**Honours Project Draft**

Kari McMahon

**Abstract**

From October 6th 2014 to the 5th of May 2015 a student, Kari McMahon developed an Android application that aims bring the tradition of recipe books to the 21st century through collaboration as her honours project. The report focusses on the software development lifecycle taken while developing the application. The report outlines the need for the application, requirement gathering, design, implementation and evaluation of the application as well as focus on the project management that goes along with this process. The report will reflect on the successes, challenges and lessons learnt from various aspects of the project.

**Introduction**

From October 6th 2014 to the 5th of May 2015 a student, Kari McMahon took part in the AC40001 individual project module at the University of Dundee. The aim of the module is for the student to gain experience in carrying out an independent software development project which is the culmination of the students years of study and relevant to your future career plans. The project chosen by Kari McMahon was an Android application named Recipes For Life which aims to bring the tradition of recipe books to the 21st century through collaboration. The motivation for the application came from research into recipe applications where she found no tools that enabled multiple people to contribute and combine recipes in shared cookbooks, the way you would with a traditional cookbook. As we move more and more into a digital age we could lose the idea and benefits of collaborative cookbooks which motivated the creation of the application. This report will explore the motivation further and focus on the software development lifecycle that was taken when developing this application. Outlining aspects such as requirement gathering, design, implementation and evaluation of the application as well as focus on the project management side of the project. The report will reflect on the successes, challenges and lessons learnt from the various aspects of the project.

**Background**

A recipe book is defined as a “book of directions explaining how to prepare and cook various kinds of foods” (Merriam-Webster, 2014). Although many successful cookbooks do not just represent directions but they represent people’s experiences and their traditions which we can identify with and get excited about often through tantalising pictures, interesting descriptions and innovative ideas (Ruhlman, 2012). Browsing through the Apple and Android application stores there is a fair share of cooking based applications such as apps that are recipe books, apps to find and store new recipes and cooking toolkits to aid your work in the kitchen. Yet none of these applications represent the way many successful recipe books are created which is through collaboration. Many of the great recipe books contain recipes that have been passed down through the generations and altered or tips have been added by friends and family testing the recipes. There is no application on the app stores that enable users to maintain a cookbook over the years collaborating with others to create personalised cookbooks between friends, families, clubs or maybe even with strangers who have similar interests. Recipe books have always enabled others to add new recipes, alterations and pass on to others and as we move more and more into a digital aid we need an application that enables us to maintain this tradition. The recognition of this sparked research into the history of cookbooks and collaborative cookbooks to see if there would be benefits to building an application based on this.

Recipes have been around for thousands of years with the earliest collection of recipes to survive in Europe being the [De Re Coquinaria](http://en.wikipedia.org/wiki/De_re_coquinaria) which dates back to the 5th century AD. Several pieces of research have looked at the history of cookbooks. For example the paper Cookbooks As A Social And Historical document – A Scottish Case Study examines whether Scottish cookbooks published between 1890 and 1990 are historical markers of major events and technological advances in society. The results from the paper validated the claim that cookbooks are a social, historic and cultural document stating that “although cookbooks might not record events in society as historical facts nevertheless their contents are often a response to historical events.” (Mitchell, 2001). In the book Eat My Words: Reading Women’s Lives Through The Cookbooks They Wrote the author examines cookery books from the US and UK from the 17th to mid-twentieth century. The book documents how women from “diverse backgrounds have found the homely cookbook a suitable place to record their stories and thoughts as well as their recipes.” With the aim to tell the untold stories of these women as well as getting others to consider cookbooks as worthy objects of serious textual analysis. The book demonstrated how recipe books told the history of that time but also how we could still maintain a connection with these recipe books are with the author making a strong point “How deep are the connections between us – a 17th century mother and myself –despite the time that has elapsed.” (Theophano, 2003) These books and papers mentioned demonstrate how cookbooks can be used as historic documents which can tell the story of the time and can be beneficial for future research and understanding of that period of time.

Since the application being developed aims to be a collaborative recipe book tool a lot of the background reading was focussed on collaborative cookbooks and the benefits of they bring. The majority of research showed collaborative cookbooks brought many benefits for example Intensifying Taste, Intensifying Identity: Collectively Through Cookbooks is a paper examining whether community cookbooks implicitly rebuke a social order that devalues women’s work by focussing on Lutheran church women. The author found that the process gave a voice to the community through building the cookbook by organizing it, discussing experiences and producing and selling them. The author found the cookbook was not just a way to raise money but also a way to recognise each of ladies knowledge and experience of cooking and share it with those who were interested. (Ferguson, 2012) Similarly in the book Recipes For Reading contains a collection of essays that demonstrate the different stories available in cookbooks and in particular community cookbooks. The book states recipes are increasingly becoming “readable with great benefit to our knowledge of women’s experiences and discourses”. The author states that cookbooks “tell stories – autobiographical in some case, historical sometimes and perhaps factious or idealized in other instances.” (Bower, 1997). Although Eat My Words was a book which focussed on the historic aspect of recipe books it made a strong point about cookbooks which summarized the benefits of community cookbooks perfectly stating that “modifications and modernizations of old recipes and the invention of new dishes in a woman’s cookbook represent the combined effort of many people. Contributions may come from past circles, sometimes from one or more cultures and while we tend to think of cooking as a delight to our senses, the relationships formed through these culinary compositions are social, cultural and economic.” (Theophano, 2003) The research into cookbooks showed that there was a possibility for this type of application in the market as it was shown through the various pieces of research that cookbooks bring people together, document history and become a way to share and bond over your experiences.

From recognising the benefits that collaborative cookbooks and cookbooks in general bring from the background research. The focus turned to researching similar products in particular apps for tablets. The research was focussed on tablets as they seem to be increasingly used more in kitchens with AllRecipes.com stating in 2013 that social referrals came from tablet devices were up 787% from 2012 to 2013 as well as an increase in page views from tablets. As well as increasing amounts of kitchen accessories for tablets such as tablet kitchen stands, covers to protect from spillages and styluses to use when cooking instead of touching the screen with dirty hands, With accessories like these making their way onto the market tablets seem the way to go when creating technology applications for the kitchen. The applications available for tablets often fall into two categories recipe discovery or recipe management with some apps merging the two categories. For example some of the most popular recipe discovery applications are All The Cooks a social cooking application available for all OS’s enables users to find new recipes, write reviews, add photos or ask questions for the recipes as well as adding your own recipes as well as neat features like shopping lists, conversion features and nutrition facts. Whereas another application is BigOven which is for all OS’s which is strong in both recipe management and recipe discovery enabling users to import recipes from websites, take images of recipes and convert to recipe as well as manually typing recipes. In BigOven users can also browse nearby or popular recipes. BigOven is a strong competitor with over 8 million downloads and won various awards. There also many other applications available on the app stores which do variations of things that have been discussed. Although there is a large amount of applications, they all lacked a collaborative tool feature enabling users to create multiple cookbooks which they could set up for others to maintain. Again this research showed there is room on the market for the project although from the research of various applications you are able to see the successes and what people like about these applications which can work in addition to the collaboration features of this project.

**Specification**

Initial requirements from research. Changed based on survey.

Project plan – work schedule.

**Design**

Before implementation could start on the project several design decisions had to be made which can be seen below.

**OS / Device Selection**

As stated earlier the application would be developed predominately for tablets as they are increasingly becoming more popular for use in the kitchen. This meant a tablet operating system needed to be selected before development. In terms of tablet operating systems there are three main options you can develop for which are Android, IOS or Hybrid / Web apps. In 2014 IDC.com stated the worldwide smartphone operating system market share for Q2 2012 was 84.7% for Android and 11.7% for ios with others making up 3.7% of the market share , they stated the predominant vendors market share was 24.4% with Samsung and 11.7% with Apple and in 2012 IDC.com stated the tablet OS market share was 53.8% Apple and Android 42.7% (IDC, n.d.), (IDC, 2014), (McCracken, 2013). In terms of the two major operating system competitors it is quite a close call between them but Android comes out slightly stronger having a wider overall reach over Apple. To help make the decision personal knowledge and research was used to weigh up the advantages and disadvantages of each which can be seen in figure 1 and 2.

**Figure 1: Advantages of Android, IOS and Native Web Apps[[1]](#footnote-1)**

|  |  |  |
| --- | --- | --- |
| **Android** | **IOS** | **Hybrid / Web Apps** |
| Already experienced in Android development | Large amount of resources available for help | Works on both operating experience |
| Not as many recipe applications available on android hence more market share for the application | Popular operating system | Has some experience of html/css with minor javascript experience |
| Large amount of resources available for help | Standardized marketplace | Save time as they port to multiple platforms |
| Standardized market place | Easier to build nicer UX/UI features | Merge web and native features |
| Easier to build nicer UX/UI features | Make use of own hardware and software features | Consistency between apps |
| Make use of own hardware and software features |  |  |

**Figure 2: Disadvantages of Android, IOS and Native Web Apps1**

|  |  |  |
| --- | --- | --- |
| **Android** | **IOS** | **Hybrid / Web Apps** |
| Only covers one OS | Only covers one OS | Little help available, it’s relatively new in comparison to Android or IOS |
|  | No experience with ios or objective C development | Complex to set up and fidgety |
|  | Need to own a mac to develop so would only be able to work from computing building | No centralized market place. |
|  |  | Often work arounds are needed when porting to different apps |
|  |  | Web apps cannot work offline |

With Android leading the worldwide market share in smartphones in the last year and having the largest amount of advantages with the smallest amount of disadvantages Android was a naturally choice. Although hybrid/web apps were also a strong choice but the main reason for not selecting this option was the lack of support available at the moment was a concern especially for a large project like this. The operating system that was selected was Android and the device that was used to test the application was a Samsung Galaxy Tab S4 as Samsung is the largest Android vendor according to IDC.com (IDC, 2014), so it felt best to test on a Samsung device.

**Development Tools Selection**

To develop on Android several design decisions had to be made. The main decision is the IDE to develop on, there are two main IDE’s.to choose from which is Eclipse and Android Studio. The default is Eclipse with Android Studio being an intellij editor which is set to replace Eclipse at some point in the future but is currently in its beta stage. The choice for the project was Eclipse due to the fact Android Studio is still it’s beta stages and may be less stable than Eclipse which is an important factor when taking on a large project, it’s better to work with something stable and has support.

The emulator selection for the project is Genymotion. Genymotion is an android emulator which is trusted by 1500000 developers (Genymotion, n.d.). This is the alternative in comparison to the Android emulators provided and from past experience of using both there is a significant difference in speed and Genymotion is a lot quicker to use.For unit testing Junit will be used as the android test suites are based on Junit (Android, n.d.) And refactoring would be done on the project based on the refactoring rules that can be found in the appendix.

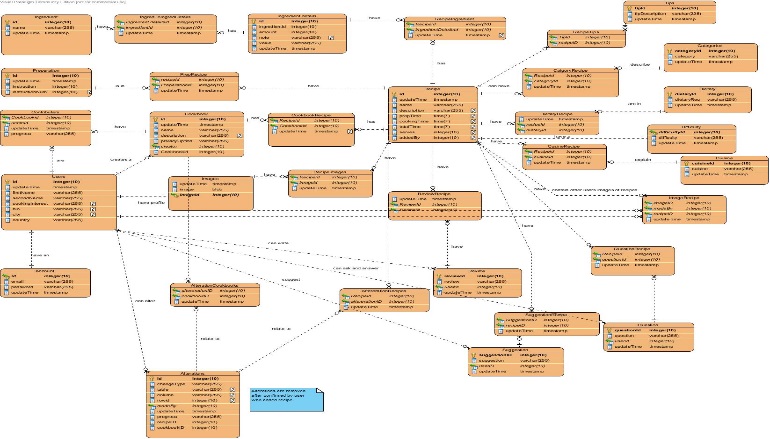
The versioning control system that was to be used for the project as Github. It enables you to access files anywhere, revert to old versions and store a range of files whether it’s code or a word document. The Github GUI feature also it is easy to manage commits without having to use the console. Already having experience with Github from several modules project work and also having a product account made it an appropriate choice. Also with over 6 million people using Github there is a lot of support available if struggling with any Github features (Github, n.d.).

**Database Design**

The application will be storing and handling a lot of data, so database consideration was a large section of the design stage of the project. Android has a SQLite database built into the phone which enables you to query a database on the phone without having to connect to the internet enabling the application to work offline. The developer can also have an option of syncing the SQLite database with another SQLite database or SQL database on the server when there is internet available. Working offline was a consideration when designing the application as users should be able to access recipes or grocery lists when offline and be able to access social aspects as new recipes added when internet is available. Based on the knowledge of Android having a built in SQLite database it seemed as if that would a natural choice but then from research found that CouchDB offered an option to have a NoSql database on the Android device. At first glance NoSql seemed like the best option for the data in the application as with SQL there was going to be a large amount of tables and quite a few complex joins. But then with closer analysis it seemed harder to visualise queries necessary for the application with CouchDB and with it being relatively new with lack of support and documentation for Android it just did not seem the right fit for the application.

The final database design decision was a SQLite database on the Android phone which would sync with a SQL database on the server. They would sync through passing xml or json via php. The syncing would occur on whenever an internet connection is available based on last changed time. SQL was choosen over SQLite for the server side mainly because SQLite doesn’t have a strong database browser that makes it easy to view large amounts of data without it costing large amounts of money whereas SQL has free database browsers that can handle this. The choice of database browser for SQL server is mysql and to view sqlite in Android using the SQLite database browser.

After these decisions were made the database tables and columns were designed. This was quite an iterative process and changed quite a few times based on design sketches or requirements. The database design can be seen in figure 3 and also in the appendix.



**Figure 3: Database Design**

**Project Planning**

Being such a large project various tools and techniques needed to be selected to help manage the project and make sure it’s on a track including the selection of a software development process. The software development process selected was the iterative approach. The approach enables you to gain all your initial requirements then do iterations where you take 2 or 3 high priority requirements develop them in a timeframe of 2 -3 weeks , test , refactor and evaluate. The approach enables you to maintain user centred design and approach throughout the project as well making sure it is well tested and maintain. The approach is a way to create a product with users in mind and enables flexibility with changing requirements and design making it easier to get the core aspects of the application done. The decision of the software development process was decided against two other approaches which was Waterfall and Agile. Waterfall was not appropriate as it is not flexible and has no room for user centred design. Agile was appropriate but was too client and team focussed which was not relevant to the project. Iterative has a lot of similar principles to Agile but is less client focussed such as there is no team retrospectives or daily stand ups so iterative seemed a better option for the project.

To manage the project various tools were selected. Trello was used to manage the project requirements as it is easy to categorise requirements into sections, colour code and move them around which is great when using an iterative approach where requirements could be changing. Minutes was used to track supervisor meetings which were a good way to see progress being made each week and log books were used to track notes, ideas, thoughts as well as document daily progress. Github also logs progress as for the project as it’s easy to see when things are being committed.

A project plan was created early in the project to document the overall plan for the supervisor. This was then moved into Gantt chart to help visualise all the tasks and milestones for the project over the year, the gantt chart was used as a project overview. For each of the development iterations a backlog would be used to split up requirements into small tasks so it was easy to see progress being made in iterations and what still had to be done. A risk assessment was also created for the project to help understand the risks could occur and how to manage them.

**Application design**

The application will be designed with a Model-View-Controller pattern as the application is heavily database focussed and enables me to keep the business logic and view separate. The benefits of the pattern is that it limits code duplication and enforces code re-use making code more flexible and easy to test functionality independently (Kotek, 2002)

**Design Sketches**

Initially when coming up with the project idea and making design decisions, a few design sketches were drawn on paper. Then after all the design decisions were finalised the design sketches were moved from paper to Axure which would then be used to gain users opinions before development started. The design sketches enabled an iterative process as it inspired new requirements and caused changes to the database design that was not originally thought of. In figure 4 is an image of the paper and axure design sketches. More can be seen in the appendix.

****

**Figure 4 : Design Sketches**

**Ethics**

Ethics designed for various user participation

**Target Market**

Personas created

**Implementation**

**Evaluation**

**Final Product**

**Critical Appraisal**

**Summary & Conclusion**

**Future**

**References**

**Appendix**

1. (Budiu, n.d.)**,** (Gorbsky, 2013)**,** (McCracken, 2013) [↑](#footnote-ref-1)