8.6 Class diagram.....	53
8.7 Navigation and Main Menu	55
9.0 Implementation	56
9.1 Introduction.....	56
9.2 Operating System Version	56
9.3 Extensions, Plugins and Themes.....	57
9.4 Building the Application	58
9.4.1 Building an Activity	58
9.4.2 Validating User Input	59

9.4.3 Creating the Database and Tables.....	59
9.4.4 Placing Data into A Table.....	60
9.4.5 Auto-Populating Data from the Database.....	61
9.4.6 Passing Data from one Activity to Another	62
9.5 SQLite Manager Firefox Plugin	62
9.6 File Structure	63
9.7 Utilising the Application	63
9.7.1 Signing Up.....	63
9.7.2 Adding Ingredients	63
9.7.3 Adding Items to the Pantry	64
9.7.4 View Pantry Items and Expired Pantry Items	64
9.7.5 Add Recipes and See all Recipes Available	65
10. 0 Results and Evaluation.....	66
10.1 Introduction.....	66
10.2 Test Plan	66
10.2.1.1 Sign Up Class and Activity Evaluation.....	66
10.2.1.2. Evaluation	66
10.2.2.1 Profile Class and Activity Evaluation	67
10.2.2.2. Evaluation	67
10.2.3.1 ExpiredPantryActivity Class and Activity Evaluation	67
10.2.3.2 Evaluation	67
10.3 User Testing and Feedback.....	68
10.3.1 Average Testers Feedback.....	68
10.3.2 Security Analyst Feedback.....	68
10.3.3 Security Analyst Comments.....	68
11.0 Survey Analysis	69
11.1 Question 1: What is your gender?.....	69
11.1.1 Results and Diagram.....	69
11.1.2 Result Expectancy.....	69
11.2 Question 2: What is your age?	69
11.2.1 Results and Diagram.....	69
11.2.2 Result Expectancy.....	69
11.3 Question 3: Which Smartphone do you own?	70
11.3.1 Results and Diagram.....	70

.....	70
11.3.2 Result Expectancy.....	71
11.3.4 Link to Literature Review.....	72
11.4 Question 4: What kind of apps do you use the most? (Select up to 3 options).....	72
11.4.1 Results and Diagram.....	72
11.4.2 Result Expectancy.....	72
11.4.3 Link to Literature Review.....	72
12.0 Evaluation	74
12.1 Must Requirements	74
12.2 Should Requirements	74
12.3 Requirements Evaluation	74
12.4 Aim and Objectives Satisfaction	75
12.5 Data Correlation Regarding Literature Review.....	75
13.0 Conclusions and Recommendations	76
Does the use of a mobile application allow for efficient monitoring of food expiry dates? ...	76
Are there any limitations to the use of mobile applications used for the generation of recipes?	76
Can a mobile application be a viable method in tracking nutrition intake automatically?	76
13.1 Further Work	77
14.0 Reflection.....	78
15.0 References	79
16.0 Appendices	87
Appendix A: MoSCoW Requirements.....	87
Appendix B: Behavioural Diagrams	89
Expiry Dates Use Case	89
Personal Profile and Nutrition Use Case	90
Appendix C: CD ROM Application.....	90
Appendix D: All Android Studio Plugins Used.....	91
Appendix E: Test Plan	92
Appendix F: User Testing Images	95
Appendix G: Survey Results Analysis	96
Appendix H: Survey Design.....	100

Table of Figures:

Figure 1: Nokia 6110 (first mobile running snake)	14
Figure 2: Computerworld Magazine Snippet (June 8th 1981)	14
Figure 3: App Store Daily Growth	15
Figure 4: Apple App Store Category Popularity Ranking (March 2016)	16
Figure 5: Fitbit Activity Tracker Models.....	17
Figure 6: Various Yummly Interface Views.....	19
Figure 7: Various Cookpad Interface Views.....	21
Figure 8: Love Food Hate Waste Marketing Campaign	22
Figure 9: FridgeCheck Ingredients Interface	23
Figure 10: Sensoria Smart Socks and Integrated Anklet	24
Figure 11: Mobile Health Infographic (GreatCall).....	25
Figure 12: Annual Global Food Waste Statistics (Chicago Council).....	26
Figure 13: Intermarche Inglorious Campaign.....	27
Figure 14: OS App Development Market Share (2013)	31
Figure 15: Research Onion	34
Figure 16: Example Waterfall Methodology	42
Figure 17: Spiral Methodology Example.....	42
Figure 18: XP Methodology Workflow	45
Figure 19: Scrum Methodology Overview	46
Figure 20: ERD for FGF.....	50
Figure 21: Top-Level Use Case for FGF	51
Figure 22: Generate Recipes Use Case Diagram	52
Figure 23: Application Class Diagram	53
Figure 24: Main Menu Application Navigation Structure	55
Figure 25: FGF Main Menu Activity.....	55
Figure 26: Create New Project Initial Window (Android Studio)	56
Figure 27: Java Development Kit 8 Plugin Products	57
Figure 28: Select Activity Page	58
Figure 29: Configure Activity Setup.....	58
Figure 30: XML code for activity object layout.....	58
Figure 31: Android Studio App SignUp Setup.....	58
Figure 32: Password Validation Java Code	59
Figure 33: Database Initialization and Table Creation SQLite Queries.....	59
Figure 34: Inserting Data into FoodItems Table	60
Figure 35: SQL query to place data in FoodItems Table.....	60
Figure 36: Cursor to obtain table row	61
Figure 37: Database Handler Cursor Code	61
Figure 38: Bundle to pass sign up details to profile activity.....	62
Figure 39: Bundle After Passed to Profile.....	62
Figure 40: ADM Screen	62
Figure 41: FoodGloriousFood DB in SQLite Manager	62
Figure 42: Java Class File Structure of FGF	63
Figure 43: Activity Layout File Structure of FGF	63

Figure 44: FGF SignUp Activity	63
Figure 45:FGF Profile Activity	63
Figure 46: FGF Ingredient Activity	64
Figure 47: FGF AddToPantry Activity.....	64
Figure 48: FGF Pantry Activity	64
Figure 49: FGF Expired Pantry Items Activity	64
Figure 50: FGF AddRecipe Activity.....	65
Figure 51: Email exchange between developer and security specialist	68
Figure 52: Survey Q.4 response with OS division	70
Figure 53: Survey Q.6 response with OS division	70
Figure 54: Survey Q.5 response with OS division	70
Figure 58: Survey Q.9 response with OS division	71
Figure 58: Survey Q.8 response with OS division	71
Figure 58: Survey Q.7 response with OS division	71
Figure 58: Survey Q.10 response with OS division	71
Figure 59: Survey Q.4. combined results.....	72

1.0 Introduction to the Proposed Research

Over recent years, a surge in the use of technology for everyday life has been evident, with people becoming more and more attached and reliant on their technology devices. Research into the use of mobile phone technology has revealed that an overwhelming number of individuals feel as though they cannot live without their smartphones and have declared that mobiles provide them with freedom and a connection to the world.

Resultantly, this technological dependence has revealed an entirely new market for organisations and developers, who are able to provide smartphone owners with mobile applications in order to augment their experience, increase productivity and monitor their lifestyle. However, one daily necessity that has received little consideration in the mobile application industry is recipe generation and food management.

Although many methods have been deployed to digitally *assist* an individual in producing meals or recipe concepts, such as e-books, blogs and online tutorials, there are currently no simplistic or adequate solutions for the provision of a tailored recipe index based on food items currently within a user's possession.

This documentation details the processes required to accomplish the production of an Android mobile application which will allow a user to input food items into a digital pantry to establish a personalised database from which recipes may be generated. The application will include additional features, such as notifications informing the user of any advancing expiry dates on the items within their virtual pantry. A traffic-light prioritisation system will be allocated to the food items, encouraging users to utilise recipes including items close to their expiry date, thusly aiding in food waste reduction.

The application will require the input of personal information pertaining to the user, so as to make the recommended health-related figures relative to the user and as accurate as possible. Filters will be allocated to the recipe generator so that users may restrict the recipes available in order to make improved dietary choices.

A compilation of introductory objectives will aid in directing the research and development processes to satisfy the overall aim of the project. Preliminary actions in the form of a literature review will be completed to ascertain whether the current market would benefit from any additional research pertaining to the topic, in conjunction with a clear establishment of any existing related technologies.

A research strategy will be actualized to gather information pertaining to the general subject. Sufficient secondary research should form the basis of the literature review, whereas primary research will provide data and metrics pertaining to the proposed application to establish user requirements. Additional primary research will be essential during the evaluation process; user testing and feedback will conclude the success of the application. The development methodology segment will evaluate a selection of paradigms applicable to the project and a determination of the most adequately suited will be applied to the development process.

In order to ensure the application is supplemented with an effective design, a series of structural and behavioural diagrams will be created. All initial design inputs will be displayed and annotated. A testing plan will depict the success or failure of the application processes, and determine their relation to objectives satisfaction. An analysis of survey results will explore predetermined expectancies and response relations to the literature review. Graphs may be supplemented to illustrate statistics.

To determine the overall success of the application, an evaluation will be executed in which user feedback will be gathered and investigated against the aim and objectives to determine their satisfaction. To conclude, project outcomes and recommendations will be disclosed, in addition to a reflection of the general proceedings of the project.

2.0 Review of Literature

2.1 Introduction

During this section of the document, a thorough review of any research relating to the area of study will be completed with the inclusion of all current existing solutions and technologies, in conjunction with any inadequacies discovered. This section will aim to discover whether the proposed research topic conforms to the area of study, in addition to whether the current state of the market would benefit from any additional research.

A statistical analysis will be executed regarding the current conditions of global health and fitness to determine whether the application would be suited to penetrate that market. The same method of analysis will be applied to the food waste management industry, so that it can be determined whether the proposed application would be suited to a variety of market areas. An appropriate development platform will be determined based on an analysis of any current platforms available, in conjunction with their benefits, successes and market demands.

2.2 What are Mobile Phones?

In the Encyclopedia of Mobile Phone Behaviour [1], the mobile phone is defined as a “telephone with access to a cellular radio system that enables it to be used over a wide area, without a physical connection to a network.” During the year 1973, the first mobile phone prototype was created by a technology development company named Motorola, and weighed a hefty 4.4 pounds. The phone allowed for a mere 30 minutes of talk-time and took around 10 hours to charge.

Fast forward ten years to 1983 to the release of the world’s first commercial mobile phone, the Motorola DynaTAC 8000X [2], which granted 30 minutes of talk-time, stored up to 30 numbers and cost a staggering \$3995 (£2635). From then onwards, other mobile phone developers began entering the market, such as Nokia, who launched their first handheld mobile, the Mobira Cityman 900, in 1989 [3]. From 1990 to present day, mobile phone development has increased swiftly and drastically, with devices becoming lighter and more visually appealing, and costs lowering so that the typical consumer is able to purchase one.

2.3 What are Mobile Applications?

As defined by the Management Association for Information Resources [4], a mobile application is:

“a software application developed specifically for use on small, wireless computing devices, such as smartphones and tablets, rather than desktop or laptop computers.”



Figure 1: Nokia 6110
(first mobile running
snake)

Mobile applications were initially presented towards the end of the twentieth century, with the first mobile application for the average consumer being the Short Message Service (SMS), introduced in 1990 [5]. Over the next decade more applications appeared in the form of small arcade games and utility applications such as calculators, clocks and calendars, first on the Nokia 6110 [6]. The real game changer occurred when Apple launched their ‘App Store’ on July 10th, 2008, containing a mere approximate 550 applications. In an interview with The Wall Street Journal, Steve Jobs [7] announced that within one month of activity, the number of app downloads had reached sixty million. By November 2009, the number of applications on the store had reached 100,000.

These days, smartphones are typically purchased with basic applications preinstalled, with the consumer having access to additional applications via a distribution platform specified to the device, called an ‘app store’. The term ‘app’ is an abbreviation of ‘application program’ and has become universally acknowledged in recent years, being listed as ‘Word of the Year’ in 2010 by the American Dialect Society [8]. According to the Oxford English Dictionary [9], the earliest use of the term ‘app’ occurred

in 1985 by Apple, when discussing their MacApp product. However, there is proof of the use of the phrase four years prior, in the jobs section of a Computerworld Newsletter [10] from June 8th, 1981.

SR COBOL PROG/ANALYST
Mktg div of leading computer vendor seeks P/A with exp in sales/mktg apps. Must have COBOL & either interactive, data base or distributed exposure on any vendor's hardware. Outstanding benefits & working conditions. Salary \$25,000

Gerry Battista
* **ROBERT HALF**
of Boston, Inc.
* **100 Summer Street**
Boston, MA 02110
(617) 423-1200

Figure 2: Computerworld Magazine Snippet
(June 8th 1981)

2.3.1 The Platform Wars: Android vs. iOS

The most prominent application distribution platforms are currently the iOS Apple App Store and the Google Play store, previously known as the Android Market. Statista [11] announced that in 2014, the total revenue generated from worldwide app downloads amounted to 8.3 billion US Dollars. In November 2015, the Google Play Store had 1.8 million apps available for download, surpassing the iOS App Store's slightly lower 1.5 million apps.

Over the past 5 years, Google has retained an approximate growth of 200,000 additional apps annually, whereas iOS sees an average yearly addition of 300,000 apps [12]. The Apple App Store reached a total of 100 billion cumulative downloaded apps since its opening in July 2008 to June 2015. However, from June 2014-15, Google Play had 50 billion app downloads, doubling iOS, who stood at 25 billion [13].



Figure 3: App Store Daily Growth

During the first quarter of 2015, the Wall Street Journal [14] announced that Google Play had 70% more app downloads in comparison to the Apple App Store, though Apple's mobile app revenue amounted to approximately 70% greater than that of Google Play. Although Google Play has a larger market share, totalling at approximately 80%, their customer base displays lower spending per-user, allowing iOS to surpass Android on revenue generated by app downloads [15].

Many speculations have been made in regards to this market insight; one prominent suggestion being the probability that due to the expensiveness of the products, Apple product owners come from families with higher incomes, thus assumptions were made that the users are therefore willing to spend more on apps than Android owners [16]. However, Windsor [17] suggested that although Google Play Store's revenue is less than its direct competitor, it is forecasted that Google Play will match Apple's mobile app revenue by 2018 and subsequently surpass it.

Although Google Play holds a vast amount of mobile apps, only 31.2% of these are paid apps [18]. Thus, in November 2013, 98% of their revenue generated by app downloads was a result of 'freemium' apps, a free app with in-app purchasing. A mere 1% of revenue was generated from paid apps. Apple's App Store also relies heavily on freemium apps to generate revenue,

with their freemium app-related profit growing from 77% in January 2013 to 92% in November 2013 [19].

2.3.2 Mobile Health and Fitness Technologies

2.3.2.1 Mobile Health Applications

With worldwide obesity levels rising considerably over the last few decades, and having “more than doubled since 1980” as stated by the World Health Organisation [20], people are in desperate need to develop methods in which to manage their nutritional intake. As a result, mobile applications for health and fitness are becoming increasingly more popular, “with over 97,000 health and fitness related mobile apps currently on Google Play and Apple App Store, and 4 million downloads per day” according to GreatCall [21], self-branded as America’s leader in mobile health and safety.

Although gaming applications are ranked as the most popular category, dominating the Apple App Store at 22.21% [22] and 14.8% [23] on Google Play, Digital Trends [24] argue that Health and Fitness is the “fastest growing app category, reflecting just how much the mobile industry is focused on our well-being.” Resultantly, Google have now introduced ‘Google Fit’, Apple has provided customers with ‘Health’ and Microsoft competed with their ‘Microsoft Health’ pre-installed applications that are integrated into the features of their phones.

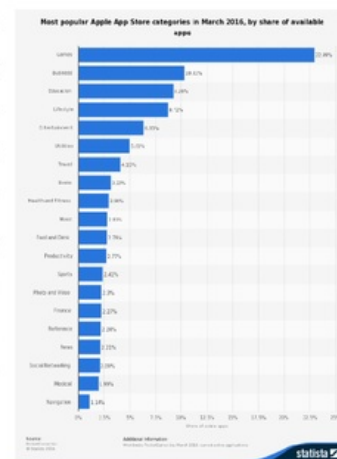


Figure 4: Apple App Store Category Popularity Ranking (March 2016)

Aside from preinstalled applications, there are many popular health-related mobile apps available to download from app stores. These may be partitioned into varying categories, such as nutrition logging, activity logging, recipe inspiration and generation and so forth.

2.3.2.1.1 Activity Logging

Noted are the three most-popular mobile apps for iOS and Android that focus on allowing users to log and track physical activity.

Fitbit: Designed to work with activity

trackers and Fitbit smart scales, this mobile application tracks various

activities performed by users, including walking and running, by wirelessly syncing to the device. Depending on the type of activity tracker worn by the user, the app can also track sleep and heart rate patterns. The app has over 10 million downloads on Google Play alone, and is also available for iOS devices. Users may log their food consumption and the app will then calculate their calorie intake and outtake. Challenges can be set and by competing with friends, leader boards are generated to display who is ranked highest, thus aiding in retaining motivation. Additional workouts may be logged to result in a more accurate daily overview. Progress may be monitored through the use of various timeline graphs.

Google Fit: Although available for Android users only, the Google Fit app has received between ten and fifty million downloads, ranking as one of the most-downloaded fitness apps of all time. With the ability to link the app to an Android Wear watch, activities such as sleep, speed, elevation, distance and routes can be tracked. Without a synced watch, the phone alone is able to log activities such as walking, running and cycling. Users can set goals and utilise the app to help achieve them through notifications and activity graphs. However, although calorie outputs are recorded, users are unable to log calorie intake, and so net calories are not calculated.

Lifelog: By collecting data from various apps in conjunction with utilising inbuilt phone sensors and wearable technology, Lifelog tracks various activities, such as walking, cycling, running and sleeping. Calories are tracked automatically and updated with every step, allowing for more accurate outputs. Graphs are displayed over days, weeks, months or years, so that users may view activity trends over various time frames. Notifications alert the user when certain goals have been met; all goals are customisable.



Figure 5: Fitbit Activity Tracker Models

The app also logs social activity, such as time spent on social media, time spent listening to music or playing games, and how many photos were taken on the device that day. Notable moments may be bookmarked for future reference. All activities are logged in a daily interactive timeline, and the user may 'play' the day to check all activities.

One downfall is that calorie intakes cannot be noted, so the user has no way to keep track of net calories. The app also does not allow for GPS logging, nor does it display pace or elevation for tracking runs or hikes.

2.3.2.1.2 Nutrition Logging

Below are the top three mobile applications for iOS and Android that focus on allowing users to track their nutritional intakes.

My Fitness Pal: With over ten million downloads and 1.6 million reviews on Google Play alone, My Fitness Pal is deemed as one of the most popular free apps in the Health and Fitness category, having achieved an overall customer rating of 4.6 out of 5. Available on both Google Play and Apple App Store, the app focuses on calculating the daily nutritional intake of a user by allowing frequent logging using an existing database of over 5 million foods. Users may input items by using a barcode scanner or enter the name of an item and select the respective option from the list provided. All food input options are fully customisable so that quantities can be taken into consideration to ensure accurate logging. The user may set personal goals and targets using features of the app and review progress with the incorporation of charts.

In conjunction with nutrition logging, the app also allows for activity logging, with a database of over 350 cardio exercises. My Fitness Pal also allows the user to connect with other apps, such as Fitbit, Jawbone UP and MapMyFitness, to reduce time consumption through multiple data entry.

WeightWatchers: As the mobile app that corresponds to one of the most well-known weight loss programs worldwide, WeightWatchers allows users to track their food intake through the incorporation of barcodes for reduced data entry, whilst conjointly displaying the "ProPoints" (values given to different food items to note how good they are for you) for each food item, in order to make better choices whilst shopping. Users may add custom meals or recipes and calculate their ProPoints, or search through a large database of predefined recipes. Weight-loss progress can be monitored through the use of charts generated based on user input. The app also grants activity logging, allowing the user to connect to various fitness devices, such as Fitbit or Jawbone trackers.

Lost it!: Succeeding the input of personal information, such as weight, height, gender and weight-loss goals, Loselt! generates a custom plan for the user by setting a daily calorie intake allowance. The user then actively logs all food intakes and the application calculates the daily intake for the user, notifying them if they are under or over their allowance. The app provides access to a large database of items with predefined nutritional values for various foods, to allow for simpler logging. The app allows users to challenge friends or other users to aid in motivation and continuation. One downfall of the app is that it tracks basic nutritional values; users cannot record their vitamin intakes such as calcium or iron, which may be necessary for anaemic or calcium deficient users.

2.3.2.1.3 Recipe Inspiration and Generation

Listed are the most popular mobile applications for iOS and Android with a primary focus of providing users with meal ideas and recipe instructions.

Yummly: Available on both iOS and Android, Yummly has had over one million downloads on Google Play alone, and has an overall customer rating of 4.5 out of 5 on both platforms. The application contains various customisable features to allow for a more tailored experience, such as the ability to filter recipes to exclude allergies or dislikes, or to suit to a particular



Figure 6: Various Yummly Interface Views

diet such as vegetarian, gluten free and so on. Recipes can also be discovered based on cuisine filtration or nutritional requirements. The application incorporates a time-saving feature by allowing users to order or filter recipes based on cook-times. Recipes can be saved to favourites and then sorted into food types to allow for simplified location.

Users may add singular ingredients from a recipe or the entire ingredients of a recipe to a digital shopping list, which is then automatically organised based on store aisles or recipes, depending on the users' selection. The user may tick items from the list as they shop to minimise confusion. One negative aspect of the application is that it does not allow users to add recipes to a meal plan, nor does it allow users to generate recipes based on ingredients they already possess.

In The Kitchen: This application allows users to access thousands of recipes by various chefs from the Food Network, with new recipes being added every month. Tutorial videos are provided to simplify the process and aid the user in correctly producing a meal. The application allows users to save their favourite recipes in conjunction with generating personal recipes and storing them on the app. Users may also add ingredients to a digital shopping list which may be accessed without a network connection. Searches can be made by chef or by ingredient, and recipes can be tailored by adding notes, tips or substitutions.

However, the app does not allow for tailored recipes, nor does it allow users to generate meal plans or add recipes to a schedule. The users cannot save ingredients to a digital pantry or kitchen to monitor what items they already have.

Cookpad: Previously known as “AllTheCooks”, this app is available on both iOS and Android and has received over 10 million downloads. With over 150,000 recipes available, the app is a combination of a social and health app as recipes are generated by the users. Other users can review and respond to recipes and post pictures of their attempts, with an option of posting the recipe to various social media sites. The availability of forums allows for discussions on various food and health-related topics. Items may be added to a digital shopping list for offline

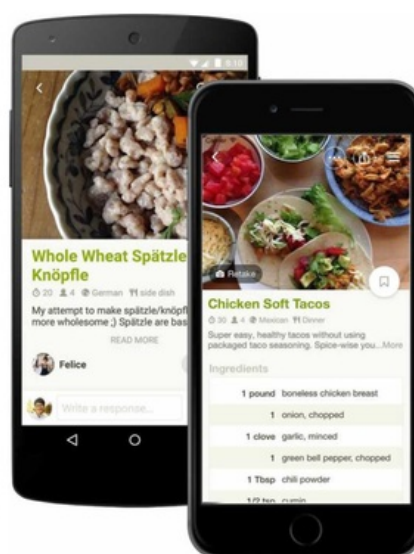


Figure 7: Various Cookpad Interface Views

access.

Although the app provides calorie and nutritional information for most recipes, users are not able to browse recipe indexes with nutrition-based filtrations, nor can filters be added to exclude items from recipes, which may prove to be problematic for users with allergies. Recipes cannot be generated based on existing items obtained by the user, nor can recipes be added to a meal plan; they may only be saved.

2.3.2.1.4 Direct Competition of the Proposed Application

Listed are the three most notable mobile applications for Android or iOS that are considered a direct competitor of the proposed application, based on the similarity of features and functions. A summarised review will be provided for each, with a note of all similar features included in conjunction with any features that are not present.

Love Food Hate Waste: This mobile app aims to tackle food waste by providing recipes that aim to use up leftover or surplus food items. Users may add items to a digital shopping list and, once purchased, add it to a digital fridge, freezer or cupboard. Recipes can be added to



Figure 8: Love Food Hate Waste Marketing Campaign

a 14-day meal planner to aid in organizing food use. The app includes a portion planning tool, with adult and child options, so that users are sure not to cook excess food.

However, the app has a 2.7 out of 5 rating, with many reviews outlining its limited choice of food items, in conjunction with being unable to select where food items are stored. The app does not allow for users to enter food items through barcode scanning and so it is time consuming to enter all foods. All items are calculated in grams, proving to be problematic for individuals working in units or pounds. Items are added to 'My kitchen' by manual selection from a list as there is no search option. Users are also unable to track nutritional values for the food entered, and so recipe calculations and daily intakes must all be completed manually.

Food Planner: With over 1 million downloads, Food Planner is the most downloaded recipe management and planning app and is available for both iOS and Android users. Users are able to access multiple databases of recipes and import said recipes into the app. Meal plans can be generated manually over various time frames, or users can complete in-app purchases for pre-defined meal plans. A digital inventory is available, which stores all items the user wishes to enter. Items can be added to a digital shopping list to be used as a reference when purchasing items to use in recipes.

Although the app has a decent rating of 4.0, many users have noted that the app requires a lot of initial set up and there are no auto-fill options available, which is time consuming. The barcode scanner is temperamental and does not acknowledge or recognize many items.

Users are unable to generate recipes based on the items in their inventory, nor are items removed from the inventory once a recipe is added to a meal plan. No nutritional information can be given for food items so should users wish to track their daily intake they must calculate all values manually. Recipes cannot be generated based on items within the inventory; they must be generated manually and stored within the 'Recipes' section, or downloaded using in-app purchasing. Items within the inventory cannot be given an expiry date, and so although users may rely on the inventory to keep track of what items to buy for planned meals, it does not take into account items that may expire.

Fridge Check: Available for Android only, Fridge Check is an app that allows users to enter a series of ingredients and select a time frame and a category, which will then generate a collection of recipes with the ingredients listed. Should the recipe require an ingredient that is not entered, users may send an in-app SMS to a friend, family member or partner to pick up said item. Items the user wishes to exclude for preference or dietary purposes can be entered, and recipes that include these items will not appear. Recipes can be added to favourites for future reference.

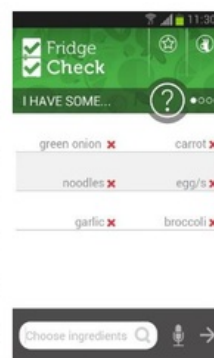


Figure 9: FridgeCheck Ingredients Interface

One issue with this application is that each time a user wishes to generate a recipe, they must enter all of their ingredients, which is time consuming, as there is no option to save ingredient data to the application. Measurements seem to be inconsistent or unclear, with no labels being added to numeric values to differentiate between grams, cups, teaspoons and so on. The application does not allow for meal planning, nor does it calculate nutritional intake.

2.3.2.2 Wearable Technologies

A decade ago, when considering wearable technologies, it was typically associated with spy gear and the tools of fictional characters such as inspector gadget. The average consumer could only dream of owning such futuristic technology, and certainly not any time in the near future. However, the technology advancements emerging as an outcome of this global health focus are wearable fitness gadgets that correspond with software platforms to track and log a variety of health activities and behaviours.

Among these are the FitBit wristbands, the Apple watch and the Jawbone trackers. Furthermore, a company named Sensoria has developed sensory clothing, with an attire range including a fitness sports bra, socks with textile sensors connecting to an anklet and a fitness t-shirt with an inbuilt heart rate monitor. Procured from a review by Gregory Ferenstein on The Verge [25], Ferenstein describes the sensorial ankle-sock duo as “one part ankle bracelet, three parts personal trainer.”



Figure 10: Sensoria Smart Socks and Integrated Anklet

2.3.2.3 Mobile Health and Hardware Integration

Mobile phones are now integrating health features into the physical design of their devices, such as the Samsung S5 and S6 with their built-in heart rate monitors. The sensor corresponds with the mobile application, 'S Health', to monitor the users heart rate and to detect trends and fluctuations in measurements over various time frames.

As stated by Andrew Martonik [26], editor for Android Central, this technological implementation is viewed as a “big hardware advancement” for Samsung. With the projected mobile health market being a “26 billion dollar industry by 2017” [21], it is assumed that mobile companies plan to continue with health feature integration in future devices.

2.3.2.4 Current Mobile Health and Wearable Technology Market Trends

Mobile health has now been abbreviated to mHealth, “a term used for the practice of medicine and public health supported by mobile devices” [27] and can be categorised as medical and non-medical. In research completed by the Institute for Healthcare Informatics (IMS) [28], it was discovered that on the Apple Store alone, there are slightly over 40,000 health related apps, with 70% of these being targeted at the general population. The remaining 30% are more specialised and aimed at professionals or specified sectors.

Simms [29], co-founder of Apadmi, the UK’s leading mobile app developer, defines wearable technology as “clothes and accessories that incorporate computer and advanced electronic elements.” The wearable technology market was estimated to be worth \$800 million in the year 2015, and is forecasted to grow to \$6 billion by 2018. In research completed by YouGov [30], it was discovered that in October 2014 2.8 million people or 6% of the UK population

owned a wearable technology device. It was also projected that this figure would more than double to 6.1 million by September 2015. Of these wearable technologies, the most popular category was fitness devices, with 3% of UK citizens owning a fitness band. During a Wearable Future Report conducted by PriceWaterhouseCoopers [31], it was revealed that 21% of Americans of adult age owned a wearable device, but only 10% of these individuals claimed to use their gadget daily.

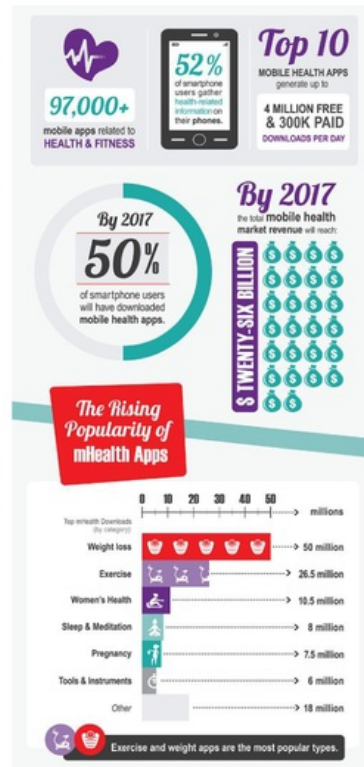


Figure 11: Mobile Health Infographic (GreatCall)

2.3.2.5 Mobile Health and Wearable Technology Advancements and Failures

The utilisation of mHealth is becoming increasingly popular for medical purposes, in particular in developing countries where access to quality healthcare is sparse. However, there are concerns regarding the accuracy and relevancy of mobile health applications. As stated by the FDA (Food and Drug Administration) [32], they retain a degree of responsibility regarding

public health and as of September 2013, have issued the “Mobile Medical Applications Guidance for Industry and Food and Drug Administration Staff”.

This includes information on the oversight of medical mobile applications, with a primary focus on apps that may present a danger to the public if they do not function correctly, and information pertaining to “apps that cause smartphones or other mobile platforms to impact the functionality or performance of traditional medical devices.” [33]

As previously mentioned, the adaption of wearable technology has surpassed wearable bands and has now entered the clothing market, where fitness gear with digital sensory integration is available. However, wearable technology is also being adopted by the healthcare industry to improve and support care for patients and the general public. For example, Vital Connect [34] has developed a patch with an embedded biosensor that aims to monitor heart rate, breathing, temperature and body position, which is connected to a mobile device via Bluetooth and can detect if a person has fallen or if any unusual activity is identified. With more and more companies entering the market and bringing alternative designs and ideas, wearable technology seems to be increasing in popularity and functionality. However, many believe that is it merely a trend and will not remain a popular or sustainable.

2.4 Global Food Waste Statistics

According to research conducted by Natural Resources Defence Council (NRDC) [35], the average household wastes approximately \$28-43 (£18-28) per month on excess food. As stated

in a report published in 2011 by the United Nations Environment Programme (UNEP) in correlation with the World Resources Institute (WRI) [36], it was estimated that the yearly food waste generated is approximately worth 1 trillion US dollars; amounting to roughly one-third of all food produced worldwide and weighing around 1.3 billion



Figure 12: Annual Global Food Waste Statistics (Chicago Council)

tonnes per annum. A press statement was released in 2013 by

the British Institution of Mechanical Engineers (IME) [37], stating that the revised food waste

generated worldwide had risen to 50% each year. This undoubtedly has a devastating effect on our environment, and leads to a “major squandering of resources, including water, land, energy, labour and capital and needlessly produces greenhouse gas emissions, contributing to global warming and climate change.” [38]

2.4.1 Organisational Management and Solution Determination

Many companies and charities work with the goal of reducing food waste using various methods; some aim to reduce the amount of food generated initially, whereas others, mainly charities, aim to redistribute the surplus food to others in need. Both methods effectively tackle food waste, though the greater part of the worldwide population still continue to purchase more than necessary, which in turn becomes wasted. Several campaigns are currently in place to raise awareness of food waste, including World Food Day, occurring annually on October 16th and World Environment Day, running June 5th.

2.4.2 Management Successes and Failures

In 2014, Tesco admitted that it had wasted 30,000 tonnes of edible food from its stores that year, resulting in a partnership with FareShare, a food redistribution charity. Following a legislation in France which prohibited supermarkets from disposing of unspoiled food as opposed to distributing the surplus to charities, Labour MP Diane Abbot [39] proposed a similar motion banning UK supermarkets from discarding of items approaching their best-before date.



Figure 13: Intermarché Inglorious Campaign

Intermarché, the third largest grocer in France, introduced an ‘Inglorious Fruits and Vegetables’ campaign, which sold imperfect produce at a 30% discounted price. The campaign was successful, with an average sale of 1.2 tonnes per store during the first two days [40]. Asda has trialled a similar scheme at five UK stores, with the aim to eventually achieve nationwide integration.

Marks & Spencer has committed to reducing in-store food waste by one-fifth by 2020, and has created a partnership with Neighbourly, an app that connects community projects to businesses. One M&S store in Bristol redistributed 4 tonnes of food waste to local charities over six months, aiding in proving the success of the scheme, which in turn caused M&S to incorporate a nationwide expansion which should be complete by spring 2016.

However, WRAP (Waste Resources Action Programme) [41] announced that with an estimation of 15 million tonnes of annual food waste within the UK alone, only 1% is a result of stores; over half of the total waste comes from food disposal within households. Therefore, more effort should be administered to encourage individuals to revise their food purchasing habits and improve meal planning to achieve a reduction in waste.

2.5 Review of technical solutions: IOS

2.5.1 History and Market Penetration

There are over 40,000 health-related apps available for download at the App Store, with more being added daily. Apple announced in January 2015 that they had sold 1 billion iOS devices, meaning that over 1 billion users now have access to the Apple App store. Their app store revenue trumps Google Play by 80%, due to a strong market-hold in China [42]. As of January 2016, the statistics showed that the average price of apps for iOS was 1.16 US dollars and over 90% of the apps available were free or freemium types.

In order to determine the average development time for an app, Kinvey [43], a digital business platform, collaborated with AYTM to survey 100 developers. They discovered that the average development time amounted to an average of 18 weeks, or four and a half months.

2.5.2 Development Platforms Considered

With many different tools available to aid in development, creating an app is now becoming easier and easier. Programming languages are becoming less complicated and more intuitive, allowing people with less experience and expertise to write code. Apple have launched a modern programming language named Swift, with a concise but expressive syntax, allowing for eased development. It works side by side with Objective-C, another programming language used to write code for iOS.

Xcode is an integrated development environment (IDE) which contains a collection of software development tools, designed for iOS and OS X, which supports C, C++, Objective-C, Objective-C++, Java, AppleScript, Python, Ruby, Rez and Swift. Xcode provides developers with a consolidated system for user interface design, coding, debugging and testing. However, it is available on a mac system alone.

Visual Studio accommodates cross-platform development, allowing users to build apps for iOS as well as Android and Windows devices, using C# and the .NET Framework, HTML and JavaScript, or C++. However, a Mac is needed to compile IPA files, an iDevice application archive file to store the app, and to deploy the app to a device.

Simpler tools which require minimal or no coding include Appy Pie, with a drag and drop feature to reduce development time and effort. Similarly, PhoneGap is a free and open source framework that can be deployed across multiple mobile platforms, with pre-defined standardized components for easier development.

2.5.3 Ease of Development

Apple offers a series of incremental lessons online, in particular for the innovative Swift programming language. Aside from software-dedicated companies offering guides and tutorials, there are a range of blogs, articles and videos that provide users with the ability to receive aid in development. Many virtual communities are also dedicated to assisting each other with any programming issues and solutions. As previously stated, platforms such as Appy Pie and PhoneGap require minimum software development skills or knowledge of programming languages.

2.5.4 Adaptability

With iOS, an app may be updated frequently, with the ability to launch updated versions in an incremental manner. However, depending on the development platform being used, adapting or altering the app may range from easy to difficult.

2.5.5 Hardware Devices

There are three major iOS hardware devices that need to be considered in the development process. A range of iPhones mean a range of screen sizes and resolutions, and so it is imperative that the application is displayed correctly on each device. The iPod lines also need to be considered in the development stage. The iPads have sufficiently larger screens, with several models differing in size and screen resolution. When using a software development tool, consideration for the entire range of iOS enabled devices should be established.

2.6 Review of technical solutions: Android

2.6.1 History and Market Penetration

The Android platform now has approximately sixty various companies producing mobiles, with Google Play hitting 1 billion active monthly users [44]. There are more than 60,000 health-related apps available for download at the app store, with over 1.5 billion app downloads per month. Google Play outweighs Apple's App Store with 90% more worldwide app downloads and holds 80% of the market share for mobile phones, connecting to more individuals globally

than any other app store. With the inclusion of free apps, the overall average cost of apps on Google Play amounted to a mere 0.06 US dollars. According to a cripsycodes infographic [45], the average development time for an Android app amounts to 18 weeks, matching the estimation for the development period of an iOS app.

2.6.2 Development Platforms Considered

Android Studio is the official integrated development environment for developing apps for the Android Platform. It offers code templates in order to aid users in building common features seen in an app, and allows for multi-screen app development, so that a user may develop for a range of devices, including phones, tablets, android wear, android TV and Google Glass.

Visual Studio is a strong contender for app development, with multiple programming languages supported; in particular JavaScript would be most useful for producing an android app using Visual Studio [46]. The Visual Studio Emulator for Android allows for improved testing across various Application Program Interfaces (API) levels through the use of virtual machines, to enable the app to be visually and systematically compatible with the most popular hardware on the market, running on the latest Android versions.

Google App Inventor is a rapid application development tool developed by Google, allowing users to create mobile applications for Android. It has a graphic interface that incorporates a drag-and-drop concept to allow for ease of development, and uses java programming language. It is an online tool that requires no download, with the option to set up a device to the app being built for a live view.

As with iOS, Appy Pie and PhoneGap also allow for Android development, with its range of pre-defined drag-and-drop tools allowing for rapid and simplified development. However, the Appy Pie app is a HTML5 based hybrid app, and although free to an extent, lacks in fundamental features such as app editing and removing Appy Pie's branding. In order to gain access to all features, a subscription must be set up with a monthly fee of \$33.

2.6.3 Ease of Development

As with Apple, Android offers an online training program comprised of guides and lessons [47] to deal with various processes involved in Android app development. This ranges from the initial setup, to adding multimedia, to distributing and monetizing the app using Google Play. Google also provides a free online course, with the approximation of 10 weeks course length at 6 hours per week, involving instruction videos for visual learning. There are various code-level support community resources available for Android, such as Android Developers and Android

Development Community. Various additional communities and forums also provide support for developers.

2.6.4 Adaptability

Google Play allows for Alpha and Beta deployment so that any issues or alterations may be noted and fixed before a public release. App updates are also supported so that new functions, features or bug fixes can be added. As with iOS apps, the complexity of adapting the application depends on the development platform used to initially create the app.

2.6.5 Hardware Devices

Any device with access to Google Play may be a potential customer for an app created for Android devices. As there are so many varied devices and platform versions for Google Play customers, it is important to accommodate multiple devices and platform versions during the development process by ensuring the app is suited to run on various screen sizes and resolutions. During the launch phase, this platform and screen compatibility range will determine who is able to view the app. The more users able to download the app, the larger the customer market will be.

2.7 Review of Literature Conclusion

During the current market evaluation, it was discovered that other apps are available that aim to combat the same or similar issues as the proposed application. The most direct competitor was Love Food Hate Waste, tackling food waste reduction and meal planning. However, one feature not included is the ability to track the nutritional intake of a user, a vital feature of the proposed application which will aim to combat unhealthy eating habits of the target market.

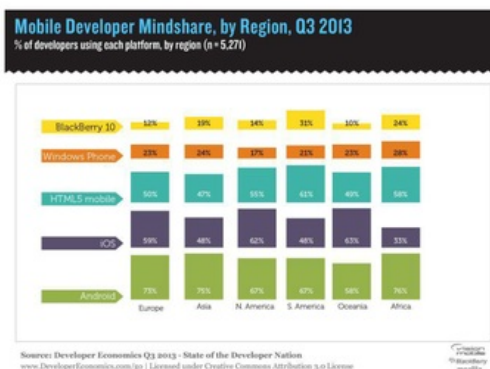


Figure 14: OS App Development Market Share (2013)

The platform chosen for development is Android, due to their 80% market share and larger quantity of health-related apps available on Google Play, signalling a clear market. As Google Play has exceeded Apple's App Store monthly downloads by 90%, it is clear that there is a larger target audience for the development on an

Android app.

3.0 The Overarching Aim and Objectives

3.1 Aims

The overarching purpose of the investigation is to develop an android mobile application for food input and recipe generation to support food waste reduction and tracking nutrition to aid in developing and maintaining a healthy lifestyle.

3.2 Objectives

There are several objectives that need to be considered and satisfied in order to conduct and complete a successful generation of the proposed application. Below is a condensed list of 8 fundamental objectives to be completed:

- 1) Conduct a thorough analysis of any competition and related technologies of the proposed application.
- 2) Complete an evaluation of mobile application development methods, using secondary resources for information.
- 3) Perform a detailed analysis of the application requirements, with the consideration of a survey for feature determination and the completion of a structured interview for qualitative information on any additional requirements.
- 4) Determine the core features with a prioritisation for the most necessary feature implementations by referring to the requirements analysis and with consideration for the application design and an appropriate user interface.
- 5) Complete the development of the application, ensuring all necessary features have been implemented.
- 6) Execute user testing and analysis to determine the functionality of the application.
- 7) Complete an evaluation of the finalised application, detailing any findings discovered during the process.
- 8) Note any future recommendations for the project and for the improvement of the application.

4.0 Aims and Objectives Satisfaction Determination

Once the development cycle is complete, the objectives can be analysed and re-evaluated if they are not entirely satisfied. Through the use of various levels of testing, with the inclusion of beta testing, any errors in the program can be reviewed, in conjunction with any additional features that may need to be included. During the final evaluation stage where any findings will be noted, the objectives can be referred to once more. At this stage any aims or objectives that are not met will need to be noted and included in the future recommendations section.

5.0 Research Methodology

Within this section the intended research methodology selected for this project will be noted and discussed, identifying various research sub-methods to determine the most effective way to gather information pertaining to the research topic. A research methodology figure, the research onion, will be referred to in order to determine which approach is most suited to the subject at hand and justification of choice will be supplemented. All advantages and disadvantages of the selected research approaches will be noted and analysed/reviewed.

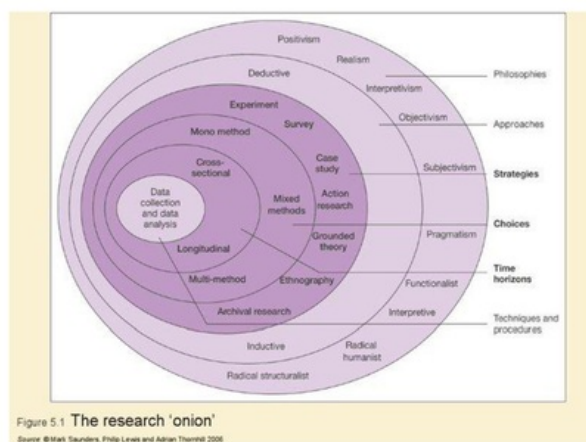


Figure 15: Research Onion

5.1 Data Accumulation

The kind of research used to collect information pertaining to the research topic will be primary and secondary research. Secondary research will detail the findings of other researchers and the reason for their studies, whereas primary research is based on the collective elementary findings obtained by the researcher to direct at the issue being deliberated. The focus of the secondary research will be to gain information already contributed to the field of study to develop a literature review and determine how the proposed project conforms to other contributions to the field, whereas primary research will

be conducted through the means of questionnaires and interviews, to determine required features for the application.

5.1.1 Secondary Research

Several methods will be employed to acquire useful information to develop a literature review. As suggested by Kothari in Research Methodology: Methods and Techniques, "Secondary data may either be published data or unpublished data." Whilst referring to published data, we may consider "technical and trade journals, books, magazines and newspapers, reports and publications of various associations connected with business and industry, and reports prepared by research scholars, universities, economists etc. in different fields." The primary examples of research that will be examined will include scholarly articles and books, in conjunction with alternative technologies and applications that inhere with the proposed project. Business-related books will be referred to in order to review the most appropriate methods to adopt to the branding of the product, whereas software development books will be utilized to aid in the development of the application.

5.1.2 Primary Research

The primary research will be conducted to identify desired user requirements, in conjunction with acknowledging key application requirements. Various methods will be adopted to gather this information, including the circulation of questionnaires to a diverse audience and the conduction of specified interviews. User testing will also be a primary research method, concluding the overall success of the finalised application. This will supply the means to evaluate the success or failure of the project and to determine any recommendations for future improvement.

5.2 Research Philosophy and Approach

The intended research project is the development of a mobile application with user-specified features. As a result, the philosophy paradigm most suited to this type of developmental project is the Positivism Philosophy, producing quantitative information that is highly specific and precise [48]. This may be achieved through various methods as noted by Mackenzie & Knipe, in Research Dilemmas: Paradigms, Methods and Methodology, such as survey research, correlational studies, cross-sectional designs and longitudinal studies [49]. This approach will allow for the collection of quantifiable data; surveying and interviewing will ensure the

determination of generalised features so that the proposed application will be suited to its users.

However, Bahl and Milne [50] state that elements of a phenomenological paradigm may be utilised to support the findings of the quantitative methods:

“While the methods used in positivist research have primarily been quantitative, qualitative methods have been used but only to support the quantitative methods in the development of measures, in the development of theory using grounded theory approaches or clarifying existing quantitative results.”

The phenomenological approach focuses primarily on a humanistic point of view, generating qualitative data using small samples. As defined by Peter Banister, Geoff Bunn, and Erica Burman in *Qualitative Methods in Psychology: A Research Guide* [51]:

“The distinctiveness of phenomenological research specifically is that it offers the ability to explore the diversity and variability of human experience in all its complexity from first-person accounts, with the intention to gain new understandings of how the world is experienced from those actually experiencing the phenomena of interest.”

In short, this data collective method focuses on the holistic accounts of events and experiences of participants, providing information of high quality rather than measurable data. “Phenomenology studies focus on experiences, events and occurrences with disregard or minimum regard for the external and physical reality.” [52] This method will be incorporated into the findings of the project by allowing for the implementation of features and functions as determined by individuals through their qualitative input.

A pluralistic combination of a deductive and inductive viewpoint [53] will be applied as the selected research approach, due to the nature of the research project and the use of both positivistic and phenomenological paradigms. A deductive research approach is linked with a positivistic paradigm, aiming to develop a theory which in turn will determine and allow for testing of a hypothesis. A deductive approach will be suited to the evaluation of the problem in addition to its solution, due to the vast majority of research that has already been conducted

on various aspects of the proposed application. However, the inductive approach will be of value whilst collecting and analysing data, as the proposed application is quite specific in its field of study.

5.3 Research Strategy and Time Horizon

As there are many types of research strategies, there is need to determine which one would be best suited for the proposed project. Each research strategy may be used individually or collectively to form a combined strategy. For this project, elements of various strategies will be utilised, including a case-study strategy, a survey strategy and an experimentation strategy.

A case study approach refers to a single unit of analysis; in this case referring to a technology. A characteristic of the case-study strategy is the multiple use of data collection methods, both qualitative and quantitative [54]. As a result, a case-study strategy will be required to complete the first and second objectives, which are to “conduct a thorough analysis of any competition and related technologies of the proposed application” and to “complete an evaluation of mobile application development methods, using secondary resources for information.”

As defined by Saunders et al [55] in Research Methods for Business Students, the survey strategy “is most frequently used to answer who, what, where, how much and how many questions. It therefore tends to be used for exploratory and descriptive research.” It is therefore useful for gathering a large quantity of standardised data that can easily be manipulated. Therefore, this strategy will allow for the satisfaction of the third and fourth objectives; “to perform a detailed analysis of the application requirements, with the consideration of a survey for feature determination and the completion of a structured interview for qualitative information on any additional requirement” and to “determine the core features with a prioritisation for the most necessary feature implementations by referring to the requirements analysis and with consideration for the application design and an appropriate user interface.”

An experimental strategy examines the “cause-and-effect relationship between two variables” [56], monitoring the change of one variable whilst manipulating the other. This strategy will be applicable due to the nature of the project and the hypothesis that users will reduce their food waste through utilising the application. This strategy will therefore complete the objective, which is to “complete an evaluation of the finalised application, detailing any findings discovered during the process.”

As the time frame for the completion of the project is under one calendar year, the time horizon allocated to this particular project will be cross-sectional. This will be applicable to all research strategies, due to the nature of the research area of study and the need to rapidly determine requirements and build a working application. As defined in *A Dictionary of Research Methodology and Statistics in Applied Linguistics* by Hossein Tavakoli [57];

“A cross-sectional study is one that produces a snapshot of a population at a particular point in time. Details about an event or phenomenon are gathered once, and once only, for each subject or case studied.”

5.4 Data Collection Method Including the Study Population and Sample

As the research project requires a combination of both quantitative and qualitative data to ensure the best result, a concerted method research approach will be applied. Questionnaire surveys will be distributed to a diverse audience in order to ensure the requirements for the application are suited to a wider target market or to determine whether the application should be particularized to a specific audience. The intended age range will be any individual with a smartphone and purchasing their own food produce. Therefore, the population will range between 18 – 65 years old. The sample will be selected by approaching students and staff members within all campuses of Swansea University of Wales, Trinity Saint David to complete the questionnaire. A sample of approximately 20 individuals will be considered, so that the pool is large enough to gain an extensive amount of valuable and quantifiable data, though not so large as to make the recommended application features too diverse to manipulate and implement.

Interviews will be conducted with selected individuals who have more precise knowledge and information on the proposed area of study, so that prioritisation of fundamental user requirements can be given. The information gathered should be of high value and of a qualitative nature. A potential interviewee would be someone who is able to contribute to the suggested health-related features to be implemented, in conjunction with displaying an active interest in global issues such as food waste. An additional interview may be conducted with an individual with a general understanding of mobile application development and feature specifications, in order to ensure the system is functional.

Secondary data will be obtained from reliable sources, including quantitative values of target market estimations pertaining to the proposed application category. All information gathered will enable the accomplishment of definitive functionality, suited to all users.

6.0 Ensuring Reliability and Validity

As with all research, in particular primary research, caution must be taken to ensure the information retrieved is adequate and suitable for the area of study. Statistical reliability will be ensured through the use of sources triangulation, involving referring to three sources to ensure the statistics are accurate and correspond with other sources of the same or similar nature. Noted by Haider [58], qualitative studies “make use of interviews, observation, surveys and company documents for reliability and accuracy of results.” Reliability will be ensured during the data collection stage by using a wide pool of informants in the surveying process, in conjunction with interviewing individuals who possess knowledge on the topic at hand.

Baumgarten [59] states that “validity is a way to ensure quality in measurement, attest expected causalities and allow generalization.” Seliger and Shohamy [60] deduced that there are two kinds of validity; internal and external. Whereas internal validity is jeopardised by confounding variables within the study itself [61] [62], external validity refers to “the degree to which the conclusions in your study would hold for other persons in other places and at other times” [63]. Validity will therefore be achieved by ensuring all data collection methods endeavour to obtain only valid and appropriate data, suited to the field of study.

7.0 Development Methodology

Described by Holland and Crompton [64], a software development methodology “is a framework that is used to structure, plan and control the process of developing an information system.” A software development methodology will be used to aid in the development and completion of the application, with the inclusion of all necessary features [65]. This strategy will aim to satisfy the fifth, sixth and eighth objective, those being to “complete the development of the application, ensuring all necessary features have been implemented”, to “execute user testing and analysis to determine the functionality of the application” and to “note any future recommendations for the project and for the improvement of the application.

7.1 Selected Development Platform

On account of the time constraint of the project, the development of the application may only be completed for one mobile platform. In order to design and develop a mobile application to a satisfactory standard, an adequate amount of time must be allocated to the development process. Therefore, the development of not only one but two applications would require learning how to develop a mobile application using two distinctive programming languages, which would be unfeasible in the amount of time available.

As a result of the research conducted in the literature review, it is determined that Android is the most appropriate mobile platform to develop an application for, due to its mobile market penetration and advantage. Whereas development for an IOS platform would restrict the application to Apple devices only, the Android platform is available and connected to a variety of mobile and tablet products and brands.

7.2 Review of Development Methodologies

To determine the best-suited development methodology for the proposed project, an analysis of available methodologies will be conducted, with the inclusion of a review of the most appropriate models. A final methodology will be selected and a justification will be supplemented.

7.2.1 Classic Methodologies

7.2.1.1 Waterfall

Invented by Winston W. Royce [66], the waterfall is a sequential design methodology, in which the processes are followed preceding the completion of the previous process. Each phase or process must be completed before the next phase can commence. The structural phases are as follows: requirements analysis, design, implementation, verification and maintenance. Basu [67] states

that “the waterfall model was the first process model introduced in the 1970s and is also referred to as a linear-sequential life cycle model.”

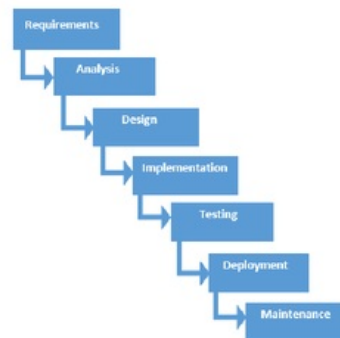


Figure 16: Example Waterfall Methodology

However, in reality the waterfall model was an adaption of a previous process model, the *Nine Phase, Stage-Wise Model* [68]. Deek et al [69] illustrate the Nine-Phase model as “a one-directional, sequential model that was enhanced by the waterfall model through the introduction of bidirectional relations between the successive model stages, corresponding to feedback elements in the development process.” The waterfall methodology is most suited to “low-speed, low-change projects in which there is a very clear and stable definition of the requirements, the product, and the supporting technology.” [70]

7.2.1.2 Spiral

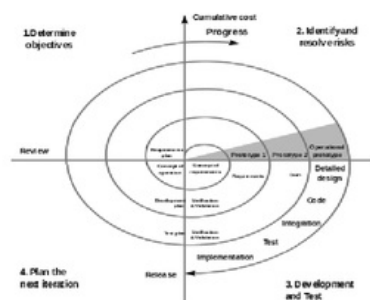


Figure 17: Spiral Methodology Example

Wasson [71] portrays the Spiral Development Model as a methodology in which the “constituent activities, typically requirements analysis, preliminary and detailed design, coding, integration, and testing, are performed iteratively until the software is complete.” Boehm and Hansen [72] expand on the model, explaining that it is a “risk-driven process model generator that is used to guide multi-stakeholder concurrent engineering of software-intensive systems.

In simple terms, this development methodology is customarily preferred for extensive and expensive development projects, combining features of both prototyping and waterfall models.

7.2.1.3 Agile

The agile approach is commonly used in software development, providing the ability to monitor the progress of the project throughout its developmental lifecycle. Bernard et al [73] explain that it helps developers to “respond to unpredictability through incremental, iterative work cadences, known as sprints.” The agile concept was influenced by a number of developers and their methodologies [74]. Prior to the formation of the Manifesto for Agile Software Development (also known as Agile Manifesto), published in 2001, the class of methodologies was originally represented as light-weight methodologies or processes [75]. Alistair Cockburn [76], author of *Agile Software Development: The Cooperative Game* depicts that there were four original methodologies that had a prominent influence on the agile approach, those being the Scrum, DSDM (Dynamic Systems Development Method), ASD (Adaptive Software Development) and XP (Extreme Programming).

7.2.2 Agile Methodologies

7.2.2.1 Rapid Application Development (RAD) Methodology

Introduced by James Martin [77] during the 1980’s as a formalised adaption of Rapid Iterative Production Prototyping (RIPP), the RAD methodology combined:

“...all necessary elements of what the market desperately needed from the software development sector: to adhere to the drastically increased demands for fast and cheap software applications, especially after the introduction of the internet.”

The methodology sanctions swift, efficient and systematic software development by combining characteristics from both prototyping and Joint Application Design (JAD), allowing for the accelerated development of an application at high quality and low cost [78]. Due to the establishment of minimal initial planning, RAD is most effective when requirements are prone to change as they may evolve or alter during the development process. David and Yen [79] explain that the RAD lifecycle has four definitive phases; requirements planning, user design, construction and cutover. During the user design phase, the developer should scrutinise the requirements before producing a prototype representing all system processes, inputs and outputs, of which the user may then review and rectify to produce and approve a working prototype [80]. This prototype is then employed as the basis of the final application and is augmented during the construction phase to produce the finalised product [81].

7.2.2.2 Extreme Programming (XP) Methodology

A method coined by Beck [82] in 1996 and initially published in 1999, Extreme Programming (XP) is a concept that was developed with the aid of Cunningham and Jeffries and specifies the best practices for planning, designing and producing

software by emphasising face-to-face dual collaboration.

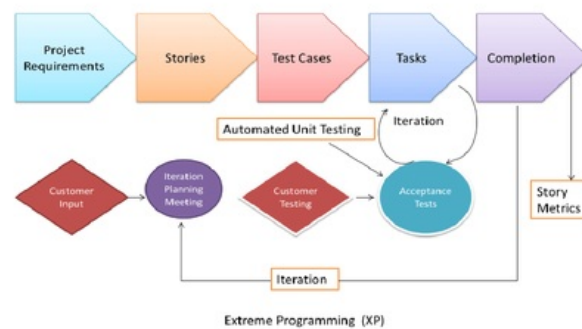


Figure 18: XP Methodology Workflow

Beck [83] states that through the incorporation of XP, “software can be developed at a lower cost, with fewer defects, with higher productivity, and with a much higher return on investment.”

Farid [84] outlines the primary practices utilised in XP as “iterative development, pair programming, collective code ownership, frequent integration, onsite customer input, unit testing and refactoring.”

During the coding phase, the developer collaborates dually with another developer on a single workstation, incorporating ‘pair programming’, in which each plays a role; one developer becomes *the driver*, whilst the other becomes *the navigator* [85]. The driver remains in control of the workstation whilst the navigator provides input and notes defects. The roles are methodically reversed.

The XP methodology supports the fluidity of user-based requirement alterations, and allows for these contingencies by establishing frequent communication with the customer and collaboration with other programmers. Requirements are typically determined via user stories, in which the customer describes what they require the system to achieve whilst placing prioritisation on the most necessary features. Baumeister [86] expands that these user stories are then translated by an analyst into use-cases to generate mock-ups. The developer will customarily write the unit tests before producing any code, which aids in the overall improvement of systematic cohesion and provides predefined regression tests [87]. XP utilises refactoring, which is the concept of altering and augmenting existing code to improve its design, aiding in the accelerated delivery of software [88]. Unit testing is heavily relied upon during this phase, so as to ensure the code remains functional.

7.2.2.3 Scrum Methodology

During the annual Object-Oriented Programming, Systems, Languages & Applications (OOPSLA) conference in 1995, Sutherland and Schwaber [89] presented a paper on the Scrum Methodology, introducing the approach to the public. This methodology employs a definitive focus on the rapid production of a product, allowing for flexibility in requirements. First defined as "a flexible, holistic product development strategy where a development team works as a unit to reach a common goal" [90], the methodology is most suited to a team of developers, those of which adopt daily face-to-face communication.

There are three fundamental roles in the scrum framework; the product owner, the development team and the Scrum Master. The product owner functions as a stakeholder representative by negotiating priorities, defining schedules and

organising milestones in conjunction with corroborating this information to any key stakeholders [91]. The development team handle all analysis, design, development, testing and documentation of the product, through cross-functional collaboration of multiple overlapping phases. The scrum master essentially plays the role of the development team director, ensuring all processes of the development are conducted correctly and are delivered within the specified time frame. The scrum master is also accountable for negotiating and corresponding with the product owner to establish the requirements and maintain a backlog of all processes [92].

Scrum endorses iterative and incremental approaches through the inclusion of spirally-structured development cycles named 'sprints' [83]. These sprints are established preceding their conduction and are confined to a definitive time frame, most commonly two weeks. The commencement of each sprint involves a sprint planning event, in which the tasks to be completed are defined with their corresponding time-frame estimations [94]. Towards the end of the sprint, a review is completed to determine the progress completed during the event, which is then relayed to the stakeholders. Watkins [95] suggests that "retrospectives should be held at the end of each iteration to ensure that the lessons learned during that iteration are discussed and recorded" so that they may suitably applied during the subsequent sprint.

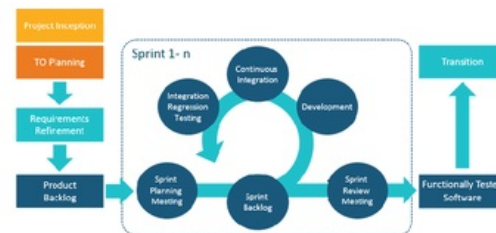


Figure 19: Scrum Methodology Overview

Succeeding the review of each sprint, an adequate amount of product development is usually seen, which is then consulted with to determine the need for another sprint. Once the product is seemingly complete and the stakeholders are satisfied, the sprint cycles will conclude.

7.3 Intended Development Methodology with Justification

Due to the structure of the project and the pre-defined requirements of the application, a suited methodology would be one that correlates with a set of particular requirements and strives to ensure the accomplishment of their integration. As RAD accentuates the necessity of readjusting requirements during the developmental lifecycle [96], it would not be completely suited to this project.

Resultantly, a pluralistic amalgamation of elements from both Scrum and Extreme Programming practices will be incorporated, although not all elements from both methodologies are suited. For example, the simultaneous dual-development factor of XP will be incompatible with this project due to it being an individually completed task.

One component of the XP methodology that will be adopted is the short iterations, due to the cross-sectional time-frame of the project. As the Scrum methodology allows for the completion of lower priority tasks during an iteration or sprint if it coheres to the scheduling or coding procedures being completed at the time (as opposed to XP's strict prioritization scheduling) this method will be employed so that the MoSCoW (Must have, Should have, Could have, and Won't have) prioritization technique may be supported.

Due to the supervisory requirement of the project, the scrum master role may be adopted by a selected project supervisor as they oversee the completion of project-related tasks during predetermined weekly conferences, in conjunction with maintaining a backlog of processes in the form of project meeting documents.

7.3.1 Overcoming Disadvantages of the Approach

A slight disadvantage of the proposed approach is the uncertainty of which elements of each methodology to combine in order to complete the development of the project most effectively. For example, the incorporation of short iterations may provide an insufficient amount of time between each development process, in particular when other work commitments arise during the proposed time scale. To overcome any disadvantages, a set amount of time must be dedicated weekly to the project in order to remain on schedule with the allowance for any complications that may arise.

7.3.2 Taking Advantage of any Strengths of the Approach

Schwaber and Mar [97] communicate the tactical method in which XP and Scrum are combined in industry:

“The phrase Scrum process with XP engineering practices refers to the use of Scrum to manage the steps taken to develop software, in conjunction with the use of XP to ensure the quality of the software.”

To put it simply, due to the fact that both of the proposed methodologies complement each other with their similar features, their weaknesses are resolved by the strengths of the opposing methodology. Consequently, the incorporation of elements of both agile practices in tandem will engender a bespoke developmental technique specified to the proposed project, enabling an augmented methodology.

8.0 Android Application Design

8.1 Introduction

Conducive to creating an effective and consistent mobile application, initial requirements and system designs were specified. This section will highlight and justify any design decisions made regarding the project, in conjunction with the provision of structural and behavioural diagrams with the supplementation of an explanation of each. The development environment and languages used to create the mobile application will be noted, as will any data structures. The name assigned to the application was “Food Glorious Food”, which eulogizes the fundamental element of the system.

8.2 Requirements Specification

During the research methodology section, a data accumulation strategy was appropriated. Information was obtained during the analysis of the survey and interview responses, which was then used in conjunction with findings from the literature review to form the basis of the requirements. These requirements were then ordered based on the MoSCoW prioritisation criteria. The complete list of requirements can be found in Appendix A.

8.3 Development Environment and Language

As previously established during the research methodology section, there are a wide variety of development platforms available to create Android applications. It was deduced that the most appropriate integrated development environment for android app generation is Android Studio, due to its rich layout editor which allows for the drag-and-drop of user interface components and the ability to support various screen sizes and android versions.

Applications generated in Android Studio are written in Java, which is ideal due to the vast majority of online communities devoted to the Java language. However, each activity screen also uses Extensible Markup Language (XML) to specify formatting, which may take some time to adjust to. Android Studio also works alongside Structured Query Language (SQL) to form the database, table formats and database queries.

8.4 Database Design

The initial focus of the design work was to form a database structure in order to allow all of the component tables to interact in a functional manner. An Entity Relationship Diagram was formulated using Star UML, an open-source modelling tool defined by Weyns et al [98] as “a UML modelling framework supporting the MDA (Model Driven Architecture) approach”.

The diagram illustrates how each dimensional table interacts with the others, through the definition of relationships such as the one seen between the ‘Food Items’ and ‘Recipe Ingredients’ tables. Primary and foreign keys were labelled and composite keys were defined where tables contain more than one primary key. Section 8.4.3 demonstrates the SQL query used to create these tables in the application.

8.4.1 Entity Relationship Diagram

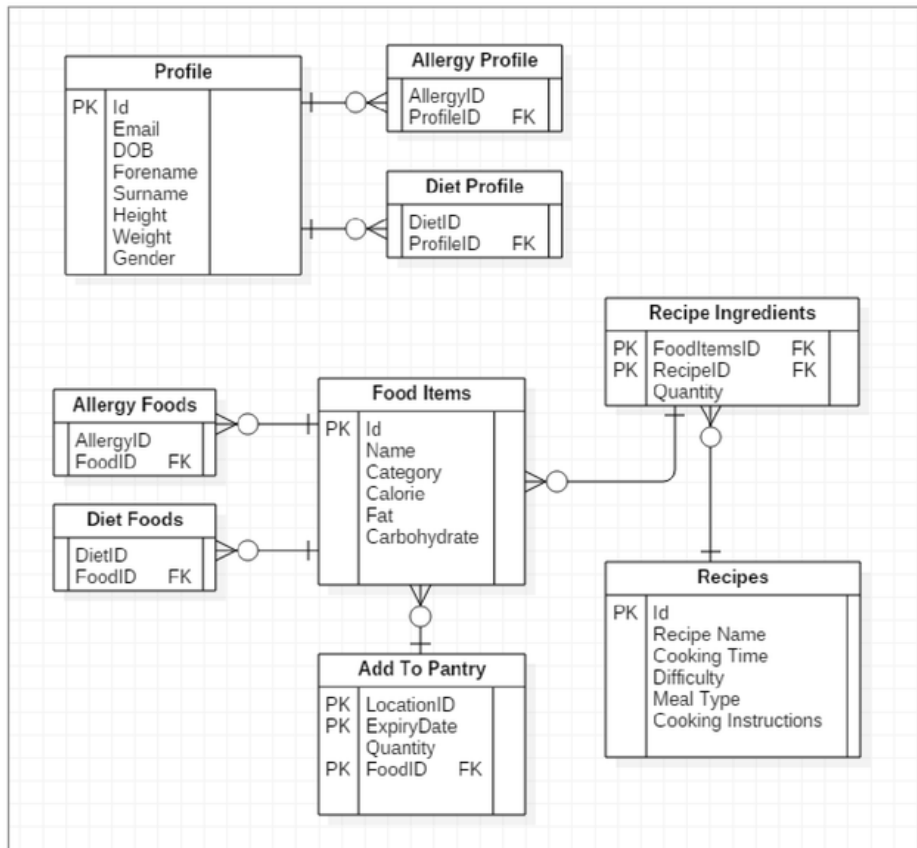


Figure 20: ERD for FGF

8.5 Use Case Diagrams

8.5.1 Top Level Use Case

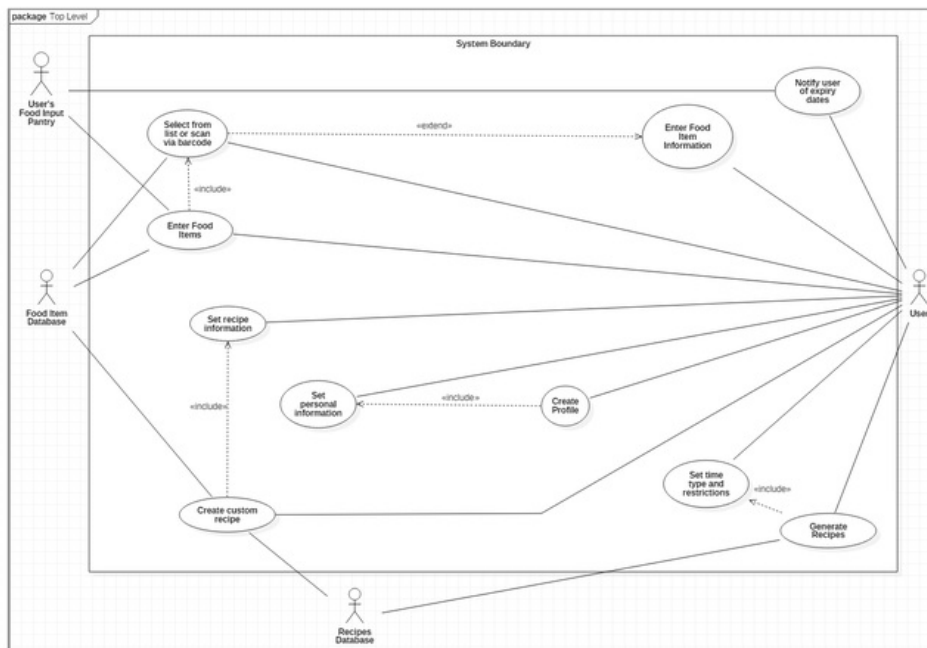


Figure 21: Top-Level Use Case for FGF

A use case is a high-level diagram intended to illustrate the functionality of a system in a visually comprehensible format. Typically used in industry to explicate processes to less-technical participants, B'Far [99] described use cases as diagrams "designed to model a system's behaviour without delving into its inner-workings".

Figure 21 illustrates the top-level functionality in a brief format, depicting actor interaction with possible scenarios that may occur within the system. Actors do not necessarily need to be humans – rather, they represent a person or an external system involved in the application.

For example, in order to generate a custom recipe, the user must interact with the recipes database; a collection of data external to the current system process. This procedure includes setting recipe information, which will be investigated in more detail with another use case.

8.5.2 Generate Recipes Use Case

The diagram below focuses on the “generate recipes” process in more detail. When the user decides to generate a recipe, they can set a time, meal type, restrictions and a maximum calorie, carbohydrate or fat intake, so as to filter the recipes. Alternatively, all recipes available to the user may be displayed with no filtration. The process then interacts with the user’s virtual pantry, removing items used by the recipe. In order to generate a custom recipe, the user will interact with the recipe database, where they are responsible for allocating a name, difficulty, meal course, cooking time and a cooking method. This process includes selecting ingredients to be used, which then interacts with the food item database. To view additional Use Case Diagrams, see Appendix B.

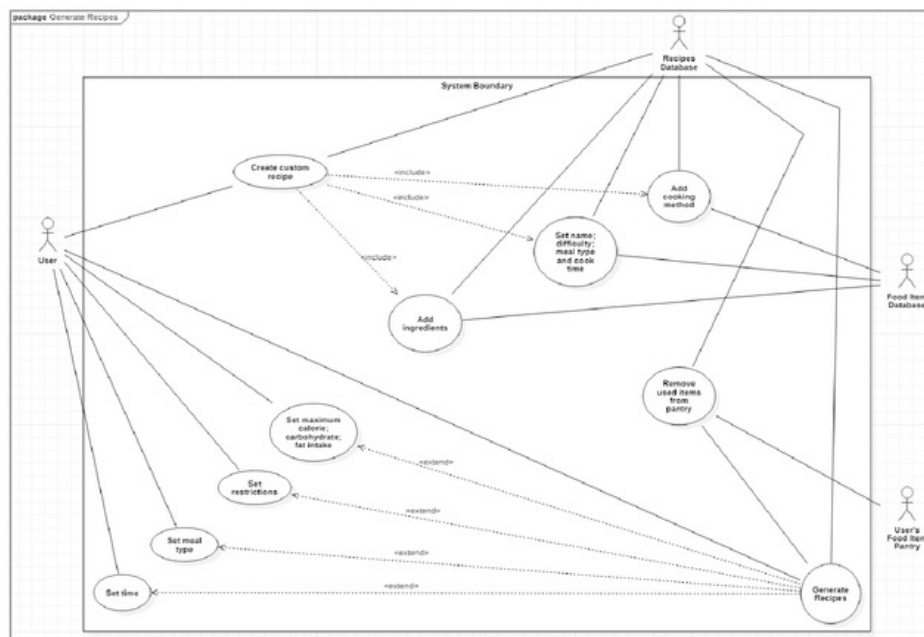


Figure 22: Generate Recipes Use Case Diagram

Fowler [100] believes that class diagrams are “subject to the greatest range of modelling concepts”, though most are generated with only essential elements. Figure 23 illustrates the method in which attributes and methods are assigned and utilised within a class, in conjunction with if and how each class interacts with another. The visibility of attributes and methods are denoted by a symbol preceding them. Three variations of visibility are utilised in the class diagram below; “+” signifies a public attribute or method, “-” represents a private method and “#” denotes a protected method.

[illegible]

Figure 23: Application Class Diagram

are visible to all elements with access to the class.

8.7 Navigation and Main Menu

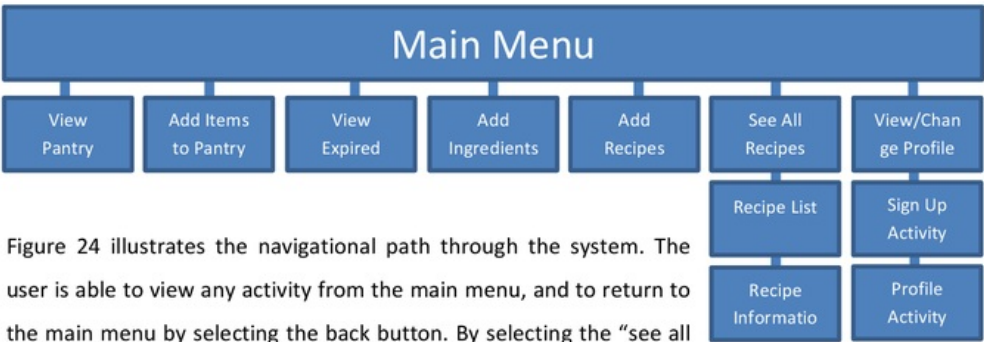


Figure 24: Main Menu Application Navigation Structure

Figure 24 illustrates the navigational path through the system. The user is able to view any activity from the main menu, and to return to the main menu by selecting the back button. By selecting the “see all recipes” button, the user is directed to an activity that generates a list of all recipes available. The user may then click on a desired recipe to view further information.

From there, the user is able to “create recipe”. Should the user wish to alter their profile settings, they must initially sign in by entering their password. They are then directed to the profile activity from which they are able to return to the main menu.

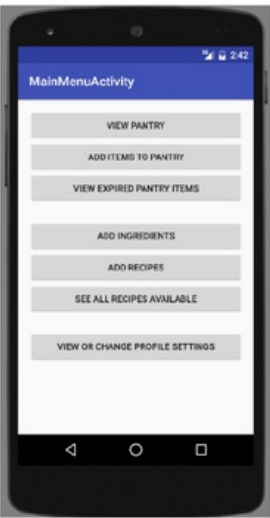


Figure 25: FGF Main Menu Activity

Figure 25 displays the main menu loaded onto the AVD emulator. Buttons are grouped together when performing similar tasks; all pantry actions are grouped together, as are all ingredient and recipe actions. The profile settings button remains separated from any grouping.

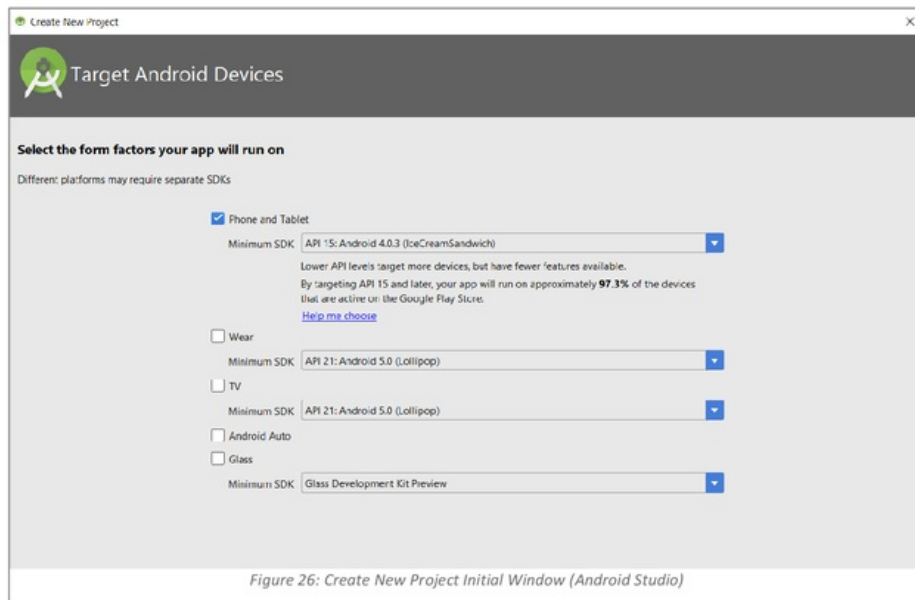
9.0 Implementation

9.1 Introduction

During this section, the implementation of the proposed application, Food Glorious Food, will be discussed, listing any add-ons used to augment the functionality. Here are the stages completed to actualise the application, following the designs defined in the previous section. A guide on how each element of the application was constructed will be included, followed by any significant tools or techniques applied during code generation. Images of the final product will be enclosed with a brief description to supplement the purpose of the implementation decisions. The completed prototype is available on the CD ROM located at Appendix C.

9.2 Operating System Version

When developing an application to be released to the public, it is imperative that actions are taken to attempt to direct the product at as many individuals as possible. Therefore, when selecting the Software Development Kit (SDK) platform suitable to target the Application Program Interface (API) available to the largest percentage of devices, Android 4.0.3 (IceCreamSandwich) was the most fitting option. This is due to the aforementioned operating system version being compatible with 97.3% of the devices currently active on the Google Play Store. Figure 26 displays the initial screen displayed when creating an application using Android

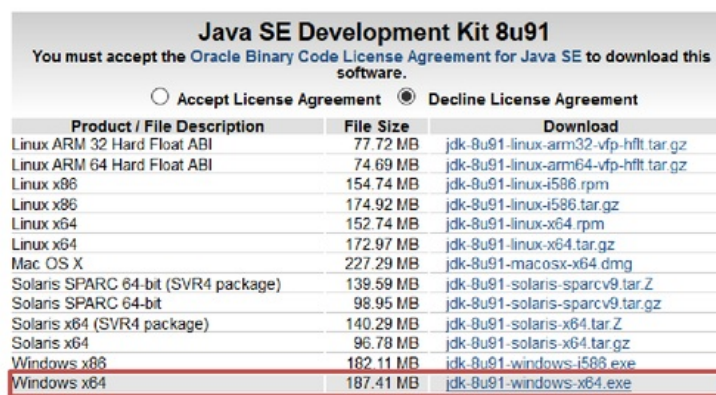


Studio.

9.3 Extensions, Plugins and Themes

Android Studio allows for Virtual Device Emulator plugins to run and debug an application virtually, as opposed to connecting to a physical device. The emulator used for the development process was the NEXUS 5 API 23 x86, due to its average screen size and resolution in comparison with other devices.

The fundamental plugin required to support Android Studio is the Java Development Kit 8 plugin. This was downloaded from Oracle's website as a binary file and then run using an installer. Figure 2.7 displays the version downloaded. In order to view all other plugins used, please refer to Appendix D.



Product / File Description	File Size	Download
Linux ARM 32 Hard Float ABI	77.72 MB	jdk-8u91-linux-arm32-vfp-hflt.tar.gz
Linux ARM 64 Hard Float ABI	74.69 MB	jdk-8u91-linux-arm64-vfp-hflt.tar.gz
Linux x86	154.74 MB	jdk-8u91-linux-i586.rpm
Linux x86	174.92 MB	jdk-8u91-linux-i586.tar.gz
Linux x64	152.74 MB	jdk-8u91-linux-x64.rpm
Linux x64	172.97 MB	jdk-8u91-linux-x64.tar.gz
Mac OS X	227.29 MB	jdk-8u91-macosx-x64.dmg
Solaris SPARC 64-bit (SVR4 package)	139.59 MB	jdk-8u91-solaris-sparcv9.tar.Z
Solaris SPARC 64-bit	98.95 MB	jdk-8u91-solaris-sparcv9.tar.gz
Solaris x64 (SVR4 package)	140.29 MB	jdk-8u91-solaris-x64.tar.Z
Solaris x64	96.78 MB	jdk-8u91-solaris-x64.tar.gz
Windows x86	182.11 MB	jdk-8u91-windows-i586.exe
Windows x64	187.41 MB	jdk-8u91-windows-x64.exe

Figure 27: Java Development Kit 8 Plugin Products

The installation and creation of the AVD was fairly simple and was completed by launching the AVD Manager in Android Studio, under tools. The AVD was then configured to allow for pc keyboard input to speed up the development process. Additional RAM was allocated to the emulator in order to accelerate processing speed. H.A.X.M (Hardware Accelerated Execution Manager) was another plugin installed in order to execute the application more rapidly, so as to reduce delays in debugging. This was downloaded by opening the Android SDK Manager and selecting the "Intel x86 Emulator Accelerator (HAXM installer)" followed by selecting "install package". Ensuing this, the package had to then be installed by locating the installation folder on the computer drive and running the installer.

As opposed to installing a new theme and incorporating it throughout the system, the standard Android theme was retained due to its familiar appearance and its visual consistency. The theme is clean and crisp, inciting a more professional presence.

9.4 Building the Application

Following the selection of an API, the initial activity was generated. When adding an activity, Android Studio creates the activity XML and the content XML layout, in conjunction with a corresponding java class. This method was used for each layout activity.

9.4.1 Building an Activity

Figure 28 and 29 depict the screens displayed when setting up the initial activity. To begin, a “Blank Activity” was selected and named “MainActivity”. As Android Studio is object-oriented, generating elements is a simple process. A palette is displayed with a series of layouts, widgets, text fields and



Figure 28: Select Activity Page

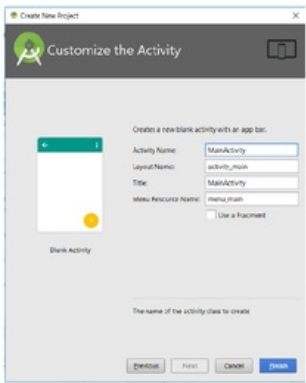


Figure 29: Configure Activity Setup

containers, among other features.

To create a sign up activity, a selection of text fields was added to the “content_main.xml” layout. A button was then added to the bottom of the file, and changes were made to the properties of the text fields. For example, the “android:text=Name” was changed to “android:hint=Forename” to remove the fixed text so that it will disappear once a user begins

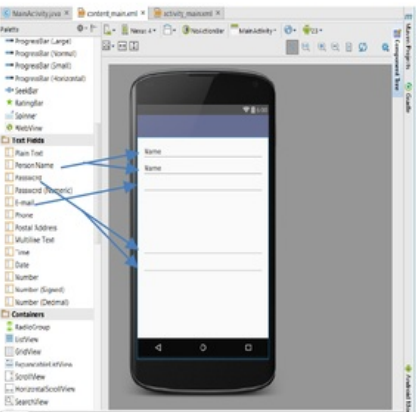


Figure 31: Android Studio App SignUp Setup



Figure 30: XML code for activity object layout

to enter data.

9.4.2 Validating User Input

Validation was completed on the password fields in the corresponding java class; for the sign up button to work, both passwords had to match. Additional validation was placed on each text field to ensure the data entered conformed to the data required.

```
if (Password.matches("")) {
    Toast.makeText(getApplicationContext(),
        "Please enter a password",
        Toast.LENGTH_LONG).show();
    return;
}

if (!CheckPassword.matches (Password))
{
    Toast.makeText(getApplicationContext(),
        "Please make sure both passwords
        match", Toast.LENGTH_LONG).show();
    return;
}
```

Figure 32: Password Validation Java Code

9.4.3 Creating the Database and Tables

To store data, Android Studio extends the SQLite database management system. This feature was used to generate the database used to store all information pertinent to the application. Tables were completed using public string finals. Figure 33 displays the code used to generate the application database and a profile table within said database, used to store data relating to the user.

The profile table created includes an auto-incremental primary key, referred to as Id. The email, forename, surname, password and date of birth as set as texts, whereas the height is set as an integer and the weight as an integer. The gender data type is a Boolean; male = 0,

```
public static final int DATABASE_VERSION = 1;
public static final String DATABASE_NAME = "FoodGloriousFoodDB.db";
public static final String TABLE_CREATE = "CREATE TABLE IF NOT EXISTS
Profile (Id INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL, Email TEXT,
Forename TEXT, Surname TEXT, Password TEXT, DOB TEXT, Height INTEGER,
Weight NUMERIC, Gender BOOLEAN)";
```

Figure 33: Database Initialization and Table Creation SQLite Queries

female = 1.

9.4.4 Placing Data into A Table

```
Button saveButton = (Button) findViewById(R.id.saveIngredient);
saveButton.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        InsertFoodToDatabase();
    }
});

public void InsertFoodToDatabase() {

    TextView FoodNameView = (TextView) findViewById(R.id.autoFoodName);
    Spinner CategoryView = (Spinner) findViewById(R.id.editCategory);
    TextView CalView = (TextView) findViewById(R.id.editCal);

    String Name = FoodNameView.getText().toString();
    String Category = CategoryView.getSelectedItem().toString();
    Float Calorie = Float.parseFloat(CalView.getText().toString());

    DatabaseHandler dbh = new DatabaseHandler(this);
    long id = dbh.SetFoodItems (Name, Category, Calorie, Fat,Carbohydrate);
}
```

Figure 34: Inserting Data into FoodItems Table Function

Figure 34 shows the code generated to pass data into the Food Items table. An event is called using an onClickListener, assigned to the “save ingredient” button. The java class then calls the “InsertFoodToDatabase” function, which pulls the data entered by the user into the “autoFoodName” on the activity. This is completed for all other items on the activity. The information is then converted from a textview to a string. The database handler is called and the converted data is passed to the table, using the “SetFoodItems” function created within

```
public long SetFoodItems(String name, String category, Float calorie,
Float fat, Float carbohydrate) {
    SQLiteDatabase db = this.getWritableDatabase();
    ContentValues values = new ContentValues();
    values.put("Name", name);
    values.put("Category", category);
    values.put("Calorie", calorie);
    values.put("Fat", fat);
    values.put("Carbohydrate", carbohydrate);

    long id = db.insert("FoodItems", null, values);
    return id;
}
```

Figure 35: SQL query to place data in FoodItems Table

the Database Handler. Figure 35 displays this “SetFoodItems” function.

9.4.5 Auto-Populating Data from the Database

By calling the database handler and creating a cursor to call all records from a table, an if statement was created to loop through the data and pull a record. This was the case with the profile details; as there is only ever one user with login details stored in the local database, he

```
DatabaseHandler dbh = new DatabaseHandler(this);
Cursor cr = dbh.ObtainSignUpActivityDetails();

cr.moveToFirst();

if(cr.getCount() > 0) {

    TextView DOBView = (TextView) findViewById(R.id.editDOB);
    DOBView.setText(cr.getString(cr.getColumnIndex("DOB")));
    TextView HeightView = (TextView) findViewById(R.id.editHeight);
    HeightView.setText(cr.getString(cr.getColumnIndex("Height")));
    TextView WeightView = (TextView) findViewById(R.id.editWeight);
    WeightView.setText(cr.getString(cr.getColumnIndex("Weight")));
}
```

Figure 36: Cursor to obtain table row

following code was used to pull the user information and place it on the activity.

The code for the corresponding cursor used a string query of the database table “profile” to match the profile id to the value 1. The record was then passed through to a cursor and returned.

```
public Cursor ObtainSignUpActivityDetails() {
    SQLiteDatabase db = this.getReadableDatabase();
    String Query = "SELECT * FROM Profile WHERE Id=1";
    Cursor CR = db.rawQuery(Query, null);
    return CR;
}
```

Figure 37: Database Handler Cursor Code

9.4.6 Passing Data from one Activity to Another

```
Bundle SignUp = new Bundle();
SignUp.putString("EmailView", Email);
SignUp.putString("PasswordView", Password);

Intent intent = new Intent(SignUpActivity.this, Profile.class);
intent.putExtras(SignUp);
startActivity(intent);
```

Figure 38: Bundle to pass sign up details to profile activity

```
Bundle SignUp = null;
SignUp = this.getIntent().getExtras();
String PassedEmail = SignUp.getString("EmailView");
String PassedPassword =
```

Figure 39: Bundle After Passed to Profile

To pass the login data from the sign up activity to the profile activity, a bundle was created. All relevant strings were placed in the bundle and passed to the proceeding event, using an intent. The "SignUp" bundle is retrieved in the Profile class, and all values are converted to strings.

9.5 SQLite Manager Firefox Plugin

To view the database generated through the application functions, an SQLite Manager plugin was added to Firefox. The database was then pulled from the application, using the Android Device Monitor when the emulator was running, by locating the database file in the file explorer tab.

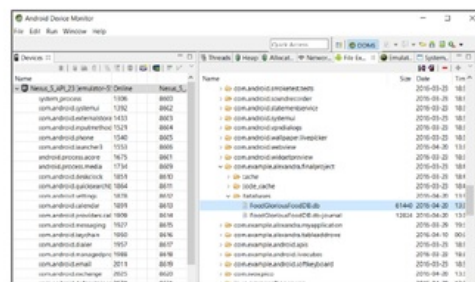


Figure 40: ADM Screen

The file was then pulled and opened using the SQLite Manager, where all records added to the database could be viewed. This was useful during the debugging process to ensure all data was being passed as intended, to the correct table, in the correct format.

TABLE AddToPantry				
FoodID	LocationID	ExpiryDate	Quantity	
1	Fridge	19/3/2016	400	
8	Cupboard	17/3/2016	12	
9	Cupboard	17/3/2016	12	
10	Fridge	15/2/2016	200	
11	Cupboard	19/4/2016	500	
12	Fridge	8/5/2016	3	
13	Fridge	28/4/2016	200	

Figure 41: FoodGloriousFood DB in SQLite Manager

9.6 File Structure

Here all files used in the application are displayed. Figure 42 notes all java classes and enums, whereas figure 43 shows every layout used. Together, the elements work to form the completed application.

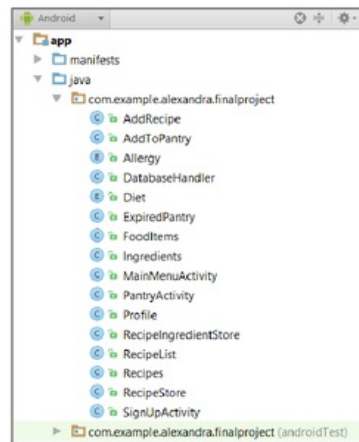


Figure 42: Java Class File Structure of FGF

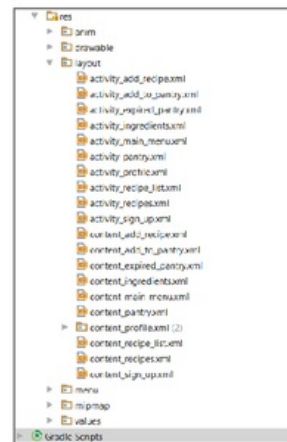


Figure 43: Activity Layout File Structure of FGF

9.7 Utilising the Application

9.7.1 Signing Up

When the application is initially loaded, all fields are blank. The user then enters their personal details in order to create an account. Subject to the validations checks being successful, the user is then directed to the profile activity. Again, the user must fill in all details. The currentBMI, classifications and target weight values are automatically generated from the users' height and weight input and cannot be altered. The user must then select any allergies they have or diets they follow, before selecting the save button.



Figure 44: FGF SignUp Activity



Figure 45: FGF Profile Activity

9.7.2 Adding Ingredients

To add an ingredient to the database, the user must enter a food item, select a category from a drop-down list, set the nutrition information and tick which allergens the item includes. In addition, the user may make note of which diets the food item is

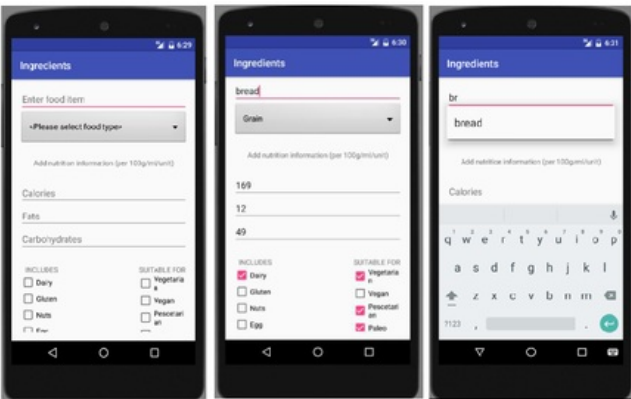


Figure 46: FGF Ingredient Activity

suited to.

Succeeding a successful data entry, the item will be added to the database. When the user attempts to enter the same ingredient again, the food item will appear in the auto complete food item text box. This will then load all corresponding nutrition information.

9.7.3 Adding Items to the Pantry

When adding items to the pantry, the user must enter a food item name, select a type and location, enter a numeric quantity, select an expiry date from a date picker and enter any nutrition information. If the user in attempting to enter an item already previously input using the previous method, the activity will load the item name into the auto text view and fill out the nutrition information. When the “save food item” button is selected, the class will place certain data in the pantry database and certain data into the ingredients database.

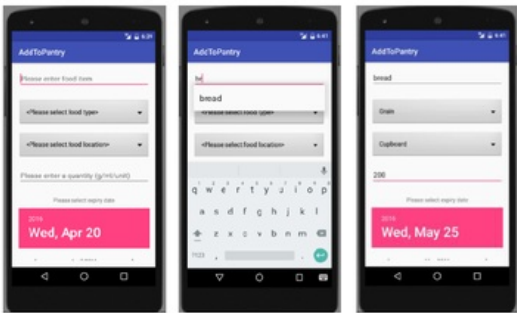


Figure 47: FGF AddToPantry Activity

9.7.4 View Pantry Items and Expired Pantry Items

The pantry items activity displays all items added to the pantry, excluding those that have surpassed their expiry

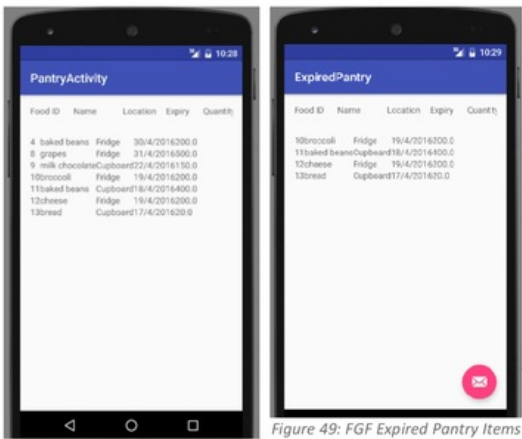


Figure 48: FGF Pantry Activity

Figure 49: FGF Expired Pantry Items Activity

dates. Any items that have moved past the expiry date will be removed from the pantry activity and stored in the expired pantry items. The items are organised by when they were added to the pantry.

9.7.5 Add Recipes and See all Recipes Available

In order to add a recipe, the user must enter a recipe name, cooking time select a difficulty from a 1-5 rating, select a meal type and enter cooking instructions. To link food items to that recipe, the user can enter a food ingredient and a quantity. If the recipe calls for more than one ingredient, the "add" button may be selected to generate a new row.

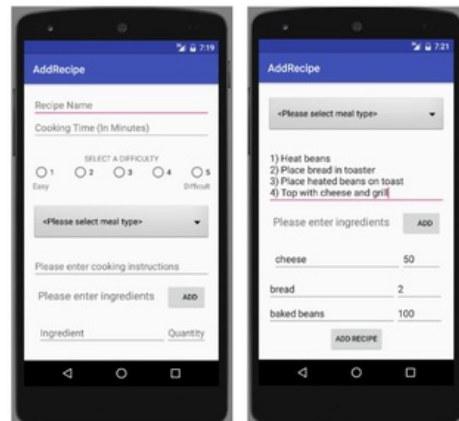


Figure 50: FGF AddRecipe Activity

10.0 Results and Evaluation

10.1 Introduction

This documentation segment will reveal the success of the application based on an accumulation of testing processes. A systematic test plan will be illustrated and conducted with an evaluation noting the success of each class function. User testing and evaluation will be examined, providing feedback from each participant. View all test plans in Appendix E.

10.2 Test Plan

The test plan will contain a series of tests executed on each individual java class and activity used in the application. This plan will denote a test number, the function being tested, the expected and actual outcomes, any issues faced during the testing process and a comment noting the overall success or fail of the test. An evaluation of each class being tested will be delivered, providing additional insight into any issues faced and possible solutions.

10.2.1.1 Sign Up Class and Activity Evaluation

Test Class: SignUpActivity					
Test #	Function Tested	Expected Outcome	Actual Outcome	Issues	Comments
1	Passing Bundle to Profile Activity	Successful pass of bundle using intent	Data was passed and added to the Profile Table	N/A	The outcome was as intended
2	Automatically load login details of user	Users' Forename, Surname, Email and Password will be visible	Data was displayed onCreate	N/A	The outcome was as intended
3	Validation of text fields	Sign up is unsuccessful if forename or surname includes symbols, is email does not match email format, or if passwords do not match	Toast is generated if forename or surname are not text-only, or if passwords do not match	Uncertainty regarding the email format – difficult to test all variations of email formats	Validation worked as intended

10.2.1.2. Evaluation

The general consensus is that the sign up activity is a success. Data is loaded, stored and displayed as intended and the user is able to interact with the elements and processes with no issues. However, when initialising the sign up activity, if the user had already created a login then they should be alerted that they are logging in, rather than signing up again. Although a superficial issue, it may cause slight confusion among users.

Input types did not need to be tested as Android removes the ability to enter data that is not desired. For example, in a text view with a number input, letters or decimal points cannot be input.

10.2.2.1 Profile Class and Activity Evaluation

Test Class: ProfileActivity					
Test #	Function Tested	Expected Outcome	Actual Outcome	Issues	Comments
1	Passing Bundle from SignUp Activity	Successful pass of bundle using intent	Data was passed and added to the Profile Table	N/A	The outcome was as intended
2	Insert Profile data into Profile Table of database	Record will appear in the SQLite DB	Record was added to the profile table of the db	N/A	The function worked as expected
3	Calculating currentBMI, classification and optimum weight	BMI, classification and optimum weight will be calculated according to the users' height/ weight input onTextChanged	BMI, classification and optimum weight outputs were accurate	If user deletes height/ weight, the bmi, class or opt.weight values don't delete	The function worked as intended when valid input was given
4	Automatically load profile details of user	Users' DOB, height, weight, gender and allergies/diets should load onCreate	Data was displayed onCreate	N/A	The outcome was as intended
5	Validation of text fields	Profile save unsuccessful if gender is not selected, or height/weight is not entered	Toast is generated if data is not input	No validation was executed on the DOB field	Validation worked as intended - exception of no DOB validation

10.2.2.2. Evaluation

The profile activity works as intended, with little room for error. Various inputs were attempted to cause a program crash but none resulted in extreme errors. The height and weight input types may prove to be problematic for users who are unsure of metric conversions; the application takes cm and kg as input values. As mentioned, there is no procedure to clear the BMI, classification and optimum weight fields if the height and weight input is removed, which results in invalid outputs.

10.2.3.1 ExpiredPantryActivity Class and Activity Evaluation

Test Class: ExpiredPantry					
Test #	Function Tested	Expected Outcome	Actual Outcome	Issues	Comments
1	Load list of expired items from Add to Pantry db table	List of all expired food items, corresponding data should be loaded in a table format	All expired food items are loaded	Table row layout is not as desired. Items with current date are being displayed as expired	Function works as intended

10.2.3.2 Evaluation

This class works as intended, loading any items added to the pantry that are past their expiration date, raising awareness of food waste. To augment this feature, a price could be added to each unit of each food item, with a total displaying how much money has been wasted on expired food. Additionally, there is no feature to remove these expired items from

the AddToPantry table. If the user uses the app for a long period of time, this may be problematic.

10.3 User Testing and Feedback

To determine the success of the application in a realistic environment, it was tested using actual users. Three averagely skilled mobile users tested the application once it had been programmed to run on a mobile device. An additional user test was executed by a Security Analyst for an app development company in order to gain an evaluation from an accredited source.

10.3.1 Average Testers Feedback

Rating /100	USER 1 TEST FEEDBACK									
	Appearance			Navigation			Features			General
	Consistency	Layout	Impact	Consistency	Understandability	Simplicity	Sufficiency	Practicality	Usability	
	68	52	40	65	35	60	70	68	62	65
	53			53			67			60

Rating /100	USER 2 TEST FEEDBACK									
	Appearance			Navigation			Features			General
	Consistency	Layout	Impact	Consistency	Understandability	Simplicity	Sufficiency	Practicality	Usability	
	75	45	45	70	40	40	55	60	60	65
	55			50			58			57

Rating /100	USER 3 TEST FEEDBACK									
	Appearance			Navigation			Features			General
	Consistency	Layout	Impact	Consistency	Understandability	Simplicity	Sufficiency	Practicality	Usability	
	90	60	25	50	70	65	50	65	40	65
	58			62			52			59

10.3.2 Security Analyst Feedback

Rating /100	SECURITY ANALYST TEST FEEDBACK									
	Appearance			Navigation			Features			General
	Consistency	Layout	Impact	Consistency	Understandability	Simplicity	Sufficiency	Practicality	Usability	
	65	40	35	60	30	45	45	55	50	65
	47			45			50			52

10.3.3 Security Analyst Comments

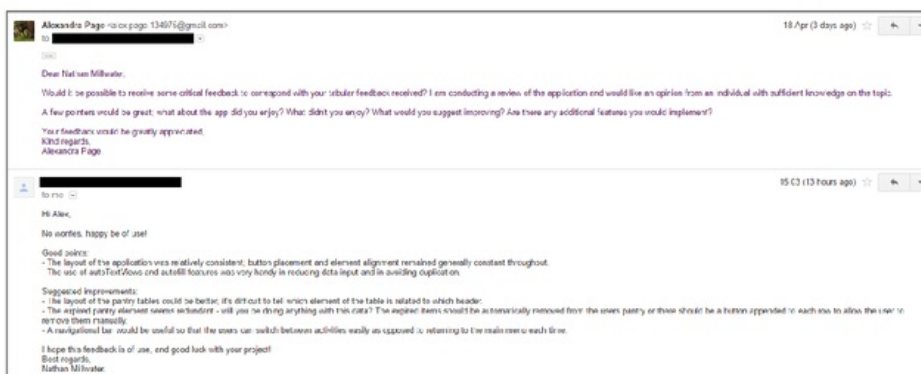


Figure 51: Email exchange between developer and security specialist

11.0 Survey Analysis

An online survey was completed using SurveyMonkey to examine the requirements of the proposed application. In order to determine a target market and operating system, application pricing strategies and how frequently the application would be used, the survey was sent out on social media to target a wide audience. Subsequent to receiving fifty responses, the survey was inactivated and the responses were collected and analysed. To view each question and its analysis, see Appendix ?

11.1 Question 1: What is your gender?

11.1.1 Results and Diagram

Based on the question summaries, twenty-eight respondents were female and the remaining twenty-two were male. The majority of female respondents were aged 18 to 24 and were Android owners. The majority of male respondents were aged 25 to 34 and were Android owners.

11.1.2 Result Expectancy

Due to the survey being available online, the expectation of an equal gender split was reasonable. Although the female-to-male ratio was slightly unbalanced with a slight weight on female respondents, there were no major unexpected results.

11.2 Question 2: What is your age?

11.2.1 Results and Diagram

Over half of the respondents were aged 18-24 and 15 aged 25-44. Only one respondent was under 18, with the remaining being 45 or older.

11.2.2 Result Expectancy

As no particular age group was targeted to complete the survey, the results are widely varied as expected. The two most popular age groups are as anticipated as they tend to be the most active users of social media. The age of the target market was initially between 18 and 65, conforming to the survey results.

11.3 Question 3: Which Smartphone do you own?

11.3.1 Results and Diagram

Based on the results, 30 of 50 respondents selected Android as their answer, deeming it the most popular phone operating system. 19 respondents revealed they own an iPhone, with the remaining one respondent owning a Windows Phone. A series of graphs were compiled with a breakdown of each demographic question response filtered using the phone operating system.

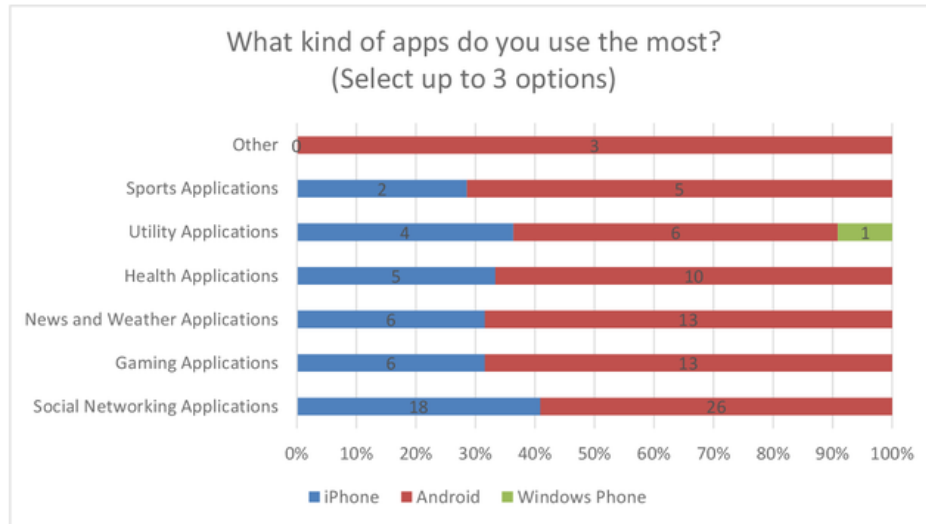


Figure 52: Survey Q.4 response with OS division

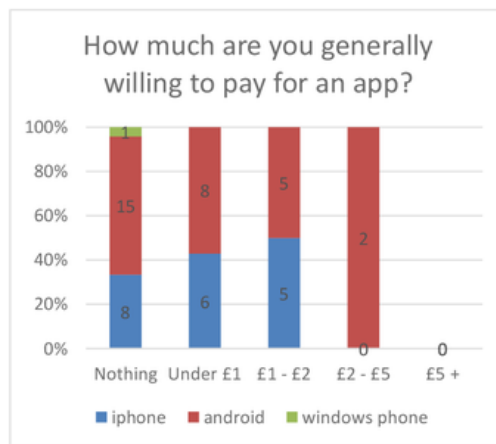


Figure 54: Survey Q.5 response with OS division



Figure 53: Survey Q.6 response with OS division

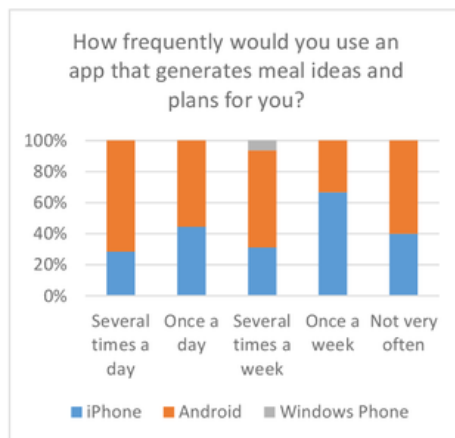


Figure 58: Survey Q.9 response with OS division

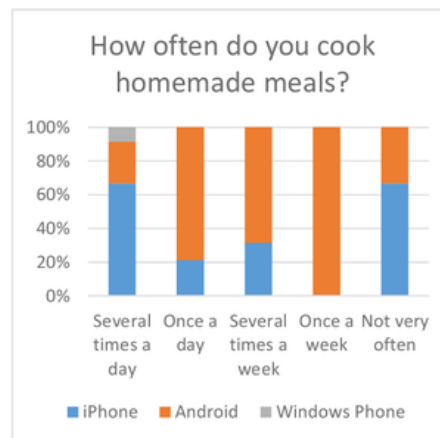


Figure 58: Survey Q.8 response with OS division

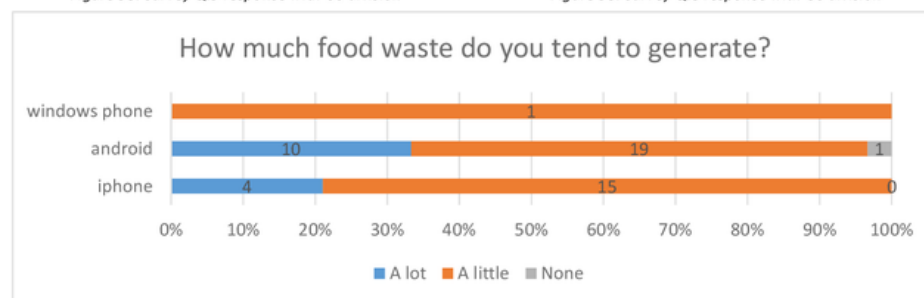


Figure 58: Survey Q.7 response with OS division

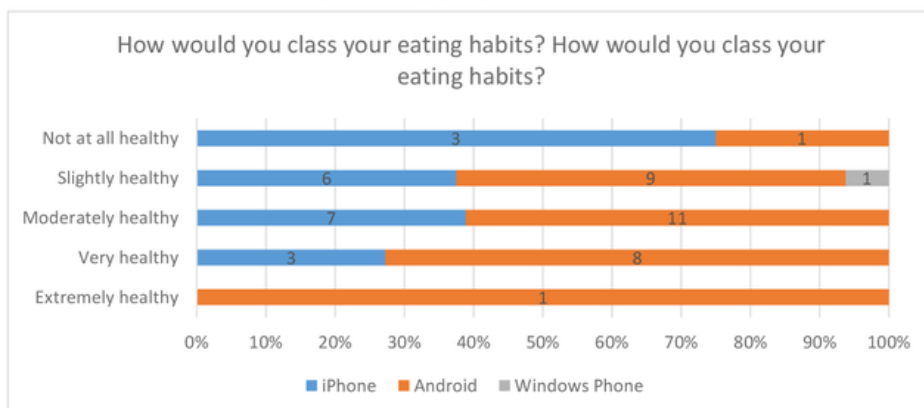


Figure 58: Survey Q.10 response with OS division

11.3.2 Result Expectancy

Statistics showed that Android has a larger market share, so it was expected that it would be the most popular choice. However, focusing on the age of the majority of respondents and the basis that iPhones are typically deemed by the younger generations are more prestigious, a closer gap was expected.

11.3.4 Link to Literature Review

Wall Street Journal reported that Android has an 80% market share, and so it was unsurprising that Android was the more popular phone option. However, users are reported to display higher spending per-user than Android owners, which was not evident from the results shown. Based on the respondents, the iPhone users were more likely to purchase apps, but less likely to spend as much as Android users. More Android users were willing to complete IAP than iPhone users.

11.4 Question 4: What kind of apps do you use the most? (Select up to 3 options)

11.4.1 Results and Diagram

Social networking applications were selected as the most popular, with 88% of respondents selecting this option as one of their top 3 app categories. The second most popular category was gaming applications, with 19 responses. Health

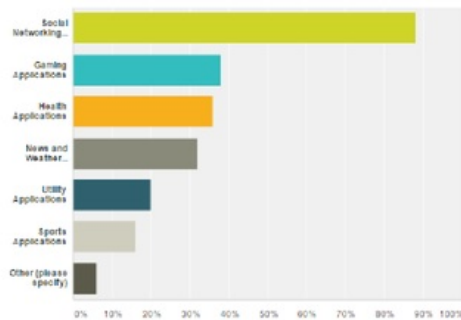


Figure 59: Survey Q.4. combined results

applications were the third-most popular, with 18 responses. Sports applications were the least popular, with only 8 of 50 respondents claiming to use them frequently.

11.4.2 Result Expectancy

Initial predictions of the results put social networking as the most popular choice, due to most smartphone owners utilising social apps several times a day. Gaming applications was also expected to rank highly. However, the popularity of health applications was unexpected; utility and news/weather applications were predicted to be used more frequently. An explanation for the unexpected results may be related to the majority of respondents being young; younger individuals may be more focused on health and wellbeing rather than global issues, and so monitoring health applications may be more of a priority than monitoring news.

11.4.3 Link to Literature Review

Digital Trends released a statement claiming that health and fitness applications are the fastest-growing app category, which may account for the high ranking of health applications in the survey summary. Based on app store statistics, gaming applications are ranked as the most popular category, which was not the case based on the survey responses.

12.0 Evaluation

This section will evaluate whether the requirements were met, in conjunction with a conclusion of the requirements satisfaction determination. A decision will be made as to whether the aim was fulfilled, in addition to the project objectives.

12.1 Must Requirements

#	Requirement	Comment	Satisfied
1	Enter produce items for cupboards, fridges and freezers	Achieved using the Ingredients and AddToPantry classes. Additional features used to reduce data entry for the user such as auto complete text views and automated data input using db queries.	Yes
2	Generate recipes based on produce input	Although this feature does work, the items used for a recipe are not removed from the pantry, which will reduce accuracy.	Yes – with exception
3	Allow users to create a custom recipe	This was achieved using the AddRecipe activity and class.	Yes
4	Notify users when items are close to their expiry date	Although users are not noted when items are close to their expiry date, items that have surpassed their expiry date are displayed in an ExpiredPantry activity. However, these expired items are not removed from the pantry items table.	No – with exception

#	Requirement	Comment	Satisfied
1	Allow user to create a personal profile	This was achieved using the sign up and profile activity. However, the sub-requirement “Remove any recipes where restricted items appear” was not achieved.	Yes – with exception
2	When users enter items, allow selection via list	AutoCompleteTextViews were implemented throughout the application to reduce data duplication	Yes
3	Allow user to sort and filter recipes	No filtration options were made available on the recipes list class and activity.	No

12.2 Should Requirements

12.3 Requirements Evaluation

No “could have”, or “would like” requirements were met. A reason for this is that the initial requirements were arguably extensive and required a vast amount of programming preceding the actualisation of the features in which the requirements would be satisfied, which left little

time to complete the requirements to the full extent necessary to be considered fulfilled. Additional functionality not noted in the requirements was also fulfilled.

12.4 Aim and Objectives Satisfaction

The initial aim was to allow for food input and the tracking of nutrition information to develop healthy eating habits. The project did not entirely meet the aim as there was no nutrition logging features, simply a method to note nutritional information for food items input into the pantry. However, the application did include a series of other features and satisfied the aim of food input.

All objectives were completed; an analysis of competitors and related technologies was completed during the literature review. An evaluation of mobile application methods was accomplished in the development methodology section. Requirements and feature determination was completed through the distribution of surveys and interviews with relevant persons. The application was produced and design decisions were defined. Several forms of testing, including functional tests and user testing determined the success of the application.

12.5 Data Correlation Regarding Literature Review

During the literature review, it was discovered that although many individuals seem concerned with global food waste, not many take the initiative to reduce their personal waste. It was also found that although health applications are the fastest-growing category, little focus has been on managing food intake through recipe generation.

During the user testing process, the functionality section returned a 57% average of sufficiency, practicality and usability regarding the application, enforcing the theory that there is a need for an adequate nutrition and food waste management software such as the one created.

13.0 Conclusions and Recommendations

The overall report generated has provided the basis of an analysis, in which the use of mobile technologies to improve recipe generation and waste management can be examined. This section will provide a summary of all work included in this project, with suggestions regarding the possibility of further developments into the topic. The aim will be addressed in regards to research statements defined in the literature review and a conclusion will be derived based on the outcome of the data relating to these statements.

Does the use of a mobile application allow for efficient monitoring of food expiry dates?

Based on the testing process and the results derived from both user feedback and test runs, the application was successful in taking note of expiry dates and keeping track of expired food items. With the addition of improved functionality, the application should be able to provide more efficiency when notifying the user of expiry dates, such as notification centre messages and the prioritisation of food items close to their expiration dates.

Are there any limitations to the use of mobile applications used for the generation of recipes?

Problems were encountered during the development process in regards to efficient recipe generation. For example, deducing how a food item should be stored proved to be problematic; should chicken breasts be stored in weight or in units? With further work on the functionality, solutions to these problems will most likely arise. Another issue faced is that if a user does not use the application to generate each meal they create, there will be inconsistencies in regards to their virtual kitchen and their actual kitchen, where food items existing on the application may have already been used in reality.

Can a mobile application be a viable method in tracking nutrition intake automatically?

As the nutrition tracking element of the application was not completed, further development to conclude necessary functionality would allow for the application to be tested and evaluated in a physical environment. Based on products completing similar actions on the market it is most likely that an application can eliminate human responsibility of manually tracking nutrition.

By conducting a thorough analysis of the current mobile technology market relating to the nutrition and recipe elements of the application, a definitive market gap was discovered. Through creating a detailed design strategy and implementing the designs during the application generation, it was ensured that the application would prove to be a practical tool which can be used on a daily basis. An evaluation of the product located all insufficiencies and areas of concern within the application, which in turn provided a definitive basis of possible further work on the application or area of interest.

To summarise, through the inclusion of expiry date tracking and the storage of food items, a user was provided with a digital pantry in which monitoring expiry dates of all food items owned was a simple task. The ability to generate recipes based on these stored items provides a tool for people who struggle to form meal concepts and thus it could be used in other industries, such as for overweight individuals who require a healthy diet plan or nutrition tracking software to monitor their intakes.

13.1 Further Work

The completion of the expiry date process would allow for augmented monitoring and alerting of expiring food items. The removal of expired items from the database would ensure recipes cannot be generated based on out-of-date food items. The nutrition tracking feature should be expanded to track the nutrition information of a meal, so that users may monitor their intake.

The implementation of a meal plan function would allow a user to pre-plan meals, thus reducing time spent deciding what to eat each day, in conjunction with improving efficiency through scheduling expiring items. A barcode scanner could be implemented in order to further reduce the time spent inputting food items.

14.0 Reflection

Upon consideration of all challenges faced during the actualization of the application and corresponding documentation, it was deduced that the greatest obstacle was the prioritisation of conflicting commitments. In learning a new skill, it was imperative that time management methods were improved, so as to allow for the additional period required to become accustomed with a new software and series of languages.

Understanding the current global situation regarding food waste (and resultantly climate change and economic devastation) was considered a major beneficial outcome of the project, peaking an interest and inciting further non-academic research.

An interest in mobile application development was discovered during the process. However, it is a difficult concept to grasp during the short duration of the project cycle as it is not a topic covered on the Business Information Technology stream. As a result, the project completion status was affected and many requirements were unsatisfied or left incomplete.

Several methods could have been adopted at an earlier stage, which would have aided in optimizing productivity and ultimately project completion. These include research into the topic and the intended software development platform at an earlier stage, in conjunction with the deployment of the application at an earlier period to gain feedback from a larger demographic.

15.0 References

- [1] Zheng, Y. "Mobile Phones and Libraries/Information Centres" in *Encyclopedia of Mobile Phone Behavior*, IGI Global, 2015. Page 296.
- [2] Fling, B. "A Brief History of Mobile" in *Mobile Design and Development: Practical concepts and techniques for creating mobile sites and web apps*, O'Reilly Media, Inc, 2009. Page 4-5.
- [3] Gonzalez, T., Diaz-Herrera, J. and Tucker, A. "Operating Systems" in *Computing Handbook, Third Edition: Computer Science and Software Engineering*, 3rd Edition, CRC Press, 2014. Page 611-612.
- [4] Management Association, Information Resources. "Mobile Technology and Learner Autonomy in Language Learning" in *Human-Computer Interaction: Concepts, Methodologies, Tools, and Applications: Concepts, Methodologies, Tools, and Applications*, IGI Global, 2015. Page 308.
- [5] Hillebrand, F. "The SMS of September 1990" in *Short Message Service (SMS): The Creation of Personal Global Text Messaging*, John Wiley & Sons, 2010. Page 55-57.
- [6] Tarkoma, S. "Mobile Middleware" in *Mobile Middleware: Supporting Applications and Services*, John Wiley & Sons, 2009. Page 2.
- [7] Elmer-DeWitt, P. "Steve Jobs: 60 million iPhone apps downloaded" for Fortune Magazine. August 11, 2008. [Online: Accessed 02/11/2015]. Available from: <http://fortune.com/2008/08/11/steve-jobs-60-million-iphone-apps-downloaded/>
- [8] Metcalf, A. (January 7, 2011). "App" 2010 Word of the Year, as voted by American Dialect Society. Contacts for Word of the Year: American Dialect Society. Page 2.
- [9] GET OXFORD DICTIONARY QUOTE FOR APP
- [10] (June 8, 1981). "Robert Half" in *Computerworld*. Volume 15 (No. 23), Page 81-107.
- [11] "Worldwide revenue of app stores from 2008 to 2014 (in billion U.S. dollars)" for Statistica. [Online: Accessed 03/11/2015] Available from: <http://www.statista.com/statistics/266998/worldwide-revenue-of-app-stores/>
- [12] Cabural, M. (January 2013). "Google Play Store Has More Apps Than Apple App Store" in *ValueWalk*.
- [13] Gesenhues, A. (June 2015). "Google Play Gaining Ground Against Apple, As Android App Downloads Outnumber iOS 2-to-1" for *Marketing Land*.

- [14] Wakabayashi, D. (April 2015). "Apple's App Store Widens Revenue Lead Over Google Play - Report" for *Wall Street Journal*.
- [15] "Smartphone OS Market Share, 2015 Q2" for *International Data Corporation*. [Online: Accessed 03/11/2015] Available from: <http://www.idc.com/prodserv/smartphone-os-market-share.jsp>
- [16] West, J. and Mace, M. "Browsing as the killer app: Explaining the rapid success of Apple's iPhone." *Telecommunications Policy* 34.5 (2010): 270-286.
- [17] Patterson, S. M. "Revenue from the Google Play store will overtake Apple's app store in 2018" for *Quartz*, 2014.
- [18] Arora, S. "To free, or not to free: the impact of free versions, average user ratings, and App characteristics on the adoption speed of paid mobile Apps." Diss. 2014.
- [19] Poeter, D. (December 2013). "Report: Google Play Grows at Apple App Store's Expense in 2013" for *PC Mag*.
- [20] Linehan, C. et al. (January 2013). "Crowd Saucing: Social Technology for Encouraging Healthier Eating" in *Interactions*. (1072-5520). Volume 20 (Issue 1), Page 53.
- [21] Greatcall, "Is Mobile Healthcare the Future? [Infographic]" [Online: Accessed 06/10/2015] Available from: <http://www.greatcall.com/lp/is-mobile-healthcare-the-future-infographic.aspx>
- [22] Statista - The Statistics Portal, "Most popular Apple App Store categories in September 2015, by share of available apps", updated September 2015. [Online: Accessed 07/10/2015]. Available from: <http://www.statista.com/statistics/270291/popular-categories-in-the-app-store/>
- [23] Statista - The Statistics Portal, "Most popular Google Play app store categories in 4th quarter 2012, by share of listed apps" updated October 2012. [Online: Accessed 07/10/2015]. Available from: <http://www.statista.com/statistics/256772/most-popular-app-categories-in-the-google-play-store/>
- [24] Boxall, A. for Digital Trends. "2014 is the year of health and fitness apps, says Google", updated December 11th, 2014. [Online: Accessed 07/10/2015]. Available from: <http://www.digitaltrends.com/mobile/google-play-store-2014-most-downloaded-apps/>
- [25] Ferenstein, G. for The Verge. "The smartest socks ever: hacking my running with Sensoria's fitness tracker", published October 9th, 2014. [Online: Accessed 07/10/2015]. Available from: <http://www.theverge.com/2014/10/9/6951929/sensoria-running-fitness-tracker>

- [26] Martonik, A. for Android Central. "How to use the Heart Rate Monitor on the Galaxy S5", published July 10th, 2014. [Online: Accessed 08/10/2015]. Available from: <http://www.androidcentral.com/how-use-heart-rate-monitor-galaxy-s5>
- [27] Ortuño, F. and Rojas, I. "Impact of Health apps in Health and Computer Science Publications" in *Bioinformatics and Biomedical Engineering: Third International Conference*, Springer, 2015. Page 24.
- [28] Aitken, M. "Patient Apps for Improved Healthcare: From Novelty to Mainstream" in *IMS Institute for Healthcare Informatics*, 2013. Page 1.
- [29] Simms, H. "Introduction - The background and aims of this study" in *Apadmi's Wearable Tech Study: What Do Businesses Need To Know About Wearable Technology*, 2015. [Online: Accessed 26/11/2015] Available from: <http://www.apadmi.com/wearable-technology-trends/>
- [30] Fordham, L. (September 2015). "What place does wearable technology have in workplace health strategies?" in *Employee Benefits Magazine*, (1366-8722).
- [31] Murphy, D. (October 2014). "Consumers not ready to spend Big Bucks on Wearables" in *PC Mag*, (0888-8507).
- [32] Yock, P. G. et al. "Stage 4: Concept Screening" in *Biodesign*, Cambridge University Press, 2015. Page 331-332.
- [33] Holzinger, A. "System Design Usability and Evaluation" in *Biomedical Informatics: Discovering Knowledge in Big Data*, Springer, 2014. Page 513.
- [34] NewsRx LLC, (February 2015). "Vital connect, inc.; Vital Connect Receives Medical Device Ninsho Certification to Market HealthPatch(R) MD in Japan" in *Telecommunications Weekly*, (1945-841X). Page 280.
- [35] Natural Resources Defense Council. "Food Facts" in *How much do we waste?* Published March 2013. [Online: Accessed 08/10/2015]. Available from: http://www.nrdc.org/living/eatingwell/files/foodwaste_2pgr.pdf
- [36] Vaqué, L. G. (January 2015). "Food Loss and Waste in the European Union" in *European Food & Feed Law Review* (1862-2720). Volume 10 (Issue 1). Page 21-22.
- [37] Anonymous. (March 2013). "Global food - waste not, want not" in *Appropriate Technology* (0305-0920). Volume 40 (Issue 1). Page 5.

- [38] Lucia-Monica, S. and Marius, S. (2013). "Responsible Management Of Food Waste" in *Managerial Challenges of the Contemporary Society*. Proc. 6th. Cluj-Napoca: Babes Bolyai University (101-104).
- [39] Abbot, D. "Early day motion 66: Supermarket Disposal of Edible Food" in *House of Commons Business Papers*, 2015.
- [40] Androich, A. (April 2015). "The Misfits Strike Back" in *Canadian Grocer*, (0008-3704). Volume 129 (Issue 2). Page 17.
- [41] Great Britain, Parliament. House of Commons. "Tackling Food Waste" in *Food Security: Demand, Consumption and Waste*, The Stationery Office, 2015. (Volume 703 of HC). Page 15.
- [42] Schnick, S. (2013). "Are you serious about launching an app or not?" in *Fierce Developer*.
- [43] Rice, K. "How long does it take to build a mobile app?" for *Mobile Backend as a Service*, Kinvey, 2013. [Online: Accessed 16/11/2015] Available from: <http://www.kinvey.com/blog/2086/how-long-does-it-take-to-build-a-mobile-app>
- [44] Cui, Y. and Liang, K. "A Probabilistic Top-N algorithm for mobile applications recommendation." in *Broadband Network & Multimedia Technology (IC-BNMT)*, 2013 5th IEEE International Conference on. IEEE, 2013.
- [45] Hamlyn, M. (2014). "How Long Does It Take to Make an App?" in *The App Maker Blog*. [Online: Accessed 16/11/2015] Available from: <https://www.appmakr.com/blog/how-long-does-it-take-to-make-an-app/>
- [46] Swift, O. "Android Software Development" in *Android: App Development & Programming Guide: Learn In A Day!* 2015. Page 31-39.
- [47] Hu, W. et al. "Smartphone software development course design based on android." *Computer and Information Technology (CIT)*, 2010 IEEE 10th International Conference on. IEEE, 2010.
- [48] Information Resources Management Association, "Types of Philosophy" in *Digital Democracy: Concepts, Methodologies, Tools and Applications*, Vol 1, IGI Global, 2012. Page 169.
- [49] Mackenzie, N and Knipe, S. "Research dilemmas: Paradigms, Methods and Methodology" in *Issues In Educational Research*, Vol 16, IEER, 2006.

- [50] Bahl, S and Milne, G. "Mixed methods in interpretive research: an application to the study of self-concept" in *Handbook of Qualitative Research Methods in Marketing*, Edited by Russell W. Belk, Edward Elgar Publishing, 2006. Page 198.
- [51] Banister, P., Bunn, G. and Burman, E. "Defining and mapping phenomenology as an approach" in *Qualitative Methods In Psychology: A Research Guide*, Second Edition, McGraw-Hill Education, 2011. Page 7.
- [52] Armstrong, M. "Research Philosophy" in *Armstrong's Essential Human Resource Management Practice: A Guide to People Management*, Kogan Page Publishers, 2010. Page 387.
- [53] Information Resources Management Association, "Risk Management Research Design Ideologies, Strategies, Methods and Techniques" in *Research Methods: Concepts, Methodologies, Tools and Applications*, IGI Global, 2015. Page 366.
- [54] Willis, J. "Qualitative Research Methods" in *Qualitative Research Methods in Education and Educational Technology*, Information Age Publishing, 2008. Page 219-221.
- [55] Saunders, M., Lewis, P. and Thornhill, A. "The purpose of your research" in *Research Methods for Business Students*, Fifth Edition, Pearson Education, 2009. Page 144.
- [56] Mangal, S.K. and Mangal, S. "Nature and Types" in *Research Methodology in Behavioural Sciences*, PHI Learning Pvt. Ltd, 2013. Page 48.
- [57] Tavakoli, H. "Cross-Sectional Study" in *A Dictionary of Research Methodology and Statistics in Applied Linguistics*, Rahnama Press, 2012. Page 138.
- [58] Haider, A. "Research Paradigms" in *Information Systems for Engineering and Infrastructure Asset Management*, Springer Science & Business Media, 2013. Page 135.
- [59] Baumgarten, M. "The quality conflict – indicators and paradigms" in *Paradigm Wars – Validity and Reliability in Qualitative Research*, GRIN Verlag, 2013. Page 4.
- [60] Seliger, H. W. and Shohamy, E. "The Components of Research" in *Second Language Research Methods*, OUP Oxford, 1989. Page 95.
- [61] Chang, M. "Planning, Execution, Analysis and Reporting" in *Adaptive Design Theory and Implementation Using SAS and R*, Taylor and Francis Group, 2008. Page 331.
- [62] (Dr) Price, I. "Internal and external validity" in *School of Psychology University of New England*,
 [Online: Accessed 19/10/2015] Available from:
http://webstat.une.edu.au/unit_materials/c2_research_design/validity_internal_external.htm

- [63] Coldwell, D. and Herbst, F. "The Research Process" in *Business Research*, Juta and Company Ltd, 2004. Page 41.
- [64] Holland, D. and Crompton, J. "Digital Engineering and the Digital Engineer" in *The Future Belongs to the Digital Engineer*, Xlibris Corporation, 2013. Page 102.
- [65] Janes, A. and Succi, G. "The GQM Strategies Approach" in *Lean Software Development in Action*, Springer, 2014. Page 166-167.
- [66] Münch, J. et al. "Prescriptive Process Model Classes" in *Software Process Definition and Management*, 2012. Page 25.
- [67] Basu, A. "Introduction to Software Quality Engineering" in *Software Quality Assurance, Testing and Metrics*, 2015. Page 13.
- [68] Theuerkorn, F. "LEA Framework" in *Lightweight Enterprise Architectures*, 2005. Page 41.
- [69] Deek, F., McHugh, J., Eljabiri, O. "The Waterfall Model" in *Strategic Software Engineering: An Interdisciplinary Approach*, 2005. Page 11 - 12.
- [70] Tucker, A. "Human-Centered System Development" in *Computer Science Handbook*, Second Edition, 2004. Page 47 - 48.
- [71] Wasson, C. "Definitions of Key Terms" in *System Analysis, Design and Development: Concepts, Principles, and Practices*, 2006. Page 291.
- [72] Boehm, B. and Hansen, W. "The Spiral Model as a Tool for Evolutionary Acquisition" in *The Journal of Defense Software Engineering*, May 2001. Page 4.
- [73] Bernard, A., Rivest, L. and Dutta, D. "Proposition of a Methodology to Deploy KM in PLM Environment" in *Product Lifecycle Management for Society*, 2013. Page 312.
- [74] Beck, K et al. Manifesto for Agile Software Development. [Online: Accessed 26/10/2015] Available from: <http://www.agilemanifesto.org/>
- [75] Cobb, C. "Agile History and the Agile Manifesto" in *The Project Manager's Guide to Mastering Agile: Principles and Practices for an Adaptive Approach*, 2015. Page 21-25.
- [76] Cockburn, A. "The Agile Alliance" in *Agile Software Development: The Cooperative Game*, Second Edition. Appendix A. The Agile Software Development Manifesto.

- [77] "Information Systems and Software Development" in *Software Design and Development: Concepts, Methodologies, Tools, and Applications: Concepts, Methodologies, Tools, and Applications*, IGI Global, 2013. Page 638.
- [78] Plant, R. and Murrell, S. "Rapid Application Development" in *An Executive's Guide to Information Technology: Principles, Business Models, and Terminology*, Cambridge University Press, 2007. Page 279-280.
- [79] David, W. and Yen, D. "Rapid Application Development" in *The Information System Consultant's Handbook: Systems Analysis and Design*, CRC Press, 1998. Page 248-249.
- [80] Shelly, G. and Rosenblatt, H. "Rapid Application Development" in *Systems Analysis and Design*, Cengage Learning, 2011. Page 146-147.
- [81] McConnell, S. "Managing Customer Expectations" in *Rapid Development*, Microsoft Press, 1996. Page 242-243.
- [82] Layton, M. "Agile Frameworks" in *Agile Project Management For Dummies*, John Wiley & Sons, 2012. Page 69.
- [83] Beck, K. and Andres, C. "Preface" in *Extreme Programming Explained: Embrace Change*, Addison-Wesley Professional, 2004. Page xxi.
- [84] Farid, M. "Extreme Programming Project Velocity Model" in *Artificial Intelligence Applications for Improved Software Engineering Development: New Prospects*, IGI Global, 2009. Page 15.
- [85] Maurer, F. and Wells, D. "Pair Learning: With an Eye Toward Future Success" in *Extreme Programming and Agile Methods - XP/Agile Universe 2003*, Springer, 2011. Page 186.
- [86] Baumeister, H., Marchesi, M. and Holcombe, M. "UC Workbench - A Tool for Writing Use Cases and Generating Mockups" in *Extreme Programming and Agile Processes in Software Engineering*: Springer Science & Business Media, 2005. Page 231.
- [87] Wells, D. and Williams, L. "An Informal Formal Method for Systematic JUnit Test Case Generation" in *Extreme Programming and Agile Methods - XP/Agile Universe 2002*, Springer Science & Business Media, 2002. Page 131.
- [88] Pearman, G. and Goodwill, J. "Introducing XP" in *Pro.NET 2.0 Extreme Programming*, Apress, 2006. Page 10.
- [89] Schwaber, K. "SCRUM Development Process: Advanced Development Methods" in *OOPSLA Business Object Design and Implementation Workshop*, Springer, 1995.

- [90] Takeuchi, H. and Nonaka, I. "New New Product Development Game" in *Harvard Business Review* 64, January 1986.
- [91] Luffingwell, D. "Perspectives on Dual Roles of Product Owner and Product Manager" in *Agile Software Requirements: Lean Requirements Practices for Teams, Programs, and the Enterprise*, Addison-Wesley Professional, 2010. Page 203-204.
- [92] Hughes, R. "Summaries of Generic Agile Development Methods" in *Agile Data Warehousing for the Enterprise: A Guide for Solution Architects and Project Leaders*, Elsevier, 2015. Page 21.
- [93] Gregoire, M. "Maximizing Software Engineering Methods" in *Professional C++*, John Wiley & Sons, 2014. Page 790.
- [94] Larman, C. "Workproducts, Roles and Practices" in *Agile and Iterative Development: A Manager's Guide*, Addison-Wesley Professional, 2004. Page 117.
- [95] Watkins, J. "Agile Testing: How to Succeed in an Extreme Testing Environment" in *Agile Testing: How to Succeed in an Extreme Testing Environment*, Cambridge University Press, 2009. Page 230.
- [96] Sharma, S. and Pandey, S. (August 2013) "Revisiting Requirements Elicitation Techniques" in *International Journal of Computer Applications (0975 – 8887)*. 75 (No. 12), Page 3.
- [97] Mar, K. and Schwaber, K. "Scrum Process with XP Engineering Practices" in *Scrum with XP*. [Online: Accessed 18/11/2015] Available from: <http://faculty.salisbury.edu/~xswang/research/papers/serelated/scrum/scrumxp.pdf>
- [98] Weyns, D. et al. "F or MAAD: Towards a Model Driven Approach" in *Agent-Oriented Software Engineering XI: 11th International Workshop*, Springer, 2011. Page 162.
- [99] B'Far, R. "Introduction to UML" in *Mobile Computing Principles: Designing and Developing Mobile Applications with UML and XML*, Cambridge University Press, 2004. Page 164.
- [100] Fowler, M. "Class Diagrams: The Essentials" in *UML Distilled: A Brief Guide to the Standard Object Modeling Language*, Addison-Wesley Professional, 2004. Page 35.

16.0 Appendices

Appendix A: MoSCoW Requirements

Must have

1. Enter produce items for cupboards, fridges and freezers
 - 1.1. Select location of produce (fridge, freezer, cupboards)
 - 1.2. Select produce type (dairy, grain, vegetable, fruit, meats, other)
 - 1.3. Store product names, expiry dates and quantity
 - 1.4. Enter nutrition information
2. Generate recipes based on produce input
 - 2.1. View time available, meal type, restrictions, ingredients and cooking instructions
 - 2.1.1. View calorie/carbohydrate/fat intake amount
 - 2.2. Remove items used for recipe from pantry
3. Allow users to create a custom recipe
 - 3.1. Set name, meal type and cook time
 - 3.2. Add ingredients from list
 - 3.3. Add cooking method
4. Notify users when items are close to their expiry date
 - 4.1. Remove expired items from pantry
 - 4.2. Store expired items in table

Should Have

1. Allow user to create a personal profile
 - 1.1. Set height, weight, gender and age
 - 1.2. Generate current BMI and healthy BMI for user
 - 1.3. Select any restrictions (allergies, diets – vegan, vegetarian, pescatarian)
 - 1.3.1. Remove any recipes where restricted items appear
2. When users enter items, allow selection via list
 - 2.1. Auto-fill list as user types in search bar
 - 2.2. Have nutrition information pre-loaded for food items
3. Allow user to sort and filter recipes
 - 3.1. Based on time/difficulty/meal type

Could Have

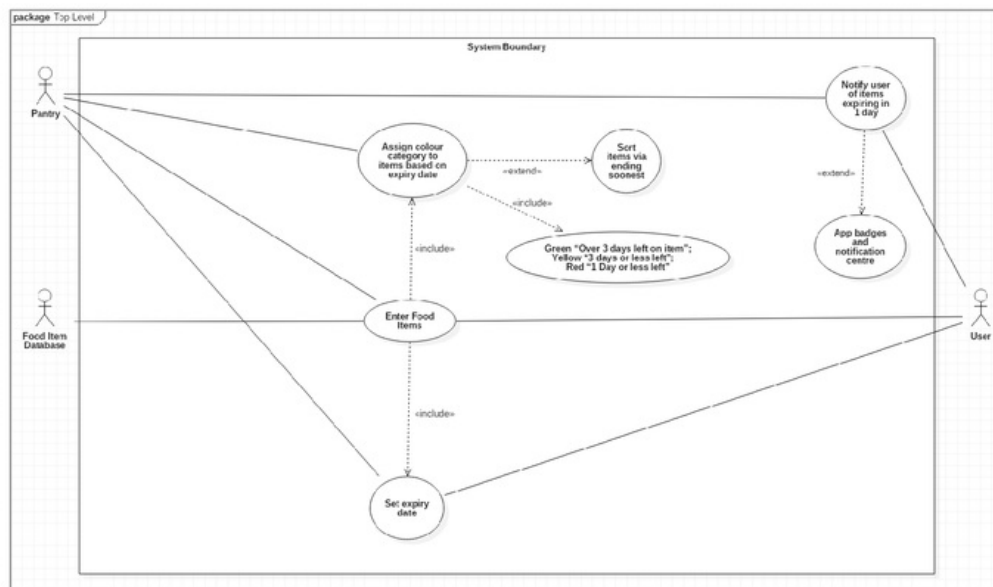
1. Allow the user to add any recipes to a 'meal plan'
 - 1.1. Select recipe
 - 1.2. Select day slot and time slot
2. Allow daily plan view or weekly plan view
 - 2.1. Allow user to view all meals for the day or week
 - 2.2. Allow users to select day to view nutritional information
 - 2.3. Allow user to remove/change recipes added to a meal plan
3. Use barcode scanner to scan items into pantry
 - 3.1. Use phone camera to scan items instead of entering items
4. Add nutritional intake to users' nutrition tracker in the health section
 - 4.1. Calculate daily intake based on each meal generated for that day
 - 4.2. Add nutrition information for recipes added to meal plan for each day
 - 4.3. Calculate percentage of nutritional values used each day
 - 4.3.1. Warn users if their intake is over their suggested daily intake
 - 4.3.2. Traffic light system for food intake (blue = under, green = ok, yellow = high, red = over)

Would like but Won't get

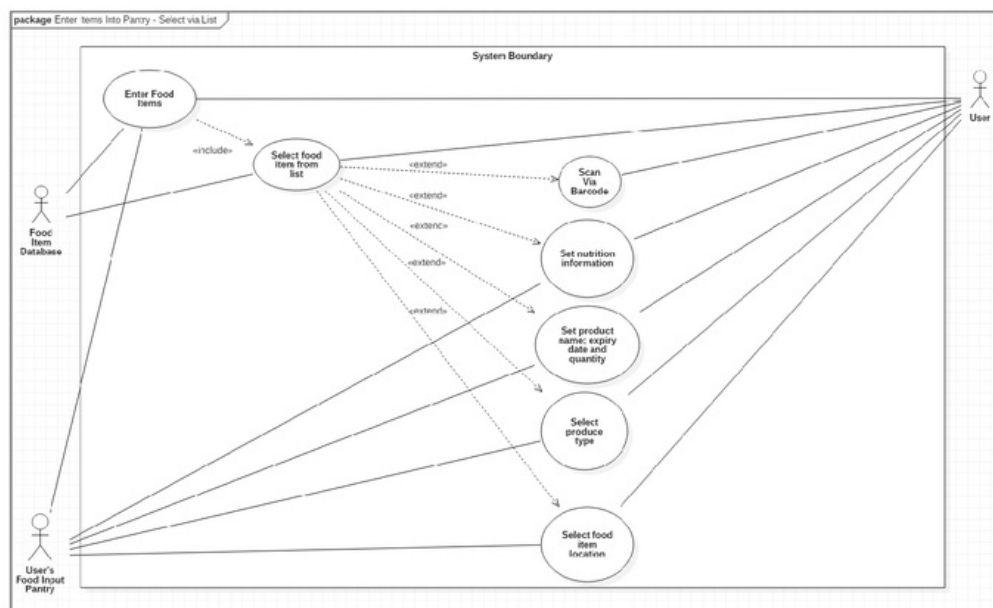
1. Connection to fitness tracker
 - 1.1. Fitbit/jawbone/apple watch connection to merge tracking data to app
 - 1.2. Can calculate net calorie intake by subtracting calories burnt from calories eaten
2. Connect to supermarket
 - 2.1. Able to scan barcode on a receipt to add all items to database
 - 2.1.1. Reduce food input time
3. Integrate step tracker into app health section
 - 3.1. Calculate approximate calorie usage from steps

Appendix B: Behavioural Diagrams

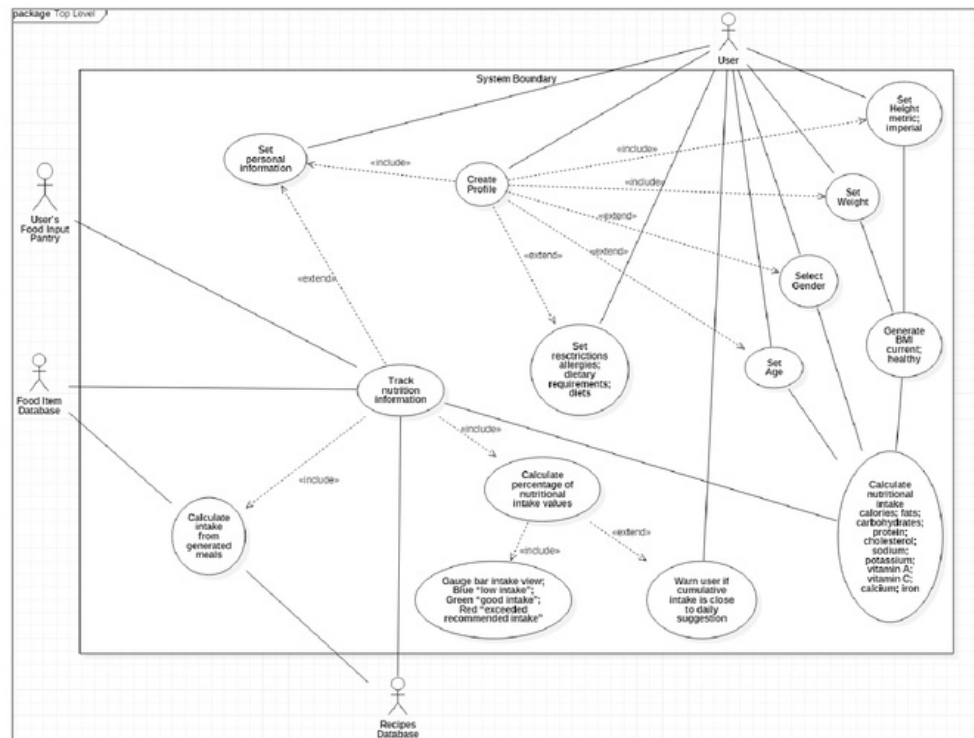
Enter Food Items Use Case



Expiry Dates Use Case



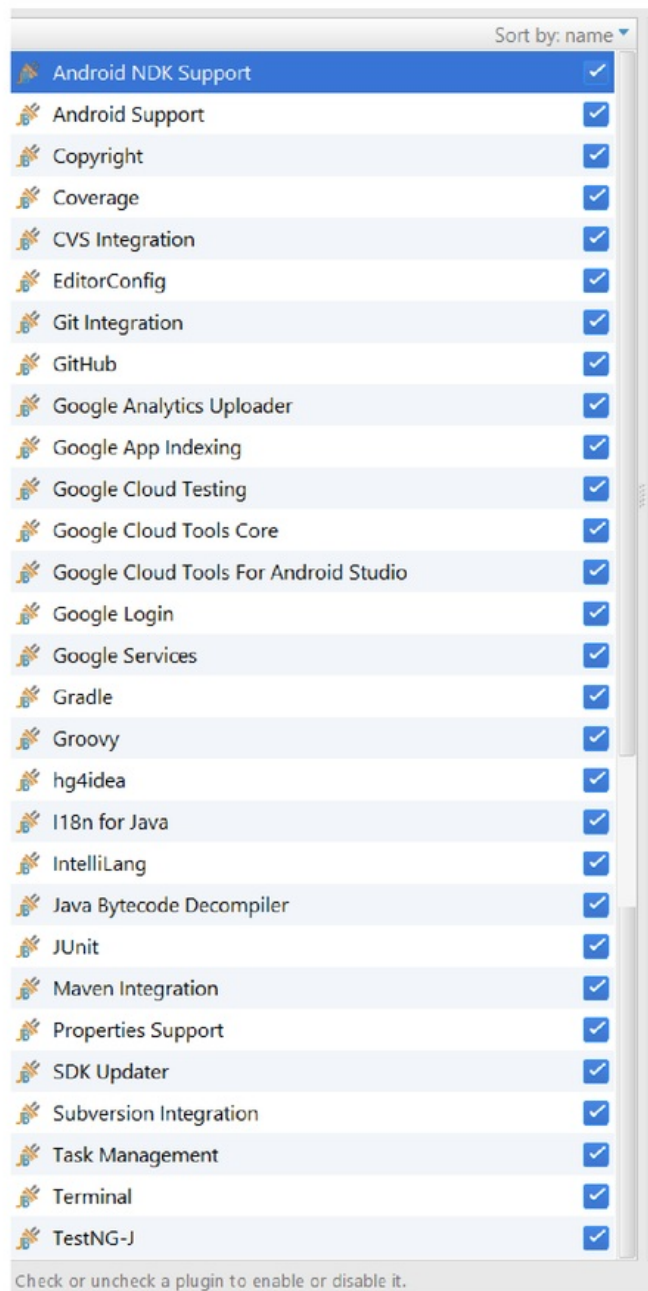
Personal Profile and Nutrition Use Case



Appendix C: CD ROM Application

Appendix D: All Android Studio Plugins Used

Here is a list of all plugins accessed by the project application.



Appendix E: Test Plan

Ingredients Class and Activity Evaluation

Test Class: Ingredients					
Test #	Function Tested	Expected Outcome	Actual Outcome	Issues	Comments
1	Load food category options into spinner	A list of food categories would be displayed in the spinner	Food categories displayed on form load	No solution was found to assign a hint or title to spinner	Spinner worked as expected
2	Food Name generates auto complete view	If an item with the same name has been added to FoodItems DB, the autocomplete will display the name	"milk" was added and then the form was reopened. "m" was entered and milk was displayed as option	If food item is already created, it can be created again & duplicate instances of "milk" will be displayed	Auto complete works as intended
3	Update nutrition based on auto selection using cursor to load from DB	If "milk" is selected from auto complete option, nutrition info, allergies, diets and category should load	Nutrition info, allergies and diets load when item is selected from auto complete view but category does not work	Spinner.setSelection code is incorrect and the category is not automatically set	DB cursor loads correct data to correspond with the auto text selected
4	Set Allergy/Diet of food item is DB tables	Diet ID/Allergy ID and Food ID should be placed in db using if statement and enums	Corresponding food item id and diet/allergy placed in FoodAllergy and FoodDiet tables	N/A	Function works as intended
5	Food Item is added to DB	Record should appear in FoodItems table	Records appears in table in db	N/A	Function works as intended

Evaluation

The duplication of food items is problematic as if a user adds the same ingredient multiple times with different nutrition information, when selecting the ingredient to add to pantry they will not know which version to select. This problem may be avoided by optimising the String Query on the FoodItems table in the database handler by adding an if statement to compare values. A solution was implemented with the spinner by adding a value called "select category" and then using validation to disallow that option from being saved.

PantryActivity Class and Activity Evaluation

Test Class: PantryActivity					
Test #	Function Tested	Expected Outcome	Actual Outcome	Issues	Comments
1	Load list of food items from AddToPantry db table	List of all food items and corresponding data should be loaded in a table format	All food items are loaded	Table row layout is not as desired – column items are grouped together with no spacing	Function works as intended
2	Items past expiry date to be removed from view	All rows displayed should have	Expired items are still displayed	No accurate syntax could be discovered for removing the expired items from the pantry view	Function does NOT work

Evaluation

In order to remove the expired items from the Pantry view, the values of the expired cursor needed to be compared against the values of the pantry items cursor. However, when creating the for statement, the correct syntax could not be determined, resulting in a non-functional operation. This will prove to be problematic during the recipe generation process as items past their expiry date may be used to generate recipes.

ExpiredPantryActivity Class and Activity Evaluation

Test Class: ExpiredPantry					
Test #	Function Tested	Expected Outcome	Actual Outcome	Issues	Comments
1	Load list of expired food items from Add To Pantry db table	List of all expired food items and corresponding data should be loaded in a table format	All expired food items are loaded	Table row layout is not as desired. Items with current date are being displayed as expired	Function works as intended

Evaluation

This class works as intended, loading any items added to the pantry that are past their expiration date. This function will help a user raise awareness of their food waste. To augment this feature, a price could be added to each unit of each food item, with a total displaying how much money has been wasted on expired food. Additionally, there is no feature to remove these expired items from the AddToPantry table. If the user uses the app for a long period of time, this may be problematic.

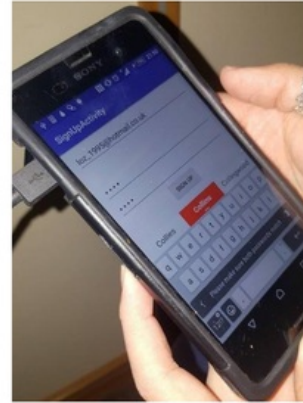
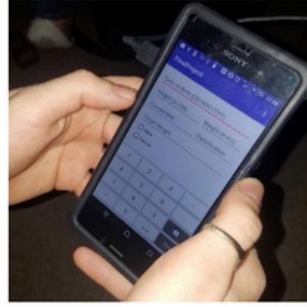
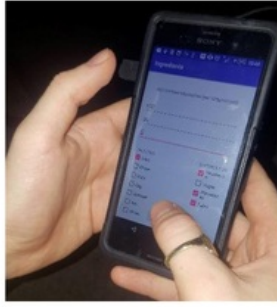
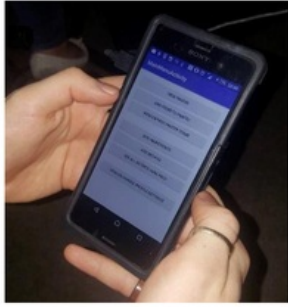
AddToPantry Class and Activity Evaluation

Test Class: AddToPantry					
Test #	Function Tested	Expected Outcome	Actual Outcome	Issues	Comments
1	Load food category and location options into spinner	A list of food categories and locations would be displayed in the spinners	Food categories and food locations displayed on form load	No solution was found to assign a hint or title to spinner	Spinners worked as expected
2	Food Name generates auto complete view	If an item with the same name has been added to FoodItems DB, the autocomplete will display the name	"milk" was added and then the form was reopened. "m" was entered and milk was displayed as option	If food item is already created, it can be created again & duplicate instances of "milk" will be displayed	Auto complete works as intended
3	Update nutrition on auto selection using cursor to load from DB	If "milk" is selected from auto complete option, nutrition info, allergies, diets and category should load	Nutrition info, allergies and diets load when item is selected from auto complete view but category does not work	Spinner.setSelection code is incorrect and the category is not automatically set	DB cursor loads data to correspond with the auto text selected
5	Food Item is added to Pantry DB	Record should appear in FoodItems table	Records appears in table in db	N/A	Function works as intended

Evaluation

The database was displaying a different expiry date to what was selected using the date picker. A solution was found in which the date picker month value was incremented by 1 to maintain accuracy. The same issue regarding the spinners noted in the previous test plan was valid here, with the same solution being implemented.

Appendix F: User Testing Images



Appendix G: Survey Results Analysis

Question 5: How much are you generally willing to pay for an app?

Results and Diagram

Almost half of the respondents stated they would not pay anything for apps, whilst 28% said they would pay a maximum of £1 for an app. 20% said they would spend between £1 and £2. The remaining 4% revealed they would pay between £2 and £5, with no respondents willing to pay above £5 for an app.

Result Expectancy

The results were relatively as predicted; as there are so many applications available on app stores, it is uncommon for an individual to spend a large sum of money on apps. Due to the majority of paid apps having free, lite or freemium versions, it was expected that most respondents would be unwilling to spend any money on apps.

Link to Literature Review

Based on the survey results, 92% of respondents were unwilling to spend more than £2 on an app, which correlates to statistics released by Google Play, stating that only 1% of their revenue is generated from paid apps. Within the Apple App Store, 90% of apps are free or freemium, supporting these survey statistics.

Question 6: Do you ever use in-app purchasing?

Results and Diagram

When provided with three options, 48% of respondents claimed to never use in-app purchasing, whilst 44% said they use in-app purchasing every now and then, though not frequently. The remaining four respondents said they complete in-app purchasing on a regular basis.

Result Expectancy

The results were not exactly as expected; due to the majority of app store revenue deriving from in-app purchasing within freemium apps, it was predicted that the amount of respondents claiming to never complete in-app purchases would be lower. However, it was anticipated that the number of individuals completing IAP very often would be low.

Link to Literature Review

According to a PC Mag report, 92% of Apple's Store revenue is generated from freemium apps with in-app purchasing, as is 98% of Google Play's revenue. This corresponds to 52% of the survey respondents completing in-app purchasing, frequently or infrequently.

Question 7: How much food waste do you tend to generate?

Results and Diagram

Based on the combined results collected from the survey, 98% of respondents produced a little to a lot of food waste, with a mere 2% claiming they did not generate any food waste. The majority of responses viewed the food waste as a little, whereas 28% admitted to wasting a lot of food.

Result Expectancy

It was expected that the results would be similar to the outcome, as many people purchase surplus food and as a result have no time to eat it before it spoils. Contrarily, it was predicted that the disposition of people wasting a little of food would be at a balance with respondents wasting a lot, due to the statistics of food waste within the UK.

Link to Literature Review

As previously mentioned, WRAP estimated that of 15 million tonnes of annual food waste within the UK, over half is a result of food disposal within households. The result of the survey question corresponds to these figures, with almost all respondents admitting to contributing to the mass of food waste. This reinforces the need for a system to control food expiry dates, which is the primary aim of the proposed application.

Question 8: How often do you cook homemade meals?

Results and Diagram

The question was provided with five possible responses, varying from several times a day to not very often. Over half of the respondents claimed to prepare homemade meals once a day or more, with 38% of all responses cooking several times a week. The remaining five respondents stated they cooked once a week or less.

Result Expectancy

There were more respondents who claimed to prepare homemade meals several times a week than predicted; although it is not customary for people to cook all of their meals at home due

to other obligations, it was expected that the vast majority of respondents would select once a day as their cooking frequency, such as breakfast or supper. A possible explanation for the unforeseen results may be that respondents do not class meals such as breakfast as homemade.

Question 9: How frequently would you use an app that generates meal ideas and plans for you?

Results and Diagram

When faced with this question, the responses were generally divided. 28% of respondents said they would use the app several times a day, whereas 18% said they would use it once a day. The most popular choice was several times a week, with 32% of respondents opting for this choice. 12% of respondents conceded they would use it once a week, whilst the last 10% revealed they would not use it often.

Result Expectancy

Due to the nature of the application and the fact that there are not many similar apps available, it was anticipated that the respondents would be more likely to use the application as opposed to not. Although 39 respondents said they

	Several times a day	Once a day	Several times a week	Once a week	Not very often	Total
Q2: Under 18 (A)	100.00% 1	0.00% 0	0.00% 0	0.00% 0	0.00% 0	2.00% 1
Q2: 18 to 24 (B)	29.69% 6	17.24% 5	44.83% 13	13.79% 4	3.45% 1	58.00% 29
Q2: 25 to 34 (C)	37.50% 3	37.50% 3	12.50% 1	0.00% 0	12.50% 1	16.00% 8
Q2: 35 to 44 (D)	42.86% 3	14.29% 1	14.29% 1	14.29% 1	14.29% 1	14.00% 7
Q2: 45 to 54 (E)	25.00% 1	0.00% 0	25.00% 1	25.00% 1	25.00% 1	8.00% 4
Q2: 55+ (F)	0.00% 0	0.00% 0	0.00% 0	0.00% 0	100.00% 1	2.00% 1
Total Respondents	14	9	16	6	5	50

would use it several times a week or more, it was expected that less people would have selected 'not very often'. The majority of respondents who claimed that health apps were one of the most frequent app categories they used said they would use the app once to several times a day. There is a correlation between the age of the respondent and the frequency of which they would use the app, with older respondents less likely to use the app frequently.

Question 10: How would you class your eating habits?

Results and Diagram

Five responses were available to this question, from not at all to extremely healthy. Only one respondent classed their eating habits as “extremely healthy” while 22% claimed their diet was very healthy. The most popular rating was “moderately healthy” with 18 respondents selecting this answer. The second most popular choice was slightly healthy, with 16 responses. The remaining 8% opted for “not at all healthy”.

Statistical Analysis

By adding a comparison filter to the question summaries, viewing the responses of each question against the others was possible. This was completed for the final question, in order to view answers given by respondents for how frequently they cook homemade meals against how healthy their diet is.

	Extremely healthy	Very healthy	Moderately healthy	Slightly healthy	Not at all healthy	Total
Q8: Several times a day (A)	0.00% 0	50.00% 9	33.33% 4	8.33% 1	8.33% 1	24.00% 12
Q8: Once a day (B)	7.14% 1	14.29% 2	35.71% 5	42.86% 6	0.00% 0	28.00% 14
Q8: Several times a week (C)	0.00% 0	15.79% 3	31.58% 6	47.37% 9	5.26% 1	38.00% 19
Q8: Once a week (D)	0.00% 0	0.00% 0	50.00% 1	0.00% 0	50.00% 1	4.00% 2
Q8: Not very often (E)	0.00% 0	0.00% 0	66.67% 2	0.00% 0	33.33% 1	6.00% 3
Total Respondents	1	11	18	16	4	50

As shown, for the respondents who selected the frequency of their homemade meals as several times a day, six viewed their diet as very healthy and four as moderately healthy. One respondent claimed their diet, although mostly homemade, was not at all healthy. A respondent who ate homemade meals once a day viewed their diet as extremely healthy, with two considering theirs very healthy and five stating they were moderately healthy. The most popular frequency of homemade meals in comparison to diet healthiness was several times a week with a diet rating of slightly healthy.

Result Expectancy

There is typically a correlation between how frequently an individual will cook themselves homemade meals and how healthy their diet is. However, when classing each response from question 9 and comparing against their response for their eating habits, unexpected responses were shown. This may be a result of respondents having different views of what constitutes as a healthy or unhealthy diet, or the type of homemade meals each respondent prepares.

[Link to Literature Review](#)

Based on a report by WHO, it was noted that worldwide obesity levels have more than doubled since 1980. As 76% of the respondents viewed their diet as moderately healthy to unhealthy and only 2% viewed their diet as extremely healthy, the results relate to the current state of global health. This lack of a healthy diet among respondents demonstrates that there is a requirement for a method to control nutritional intakes and provide an aid to generating healthier meals.

[Appendix H: Survey Design](#)

PGF App

1. What is your gender?

☐ Female

☐ Male

☐ Other

2. What is your age?

☐ Under 18

☐ 18 to 24

☐ 25 to 34

☐ 35 to 44

☐ 45 to 54

☐ 55+

3. Which Smartphone do you own?

☐ iPhone

☐ Android phone

☐ Other (please specify)

* 4. What kind of apps do you use the most? (Select up to 3 options)

☐ Gaming Applications

☐ Utility Applications

☐ News and Weather Applications

☐ Health Applications

☐ Sports Applications

☐ Social Networking Applications

☐ Other (please specify)

5. How much are you generally willing to pay for an app?

☐ Nothing (Free Apps)

☐ Under £1

☐ £1 - £2

☐ £2 - £5

☐ £5 +

6. Do you ever use in-app purchasing?

☐ Extremely often

☐ Not so often

☐ Never

7. How much food waste do you tend to generate?

☐ A bit

☐ A little

☐ Never

8. How often do you cook homemade meals?

☐ Several times a day

☐ Once a day

☐ Several times a week

☐ Once a week

☐ Not very often

9. How frequently would you use an app that generates meal ideas and plans for you?

☐ Several times a day

☐ Once a day

☐ Several times a week

☐ Once a week

☐ Not very often

10. How would you class your eating habits?

☐ Extremely healthy

☐ Very healthy

☐ Moderately healthy

☐ Slightly healthy

☐ Not at all healthy

Done