

**Kusinasyon: An Android-based Filipino Cuisine Application  
with Voice Assisted Technology**

**A Capstone Project  
Presented to the Faculty of the  
Information and Communications Technology Program  
STI College Global City**

**In Partial Fulfilment  
of the Requirements for the Degree  
Bachelor of Science in Information Technology**

**John Fritz F. Delafer  
Neonie Quell A. Ponce  
Dean Clyte B. Rebustillo  
Rhey Franz A. Reyes**

**December 2022**

## **ENDORSEMENT FORM FOR ORAL DEFENSE**

**TITLE OF PROJECT:** **Kusinasyon: An Android-based Filipino Cuisine Application with Voice Assisted Technology**

**NAME OF DEVELOPERS:** John Fritz F. Delafer  
Neonie Quell A. Ponce  
Dean Clyte B. Rebustillo  
Rhey Franz A. Reyes

In Partial Fulfilment of the Requirements  
for the degree Bachelor of Science in Information Technology  
has been examined and is recommended for Oral Defense.

### **ENDORSED BY:**

Joselito G. Oyao  
**Capstone Project Adviser**

### **APPROVED FOR ORAL DEFENSE:**

Joselito G. Oyao  
**Capstone Project Coordinator**

### **NOTED BY:**

Mark Frederick V. Salonga  
**Program Head**

**December 2022**

## **APPROVAL SHEET**

This capstone project titled: **Kusinasyon: An Android-based Filipino Cuisine Application with Voice Assisted Technology** prepared and submitted by **John Fritz F. Delafer, Neonie Quell A. Ponce, Dean Clyte B. Rebustillo, and Rhey Franz A. Reyes**, in partial fulfillment of the requirements for the degree of Bachelor of Science in Information Technology, has been examined and is recommended for acceptance and approval.

**Joselito G. Oyao**  
Capstone Project Adviser

Accepted and approved by the Capstone Project Review Panel  
in partial fulfillment of the requirements for the degree of  
Bachelor of Science in Information Technology

Dexter B. Oseña  
**Panel Member**

Jesebelle Salindo  
**Panel Member**

Mark Frederick V. Salonga  
**Lead Panelist**

### **Noted:**

Joselito G. Oyao  
**Capstone Project Coordinator**

Mark Frederick V. Salonga  
**Program Head**

**December 2022**

## TABLE OF CONTENTS

	Page
<b>Title Page</b>	<b>ii</b>
<b>Endorsement Form for Oral Defense</b>	<b>ii</b>
<b>Approval Sheet</b>	<b>iii</b>
<b>Table of Contents</b>	<b>iv</b>
<b>List of Tables</b>	<b>vi</b>
<b>List of Figures</b>	<b>vii</b>
<b>List of Appendices</b>	<b>ix</b>
<b>Acknowledgement</b>	<b>x</b>
<b>Abstract</b>	<b>xi</b>
<b>Preface</b>	<b>xii</b>
<b>Introduction</b>	
<b>Project Context</b>	<b>1</b>
<b>Purpose and Description of the Project</b>	<b>3</b>
<b>Objectives of the Study</b>	<b>4</b>
<b>Scope and Limitations of the Study</b>	<b>5</b>
<b>Review of Related Literature/Systems</b>	
<b>Review of Related Literature</b>	<b>8</b>
<b>Synthesis</b>	
<b>Related Studies and/or Systems</b>	<b>12</b>
<b>Synthesis</b>	
<b>Technical Background</b>	
<b>Overview of Current Technologies to be Used in the System</b>	<b>17</b>
<b>Calendar of Activities</b>	<b>18</b>
<b>Resources</b>	<b>22</b>
<b>Methodology, Results, and Discussion</b>	
<b>Requirements Analysis</b>	<b>23</b>
<b>Requirements Documentation</b>	<b>24</b>
<b>Design of Software, System, Product, and/or Processes</b>	<b>27</b>

<b>Development and Testing</b>	<b>51</b>
<b>Description of Prototype</b>	<b>57</b>
<b>Conclusions and Recommendations</b>	
<b>Conclusion</b>	<b>58</b>
<b>Recommendation</b>	<b>59</b>
<b>Bibliography</b>	<b>59</b>

## LIST OF TABLES

<b>Table</b>		<b>Page</b>
1	<b>Gantt Chart of Activities</b>	21
2	<b>Objective 1 Weighted Mean</b>	51
3	<b>Objective 2 Weighted Mean</b>	52
4	<b>Objective 3 Weighted Mean</b>	53
5	<b>Functionality and Usability (User-Friendliness) Weighted Mean</b>	54
6	<b>Functionality and Usability (Performance) Weighted Mean</b>	55
7	<b>Functionality and Usability (Design and Interface) Weighted Mean</b>	56
8	<b>Survey Form Questionnaire</b>	93
9	<b>User-Friendliness Tally</b>	147
10	<b>Performance Tally</b>	147
11	<b>Design and Interface Tally</b>	147
12	<b>Objective 1 Tally</b>	147
13	<b>Objective 2 Tally</b>	148
14	<b>Objective 3 Tally</b>	148
15	<b>Respondents Age Tally</b>	148
16	<b>Respondents Gender Tally</b>	148
17	<b>Sample Data of Survey Questionnaire</b>	149

## LIST OF FIGURES

<b>Figure</b>		<b>Page</b>
<b>1</b>	<b>Review of Related System (Dapogan)</b>	<b>13</b>
<b>2</b>	<b>Review of Related System (Souschef)</b>	<b>15</b>
<b>3</b>	<b>Agile Methodology</b>	<b>24</b>
<b>4</b>	<b>Process of the Application</b>	<b>26</b>
<b>5</b>	<b>Sign-in Activity</b>	<b>27</b>
<b>6</b>	<b>Create an Account Activity</b>	<b>28</b>
<b>7</b>	<b>Home Fragment</b>	<b>29</b>
<b>8</b>	<b>Ingredients Dialog</b>	<b>30</b>
<b>9</b>	<b>Ingredients Bag Activity</b>	<b>31</b>
<b>10</b>	<b>Recommended Cuisine Activity</b>	<b>32</b>
<b>11</b>	<b>View Recipe Activity</b>	<b>33</b>
<b>12</b>	<b>Public Recipes Fragment</b>	<b>34</b>
<b>13</b>	<b>Private Recipes Fragment</b>	<b>35</b>
<b>14</b>	<b>Account Fragment</b>	<b>36</b>
<b>15</b>	<b>Create a Circle Activity</b>	<b>37</b>
<b>16</b>	<b>Join a Circle Activity</b>	<b>38</b>
<b>17</b>	<b>Join a Circle Confirm Activity</b>	<b>39</b>
<b>18</b>	<b>My Recipes Activity</b>	<b>40</b>
<b>19</b>	<b>View Circle Info Activity</b>	<b>41</b>
<b>20</b>	<b>Invite a User Activity</b>	<b>42</b>
<b>21</b>	<b>Share Public Recipe Activity</b>	<b>43</b>
<b>22</b>	<b>Share Private Recipe Activity</b>	<b>44</b>
<b>23</b>	<b>Circle Management Dialog</b>	<b>45</b>
<b>24</b>	<b>Share Recipe Dialog</b>	<b>46</b>
<b>25</b>	<b>Public Recipes for Approval Activity</b>	<b>47</b>
<b>26</b>	<b>View Recipe for Approval Activity</b>	<b>48</b>
<b>27</b>	<b>Manage Recipe (Vote) Dialog</b>	<b>49</b>
<b>28</b>	<b>Manage Recipe (Author) Dialog</b>	<b>50</b>
<b>29</b>	<b>Sample Recommended Cuisine Recipe</b>	<b>114</b>

<b>30</b>	<b>Sample Private Circle</b>	<b>115</b>
<b>31</b>	<b>Sample Private Recipe</b>	<b>116</b>
<b>32</b>	<b>Sample Public Recipe</b>	<b>117</b>
<b>33</b>	<b>User's Publicly Shared Recipe List</b>	<b>118</b>
<b>34</b>	<b>User's Privately Shared Recipe List</b>	<b>119</b>
<b>35</b>	<b>View Circle Members</b>	<b>120</b>
<b>36</b>	<b>Sign-in Guide</b>	<b>122</b>
<b>37</b>	<b>Create Account Guide</b>	<b>123</b>
<b>38</b>	<b>Home Guide</b>	<b>124</b>
<b>39</b>	<b>Select Ingredient Guide</b>	<b>125</b>
<b>40</b>	<b>Ingredients Bag Guide</b>	<b>126</b>
<b>41</b>	<b>Recommended Cuisines Guide</b>	<b>127</b>
<b>42</b>	<b>Public Recipes Guide</b>	<b>128</b>
<b>43</b>	<b>Private Recipes Guide</b>	<b>129</b>
<b>44</b>	<b>User Account Guide</b>	<b>130</b>
<b>45</b>	<b>Create Private Circle Guide</b>	<b>131</b>
<b>46</b>	<b>Join Private Circle Guide</b>	<b>132</b>
<b>47</b>	<b>Confirm Join Private Circle Guide</b>	<b>133</b>
<b>48</b>	<b>My Recipes Guide</b>	<b>134</b>
<b>49</b>	<b>Circle Management Guide</b>	<b>135</b>
<b>50</b>	<b>Share Recipe Selection Guide</b>	<b>136</b>
<b>51</b>	<b>View Circle Info Guide</b>	<b>137</b>
<b>52</b>	<b>Invite a User Guide</b>	<b>138</b>
<b>53</b>	<b>Share Private Recipe Guide</b>	<b>139</b>
<b>54</b>	<b>Share Public Recipe Guide</b>	<b>140</b>
<b>55</b>	<b>Public Recipes for Approval Guide</b>	<b>141</b>
<b>56</b>	<b>View Public Recipe for Approval Guide</b>	<b>142</b>
<b>57</b>	<b>Manage Recipe for Approval Guide</b>	<b>143</b>
<b>58</b>	<b>View Recipe Guide</b>	<b>144</b>
<b>59</b>	<b>Manage Recipe (Author) Guide</b>	<b>145</b>
<b>60</b>	<b>SPSS Survey Data</b>	<b>151</b>

## LIST OF APPENDICES

<b>Appendix</b>		<b>Page</b>
<b>A</b>	<b>References</b>	<b>66</b>
<b>B</b>	<b>Resource Person/s</b>	<b>69</b>
<b>C</b>	<b>Relevant Source Code</b>	<b>75</b>
<b>D</b>	<b>Evaluation Tool or Test Documents</b>	<b>110</b>
<b>E</b>	<b>Sample Input/Output/Reports</b>	<b>113</b>
<b>F</b>	<b>User's Guide</b>	<b>121</b>
<b>G</b>	<b>Survey Results and Tally</b>	<b>146</b>
<b>K</b>	<b>Project Involvements</b>	<b>152</b>
<b>L</b>	<b>Personal Technical Vitae (one page per member)</b>	<b>155</b>

## **ACKNOWLEDGEMENT**

The developers would like to acknowledge and give thanks to the group's adviser and capstone coordinator, Mr. Joselito G. Oyao, for guiding the developers throughout the development of the application. In addition, the developers would like to thank Mr. Mark Frederick V. Salonga, Mr. Dexter B. Oseña, and Ms. Jesebelle Salindo for giving suggestions and improvements to make the application better. Moreover, the developers would like to thank Raquel Fuasan-Bautista for reviewing the project. Furthermore, the developers would like to thank their classmates and Mr. Frederick B. Ponce for supporting the project and making it possible.

## ABSTRACT

Title of project: **Kusinasyon: An Android-based Filipino Cuisine Application with Voice Assisted Technology**

Developers:  
**John Fritz F. Delafer  
Neonie Quell A. Ponce  
Dean Clyte B. Rebustillo  
Rhey Franz A. Reyes**

Degree: **Bachelor of Science in Information Technology**  
Date of Completion: **2023**

Key words: **Filipino, Voice Assisted Technology, Cooking, Recipe, Cuisine. Android, Food, Recipe Sharing, Ingredients**

---

With the ever-growing food industry, people are being dependent on restaurant foods since it is more convenient rather than preparing and cooking on their own. As a result, these people are being ignorant from the combination of ingredients and food preparation. Making it harder for some to think of and cook their own food. Food in the Philippines are also diverse since each archipelago or places have their own unique style of preparing and cooking a food. Additionally, since most people are ignorant from food preparation and cooking, when they cook, they heavily rely on cookbooks since these contains the recipe instructions that they can follow for the cooking procedure of a food. However, for some it is inconvenient to touch cookbooks or recipe e-books while cooking. Due to the gathered problems, the developers designed and developed a project that aims to help people combine their available ingredients and recommend Filipino cuisines based on those ingredients. The project also enables users to share Filipino recipes with each other to help them discover the diverse culture of Filipino foods. Moreover, the project is embedded with a voice assisted technology which can help users to conveniently follow recipe instructions while cooking. The developers conducted surveys to help determine whether the project resolved the gathered problems. Based on the results, it is safe to say that the developers successfully fulfilled the project's purpose and completed its objectives.

## PREFACE

The idea behind this project came to life when the developers thought of the current generation and their primary knowledge about cooking. The developers discovered that most people nowadays do not have an idea about combination of ingredients which is basically the basis for cooking. That is why the developers provided an Android application that has the feature to recommend cuisine, specifically Filipino foods, based on the entered ingredients by the user.

In addition, the developers created an environment for the users to share their knowledge on Filipino recipes since the Philippine is a country with a wide culture for food. This was made possible by allowing them to post and share recipes that they have knowledge on.

Moreover, the developers thought of an idea to help people who heavily rely on cook/recipe books make cooking more convenient by letting users to state voice commands instead of manually navigating and reading the recipe instructions.

The project's idea and features were also solidified with the contribution of the project adviser and panelists which led to the project's great progress that led to the application's quality being assured as successful. The developer's document writing process was also guided by the project adviser which resulted to the document being organized and structured.

At the end, the developers defended the capstone project successfully, with the final verdict of conditional passed with minor revision. These revisions were addressed and accomplished within the given period of time that helps with the betterment of the capstone project.

## INTRODUCTION

### Project Context

Food is one of the factors that plays an important role in every person's life on this planet. It provides us the basic energy that we need to support our body to function properly. Human beings can survive without other necessities like clothes, but one cannot survive without food. Food simply helps us to survive because it protects our bodies from the different deadly diseases by providing us the nutritional values that we need to strengthen our body's systems.

In our modern world there are lots of people who loves to eat. It is because eating is a satisfying activity since it allows them to enjoy the taste they particularly want to have and at the same time fulfil their hunger. In some people, they love to consume foods because eating simply helps them uplift their mood and become joyful.

Nevertheless, even if a lot of people loves to eat, it is still a challenge for a few when it is now their turn to cook. This is because it turns out that these few people do not actually have knowledge about food preparation and cooking. According to Magner (2019), millennials consume restaurant foods more often and are not particularly confident when it comes to their kitchen abilities than other generations. In addition, according to Melore (2021), a breakdown of the poll conducted by Cinch Home Services turns out that millennials or young adults are the least likely to answer that they often or always cook. As a result, these millennials or young adults are the most likely to order restaurant foods and take outs. Furthermore, according to Brown (2021), the research that was conducted by a hot dog brand Ye Olde Oak concluded that almost half of millennials do not know how to cook meals from scratch.

People not knowing how to cook is a major problem. This is because it makes them rely more on prepackaged or restaurant foods which are far more expensive than

home cooking. Prepackaged or restaurant foods are also unhealthy since these are processed foods that come with unhealthy levels of sugar, sodium, and fats. In addition to that, not having enough knowledge about cooking simply makes a person ignorant from the combination of ingredients that are used for cooking a certain food. Hence, it will be difficult for that person to think of and come up with a food that s/he can cook whenever given with ingredients only.

Moreover, the Philippines is a country composed of various islands with different primary resources. Therefore, Filipino cuisines have different variations depending on peoples' primary ingredients, cooking methods, and preferred tastes. According to Mendiola (2022), the founder of Philippine Culinary Heritage Movement stated that foods in Northern Luzon have simpler cooking methods than from the place he came from. In Visayas, people use fresh or dried seafoods more often, and in Mindanao the foods are bold, versatile, and culture driven. Aside from primary resources, Filipino cuisines may also have a different variation of food due to the Philippines being influenced by other countries such as Malaysia and Spain. According to Ivan (2020), Filipino foods are unique as it draws inspiration from several influences. Filipino foods were essentially developed during the many years of colonization influenced by several countries and Pacific islanders.

Another tricky part when it comes to cooking is that there might be a case where people depend on food recipes. Therefore, they need to constantly learn what are the next instructions that they need to follow to achieve the expected outcome of the food being prepared and cooked. However, there are situations where people are busy preparing the ingredients and they do not have the time to continuously look on the instructions, especially in the case of mobile recipe applications where a person needs to hold a smartphone and use his/her finger to swipe or click the screen to look for the next instruction. It might also be a problem during cooking to use a smartphone because cooking involves hands-on activities which can make the hands of a person dirty. Thus, it could feel uncomfortable for users especially those who are concerned with cleanliness and proper hygiene to hold or touch his/her smartphone. According to Chang and Stolyar (2019), a team from Scripps

Lifestyle Studios observed how people would interact with their smartphones while cooking. They figured out that people would be often left no choice but to dirty their screens when using their smartphones to see more steps and information about the food they were cooking.

After analyzing the identified problems, the developers formed an idea of a project that seeks to help people to cook food (specifically Filipino cuisine), based on available ingredients and at the same time share different Filipino recipes with the help of a voice assisted technology. This project is called Kusinasyon.

### **Purpose and Description of the Project**

The project name, Kusinasyon, was derived from the Filipino word “kusina” which means kitchen, and “nasyon” which means nation. The project intends to design and develop an Android application that helps people to come up and cook Filipino cuisine based on available ingredients, and at the same time share different Filipino recipes with the help of a voice assisted technology.

Much like other Android application about cooking and recipes, Kusinasyon provides a list of Filipino cuisines where the user can select what cuisine s/he wants to cook. However, the key difference is that Kusinasyon can recommend a cuisine based on user’s available ingredients. With this approach, it can help the people to effortlessly think what food they can cook. Another distinctive part of the project is that the foods in the application will display its origin place to easily identify what part of the Philippine archipelago it originated from, whether it is from Luzon, Visayas, or Mindanao. Renowned Filipino foods that do not specify their origin will be listed under “More” categories.

Moreover, the project enables its users to interact with each other by allowing them to share their own Filipino recipes. Furthermore, the project is embedded with a voice assisted technology which can be utilized by the users to conveniently use the application.

## **Objectives of the Study**

The aim of the study is to develop an Android application that can help the users to come up and cook Filipino cuisines based on available ingredients, and at the same time share different Filipino recipes with the help of a voice assisted technology.

The study would like to accomplish the following specific objectives:

- To design and develop an Android application that can recommend Filipino cuisines based on available ingredients;

The function allows the users of the application to enter their available ingredients at their kitchen. Then from the given data, the application searches on the database for a Filipino cuisine that can be cooked. With this function, the users can save time from the trouble of thinking and coming up what food to be cooked.

- To design and develop an Android application that enable the users to share Filipino recipes with each other; and

The function is a way for the users of the application to share their Filipino recipe with other users. With this function, the users can discover and cook different variation of foods made by other people. It will also help the users not to be limited on the pre-defined recipes that is already embedded in the application.

- To design and develop an Android application with voice assisted technology.

The function helps the users to cook more conveniently while using their smartphones with voice assisted technology. This is accomplished by allowing them to state commands instead of manually clicking or swiping the screen.

## **Scope and Limitations of the Study**

### **Scope**

The developers' target users are people who have trouble coming up what food to cook, and those who are willing to share their knowledge on Filipino recipes.

In addition, all Filipino cuisines are included since there are some that are not available to be found on the internet. Similarly, a certain cuisine that does not specify its origin, but is renowned as a Filipino food is listed under “More” categories.

Likewise, to make sure that the shared public recipes of the users are Filipino cuisines, the user needs to join a circle first, then seventy-five percent (75%) of the circle members will be the one to vote, whether to approve or to reject, the recipe to be publicly shared. Recipes that are shared, marked as public, and approved will be added to the list of foods that will be used on the recommendation feature.

### **Limitations**

For the limitations of the project, first is that the recipe sharing feature is not included with a comment or chat section for recipes. It means the only interaction between the users of the application is by sharing recipes with each other.

In addition, the commands that can be stated using the voice assisted technology are limited based on the instructions of the recipe. It cannot perform advanced operations like Alexa or Siri. The voice assisted technology also recognizes English language only. Furthermore, the responsiveness of the voice assisted technology depends on the stability and speed of the internet connection of the user.

## **Features of the Developed System**

- **Cuisine Recommendation based on Available Ingredients:** The application provides a list of different ingredients (e.g., vegetables, dairy and eggs, herbs, and spices) in which the user can select from. Afterwards,

the selected ingredients of the user will be collected by the system and match it from the database of the application to look for a cuisine that matches the input ingredients. Whenever the system finds a match, it will automatically recommend the cuisine to the user. If the user clicked the recommended cuisine, it would immediately show the details of it.

- **Ingredients Bag:** The application has an ingredients bag that serves as the collection for all the selected ingredients of the user (just like a cart). With ingredients bag, the user can view the added ingredients and remove it whenever needed.
- **List of Different Filipino Cuisines:** The application has a module that categorizes the three main island of the Philippines (Luzon, Visayas, Mindanao). Also, a “More” module will be added which contains the renowned Filipino foods that does not have a specific origin. All the modules will contain the list of foods that will be used on the recommendation feature. Additionally, it functions as a free way for the users to freely look for cuisines without the need to use the cuisine recommendation feature.
- **Recipe Sharing:** The application has a recipe sharing feature that can be used by the users to share their own variation of Filipino recipes with other users.
- **Private Circle:** The recipe sharing feature has a private circle (group of users) where the users of the application can share their recipe privately with each other. Nevertheless, if the users of the application want to share their recipe publicly, they can still do so by marking it as public.
- **Voice Assisted Technology:** The application is embedded with a voice assisted technology in which it can be utilized by the users to conveniently determine the recipe instructions while cooking.

Below are the keywords or commands that can be used on the voice assisted technology feature:

1. **Friday** - the name of the voice assistant that will be embedded on the application. This will be the first keyword that is needed to be stated before saying other commands.
2. **Start cooking** – triggers the voice technology to start assisting the user. This will also cause the voice technology to read the first cooking instruction.
3. **What's next** – makes the voice technology to go on the next given cooking instruction and read it.
4. **Go back** – makes the voice technology to go on the previous cooking instruction and read it.
5. **Repeat current instruction** – makes the voice technology to read again the current cooking instruction.
6. **Go to step (n)** – makes the voice technology to go on a specific number of cooking instruction and read it.
7. **Volume up** – causes the voice technology to raise the media volume by one-step on the user's smartphone.
8. **Volume half** – causes the voice technology to set the media volume to half on the user's smartphone.
9. **Volume max** – causes the voice technology to set the media volume to max on the user's smartphone.

## **REVIEW OF RELATED LITERATURE/SYSTEMS**

### **Review of Related Literature**

As stated by Garcia (2021), on his literature titled "Acceptability, Usability, and Quality of a Personalized Daily Meal Plan Recommender System: The Case of Virtual Dietitian" that Filipinos have inadequate nutrition problems and challenges of personalized nutrition, therefore they developed the Virtual dietitian which is a nutrition intervention tool to solve the nutritional problems and challenges. It is a food recommender which has input requirements such as knowledge, recommendation, and nutrition care. These requirements serve as a core knowledge and personalization variables as inputted by the users. This feature of the literature is related to the first objective of our study which has to do with a system that can recommend a Filipino cuisine.

According to Villanueva (2020), titled “Food-related online community flourishes during lockdown” stated that there is a saying that food connects people which is the reason that there has been an exponential growth of numbers of members on Facebook groups. One of the said Facebook group admins of Let’s Cook Pare? named Salgado had stated that the Facebook group was created before the enhanced community quarantine (ECQ) in middle of March had 10,000 members. Due to the lockdown in end of April for Luzon and for the National Capital Region (NCR) is the end of May give rise to the numbers to the group members which grows into a social group of 60,000 members and till now still counting which is now 90,000 members. This groups have increased due to some might encountered at the time given has acknowledge that they have interest in cooking. This has also created an impact on socializing of one’s society that having an interest and likings may cause a group of community to share information on what are the things they also have interest with. It was also stated that these factors are seen to “sustain our community’s active online presence, aside from Filipinos’ innate interest in food,”

According to Salgado (2020), in line with the objectives of this project that this group sharing may also help users on the platform to have a circle of community that they may share their knowledge regarding Filipino cuisines and also have a community that everyone may benefit on their group sharing of information.

In the article of Nacario (2021), titled "Insight: Making voice technology work for everyone" it is stated that in 2020-year era, voice implementation usage skyrocketed within smartphone users alone. It was also stated that implementing voice driven algorithm in Google products like smartphone operating system (Android) shows a significant effect such as convenient substitute for typing, as the author of this research also stating that voice assisted implementation of the application may benefit the user by conveniently conveys common commands such as dictating special keywords that was programmed from the voice API within the system. This statement relates to one of the developers' objectives which is to design and develop an Android application with voice assisted technology.

According to the gathered local literatures, Garcia (2021) stated on his study titled "Acceptability, Usability, and Quality of a Personalized Daily Meal Plan Recommender System: The Case of Virtual Dietitian", it was developed to help users to generate a list of possible meals which is being recommended based on the users inserted or inputted variables. Garcia also developed the recommender system is to solve a problem which is to help create a meal plan to make users have the proper dietary intervention which is to solve the nutritional problems and challenges. This is also like the aspect of the first objective of the developers of Kusinasyon which is to help users decide what to cook or problems on what are the possible cuisines they can create based on the available ingredients in their kitchen. This is when the Kusinasyon will generate possible cuisines and recommend it to the user based on what he/she has inputted. The recommendation of cuisine will help on having lesser time consumption on thinking what are the possible cuisines that can be cook and help users to choose the cuisines that they want to cook based on their preferences. Because of the diversity of the Philippines there are uniqueness of dishes which there are some parts of the Philippines where there are

similar dishes but have different ingredients and methods. This is where it is related and beneficial for the users and developers to learn and explore on what are the possible cuisines that can be cooked and generated on Kusinasyon. Recipe sharing platform was also done by Salgado (2020) which is one of the Facebook group admins of Let's Cook Pare? Which created a Facebook group that help users to share their personal dishes and knowledge about cuisines which is also a created an impact on numerous people that are now a community that shares interest and sharing of information. Related to Kusinasyon's recipe sharing feature will benefit the users by expanding their knowledge about the cuisines of the Philippines by being in a circle or group of users that will be used as an environment where the users can provide cuisines that they know which other users can also cook on their kitchens. Voice assisted technology is also a feature of Kusinasyon which will help the users to have a more convenient way of having instructions or directions which will help them cook without touching the screen of their phone and is supported also as stated of Nacario (2021), titled "Insight: Making voice technology work for everyone" that implementing voice driven algorithm can benefit the users by conveniently saying common commands such as dictating special keywords based on what are the programmed keywords that is within the API of the system.

According to Trang Tran (2018), "An overview of recommender systems in the healthy food domain" nowadays that the internet has become a wide aspect of technology this means that more information is being loaded into it. This causes users to have a difficult time finding the specific or proper information that he/she is looking for. Therefore, recommending systems are being implemented as a tool to make searching for a specific category or information much faster. Recommending system or tools has gained multiple attention that it is now used in multiple platforms and usually used nowadays on food related recommendation. The literature implies that recommendation systems are making decision making and even meal planning uncomplicated.

In the article of Taylor (2021), titled "Sharing food and recipes has the power to bring people together", Bejok states that people love to cook and eat, sharing food

culture helps the people learn about each other distinctive recipes and cuisines. As Bejok settles into her life in Australia the opportunity to share recipe from distinctive cultures has been vital. Australia there are numerous diverse cultures and be able to taste distinctive food. Another statement of Sally Win states that Win's elder sister passed away in Yangon and whereas cooking her sister's recipes, the feeling remains exceptionally raw. Her eyes fill with tears when she talks approximately her sister and Win clarifies it's impossible, she will be able to return to Myanmar where her mother, who is presently 86, still lives. The sharing of the recipes has been an imperative bridge connecting her present and old homeland. The statements relate to one of our specific objectives which allows user to share their Filipino cuisines and recipe.

According to Liu et al. (2020), "Design and Usability Evaluation of Mobile Voice-Added Food Reporting for Elderly People: Randomized Controlled Trial" which in this literature it is mentioned that voice technology is now being used on multiple platforms and usually used on a daily health maintenance. One of the obstacles that are the possible user that can't utilize technologies like this for example are the older users that may encounter this kind of technology. This guided the developers to conclude and create a Voice-only reporting (VOR) and voice-button reporting (VBR) which are two different mobile voice-added apps for food intake reporting that the developers built and analyzed (VBR). Every application has its own interactive method for recording food intake. Users report the major ingredients of each dish vocally using VOR, but VBR allows users to report meal intake using both speech and existing touch screen inputs. The accuracy, efficiency, and user perception of the two apps were used to evaluate their mutual usability. the researchers contribute to a user-centered design model's evidence-based evaluation of prototype design and selection. The findings can be used as a guide when choosing the best user interaction design. Having ease of access will help the users to have great satisfactory experience in using the application. Due to that reason, this literature has enough relevance to guide the developers of this project to create an application that utilizes the importance and the benefits of having a voice assisted technology.

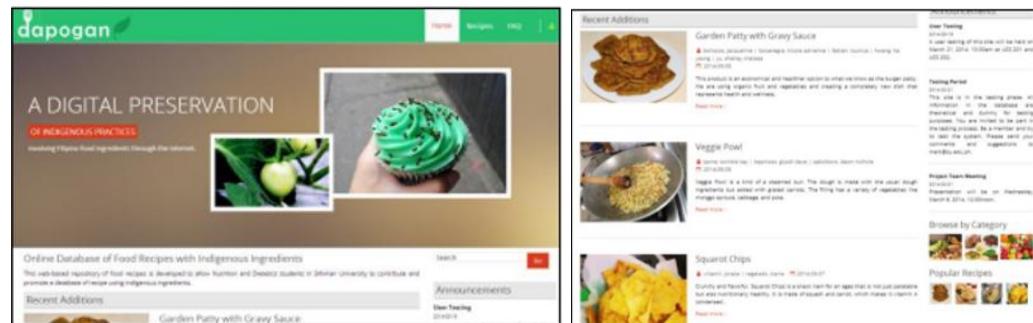
According to the gathered foreign literatures, it provides helpful information to the developers about how the recommending system principles builds the foundation of this project, Kusinasyon. As recommending system gained popularity amongst different system platforms, it was stated in the literature that implementing this kind of system helps improves the searching of information makes it more less difficult and faster process to the users. Correlating this to the project, which is a recommender system that will recommend a list of cuisines based on what ingredients the user inputs to the application. It also gave the developers knowledge about how the Philippine food culture diversely connected to each other without losing the roots of that specific recipe. Moreover, stating that food recipe sharing can occur not just locally but also internationally. It gave the developers idea and knowledge to implement a feature called "Recipe Sharing" within the project Kusinasyon. It allows the user to share their own desired recipe to other users too. It helps the developers to build the foundation of implementing voice assisted technology to the project. Additionally, it gives the user more time in cooking with just dictating instead of typing and touching the device.

### **Related Studies and/or Systems**

In research of Cruz (2017), the recommendation application "Fatchum": A Mobile Application Recipe Recommender System is to identify and assess on what can help develop users experience while still maintaining their goal as a recipe recommendation application."Fatchum" is develop to provide recipe names which can help users to create dishes based on what information and instructions that has been provided by the mobile applications recommendation system based on the users preference which also gave no boundaries on the users availability of usage on the application based on the users gender, ethnicity, age but instead use it as a tool to help the user increase their user satisfaction or experience on the mobile application. It was determined that there are two aspects on how mobile application will achieve its purpose while still maintaining the usability of the application which was concluded into two factors. The two factors that was determined is the data collecting of the users' demographic factors and the design related factors may

create a substantial impact on the application which is determined that it may still accomplish the main objective due to its factors of influencing the app's functionality or usability.

In research from Marcial (2017), titled “Developing Online Database of Food Recipes with Indigenous Ingredients” which is an online database of food recipes with indigenous ingredients. This related system is called “dapogan”, which enables the food developer to share and promote food made from indigenous ingredients utilizing the use of internet. This related system helps in the preservation and sharing of local knowledge and indigenous practices in regards of Filipino food recipes with indigenous ingredients. Using the internet and a 24/7 online, systematic, and up-to-date database management system of food recipes with indigenous ingredients, this will promote foods made from indigenous ingredients using digital technology. Conforming with the project’s objectives, which also aims to create an Android application that will enable the users to share Filipino cuisines and recipes within circles of communities.



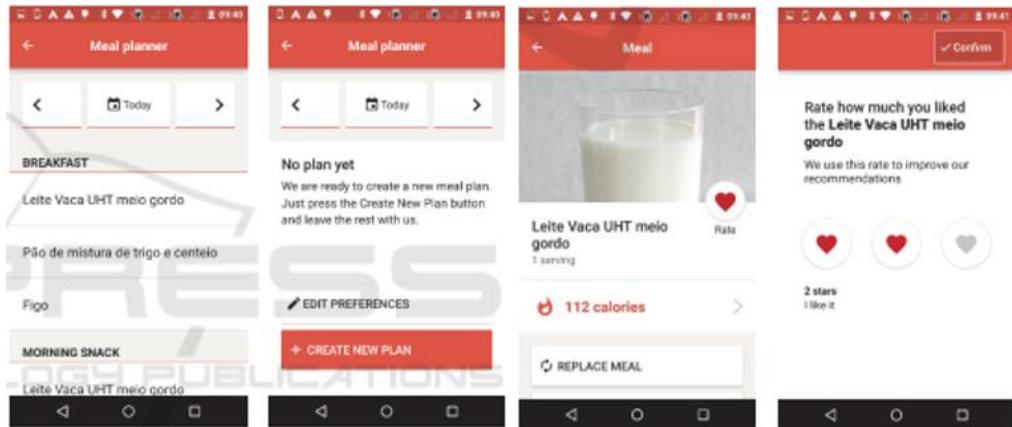
**Figure 1: Dapogan**

In research from Guigue et al. (2019), titled “A voice-based multi-platform first aid application using the Jaccard similarity index algorithm.” stated that the implementation of voice driven commands makes the usage of the application more intuitive and convenient. The voice-based technology implementation can recognize three languages such as English, Filipino and Cebuano to cater a diverse user whose preferring their own desired language. Also, it can recognize common voice commands such as locating nearby hospital or pharmacy according to your

location. The system voice-assisted technology implementation that recognized commands from the user have relevance and can serve as guide for the development of this project.

According to the gathered local systems, the system titled “Fatchum” which is a recommender system, gave idea to the developers that a recommender system would help in generating a recipe or cuisine based on the user’s input and data coming from the database of the application. Instead of spending time thinking what the users can cook, the recommender system will help by generating food recommendation ideas based on users’ ingredients. The online database of food recipes with indigenous ingredients called “dapogan”, is a related system that has an online database of food recipes with indigenous ingredients. This related system will help in the sharing of local knowledge and indigenous practices in regards of Filipino food recipes with indigenous ingredients. The statements relate to one of our specific objectives which allows user to share their cuisines and recipe within the Kusinasyon’s feature called recipe sharing. A voice-based multi-platform first aid application have implemented voice driven commands that makes the usage of the application more intuitive and convenient, this can serve as guidance to the development of the project. Kusinasyon will be ahead of the related systems by adding the voice assisted technology feature, by initializing command words that can be recognize by the system to perform basic functionality of the application.

In research from Ribeiro (2017), titled "SousChef: Mobile Meal Recommender System for Older Adults" which is a food recommender system which generates a meal plan considering the information from the user, including anthropometric measures, personal preference, and activity level, which targeting the older adults age bracket. The system implemented a content-based recommender system and mainly employs information retrieval techniques. The recommender system has three steps to follow to create the appropriate personal meal plan for the user. First is the calculation of nutritional requirements, selection of food items for each meal and scaling the meals to match user's caloric needs.



**Figure 2: Souschef**

In research from Tsai (2018), titled A Personal Emotion-Based Recipe Recommendation Mobile Social Platform: Mood Canteen. This system is a mobile recommendation system that generates its' food recommendation from personal emotion that share recipes through social networking services. The system is a mobile social platform that enable its user to alleviate their negative emotions and advocate positive emotions by the way of sharing recipe and meals, this feature is done by the process of creating personalized meal recommendation that provides users with the appropriate food selection that is based on the personal emotional state of the user and the emotional food database. The system recipe sharing through social media platform have relevance to the objective of the developers developing project.

As stated by Angara et al. (2017), in their system Foodie Fooderson, conversational agent are software agents that utilizes natural language interfaces, such as text or voice, to interact with human, brands, or services. Prominent examples of these agents are Apple's Siri, Microsoft's Cortana, Google Assistant, Amazon's Alexa, and Mark Zuckerberg's Jarvis. These agents are built with the services provided by powerful commercial reasoning engines. These constitute a new trend in digital gateways for accessing information, making decisions, and communicating with technological devices through sensors and actuators. Referring to Weizenbaum (1966), The concept of conversing with a machine has been around since 1966

when Weizenbaum created ELIZA, the first natural language processing computing program. The concept of bots was not to replace human, but rather to be a companion or a tool as extensions of the human mind. According to Hirschberg and Manning (2015), Natural language processing and artificial intelligence have advanced to such a degree that computers are able to almost predict what a user's intentions accurately. These tools have relevance to the application and can be used as guide for the developer of the application to produce a worthy application that can communicate with its user.

According to the gathered foreign systems, the developers conclude that having a system that can recommend cuisines based on provided contents by the users can greatly help them to select the appropriate cuisine that personally suits them. In addition, having a recipe sharing feature on the project like on the Mood Canteen system, could expand the knowledge of the users on the different variation of foods, especially in the Philippines since it is a country that has a broad culture of foods. Furthermore, food related systems with voice assisted technology like Foodie Fooderson can be a great way for humans to be engaged with technology for better assistance of basic tasks. Similarly, the project, Kusinasyon, will utilize the use of voice assisted technology to aid the users on using the application while they are busy performing cooking related activities.

## **TECHNICAL BACKGROUND**

### **Overview of Current Technologies to be Used in the System**

This chapter covers the overview of the current technologies that are used in the development of the application, Kusinasyon.

Since the application is in the form of an Android-based application. The developers used Java as the backend programming language to develop the application. Functions and behaviors of the application are written in Java.

The application uses Firebase Realtime Database for the system database. Simply because Firebase Realtime Database has a quick read and write speed which significantly increases the performance of the application.

For the user authentication, the application uses Firebase Authentication. This is because Firebase Auth has an integration of Google sign-in API which enables the users to log in to the application using their Gmail account. The Firebase Auth also has a feature that let users to register other e-mails like Yahoo which can be used to sign-in on the application once verified.

On creating and designing the user interface of the application. The language that is used is Extensible Markup Language (XML). This is because Android Studio uses XML for user interface components since XML is a lightweight markup language that does not make heavy layouts.

As the application features a voice assisted technology. The technology that is used to fulfill it is the Picovoice AI and Azure Speech Service. The Picovoice AI will give the application the ability to have a wake word. This wake word will be used as a replacement for the button click so that users do not have to click their screens before stating a command. On the other hand, the Azure Speech Service helps the application to read information to the users just like a real person using natural language processing.

## **Calendar of Activities**

The Gantt chart presents the summary of activities. Listed are the activities and opposite them are their duration or periods of execution.

In planning stage, the developers started from selecting of possible field and topic that could match the developers' skills and comfort programming language. After the election of topic, the developers' construct three (3) titles for the chosen focus area. The titles were consisting of three (3) topics, one on each of the platform which are Android, web-based, and desktop. The developers came up with the title during a brainstorming session. These sessions occurred between the last week of February and the first week of March. The developers then met with the research project coordinator and adviser for a formal presentation of the three selected projects, which took place in the second week of March. The project coordinator and adviser gave some comments and recommendations in creating the chosen title after the three (3) titles were presented and the Android-based application was selected.

After the planning stage, the developers go on to the analysis stage of the application's development. The developers gathered information on the application topic and identified the issues that began in the third week of March. As the developers changed every recommendation given during the meetings, the project's context was constantly improved. The developers also start defining the application, which includes the project's objective and description. The developers then began listing the project's objectives, which is the building of the application's aim based on the identified concerns. It was completed during the 3rd week of April to 1st week of May, which matched with the application's definition. The study's objectives were written by the developers based on the specified problems. The defining of the application, which took place from the third week of April to the first week of May, corresponded with the establishing of the application's scope and limitations to determine the scope of the system being constructed. The first two months of the first semester were spent drafting the first chapter of the study.

By the second week of April, the capstone project coordinator had finished teaching the RRLs and RRSs. This is when the developers started on finding and searching for possible RRLs and RRSs which will be used to support the project of the developers are proposing. Starting from the 1st week of May the RRLs and RRSs are accomplished which also included some of the revisions that has been made. The RRLs and RRSs were split apart to speed up the process. Then, between the first and second weeks of May, the developers began determining what improvements could be made from the gathered related systems. The developers discussed and brainstormed a list of features that may be added to this system as well. Listing the hardware and software resources to be used in way to construct the “Technology Evaluation” in the second week of May.

At the 4th week of May the developers did the final evaluation of chapters 1 to 3 to proceed on the schedule of defense proposal and final submission of the document chapters 1 to 3. After the result of the proposal defense the document have been revised and minimal changes on the application has been implemented. The developers resumed the design of the systems user interface and prototyping between the 2nd week and 3rd week of July.

On the 2nd week of September, the evaluating of technologies that is being used to develop was resumed until the 2nd week of October. The first orientation of Capstone 2 was on the 23rd of September, it is the 3rd week of the schedule of activities based on the capstone guidelines in which our agenda is about how many weeks can the developers make a consultation for their advisers regarding their capstone projects. Topic regarding what week will be the last day of consultation, endorsement for final defense, release final defense schedule, week of final defense, week of re-defense, date of submission for regarding the revisions and the release of grades was tackled. The 2nd week of September was Designing of the user interface was continued starting on the 2nd week of September along with the prototyping continuously until the 3rd week of November.

At the week 1st to the 4th of the Schedule of activities the developers met their advisor for consultation where in there was a testing of the application and checking for bugs has been conducted. The developers showed their application to the advisor whereas the 1st objective and 3rd objective which is partially finish has been demoed to gather feedbacks. The developers' advisor suggested to widen the scope of the voice assistant technology, add more commands for the voice assistant and to finish and present the 2nd objective of the developed application.

On the 5th week of the consultation with the adviser the developers demoed the accomplishment activities of the developed system where in additional keywords/commands has been implemented, navigation of the application was changed and improved the application performance, sign-in and sign-out of the application using Gmail account was implemented. Repeating of certain steps for the voice command was added and the 2nd objective was finished where in the user can create a circle/group to share their Filipino recipes/cuisine. In addition, the advisor suggested to widen the scope of the voice technology but with limitation and publishing depending on the minimum requirements of Google Play Store policy which the developers have accomplished on the same week and was demoed to the advisor before the recommendation or endorsement for final defense has ended. The developers conducted a survey to various people within the 2nd week of November with the use of Likert Scale.

Furthermore, the developers were scheduled to defend their capstone project during the 4th week of November. As a result, the project was approved with minor revisions such as sign-up page validation and email verification, an option for users to tag their recipe as "Only Me," and 75% total member approval of the recipe via circle before releasing it to public recipes. By the 1<sup>st</sup> week of December, the project was polished and implemented the four minor revisions mentioned before. Lastly, the remaining 3<sup>rd</sup> and 4<sup>th</sup> week of December was utilized to maintain the system integrity of the project.

## Gantt Chart of Activities

**Table 1: Gantt Chart of Activities**

MONTH	FEBRUARY	MARCH	APRIL	MAY	JUNE	JULY	AUGUST	SEPTEMBER	OCTOBER	NOVEMBER	DECEMBER
ACTIVITY											
Selection of possible topic			■	■							
Constructing titles		■	■								
Presentation of titles			■								
Analysis of the problem contexts			■	■	■	■					
Defining the application				■	■	■					
Constructing the aim of the application based on the determined problems				■	■	■					
Establishing the scope and limitation of the application				■	■	■					
Gathering possible RRLs and RRSs				■	■	■	■				
Identifying improvements that can be made from the gathered related systems					■	■					
Evaluating the technologies that will be used on the development		■	■			■	■				
Designing possible user interface				■	■	■		■	■	■	
Prototyping					■	■		■	■	■	
Test cases and survey									■	■	
Computation of results										■	
Project defense										■	
Polishing of the app											■
Maintaining of the project											■

## **Resources**

- **Hardware**

The hardware resources that the developers used to complete the project is a computer with an Intel Core i5 8<sup>th</sup> Generation CPU and at least 16GB RAM. Additionally, to test the app during development, an Android smartphone with minimum specifications of API Level 28 (Android 9: Pie), 4GB RAM and octa-core CPU is needed. This is also the specification for the users' smartphone when they will use the application. Furthermore, the Android smartphone needs to have a working microphone since the project will utilize the use of voice assisted technology.

- **Software**

The software resources that the developers needed to complete the project was Android Studio which is the official Integrated Development Environment (IDE) for developing Android-based applications. Also, an Android virtual device or emulator which are used during the development of the application for testing.

To help the developers develop the application without any problem on the application files. The developers used software like Git and GitHub. This enables the developers to control the versions of the application during development.

Moreover, to help the developers properly design the user interface of the application. Adobe Photoshop CS6 was used since this software contains helpful tools for designing.

Furthermore, a web browser like Google Chrome is important to help the developers gather information to completely design and develop the project, Kusinasyon.

## **METHODOLOGY, RESULTS, AND DISCUSSION**

### **Requirements Analysis**

Plan. Kusinasyon's plan is to create an Android application that will assist the user to come up with a recipe by recommending Filipino cuisines based on available ingredients on their kitchen. The application will also be a tool for users to explore different variation of Filipino cuisines with the help of recipe sharing. Kusinasyon, as a companion in the kitchen, can help the users to navigate the recipe's instruction hand's free with the help of a voice assisted technology.

Design. The design of the Android application composes of a user-friendly interface (UI). It uses a combination of Material Design two (2) and three (3) that exhibit a well ordered, and straightforward UI. Furthermore, the application showcases logo and icons that are appropriate with the contents. The information provided inside the application are concise and informative, uncomplicated that is easily understood by most of the users.

Development and Testing. The development of the Android application is done in alternating manner with the testing. Development and testing are implemented in cycle to ensure the application is working properly and efficiently, to minimize the time consumed in fixing bugs and errors.

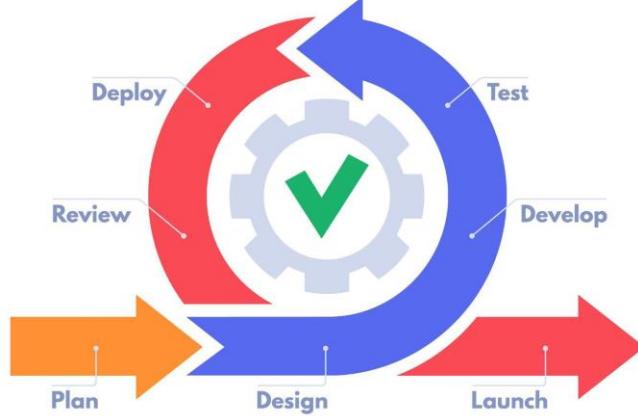
Deployment. The developers are polishing the Android application for the publishing. Nevertheless, revisions and feedbacks are more than welcome to improve the application making it more functional and convenient for its users.

Review. In this stage the developers acquire the capstone adviser and panelists' feedback and critics on the Android application. Once the comments are stated, the developers will prepare for the polishing of the application as stated.

Launch. Following the revisions and improvements of the application in accordance with feedbacks of the capstone adviser and panelists, the developers

will now publish the application on Google Play Store ready for users' download and use.

## AGILE METHODOLOGY



**Figure 3: Agile Methodology**

### Requirements Documentation

#### Functional Requirements

- Enable users to create an account on the application using different email providers.
- A verification email is sent to the user's email whenever s/he registers for the first time on the application.
- Allow users to sign-in using their registered e-mail or by using their Google account.
- Has a landing page and search function that will help users to select ingredients.
- Generates recipe based on the selected ingredients of the users.
- Has a wide list of recipes categorized according to the main archipelago of the Philippines (Luzon, Visayas, Mindanao).
- Renowned Filipino cuisines or recipes must be listed under "More" category.
- Integrated with a voice assistant that accepts voice inputs from the user.

- The application should be able to read recipe instructions.
- Allow users to create and join private circles.
- Allow users to share recipe publicly and privately.
- Has a module that displays the shared recipes of the user on his/her profile.

### **Non-Functional Requirements**

- The verification email must be sent within five (5) seconds after signing in with the newly created user's account.
- Signing in with Google or with the registered account must automatically use the recorded name within that email.
- The sign-in or create account module must load within five (5) seconds.
- The application shall recommend a cuisine based on what recipe can be cooked on the user's selected ingredients within three (3) seconds.
- The voice assisted technology should perform an action depending on the user's voice input.
- The voice assisted technology must respond within one (1) to two (2) seconds after saying the keyword “Friday”.
- Creating and joining a circle must be real time and with a response of no more than three (3) seconds.
- The recipe sharing feature must be real time and with a response of no more than five (5) seconds.
- Publicly shared recipes must have seventy-five (75) percent approval of the total circle members before publishing the recipe to the “Public Recipes” module.
- The application system requirement must be an Android device with a minimum Android version of Android 9 (Pie) as its operating system to successfully install the application.
- The minimum application hardware system requirement must be 4 Gigabytes RAM and octa- core CPU for optimal user experience.

## UML Use Case Diagram

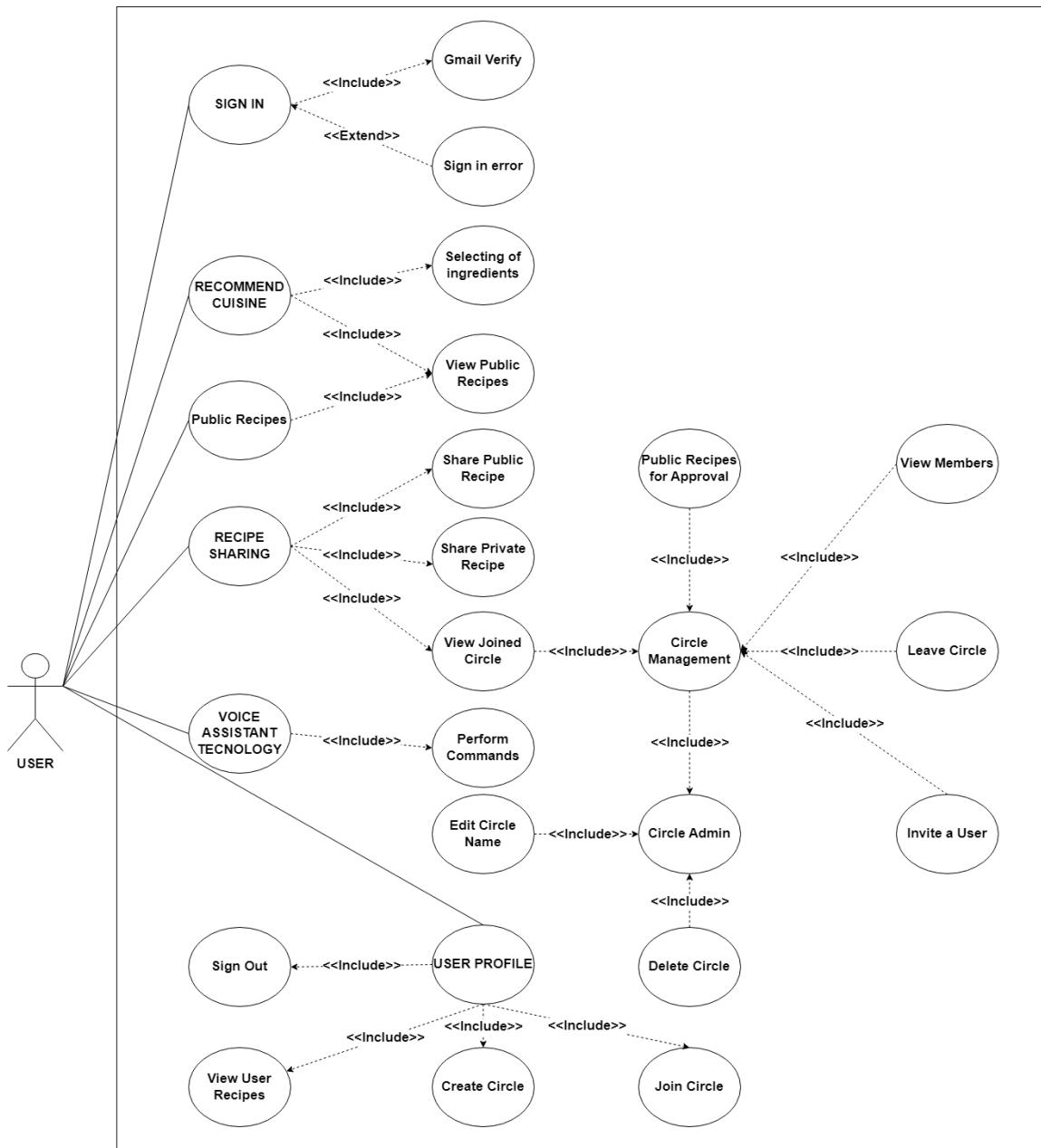


Figure 4: Process of the application

## Design of Software, System, Product, and/or Processes

### Title: Sign-in Activity

**Description:** In this activity, the user can sign-in with his personal email or sign-in with a Gmail account.

SMART 4G+ | 44% | 8:45



### Sign In

Let's find you a Filipino recipe!

Email

Password



Sign In

Or continue with

Continue with Google

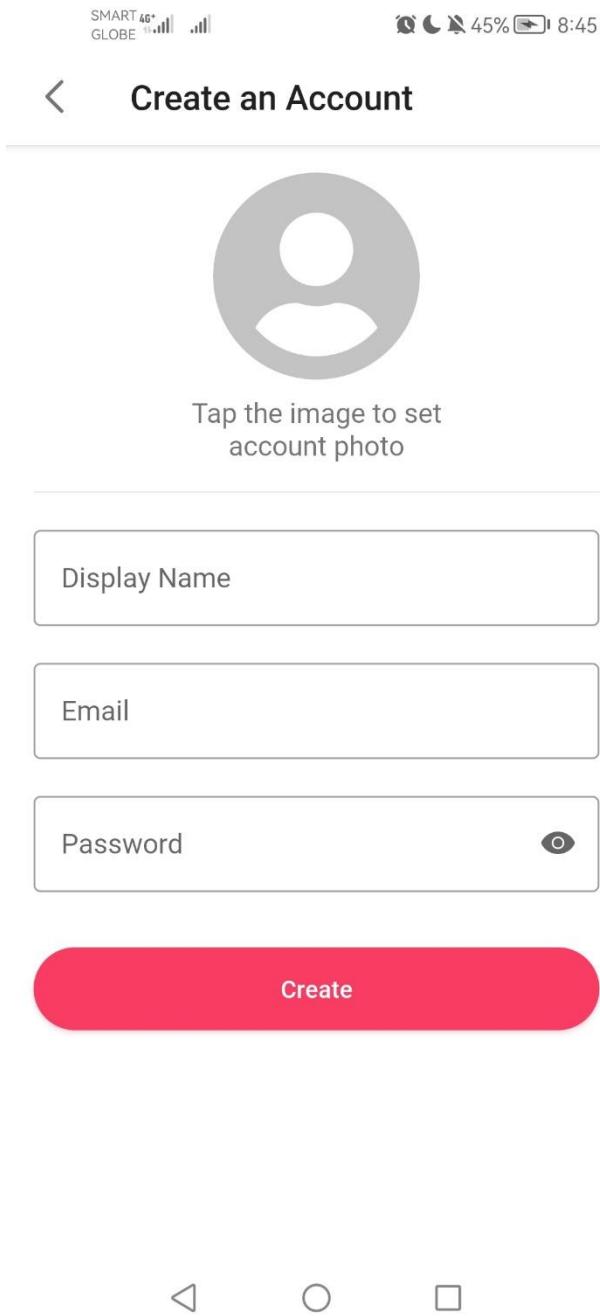
Not registered yet? [Create an Account](#)



**Figure 5: Sign-in Activity**

**Title: Create an Account Activity**

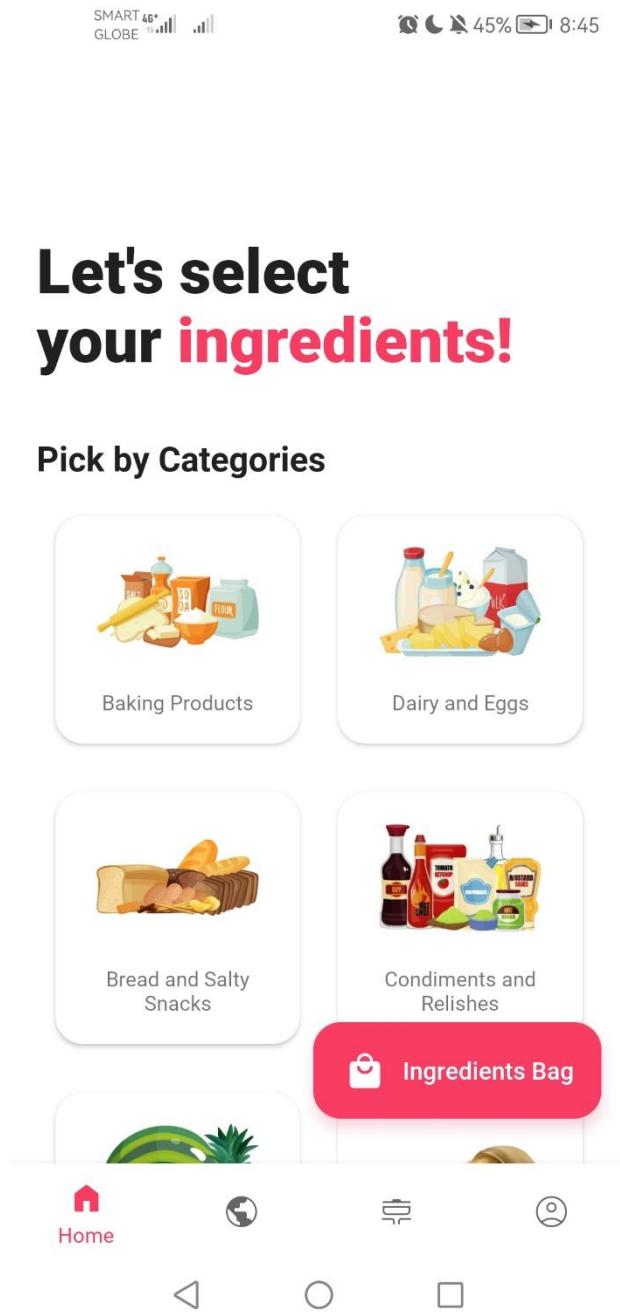
**Description:** In this activity, the user can create an account using a personal email to be used as a sign-in credential in the application.



**Figure 6: Create an Account Activity**

## Title: Home Fragment

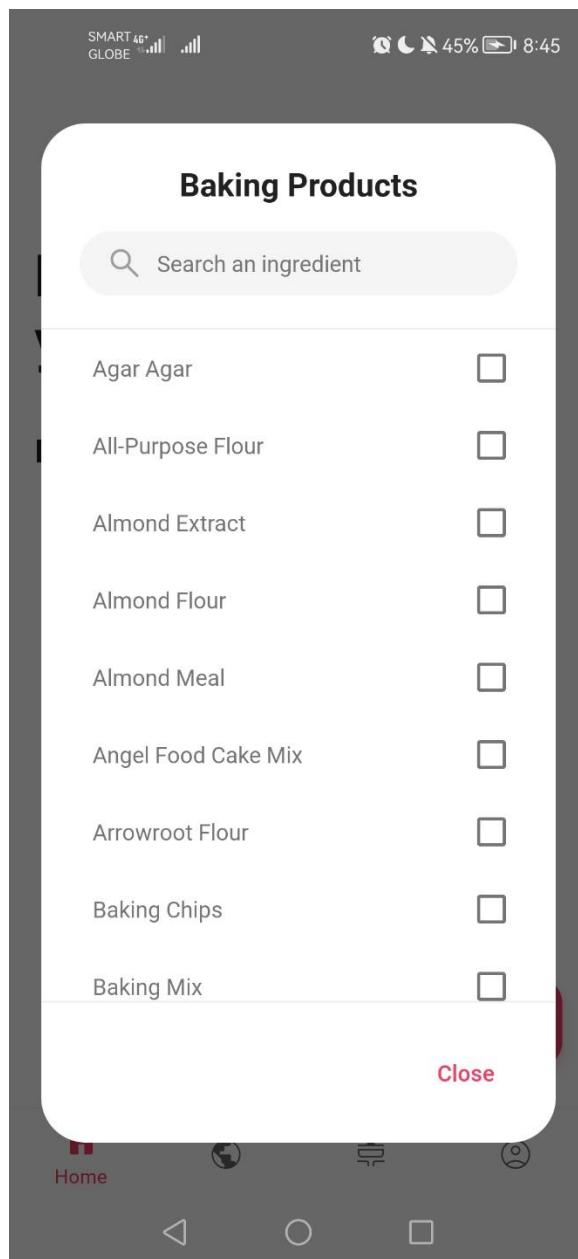
**Description:** In this fragment, the user can pick an ingredient by category to be used for the recommendation of cuisine/recipe feature.



**Figure 7: Home Fragment**

### Title: Ingredients Dialog

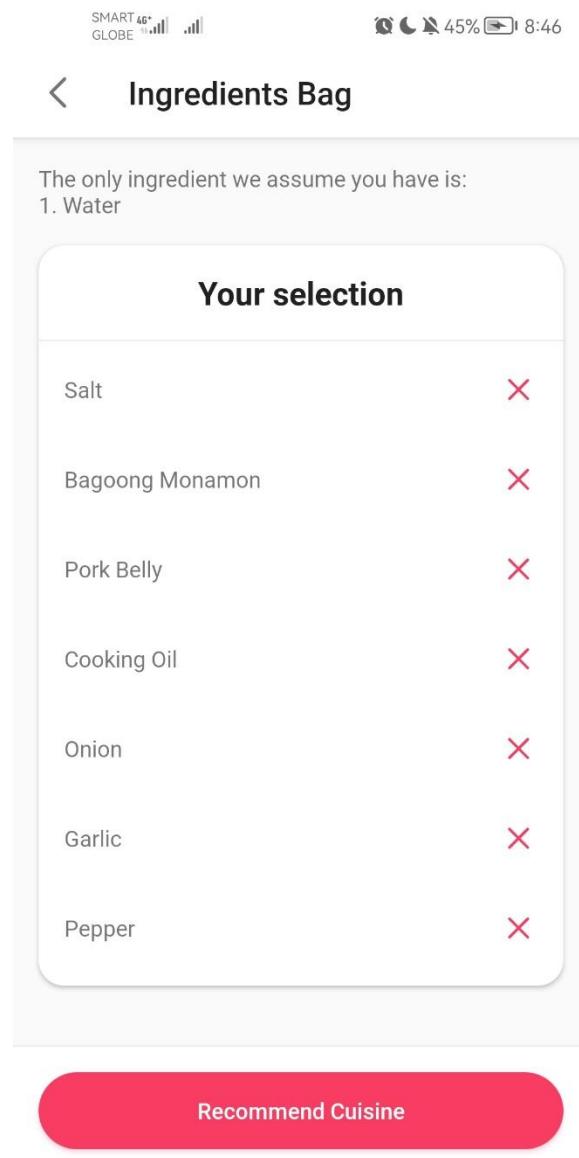
**Description:** In this dialog, the user can pick the ingredients that is available at his/her kitchen. The selected ingredients will be used for the recommendation feature.



**Figure 8: Ingredients Dialog**

## Title: Ingredients Bag Activity

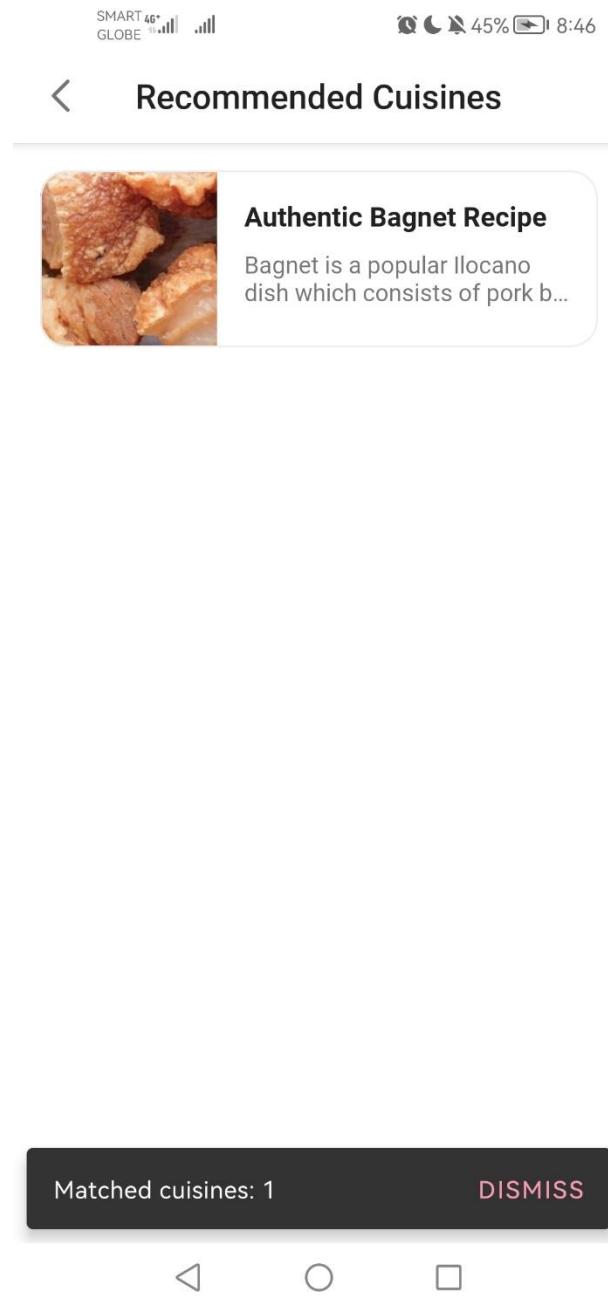
**Description:** In this activity, all the selected ingredients of the user will be shown. This is also where the user will generate a cuisine based on the list of selected ingredients.



**Figure 9: Ingredients Bag Activity**

## Title: Recommended Cuisines Activity

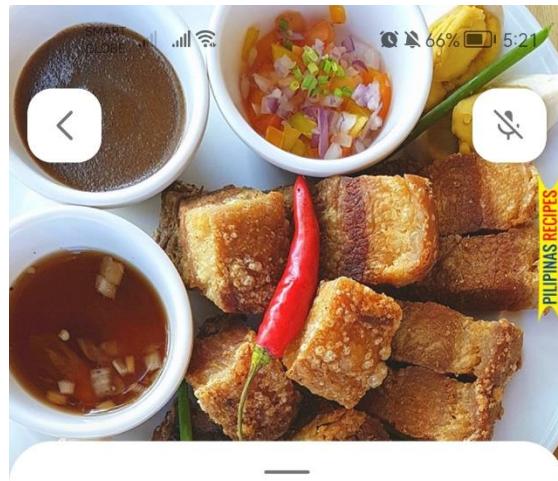
**Description:** In this activity, the recommended cuisines based on the selected ingredients by the user will be displayed.



**Figure 10: Recommended Cuisines Activity**

## Title: View Recipe Activity

**Description:** In this activity, the user can view the details of the recipe like its primary details, ingredients, and instructions. This is also where the voice assisted technology can be used.



### Crispy Bagnet Ilocos

Details

Ingredients

Instructions

#### Instructions

- 1 | Preheat oven to 175F.
- 2 | Pour water into a pressure cooker. Add lemongrass, salt, whole peppercorn, and onion. Let boil.
- 3 | Add pork belly. Boil for 3 minutes. Cover the pressure cooker. Cook for 15 minutes.
- 4 | Let the pressure out completely. Put the pork belly on a clean plate and let it cool-down for 10 minutes.



Figure 11: View Recipe Activity

## Title: Public Recipes Fragment

**Description:** In this fragment, all the public recipes is listed and categorized by cuisines' origin.

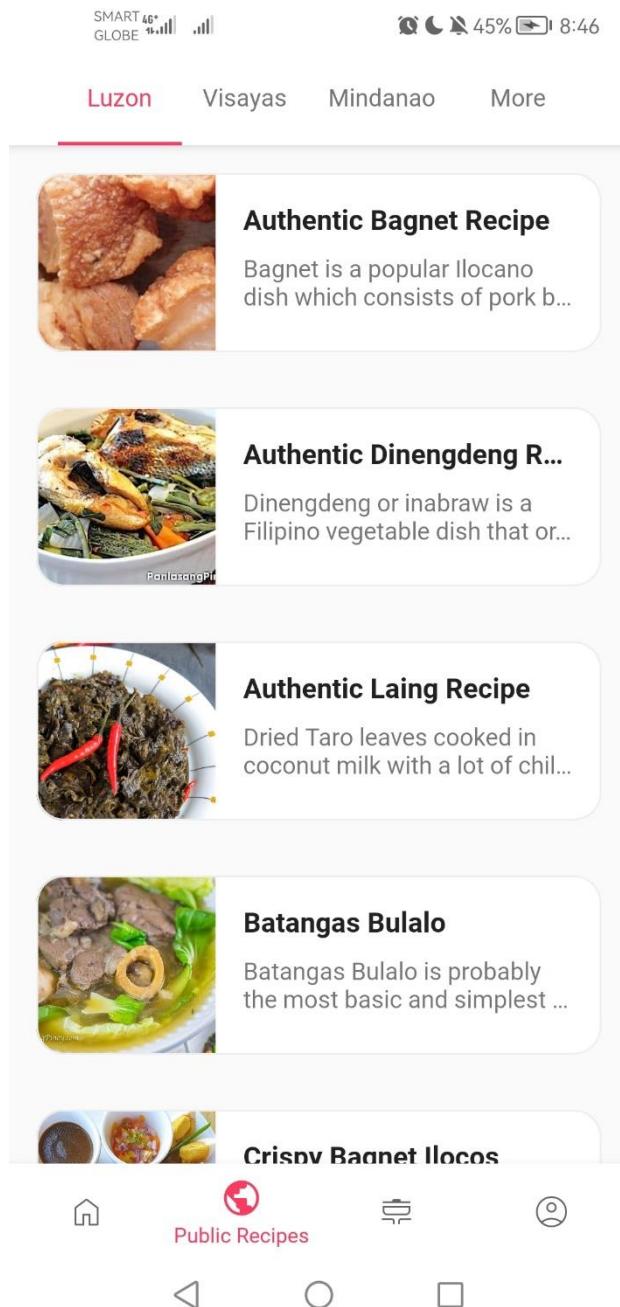
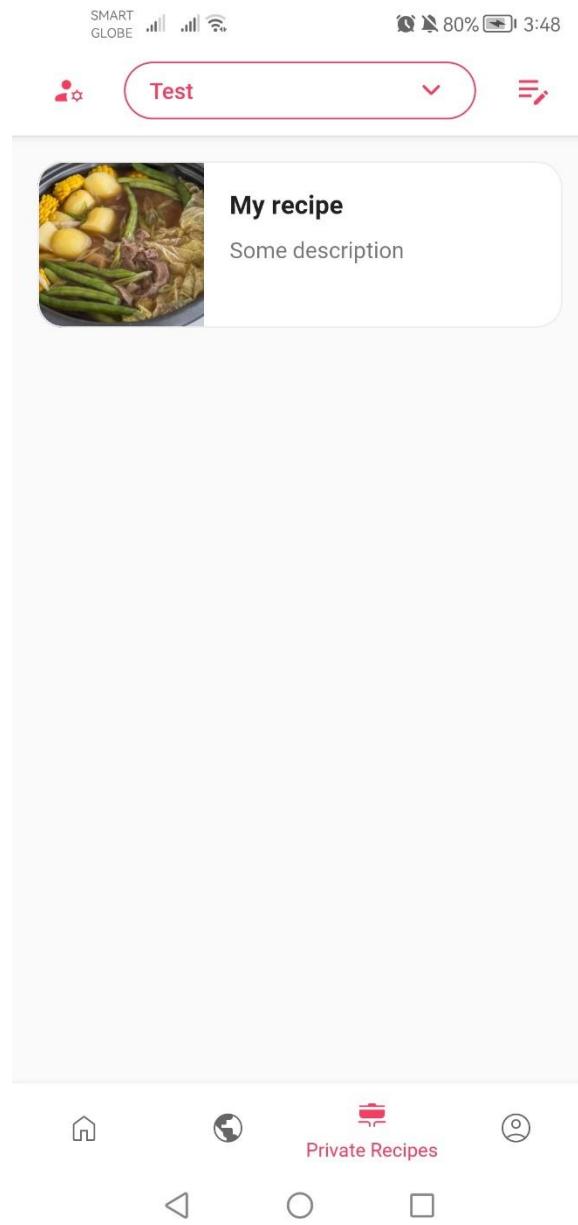


Figure 12: Public Recipes Fragment

### Title: Private Recipes Fragment

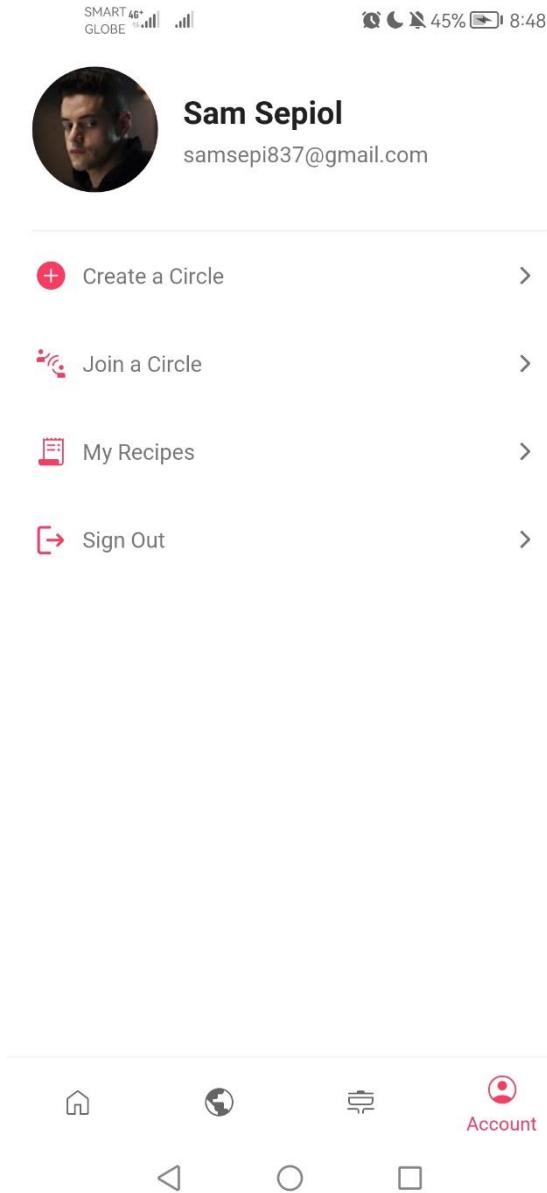
**Description:** In this fragment, the circles of the user and recipes shared in it is displayed.



**Figure 13: Private Recipes Fragment**

## Title: Account Fragment

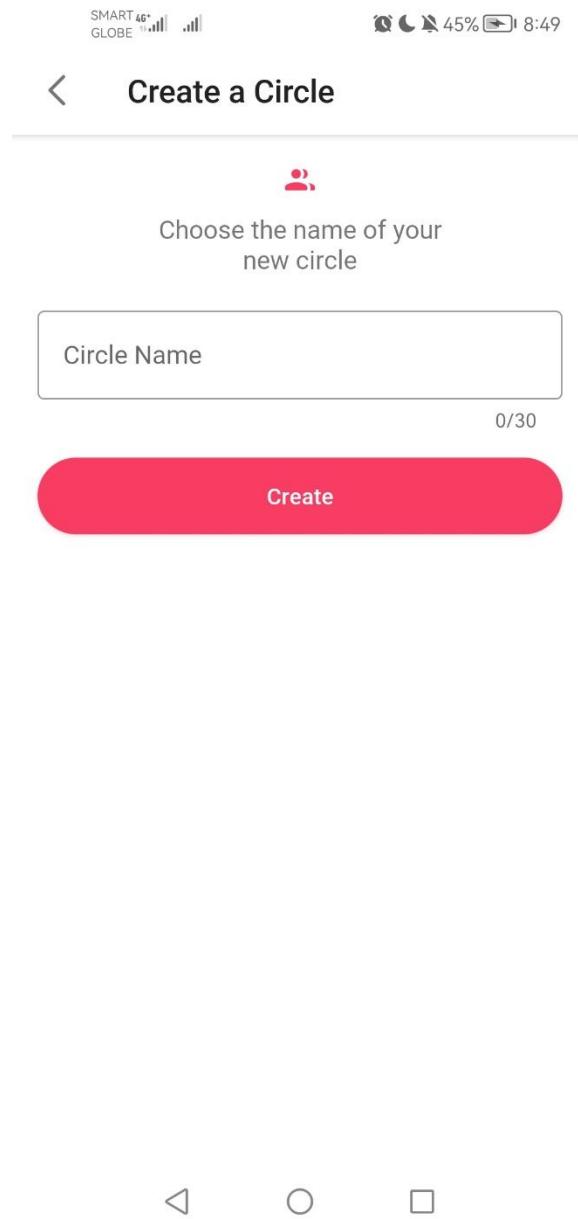
**Description:** In this fragment, the user can view his/her account details. This is also where the user can create and join a private circle and see all his/her shared recipes both private and public.



**Figure 14: Account Fragment**

**Title: Create a Circle Activity**

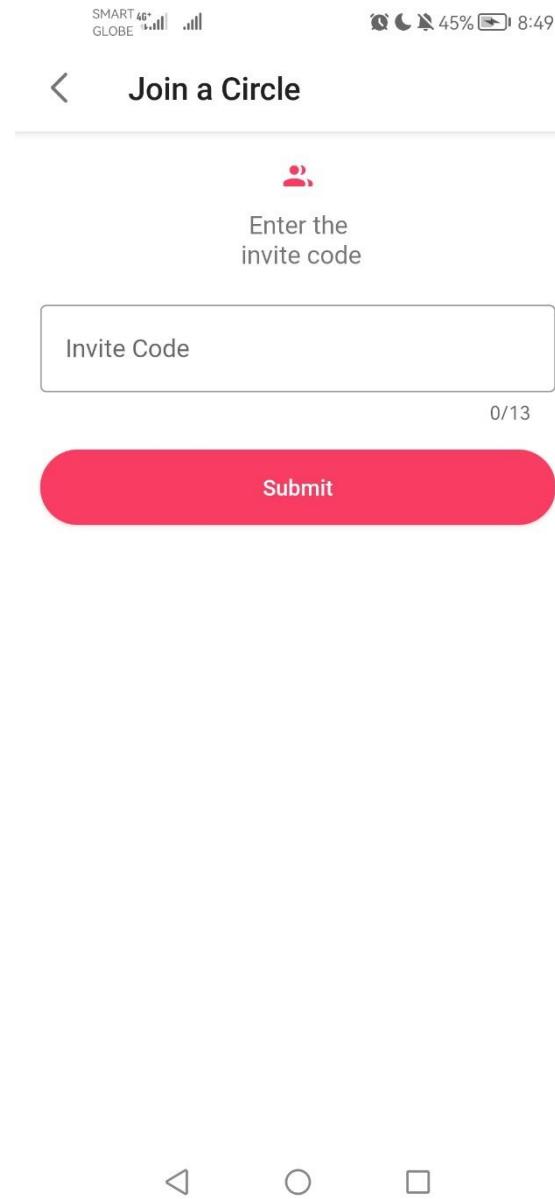
**Description:** In this activity, the user can create a private circle which can be used for sharing private recipes.



**Figure 15: Create a Circle Activity**

## Title: Join a Circle Activity

**Description:** In this activity, the user can enter the invite code of a private circle and become a member of it.



**Figure 16: Join a Circle Activity**

### Title: Join a Circle Confirm Activity

**Description:** In this activity, the user will confirm whether the circle s/he wants to join to is correct. The creator of the circle will also be shown.

SMART 4G+ .lll | .lll | 94% 9:45

**Great. You're about to join  
the Test circle.**

Here's who is waiting for you:



Sam Sepiol

Join

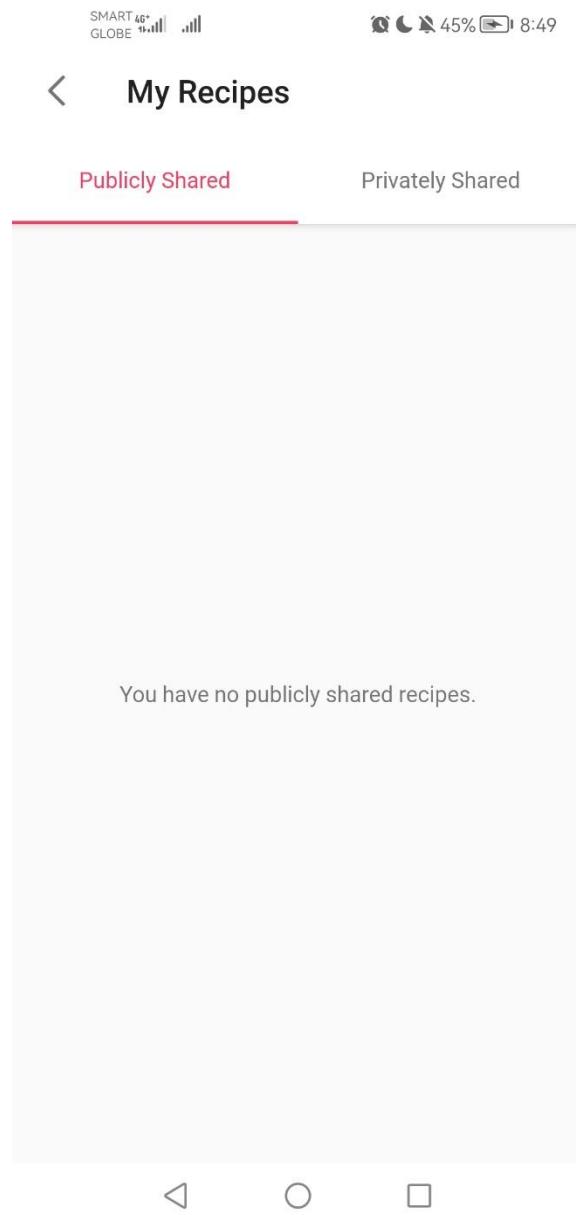
Cancel



**Figure 17: Join a Circle Confirm Activity**

## Title: My Recipes Activity

**Description:** In this activity, the user can see all his/her shared recipe both public and private.



**Figure 18: My Recipes Activity**

**Title: View Circle Info Activity**

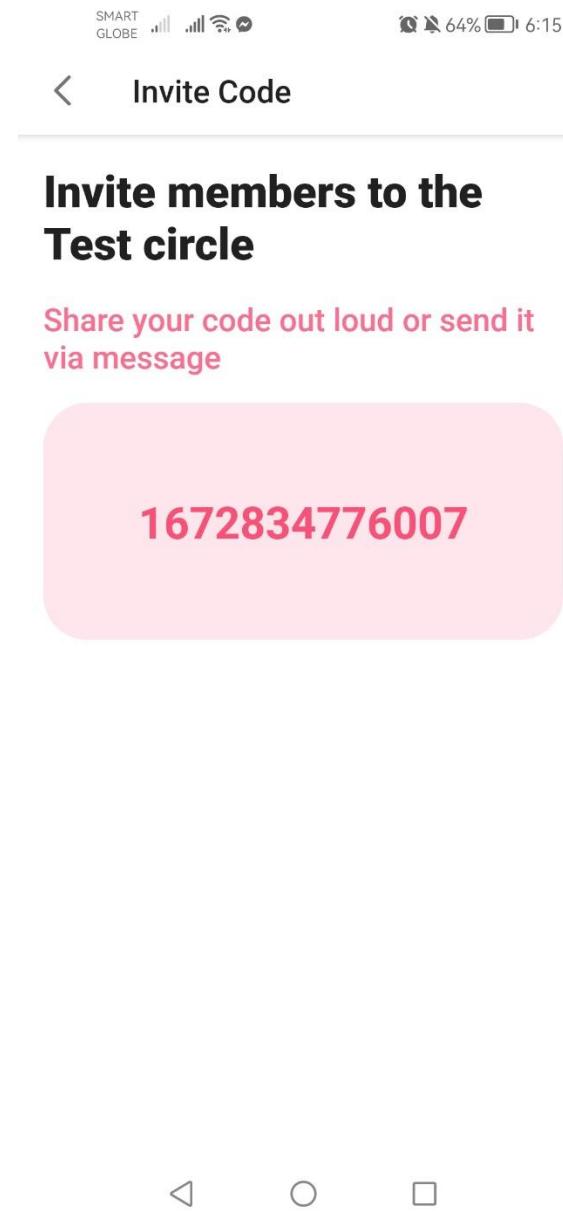
**Description:** In this activity, the user can view the primary information about the circle.

The screenshot shows a mobile application interface for a 'View Circle Info' activity. At the top, there are navigation icons: a back arrow, a search icon, a refresh icon, a 100% battery icon, and a timestamp of 10:31. Below these is a header bar with a left arrow, the text 'View Circle Info', and a three-dot menu on the right. The main content area has a title 'Test' in large bold letters, followed by '5 people' in a smaller font. A horizontal line separates this from the 'Admin' section, which lists 'Sam Sepiol' with a profile picture. Another horizontal line separates the 'Admin' section from the 'Members' section, which lists four members: 'Baymax' (with a red and white robot profile picture), 'Patrick' (with a cartoon character profile picture), 'Neo' (with a person in a dark suit profile picture), and 'Snorlax' (with a blue and white dog profile picture). At the bottom of the screen are three large, light-gray navigation icons: a triangle pointing left, a circle, and a square.

**Figure 19: View Circle Info Activity**

**Title: Invite a User Activity**

**Description:** In this activity, the user can view and copy the invite code of the circle which can be used by other members to join.



**Figure 20: Invite a User Activity**

## Title: Share Public Recipe Activity

**Description:** In this activity, the user will enter the details of the recipe as well as its instructions and ingredients.

SMART 4G+ | 100% | 10:33

Share to Public

Step 1 of 6

### Name and Origin

Name

0/100

luzon

BACK NEXT

◀ ○ □

**Figure 21: Share Public Recipe Activity**

**Title: Share Private Recipe Activity**

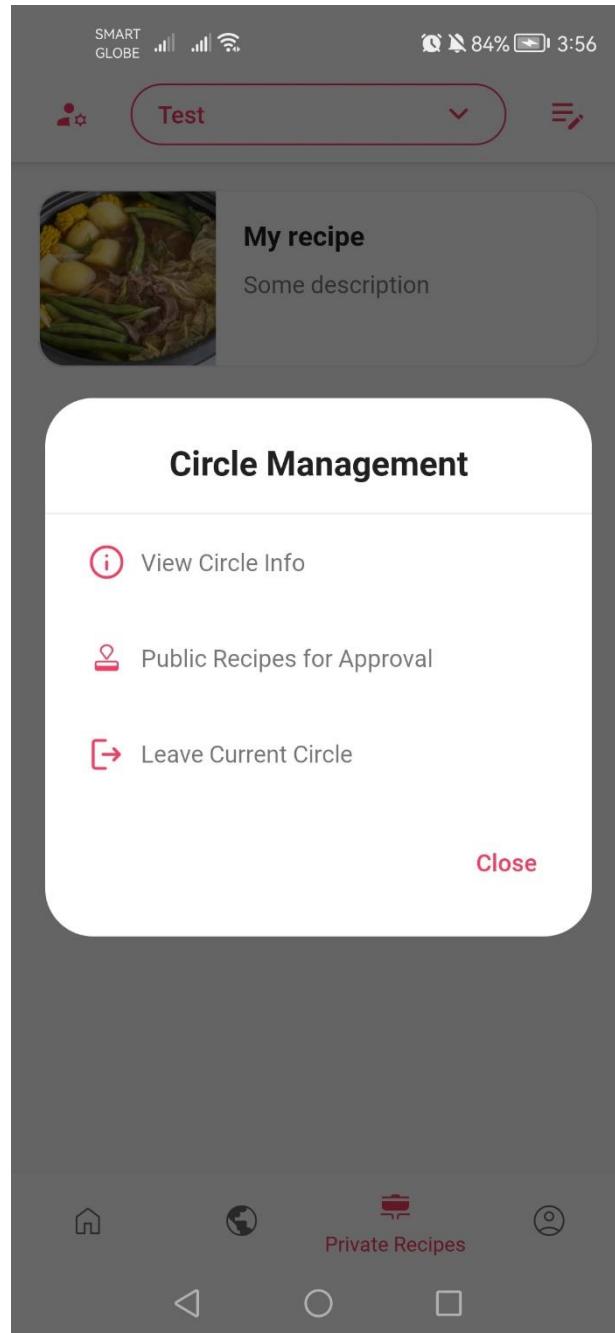
**Description:** In this activity, the user will enter the details of the recipe as well as its instructions and ingredients.

The image shows a smartphone screen displaying a mobile application titled "Share to Test". At the top, there is a status bar with icons for signal strength, battery level (100%), and time (10:31). Below the status bar, the app title "Share to Test" is centered above a horizontal line. To the left of the line is a back arrow icon, and to the right is a gear icon. The main content area is titled "Step 1 of 5" and "Name and Origin". It contains two input fields: one for "Name" (with a character count of 0/100) and another for "Origin" containing the text "luzon". At the bottom, there are navigation buttons: "BACK" (left), "NEXT" (right, in red), and three circular icons (triangle, circle, square).

**Figure 22: Share Private Recipe Activity**

**Title: Circle Management Dialog**

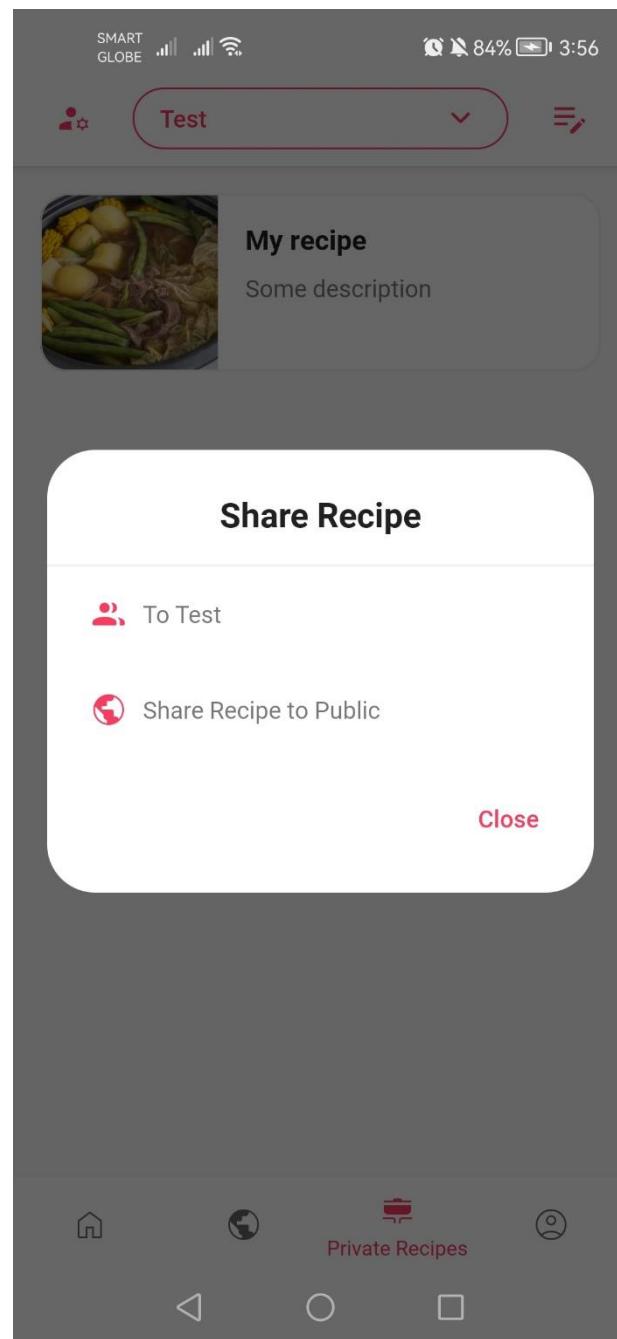
**Description:** In this dialog, the user can select what activity to do in the circle.



**Figure 23: Circle Management Dialog**

### Title: Share Recipe Dialog

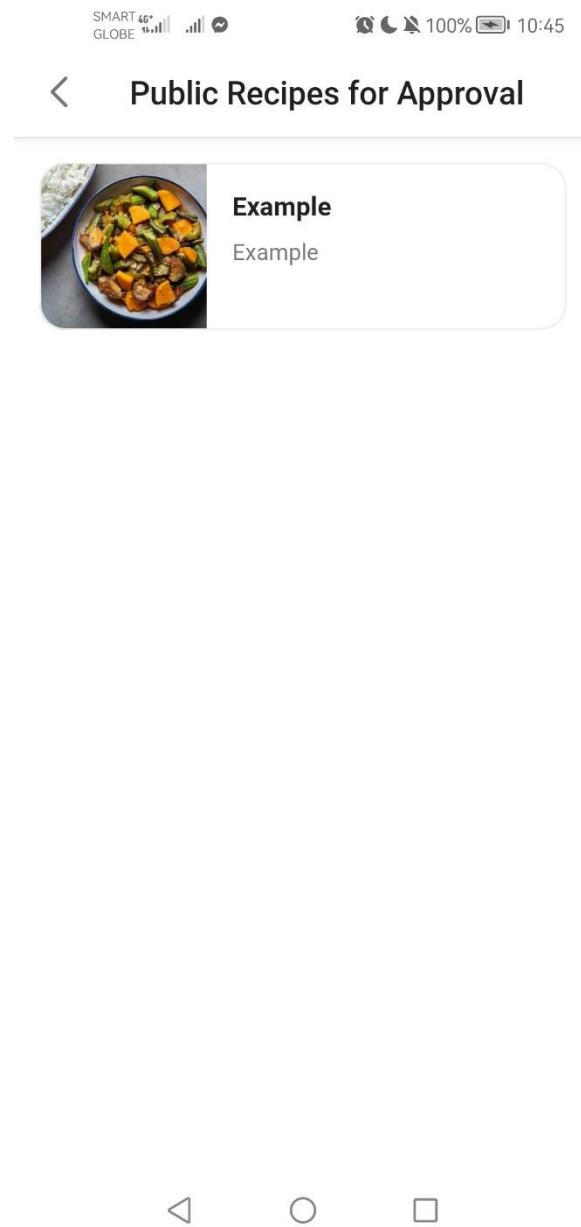
**Description:** In this dialog, the user has the option whether to share a recipe privately or publicly.



**Figure 24: Share Recipe Dialog**

## Title: Public Recipes for Approval Activity

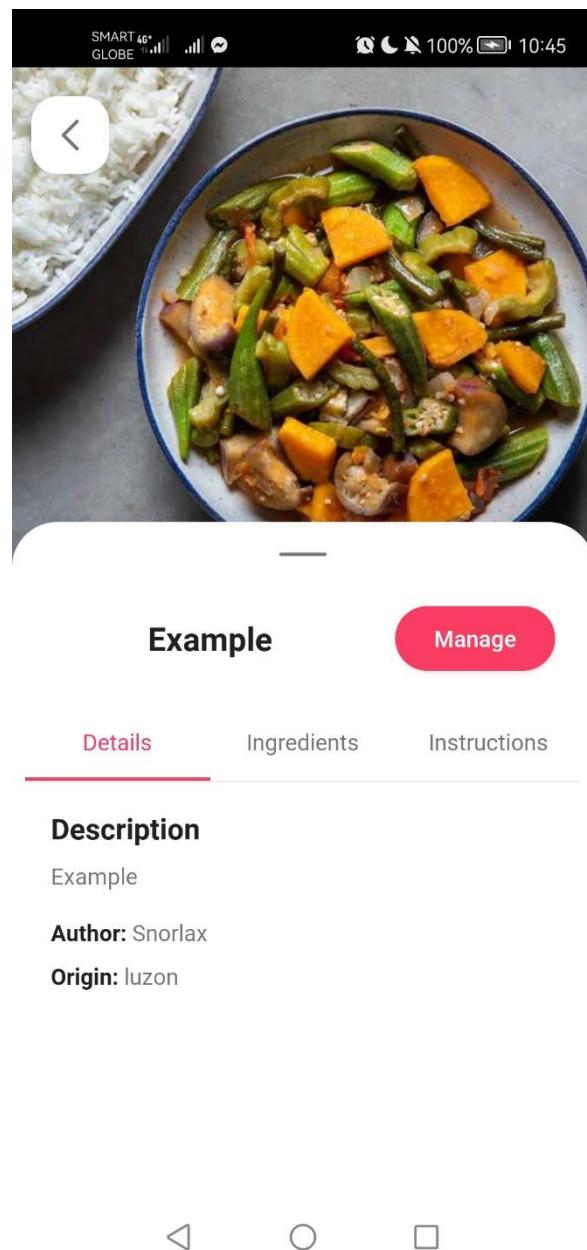
**Description:** In this activity, the member of a circle can view the public recipes s/he needed to approve or reject.



**Figure 25: Public Recipes for Approval Activity**

## Title: View Recipe for Approval Activity

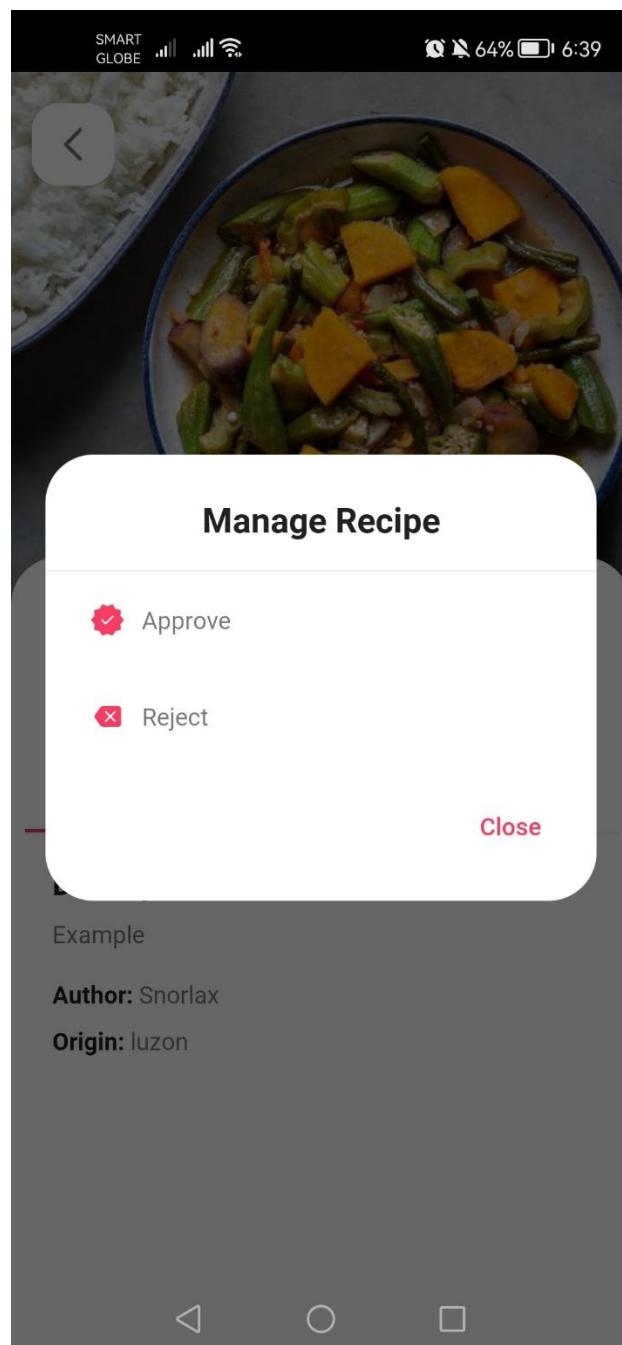
**Description:** In this activity, the voter can view the details of the recipe to determine whether to approve or reject it.



**Figure 26: View Recipe for Approval Activity**

**Title:** Manage Recipe (Vote) Dialog

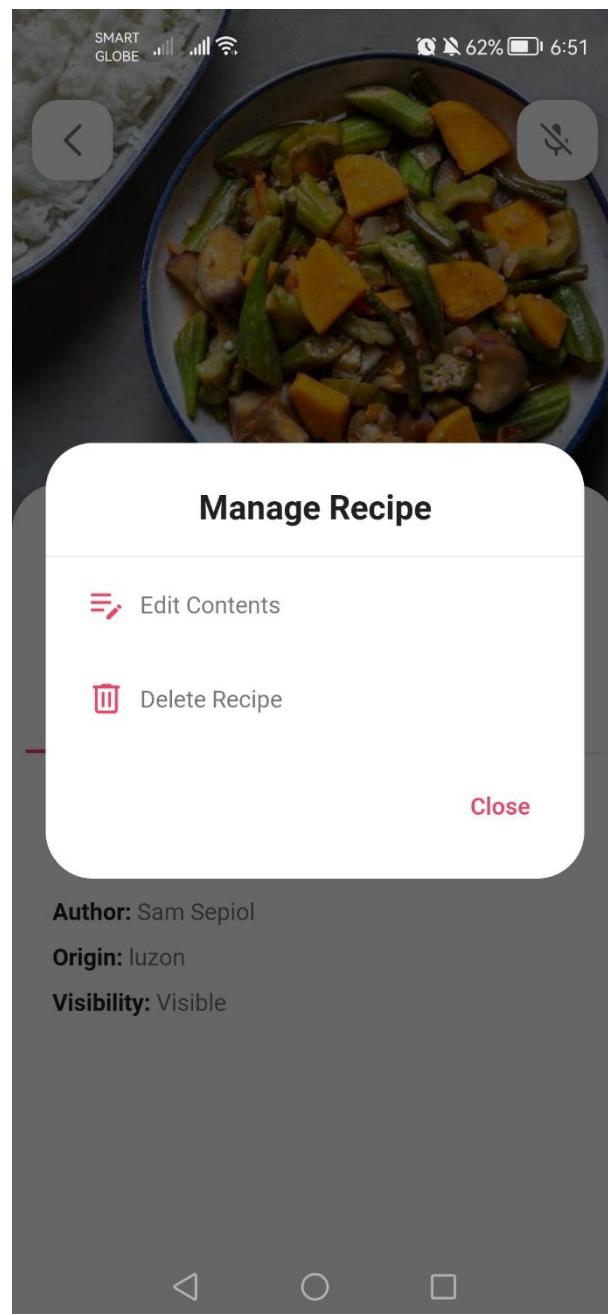
**Description:** In this dialog, the voter of the recipe can select whether to approve or reject the public recipe to be shared.



**Figure 27: Manage Recipe (Vote) Dialog**

### Title: Manage Recipe (Author) Dialog

**Description:** In this dialog, the author of the recipe has the option whether to edit the content of the recipe or delete it permanently.



**Figure 28: Manage Recipe (Author) Dialog**

## **Development and Testing**

### **Interpretation**

The weighted mean is the mean that is derived from calculating the weight (or probability) associated with the event or outcome. This assignment of different weights gives us the flexibility to assign more power to the more relevant data point and less power to a less relevant data point.

**Objective 1:** To design and develop an Android application that can recommend Filipino cuisines based on available ingredients.

**Table 2: Objective 1 Weighted Mean**

CUISINE RECOMMENDATION						
Items	Strongly Disagree	Disagree	Agree	Strongly Agree	Weighted Mean	Verbal Interpretation
The cuisine recommendation helps the user to quickly decide what to cook.		1	9	20	3.63	Strongly Agree
The cuisine recommendation filters the recipes depending on the selected ingredients.			13	17	3.56	Strongly Agree
The process of selecting ingredients for the cuisine recommendation is straightforward and easy.		1	7	22	3.70	Strongly Agree
Area Mean					3.63	Strongly Agree

**Discussion:** The indicated average weighted mean is “3.63” which have a verbal interpretation of “Strongly Agree”. The analysis states that the application meets the objective to develop an Android application that can recommend Filipino cuisines based.

**Objective 2:** To design and develop an Android application that will enable the users to share Filipino recipes with each other.

**Table 3: Objective 2 Weighted Mean**

RECIPE SHARING						
Items	Strongly Disagree	Disagree	Agree	Strongly Agree	Weighted Mean	Verbal Interpretation
The application allows the user to share their own recipe.			5	25	3.83	Strongly Agree
The application allows the user to edit and delete their shared recipe.			6	24	3.80	Strongly Agree
The recipe sharing helps the user to discover new recipes.			8	22	3.73	Strongly Agree
The application allows the user to share recipes privately and publicly.			4	26	3.86	Strongly Agree
The process of creating and joining a circle is straightforward and easy.			8	22	3.73	Strongly Agree
<b>Area Mean</b>					<b>3.79</b>	<b>Strongly Agree</b>

**Discussion:** The indicated average weighted mean is “3.79” which have a verbal interpretation of “Strongly Agree”. The analysis states that the application meets the objective to design and develop an Android application that will enable the users to share Filipino recipes with each other.

**Objective 3:** To design and develop an Android application with voice assisted technology.

**Table 4: Objective 3 Weighted Mean**

VOICE ASSISTED TECHNOLOGY						
Items	Strongly Disagree	Disagree	Agree	Strongly Agree	Weighted Mean	Verbal Interpretation
The voice assistant helps the user to cook efficiently.		1	12	17	3.53	Strongly Agree
The voice assistant can easily recognize the voice commands.		1	15	14	3.43	Agree
The voice assistant can follow the user's commands.		1	10	19	3.60	Strongly Agree
<b>Area Mean</b>					<b>3.52</b>	<b>Strongly Agree</b>

**Discussion:** The indicated average weighted mean is “3.52” which have a verbal interpretation of “Strongly Agree”. The analysis states that the application meets the objective to design and develop an Android application with voice assisted technology.

**Functionality and Usability:** The application's functionality and usability, such as user-friendliness, performance, and design and interface.

**Table 5: Functionality and Usability (User-Friendliness) Weighted Mean**

USER - FRIENDLINESS						
Items	Strongly Disagree	Disagree	Agree	Strongly Agree	Weighted Mean	Verbal Interpretation
The purpose of the application increases the user's interest in cooking.		2	9	19	3.56	Strongly Agree
The application has a clear and consistent layout.			5	25	3.83	Agree
Minimum controls are needed to operate the application.		2	12	16	3.46	Strongly Agree
<b>Area Mean</b>					<b>3.61</b>	<b>Strongly Agree</b>

**Discussion:** The indicated average weighted mean is “3.61” which have a verbal interpretation of “Strongly Agree”. The analysis states that the application meets the functionality and usability that the application is user-friendly.

**Table 6: Functionality and Usability (Performance) Weighted Mean**

PERFORMANCE						
Items	Strongly Disagree	Disagree	Agree	Strongly Agree	Weighted Mean	Verbal Interpretation
The size of the application is lightweight.		1	8	21	3.66	Strongly Agree
The application's processing of data is quick.		1	9	20	3.63	Strongly Agree
The application displays information in real-time.		1	8	21	3.66	Strongly Agree
Area Mean					<b>3.65</b>	<b>Strongly Agree</b>

**Discussion:** The indicated average weighted mean is “3.65” which has a verbal interpretation of “Strongly Agree”. The analysis states that the application meets the functionality and usability that the application is well-performing.

**Table 7: Functionality and Usability (Design and Interface) Weighted Mean**

DESIGN AND INTERFACE						
Items	Strongly Disagree	Disagree	Agree	Strongly Agree	Weighted Mean	Verbal Interpretation
The size of the application is lightweight.			9	21	3.70	Strongly Agree
The application's processing of data is quick.			6	24	3.80	Strongly Agree
The application displays information in real-time.			10	20	3.66	Strongly Agree
Area Mean					3.72	Strongly Agree

**Discussion:** The indicated average weighted mean is “3.72” which has a verbal interpretation of “Strongly Agree”. The analysis states that the application meets the functionality and usability that the design and interface is favorable.

## Description of Prototype

Kusinasyon: An Android-based Filipino Cuisine Application with Voice Assisted Technology is an application that primarily recommends cuisine/recipe that can be cooked based on the available or selected ingredients of the user. It is ideal for users who do not have enough idea in recipes since foods are combination of ingredients. This works by allowing the users to select from a given list of ingredients. After that, the selected ingredients will be collected and matched by the application to the accessible recipes. Whenever a cuisine or recipe is matched, it will automatically recommended to the user as a cuisine that can be cooked.

Additionally, the application has a recipe sharing feature in which the users can use to share their recipes publicly or privately. Publicly shared recipes can be seen by all users of the application. On the other hand, privately shared recipes can only be seen by the users which are a part of the private circle that is either created or joined by a user.

Moreover, the application features voice assisted technology in which can be utilized by the users to have an assistant on the recipe instructions during the cooking process. This is helpful for users who are conscious about dirtying the screen of their smartphone while cooking.

## **CONCLUSIONS AND RECOMMENDATIONS**

This chapter covers the summary of the results based on the methodology, results, and discussion chapter, the conclusion acquired from the findings, and the recommendations made by the developers.

### **Conclusion**

The first objective, which is to develop an Android application that can recommend Filipino cuisines based on available ingredients. Given the result obtained by the developers by conducting a survey, that have indicated average weighted mean of “3.63” which have a verbal interpretation of “Strongly Agree”. It can be safe to conclude that the first objective was successfully achieved by the developers. The result signifies that the Android application can help the user to decide easily and quickly what to cook with the help of cuisine recommendation.

The second objective is to design and develop an Android application that will enable the users to share Filipino recipes with each other. Given the result obtained by the developers by conducting a survey, that have indicated average weighted mean of “3.79” which have a verbal interpretation of “Strongly Agree”. It can be safe to conclude that the second objective was successfully achieved by the developers. The result signifies that the Android application will help the users to have a medium to share their own recipe and acquire new idea about Filipino cuisine.

The third objective is to design and develop an Android application with voice assisted technology. Given the result obtained by the developers by conducting a survey, that have indicated average weighted mean of “3.52” which have a verbal interpretation of “Strongly Agree”. It can be safe to conclude that the third objective was successfully achieved by the developers. The result signifies that the Android application’s voice-assisted technology is efficient and responsive to users’ commands. Overall, the developed Android application achieved all the requirements stated to complete the project.

## **Recommendation**

After completing the project and final defense, the project developers recommend to future developers to add an additional interaction feature in building a social-media-like app to enable users have more exchange of information with each other. This interaction feature can be a chat system or a comment for each post inside the application. Additionally, the project developers recommend utilizing the features of voice technologies since it adds a great feature and function to an application. Voice technologies can be a way for the users to perform tasks with such an ease which can be a great innovation for a great future.

## **Bibliography**

- **Activity** - a single, focused thing that the user can do. Almost all activities interact with the user, so the Activity class takes care of creating a window for you in which you can place your UI.
- **Acceptance testing** - a quality assurance (QA) process that determines to what degree an application meets end users' approval.
- **Agile** - a software development refers to a group of software development methodologies based on iterative development, where requirements and solutions evolve through collaboration between self-organizing cross-functional teams.
- **AI** - is the simulation of human intelligence processes by machines, especially computer systems. Specific applications of AI include expert systems, natural language processing, speech recognition and machine vision.
- **Android Studio** - the official integrated development environment (IDE) for Android application development. It is based on the IntelliJ IDEA, a Java integrated development environment for software, and incorporates its code editing and developer tools.

- **Android Virtual Device (AVD)** - a configuration that defines the characteristics of an Android phone, tablet, Wear OS, Android TV, or Automotive OS device that you want to simulate in the Android Emulator.
- **API** - stands for Application Programming Interface. In the context of APIs, the word Application refers to any software with a distinct function. Interface can be thought of as a contract of service between two applications.
- **API Levels** - generally mean that as a programmer, you can communicate with the devices' built-in functions and functionality.
- **Azure Speech Service** - The Speech service provides speech-to-text and text-to-speech capabilities with an Azure Speech resource. You can transcribe speech to text with high accuracy, produce natural-sounding text-to-speech voices, translate spoken audio, and use speaker recognition during conversations.
- **Backend** - relating to or denoting the part of a computer system or application that is not directly accessed by the user, typically responsible for storing and manipulating data.
- **Bug** - an unexpected problem with software or hardware. Typical problems are often the result of external interference with the program's performance that was not anticipated by the developer.
- **Circle** - a group of people with shared professions, interests, or acquaintances.
- **Components** - Application components are the essential building blocks of an Android application.
- **Conforming** - is the degree of adherence to preset expectations. The degree to which a product meets its prespecified criteria is termed as conformance in the context of software engineering.
- **Embedded** - software that is placed permanently inside some kind of device to perform a very specific set of functions.

- **Extensible Markup Language (XML)** - is a simple text-based format for representing structured information: documents, data, configuration, books, transactions, invoices, and much more.
- **Fragment** - represents a reusable portion of your app's UI. A fragment defines and manages its own layout, has its own lifecycle, and can handle its own input events.
- **Git** - a free and open-source distributed code management and Version control system that is distributed under the GNU General Public License version 2.
- **GitHub** - a web-based version-control and collaboration platform for software developers.
- **Hardware** - refers to the external and internal devices and equipment that enable you to perform major functions such as input, output, storage, communication, processing, and more.
- **Integrated development environment (IDE)** - a software application that helps programmers develop software code efficiently. It increases developer productivity by combining capabilities such as software editing, building, testing, and packaging in an easy-to-use application. Hardware resources - the assignable, addressable bus paths that allow peripheral devices and system processors to communicate with each other. Hardware resources typically include I/O port addresses, interrupt vectors, and blocks of bus-relative memory addresses.
- **Integration** - the act of bringing together smaller components or information stored in different subsystems into a single functioning unit.
- **Intervention** - the act of interfering with the outcome or course especially of a condition or process (as to prevent harm or improve functioning).
- **Latency** - is a noun referring to something inactive, dormant, or lying-in wait. It's the state of not being visibly active yet. In simple terms it is the delay.

- **Layout** - is the process of calculating the position of objects in space subject to various constraints. This functionality can be part of an application or packaged as a reusable component or library.
- **Lightweight** - In computing, lightweight software also called lightweight program and lightweight application, is a computer program that is designed to have a small memory footprint (RAM usage) and low CPU usage, overall, a low usage of system resources.
- **Likert scale** - (typically) provides five possible answers to a statement or question that allows respondents to indicate their positive-to-negative strength of agreement or strength of feeling regarding the question or statement.
- **Material design** - an Android-oriented design language created by Google, supporting onscreen touch experiences via cue-rich features and natural motions that mimic real-world objects.
- **Natural language processing** - is the ability of a computer program to understand human language as it is spoken and written.
- **Operating system (OS)** - manages all the software and hardware on the computer. It is also the most important software that runs on a computer. It manages the computer's memory and processes.
- **Picovoice AI** - Picovoice's unique approach to speech recognition offers affordable and accessible voice AI to anyone and any device, resulting in an immersive experience.
- **Programming language** - is a way for programmers (developers) to communicate with computers.
- **Prototype** - a draft version of a product that allows you to explore your ideas and show the intention behind a feature or the overall design concept to users before investing time and money into development.
- **RAM** - Random-access memory is a form of computer memory that can be read and changed in any order, typically used to store working data and machine code.

- **Reasoning engine** - is an application or subsystem that makes logical inferences based on a set of axioms (the rules) and input data.
- **Real-time computing** - also known as reactive computing, is used to describe a computer system that reacts to events by performing tasks within a specific time interval.
- **Requirements analysis** - also called requirements engineering, is the process of determining user expectations for a new or modified product. These features, called requirements, must be quantifiable, relevant, and detailed.
- **Responsive design** - an approach to web page creation that makes use of flexible layouts, flexible images, and cascading style sheet media queries.
- **Software** - the programs and other operating information used by a computer.
- **Software resource** - a logical concept that is obtained by viewing the base hardware resource notion as something that is more purpose oriented at the application level.
- **Skyrocketed** - to rise extremely quickly or make extremely quick progress towards success.
- **Technology evaluation** - a set of principles, methods, and techniques/tools for effectively assessing the potential value of a technology and its contribution to a company, a region, or an industrial sector.
- **User Interface** - is the process designers use to build interfaces in software or computerized devices, focusing on looks or style. Designers aim to create interfaces which users find easy to use and pleasurable.
- **Use case diagram** - a way to summarize details of a system and the users within that system. It is generally shown as a graphic depiction of interactions among different elements in a system.
- **Voice – assisted technology** - Voice recognition technology is a software program or hardware device that can decode the human voice. Sometimes referred to as voice-activated or speech recognition software, this

technology has become more and more popular in recent years among everyday consumers.

- **Voice - based reporting** – Generating reports can be done using voice activation technology based on the collected data.
- **Voice - only reporting** – only process voice generated process on doing task and collecting of data.
- **Wake word** - a word or words that you say to make an electronic device, or a feature on a device, ready to work.

## **APPENDICES**

## **APPENDIX A (REFERENCES)**

## REFERENCES

- Magner, E. (2019, April 15). People Are Cooking Less Than Ever Before—Here's Why We're Determined to Change That. Well+Good. <https://www.wellandgood.com/home-cooking-decline/>
- Melore, C. (2021, July 5). Millennials are most adventurous in the kitchen — but know least about cooking & food safety. Study Finds. <https://www.studyfinds.org/millennials-kitchen-knowledge/>
- Brown, K. (2021, February 21). A whole generation that can't cook. Grampian Online. <https://www.grampianonline.co.uk/news/a-whole-generation-that-can-t-cook-228811/>
- Mendiola, I. (2022, April 28). Identifying Filipino cuisine through its regions. Food Philippines. <https://foodphilippines.com/story/identifying-filipino-cuisine-through-its-regions/>
- I. (2020, May 2). What Makes Filipino Cuisine Unique? Nanam. <https://nanam.co.nz/2018/10/04/what-makes-filipino-cuisine-unique/>
- Chang, L., & Stolyar, B. (2019, January 28). 'In the Kitchen' app helps you cook using voice commands. Digital Trends. <https://www.digitaltrends.com/mobile/app-attack-in-the-kitchen/>
- Garcia, M. B., Mangaba, J. B., & Tanchoco, C. C. (2021). Acceptability, Usability, and Quality of a Personalized Daily Meal Plan Recommender System: The Case of Virtual Dietitian. In 2021 13th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management.
- Villanueva, J. (2020, August 31). Food-related online community flourishes during lockdown. Philippine News Agency. Retrieved April 27, 2022, from <https://www.pna.gov.ph/articles/1113946>
- Nacario, B. (2021, August 26). Insight: Making voice technology work for everyone. Adobo Magazine Online. Retrieved May 7, 2022, from <https://www.adobomagazine.com/insight/insight-making-voice-technology-work-for-everyone/>
- Trang Tran, T. N., Atas, M., Felfernig, A., & Stettinger, M. (2018). An overview of recommender systems in the healthy food domain. Journal of Intelligent Information Systems, 50(3), 501-526.

Taylor, C. (2021, January 2). Sharing food and recipes has the power to bring people together. ABC News. Retrieved May 9, 2022, from <https://www.abc.net.au/news/2021-01-02/sharing-food-and-recipes-has-the-power-to-bring-people-together/12947408>

Liu, Y. C., Chen, C. H., Lin, Y. S., Chen, H. Y., Irianti, D., Jen, T. N., ... & Chiu, S. Y. H. (2020). Design and usability evaluation of mobile voice-added food reporting for elderly people: randomized controlled trial. *JMIR mHealth and uHealth*, 8(9), e20317.

Cruz, Z. M. C., Alpay, J. J. R., Depeno, J. D. D., Altabirano, M. J. C., & Bringula, R. (2017, October). Usability of "Fatchum" A Mobile Application Recipe Recommender System. In Proceedings of the 6th Annual Conference on Research in Information Technology (pp. 11-16).

Marcial, D. E., Onte, M. B., & Mana-ay, A. K. (2017). Developing Online Database of Food Recipes with Indigenous Ingredients. *International Journal of Scientific Engineering and Science*, 1(10), 50-54.

Guigue, J. R. P., Pogado, R. G. E., Marata, T. C. C., & Aborde, M. V. (2019). A voice-based multi-platform first aid application using the Jaccard similarity index algorithm.

Ribeiro, D., Machado, J., Ribeiro, J., Vasconcelos, M. J. M., Vieira, E. F., & De Barros, A. C. (2017, October). SousChef: Mobile Meal Recommender System for Older Adults. In *ICT4AgeingWell* (pp. 36-45).

Tsai, T. H., Chang, H. T., Hsu, C. Y., Lin, S. Y., Yan, W. C., & Chen, Y. C. (2018, July). A Personal Emotion-Based Recipe Recommendation Mobile Social Platform: Mood Canteen. In *International Conference on Universal Access in Human-Computer Interaction* (pp. 574-582). Springer, Cham.

Angara, P., Jiménez, M., Agarwal, K., Jain, H., Jain, R., Stege, U., ... & Ng, J. W. (2017, November). Foodie fooderson a conversational agent for the smart kitchen. In *CASCON* (pp. 247-253).

## **APPENDIX B (RESOURCE PERSON/S)**

## RESOURCE PERSON/S

STI College Global City  
ICT Department  
**PROOFREADER'S CERTIFICATION**

### IT CAPSTONE PROJECT

Control No.	2201 -	1	2
-------------	--------	---	---

#### Certification

Date	January 14, 2023
------	------------------

This hereby certifies that the research paper entitled:

#### Research

Title of Project	Kusinasyon: An Android-based Filipino Cuisine Application with Voice Assisted Technology
------------------	--

written by the following developers:

#### Developers

##### Name (LN, FN M.I.)

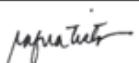
Delafer, John Fritz F.

Ponce, Neonie Quell A.

Rebustillo, Dean Clyte B.

Reyes, Rhey Franz A.

has been edited by the undersigned.

Proofreader		Signature	Date Signed
Name (FN M.I. LN)	Raquel Fuasan-Bautista		January 14, 2023

Attachment: Photocopy of any of the Proofreader's ID Cards, Proofreader's CV

## CURRICULUM VITAE

**RAQUEL FUASAN-BAUTISTA**  
2474 HMCL Building, Unit 511  
Juan Luna Street, Tondo, Manila  
[raquelaufasan@gmail.com](mailto:raquelaufasan@gmail.com)  
0926 - 063 - 0430



### PERSONAL INFORMATION

Date of Birth : March 25, 1984  
Place of Birth : Dumalag, Capiz  
Civil Status : Married  
Religion : Roman Catholic  
Language Spoken : English, Filipino, Ilonggo

### EDUCATIONAL BACKGROUND

**Graduate Studies** : The National Teachers College  
Master of Arts in Education  
Major in English  
2014 – Present

**Tertiary Level** : City College of Manila  
Bachelor of Secondary Education  
Major in English  
2001 – 2005

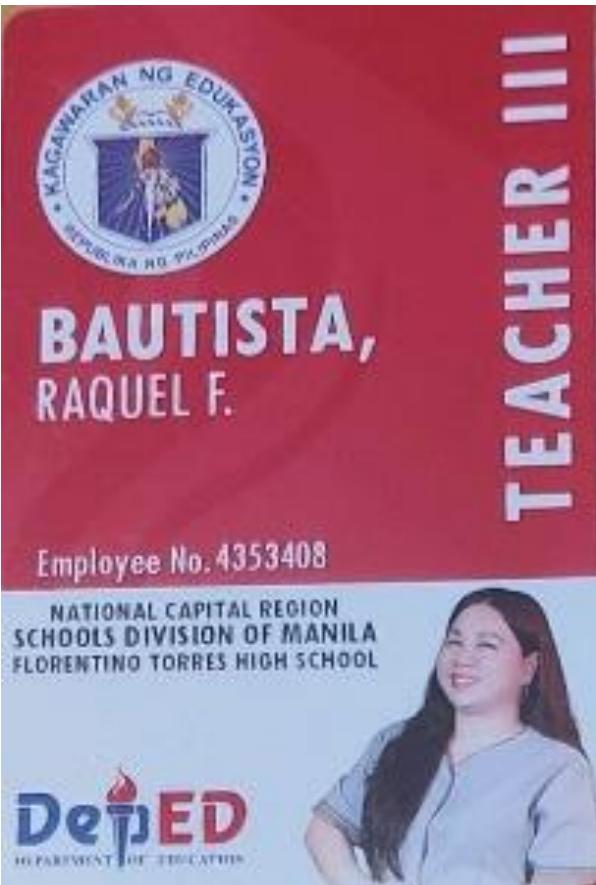
**Secondary Level** : Manuel A. Roxas High School  
1997 – 2001

**Elementary Level** : Abongan Elementary School  
1991 – 1997

### PROFESSIONAL EXPERIENCES

**Teacher III** Junior High School  
Florentino Torres High School  
March 19, 2019

**Academic Qualification** English Coordinator, Grade 10 level



Department of Education  
SCHOOLS DIVISION OF MANILA  
FLORENTINO TORRES HIGH SCHOOL  
2517 Juan Luna St., Gagalangin Tondo Manila

[raquelafuasan@gmail.com](mailto:raquelafuasan@gmail.com)

MA. PURA S. TALATTAD  
Principal IV

In case of emergency, please contact:

JOEL H. BAUTISTA 09684517740	
Other Information	
Home Address: 2474 Juan Luna St., Gagalangin, Tondo, Manila	
Blood Type:	O
TIN:	230-544-284
GSIS BP No.:	2003-485-093
Pag-Ibig No.	1050-0263-8362
Philhealth No.	19-090543975-1

Raquel F. Bautista  
Name and Signature of Employee

# FREDERICK PONCE

Technical Implementation Engineer



## Personal details

### Name

FREDERICK PONCE

### Email address

frederickponce@icloud.com

### Phone number

0906 221 6828

## Education

### BS Information Technology

Asian College, Aurora Blvd, Quezon City

2005 - 2009

## Employment

### Technical Implementation Engineer

Willis Towers Watson, Taguig

Jul 2021 - Present

- Wear multiple hats at times as a technical implementation engineer and as a custom report developer for various products.

- SQL development such as query optimization, writing user-defined stored procedures, views, and functions.

- Implement portal solution from the base product, and implement custom web modules, web services, and workflows using industry best practices.

- Involve in environment provisioning, customization, configuration, and deployment for various clients.

- Ensure requirements for changes and enhancements are properly defined and follow appropriate channels for testing and migrating to test and production environments.

- Follow documentation, process, and protocols during product implementations and identify areas of documentation improvement and helps improve it.

- Perform unit and comprehensive testing to ensure overall functionality and technical quality of deliverables.

- Develop comprehensive test plans, scripts, and cases, and participate in code reviews.

- Coaching and guiding colleagues in the progression and development of their technical skills.

### Custom Report Developer

Willis Towers Watson, Taguig City

Mar 2012 - Present

- Create custom applications that automate report development tasks.

- Serve as the primary code reviewer to ensure that all the reporting objects being created are standards-based.

- Regular interaction with managers, analysts, and developers across different countries.

- Serve as the technical lead of custom report developers based in Manila.

- Create multilingual reports using SSRS, DUNDAS, and IZENDA.

- Responsible for optimizing complex stored procedures, views, and user-defined functions to enable reports to accommodate a large volume of data.
- Make use of code reusability by writing custom codes within SSRS.
- Write and debug complex stored procedures, views, and user-defined functions.

**Senior Application Systems Engineer**

Jan 2011 - Feb 2012

Weserv Systems International, Inc., Taguig City

- Work with managers, analysts, and business applications professionals across the organization.

- Provide guidance and advice to less experienced developers.

- Develop SSRS reports that are automatically sent to specific recipients through email.

- Design SSRS reports based on clients' needs.

- Work on multiple projects simultaneously.

- Work on all phases of application systems analysis and programming activities.

- Regular interaction with Internal Support Teams, Account Teams, and External Clients.

**Business Intelligence Consultant**

Jul 2009 - Jan 2011

Workcentric Solutions Consulting Inc., Pasig City

- Write, code, test, and analyze software programs and applications. This includes researching, designing, and modifying software specifications throughout the production life cycle.

- Prepare systems and programming documentation for users and other programmers by internal and external standards.

**TOOLS**

SQL Server	PowerBI
SSRS	Dundas
SSIS	Azure
SSAS	Jira
VB.Net	XAML
C#.Net	
Powershell	
TFS	

## **APPENDIX C (RELEVANT SOURCE CODE)**

## RELEVANT SOURCE CODE

### Title: Java Code for Cuisine Recommendation Function

**Description:** The code below is the code written to fulfill the first objective of the project which is cuisine/recipe recommendation based on input ingredients.

```
package com.neoniequellponce.kusinasyon.activity;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;

import androidx.appcompat.app.ActionBar;
import androidx.appcompat.app.AppCompatActivity;
import androidx.recyclerview.widget.LinearLayoutManager;

import com.google.android.material.snackbar.BaseTransientBottomBar;
import com.google.android.material.snackbar.Snackbar;
import com.neoniequellponce.kusinasyon.R;
import com.neoniequellponce.kusinasyon.adapter.AdapterRecipe;
import com.neoniequellponce.kusinasyon.database.DbPublicRecipes;
import
com.neoniequellponce.kusinasyon.databinding.ActivityRecommendCuisineBinding;
import com.neoniequellponce.kusinasyon.model.ModelRecipe;

import java.util.ArrayList;
import java.util.List;

public class ActivityRecommendCuisine extends AppCompatActivity implements
    AdapterRecipe.OnRecipeClickListener {

    public static final String LIST = "list";

    private ArrayList<String> mIngredientList;
    private List<ModelRecipe> mMatchedCuisineList;

    private ActivityRecommendCuisineBinding mBinding;
    private AdapterRecipe mAdapter;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        mBinding = ActivityRecommendCuisineBinding.inflate(getLayoutInflater());
        setContentView(mBinding.getRoot());

        //Get final selection of ingredients
```

```

mIngredientList = getIntent().getStringArrayListExtra(LIST);
mMatchedCuisineList = new ArrayList<>();

setActionBar();
matchRecipes();
setRecyclerView();

if (!mMatchedCuisineList.isEmpty())
mBinding.tvIndicator.setVisibility(View.GONE);
}

private void setActionBar() {
    setSupportActionBar(mBinding.appBar.toolbar);

    ActionBar actionBar = getSupportActionBar();
    actionBar.setTitle("Recommended Cuisines");
    actionBar.setDisplayHomeAsUpEnabled(true);
    actionBar.setHomeAsUpIndicator(R.drawable.ic_arrow_back_ios_new);

    mBinding.appBar.toolbar.setNavigationOnClickListener(v -> onBackPressed());
}

private void matchRecipes() {
    DbPublicRecipes dbPublicRecipes = new DbPublicRecipes();
    dbPublicRecipes.getAllRecipes(new
DbPublicRecipes.OnPublicRecipesGetListener() {
    @Override
    public void getSystemRecipes(List<ModelRecipe> systemRecipeList) {
        matchRecipesHelper(systemRecipeList);
        setSnackbarMatchCuisine();
        setIndicatorVisibility();
    }

    @Override
    public void getUserRecipes(List<ModelRecipe> userRecipeList) {
        matchRecipesHelper(userRecipeList);
        setSnackbarMatchCuisine();
        setIndicatorVisibility();
    }
});

private void matchRecipesHelper(List<ModelRecipe> recipeList) {
    for (ModelRecipe recipe : recipeList) {
        int containsCtr = 0;

        for (String ingredient : mIngredientList) {
            //Get recipe that matches the input ingredients
            if (recipe.getKeyIngredients().contains(ingredient)) {
                //Increase matched recipe size
                containsCtr += 1;
            }
        }
    }
}

```

```

        if (containsCtr == recipe.getKeyIngredients().size()) {
            mMatchedCuisineList.add(recipe);
            if (mAdapter != null) mAdapter.notifyDataSetChanged();
        }
    }
}
}

private void setSnackbarMatchCuisine() {
    String msg = "Matched cuisines: " + mMatchedCuisineList.size();
    Snackbar snackbar = Snackbar.make(mBinding.layoutParent, msg,
    Snackbar.LENGTH_INDEFINITE);

    snackbar.setAnimationMode(BaseTransientBottomBar.ANIMATION_MODE_SLIDE);
    snackbar.setAction("DISMISS", v -> snackbar.dismiss());
    snackbar.show();
}

private void setIndicatorVisibility() {
    if (!mMatchedCuisineList.isEmpty()) {
        mBinding.progressIndicator.setVisibility(View.GONE);
        mBinding.tvIndicator.setVisibility(View.GONE);
    } else {
        mBinding.progressIndicator.setVisibility(View.GONE);
        mBinding.tvIndicator.setVisibility(View.VISIBLE);
    }
}

private void setRecyclerView() {
    LinearLayoutManager layoutManager = new LinearLayoutManager(this);
    mAdapter = new AdapterRecipe(this, mMatchedCuisineList, this);

    mBinding.recyclerView.setHasFixedSize(true);
    mBinding.recyclerView.setLayoutManager(layoutManager);
    mBinding.recyclerView.setAdapter(mAdapter);
}

@Override
public void onRecipeClick(int position) {
    ModelRecipe recipe = mMatchedCuisineList.get(position);

    Intent intent = new Intent(this, ActivityViewRecipe.class);
    intent.putExtra(ActivityViewRecipe.RECIPE, recipe);
    startActivity(intent);
}
}

```

**Title:** Java Code for Viewing Recipe with Voice Assisted Technology

**Description:** The code below is the code written to fulfill the third objective of the project which is having a voice assisted technology.

```
package com.neoniequellponce.kusinasyon.activity;

import android.content.Context;
import android.content.Intent;
import android.graphics.Color;
import android.media.AudioManager;
import android.media.MediaPlayer;
import android.os.Bundle;
import android.os.Handler;
import android.os.Looper;
import android.speech.RecognitionListener;
import android.speech.SpeechRecognizer;
import android.view.View;
import android.view.ViewGroup;
import android.view.Window;
import android.view.WindowManager;
import android.widget.Toast;

import androidx.annotation.Nullable;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.content.ContextCompat;
import androidx.core.view.ViewCompat;
import androidx.recyclerview.widget.RecyclerView;
import androidx.viewpager2.widget.ViewPager2;

import com.google.android.material.bottomsheet.BottomSheetBehavior;
import com.google.android.material.card.MaterialCardView;
import com.google.android.material.tabs.TabLayout;
import com.google.firebase.auth.FirebaseAuth;
import com.neoniequellponce.kusinasyon.R;
import com.neoniequellponce.kusinasyon.adapter.AdapterViewPager;
import com.neoniequellponce.kusinasyon.databinding.ActivityViewRecipeBinding;
import com.neoniequellponce.kusinasyon.dialogs.DialogManageRecipeAuthor;
import com.neoniequellponce.kusinasyon.dialogs.DialogResult;
import com.neoniequellponce.kusinasyon.dialogs.DialogSpeakNow;
import com.neoniequellponce.kusinasyon.fragment.FragmentRecipeDetails;
import com.neoniequellponce.kusinasyon.fragment.FragmentRecipeIngredients;
import com.neoniequellponce.kusinasyon.fragment.FragmentRecipeInstructions;
import com.neoniequellponce.kusinasyon.model.ModelRecipe;
```

```

import com.neoniequellponce.kusinasyon.service.AppService;
import com.neoniequellponce.kusinasyon.utility.UtilPermission;
import com.neoniequellponce.kusinasyon.voice.AzureSpeechService;
import com.neoniequellponce.kusinasyon.voice.PorcupineHandler;
import com.neoniequellponce.kusinasyon.voice.SpeechHandler;
import com.squareup.picasso.Picasso;

import java.util.ArrayList;
import java.util.HashMap;
import java.util.List;
import java.util.Objects;

import ai.picovoice.porcupine.PorcupineManagerCallback;

public class ActivityViewRecipe extends AppCompatActivity
    implements DialogManageRecipeAuthor.OnRecipeDeleteListener {

    //region Fields
    public static final String RECIPE = "recipe";
    public static final int RQ_EDIT_CONTENT = 1;
    private int mRecipeInstructionToken = 1;
    private long mBackPressedTime;
    private boolean mIsPorcupineOn = false;
    private boolean mIsCookingStarted = false;

    private ModelRecipe mRecipe;
    private HashMap<Integer, Object> mRecipeInstructMap;

    private PorcupineHandler mPorcupineHandler;
    private AzureSpeechService mAzureSpeech;
    private MediaPlayer mMediaPlayer;
    private SpeechHandler mSpeechHandler;
    private SpeechResultRunnable mSpeechResultRun;

    private ActivityViewRecipeBinding mBinding;
    private Handler mHandler;

    private DialogManageRecipeAuthor mDiaManage;
    private DialogResult mDiaResult;
    private DialogSpeakNow mDiaSpeakNow;
    private Toast mToastPressBackTwice;

    private AdapterViewPager mAdapter;
    private FragmentRecipeDetails mFragDetails;
    private FragmentRecipeIngredients mFragIngredients;
    private FragmentRecipeInstructions mFragInstructions;
}

```

```

//endregion

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    mBinding = ActivityViewRecipeBinding.inflate(getLayoutInflater());
    setContentView(mBinding.getRoot());

    mRecipe = getIntent().getParcelableExtra(RECIPE);
    mHandler = new Handler(Looper.getMainLooper());
    mDiaManage = new DialogManageRecipeAuthor(this, this, this);
    mDiaManage.setRecipe(mRecipe);

    //region Methods
    setFullScreen();
    setBottomSheet();
    checkIfUserIsAuthor();
    createFragments();
    setAdapter();
    setFragmentBundles();
    setRecipeInstructionMap();
    //endregion

    mBinding.fabBack.setOnClickListener(v -> finish());
    mBinding.fabAssistant.setOnClickListener(v -> fabAction());
    Picasso.with(this).load(mRecipe.getImageUrl()).into(mBinding.recipeImg);
    mBinding.btmSheet.tabLayout.addOnTabSelectedListener(tabSelection);
    mBinding.btmSheet.pager.setAdapter(mAdapter);
    mBinding.btmSheet.pager.registerOnPageChangeCallback(pageChange);
}

private void setFullScreen() {
    Window window = getWindow();

    window.clearFlags(WindowManager.LayoutParams.FLAG_TRANSLUCENT_STATUS);
    window.addFlags(WindowManager.LayoutParams.FLAG_DRAW_SYSTEM_BAR_BACKGROUNDS);

    window.getDecorView().setSystemUiVisibility(View.SYSTEM_UI_FLAG_LAYOUT_FULLSCREEN |
        View.SYSTEM_UI_FLAG_LIGHT_STATUS_BAR);
    window.setStatusBarColor(Color.TRANSPARENT);
}

```

```

        ViewCompat.setOnApplyWindowInsetsListener(mBinding.rlParent, (v,
insets) -> {
            ViewGroup.MarginLayoutParams fabBackParams = (ViewGroup.MarginLayoutParams)
mBinding.fabBack.getLayoutParams();

            ViewGroup.MarginLayoutParams fabAssistantParams = (ViewGroup.MarginLayoutParams)
mBinding.fabAssistant.getLayoutParams();

            fabBackParams.setMargins(0, insets.getSystemWindowInsetTop(), 0, 0);
            fabAssistantParams.setMargins(0, insets.getSystemWindowInsetTop(), 0,
0);

            mBinding.fabBack.setLayoutParams(fabBackParams);
            mBinding.fabAssistant.setLayoutParams(fabAssistantParams);
            insets.consumeSystemWindowInsets();
            return insets;
        });
    }

    private void setBottomSheet() {
        BottomSheetBehavior<MaterialCardView> sheetBehavior;
        sheetBehavior = BottomSheetBehavior.from(mBinding.btmSheet.cvContainer);
        sheetBehavior.setHideable(false);
        sheetBehavior.setState(BottomSheetBehavior.STATE_COLLAPSED);

        // Set bottom sheet minimum height
        int sheetBehaviourMinHeight = (int) (getResources()
                .getDisplayMetrics().heightPixels * 0.65d);

        sheetBehavior.setPeekHeight(sheetBehaviourMinHeight, true);

        mBinding.btmSheet.tvRecipeName.setText(mRecipe.getName());
        mBinding.btmSheet.btnAdd.setOnClickListener(v ->
mDiaManage.show());
    }
}

private void checkIfUserIsAuthor() {
    FirebaseAuth auth = FirebaseAuth.getInstance();
    if (!Objects.equals(auth.getUid(), mRecipe.getAuthorUid())) {
        mBinding.btmSheet.btnAdd.setVisibility(View.GONE);
    }
}

```

```

@Override
protected void onActivityResult(int requestCode, int resultCode, @Nullable
Intent data) {
    super.onActivityResult(requestCode, resultCode, data);
    if (requestCode == RQ_EDIT_CONTENT) {
        if (resultCode == RESULT_OK) finish();
    }
}

private void createFragments() {
    mFragDetails = new FragmentRecipeDetails();
    mFragIngredients = new FragmentRecipeIngredients();
    mFragInstructions = new FragmentRecipeInstructions();
}

private void setAdapter() {
    mAdapter = new AdapterViewPager(getSupportFragmentManager(),
getLifecycle());
    mAdapter.addFragment(mFragDetails);
    mAdapter.addFragment(mFragIngredients);
    mAdapter.addFragment(mFragInstructions);
}

private void setFragmentBundles() {
    Bundle bdDetails = new Bundle();
    bdDetails.putString(FragmentRecipeDetails.DESCRIPTION,
mRecipe.getDescription());
    bdDetails.putString(FragmentRecipeDetails.AUTHOR_UID,
mRecipe.getAuthorUid());
    bdDetails.putString(FragmentRecipeDetails.AUTHOR,
mRecipe.getAuthor());
    bdDetails.putString(FragmentRecipeDetails.ORIGIN, mRecipe.getOrigin());
    bdDetails.putString(FragmentRecipeDetails.VISIBILITY,
mRecipe.getVisibility());

    Bundle bdIngr = new Bundle();
    bdIngr.putStringArrayList(FragmentRecipeIngredients.INGREDIENTS,
(ArrayList<String>) mRecipe.getIngredients());

    Bundle bdInstruct = new Bundle();

    bdInstruct.putStringArrayList(FragmentRecipeInstructions.INSTRUCTIONS,
(ArrayList<String>) mRecipe.getInstructions());

    mFragDetails.setArguments(bdDetails);
    mFragIngredients.setArguments(bdIngr);
}

```

```

        mFragInstructions.setArguments(bdInstruct);
    }

private void setRecipeInstructionMap() {
    //Put recipe instructions in hashmap for assistant
    mRecipeInstructMap = new HashMap<>();
    StringBuilder builder = new StringBuilder();

    for (int i = 0; i < mRecipe.getInstructions().size(); i++) {
        builder.delete(0, builder.length());
        builder.append(mRecipe.getInstructions().get(i));
        mRecipeInstructMap.put(i + 1, String.valueOf(builder));
    }
}

@Override
protected void onPause() {
    super.onPause();
    if (mIsPorcupineOn) stopAssistantResources();
}

private void stopAssistantResources() {
    mPorcupineHandler.stopPorcupine();
    mAzureSpeech.stopSynthesizing();
    mAzureSpeech.release();
}

private void fabAction() {
    if (!mIsPorcupineOn) {
        try {
            if (UtilPermission.hasRecordPermission(getApplicationContext())) {
                porcupineOn();
            } else UtilPermission.requestRecordPermission(this);
        } catch (Exception e) {
            Toast.makeText(this, e.getMessage(),
                        Toast.LENGTH_SHORT).show();
            e.printStackTrace();
        }
    } else porcupineOff();
}

private void porcupineOn() {
    mMediaPlayer = MediaPlayer.create(this,
        R.raw.sound_picovoice_wake_word_detect);

    //Set fab icon to active
}

```

```

        mBinding.fabAssistant.setImageResource(R.drawable.ic_mic);
        mBinding.fabAssistant.setColorFilter(ContextCompat.getColor(this,
            android.R.color.holo_green_dark));

        mDiaSpeakNow = new DialogSpeakNow(this);

        AppService.sPool.execute(() -> {
            //Set volume to max
            AudioManager           audioManager           =           (AudioManager)
getSystemService(Context.AUDIO_SERVICE);
            int                   volume                   =
audioManager.getStreamMaxVolume(AudioManager.STREAM_MUSIC);
            audioManager.setStreamVolume(AudioManager.STREAM_MUSIC,
volume,
                AudioManager.FLAG_SHOW_UI);

        mSpeechResultRun = new SpeechResultRunnable();

        runOnUiThread(() -> mSpeechHandler = new SpeechHandler(this,
speechRecognition));

        mPorcupineHandler = new PorcupineHandler(this);
        mPorcupineHandler.setPorcupineManagerCallback(porcupineCallback);

        mAzureSpeech = new AzureSpeechService();

        mPorcupineHandler.startPorcupine();
        mIsPorcupineOn = true;
    });
}

private void porcupineOff() {
    AppService.sPool.execute(() -> {
        stopAssistantResources();
        mIsPorcupineOn = false;
    });

    //Set fab icon to inactive
    mBinding.fabAssistant.setImageResource(R.drawable.ic_mic_off);
    mBinding.fabAssistant.setColorFilter(ContextCompat.getColor(this,
R.color.sonic_silver_600));
}

@Override
public void onRecipeDelete() {
    mDiaResult = new DialogResult(this, () -> {

```

```

        mDiaResult.dismiss();
        finish();
    });

    String message = "The recipe named " + mRecipe.getName() + " is now
deleted.";
    mDiaResult.setContent("Recipe Deleted", message);

    mHandler.postDelayed(() -> mDiaResult.show(), 200);
}

@Override
public void onBackPressed() {
    if (mIsPorcupineOn) {
        if (mBackPressedTime + 2000 > System.currentTimeMillis()) {
            mToastPressBackTwice.cancel();
            super.onBackPressed();
            return;
        } else {
            mToastPressBackTwice = Toast.makeText(this,
getString(R.string.press_back_twice),
Toast.LENGTH_SHORT);
            mToastPressBackTwice.show();
        }
        mBackPressedTime = System.currentTimeMillis();
    } else super.onBackPressed();
}

private TabLayout.OnTabSelectedListener tabSelection = new
TabLayout.OnTabSelectedListener() {
    @Override
    public void onTabSelected(TabLayout.Tab tab) {
        mBinding.btmSheet.pager.setCurrentItem(tab.getPosition());
    }

    @Override
    public void onTabUnselected(TabLayout.Tab tab) {

    }

    @Override
    public void onTabReselected(TabLayout.Tab tab) {

    }
};

```

```

private ViewPager2.OnPageChangeCallback pageChange = new
    ViewPager2.OnPageChangeCallback() {
        @Override
        public void onPageSelected(int position) {
            super.onPageSelected(position);
            mBinding.btmSheet.tabLayout.getTabAt(position).select();

            // Remove overscroll on horizontal swipe in viewpager
            View child = mBinding.btmSheet.pager.getChildAt(position);
            if (child instanceof RecyclerView) {
                child.setOverScrollMode(View.OVER_SCROLL_NEVER);
            }
        }
    };

private PorcupineManagerCallback porcupineCallback = keywordIndex -> {
    mSpeechHandler.startListening();
    mMediaPlayer.start();
};

private RecognitionListener speechRecognition = new RecognitionListener() {
    @Override
    public void onReadyForSpeech(Bundle params) {
        mDiaSpeakNow.show();
    }

    @Override
    public void onBeginningOfSpeech() {

    }

    @Override
    public void onRmsChanged(float rmsdB) {

    }

    @Override
    public void onBufferReceived(byte[] buffer) {

    }

    @Override
    public void onEndOfSpeech() {
        mDiaSpeakNow.dismiss();
    }
}

```

```

@Override
public void onError(int error) {
    mDiaSpeakNow.dismiss();
}

@Override
public void onResults(Bundle results) {
    List<String> resultList;
    resultList
    results.getStringArrayList(SpeechRecognizer.RESULTS_RECOGNITION);

    if (!resultList.isEmpty()) {
        mSpeechResultRun.setInput(resultList.get(0).toLowerCase().trim());
        AppService.sPool.execute(mSpeechResultRun);
    }
}

@Override
public void onPartialResults(Bundle partialResults) {

}

@Override
public void onEvent(int eventType, Bundle params) {

};

private class SpeechResultRunnable implements Runnable {

    private StringBuilder mInput;
    private StringBuilder mOutput;

    public SpeechResultRunnable() {
        mInput = new StringBuilder();
        mOutput = new StringBuilder();
    }

    public void setInput(String input) {
        if (mInput.length() > 0) mInput.delete(0, mInput.length());
        mInput.append(input);
    }

    @Override
    public void run() {
        //Clear string builder content

```

```

if (mOutput.length() > 0) mOutput.delete(0, mOutput.length());

/*Case 1: Check if input speech contains "GO TO STEP" phrase
 *Case 2: Perform other commands*/
if (String.valueOf(mInput).contains("go to step")) {
    /*Get instruction number based on index
     *"GO TO STEP (N)"*/
    int key = Integer.parseInt(String.valueOf(mInput).substring(11));

    //Check if input instruction number is valid
    if (mRecipeInstructMap.containsKey(key)) {
        mOutput.append("Step").append(" ").append(key).append(", ")
            .append(mRecipeInstructMap.get(key));
    } else mOutput.append("There is not step ").append(key);

    mAzureSpeech.speak(String.valueOf(mOutput));
} else {
    AudioManager audioManager = (AudioManager)
getSystemService(Context.AUDIO_SERVICE);
    int volume;

    try {
        String command = String.valueOf(mInput).toLowerCase();

        switch (command) {
            case "start cooking":
                if (mIsCookingStarted) {
                    mOutput.append("You already started cooking.");
                    mAzureSpeech.speak(String.valueOf(mOutput));
                } else {
                    speakInstruction();
                    mIsCookingStarted = true;
                }
                break;
            case "what's next":
                if (!mIsCookingStarted) {
                    mOutput.append("You are already on the first step.");
                    mAzureSpeech.speak(String.valueOf(mOutput));
                } else {
                    mRecipeInstructionToken++;
                    speakInstruction();
                }
                break;
            case "go back":
                if (!mIsCookingStarted) {
                    mOutput.append("You did not start cooking yet.");

```

```

        mAzureSpeech.speak(String.valueOf(mOutput));
    } else if (mRecipeInstructionToken == 1) {
        mOutput.append("You are already on the first step.");
        mAzureSpeech.speak(String.valueOf(mOutput));
    } else {
        --mRecipeInstructionToken;
        speakInstruction();
    }
    break;
case "repeat current instruction":
if (!mIsCookingStarted) {
    mOutput.append("You did not start cooking yet.");
    mAzureSpeech.speak(String.valueOf(mOutput));
} else speakInstruction();
break;
case "volume up":
    volume =
audioManager.getStreamVolume(AudioManager.STREAM_MUSIC) + 1;

audioManager.setStreamVolume(AudioManager.STREAM_MUSIC, volume,
    AudioManager.FLAG_SHOW_UI);
break;
case "volume half":
    int fullVolume =
audioManager.getStreamMaxVolume(AudioManager.STREAM_MUSIC);
    volume = fullVolume / 2;

audioManager.setStreamVolume(AudioManager.STREAM_MUSIC, volume,
    AudioManager.FLAG_SHOW_UI);
break;
case "volume max":
    volume =
audioManager.getStreamMaxVolume(AudioManager.STREAM_MUSIC);

audioManager.setStreamVolume(AudioManager.STREAM_MUSIC, volume,
    AudioManager.FLAG_SHOW_UI);
break;
}
} catch (Exception e) {
    e.printStackTrace();
    mOutput.append("I didn't understand.");
    mAzureSpeech.speak(String.valueOf(mOutput));
}
}
}

```

```

        private void speakInstruction() {
            mOutput.append("Step").append(" ")
                .append(mRecipeInstructionToken).append(", ")
                .append(mRecipeInstructMap.get(mRecipeInstructionToken));

            mAzureSpeech.speak(String.valueOf(mOutput));
        }
    }
}

```

**Title:** Java Code for Sharing Private Recipe

**Description:** The code written below is the code for sharing a private recipe inside a private circle.

```

package com.neoniequellponce.kusinasyon.activity;

import android.content.ContentResolver;
import android.net.Uri;
import android.os.Bundle;
import android.os.Handler;
import android.os.Looper;
import android.view.Menu;
import android.view.MenuInflater;
import android.view.MenuItem;
import android.webkit.MimeTypeMap;
import android.widget.Toast;

import androidx.annotation.NonNull;
import androidx.appcompat.app.ActionBar;
import androidx.appcompat.app.AppCompatActivity;

import com.google.firebase.storage.FirebaseStorage;
import com.google.firebase.storage.StorageReference;
import com.neoniequellponce.kusinasyon.R;
import com.neoniequellponce.kusinasyon.adapter.AdapterViewPager;
import com.neoniequellponce.kusinasyon.database.DbCircle;
import com.neoniequellponce.kusinasyon.database.DbUsers;
import com.neoniequellponce.kusinasyon.databinding.ActivitySetRecipeBinding;
import com.neoniequellponce.kusinasyon.dialogs.DialogLoading;
import com.neoniequellponce.kusinasyon.dialogs.DialogResult;
import com.neoniequellponce.kusinasyon.dialogs.DialogVisibility;
import
com.neoniequellponce.kusinasyon.fragment.FragmentSetRecipeDescription;
import com.neoniequellponce.kusinasyon.fragment.FragmentSetRecipeImage;

```

```

import
com.neoniequellponce.kusinasyon.fragment.FragmentSetRecipeIngredients;
import
com.neoniequellponce.kusinasyon.fragment.FragmentSetRecipeInstructions;
import
com.neoniequellponce.kusinasyon.fragment.FragmentSetRecipeNameAndOrigin;
import com.neoniequellponce.kusinasyon.model.ModelRecipe;

public class ActivityShareToCircle extends AppCompatActivity {

    //region Fields
    public static final String KEY = "key";
    public static final String NAME = "name";
    private String mCircleKey;
    private String mCircleName;
    private long mBackPressedTime;
    private int mCurrentPosition = 0;

    private DbCircle mDbCircle;
    private ModelRecipe mRecipe;

    private ActivitySetRecipeBinding mBinding;
    private Handler mHandler;

    private Toast mToast;
    private DialogVisibility mDiaVisibility;
    private DialogLoading mDiaLoading;
    private DialogResult mDiaResult;

    private AdapterViewPager mAdapter;
    private FragmentSetRecipeNameAndOrigin mFragNameOrigin;
    private FragmentSetRecipeDescription mFragDesc;
    private FragmentSetRecipeIngredients mFragIngredients;
    private FragmentSetRecipeInstructions mFragInstructions;
    private FragmentSetRecipeImage mFragImage;
    //endregion

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        mBinding = ActivitySetRecipeBinding.inflate(getLayoutInflater());
        setContentView(mBinding.getRoot());

        mCircleKey = getIntent().getStringExtra(KEY);
        mCircleName = getIntent().getStringExtra(NAME);
    }
}

```

```

        mDbCircle = new DbCircle();
        mRecipe = new ModelRecipe();
        mHandler = new Handler(Looper.getMainLooper());

        setActionBar();
        createFragments();
        setAdapter();
        setPagerProperty();
        setFragmentBundles();
        setDialogs();
        setRecipeDefaultValues();

        mBinding.btnExit.setOnClickListener(v -> backStep());
        mBinding.btnExit.setOnClickListener(v -> nextStep());
    }

    private void setActionBar() {
        setSupportActionBar(mBinding.appBar.toolbar);

        ActionBar actionBar = getSupportActionBar();
        actionBar.setTitle("Share to " + mCircleName);
        actionBar.setDisplayHomeAsUpEnabled(true);
        actionBar.setHomeAsUpIndicator(R.drawable.ic_arrow_back_ios_new);

        mBinding.appBar.toolbar.setNavigationOnClickListener(v ->
super.onBackPressed());
    }

    private void createFragments() {
        mFragNameOrigin = new FragmentSetRecipeNameAndOrigin();
        mFragDesc = new FragmentSetRecipeDescription();
        mFragIngredients = new FragmentSetRecipeIngredients();
        mFragInstructions = new FragmentSetRecipeInstructions();
        mFragImage = new FragmentSetRecipeImage();
    }

    private void setAdapter() {
        mAdapter = new AdapterViewPager(getSupportFragmentManager(),
getLifecycle());
        mAdapter.addFragment(mFragNameOrigin);
        mAdapter.addFragment(mFragDesc);
        mAdapter.addFragment(mFragIngredients);
        mAdapter.addFragment(mFragInstructions);
        mAdapter.addFragment(mFragImage);
    }
}

```

```

private void setPagerProperty() {
    mBinding.pager.setUserInputEnabled(false);
    mBinding.pager.setAdapter(mAdapter);
    mBinding.pager.setCurrentItem(mCurrentPosition, false);
}

private void setFragmentBundles() {
    //Set step indicators on each fragment
    String[] stepsArr = new String[mAdapter.getItemCount()];
    StringBuilder builder = new StringBuilder();

    for (int i = 1; i < stepsArr.length + 1; i++) {
        builder.delete(0, builder.length());
        builder.append("Step " + i).append(" of " + mAdapter.getItemCount());
        stepsArr[i - 1] = String.valueOf(builder);
    }

    //Pass value to FragmentSetNameAndOrigin
    Bundle bdNameAndOrigin = new Bundle();
    bdNameAndOrigin.putString(FragmentSetRecipeNameAndOrigin.STEP,
    stepsArr[0]);

    bdNameAndOrigin.putBoolean(FragmentSetRecipeNameAndOrigin.FOR_EDIT,
    false);

    //Pass value to FragmentsetDescription
    Bundle bdDesc = new Bundle();
    bdDesc.putString(FragmentSetRecipeDescription.STEP, stepsArr[1]);
    bdDesc.putBoolean(FragmentSetRecipeDescription.FOR_EDIT, false);

    //Pass value to FragmentSetIngredients
    Bundle bdIngr = new Bundle();
    bdIngr.putString(FragmentSetRecipeIngredients.STEP, stepsArr[2]);
    bdIngr.putBoolean(FragmentSetRecipeIngredients.FOR_EDIT, false);

    //Pass value to FragmentInstructions
    Bundle bdInstruct = new Bundle();
    bdInstruct.putString(FragmentSetRecipeInstructions.STEP, stepsArr[3]);
    bdInstruct.putBoolean(FragmentSetRecipeInstructions.FOR_EDIT, false);

    //Pass value to FragmentSetRecipeImage
    Bundle bdImage = new Bundle();
    bdImage.putString(FragmentSetRecipeImage.STEP, stepsArr[4]);
    bdImage.putBoolean(FragmentSetRecipeImage.FOR_EDIT, false);
}

```

```

        mFragNameOrigin.setArguments(bdNameAndOrigin);
        mFragDesc.setArguments(bdDesc);
        mFragIngredients.setArguments(bdIngrid);
        mFragInstructions.setArguments(bdInstruct);
        mFragImage.setArguments(bdImage);
    }

    private void setDialogs() {
        //Dialog visibility
        mDiaVisibility = new DialogVisibility(this, visibility ->
            mRecipe.setVisibility(visibility), () -> mDiaVisibility.dismiss());

        //Dialog loading
        mDiaLoading = new DialogLoading(this);

        //Dialog result
        mDiaResult = new DialogResult(this, () -> {
            mDiaResult.dismiss();
            finish();
        });
        mDiaResult.setContent("Recipe Shared",
            "Thanks for sharing!\nYour recipe is now at your circle.");
    }

    private void setRecipeDefaultValues() {
        //Set recipe author and authorUid
        DbUsers dbUsers = new DbUsers();
        dbUsers.getUserInfo(user -> {
            mRecipe.setAuthor(user.getDisplayName());
            mRecipe.setAuthorUid(user.getId());
        });
        //Set recipe visibility by default
        mRecipe.setVisibility("visible");
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        MenuInflater menuInflater = getMenuInflater();
        menuInflater.inflate(R.menu.menu_act_set_recipe_toolbar, menu);

        //Hide key ingredient info
        MenuItem menuKeyIngredInfo = menu.findItem(R.id
            .menu_act_set_recipe_key_ingred_info);

        menuKeyIngredInfo.setVisible(false);
    }
}

```

```

        return super.onCreateOptionsMenu(menu);
    }

    @Override
    public boolean onOptionsItemSelected(@NonNull MenuItem item) {
        String title = String.valueOf(item.getTitle()).toLowerCase();
        if (title.equals("visibility")) mDiaVisibility.show();
        return super.onOptionsItemSelected(item);
    }

    private void backStep() {
        if (mCurrentPosition != 0) mCurrentPosition -= 1;

        //If button text is "submit", revert to "next"
        if (mCurrentPosition != mAdapter.getItemCount() - 1)
            mBinding.btnExit.setText("Next");

        mBinding.pager.setCurrentItem(mCurrentPosition, true);
    }

    private void nextStep() {
        if (validToProceed()) {
            String btnText = String.valueOf(mBinding.btnExit.getText());

            //On finish click
            if (btnText.equalsIgnoreCase("Finish")) shareRecipe();

            //Change button text from "next" to "finish" when end is reached
            if (mCurrentPosition == mAdapter.getItemCount() - 2)
                mBinding.btnExit.setText("Finish");

            //Go to next fragment
            if (mCurrentPosition != mAdapter.getItemCount() - 1) mCurrentPosition += 1;

            mBinding.pager.setCurrentItem(mCurrentPosition, true);
        }
    }

    private boolean validToProceed() {
        //Check if inputs are not empty
        switch (mCurrentPosition) {
            case 0:
                if (mFragNameOrigin.getName().equals("")) {
                    mFragNameOrigin.requestFocusOnName();

```

```

        Toast.makeText(this, getString(R.string.name_cannot_be_empty),
                      Toast.LENGTH_SHORT).show();
        return false;
    } else return true;
case 1:
    if (mFragDesc.getDescription().equals("")) {
        mFragDesc.requestFocusOnDescription();
        Toast.makeText(this,
                      getString(R.string.description_cannot_be_empty),
                      Toast.LENGTH_SHORT).show();
        return false;
    } else return true;
case 2:
    if (mFragIngredients.getIngredientList().isEmpty()) {
        mFragIngredients.requestFocusOnIngredients();
        Toast.makeText(this,
                      getString(R.string.ingredients_cannot_be_empty),
                      Toast.LENGTH_SHORT).show();
        return false;
    } else return true;
case 3:
    if (mFragInstructions.getInstructionList().isEmpty()) {
        mFragInstructions.requestFocusOnInstructions();
        Toast.makeText(this,
                      getString(R.string.instructions_cannot_be_empty),
                      Toast.LENGTH_SHORT).show();
        return false;
    } else return true;
case 4:
    if (mFragImage.getImageUri() == null) {
        Toast.makeText(this, getString(R.string.image_cannot_be_empty),
                      Toast.LENGTH_SHORT).show();
        return false;
    } else return true;
}
return false;
}

private void shareRecipe() {
    mDiaLoading.show();

    Uri imgUri = mFragImage.getImageUri();

    StorageReference storageRef
    FirebaseStorage.getInstance().getReference("userRecipesImg");
    StorageReference fileRef = storageRef.child(System.currentTimeMillis() +

```

```

        "." + getFileExtension(imgUri));

    //Upload image to Firebase Storage
    fileRef.putFile(imgUri)
        .addOnSuccessListener(taskSnapshot -> {
            //Get image download url
            fileRef.getDownloadUrl()
                .addOnSuccessListener(uri -> {
                    //Set recipe details
                    setRecipeDetails(String.valueOf(uri));

                    //Add recipe to circle
                    mDbCircle.shareRecipe(mRecipe, mCircleKey, shared -> {
                        mDiaLoading.dismiss();

                        if (shared) mHandler.postDelayed(() -> mDiaResult.show(),
                            200);
                        else {
                            Toast.makeText(ActivityShareToCircle.this,
                                getString(R.string.failed_to_share_recipe),
                                Toast.LENGTH_SHORT).show();
                        }
                    });
                })
        .addOnFailureListener(e -> {
            mDiaLoading.dismiss();
            e.printStackTrace();
        });
    })
    .addOnFailureListener(e -> {
        mDiaLoading.dismiss();
        e.printStackTrace();
    });
}

private String getFileExtension(Uri uri) {
    ContentResolver cR = getContentResolver();
    MimeMimeTypeMap mime = MimeMimeTypeMap.getSingleton();
    return mime.getExtensionFromMimeType(cR.getType(uri));
}

private void setRecipeDetails(String imageUrl) {
    mRecipe.setKey(mDbCircle.getKey());
    mRecipe.setName(mFragNameOrigin.getName());
    mRecipe.setDescription(mFragDesc.getDescription());
    mRecipe.setImageUrl(imageUrl);
}

```

```

        mRecipe.setOrigin(mFragNameOrigin.getOrigin().toLowerCase());

        mRecipe.setPrivacy("private");
        mRecipe.setInstructions(mFragInstructions.getInstructionList());
        mRecipe.setIngredients(mFragIngredients.getIngredientList());

        mRecipe.setCircleKey(mCircleKey);
    }

    @Override
    public void onBackPressed() {
        if (mBackPressedTime + 2000 > System.currentTimeMillis()) {
            mToast.cancel();
            super.onBackPressed();
            return;
        } else {
            mToast = Toast.makeText(this, getString(R.string.press_back_twice),
                    Toast.LENGTH_SHORT);
            mToast.show();
        }
        mBackPressedTime = System.currentTimeMillis();
    }
}

```

**Title:** Java Code for Sharing Public Recipe

**Description:** The code written below is the code use for sharing a public recipe.

```
package com.neoniequellponce.kusinasyon.activity;
```

```

import android.app.Dialog;
import android.content.ContentResolver;
import android.graphics.Color;
import android.graphics.drawable.ColorDrawable;
import android.net.Uri;
import android.os.Bundle;
import android.os.Handler;
import android.os.Looper;
import android.view.Menu;
import android.view.MenuInflater;
import android.view.MenuItem;
import android.webkit.MimeTypeMap;
import android.widget.Toast;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;

```

```

import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseUser;
import com.google.firebase.storage.FirebaseStorage;
import com.google.firebase.storage.StorageReference;
import com.neoniequellponce.kusinasyon.R;
import com.neoniequellponce.kusinasyon.adapter.AdapterViewPager;
import com.neoniequellponce.kusinasyon.database.DbCircle;
import com.neoniequellponce.kusinasyon.database.DbUsers;
import com.neoniequellponce.kusinasyon.databinding.ActivitySetRecipeBinding;
import
com.neoniequellponce.kusinasyon.databinding.DialogKeyIngredientsBinding;
import com.neoniequellponce.kusinasyon.dialogs.DialogLoading;
import com.neoniequellponce.kusinasyon.dialogs.DialogResult;
import com.neoniequellponce.kusinasyon.dialogs.DialogVisibility;
import
com.neoniequellponce.kusinasyon.fragment.FragmentSetRecipeDescription;
import com.neoniequellponce.kusinasyon.fragment.FragmentSetRecipeImage;
import
com.neoniequellponce.kusinasyon.fragment.FragmentSetRecipeIngredients;
import
com.neoniequellponce.kusinasyon.fragment.FragmentSetRecipeInstructions;
import
com.neoniequellponce.kusinasyon.fragment.FragmentSetRecipeKeyIngredients;
import
com.neoniequellponce.kusinasyon.fragment.FragmentSetRecipeNameAndOrigin;
import com.neoniequellponce.kusinasyon.model.ModelRecipe;
import com.neoniequellponce.kusinasyon.model.ModelUser;

import java.util.ArrayList;
import java.util.List;
import java.util.Random;

public class ActivityShareToPublic extends AppCompatActivity {

    //region Fields
    public static final String CIRCLE_KEY = "key";
    private String mCircleKey;
    private long mBackPressedTime;
    private int mCurrentPosition = 0;

    private DbUsers mDbUsers;
    private FirebaseUser mUser;

    private ModelRecipe mRecipe;
}

```

```

private ActivitySetRecipeBinding mActBinding;
private DialogKeyIngredientsBinding mDiaKeyIngrdBinding;
private Handler mHandler;

private Toast mToast;
private MenuItem mMenuItemKeyIngrdInfo;
private DialogVisibility mDialogVisibility;
private DialogLoading mDialogLoading;
private DialogResult mDialogResult;
private Dialog mDialogKeyIngrd;

private AdapterViewPager mAdapter;
private FragmentSetRecipeNameAndOrigin mFragNameOrigin;
private FragmentSetRecipeDescription mFragDesc;
private FragmentSetRecipeIngredients mFragIngredients;
private FragmentSetRecipeKeyIngredients mFragKeyIngredients;
private FragmentSetRecipeInstructions mFragInstructions;
private FragmentSetRecipeImage mFragImage;
//endregion

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    mActBinding = ActivitySetRecipeBinding.inflate(getLayoutInflater());
    setContentView(mActBinding.getRoot());

    mCircleKey = getIntent().getStringExtra(CIRCLE_KEY);

    mDbUsers = new DbUsers();
    mRecipe = new ModelRecipe();
    mHandler = new Handler(Looper.getMainLooper());

    mUser = FirebaseAuth.getInstance().getCurrentUser();
    assert mUser != null;

    setActionBar();
    createFragments();
    setAdapter();
    setPagerProperty();
    setFragmentBundles();
    setDialogs();
    setRecipeDefaultValues();

    mActBinding.btnExit.setClickListener(v -> backStep());
    mActBinding.btnExit.setClickListener(v -> nextStep());
}

```

```

private void setActionBar() {
    setSupportActionBar(mActBinding.appBar.toolbar);

    androidx.appcompat.app.ActionBar actionBar = getSupportActionBar();
    actionBar.setTitle("Share to Public");
    actionBar.setDisplayHomeAsUpEnabled(true);
    actionBar.setHomeAsUpIndicator(R.drawable.ic_arrow_back_ios_new);

    mActBinding.appBar.toolbar.setNavigationOnClickListener(v ->
super.onBackPressed());
}

private void createFragments() {
    mFragNameOrigin = new FragmentSetRecipeNameAndOrigin();
    mFragDesc = new FragmentSetRecipeDescription();
    mFragIngredients = new FragmentSetRecipeIngredients();
    mFragKeyIngredients = new FragmentSetRecipeKeyIngredients();
    mFragInstructions = new FragmentSetRecipeInstructions();
    mFragImage = new FragmentSetRecipeImage();
}

private void setAdapter() {
    mAdapter = new AdapterViewPager(getSupportFragmentManager(),
getLifecycle());
    mAdapter.addFragment(mFragNameOrigin);
    mAdapter.addFragment(mFragDesc);
    mAdapter.addFragment(mFragIngredients);
    mAdapter.addFragment(mFragKeyIngredients);
    mAdapter.addFragment(mFragInstructions);
    mAdapter.addFragment(mFragImage);
}

private void setPagerProperty() {
    mActBinding.pager.setUserInputEnabled(false);
    mActBinding.pager.setAdapter(mAdapter);
    mActBinding.pager.setCurrentItem(mCurrentPosition, false);
}

private void setFragmentBundles() {
    //Set step indicators on each fragment
    String[] stepsArr = new String[mAdapter.getItemCount()];
    StringBuilder builder = new StringBuilder();

    for (int i = 1; i < stepsArr.length + 1; i++) {
        builder.delete(0, builder.length());
}

```

```

        builder.append("Step           ").append(i).append("          of
").append(mAdapter.getItemCount());

    stepsArr[i - 1] = String.valueOf(builder);
}

//Pass value to FragmentSetNameAndOrigin
Bundle bdNameAndOrigin = new Bundle();
bdNameAndOrigin.putString(FragmentSetRecipeNameAndOrigin.STEP,
stepsArr[0]);

bdNameAndOrigin.putBoolean(FragmentSetRecipeNameAndOrigin.FOR_EDIT,
false);

//Pass value to FragmentSetDescription
Bundle bdDesc = new Bundle();
bdDesc.putString(FragmentSetRecipeDescription.STEP, stepsArr[1]);
bdDesc.putBoolean(FragmentSetRecipeDescription.FOR_EDIT, false);

//Pass value to FragmentSetIngredients
Bundle bdIngred = new Bundle();
bdIngred.putString(FragmentSetRecipeIngredients.STEP, stepsArr[2]);
bdIngred.putBoolean(FragmentSetRecipeIngredients.FOR_EDIT, false);

//Pass value to FragmentSetKeyIngredients
Bundle bdKeyIngred = new Bundle();
bdKeyIngred.putString(FragmentSetRecipeKeyIngredients.STEP,
stepsArr[3]);
bdKeyIngred.putBoolean(FragmentSetRecipeKeyIngredients.FOR_EDIT,
false);

//Pass value to FragmentInstructions
Bundle bdInstruct = new Bundle();
bdInstruct.putString(FragmentSetRecipeInstructions.STEP, stepsArr[4]);
bdInstruct.putBoolean(FragmentSetRecipeInstructions.FOR_EDIT, false);

//Pass value to FragmentSetRecipeImage
Bundle bdImage = new Bundle();
bdImage.putString(FragmentSetRecipeImage.STEP, stepsArr[5]);
bdImage.putBoolean(FragmentSetRecipeImage.FOR_EDIT, false);

mFragNameOrigin.setArguments(bdNameAndOrigin);
mFragDesc.setArguments(bdDesc);
mFragIngredients.setArguments(bdIngred);
mFragKeyIngredients.setArguments(bdKeyIngred);
mFragInstructions.setArguments(bdInstruct);

```

```

        mFragImage.setArguments(bdImage);
    }

    private void setDialogs() {
        //Dialog visibility
        mDiaVisibility = new DialogVisibility(this, visibility ->
            mRecipe.setVisibility(visibility), () -> mDiaVisibility.dismiss());

        setDiaKeyIngrid();

        //Dialog loading
        mDiaLoading = new DialogLoading(this);

        //Dialog result
        mDiaResult = new DialogResult(this, () -> {
            mDiaResult.dismiss();
            finish();
        });

        String message = "Thanks for sharing!\nYour recipe will " +
            "be reviewed by other members for approval.";

        mDiaResult.setContent("For Approval", message);
    }

    private void setDiaKeyIngrid() {
        mDiaKeyIngridBinding =
        DialogKeyIngredientsBinding.inflate(getApplicationContext());

        mDiaKeyIngrid = new Dialog(this);
        mDiaKeyIngrid.setContentView(mDiaKeyIngridBinding.getRoot());
        mDiaKeyIngrid.getWindow().setBackgroundDrawable(new
        ColorDrawable(Color.TRANSPARENT));

        mDiaKeyIngrid.getWindow().setWindowAnimations(R.style.CustomDialogAni
        mation);

        mDiaKeyIngridBinding.btnClose.setOnClickListener(v ->
        mDiaKeyIngrid.dismiss());
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        MenuInflater menuInflater = getMenuInflater();
        menuInflater.inflate(R.menu.menu_act_set_recipe_toolbar, menu);
    }

```

```

mMenuKeyIngredInfo = menu.findItem(R.id
    .menu_act_set_recipe_key_ingred_info);

mMenuKeyIngredInfo.setVisible(false);

    return super.onCreateOptionsMenu(menu);
}

@Override
public boolean onOptionsItemSelected(@NonNull MenuItem item) {
    String title = String.valueOf(item.getTitle()).toLowerCase();
    if (title.equalsIgnoreCase("Key Ingredients")) mDiaKeyIngred.show();
    else mDiaVisibility.show();
    return super.onOptionsItemSelected(item);
}

private void setRecipeDefaultValues() {
    //Set recipe author and authorUid
    DbUsers dbUsers = new DbUsers();
    dbUsers.getUserInfo(user -> {
        mRecipe.setAuthor(user.getDisplayName());
        mRecipe.setAuthorUid(user.getId());
    });
    //Set recipe visibility by default
    mRecipe.setVisibility("visible");
}

private void backStep() {
    if (mCurrentPosition != 0) mCurrentPosition -= 1;

    /*Show toolbar menu key ingredient info if the
     *current position is on key ingredient fragment*/
    mMenuKeyIngredInfo.setVisible(mCurrentPosition == 3);

    //If button text is "submit", revert to "next"
    if (mCurrentPosition != mAdapter.getItemCount() - 1)
        mActBinding.btnNext.setText("Next");

    mActBinding.pager.setCurrentItem(mCurrentPosition, true);
}

private void nextStep() {
    if (validToProceed()) {
        String btnText = String.valueOf(mActBinding.btnNext.getText());

        //On finish click

```

```

if (btnText.equalsIgnoreCase("Finish")) uploadImage();

//Change button text from "next" to "finish" when end is reached
if (mCurrentPosition == mAdapter.getItemCount() - 2) {
    mActBinding.btnNext.setText("Finish");
}

//Go to next fragment
if (mCurrentPosition != mAdapter.getItemCount() - 1) mCurrentPosition
+= 1;

/*Show toolbar menu key ingredient info if the
 *current position is on key ingredient fragment*/
mMenuKeyIngrdInfo.setVisible(mCurrentPosition == 3);
mActBinding.pager.setCurrentItem(mCurrentPosition, true);
}

}

private boolean validToProceed() {
    //Check if inputs are not empty
    switch (mCurrentPosition) {
        case 0:
            if (mFragNameOrigin.getName().equals("")) {
                mFragNameOrigin.requestFocusOnName();
                Toast.makeText(this, getString(R.string.name_cannot_be_empty),
                    Toast.LENGTH_SHORT).show();
                return false;
            } else return true;
        case 1:
            if (mFragDesc.getDescription().equals("")) {
                mFragDesc.requestFocusOnDescription();
                Toast.makeText(this,
                    getString(R.string.description_cannot_be_empty),
                    Toast.LENGTH_SHORT).show();
                return false;
            } else return true;
        case 2:
            if (mFragIngredients.getIngredientList().isEmpty()) {
                mFragIngredients.requestFocusOnIngredients();
                Toast.makeText(this,
                    getString(R.string.ingredients_cannot_be_empty),
                    Toast.LENGTH_SHORT).show();
                return false;
            } else return true;
        case 3:
            if (mFragKeyIngredients.getKeyIngredientList().isEmpty()) {

```

```

        mFragKeyIngredients.requestFocusOnKeyIngredients();
        Toast.makeText(this,
getString(R.string.key_ingredients_cannot_be_empty),
        Toast.LENGTH_SHORT).show();
        return false;
    } else return true;
case 4:
    if (mFragInstructions.getInstructionList().isEmpty()) {
        mFragInstructions.requestFocusOnInstructions();
        Toast.makeText(this,
getString(R.string.instructions_cannot_be_empty),
        Toast.LENGTH_SHORT).show();
        return false;
    } else return true;
case 5:
    if (mFragImage.getImageUri() == null) {
        Toast.makeText(this, getString(R.string.image_cannot_be_empty),
        Toast.LENGTH_SHORT).show();
        return false;
    } else return true;
}
return false;
}

private void uploadImage() {
    mDiaLoading.show();

    Uri imgUri = mFragImage.getImageUri();
    StorageReference storageRef =
    FirebaseStorage.getInstance().getReference("userRecipesImg");
    StorageReference fileRef = storageRef.child(System.currentTimeMillis() +
    "." + getFileExtension(imgUri));

    //Upload image to Firebase Storage
    fileRef.putFile(imgUri)
    .addOnSuccessListener(taskSnapshot -> {
        //Get image download url
        fileRef.getDownloadUrl()
        .addOnSuccessListener(uri ->
        setRecipePrimaryDetails(String.valueOf(uri)))
        .addOnFailureListener(e -> {
            mDiaLoading.dismiss();
            e.printStackTrace();
        });
    })
    .addOnFailureListener(e -> {

```

```

        mDiaLoading.dismiss();
        e.printStackTrace();
    });
}

private String getFileExtension(Uri uri) {
    ContentResolver cR = getContentResolver();
    MimeMimeTypeMap mime = MimeMimeTypeMap.getSingleton();
    return mime.getExtensionFromMimeType(cR.getType(uri));
}

private void setRecipePrimaryDetails(String imageUrl) {
    List<ModelUser> finalMemberList = new ArrayList<>();
    List<ModelUser> tempMemberList = new ArrayList<>();

    DbCircle dbCircle = new DbCircle();
    dbCircle.getCircleMembers(mCircleKey, memberList -> {
        finalMemberList.clear();
        tempMemberList.clear();
        tempMemberList.addAll(memberList);

        //Remove current user from list
        for (ModelUser user : tempMemberList) {
            if (mUser.getUid().equals(user.getUid())) {
                tempMemberList.remove(user);
                break;
            }
        }

        //Get 75% of members
        long size = Math.round(tempMemberList.size() * 0.75);

        Random random = new Random();
        for (long i = 0; i < size; i++) {
            int index = random.nextInt(tempMemberList.size());
            finalMemberList.add(tempMemberList.get(index));
            tempMemberList.remove(index);
        }

        //Get member keys
        List<String> voterKeyList = new ArrayList<>();
        for (ModelUser user : finalMemberList) voterKeyList.add(user.getUid());

        mRecipe.setVoters(voterKeyList);
        setRecipeSecondaryDetails(imageUrl);
        uploadRecipe();
    });
}

```

```

        });
    }

private void setRecipeSecondaryDetails(String imageUrl) {
    mRecipe.setKey(mDbUsers.getKey());
    mRecipe.setName(mFragNameOrigin.getName());
    mRecipe.setDescription(mFragDesc.getDescription());
    mRecipe.setImageUrl(imageUrl);
    mRecipe.setOrigin(mFragNameOrigin.getOrigin().toLowerCase());

    mRecipe.setPrivacy("public");
    mRecipe.setStatus("for approval");
    mRecipe.setInstructions(mFragInstructions.getInstructionList());
    mRecipe.setIngredients(mFragIngredients.getIngredientList());

    //Add assumed ingredient
    List<String> keyIngredientList = mFragKeyIngredients.getKeyIngredientList();
    if (!keyIngredientList.contains("water")) keyIngredientList.add("water");
    mRecipe.setKeyIngredients(keyIngredientList);
}

private void uploadRecipe() {
    mDbUsers.sharePublicRecipe(mRecipe, shared -> {
        mDiaLoading.dismiss();
        if (shared) mHandler.postDelayed(() -> mDiaResult.show(), 200);
        else {
            Toast.makeText(this, getString(R.string.failed_to_share_recipe),
                Toast.LENGTH_SHORT).show();
        }
    });
}

@Override
public void onBackPressed() {
    if (mBackPressedTime + 2000 > System.currentTimeMillis()) {
        mToast.cancel();
        super.onBackPressed();
        return;
    } else {
        mToast = Toast.makeText(this, getString(R.string.press_back_twice),
            Toast.LENGTH_SHORT);
        mToast.show();
    }
    mBackPressedTime = System.currentTimeMillis();
}
}

```

## **APPENDIX D (EVALUATION TOOL OR TEST DOCUMENTS)**

## EVALUATION TOOL OR TEST DOCUMENTS

Name (optional): \_\_\_\_\_ Gender: \_\_\_\_\_ Age: \_\_\_\_\_

Directions: Read the statement on each column of the table then scale the statements based on what you see in the KUSINASYON application.

Likert Scale represent as:

4 – Strongly Agree

3 – Agree

2 – Disagree

1 – Strongly Disagree

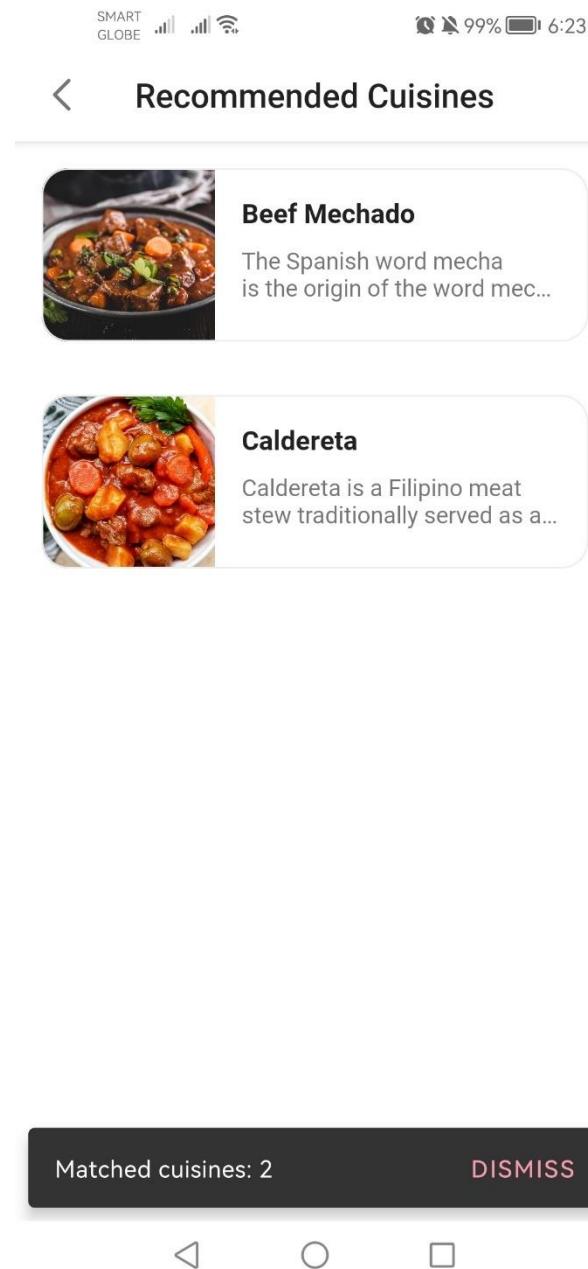
**Table 8: Survey Form Questionnaire**

		1	2	3	4
	<b>USER - FRIENDLINESS</b>				
1.	The purpose of the application increases the user's interest in cooking.				
2.	The application has a clear and consistent layout.				
3.	Minimum controls are needed to operate the application.				
	<b>PERFORMANCE</b>				
1.	The size of the application is lightweight.				
2.	The application's processing of data is quick.				
3.	The application displays information in real-time.				
	<b>DESIGN AND INTERFACE</b>				
1.	Design and choice of colors are appropriate.				
2.	The icons used in the application are appropriate.				
3.	The application components (e.g., buttons, labels, etc.) are not confusing.				

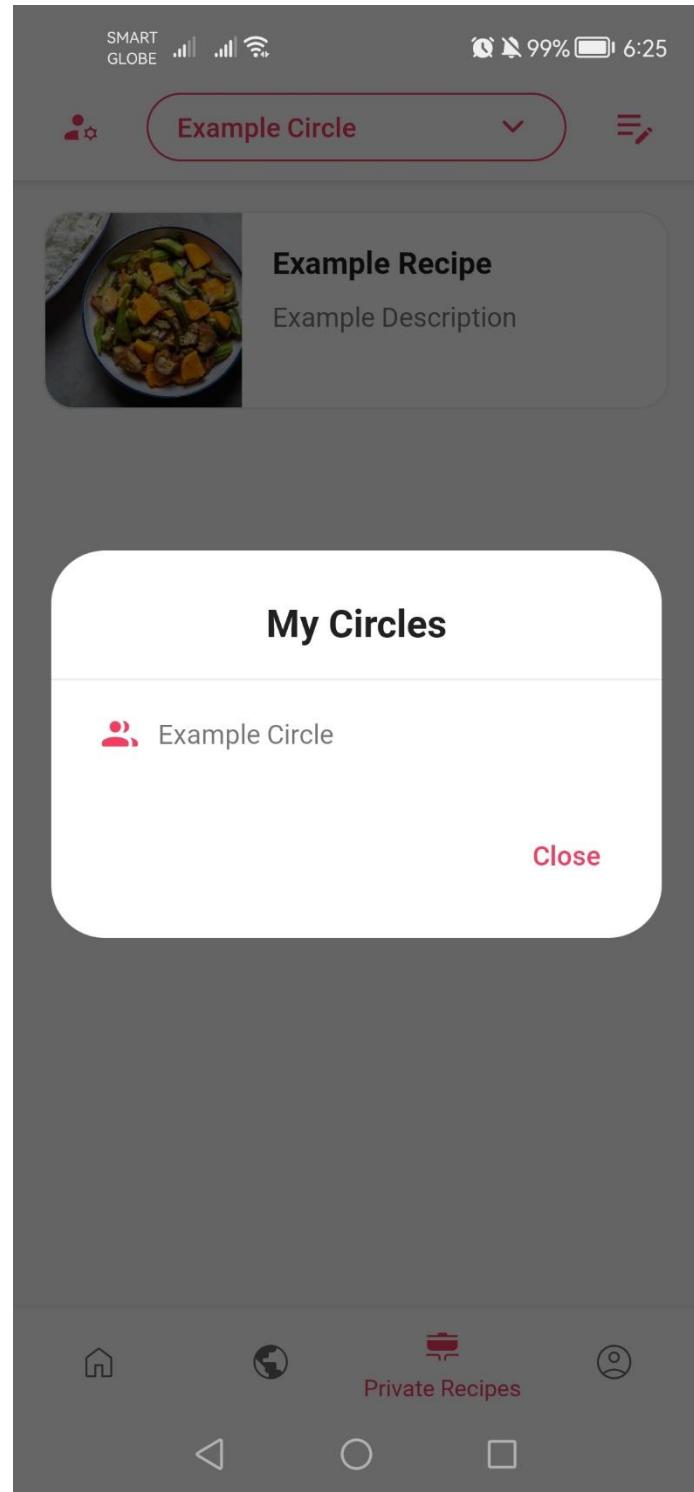
		1	2	3	4
	<b>CUISINE RECOMMENDATION</b>				
1.	The cuisine recommendation helps the user to quickly decide what to cook.				
2.	The cuisine recommendation filters the recipes depending on the selected ingredients.				
3.	The process of selecting ingredients for the cuisine recommendation is straightforward and easy.				
	<b>RECIPE SHARING</b>				
1.	The application allows the user to share their own recipe.				
2.	The application allows the user to edit and delete their shared recipe.				
3.	The recipe sharing helps the user to discover new recipes.				
4.	The application allows the user to share recipes privately and publicly.				
5.	The process of creating and joining a circle is straightforward and easy.				
	<b>VOICE ASSISTED TECHNOLOGY</b>				
1.	The voice assistant helps the user to cook efficiently.				
2.	The voice assistant can easily recognize the voice commands.				
3.	The voice assistant can follow the user's commands.				

## **APPENDIX E (SAMPLE INPUT/OUTPUT/REPORTS)**

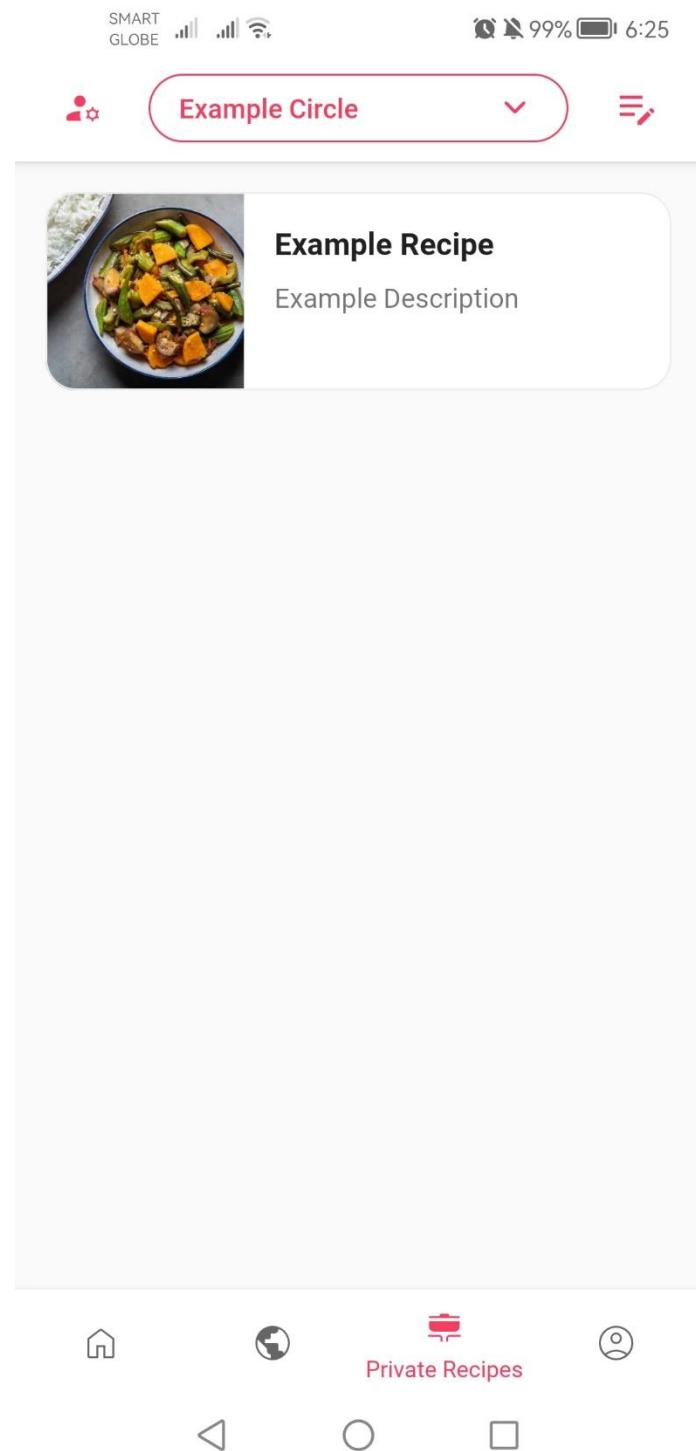
## SAMPLE INPUT/OUTPUT/REPORTS



**Figure 29: Sample Recommended Cuisine/Recipe**



**Figure 30: Sample Private Circle**



**Figure 31: Sample Private Recipe**

[Luzon](#)[Visayas](#)[Mindanao](#)[More](#)

The image shows a grid of four recipe cards, each featuring a small thumbnail image of the dish, the recipe name in bold, and a brief description.

- Authentic Laing Recipe**  
Dried Taro leaves cooked in coconut milk with a lot of chil...
- Batangas Bulalo**  
Batangas Bulalo is probably the most basic and simplest ...
- Crispy Bagnet Ilocos**  
Bagnet is a popular Ilocano dish which consists of pork b...
- My public recipe**  
Example Description

  
Public Recipes

**Figure 32: Sample Public Recipe**

## &lt; My Recipes

Publicly Shared

Privately Shared

**My public recipe**

Example Description

**Figure 33: User's Publicly Shared Recipe List**

## < My Recipes

Publicly Shared

Privately Shared



### Example Recipe

Example Description



**Figure 34: User's Privately Shared Recipes List**



## View Circle Info

**Test**

5 people

## Admin



Sam Sepiol

## Members



Baymax



Patrick



Neo

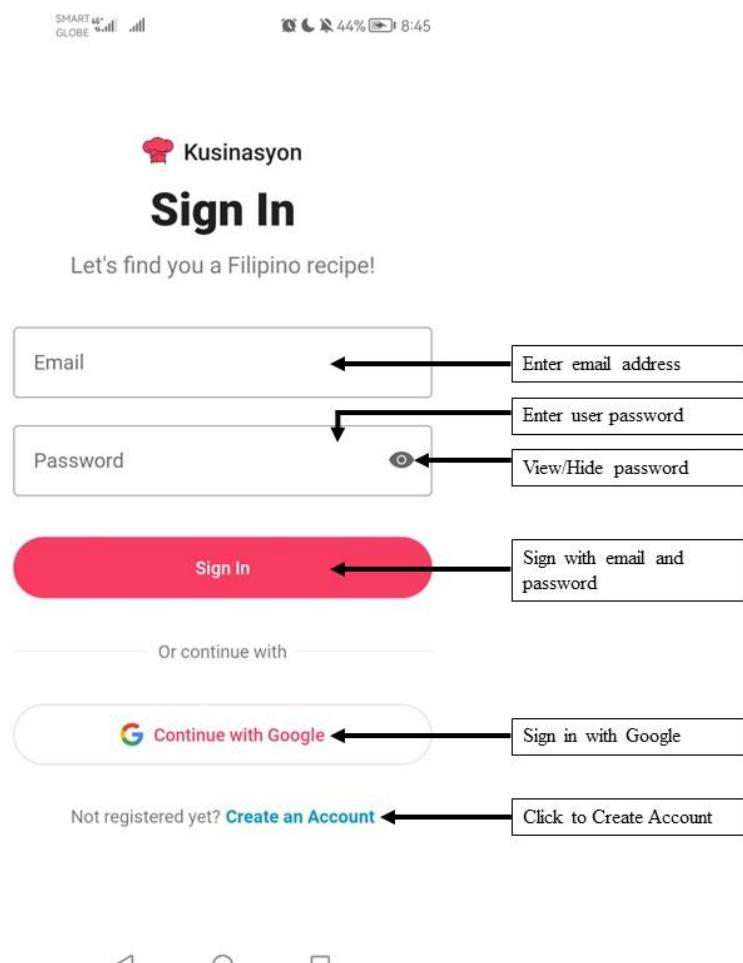


Snorlax

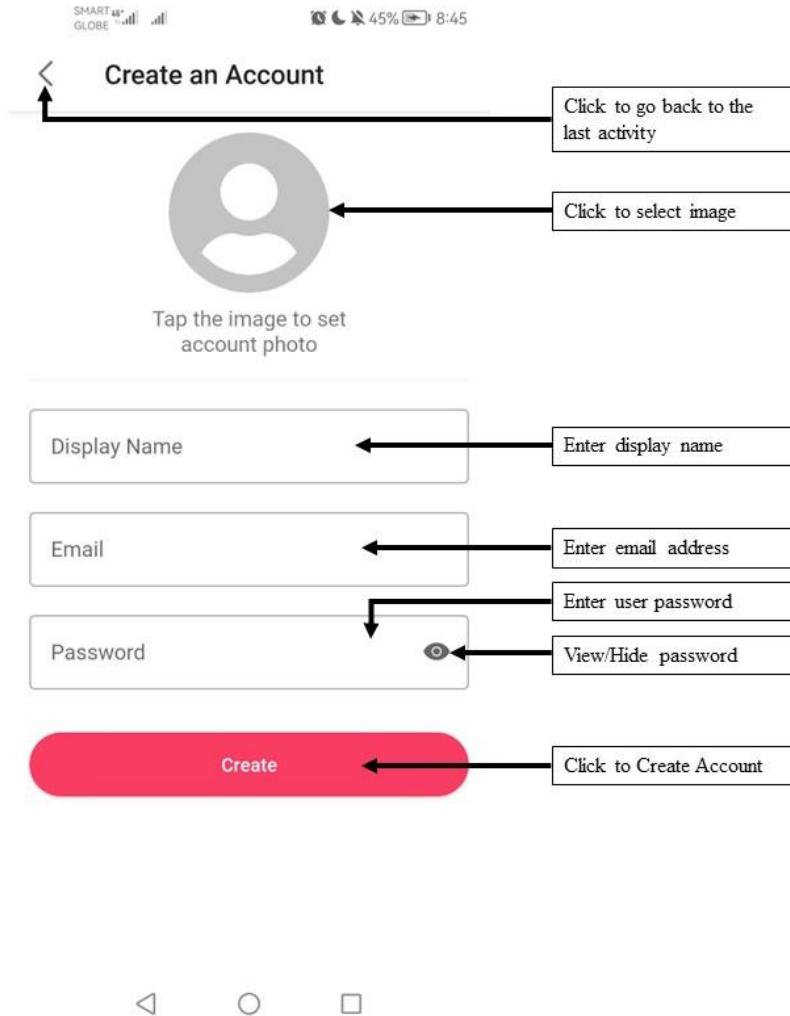
**Figure 35: View Circle Members**

## **APPENDIX F (USER'S GUIDE)**

## USER'S GUIDE



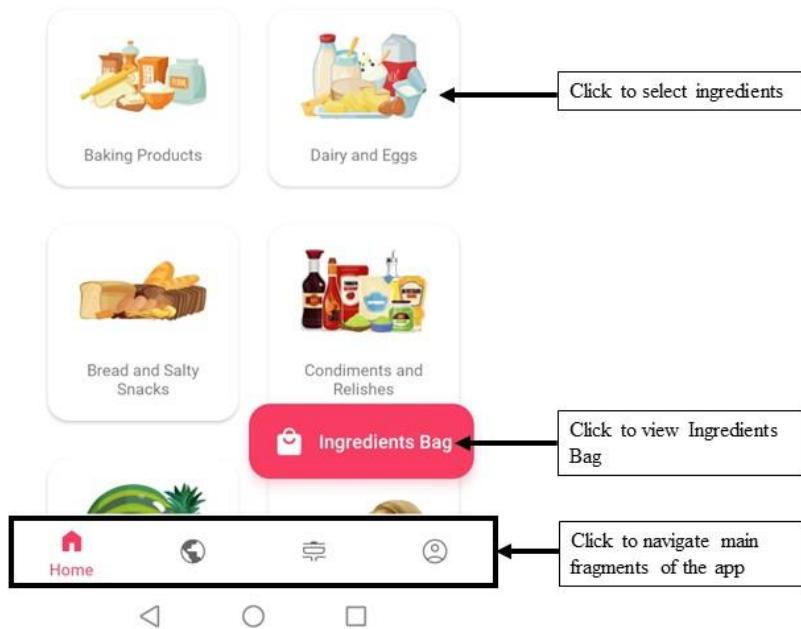
**Figure 36: Sign-in Guide**



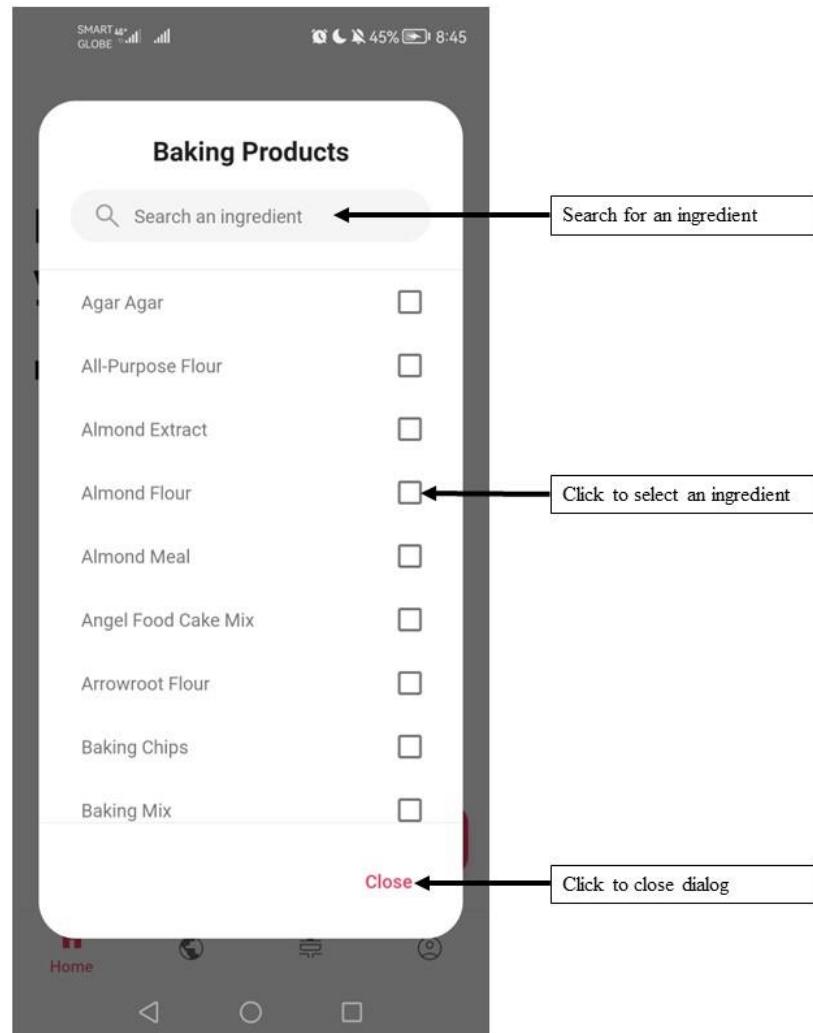
**Figure 37: Create Account Guide**

## Let's select your ingredients!

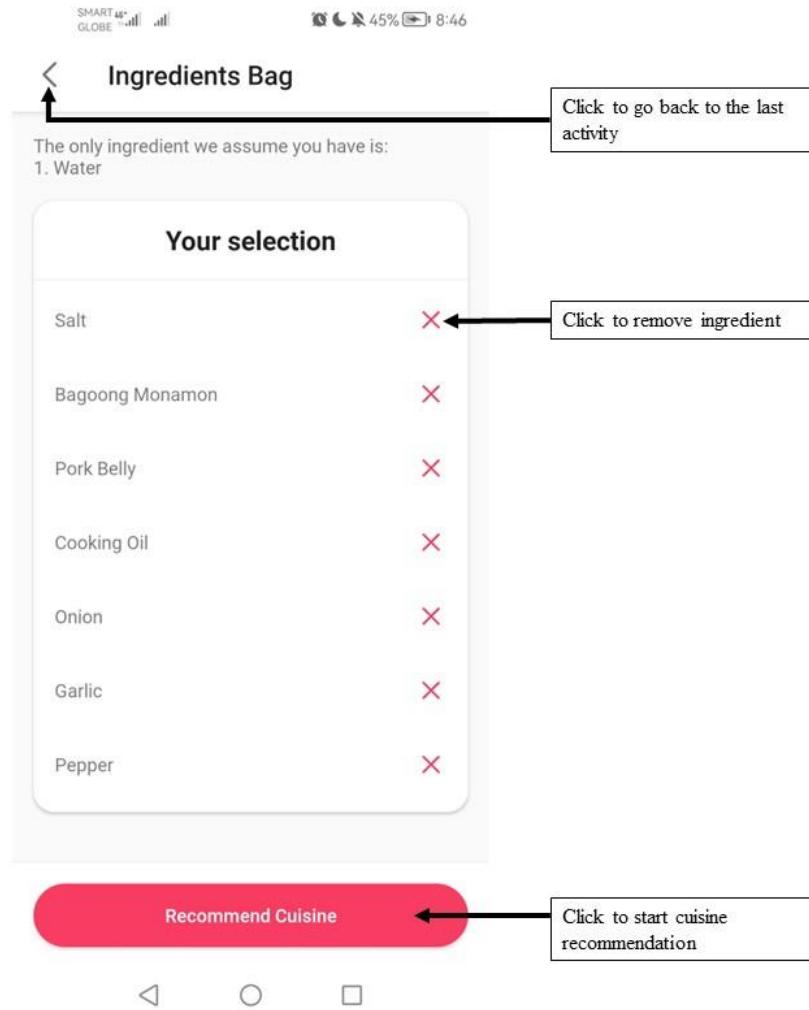
### Pick by Categories



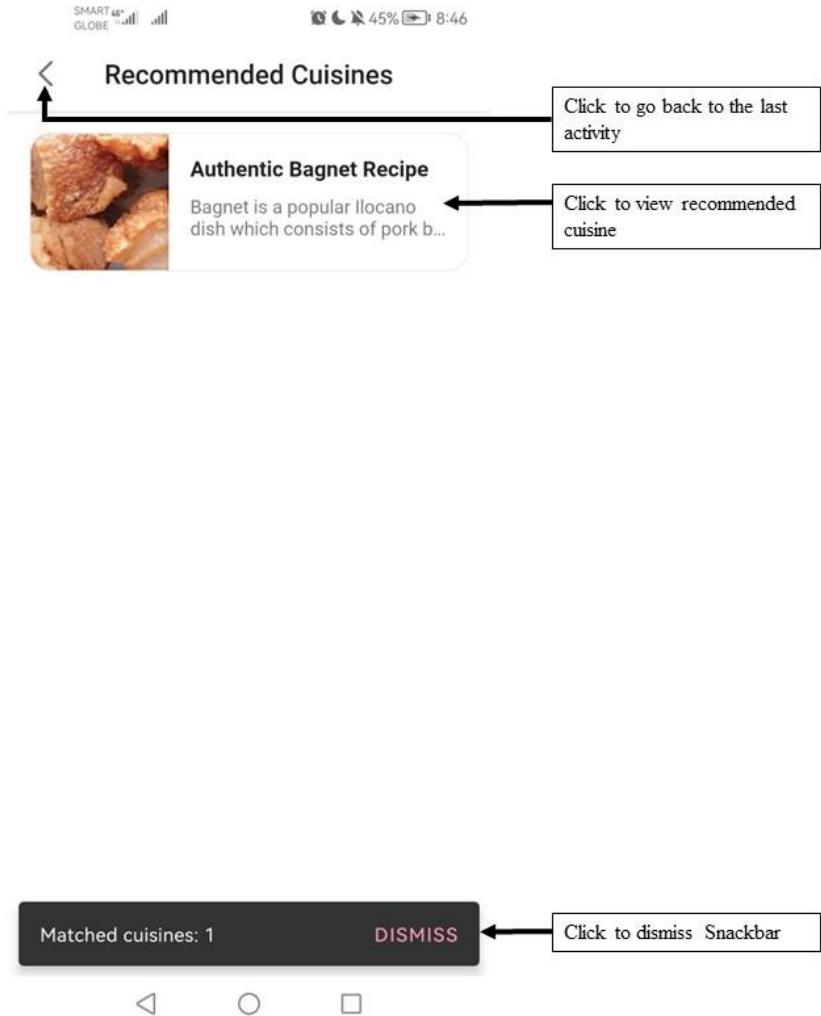
**Figure 38: Home Guide**



**Figure 39: Select Ingredient Guide**



**Figure 40: Ingredients Bag Guide**



**Figure 41: Recommended Cuisines Guide**

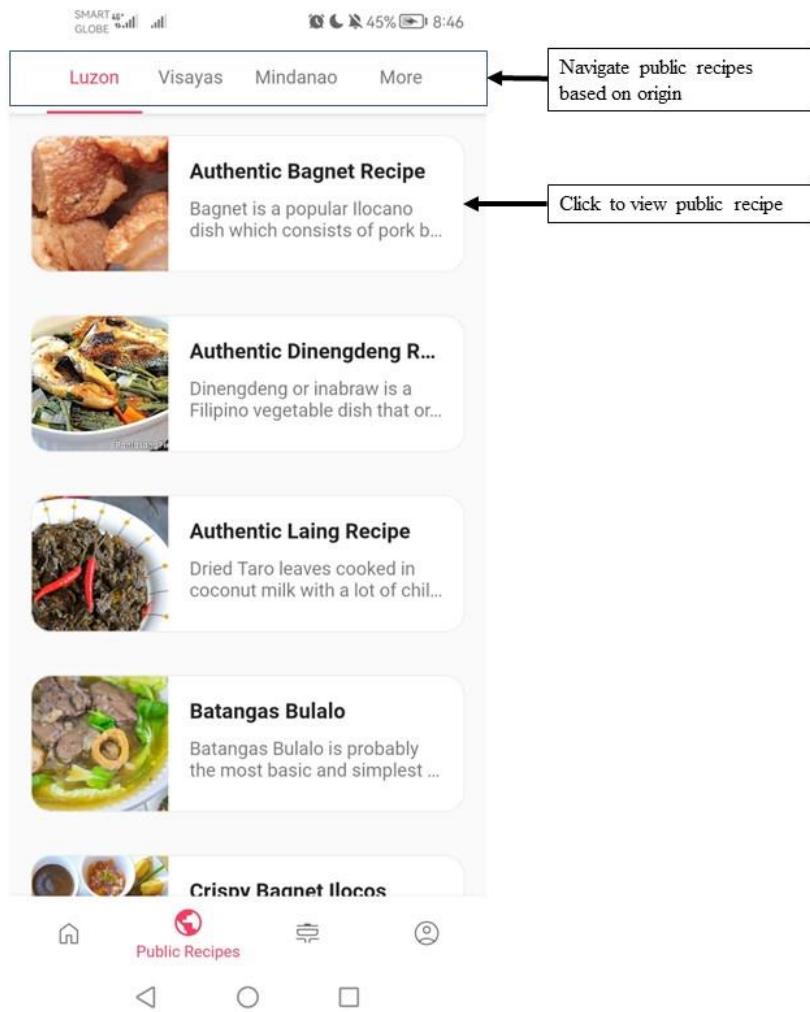
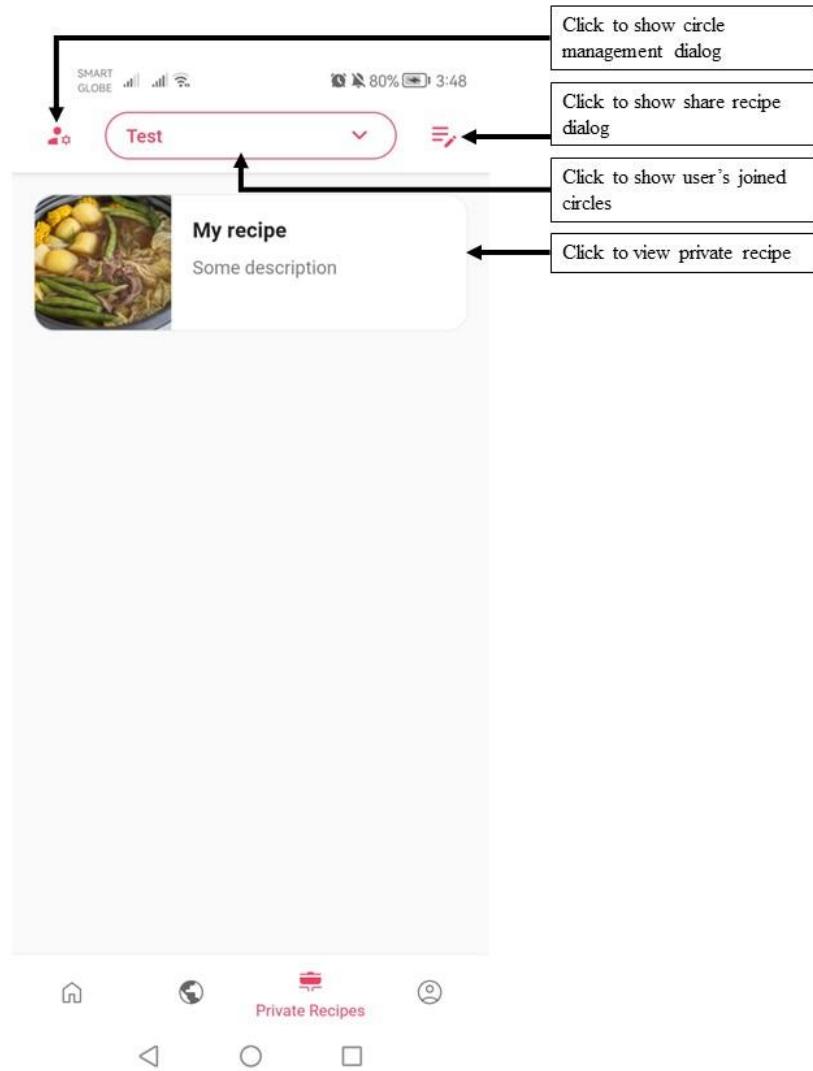
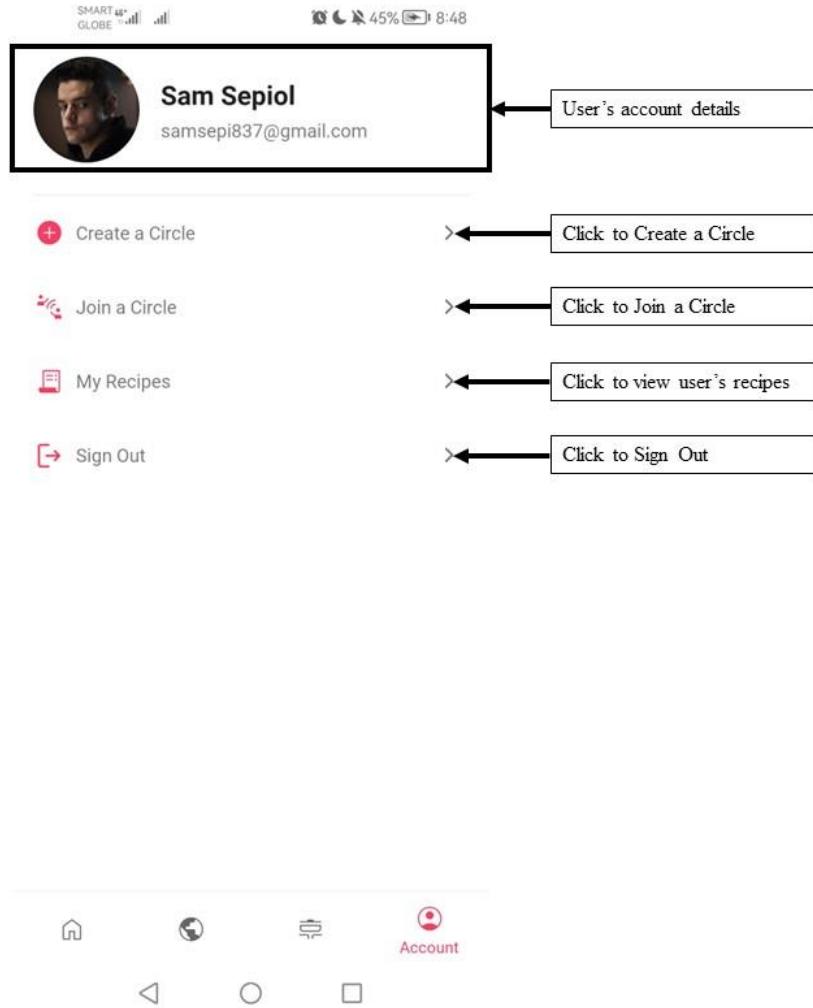


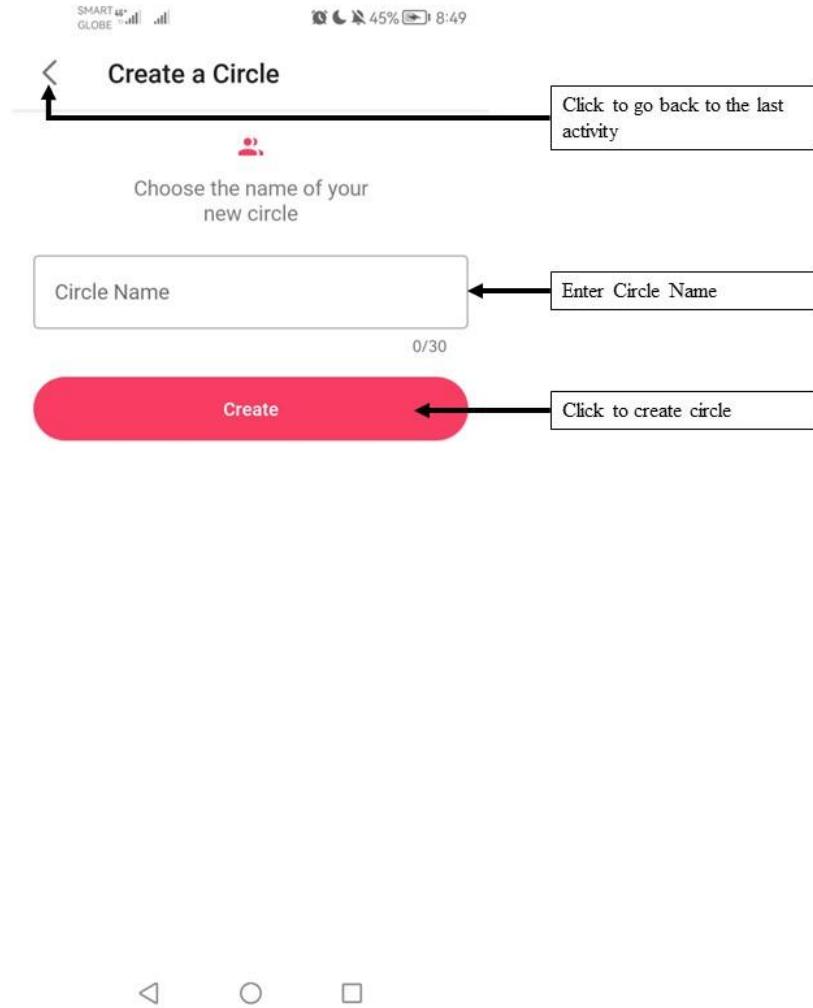
Figure 42: Public Recipes Guide



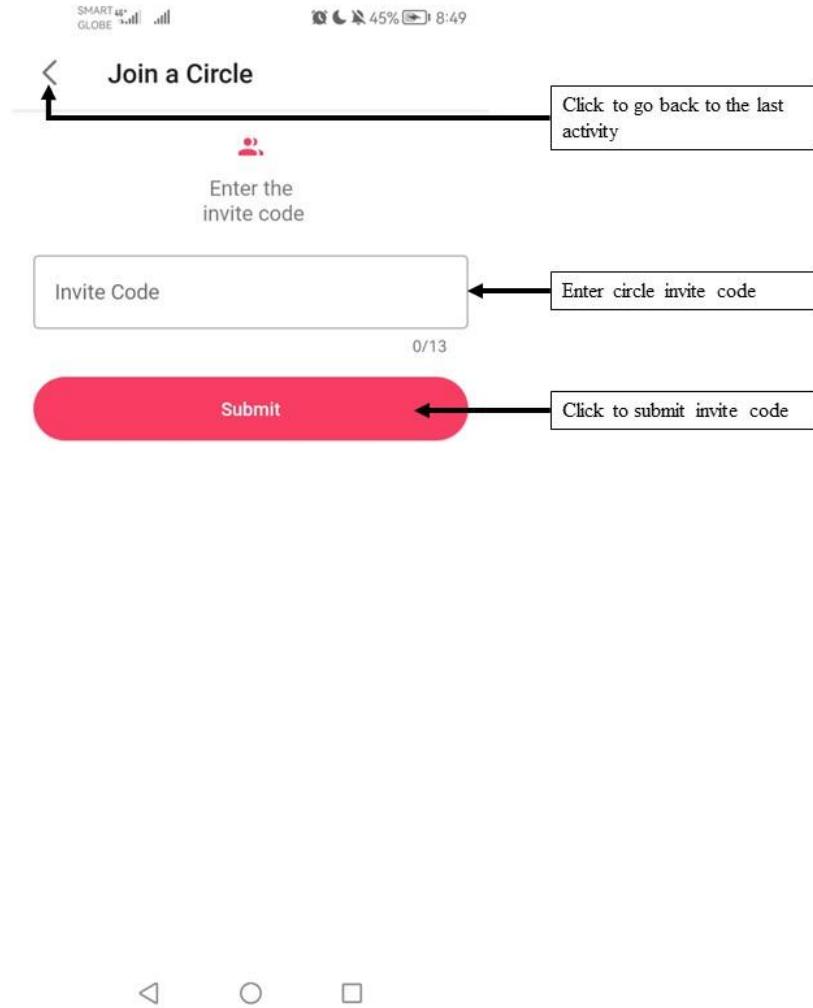
**Figure 43: Private Recipes Guide**



**Figure 44: User Account Guide**



**Figure 45: Create Private Circle Guide**



**Figure 46: Join Private Circle Guide**

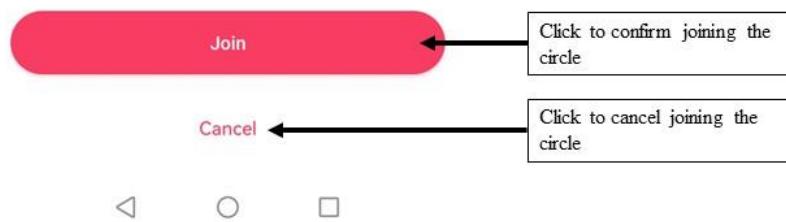
**Great. You're about to join  
the Test circle.**

Here's who is waiting for you:

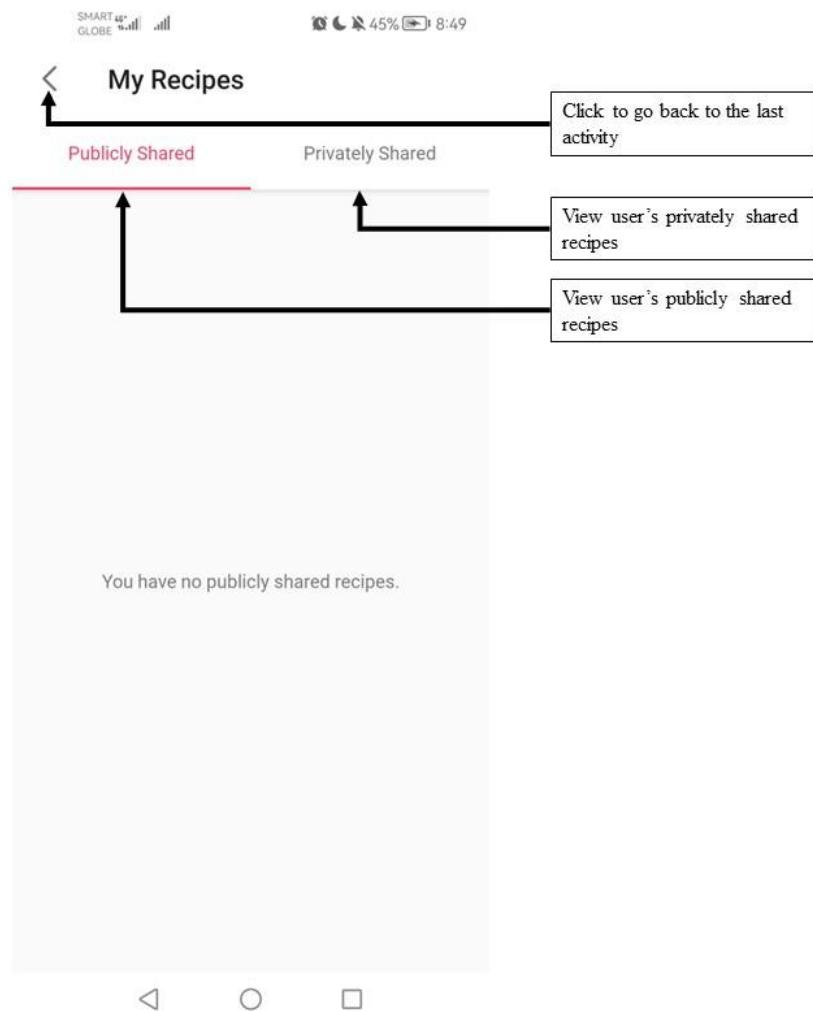


Sam Sepiol

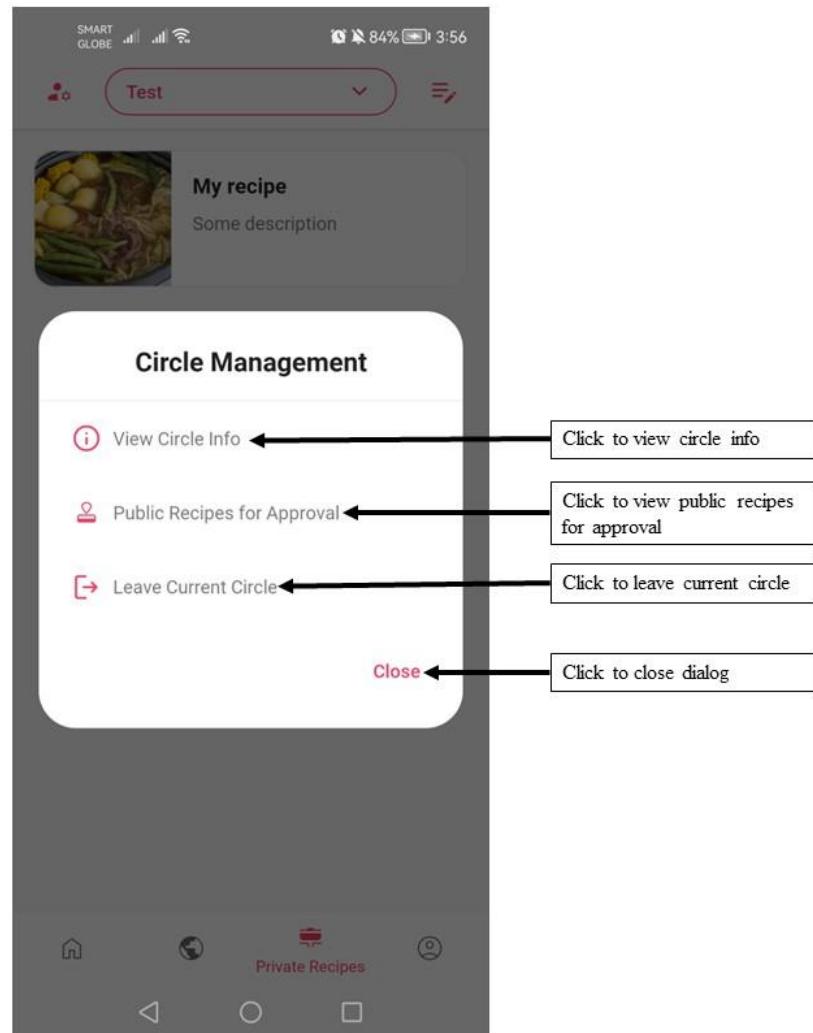
Circle admin



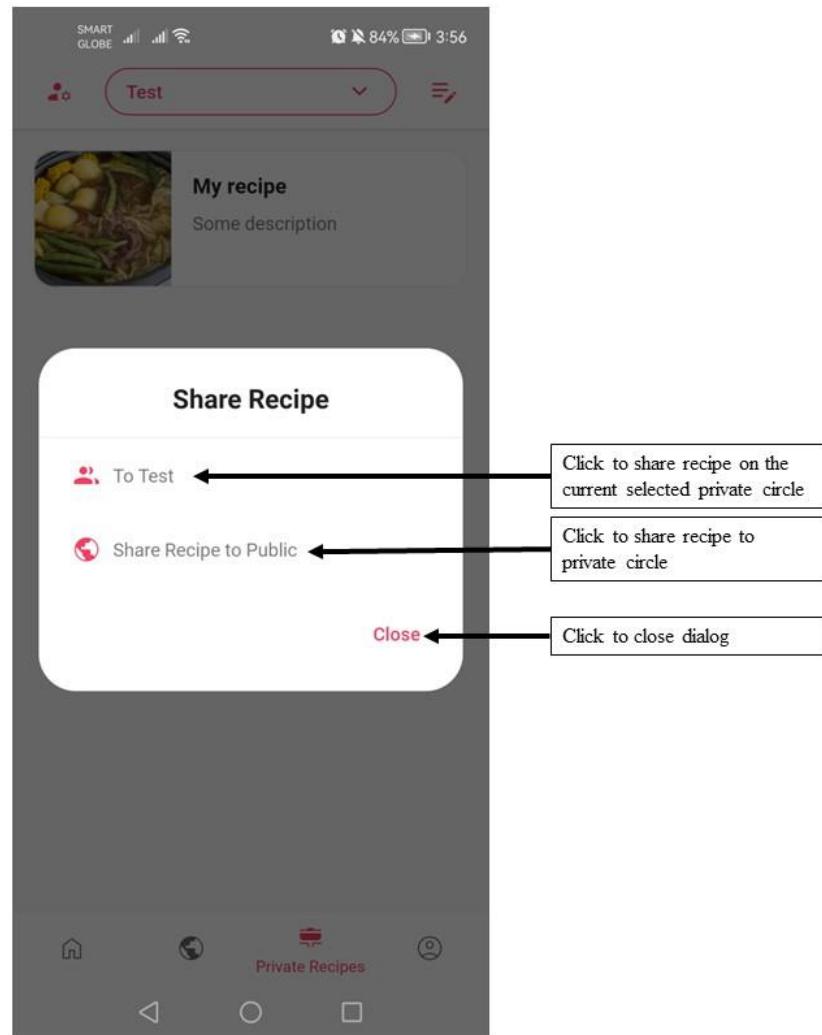
**Figure 47: Confirm Join Private Circle Guide**



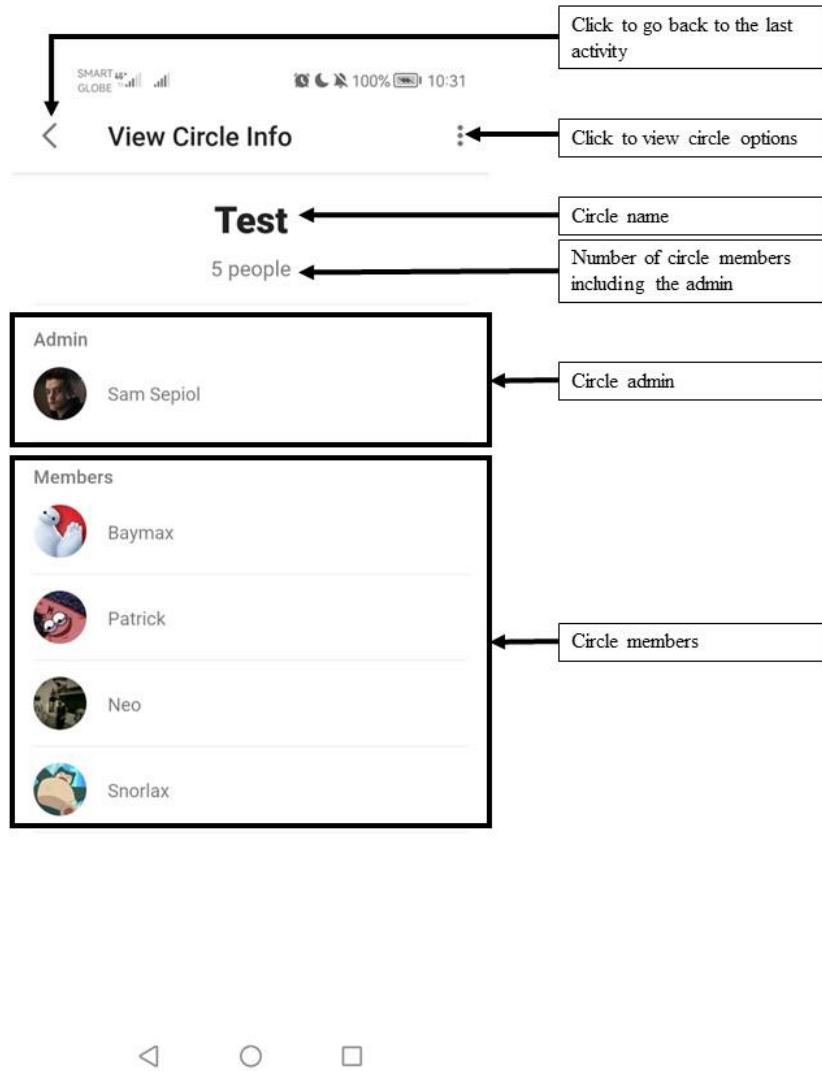
**Figure 48: My Recipes Guide**



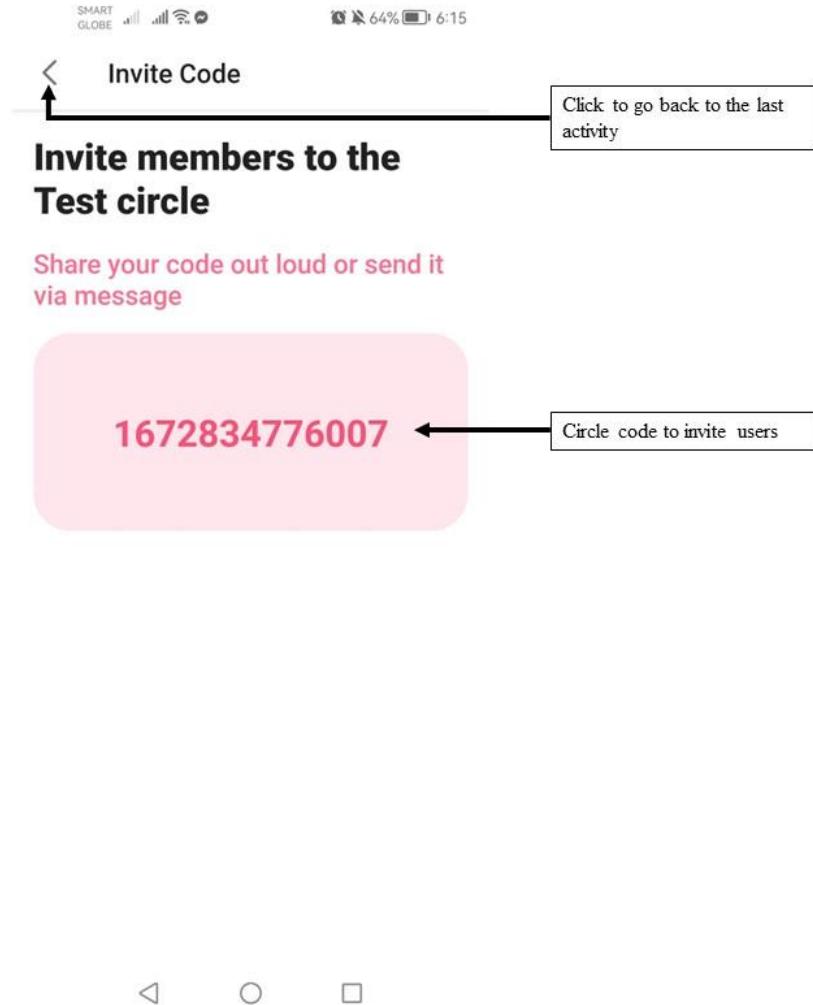
**Figure 49: Circle Management Guide**



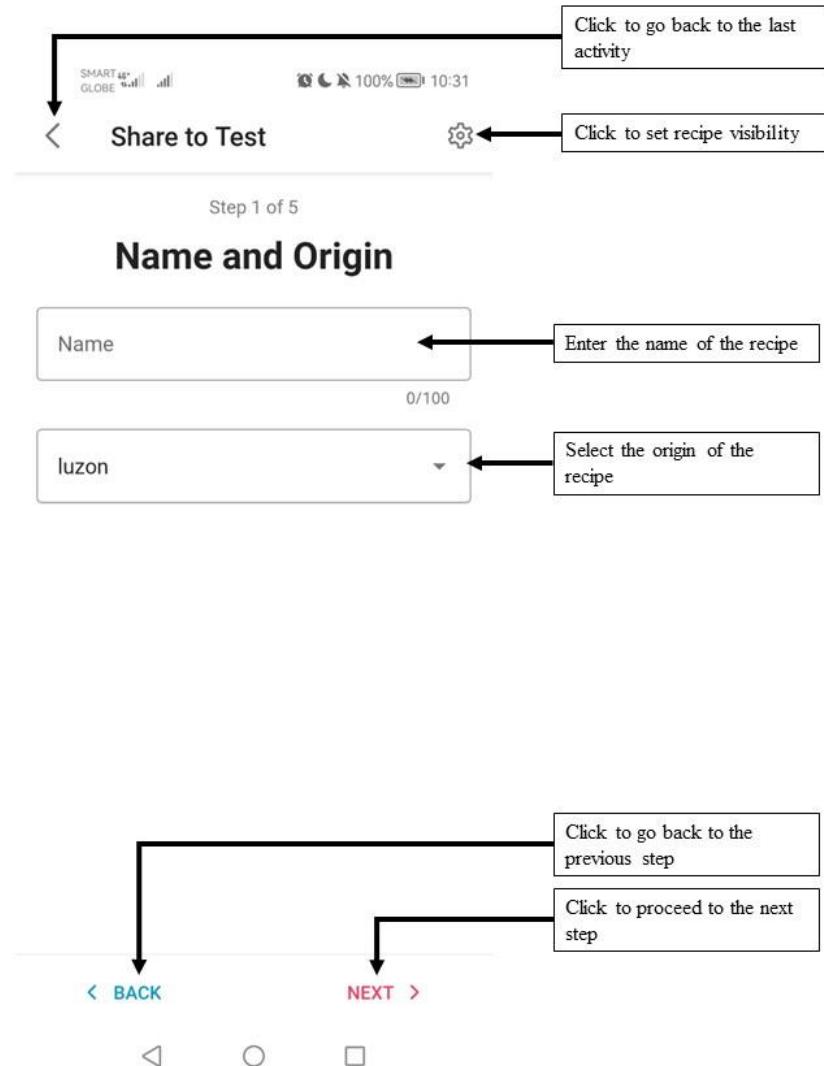
**Figure 50: Share Recipe Selection Guide**



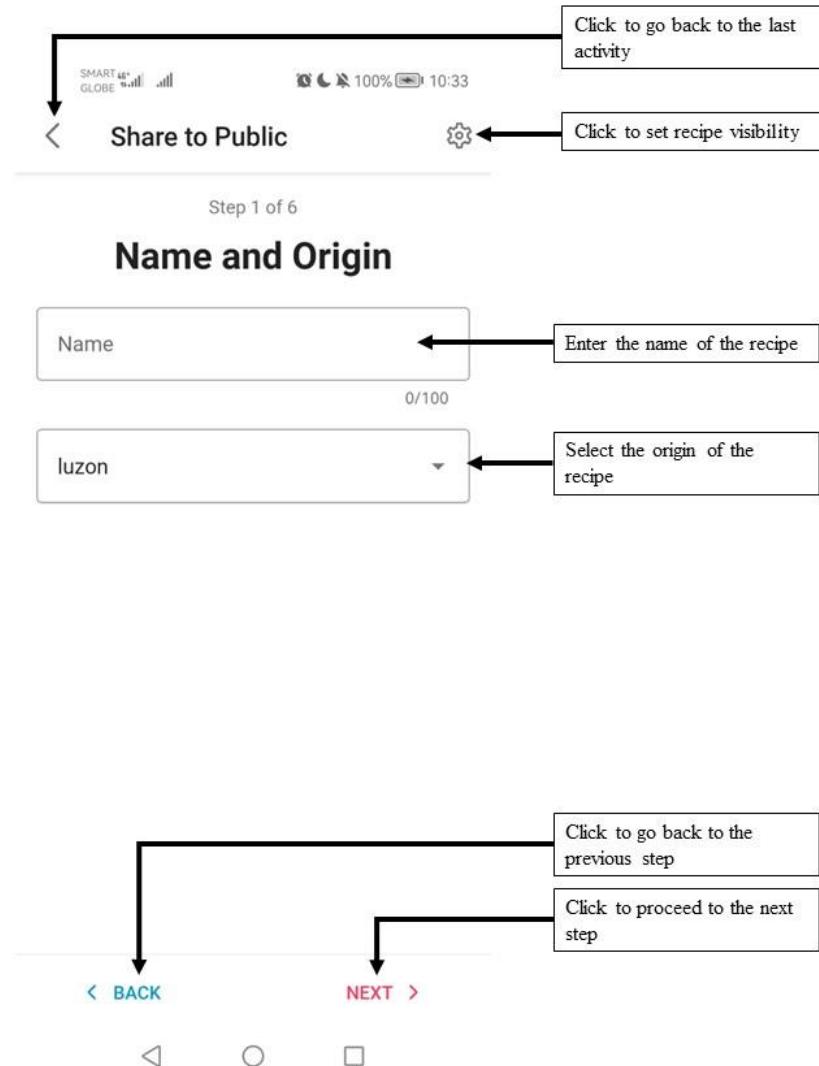
**Figure 51: View Circle Info Guide**



**Figure 52: Invite a User Guide**



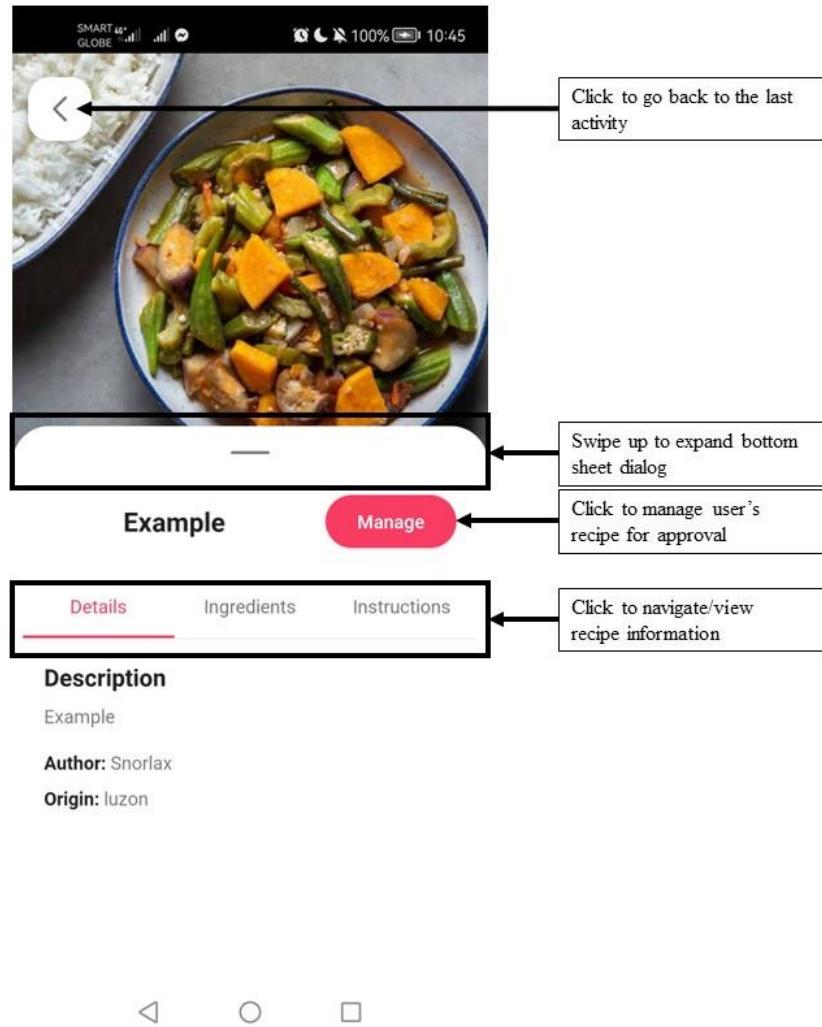
**Figure 53: Share Private Recipe Guide**



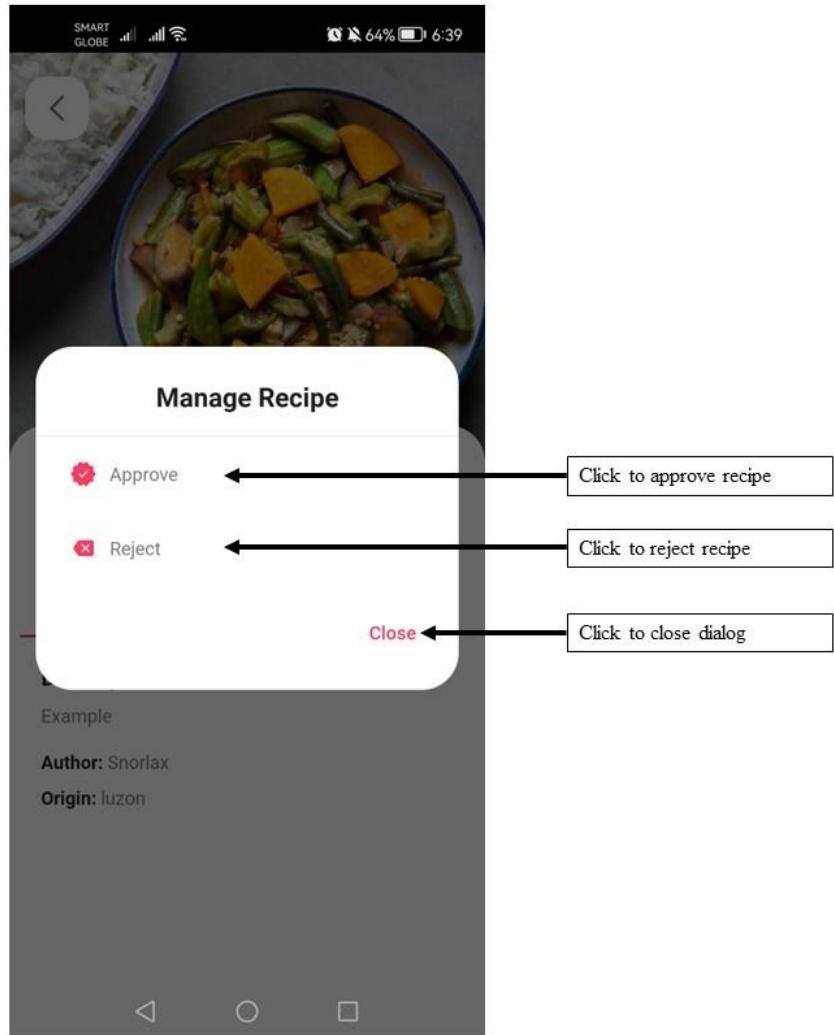
**Figure 54: Share Public Recipe Guide**



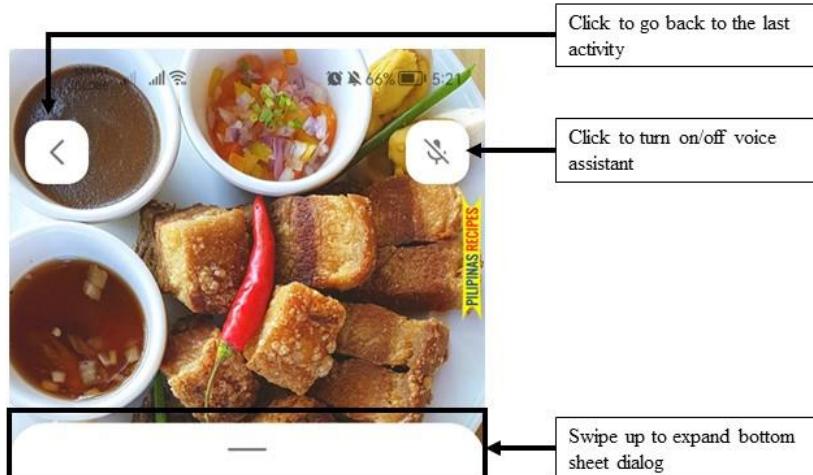
**Figure 55: Public Recipes for Approval Guide**



**Figure 56: View Public Recipe for Approval Guide**



**Figure 57: Manage Recipe for Approval Guide**



### Crispy Baguettes Ilocos

Details

Ingredients

Instructions

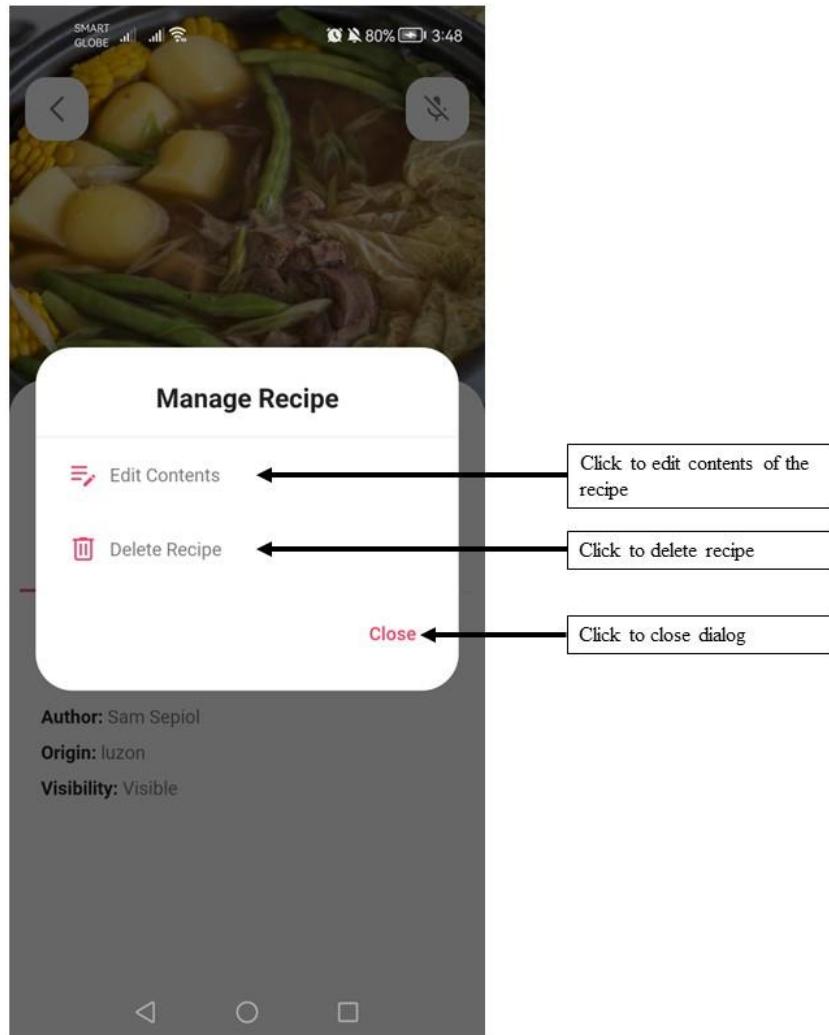
Click to navigate/view  
recipe information

#### Instructions

- 1 | Preheat oven to 175F.
- 2 | Pour water into a pressure cooker. Add lemongrass, salt, whole peppercorn, and onion. Let boil.
- 3 | Add pork belly. Boil for 3 minutes.  
Cover the pressure cooker. Cook for 15 minutes.
- 4 | Let the pressure out completely Put the pork belly on a clean plate and let it cool-down for 10 minutes.



**Figure 58: View Recipe Guide**



**Figure 59: Manage Recipe (Author) Guide**

## **APPENDIX G (SURVEY RESULTS AND TALLY)**

## SURVEY RESULTS AND TALLY

**Table 9: User-Friendliness Tally**

<b>USER - FRIENDLINESS</b>					
	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>TOTAL</b>
<b>1</b>		2	9	19	<b>30</b>
<b>2</b>			5	25	<b>30</b>
<b>3</b>		2	12	16	<b>30</b>

**Table 10: Performance Tally**

<b>PERFORMANCE</b>					
	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>TOTAL</b>
<b>1</b>		1	8	21	<b>30</b>
<b>2</b>		1	9	20	<b>30</b>
<b>3</b>		1	8	21	<b>30</b>

**Table 11: Design and Interface Tally**

<b>DESIGN AND INTERFACE</b>					
	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>TOTAL</b>
<b>1</b>			9	21	<b>30</b>
<b>2</b>			6	24	<b>30</b>
<b>3</b>			10	20	<b>30</b>

**Table 12: Objective 1 Tally**

<b>CUISINE RECOMMENDATION</b>					
	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>TOTAL</b>
<b>1</b>					<b>30</b>
<b>2</b>					<b>30</b>
<b>3</b>					<b>30</b>

**Table 13: Objective 2 Tally**

RECIPE SHARING					
	1	2	3	4	TOTAL
1			5	25	30
2			6	24	30
3			8	22	30
4			4	26	30
5			8	22	30

**Table 14: Objective 3 Tally**

VOICE ASSISTED TECHNOLOGY					
	1	2	3	4	TOTAL
1		1	7	17	30
2		1	15	14	30
3		1	10	19	30

**Table 15: Respondents Age Tally**

AGE	
16	7
17	8
18	8
19	3
20	
21	1
25	1
NO AGE	2
<b>TOTAL</b>	<b>30</b>

**Table 16: Respondents Gender Tally**

GENDER	
M	14
F	14
NO GENDER	2
<b>TOTAL</b>	<b>30</b>

Name (optional): Jerimiah Edanol Gender: M Age: 18

**Directions:** Read the statement on each column of the table then scale the statements based on what you see in the **KUSINASYON** application.

Likert Scale represent as:

**4 – Strongly Agree**

**3 – Agree**

**2 – Disagree**

**1 – Strongly Disagree**

**Table 17: Sample Data of Survey Questionnaire**

		1	2	3	4
	<b>USER - FRIENDLINESS</b>				
1.	The purpose of the application increases the user's interest in cooking.				/
2.	The application has a clear and consistent layout.				/
3.	Minimum controls are needed to operate the application.			/	
	<b>PERFORMANCE</b>				
1.	The size of the application is lightweight.			/	
2.	The application's processing of data is quick.			/	
3.	The application displays information in real-time.			/	
	<b>DESIGN AND INTERFACE</b>				
1.	Design and choice of colors are appropriate.				/
2.	The icons used in the application are appropriate.				/
3.	The application components (e.g., buttons, labels, etc.) are not confusing.			/	

	<b>CUISINE RECOMMENDATION</b>			
1.	The cuisine recommendation helps the user to quickly decide what to cook.			/
2.	The cuisine recommendation filters the recipes depending on the selected ingredients.			/
3.	The process of selecting ingredients for the cuisine recommendation is straightforward and easy.			/
	<b>RECIPE SHARING</b>			
1.	The application allows the user to share their own recipe.			/
2.	The application allows the user to edit and delete their shared recipe.			/
3.	The recipe sharing helps the user to discover new recipes.			/
4.	The application allows the user to share recipes privately and publicly.			/
5.	The process of creating and joining a circle is straightforward and easy.			/
	<b>VOICE ASSISTED TECHNOLOGY</b>			
1.	The voice assistant helps the user to cook efficiently.			/
2.	The voice assistant can easily recognize the voice commands.			/
3.	The voice assistant can follow the user's commands.			/

File Edit View Data Transform Analyze Graphs Utilities Extensions Window Help

Visible: 23 of 23 Variables

	Gender	Age	UF1	UF2	UF3	P1	P2	P3	D1	D2	D3	CR1	CR2	CR3	R51	R52	R53	R54	R55	VAT1	VAT2	VAT3
1	1.00	18.00	4.00	4.00	3.00	3.00	4.00	4.00	3.00	4.00	4.00	4.00	4.00	4.00	3.00	3.00	4.00	3.00	4.00	3.00	3.00	3.00
2	2.00	21.00	4.00	3.00	4.00	3.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	3.00	4.00	4.00	3.00	4.00	3.00	4.00	4.00
3	1.00	19.00	3.00	3.00	3.00	2.00	3.00	3.00	3.00	3.00	3.00	3.00	3.00	3.00	4.00	4.00	4.00	4.00	4.00	3.00	3.00	4.00
4	1.00	18.00	4.00	4.00	3.00	4.00	4.00	4.00	3.00	4.00	4.00	4.00	4.00	4.00	3.00	4.00	4.00	4.00	4.00	3.00	4.00	4.00
5	2.00	18.00	3.00	4.00	4.00	4.00	4.00	4.00	3.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	3.00	4.00	4.00
6	2.00	18.00	4.00	3.00	3.00	4.00	3.00	4.00	4.00	4.00	3.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	3.00	4.00	3.00
7	2.00	18.00	3.00	4.00	4.00	3.00	3.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	3.00	3.00	4.00
8	2.00	17.00	4.00	3.00	3.00	4.00	3.00	4.00	3.00	4.00	4.00	4.00	4.00	4.00	3.00	4.00	4.00	4.00	4.00	3.00	3.00	3.00
9	2.00	17.00	4.00	4.00	3.00	4.00	4.00	4.00	3.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	3.00	3.00	3.00
10	1.00	17.00	3.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	3.00	4.00	4.00	4.00	4.00	3.00	4.00	2.00
11	1.00	17.00	3.00	4.00	3.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	3.00	4.00	4.00
12	1.00	16.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	3.00	4.00	2.00	4.00	4.00	4.00	4.00	4.00
13	1.00	16.00	3.00	4.00	2.00	4.00	4.00	4.00	3.00	4.00	3.00	4.00	3.00	3.00	4.00	4.00	4.00	4.00	3.00	4.00	4.00	4.00
14	1.00	17.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	3.00	4.00	4.00	4.00	4.00	3.00	4.00	4.00
15	2.00	17.00	3.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	3.00	4.00	4.00	4.00	4.00	3.00	4.00	4.00
16	2.00	18.00	3.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	3.00	4.00	4.00	4.00	4.00	3.00	4.00	4.00
17	2.00	17.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	3.00	4.00	4.00	4.00	4.00	3.00	4.00	4.00
18	2.00	19.00	4.00	4.00	4.00	3.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	3.00	3.00	3.00
19	2.00	18.00	2.00	4.00	3.00	3.00	2.00	4.00	3.00	4.00	4.00	4.00	4.00	4.00	2.00	3.00	3.00	3.00	3.00	3.00	3.00	3.00
20	2.00	18.00	2.00	4.00	2.00	3.00	2.00	4.00	3.00	3.00	3.00	3.00	3.00	3.00	3.00	3.00	3.00	3.00	3.00	3.00	3.00	3.00
21	2.00	18.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	3.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00
22	1.00	18.00	4.00	4.00	4.00	4.00	3.00	4.00	3.00	4.00	4.00	4.00	4.00	4.00	3.00	4.00	4.00	4.00	4.00	3.00	4.00	4.00
23	1.00	16.00	4.00	4.00	4.00	4.00	3.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00
24	1.00	17.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00
25	1.00	17.00	4.00	4.00	3.00	4.00	3.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	3.00	4.00	4.00	4.00	4.00	3.00	4.00	4.00
26	1.00	16.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00
27	1.00	16.00	4.00	3.00	3.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	3.00	3.00	4.00	4.00	4.00	4.00	4.00	4.00
28	1.00	16.00	3.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00
29	1.00	16.00	4.00	4.00	3.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00
30	2.00	19.00	4.00	4.00	3.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00	3.00	3.00	3.00
31																						
32																						
33																						
34																						
35																						
36																						
37																						
38																						
39																						

Data View Variable View IBM SPSS Statistics Processor is ready Unicode ON

Figure 60: SPSS Survey Data

## **APPENDIX K (PROJECT INVOLVEMENTS)**

## PROJECT INVOLVEMENTS

(Title of the Study)	<b>Kusinasyon: An Android-based Filipino Cuisine Application with Voice Assisted Technology</b>	
Date of Defense:	<b>December 2, 2022</b>	Campus: <b>STI College Global City</b>
Time: <b>9:00AM – 10:00AM</b>		
Adviser: <b>Joselito G. Oyao</b>	Signature:	

(Name) <b>Neonie Quell A. Ponce</b>	<b>Signature:</b>
Chapter(s)/ Sub-topic(s) written in the documentation: <b>Chapter 1, 3, and Appendices</b>	
Module(s) developed in the system: <b>Recipe Sharing, Private Circle, Voice Assisted Technology</b>	
Topic(s) to be discussed during defense: <b>Project Context, Demo of the application</b>	
Other Contributions: <b>Guide in document writing and application logic.</b>	

(Name) <b>Rhey Franz A. Reyes</b>	<b>Signature:</b>
Chapter(s)/ Sub-topic(s) written in the documentation: <b>Chapter 1, Chapter 2, Appendices</b>	
Module(s) developed in the system: <b>Create and Join Private Circle</b>	
Topic(s) to be discussed during defense: <b>Purpose and Description</b>	
Other Contributions: <b>Document writing and application logic.</b>	

(Name) <b>John Fritz F. Delafer</b>	<b>Signature:</b>
Chapter(s)/ Sub-topic(s) written in the documentation: <b>Chapter 2, Chapter 4, Chapter 5, Appendices</b>	
Module(s) developed in the system: <b>Sign-in, Create Account, Firebase Email Authentication, View Recipe</b>	
Topic(s) to be discussed during defense: <b>Objectives of the Study</b>	
Other Contributions: <b>Document writing, survey tools and computations, and application flow.</b>	

(Name) <b>Dean Clyte B. Rebustillo</b>	<b>Signature:</b>
Chapter(s)/ Sub-topic(s) written in the documentation: <b>Chapter 2, Chapter 3, Appendices</b>	
Module(s) developed in the system: <b>Firebase Email Authentication, Displaying of Recipes</b>	
Topic(s) to be discussed during defense: <b>Scope and Limitations, Demo of application</b>	
Other Contributions: <b>Document writing and testing of the application.</b>	

## **APPENDIX L (PERSONAL TECHNICAL VITAE)**

Curriculum Vitae of  
**JOHN FRITZ F. DELAFER**  
**#57 M PIO FELIPE ST. ZONE 1 NEW LOWER BICUTAN**  
**, TAGUIG CITY**  
**delafer.070011@globalcity.sti.edu.ph**  
**09186063274**



#### EDUCATIONAL BACKGROUND

Level	Inclusive Dates	Name of school/ Institution
Tertiary	2019-2023	STI College – Global City
Vocational/Technical	2017-2019	STI College – Global City
High School	2013-2017	Signal Village National High School
Elementary	2007-2013	EM's Signal Village Elementary School

#### PROFESSIONAL OR VOLUNTEER EXPERIENCE

Inclusive Dates	Nature of Experience/ Job Title	Name and Address of Company or Organization
2021 - 2022	SAP Business One	STI College - Global City
Apr – June	Service Crew	Golden Arches Development Corp.
Aug – Sept, 2019	data encoder at PIO	Municipal Hall of Taguig, PIO
2018 – 2019	IT Support	STI College - Global City

Listed in reverse chronological order (most recent first).

#### AFFILIATIONS

Inclusive Dates	Name of Organization	Position
Listed in reverse chronological order (most recent first).		

#### SKILLS

SKILLS	Level of Competency	Date Acquired
Microsoft office applications	Advanced	2019
Computer Hardware Servicing	Advanced	2019
Knowledgeable in HTML	Intermediate	2017
Knowledgeable in Java, C#	Intermediate	2017
Oriented at Creatives (Photoshop, Illustrator, Premiere)	Intermediate	2018

#### TRAININGS, SEMINARS, OR WORKSHOPS ATTENDED

Inclusive Dates	Title of Training, Seminar, or Workshop
2019	Enigma: A Computer Engineering Convention “Cyber Security”
2016	Red Cross Youth Leadership Development Program

Curriculum Vitae of  
**NEONIE QUELL A. PONCE**



**Block 18 Lot 10 AFP Village Phase 2, Taguig City, Metro Manila**  
**ponce.085213@globalcity.sti.edu.ph**  
**0995-491-4161**

**EDUCATIONAL BACKGROUND**

Level	Inclusive Dates	Name of school/ Institution
Tertiary	2019-2023	STI College Global City
Vocational/Technical	2017-2019	STI College Global City
High School	2013-2017	Army's Angels Integrated School
Elementary	2010-2013	EM's Signal Village Elementary School
Elementary	2009-2010	Hansarang Christian Academy
Elementary	2007-2009	LKBP Montessori School Inc.

**PROFESSIONAL OR VOLUNTEER EXPERIENCE**

Inclusive Dates	Nature of Experience/ Job Title	Name and Address of Company or Organization
2019	Hackathon 1 <sup>st</sup> Placer	STI College Global City
2022	Hackathon 1 <sup>st</sup> Placer	STI College Global City

**Listed in reverse chronological order (most recent first).**

**AFFILIATIONS**

Inclusive Dates	Name of Organization	Position

**Listed in reverse chronological order (most recent first).**

**SKILLS**

SKILLS	Level of Competency	Date Acquired
Mobile Application Development (Android)	Advanced	2017
Desktop Application Development (Java/C#)	Intermediate	2017
Web Development (HTML/CSS)	Basic	2017
SQL	Intermediate	2018
Firebase Realtime Database	Advanced	2022
Computer Network Operations	Basic	2020
Microsoft Office 365 Applications	Advanced	2017

**TRAININGS, SEMINARS, OR WORKSHOPS ATTENDED**

Inclusive Dates	Title of Training, Seminar, or Workshop
2021	SAP Business One – Basic (Financials and Logistics)
2022	SAP Business One – (Implementation and Support)



Curriculum Vitae of

## DEAN CLYTE B. REBUSTILLO

13 – A Col. Rongo Street Central Signal Village Taguig City Zone 2  
Rebustillo.102608@globalcity.sti.edu.ph  
09288994154

### EDUCATIONAL BACKGROUND

Level	Inclusive Dates	Name of school/ Institution
Tertiary	2019-2023	STI College – Global City
Vocational/Technical	2017-2019	STI College – Global City
High School	2013-2017	Sto. Niño Catholic School
Elementary	2007-2013	D.F. Tinkerbell School

### PROFESSIONAL OR VOLUNTEER EXPERIENCE

Inclusive Dates	Nature of Experience/ Job Title	Name and Address of Company or Organization
2021 - 2022	SAP Business One	STI College - Global City
2019	Library Assistant	STI College - Global City

Listed in reverse chronological order (most recent first).

### AFFILIATIONS

Inclusive Dates	Name of Organization	Position
Listed in reverse chronological order (most recent first).		

### SKILLS

SKILLS	Level of Competency	Date Acquired
Microsoft Office Applications	Advanced	2019
Knowledgeable in Mobile Devices	Advanced	2019
Knowledgeable in Web Developing	Intermediate	2017
Knowledgeable in Java, C#	Basic	2017

### TRAININGS, SEMINARS, OR WORKSHOPS ATTENDED

Inclusive Dates	Title of Training, Seminar, or Workshop
2019	Enigma: A Computer Engineering Convention “Cyber Security”
2015	Youth For Life member

Curriculum Vitae of  
**RHEY FRANZ A. REYES**  
**BLK 41 LOT 11 YELLOW BELL ST. PEMBO, MAKATI CITY**  
**Rheyfranz100@gmail.com**  
**09494034583**



#### EDUCATIONAL BACKGROUND

Level	Inclusive Dates	Name of school/ Institution
Tertiary	2019-2023	STI College Global City
Vocational/Technical	2017-2019	STI College Global City
High School	2013-2017	Fort Bonifacio High School
Elementary	2011-2013	Pembo Elementary School
Elementary	2007-2011	Saint Genevieve School of Pateros

#### PROFESSIONAL OR VOLUNTEER EXPERIENCE

Inclusive Dates	Nature of Experience/ Job Title	Name and Address of Company or Organization
2021 - 2022	SAP Business One	STI College Global City
2018 - 2019	Guidance Office Assistant	STI College Global City
2018 – 2018 2022	Registrar Assistant Conquest Crew Cosmatsuri	STI College Global City SMX Convention SMX Convention
2022		

**Listed in reverse chronological order (most recent first).**

#### AFFILIATIONS

Inclusive Dates	Name of Organization	Position

**Listed in reverse chronological order (most recent first).**

#### SKILLS

SKILLS	Level of Competency	Date Acquired
Microsoft POWERPOINT for presentations	Intermediate	2014
Knowledgeable in Html	Intermediate	2017
Knowledgeable in Java/C#	Intermediate	2018

#### TRAININGS, SEMINARS, OR WORKSHOPS ATTENDED

Inclusive Dates	Title of Training, Seminar, or Workshop
2019	Enigma: A Computer Engineering Convention “Cyber Security”