# CHAPTER 1

# PROBLEM SETTING AND BACKGROUND

## Introduction

Many people scan through cookbooks to search for new recipes to cook. Some may enjoy scanning through pages and some may find it hassle and tiresome. Buying cookbooks is also expensive and it requires time to read them all. Hard bound cookbooks could provide a wide range of recipe choices but these recipes might have ingredients that are not available in your kitchen. And also, what if you already have ingredients ready in your kitchen and you want to know what you can do with them. A typical cookbook is of minimal help in these situations.

With the technology today, there are a lot of open source electronic recipe programs going around the internet available for download, but only having a little advantage when compared to book. Others may also require internet connection in order to launch the application. In other words, there are not so many available programs that will be both helpful in keeping tracks of ingredients and suggest recipes based from the ingredients

To take advantage of the technology today, the proponents have decided to create a mobile application that will suggest recipes that are ready to cook based on the ingredients it contains that matches up in the inventory. This inventory is where the user must enter and update the available ingredients in the kitchen for use. With the use of the inventory, the user can track the available ingredients for cooking.

## Statement of the Problem

This study aims to develop a mobile application in which will suggest to the user decide on what dishes are possible to prepare. This research sought to unravel the following:

* A cook’s basic problem is what to cook with his supplied ingredients in the kitchen.
* How will this project help the user to keep track of the ingredients?

## Goals and Objectives

This study aims to develop an application that can match ingredients from the inventory and suggest recipes. Specifically, this study aims to:

* develop a native mobile application that would run on an android phone where:
  + the user is able to:
    - add ingredients into the inventory,
    - delete ingredients from the inventory,
    - update ingredients in the inventory, and
    - browse recipes available for cooking.
  + the system:
    - will suggest recipes to the user based on the ingredients present in the inventory,
    - will store ingredients into the database for tracking.

## Significance of the Study

This study would be beneficial to the following:

To the cooks, that they may be able to use the output of this study to minimize the problem in deciding which dish to prepare.

To the community, that they may find the study helpful for the advancement of technology.

To the future researchers, that they may be able to use this study as a reference. That this may also serve as a guide or sample for their future endeavors.

## Scope and Limitation

The proposed project has the capability of suggesting recipes based on the ingredients present in the inventory, which is manually inputted by the user. The user may also update his/her inventory if he/she wishes to. This will be implemented using Phone Gap that supports Android SDK and web standards such as HTML, CSS and JAVASCRIPT. Though this project has these capabilities, there are also some limitations regarding the usage of the application.

The application is specifically intended for one user per household only. The user will have to enter the available ingredients in the kitchen in order for the application to suggest recipes to choose from. Also the user will have to manually update the inventory whenever new ingredients are bought.

Because there is only diminutive amount of memory space available in a mobile phone, there will only be a maximum of 150 main courses recipes. The proponents opt to have few data to lessen the possibility of setback when the application is launched. These data will be stored in SQLite which will serve as the database.

## Theoretical Background

This study is supported by concepts and theories that are essential in the implementation of the project. The following are as follows:

### 1.6.1. Hypertext Markup Language 5 (HTML 5)

### HTML 5 is a formatting language that developers use to generate documents on the web. The latest edition HTML 5 has enhanced features for programmers; the developer can do audio and video tagging. HTML 5 is more convenient since it has a standardized way in embedding video on a web page. It also allows direct access to the graphics card, much faster load times and much smoother operation on PC.

### 1.6.2. JQuery

### JQuery is a new kind of JavaScript Library. JQuery is a fast and concise JavaScript Library that simplifies HMTL document traversing, event handling, animating, and Ajax interactions for rapid web development. JQuery is designed to change the way that you write JavaScript.

### 1.6.3. Cascading Style Sheet (CSS)

CSS is a web-based mark-up language used to describe the style and layout of a website to the browser. CSS 3 is the latest standard for CSS with additional capabilities. CSS 3 is the presentation layer of a web page since it provides impressive visual effects, better user interface and cleaner pages that load faster than before.

### 1.6.4. JQuery Mobile

JQuery Mobile is unified HTML 5-based user interface system for a mobile device platform built on the JQuery and JQuery UI foundation. It has a flexible, and easily themeable design. JQuery mobile framework is a cross-platfrom with HTML 5, it allows you to design a single highly-branded application that will work in all popular smartphone, tablet and desktop.

### 1.6.5. PhoneGap

PhoneGap is a free and open source framework that allows you to create application using standardized web APIs for the platforms, it uses HTML, CSS and JavaScript. It can support the following features: Accelerometer, camera, compass, contacts, file, geolocation, media, network, notification (alert: sound, vibration), and storage.

### 1.6.7. Eclipse

Eclipse is a multi-language Integrated Development Environment (IDE). It has a plug-in called Android Development Tools (ADT) which is designed to give powerful, integrated environment in which to build Android applications. ADT extends the capabilities of Eclipse to let the developer to quickly set up new Android projects, create and application UI, add packages based on the Android Framework API, debug applications using Android SDK tools, and export apk files to distribute the applications.

## Definition of Terms

**Mobile Application –** asoftware that is designed to help the user perform a specific task.

**HTML 5 –** formatting language that programmers and developers use to create documents.

**Database –** a systematically arranged collection of data for one or more purposes in digital form.

**User –** is a client, possibly anyone, who is in any way directly using the system regardless of privilege, restriction and role.

**User Interface –** is the space where interaction between humans and machines occurs.

**Platform –** where the system will run on.

**Application Programming Interface (API) –** it specifies how some software components should interact with each other.

**Software Development Kit (SDK) –** a toll for creating application.

**System –** is a set of interdependent elements that together accomplish specific objective.

**Inventory –**is a detailed, itemized list, report, or record of things in one’s possession.

# CHAPTER 2

# REVIEW OF RELATED LITERATURE AND STUDIES

## 2.1. Recipe Recommendation using Ingredient Network

Recipe Recommendation using Ingredient Network was a study conducted by Chun-Yuen Teng, Yu-Ru Lin and Lada A. Adamic last 21st of May 2012.

This study is regarding about the suggestion of recipes to the user. The study tackles on ingredient substitutes, ingredient recommendations, ingredient deletion and addition, ingredient adjustments and recipe pair prediction.

In relation to the proponents’ study, the same concept will also be applied about recipe recommendation through what ingredients are present.

## 2.2. Home Library

Home Library was a project made by Shahab H. Farooqui. The system of this scheme helps store books and CDs by scanning its barcode. It can keep track of all the books and CDs that are not lent and lent to people.

In relation to the proponents’ study, the system will also keep track of some data, which are the ingredients, present in the inventory that will be used for cooking.

## 2.3. Inventory Matching Software

Inventory Matching System was a project lead by Michael Lajeunesse of Medical Materials Incorporated. The system is design for the materials used in hospitals. It allows Medical Materials to match the materials the current materials a hospital is using and the available inventory from Medical Materials.

The proponents’ study will also be using an inventory system. Though, it may contain a different data compared to this project, the same idea is applied. The application will be able to provide a list of possible recipes according to the ingredients that matches in the inventory.

## 2.4. CookBook: An Android Mobile Platform Application

The CookBook: An Android Mobile Platform Application was a study conducted by Radu Popovici, Sergiu Petrescu, and Lorina Nereanu last January 2009.

This project presents a cookbook application designed for Android mobile platform. It mainly focuses on: search recipes by ingredients, shopping list for ingredients, and exchange of recipes. The user of the application has the freedom to choose between these features. Searching recipes by ingredients allows the user to search recipes based on the ingredients it contains. The shopping list manager generates a shopping list of ingredients for the recipe that the user decided to cook. Exchange of recipes is possible via e-mail or text messages.

The proponents used this study as reference for searching recipes. The same concept will be applied in the proponents’ study which the application will suggest a list of recipes based on the available ingredients in the kitchen. However, the proponents decided that there will be an additional inventory system available for the user to keep track of the ingredients.

## 2.5. Android as a Platform for Database Application Development – Case: Winha Mobile

The Android as a Platform for Database Application Development – Case: Winha Mobile was a Bachelor’s thesis studied by Joseph Kyalo Muli of Truku University of Applied Sciences last February 14, 2013 under the guidance of instructor Tina Ferm.

This study focuses on the integration of database in an Android Operating System, specifically on mobile devices. Android Operating System & Database Package were discussed, as well as its application fundamentals, components and resources. At the end of the study an output was produced called Winha Mobile where records may be inserted, updated, viewed, and deleted. Through this output the goal of this project was met.

Through this study, the proponents were given the idea of how to implement the use of database for an android phone. The application to be developed by the proponents will have an inventory system for keeping track of the user’s ingredients which will be stored in a database.

## 2.6. Building Android Apps with HTML, CSS, and JavaScript

Building Android Apps with HTML, CSS, and JavaScript is a book written by Jonathan Stark that was published on September 2010.

This book simply serves as a guide for the reader on how to form an Android application using HTML, CSS, and JavaScript step-by-step. Also, PhoneGap is also discussed in this book in which it focuses on its Android branch.

This book will be helpful for the proponents’ project in the implementation of creating the mobile application with HTML, CSS, and JavaScript, and also as a guide for using the PhoneGap development tool.

## 2.7. Supercook.com

Supercook.com is a web-based recipe search engine that lets you search recipes by ingredients you have at home. Supercook can help you save hundreds on grocery bills by showing you how to fully utilize the available ingredients you have at home. Supercook, an internet is needed for you to be able to sign up and use their website. Recipes suggested are linked from different sites such as allrecipes.com and epicuriuos.com. When adding items and ingredients, the system automatically begins to suggest recipes based on the inputted ingredients. It allows you to search recipes by name, save favorite recipes, choose restrictions and categorizes the type of recipe you want to make. It also suggests whether to add one or two additional items to make a dish and allows you to discover more foods. Overall, Supercook is a simple and easy to use web-based application.

The proponents used Supercook as reference for their project because the project will also use almost the same concept as a recipe generator. Some function were omitted since there limitation to the project. The proponents’ project, doesn’t need and internet connection in order to run the application. It will be much more convenient for the user since it will run on a mobile device.

## 2.8. Case Study: Recipe Search MicroApp

Recipe Search MicroApp was a study conducted by Daniel Levitt which results to a web based recipe generator [www.whatcouldicook.com](http://www.whatcouldicook.com). The application runs in an open platform (guardian.co.uk) and provided all the recipe articles via Content API which made it possible for to be retrieved and republish it. The user are able to search by ingredients and search by name. Through guardian.co.uk, when the user searches for recipes, the machine of guardian.co.uk calls the application which is hosted remotely then returns the results the application retrieves within the guardian.co.uk domain pages.

This study is almost similar to the proponents’ study in how recipes are generated for the user. This study may have assumed that there is an internet connection when the application starts to generate recipes, the proponents’ study will do the opposite. There will be no need for an internet connection needed for the application to be able to generate recipes.

## 2.9. Pinoy MD: A Mobile Dietary Aide and Recipe Generator

Pinoy MD: A Mobile Dietary Aide and Recipe Generator was a project by de Gula, Erika Pauline, et . al., to be published in the International Journal of Future Computer and Communication, Vol. 2, on October 2013.

This project proposed a mobile application running in Android Operating System designed to generate recipes depending on the selected common Filipino food based on a person’s ailment. This project’s aim was to determine the acceptance rating of the nutrition experts on the system by its accuracy and reliability, and also, the rate of acceptance to the respondents by determining its user friendliness and the functionality of its system.

In comparison to the proponents’ project, the same idea will be applied in which a mobile application will be developed. Though the proponents will use a different approach in coding the program, it will also be functional in an Android Operating System.

## 2.10. Mobile Application Development: Web vs. Native

Mobile Application Development: Web vs. Native was an article written by Charland, Andre and Leroux, Brian published on May 2011.

This article discussed that most native platforms have wonderful abstractions for common user-interface controls. Web code uses HTML and CSS to create user interfaces in web views and browsers while native code, it is painted pixels on screens through the propriety APIs and abstractions for common user-interface elements and controls.

In relation to the proponents’ project, both native and web codes will be used for the development of the application.

## 2.11. The Betty Crocker Mobile Cookbook

The Betty Crocker Mobile Cookbook was project made by General Mills Incorporated. The system allows user to search recipes by entering the ingredients the user have. A database was used for the storage of the recipes. Also, the recipes come with photos, a step by step instructions and nutritional information.

In relation to the proponents’ project, the system will also generate recipes according to the ingredients entered by the user.

## 2.12. The Wait is Over

The wait is over was an article written by Andel, Tom on June 2013. This article’s concern was about the paperless electronic system being used in the food chanel logistics of food distributor Tony’s Fine Food. It was stated that the system was a cloud-based mobile application, and it is used by the driver in each stop to scan products and to compare the invoice. According to Mark Geery, chief information officer of Tony’s Fine Food, the electronic system has improved their tracking process as well as reduced paperworks.

Although this article covers a cloud-based system, it has the same impression with the proponent’s project. A mobile application will be developed by the proponents that will have an inventory system for the tracking of ingredients.

## 2.13. Allrecipes: Mobile is big factor in driving global growth

Allrecipes: Mobile is bir big factor in driving global growth was an article written by Rimma Kats published in April 13, 2013. This article states that online recipe sites continue to be the number one resource for home cooks. A survey done by allrecipes supported this statement. The company has surveyes cooks worldwide to capture the global digital food trend. During the “2013 Global Trends Measuring Cup”, it was concluded that mobile is becoming a big factor in global growth. Most of the global consumers nowadays use smartphones for seeking meal solutions and finding recipes online.

This article had given the proponents the idea on the effect of online recipe sites to the community. But instead of creating a web-based application, the proponents decided to create a stand-alone mobile application. The application will give more convenience to users because they will only have to enter their available ingredients in their kitchen then recipes are displayed based on those ingredients. Moreover, the ingredients will be stored in an inventory system where it can be tracked.

## 2.14. Going Electronics

Going Electronics was an article written by Natalie Danford last August 6, 2012 in Publishers Weekly. This article talks about the change of the cookbook category. Cookbooks has made a serious progress as years pass by, from hard-bound to e-books and in addition to that, there are now mobile applications. Recipes and photos are a jumping-off point. Today’s cooks are increasingly, relying on electronic devices, this indicates that cookbook consumers are becoming comfortable with the electronic format.

This article gave the proponents the knowledge that even professional chefs today are now adapting to change of cookbooks. With this, the proponents will be creating a mobile application in relation to cooking that would give the users convenience and a new experience.

# CHAPTER 3

# METHODOLOGY

## 3.1. Project Management

### 3.1.1. Project Assumptions

During the implementation of the project, the proponents assume:

1. That each proponent will be given specific tasks equally and will be finished on time.
2. That each proponent has basic knowledge of the ff:
   1. HTML
   2. CSS
   3. JavaScript
3. That the application will be fully functional when launched in an android phone.

### 3.1.2. Project Constraints

1. Looking for a work place.
2. Not enough time.
3. Differences in opinion in which might cause a delay in the development of the project.
4. Unable to decide immediately.
5. Availability of members.

### 3.1.3. Risk Assessment and Analysis

Risk pertains to the potential that a chosen activity that will result to an undesirable outcome during the implementation of the system.

The table shows the risk identified by the researchers as listed according to severity:

Table 3.1 Risks with high severity

|  |  |  |  |
| --- | --- | --- | --- |
| **Risk** | **Cause** | **Contingency Plan** | **Probability** |
| **Risks With High Severity** | | | |
| Accidents | Unexpected events | First aid or go to hospital | High |
| Data loss | Unable to secure back up for files | Secure back up constantly | High |
| Bugs found after developing the system | Coding problems, unable to test properly the system | Test the system properly | High |
| Health Problems | Stress, improper diet, lack of sleep | Take vitamins, find time to sleep, time management | High |
| Distractions | Lack of focus | Work on less distracting places | High |

Table 3.2 Risks with medium severity

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Risk** | **Cause** | | **Contingency Plan** | **Probability** |
| **Risks With Medium Severity** | | | | |
| Power Loss | | Brown outs, Calamities | Pray | Medium |
| Failure to deploy on time | | Unable to follow schedule | Be aware of due dates and follow the schedules given | Medium |
| No teamwork and cooperation between members | | Character or attitude problems, stress,  menstruation(girls) | Stick to the plan, continue to work, adjust to the attitudes | Medium |
| Changing of preferences during developmental stage | | Unable to comply the software properly and changing of scope of the project | Stick to the scope of the system | Medium |

Table 3.3 Risks with low severity

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Risk** | **Cause** | | **Contingency Plan** | | **Probability** |
| **Risks With Low Severity** | | | | | |
| Loss of Adviser | | Conflict of schedule,  Resignation of adviser | Look for replacement or  Continue to work | Medium | |
| Loss of member | | Death, stop schooling | Continue to work and share the work load to other members | Low | |
| Attitude problems and conflict of interests | | Character or attitude problems, stress,  menstruation(girls) | Stick to the plan and continue to work | Medium | |

### 3.1.4. Team Structure

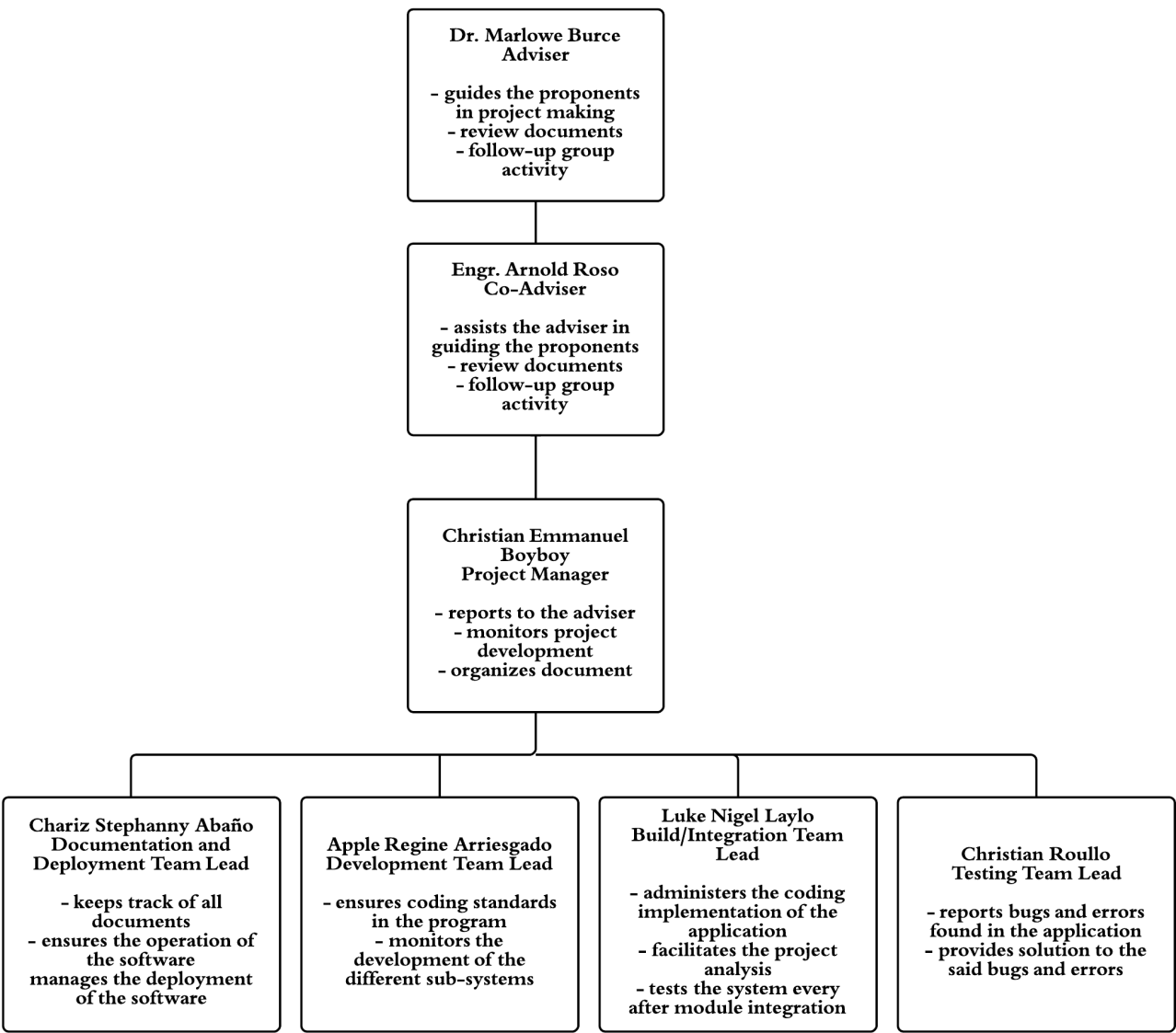
The team structure presented in Figure 3.1 represents the duty and responsibilities of each proponent in the implementation of the project. The figure presented below shows the breakdown of responsibilities that will be performed by each proponent.

Figure 3.1 Team Structure

### 3.1.5. Project Resources

The following resources have been identified by the proponents in the implementation of the project. Moreover, these resources are divided into: Human Resources, Software Resources and Hardware Resources. The following shows the breakdown of the project resources.

**Human Resources**

1. Proponents and developers
   1. Chariz Stephanny Abaño
   2. Apple Regine Arriesgado
   3. Christian Emmanuel Boyboy
   4. Luke Nigel Laylo
   5. Christian Roullo
2. Adviser
3. Dr. Marlowe Burce
4. Co-Adviser
5. Engr. Arnold Roso
6. Client Testers
7. Students
8. Teachers
9. Parents

**Hardware Resources**

1. Computers
2. Smartphones
3. Tablet

**Software Resources**

1. Phonegap
2. Microsoft Office
3. Windows 7
4. SqlLite Database
5. Google Chrome and Mozilla Firefox (Latest versions)
6. Jquery Mobile

### 3.1.6. Work Breakdown Structure

The following is the work breakdown structure during the implementation of the system. The work is divided into two parts, the *System Initialization* and *System Implementation*.

* ***System Initialization*** refers to the planning stage of the system. The research and planning took place during this stage. Documentation of the proposed system will be produced after this stage.

The following are the process during system initialization:

* **Project Proposal** refers to the proposal of the system to the panel.
  + **Capsule Proposal** is the making of the documents for the proposal of the system. The system’s goals and objectives as well as the system’s scope and limitations are discussed in this part.
* **Project Planning** refers to the stage where planning of the system and brain storming is done. A brief description of the system is created and possible clients are identified.
  + **System Analysis and Design** is where the system is analyzed and discussed thoroughly. Based on the partial plan, the system is designed, taking into consideration what are its functions, procedures and implementation according to the software and hardware requirements.
* **Project Preparation** refers to the process where the preparation on what to do on the project is done.
  + **Data Gathering** refers to the process where the needed information and resources are gathered, including the study of related works in order to create a partial plan for the system.
* **System Documentation** refers to the process where the description and plans for the system is documented as basis for the implementation of the system.
* **Presentation Preparation** is where the preparations for the presentation of the completed system are done. Visual aid and report materials are created and prepared in this stage.
* **Presentation Proper** refers to the final stage of the system implementation where the completed system is presented and reviewed.
* ***System Implementation*** refers to the development phase of the system. Division works and coding of the system as well as the testing is done on this period. And evaluation of the gathered data will be done after the system has been deployed. Documentation of the developed system will be produced after this stage.

The following are the processes during system implementation:

* + **Software Development** refers to the development of the system.

The following are the sub-processes of the software development:

* + - **UI Designing** refers to the designing of the different pages of the system.
    - **Function** refers to the functions of the system, the different algorithms and the processes are done in this process
    - **Database Construction** refers to the developing and designing of the database of the system.
  + **Software Integration** refers to the process where the individual modules are combined to form the complete project.
  + **Software Testing** refers to the process where the project is tested through a number of tests.

The following are the sub-processes during the system testing:

* **Unit Testing** refers to the individual modules testing using the test cases specific to each module.
* **Integration Testing** refers to the testing of the system after integrating all modules.
* **Alpha Testing** refers to the testing of the system by an outer testing group.
* **Beta Testing** refers to the testing of the system on the developer’s site by the potential clients.
* **User Acceptance Testing** refers to the testing of the system after it has been deployed to the client.
* **Usability Testing** refers to the testing of the project to verify that the project is user-friendly, which means easy to use and understand.
* **System Review** refers to the evaluation of the test results. Adjustments of the system are to be made after the evaluation if necessary.
* **Deployment and Installation** refers to the process where the system is released and delivered to the users.
* **Evaluation** refers to the process where the system is evaluated through the use of surveys which will be given one month after deployment in order to get the users feedback about the system.
* **System Documentation** refers to the process where the whole implementation process is documented, as well as the evaluation from the users and recommendations from the developers.
* **Presentation Preparation** is where the preparations for the presentation of the completed system are done. Visual aids and report materials are created and prepared in this stage.
* **Presentation Proper** refers to the final stage of the system implementation where the completed system is presented and reviewed.

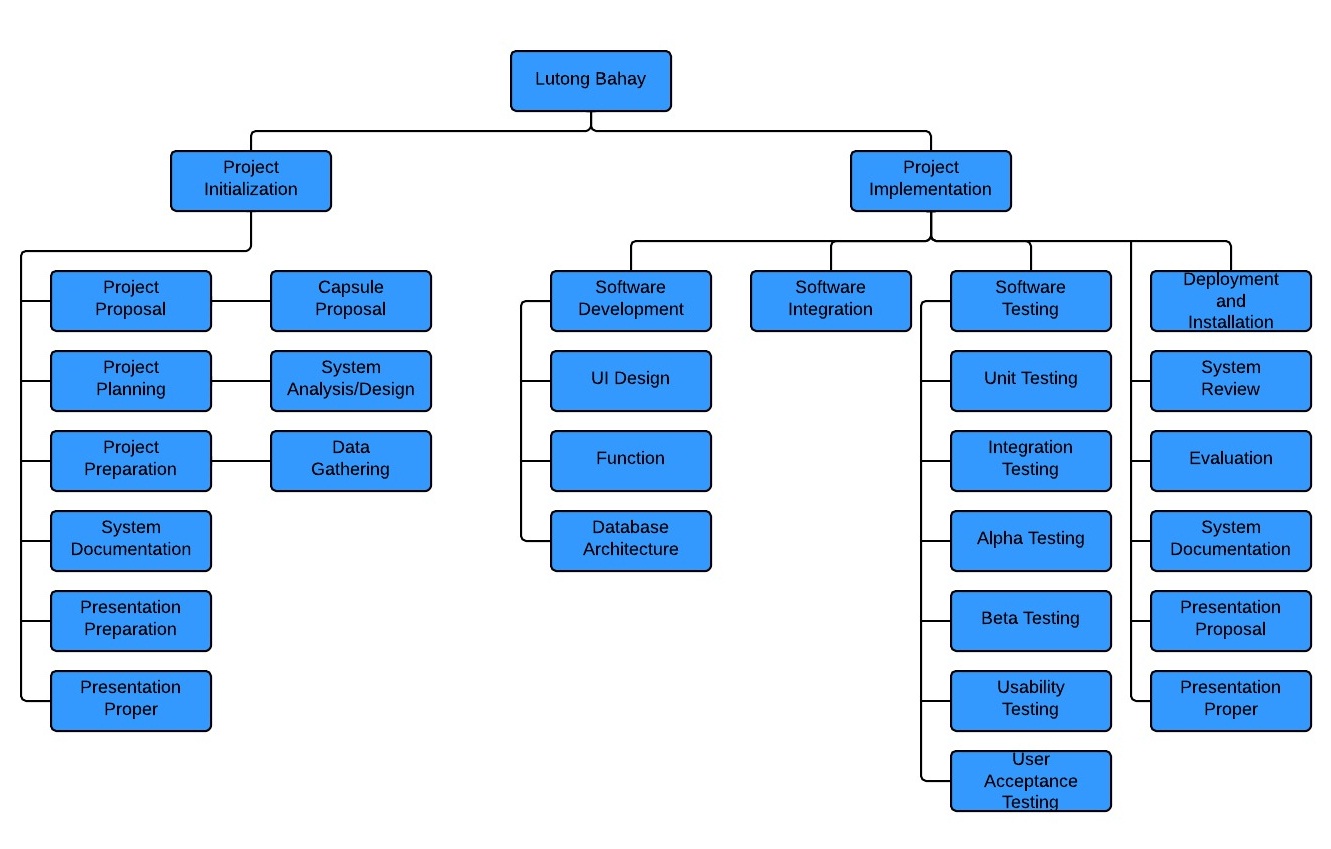


Figure 3.2 Work Breakdown Structure

Figure 3.2 is the work breakdown structure of the system. The work is divided in to two parts, the *System Initialization* and *System Implementation.*

### 3.1.7. Network Diagram and Critical Path

Figure 3.3 shows the format that will be used in representing a task in the implementation of the system.

|  |  |  |
| --- | --- | --- |
| Early Start | Duration | Early Finish |
| TASK | | |
| Late Start | Slack | Late Finish |

Figure 3.3 Program Evaluations and Review Technique (PERT) 1 Format

* **Early Start** refers to the earliest time that the task can start with all its predecessors have been accomplished.
* **Early Finished** refers to the earliest time that the task can be done.
* **Late Start** refers to the latest time that the task can be finished.
* **Slack** refers to the time that the task can be lagged by without delaying the project.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Task | Task Name | Predecessor | Duration | |
| A | Project Proposal | - | 11 days | |
| B | Project Planning | A | 25 days | |
| C | Project Preparation | B | 12 days | |
| D | System Documentation | C, B | 13 days | |
| E | Proposal Presentation Preparation | D,C | 5 days | |
| F | Proposal Presentation Proper | E | 1 day |
| G | Software Development | F | 55 days | |
| H | System Integration | G | 15 days | |
| I | System Testing | H | 10 days | |
| J | System Review | I | 10 days | |
| K | Deployment and Installation | J | 7 days | |
| L | Evaluation | K | 30 days | |
| M | System Documentation | L | 18 days | |
| N | Presentation Preparation | M | 7 days | |
| O | Presentation Proper | N | 1 day | |

**Table 3.4** Task List

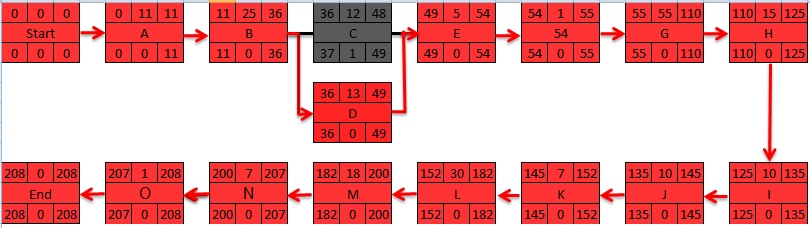


Figure 3.4 Network Diagram and Critical Path

Figure 3.4 shows the network diagram of the research where the Critical Path is shown in red arrows. The Critical Path determined the earliest time that all tasks can be done and delay in any of the tasks in the path delays the overall schedule.

### 3.1.8. Project Schedule

Figure 3.5 serves as a guide in the implementation of the system. It shows the detailed breakdown of the task list, as well as the assigned proponents for the specific task.

Task 3.5 Project Schedule

|  |  |  |  |
| --- | --- | --- | --- |
| Task Name | Owner | Duration |  |
| 1. Project Proposal    1. Capsule Proposal Making | All | 11 days  11 days |  |
| 1. Project Planning    1. System Analysis/Design | All with Adviser | 25 days |  |
| 1. Project Preparation    1. Data Gathering | All | 12 days |  |
| 1. System Documentation   4.1 Document Making  4.2 Submission of Documents | All | 13 days  12 days  1 day |  |
| 1. Presentation Preparation    1. Mock Presentation | All with Adviser | 5 days |  |
| 1. Presentation Proper | All | 1 day |  |
| 1. Software Development    1. UI Design    2. Function    3. Database Architecture | All | 55 days  15 days  20 days  20 days |  |
| 1. System Integration | All | 15 days |  |
| 1. System Testing   9.1 Unit Testing  9.2 Integration Testing  9.3 Alpha Testing  9.4 Beta Testing  9.5 Usability Testing  9.6 User Acceptance Testing | Roullo, Laylo  Abaño, Boyboy  Arriesgado, Abaño  Arriesgado, Abaño  Arriesgado. Boyboy  Roullo, Laylo | 10 days  1 day  1 day  2 days  2 days  2 days  2 days |  |
| 1. System Review   10.1 Evaluation of Testing Results  10.2 System Adjustments | All | 10 days  5 days  5 days |  |
| 1. Deployment and Installation | All | 7 days |  |
| 1. Evaluation   User Acceptance Testing  Evaluation of UAT | Laylo, Roullo, Boyboy  Abaño, Arriesgado | 30 days  28 days  2 days |  |
| 1. System Documentation | All | 18 days |  |
| 1. Presentation Proper | All | 1 day |  |

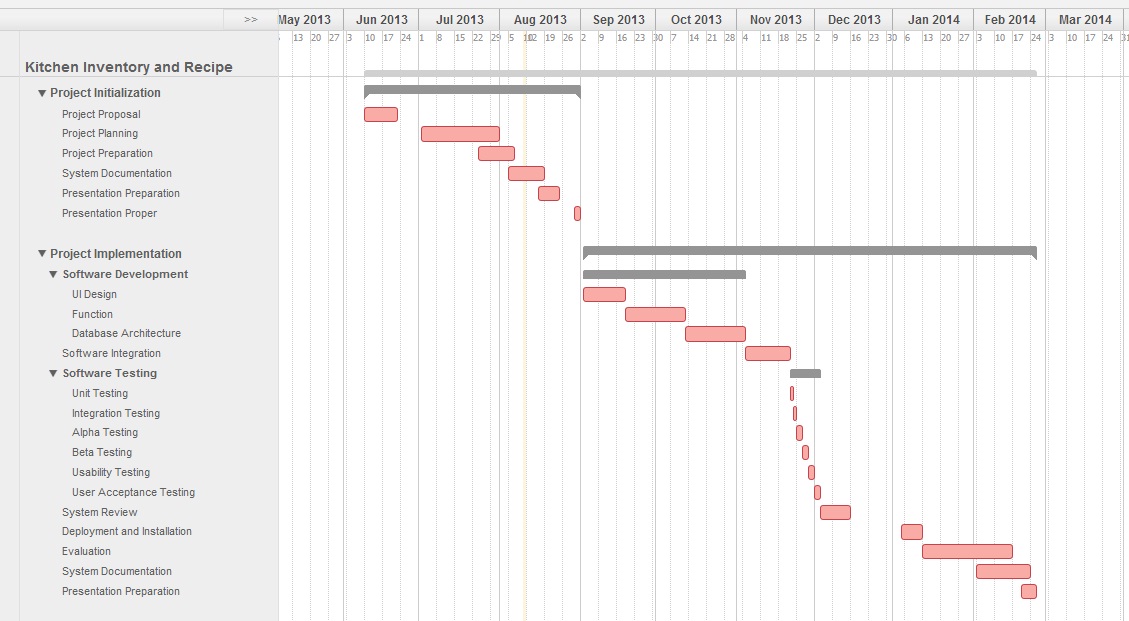


Figure 3.5 Gantt chart

Fig. 3.5 shows the Gantt chart of the project schedule. This shows a graphical timeline the proponents will be following in the implementation of the system.

### 3.1.9. Project Budget

Table 3.6 Sponsored Cost

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Item Description** | **Quantity** | **Price Per Unit** | **Supplier** | **Total Item Cost** |
|  |  | **Hardware Cost** |  |  |
| Desktop Computers | 5 | Php 15,000.00 | Computer Engineering Department | Php 75,000.00 |
|  |  |  |  |  |
| Tablet | 1 | Php 12,000.00 | Christian Emmanuel Boyboy | Php 12,000.00 |
|  |  | **Documentation Cost** |  |  |
| Printer | 1 | Php 3,200.00 | Christian Emmanuel Boyboy | Php 3,200.00 |
|  |  | ***Actual Cost*** |  |  |
| **Item Description** | **Quantity** | **Price Per Unit** | **Supplier** | **Total Item Cost** |
|  |  | **Documentation Cost** |  |  |
| Printing and Binding | 10 | Php 300.00 |  | Php 3,000.00 |
| Printer Cartridge | 8 | Php 390.00 |  | Php 3,120.00 |
| Miscellaneous |  | Php 2,000.00 |  | Php 2,000.00 |

## 3.2. Proposed Software Design

### 3.2.1. Hardware Requirements

The following are the suggested hardware requirements for the project to work properly:

* Smartphones or Tablet
  + Android OS version 2.2 or higher
  + Internal or External memory of at least 1 GB or higher
  + 1 GB of RAM or higher

User’s Side:

Any Smartphones or Tablets that runs an Androis OS version 2.2 up to the latest version with 1 GB of RAM and having an internal or external memory of 1 GB or higher.

### 3.2.2. Software Requirements

The following Software Requirements are the software tools used by the proponents in the development of the application:

* + HTML 5, PhoneGap, SQLite for Database, CSS, JavaScript, and JQuery Mobile

User’s side:

* + Android OS version 2.2 or higher

### 3.2.3. System of Software Limitations

The system is limited to one user per household only in its automation of adding, viewing, updating and deleting ingredients, and searching of recipes according to the present ingredients in the inventory. Human errors in cooking are also out of the system’s control.

The application can only be installed and launched in an Android OS.

### 3.2.4. System Process Model

Software process entails the structure of activities required to develop a software system. A software process model refers to the simplified description of organizing a project into activities. The proponents have used the Waterfall Development Life Cycle in developing the system. The Waterfall Development Life Cycle is used to describe as the classical Software Development Life Cycle. The follows the sequential process of the process model.

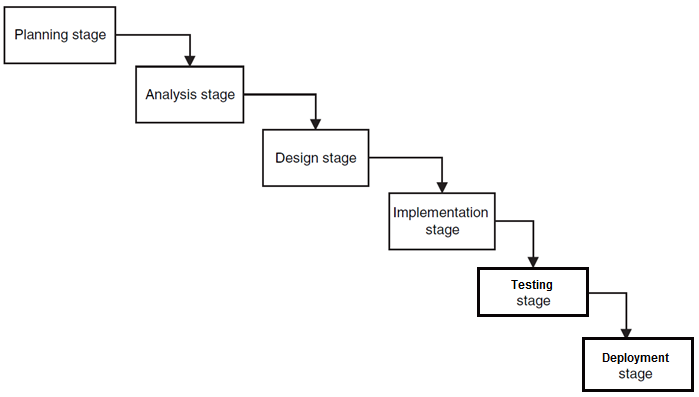


Figure 3.6 Waterfall Development Life Cycle Model

The following are the phases of the Waterfall Development Life Cycle Model:

* + **Planning Stage.** During this stage the proponents have pointed out problems and objectives with regards to the project.
  + **Analysis Stage.** In this stage the proponents have conducted a system study of some existing systems and analyze what possible requirements are missing in the software.
  + **Design Stage.** The proponents have formulated the flow of the system to be implemented and data to be used.
  + **Implementation Stage.** This stage is where the proponents develops each module and integrate to form the system.
  + **Testing Stage.** Verification and validation of the system is done in this stage.
  + **Deployment Stage.** This is where the proponents will make the software available for use.

### 3.2.5. Software Architecture

The software architecture of a system is a set of structures needed for the system’s operation which consist the software elements and hardware elements that can be used by the software, relationship and properties.

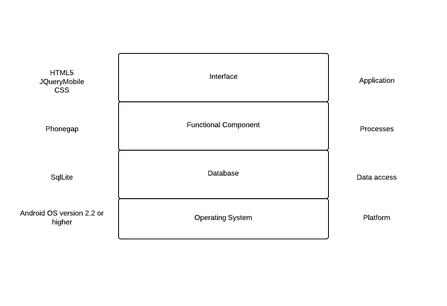


Figure 3.7 Software Architecture

Figure 3.7 shows the Software Architecture of the proposed project.

* **Application** is where the user-interface is presented. It serves as the graphics for the application to be more user-friendly. The application will use HTML 5, JQuery Mobile, and CSS.
* **Process** is responsible for the operation of the functions of the application. It serves as the logic of the application and the medium between the user interface and the database.
* **Data Access** is responsible for the storage of data. The application will be using SQLite for the storage of data.
* **Platform** is responsible to where the software would run on. The operating system serves as the framework where the application will run on. The application may run on an Android OS with a version of 2.2 or higher.

### 3.2.6. Software Analysis and Design

* + Use Case

Figure 3.8 shows what the user can do in the system, the user can view the home page of the system, can ad ingredient, update ingredient, look for recipes based on the ingredients in the inventory and can search the recipes base on the results of the generated recipes.

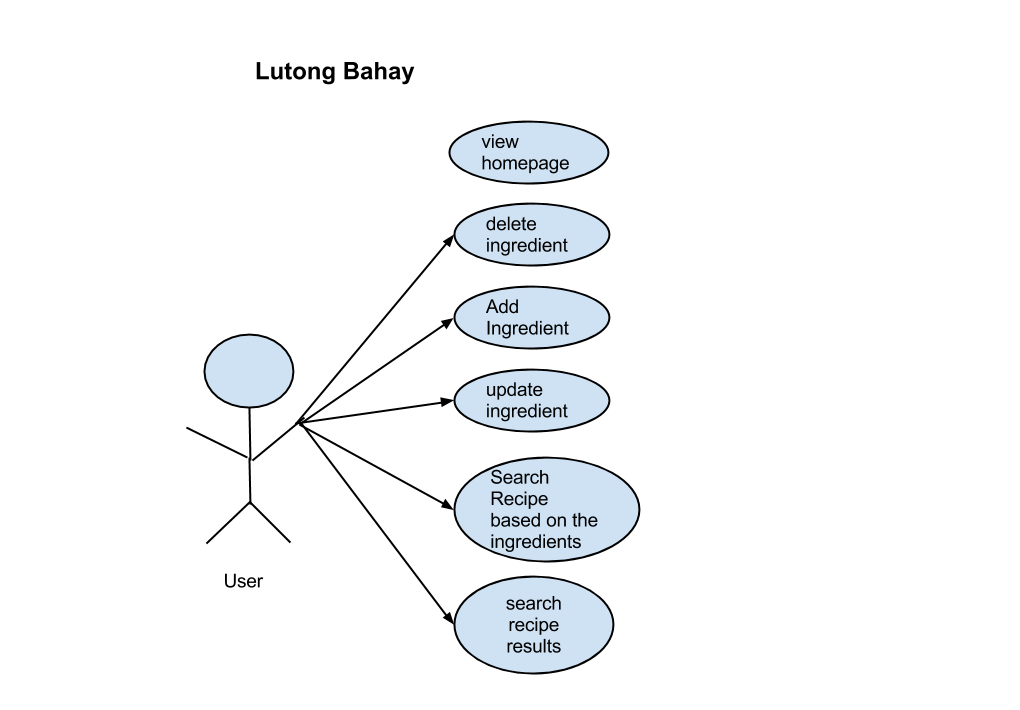


Figure 3.8 Use Case Diagram

* Flowchart

Figure 3.9 shows the main page of the program. The user has the option to choose from:

* + - Ingredients page is where the user can view the ingredients in the inventory.
    - Recipes page is where the user may search available recipes to cook based on the ingredients in the inventory.
    - Help/About page is where the user can view the about the details and the user guide of the application.
    - History page is where the user may view the list of recipes that has been recently cooked.

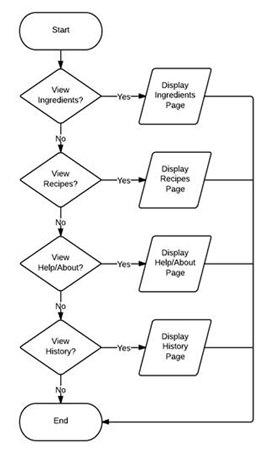


Figure 3.9 Main Flowchart

Figure 3.10 shows how the inventory system works and managed by the user.

* + - The system will display the available ingredients entered by the user.
    - User can search an ingredient present in the inventory.
    - User can add ingredients one at a time.
    - User can select an ingredient to view its information and may delete or update the ingredient.
    - User can go back to the main page of the system.

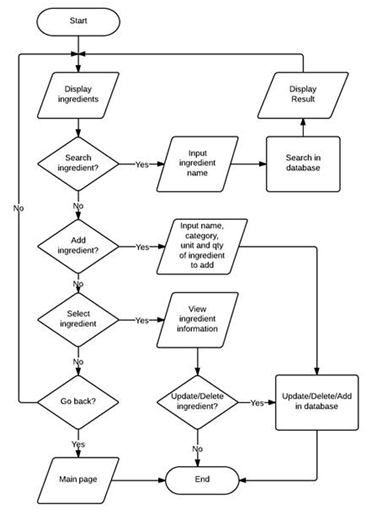


Figure 3.10 Ingredient Inventory Flowchart

Figure 3.11 shows that the user will have to input first the number of servings then the application will display the available recipes depending on the ingredients in the inventory. After which, the user can now:

* + - Search recipes from the result.
    - Select a recipe to view information then may decide whether to cook or not. When the user decides to cook the recipe, ingredients needed for cooking will be automatically deducted from the inventory. Then user may go back to be able to search recipes again.

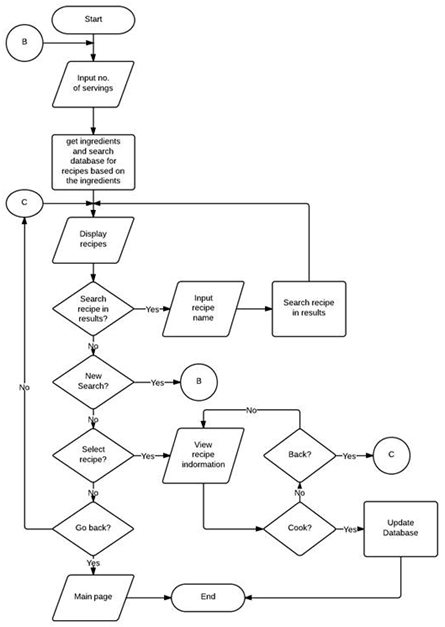


Figure 3.11 Recipe page Flowchart

* Entity Relationship Diagram

Figure 3.12 shows the database structure of the application. This is where the ingredients from the inventory and the recipes are being stored. It also shows the relationship between different database tables.

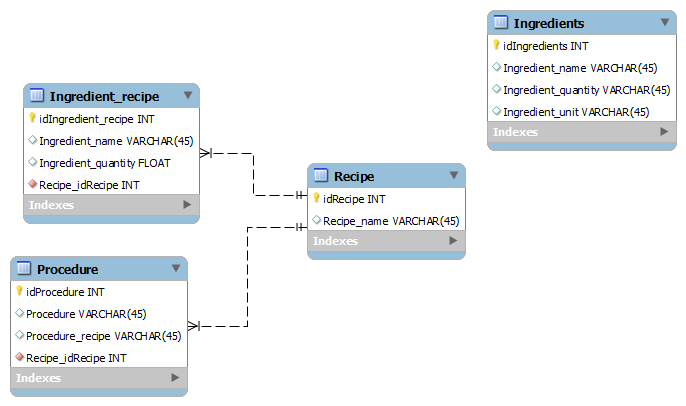


Figure 3.12 Entity Relationship Diagram

## 3.3. Software Testing

### 3.3.1. Unit Testing

The individual modules of the system will be tested according to the specific test cases.

* + - 1. **User module**

The User Module will be tested according to its performance to update the system’s information.

The following test cases will verify the module’s functionalities:

Table 3.7 Test Cases for User Module

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case** | **Description** | **Test Steps** | **Expected Results** |
| Manage Use | Test for managing the ingredient inventory of the user. | Add | Adds ingredients according to what the user inputted. |
|  |  | Update | Updates the users inputted ingredients. |
|  |  | Delete | All ingredients inputted are shown and can be deleted. |

**3.3.2. Alpha Testing**

The Alpha Testing is the type of testing that will be done by the developers themselves. This testing is done to recheck and find bugs and errors that were not found during the integration testing.

The following are the guidelines for the Alpha Testing:

* + User can input ingredients.
  + User can input recipes to search.
  + User can delete ingredients.
  + User can update ingredients.
  + User can enter the number of servings to search for recipes.
  + User can input the quantity with regards to the unit in SI.
  + User can view the guidelines and instructions.

**3.3.3. Beta Testing**

Beta Testing is a type of testing released to the clients. The system will be checked if the it meets the requirements of the clients. Output of this testing will help the proponents improve the system and also to find out bugs and errors that has not been found during alpha testing. The guideline in Alpha testing will also be used in Beta testing.

**3.3.4. User Acceptance Testing**

After the system has been deployed the User Acceptance testing will begin. This will be done by the participants in a specific location and feedbacks will be gathered.

The following guideline will be used to gather feedback from the participants during the user acceptance testing:

* Does the system provide a user-friendly user interface? Is the system’s layout comfortable?
* Does the system provide complete functions as specified? Are these functions fully functional?
* Are the data consistent throughout the system? Are these date correct and reliable?
* Does the system come with proper documents for review?

**3.3.5. Usability Testing**

Usability Testing verifies the user interface of the system if it is user-friendly and easy to understand. The feedback gathered from the participants will be used in evaluating the system’s user-friendliness.

Furthermore, the following guidelines will be used to verify the system’s usability:

* Does the system provide easy-to-use functions?
  + Do the functions come with easy to understand instructions?
* Is the system’s user interface comfortable to use?
  + Are the fonts used easy to read?
  + Are the colors used pleasing to the eye?

**3.3.6. Integration Testing**

Integration Testing is type of testing where all of the sub-systems are combined and tested as one.

Table 3.8 Test Cases for Integration Testing

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case** | **Description** | **Test Steps** | **Expected Results** |
| Add ingredient | Test if the user can add ingredient to the inventory. | Select “Ingredients Inventory”  Tab | Allows user to search for ingredients. Add button is shown. |
|  | Test if the user can input the quantity of the ingredient/s and its SI unit | Select  “Add” Tab | Allows the user to input the Quantity and SI Unit of the ingredient. The update and delete button are shown. |
|  | Test if the inputted ingredient is correct. | Select  “OK”  Button | Shows the text field of “name of the ingredient”, “category”, “unit” and the “quantity”. |
| Delete Ingredient | Test if the user can delete ingredients | Select an ingredient from the inventory then select “Delete” Tab | A window will appear to confirm deletion and will redirect back to inventory page if confirmed and ingredient will be deleted. |
| Update Ingredient | Test if the user can update ingredients | Select an ingredient from the inventory then enter new quantity of the ingredient. | Current quantity will be replaced by the entered quantity. |
| Recipe Search | Test if user can search for recipe. | Select “Recipes”  Tab | A window will appear to enter the number of servings and redirects to the recipe page. Search button and new search button is shown. |
|  | Test if user can search for more or a new recipe. | Select “New Search”  Button | Allows the user to search for another recipe |
| Cook a Recipe | Test if the system can suggest recipe base from the ingredients you inputted. | Select a Recipe from the searched results | Shows the ingredients you inputted and the procedure of the recipe. A cook button is shown. |
|  | Test if the ingredients in the inventory are deducted. | Select “Cook”  Button | Asks confirmation to the user to cook and deducts the ingredients inputted from the database. |
| View information |  | Select “About”  Tab | Shows information, guidelines and instructions of the application. |
| View cooked recipes |  | Select “History”  Button | User can view recently cooked recipes. |

## 3.4. Deployment and Maintenance

At the end of the software development and testing, the system will be deployed on the mobile to undergo user acceptance testing. Feedback from the users will be carefully evaluated to provide necessary updates for the system’s maintenance.

**3.4.1. Deployment**

Deployment involves the setting up of the system in the mobile for use and checking. The installation of the system may take several minutes as a whole. The target users are anyone who uses smartphones or tablets with Android OS.

**3.4.2. Maintenance**

Maintenance involves the periodic checking of the system while it is deployed on the mobile during the user acceptance test to check for errors in the system. It also involves updates to have a better performance.

Bugs and defects of the application are directly reported to the developers for which the developers will formulate a solution to fix it and prevent further problems in the application.

The developers are open to user-formulated suggestions and comments. These suggestions and comments are to be studied by the developers if it will be necessary for it to be part of the system.

## 3.5. Data Gathering and Interpretations

System evaluation is very essential to make room for improvement. This can be done through gathering and interpretation of user’s feedback. The interpretations would be a toll to knowing the system’s performance and would be the means to plan for the improvement of the system.

**3.5.1. Gathering of Data**

This study used the qualitative style research to gather information or feedback from the users. The analysis focused on the qualities of the mobile application. For quantification, a frequent count, as well as a tally will be used based on the user’s answers to the survey.

The gathering utilized the use of a questionnaire to acquire relevant data and feedback from the users. As a means to ensure the consistency of the answers, counterchecking questions were used.

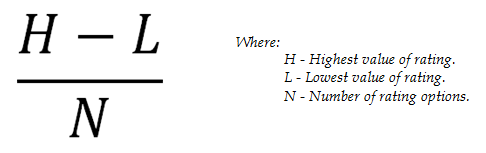
**3.5.2. Treatment of Data**

Since the proponents seek to measure the usability of the system, the proponents will conduct a heuristic kind of evaluation.

The system will be evaluated according to three qualities: User-friendliness, Portability, and Completeness. Each of these qualities was rated by the users with a spectrum ranging from one to four (1-4), one (the user strongly disagrees), two (disagree), three (agree), four (strongly agree).

The questionnaires will be answered by the users and the answers will be sorted by quality. Their comments, suggestions, reactions and feedbacks regarding quality of the mobile application will be accounted and tabulated where the frequency are shown. The tabulated results will be calculated for its average using weighted averaging method.

In order for the researcher to know the average rating of each quality, a range computation was used using the equation:



**Equation 3.1**

Substituting the numbers to the equations would result to an answer of 0.75, which means the ranges will have a 0.75 interval in between. Hence, we could say that (1.0 – 1.75) falls to the Strongly Disagree, (1.76 – 2.50) falls to the Disagree, (2.51 – 3.25) falls to the Agree, and (3.26 – 4.0) falls to the Strongly Agree.

Bar graphs will be utilized to properly exhibit the results which will then be used for the said improvements.