Applied Information Technology And Computer Science Vol. 5 No. 1 (2024) 373-392

AITCS

Homepage: http://publisher.uthm.edu.my/periodicals/index.php/aitcs
e-ISSN :2773-5141

The Development of e-Recipes Mobile Application with Text Recognition: Recipes Collector

Yap Yoke Kei[1], **Noorhaniza Wahid**[1]*

[1] Faculty of Computer Science & Information Technology,
Universiti Tun Hussein Onn Malaysia, Parit Raja, Batu Pahat, 86400, MALAYSIA

DOI: https://doi.org/10.30880/aitcs.2024.05.01.021
Received 29 July 2023; Accepted 08 November 2023; Available online 30 August 2024

Abstract: e-Recipe is a popular electronic recipe library for single-family kitchens. However, many young adults nowadays prefer collecting recipes online through a screenshot, link saving or notebook marking for their reference. Uncategorized recipes collected can be difficult to find and recording recipes physically will also cause time and space wasting. Therefore, a Recipes Collector mobile application with text recognition technology is developed on the Android platform that allows the user to organize, categorise and record recipes in the cloud. The Prototype Methodology is used to develop this application that including All Recipes, Recipe Collection, Text Recognition (OCR) and Buying List modules. The user acceptance test was conducted among the target user aged 18 to 26 using System Usability Scale (SUS). The results show the application achieved 91.5% in functionality testing and 90.75% in usability testing which is acceptable and excellent as a recipe-collecting application.

Keywords: Text Recognition, Mobile Application, e-Recipes, Material Design

1. Introduction

Recipes are lists of measured ingredients and serving sizes that can produce consistent cuisine and assist the cook in estimating the cost of food preparation [1]. Since many reading materials are wisely accessible by large audiences, specifically young adults from written form to digital form, the hardcopy now undergoes the same transformation known as e-Recipes [2]. Due to their needs, these young adults like making some alterations or additions to the recipes they have acquired from different media platforms. However, writing the recipe down on paper takes time, takes up space, and makes it more difficult to find recipes as the collection increases. Thus, the text recognition application with cloud storage is crucial for their convenience.

Text recognition is a type of Optical Character Recognition (OCR) that is commonly used to convert the text in the image which basically capture from the hard documentation and convert into a computer-readable form for modification, format, and copying of the character digitally [3].This technology is wisely implemented in personal management application such as e-recipe collection, receipt tracker and

*Corresponding author: nhaniza@uthm.edu.my

note-taking application that leads to space-saving, time-saving for manual data entering, increase information accessibility and speed of documentation workflow [4].

Meanwhile, taking screenshots, saving recipes links, and writing down recipes from online blogs, hardcopy recipe books, and social media sites like Instagram and Facebook will make it harder to find and organize these recipes. This is because there is no specific feature in the social media or phone gallery to categorize recipes and all the screenshots or saved links with different content will be saved and mixed in a single file. Additionally, recipe-collection notebooks are prone to human error, which could result in the loss of the recipes. Moreover, the e-recipe applications available on the market such as Cook Book [5], Recipes book App [6], and My Recipe Book [7] do not provided text recognition and cloud-data storing features.

Thus, the objectives of this project are to design a Recipes Collector mobile application based on a material design approach, to develop a food recipe-collecting application with text recognition technology on the Android platform, and to perform functional testing and user acceptance test on the developed application to the target users. The application focuses on young adults between 18 to 26 years who are willing to have their recipe collection. Furthermore, the application was developed on the Android platform and based on the prototype methodology.

The Recipes Collector contains 6 modules: Authentication, All Recipes, Recipe Collection, Buying List, Profile and Text Recognition (OCR). The application allows the user to add, view, update and delete their recipe at any time and on any Android device since their recipes are stored in the cloud based on their registered account. Once the user had assigned each recipe based on their category during recipe creation, the recipes could be searched by category and name. The OCR in the application could detect and extract the text captured from the text image that inserts from the Android gallery or camera. The application also could upload any images from the Android mobile phone such as recipe images and user profile pictures.

The rest of the paper covers 4 sections arranged as follows: Section 2 covers domain of study, the technology used, and comparative analysis. Section 3 describes the Prototype Methodology and the output of each phase of the project, Section 4 presents the results and discussion and Section 5 stated the conclusion of the project and future work.

2. Related Work

This section explains the study domain, technology used and the result of the comparative analysis of the similar applications.

2.1 e-Recipes

The recipe determines the nutrient people will get for that whole day's activities [8]. Before the internet become popular, people record the recipe shared by family and friends into a manuscript and gathered it into a collection. The invention of modern printing also allows the publication of recipes in magazines and some people could cut the recipe from the publication to paste into their cookbook. After the invention of the internet, searching and reviewing recipes can be considered a daily activity. Over time, the advent of searchability and hypertext made a personal electronic library called e-Recipes to be created and widely used in every single-family kitchen [9]. e-Recipes application is always more efficient than the traditional cookbook due to the flexibility of the recipe categories organization and space storage [10]. However, if the recipes are collected in the image form, there is a problem for the user copying the recipe text from the image. Therefore, text recognition can be convenient when applied to the developed application.

2.2 Technology Used

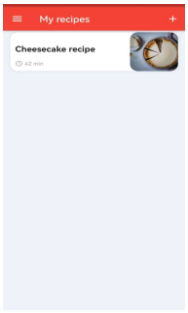
Optical character recognition (OCR) which is also known as text recognition is categorized as online and offline [11]. This paper focuses on the offline OCR. The offline OCR recognizes the item image after capturing by a scanner or camera which is a static image. Nowadays, OCR technology is widely used in various industries and applications such as CamScanner [12] and Adobe Scan [13] as examples of popular offline OCR applications. The banking and financial industry used this technology to extract important information from the cheque to faster the process of invoices, payslips, and loan and mortgage applications. Furthermore, the healthcare industry uses OCR to handle tasks such as lab reports, and doctor notes of the patient record in a short time effectively [14]. With the implementation of OCR in recipe-collecting, it is beneficial for those who prefer to record recipes in written form. This is because the machine-readable text is produced as an output and can be saved through the recipe-collecting mobile application which is much faster than recording recipes on paper-based.

The trend of Android technology is getting higher compared to the IOS technology. This is due to the compatibility of various applications with the Android operating system and the multitasking availability such as running multiple applications on a single screen [15]. Android technology has several advantages such as affordable, less vulnerable to bug and the most popular smartphone operating systems [16] that is more suitable to use for the application development. Meanwhile, Android application applies material design principal as a concept and guideline for the developer to create a consistent appearance, look and feel [17]. This principal concern about the material, motion and graphics as the basic structure, user interface (UI) and great pattern of the user experience (UX) [18]. Thus, the consistency would provide the user with a clear interface for navigation which makes their task convenient and reduces errors [19].

2.3 Comparative Analysis

A comparative table has been made between the existing application such as My Recipes Book [7], Just the Recipe [20], Cook Book [5] and the developed application namely Recipes Collector. Figure 1 shows the main menu interface of the three existing applications. Meanwhile, 9 features will be discussed and compared in Table 1. of the features are authentication, recipes adding method, recipes categories, platform, internet connection, recipes note-taking, text recognition (OCR), buying list and purchase.

Based on Table 1, several strengths and limitations of the developed application are reviewed. Recipe adding, categorizing and note-taking are the common features provided in the e-Recipes applications. Most of the e-Recipes applications use Android platform rather than IOS. In comparison, the existing application name Just the Recipe required an internet connection for the existence of authentication to provide users to store their recipe data in the cloud based on their identity. Moreover, the existing applications do not provide a buying list and text recognition features. To improve user convenience, these two features are provided in the Recipes Collector application. The limitation of the Recipes Collector application it does not support IOS devices and only supports Android version 4.4 and above. Also, the Recipes Collector application does provide user to import recipes from Uniform Resource Locator (URL).



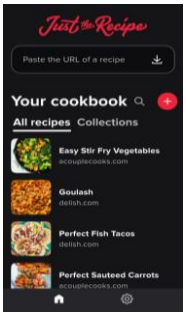


Figure 1(a): My Recipes Book [7] **Figure 1(b): Cook Book [5]** **Figure 1(b): Just the Recipe [20]**

Table 1: Comparison between existing applications and developed application

Features	My Recipes Box	Cook Book	Just the Recipe	Recipes Collector
Authentication	Does not provide user authentication		Necessary for accessibility	
Recipes adding method	Does provide add recipe from scratch	Does provide in-app recipes	Does provide add recipe from scratch or import recipe by recipe URL	Does provide add recipe from scratch
Recipes categories	Does not provide users to categorise their recipes	Does provide recipes categories to lets the user categories their recipes		
Recipes note-taking	Does not provide a note-taking function		Does provide note-taking for each recipe	
Text Recognition (OCR)	Does not provide a user text recognition			Does provide a user text recognition
Buying list	Does not provide the user buying list section			Does provide the user buying list section
Platform	Support Android 4.4 and above	Support Android 4.1 and above	Support Android 4.4 or above and support IOS 12.0 or later	Support Android 4.4 or above
Internet connection	Does not require an internet connection		Internet connections are necessary for authentication purposes	
Purchase	Free of Charge		Need to be purchased	Free of charge

3. Methodology

Prototype model is fast in prototyping reconstruction based on the evaluation and reconfiguration conducted until achieving an appropriate application prototype [21]. Hence, it is used to create the e-Recipe mobile application named Recipes Collector. There are 5 main phases shown in Figure 2 which are planning, analysis, design, implementation and testing. The project will keep iterating until the application has completely met the user requirement through testing.

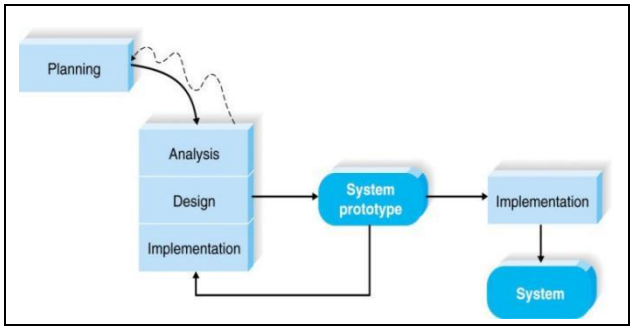


Figure 2: Prototype Model [21]

3.1 Planning

In the planning phase, the project title is confirmed, the project domain is preliminary studied, problem statement, objective, scope, expected result and project significance are identified. Hence, the Recipes Collector is proposed to be created. Besides, the Gantt Chat is created and the prototype model is selected as the project methodology. The feature and functionality of the existing application that is available in the market are studied. Therefore, the advantages, limitations and comparative analysis of the existing application are analyzed.

3.2 Analysis

A set of closed-ended questionnaire surveys was distributed to the 42-target user via Google Forms which was used to identify the user requirement. To summarize, 57.1% of respondents have their own recipe collection based on Figure 7 in Appendix A. Meanwhile, an average of 62.2% of the respondents' faced problems with personal recipe collecting such as hard to copy the recipe from the form of text image and 83.3% of the respondent had desired a text recognition feature to be added on Recipes Collector application. The result of the user requirements is tabulated in Table 2 which based on the data collected from Figure 8 and Figure 9 as attached in Appendix A. Besides, the functional and non-functional requirements of the Recipes Collector were identified and are presented in Table 3 and Table 4, respectively. Moreover, the Unified Modeling Language (UML) Diagrams were plotted due to the system analysis performed. In the UML Diagram, the class diagram represents the overall relationship of the classes in the application, while the use case diagram was used to demonstrate the overall interaction and the component of the Recipes Collector. Hence, the interaction could be further visualized through the activity and sequence diagram. Therefore, the examples of the activity and sequence diagram of Text Recognition (OCR) are shown in Figure 10 and Figure 11, while both use case and class diagrams are shown in Figure 12 and Figure 13 as in Appendix A.

Table 2: User requirement

Participant	Role	User requirement
Young Adults (Target Users from age 18 to 26)	End-user of the Recipes Collector application	● The recipes in the application should be stored in the cloud and retrieved once the user login using their email and password. ● The application should provide of Optical Character Recognition (OCR) for text recognition. ● The application should be able to search for their personal recipe. ● The application should be able to categories their personal recipe. ● The application should be able to add their note to the recipe.

Table 3: Functional requirement

No	Function	Functionalities
1	Login	● This function should allow the old user to access the mobile application by inserting their email and password. ● This function should redirect the user to the home page once the user successfully logs in to the application. ● This function should not allow the user to use an unregistered email. ● This function should not allow the user to log in when empty field exists or is invalid
2	Sign Up	● This function should allow the new user to register an account by filling in their information which are the profile picture, username, email, password and confirm password. ● This function should not allow the user to register with the same existing email. ● This function should not allow the user to register when an empty field exists or is invalid. ● This function should redirect the user to the login page once the user successfully signs up for the application.

No	Function	Functionalities
3	Reset Password	● This function should allow the user to reset their account password by inserting their email. ● This function should send a password reset link to their respective email inbox. ● This function should allow the user to enter their new password at the respective email registered
4	Profile	● This function allows the currently log-in user to view their email, username, and profile image and the total recipe collected from the database.
5	Text Recognition (OCR)	● This function should allow the user to insert the text image in the character form from the gallery or camera for text extraction. ● This function should allow users to edit and copy the extracted text. ● This function should allow the user to clear the extracted text and the inserted text image.
6	Manage Recipe	● This function should retrieve and display all the recipes listed from the database that have already been added by the user. ● This function should allow the user to add a recipe to the database by inserting the recipe image, recipe name, recipe description, recipe categories, time prepared, serving size, ingredient, step preparation, and note. ● This function should allow the user to update the recipe information that was added by the user to the database before ● This function should allow the user to delete the recipe that is stored in the database. ● This function should not allow the user to add or update the recipe when an empty field exists or is invalid. ● This function should display the recipe image, recipe name, number of served and time preparation in the all-recipes list
7	Search Recipe	● This function should display the recipe based on recipe categories. ● This function should display the recipe once the recipe name matches the recipe name inserted in the search box
8	Manage Buying List	● This function should retrieve and display all the buying lists listed from the database that have already been added by the user. ● This function should allow the user to add a buying list to the database by inserting the buying list title and items. ● This function should allow the user to update the buying list information that was added by the user to the database before ● This function should allow the user to delete the buying list that is stored in the database. ● This function should not allow the user to add or update the buying list when an empty field exists or is invalid. ● This function should display the buying lists information such as the title and some items in the unchecked or checked buying lists. ● The function should be able to update the buying list status in the database once the user tick or untick the buying list.

Table 4: Non-functional requirement

No	Requirement	Description
1	Operational	● The application should always connect with the database with the presence of the internet to have a real-time database update
2	Availability	● The application should always be ready and operate 24/7
3	Security	● The application should always require a valid and correct email and password of the user for accessibility. ● The user can only view, update and delete the recipe and buying list that they create by themselves
4	Integrity	● All the data storage is well organized to ensure data readability and not corrupted

3.3 Design

In the design phase, the wireframes were created as the user interface (UI) guideline during the UI design as shown in Table 5. Besides, the system architecture diagram and flowchart were sketch during the system design as shown in Figure 14 and Figure 15 as in Appendix A. Meanwhile, the data dictionary is presented in Table 6 which was designed during the database design.

Table 5: Wireframe

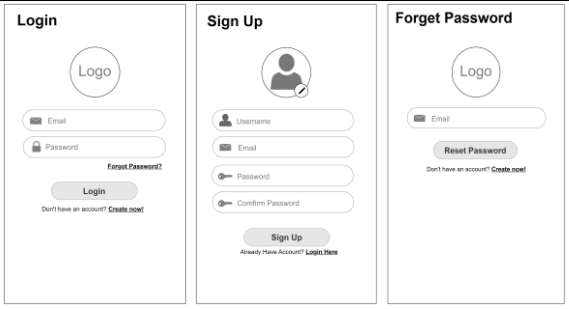
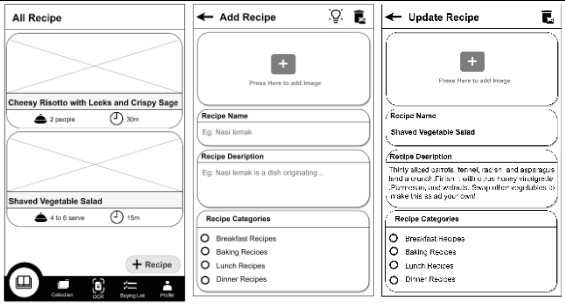
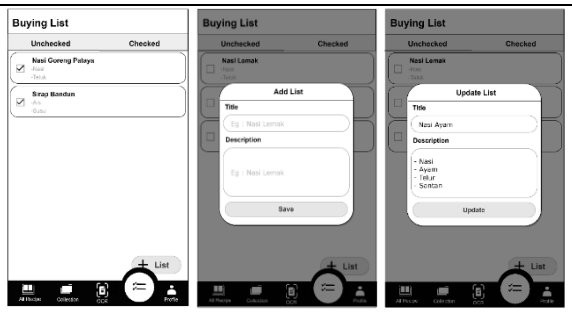
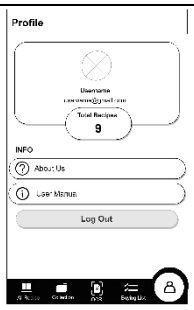
Interface and its description	
	
This is the Authentication module which consists of the Login, Sign Up and Forget Password interface.	This is the All Recipe module which allow the user to view, add, update and deleted their recipes.
	
This is the Recipe Collection module which allows the user to search their recipes by categories or name.	This is the Text Recognition (OCR) module which is used to extract text from a text image inserted from an Android mobile phone gallery or camera.
	
This is the Buying List module which allows the user to view, add, update, delete and categories their buying list.	This is the Profile module which shows the current login user's email, username and total number of recipes collected. Besides, About me and User Manual as application information.

Table 6: Data dictionary

Authentication		
Attribute	Data type	Description
email	string	User's unique email
username	string	User's username
photoUrl	string	User's profile picture
uid	string	User's unique ID
buyingIDs	array	User's buying lists IDs
recipeIDs	array	User's recipes IDs

Recipe		
Attribute	Data type	Description
image	string	User's recipe image
rID	string	User's recipe unique ID
userEmail	string	User's email under the specific recipe ID
recipeName	string	User's recipe name
recipeDesc	string	User's recipe description
recipeServe	string	User's recipe serving size
recipeTimePrepared	string	User's recipe time preparation
categoriesOption	string	User's recipe categories
recipeIngredient	string	User's recipe ingredients
recipeStep	string	User's recipe step
recipeNote	string	User's recipe note
Buying List		
Attribute	Data type	Description
bID	string	User's buying list unique ID
userEmail	string	User's email under the buying list
buyingTitle	string	User's buying list title
buyingDesp	string	User's buying list description
isCheck	string	User's buying list status

3.4 Implementation

The application was developed in this phase which involved the module and database implementation. Table 7 and Table 8 show the code segment of the main module and text recognition module with its corresponding function of the Recipes Collector respectively.

Table 7: Code segment of main module in Recipes Collector

Use case	Function and explanation	Code segment

Manage Recipe	Add recipe:
	The recipe information including the recipe name, recipe description, recipe time preparation, recipe serving size, recipe preparation steps, recipe note, recipe image, recipe categories, recipe ID and the recipe's owner email are created and stored in the recipe data collection in the Firebase after the user pressing the save button. Besides, the recipe ID will be stored under the current login user email in the array form.
		`//add recipe info into firebase`
		`if (_formKey.currentState!.validate()) {`
		`_formKey.currentState!.save();`
		`DocumentReference ref = await db.collection('colrecipes').add({`
		`'recipeName': '${recipemodel.recipeName}',`
		`'recipeDesc': '${recipemodel.recipeDesc}',`
		`'recipeTimePrepared': '${recipemodel.recipeTimePrepared}',`
		`'recipeServe': '${recipemodel.recipeServe}',`
		`'recipeIngredient': '${recipemodel.recipeIngredient}',`
		`'recipeStep': '${recipemodel.recipeStep}',`
		`'recipeNote': '${recipemodel.recipeNote}',`
		`'image': '$fullPathImage',`
		`'categoriesOption': '${recipemodel.categoriesOption}',`
		`'recipeID': "${recipemodel.rID}",`
		`'userEmail': '${global.user.email}'`
		`});`
		`setState(() => recipemodel.rID = ref.documentID);`
		`global.user.addRecipeID(recipemodel.rID);`
		`DocumentReference userRef =`
		`await db.collection("users").document(global.user.email);`
		`userRef.updateData({'recipeIDs': global.user.recipeIDs});`
		`Fluttertoast.showToast(`
		`msg: "Recipe Added",`

	Update recipe:
	The recipe information including the recipe name, recipe description, recipe time preparation, recipe serving size, recipe preparation steps, recipe note, recipe image and recipe categories will be updated in recipe data collection according to the recipe ID in the Firebase after the user pressing the update button.
		`//Update the recipe info into firebase`
		`Firestore.instance`
		`.collection('colrecipes')`
		`.document(widget.ds.documentID)`
		`.updateData({`
		`'recipeName':nameInputController.text,`
		`'recipeDesc': recipeDescriptionController.text,`
		`'recipeTimePrepared': recipeTimePreparedController.text,`
		`'recipeServe': recipeNumberServeController.text,`
		`'recipeIngredient': recipeIngredientController.text,`
		`'recipeStep':recipeStepController.text,`
		`'recipeNote': recipeNoteController.text,`
		`'image': '$fullPathImage',`
		`'categoriesOption':'$recipeCategoriesController'`

Use case	Function and explanation	Code segment

	Delete recipe:
	The recipe ID that is stored under the current login user will be deleted along with the recipe information after the user presses the delete button.
		`// create function for delete one register`
		`void deleteRecipeData(DocumentSnapshot doc) async {`
		`  DocumentReference userDocRef =`
		`  db.collection('users').document(globals.user.email);`
		`  DocumentSnapshot recipeDocRef = await userDocRef.get();`
		`  List userRecipes = recipeDocRef.data['recipeIDs'];`
		`  if (userRecipes.contains(doc.documentID) == true) {`
		`  userDocRef.updateData({`
		`  'recipeIDs': FieldValue.arrayRemove([doc.documentID])`
		`  });`
		`  }`
		`  if (globals.user.recipeIDs.contains(doc.documentID) == true) {`
		`  globals.user.recipeIDs.remove(doc.documentID);`
		`  }`
		`  await db.collection('colrecipes')`
		`  .document(doc.documentID)`
		`  .delete();`
		`  setState(() => recipemodel.rID = null);`
		`  Fluttertoast.showToast(`
		`  msg: "Recipe Deleted",`

Recipe Collection	Search by categories: This is the condition statement set in each of the recipe categories. Therefore, a list of recipes with specific categories will show in this module.	`//If categories is Cake` `if (doc.data["categoriesOption"] == "Cake") {`

	Search by recipe name:
	This is the condition statement set in the recipe search bar. If the entered recipe name matches the recipe name that is present in the Firebase, a list of found recipes will be shown in this module.
		`//If the entered recipe name match with the recipe name in firebase`
		`if (doc.data['recipeName']`
		`.toString()`
		`.toLowerCase()`
		`.startsWith(recipeModel.recipeName.toLowerCase())) {`

Manage Buying List	Add buying list:
	The buying list information including buying list ID, buying list title, buying list description, buying list status and the buying list's owner email are created and stored in the buying list data collection in the Firebase after the user pressing the save button. Besides, the buying list ID will be stored under the current login user email in the array form.
		`// add the buying list information into the firebase`
		`void createData() async {`
		`  if (_formBuyingKey.currentState!.validate()) {`
		`  _formBuyingKey.currentState!.save();`
		`  DocumentReference ref = await db.collection('Buying').add({`
		`  'buyingID': '${buying.bID}',`
		`  'buyingTitle': '${buying.buyingTitle}',`
		`  'buyingDesp': '${buying.buyingDesp}',`
		`  'isCheck': buying.isCheck,`
		`  'userEmail': '${global.user.email}'`
		`  });`
		`  setState(() => buying.bID = ref.documentID);`
		`  global.user.addBuyingID(buying.bID);`
		`  DocumentReference userRef =`
		`  await db.collection("users").document(global.user.email);`
		`  userRef.updateData({'buyingIDs': global.user.buyingIDs});`
		`  Fluttertoast.showToast(msg: "Buying List Added");`
		`  Navigator.of(context).pop();`
		`  }`

	Update buying list:
	The buying list information including the buying list title, buying and list description will be updated in buying list data collection according to the buying list ID in Firebase after the user pressing the update button.
		`// update the buying list information into the firebase`
		`  Firestore.instance`
		`  .collection("Buying")`
		`  .document(widget.ds.documentID)`
		`  .updateData({`
		`  'buyingTitle': titleController.text,`
		`  'buyingDesp': descriptionController.text,`

	Mark buying as checked:
	The buying list status will be changed from unchecked to checked and updated in the buying list collection according to the buying list ID in the Firebase once the checkbox button was clicked.
		`//mark the buying list as checked`
		`onChanged: (bool? value) {`
		`setState(() {`
		`  buying.isCheck = value!;`
		`});`
		`Firestore.instance`
		`.collection("Buying")`
		`.document(doc.documentID)`
		`.setData({`
		`  "buyingTitle": doc.data["buyingTitle"],`
		`  "buyingDesp": doc.data["buyingDesp"],`
		`  "isCheck": _isCheckTrue,`
		`  'userEmail': '${globals.user.email}'`

Text recognition is the core module in Recipes Collector. In order to implement the text recognition technology, a few steps need to be done as the following:

1. Insert the Google ML Kit plugin with version 0.9.10 into the pubspec.yaml file in Visual Studio Code editor as shown in Figure 3.
2. Import the package of Google ML Kit into the Text Recognition dart file.
3. Apply the code segment of Table 8 with the use of StatefulWidget as shown in Figure 4.

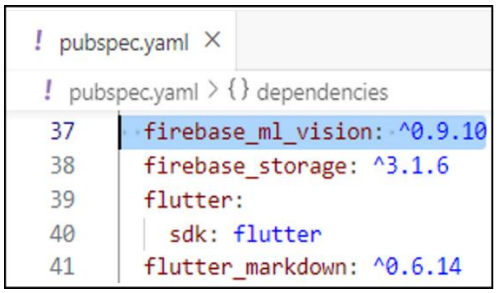


Figure 3: Google ML Kit plugin

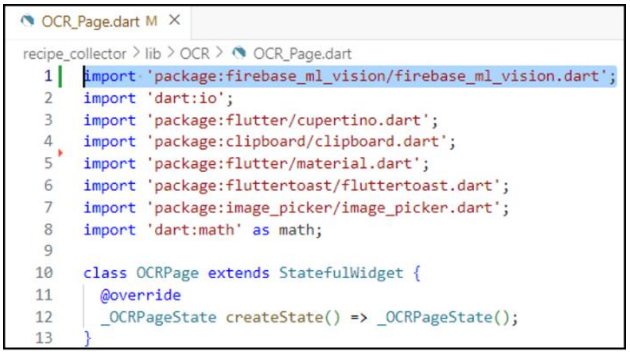


Figure 4: Importation of Google ML Kit package

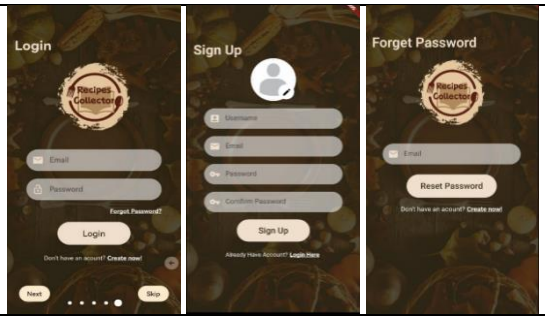
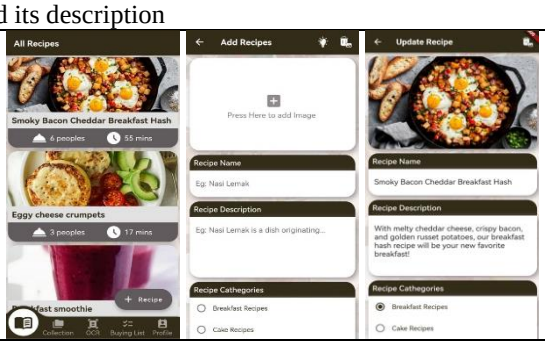
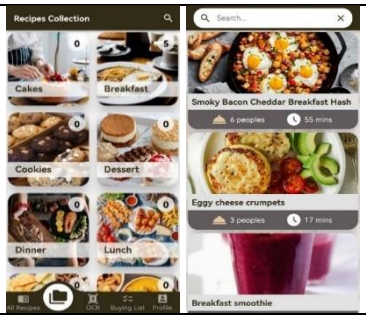
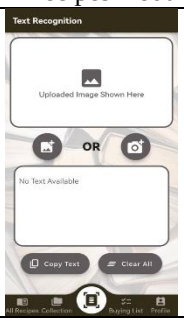
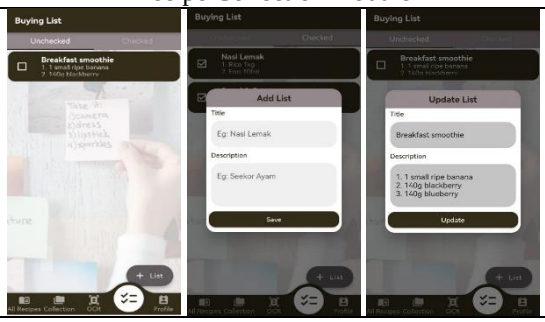
Table 8: Code segment of Text Recognition module in Recipes Collector

Use case	Function and explanation	Code segment

Text Recognition (OCR)	Recognize text:
	After inserting the image, the TextRecognizer will recognise the text presents in the image and extract the text line by line through VisionText. Once the text is extracted, it will be set in the textbox for copy and modification purposes.
		`//recognise the text`
		`Future scanText() async {`
		`_text = '';`
		`if (_image != null) {`
		`final FirebaseVisionImage visionImage =`
		`FirebaseVisionImage.fromFile(File(_image!.path));`
		`final TextRecognizer textRecognizer =`
		`FirebaseVision.instance.textRecognizer();`
		`final VisionText visionText =`
		`await textRecognizer.processImage(visionImage);`
		`for (TextBlock block in visionText.blocks) {`
		`for (TextLine line in block.lines) {`
		`_text += line.text + '\n';`
		`}`
		`}`
		`setText(_text);`
		`} else {`
		`_text = "Nothing to scan...\nPlease select an image";`
		`setText(_text);`

The module interfaces of Recipes Collector shown in Table 9 were implemented based on Material Design approach which include the concern of motion, material design and graphics [18]. The motion concern is to consider the motion moving and module's continuous connection. This to make sure the screen of each module presents a vivid feel with the implementation of a bottom navigation bar when the user wants to select. Meanwhile, all buttons in the application were implemented based on material design to achieve realistic visual cues and a sense of familiarity by casting and rising the button with shadow as well as keeping the consistency of the button size. Based on the graphic concern, a warm color such as brown and creamy neutral were selected as the application's primary color and consistency space scale was implemented in each module to led to the user's eyes pleasure with the relaxing and neutral feeling.

Table 9: Recipes Collector interfaces

Module interface and its description	
	
Authentication module	All Recipes module
	
Recipe Collection module	Text Recognition (OCR) module
	
Buying List module	Profile module

There are three databases implemented in the Firebase which are the User, Recipe and Buying list data collection. In order to implement the Recipe data collection in Firebase as shown in Figure 5, the code segment of Add recipe function in Table 7 needs to be implemented with the save recipe button. Therefore, the Recipe data collection will be created with the execution of the code segment by pressing the save recipe button.

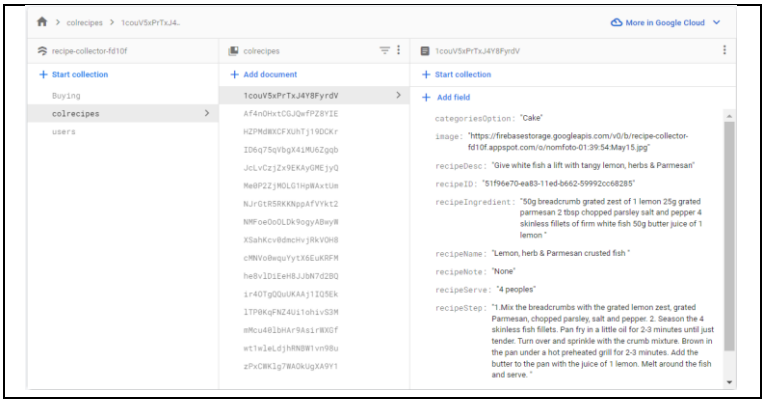


Figure 5: Recipes data collection in Firebase

3.5 Testing

The functional testing and user acceptance test were performed in this phase. In functional testing, all modules in the application were tested and fixed if the bugs are found. Table 10 tabulates the functional testing output, while the user acceptance testing is discussed in Section 4.

Table 10: Functional testing

Authentication module			
Function	Test case	Expected output	Actual output
Login	Incomplete field	Display an error message that reminds the user to complete the empty text field and login rejected	Pass
	Invalid email	Display an error message that reminds users the email does not exist and login rejected	Pass
	Wrong password	Display an error message that reminds the user the password is wrong and login rejected	Pass
	Complete login form	Successfully login and turn in the home page	Pass
Sign Up	Incomplete field	Display an error message that reminds the user to complete the empty text field and registration rejected	Pass
	Used email	Display an error message that reminds the user the email already exists and re-enter the other email, registration rejected	Pass
	Complete sign up Form	Successfully login and turn in the login page for the user to login	Pass
Forget Password	Unique email	Display an error message that reminds the email does not exist and password reset failed	Pass
	Invalid email	Display an error message that reminds the email enter a valid email and password reset failed	Pass
	One-time password reset link	The one-time reset password link directly sends to the user email if the user email exists	Pass
All Recipes module			
Function	Test case	Expected output	Actual output
Display all recipe list	Click the all recipes button	Retrieve and display all the recipes added by the current login user form database only	Pass
Add new recipe	Click the add recipe button	Save the created recipe information into the database under the current login user.	Pass
		Update the total amount of recipes that show in the profile once recipes are added	Pass
	Incomplete field	Display an error message that reminds the user to complete the empty text field and fail to save into the database	Pass
Update recipe	Select a specific recipe from the recipe list to update the recipe information	Update the recipe information in the database under the current login user.	Pass
	Incomplete field	Display an error message that reminds the user to complete the empty text field and fail to update into the database	Pass
Delete recipe	Select a specific recipe from the recipe list to delete the recipe information	Delete the selected recipe that saves under the current login user.	Pass
		Update the total amount of recipes that show in the profile once recipes are deleted	Pass
Recipe Collection module			
Function	Test case	Expected output	Actual output
Search Recipe	Search by category	Display a list of recipes based on their categories	Pass

	Search by name	Display a list of recipes whose names matched the keyword inserted	Pass
		Text Recognition (OCR) module	
Function	Test case	Expected output	Actual output
Recognize text from an image	Insert the screenshot text image from the gallery	Display the extracted text in the text field	Pass
	Insert the text image from the camera	Display the extracted text in the text field	Pass
Modified and copy the extracted text	Edit the extracted text in the text field	The extracted text is editable and arrangeable	Pass
	Copy the text in the text field	The original or modified extracted text in the text field can be copy	Pass
		Buying List module	
Function	Test case	Expected output	Actual output
Display all buying list	Click the unchecked buying list categories	Display all the unchecked buying lists	Pass
	Click the checked buying list categories	Display all the checked buying lists	Pass
Add a new buying list	Click the add buying list button	Save the created buying list information into the database under the current login user.	Pass
Update buying list	Select a specific buying list from all the unchecked or checked buying lists to update the list information	Update the buying list information in the database under the current login user.	Pass
	Mark the unchecked buying list as checked and vice versa	Display the marked buying list on the checked buying list page and vice versa	Pass
Delete buying list	Select a specific buying list from all the unchecked or checked buying lists to delete the buying list information	Delete the selected buying list that saves under the current login user.	Pass
		Profile module	
Function	Test case	Expected output	Actual output
Profile	Show the user profile details	The user's username, profile picture, user's email and the total amount of recipe collected can be shown	Pass

4. Results and Discussion

The user acceptance test aims to evaluate the application's performance under conditions that are comparable to those in which it will eventually be used. In this project, the System Usability Scale (SUS) [22] was performed with a 5-point of Likert Scale to indicate the respondent's degree of agreement with the functionality testing and usability testing. The questionnaire was distributed to 30 respondents with Google Forms as shown in Table 11. In order for the user to test Recipes Collector, the user required a minimum of Android 4.4 version of mobile platform to download Recipes Collector due to the

compatible of text recognition. All questions in the functionality testing and odd-number questions in usability testing are positive tone, while the even-number question in usability testing is negative tone. The scores for each question with positive and negative tone are shown in Table 12 and Table 13, respectively. Based on Table 14 and Table 15, the results show that the developed application has achieved 91.5% in functionality testing and 90.75% in usability testing. Hence the grade of functionality and usability in SUS in Figure 6 confirmed that the Recipes Collector is acceptable and excellent as a recipe-collecting application. At the same time, the Recipes Collector is ready to be used by young adults to solve their recipe-storing and recipe-management problems.

Table 11: Questionnaire of functionality and usability testing

No	Description	Scale					Total participant
		1	2	3	4	5	
Functionality							
1	I successfully created a new recipe and saved it in the application.	0	0	1	10	19	30
2	I can easily add ingredients and their quantities to a recipe.	0	0	0	9	21	30
3	I can straightforward add preparation steps to a recipe, including any necessary details or instructions.	0	0	0	12	18	30
4	I can categorize and organize my recipes into different folders or collections for easier access.	0	0	0	11	19	30
5	The search functionality allows me to find specific recipes quickly based on a keyword.	0	0	1	9	20	30
6	I can edit and update existing recipes easily, including modifying ingredients, serving size, or preparation steps.	0	0	1	6	23	30
7	The application allows me to capture recipe details from printed or handwritten recipes using the text recognition (OCR) feature.	0	0	1	8	21	30
8	I can easily edit or correct any inaccuracies in the recognized text before saving the it as recipe.	0	0	0	11	19	30
9	The application provides an option to create a shopping list from the ingredients of a selected recipe	0	0	0	8	22	30
10	The application provides a visually appealing and user-friendly interface for managing and organizing my collected recipes.	0	0	4	4	22	30
Usability							
1	I think that I would like to use this application frequently.	0	0	4	4	22	30
2	I found the operation of the application was complex and unhelpful.	18	11	1	0	0	30
3	I thought the application was easy to use.	0	0	1	9	20	30
4	I think that I would need the support of a technical person to be able to use this application.	18	9	2	1	0	30
5	I found the various functions in this application were well integrated.	0	0	1	6	23	30
6	I thought there was too much inconsistency in this application.	19	9	2	0	0	30
7	I would imagine that most people would learn to use this application very quickly.	0	0	0	7	23	30
8	I found the application very cumbersome to use	21	8	1	0	0	30
9	I felt very confident using the application.	0	1	1	3	25	30
10	I needed to learn a lot of things before I could get going with this application.	21	6	2	1	0	30

Table 12: Total score of positive tone

Score	0	1	2	3	4
Scale	1	2	3	4	5
Degree of agreement	Strongly disagree	Disagree	Neutral	Agree	Strongly agree

Table 13: Total score of negative tone

Score	4	3	2	1	0
Scale	1	2	3	4	5
Degree of agreement	Strongly disagree	Disagree	Neutral	Agree	Strongly agree

Table 14: Total score of functionality testing

Respondent	Questions										Net score	Percentage (%)
	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10		
R1	4	4	4	4	4	4	4	4	4	4	40	100
R2	4	4	4	4	4	4	4	4	4	4	40	100
R3	4	4	4	4	4	4	4	4	4	4	40	100
R4	3	4	4	3	2	2	2	4	3	3	30	75
R5	2	4	3	3	3	4	3	3	3	3	31	77.5
R6	3	4	3	4	4	4	4	4	4	3	37	92.5
R7	4	4	4	4	4	4	3	3	4	4	38	95
R8	4	4	4	4	4	4	4	4	4	4	40	100
R9	4	4	4	4	4	4	4	4	4	4	40	100
R10	4	4	4	4	4	4	4	4	4	4	40	100
R11	4	4	4	4	4	4	4	4	4	4	40	100
R12	4	4	4	4	4	4	4	4	4	4	40	100
R13	4	4	4	4	4	4	4	4	4	4	40	100
R14	4	4	4	4	4	4	4	4	4	4	40	100
R15	4	4	4	4	4	4	4	4	4	4	40	100
R16	3	4	4	4	4	4	4	4	4	4	39	97.5
R17	4	4	4	4	4	4	4	4	4	4	40	100
R18	4	4	4	4	4	4	4	4	4	4	40	100
R19	3	4	3	4	4	4	4	4	4	4	38	95
R20	3	3	3	3	3	3	3	3	3	3	30	75
R21	3	3	3	3	3	3	3	3	3	3	30	75
R22	3	3	3	4	3	3	3	3	3	3	31	77.5
R23	3	3	3	3	3	3	3	3	3	3	30	75
R24	4	3	3	3	3	4	4	4	4	4	36	90
R25	4	3	3	3	3	3	3	3	3	3	31	77.5
R26	4	4	4	3	3	4	4	3	4	3	36	90
R27	3	3	3	3	4	4	4	3	4	4	35	87.5
R28	4	4	4	4	4	4	4	4	4	4	40	100
R29	3	3	3	3	3	3	3	3	3	3	30	75
R30	4	3	3	3	4	4	4	3	4	4	36	90
Average score (%)												91.5

Table 15: Total score of usability testing

Respondent	Questions										Odd-number's net score	Even-number's net score	Percentage (%)
	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10			
R1	4	4	4	4	4	4	4	4	4	4	20	20	100
R2	4	4	4	4	4	4	4	4	4	4	20	20	100
R3	3	4	4	4	4	3	3	4	4	4	18	19	92.5
R4	2	2	3	3	3	2	4	2	1	4	13	13	65
R5	2	3	3	1	2	3	3	3	2	1	12	11	57.5

Respondent	Questions										Odd-number's net score	Even-number's net score	Percentage (%)
	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10			
R6	2	3	3	4	4	2	3	3	4	2	16	14	75
R7	2	3	3	2	3	4	4	3	4	4	16	16	80
R8	4	4	4	4	4	4	4	4	4	4	20	20	100
R9	4	4	4	4	4	4	4	4	4	4	20	20	100
R10	4	4	4	4	4	4	4	4	4	4	20	20	100
R11	4	4	4	4	4	4	4	4	4	4	20	20	100
R12	4	4	4	4	4	4	4	4	4	4	20	20	100
R13	4	4	4	4	4	4	4	4	4	4	20	20	100
R14	4	4	4	4	4	4	4	4	4	4	20	20	100
R15	4	4	4	4	4	4	4	4	4	4	20	20	100
R16	4	4	4	4	4	4	4	4	4	4	20	20	100
R17	4	4	4	4	4	4	4	4	4	2	20	18	95
R18	4	4	4	4	4	4	4	4	4	4	20	20	100
R19	4	4	4	3	4	4	3	4	3	4	18	19	92.5
R20	3	4	3	3	4	4	4	4	4	3	18	18	90
R21	3	4	4	3	4	4	4	4	4	3	19	18	92.5
R22	4	4	4	4	4	4	4	4	4	4	20	20	100
R23	4	3	4	4	4	4	4	3	4	3	20	17	92.5
R24	4	3	4	3	4	3	4	4	4	4	20	17	92.5
R25	3	3	3	3	3	3	3	3	3	3	15	15	75
R26	4	3	3	4	4	3	4	3	4	4	19	17	90
R27	4	3	2	2	3	3	3	4	4	4	16	16	80
R28	4	3	4	3	4	3	4	3	4	3	20	15	87.5
R29	4	3	3	3	3	3	4	4	4	4	18	17	87.5
R30	4	3	3	3	3	3	3	3	3	3	16	15	77.5
Average score (%)													90.75

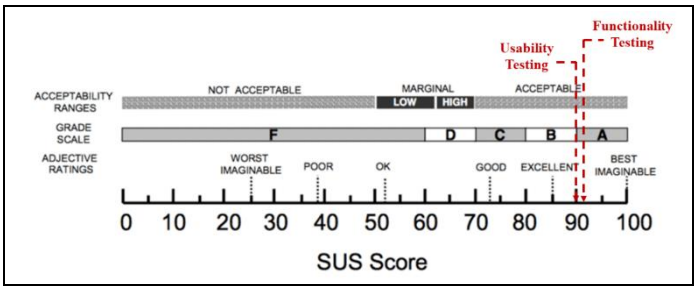


Figure 6: Grade of functionality and usability in SUS [22]

5. Conclusion

According to the findings from the testing phase, Recipes Collector is suitable for young adults to save their recipes. The first objective of the project is fully accomplished by the concern of motion, material design and graphics. The second objective is achieved with the implementation of text recognition technology in Recipes Collector to recognize and extract text image for recipe creation purposes. Lastly, the third objective is achieved by having an overall good performance result in functional testing by the developer and user acceptance testing among the target user after completing the implementation phase. Nevertheless, there are some limitations exists in the application that require some suggestions for future works. The Recipes Collector only allows users to search recipes by name or categories. Furthermore, the Recipes and buying lists cannot be shared with the social media applications such as Facebook, WhatsApp and Messenger. Therefore, some future works are suggested by adding recipe filtration feature to search the recipe based on various criteria, recipe import link feature and provide the user to save the recipe with the recipe ingredient images and preparation step images.

Acknowledgment

The authors would like to thank the Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia for its support.

Appendix A

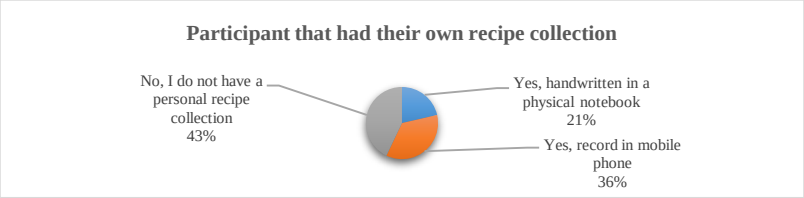


Figure 7: Participant that had their own recipe collection

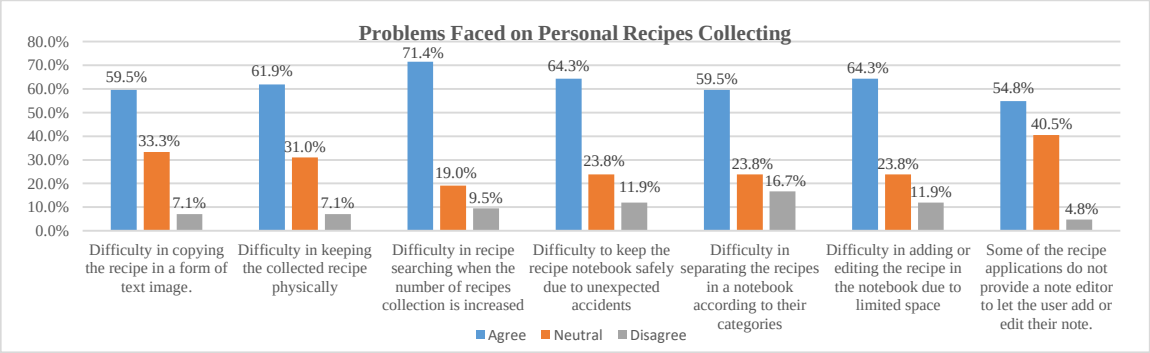


Figure 8: Problem Faced by participant on personal recipe collecting

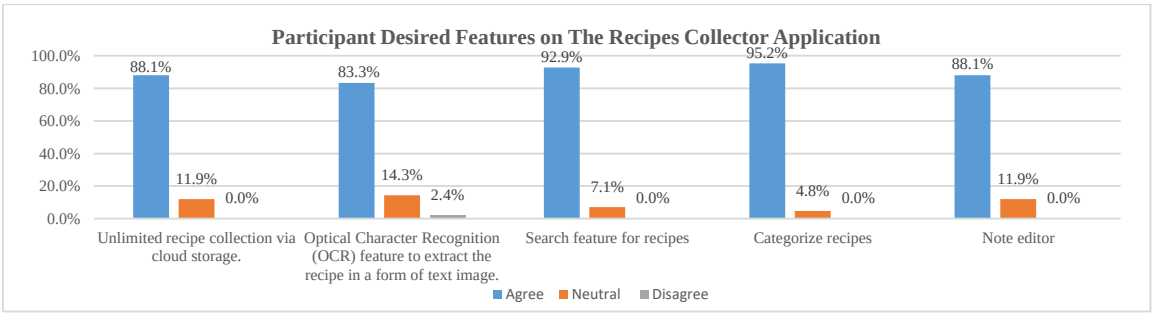


Figure 9: Participant desired on the recipes collector application

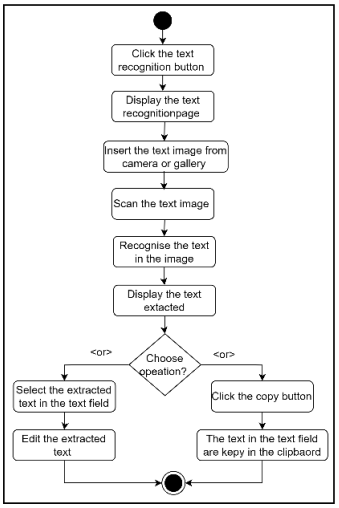


Figure 10: Activity diagram of OCR

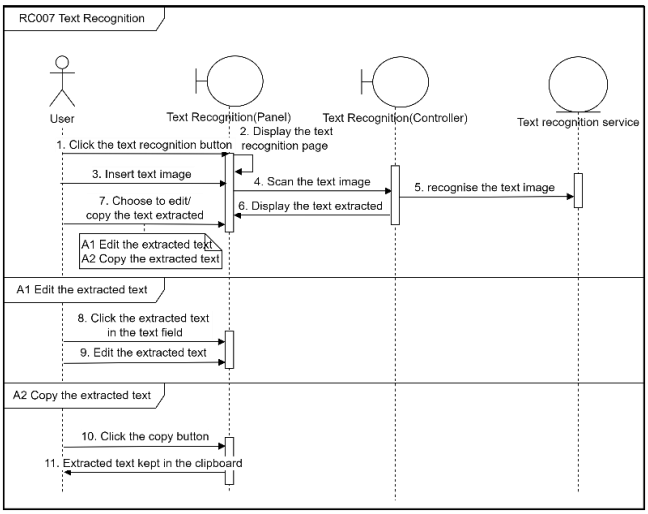


Figure 11: Sequence diagram of OCR

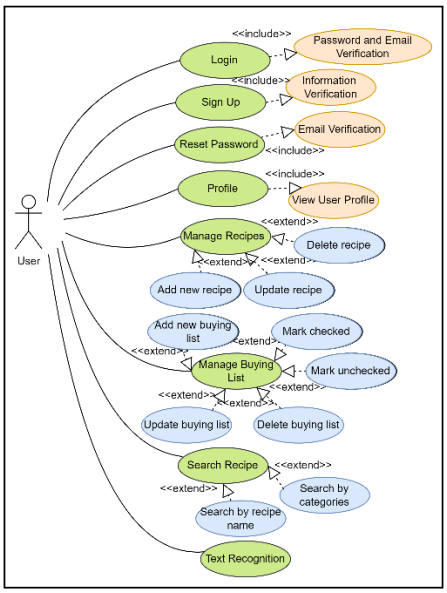


Figure 12: Use case diagram

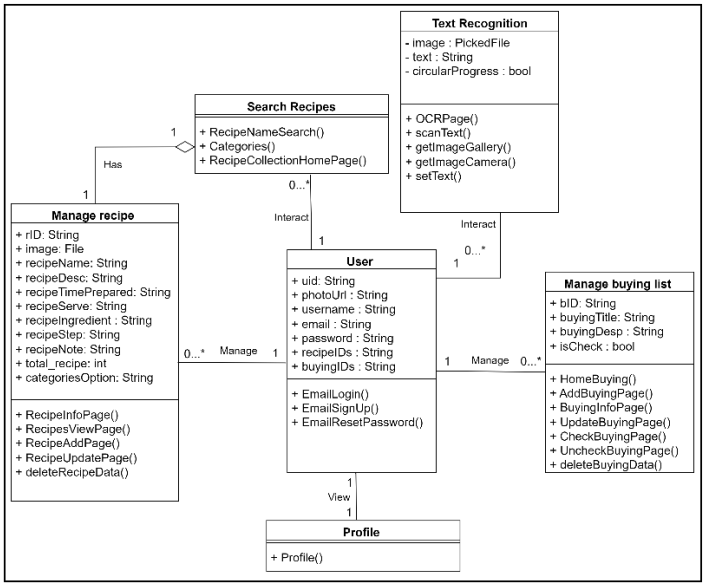


Figure 13: Class diagram

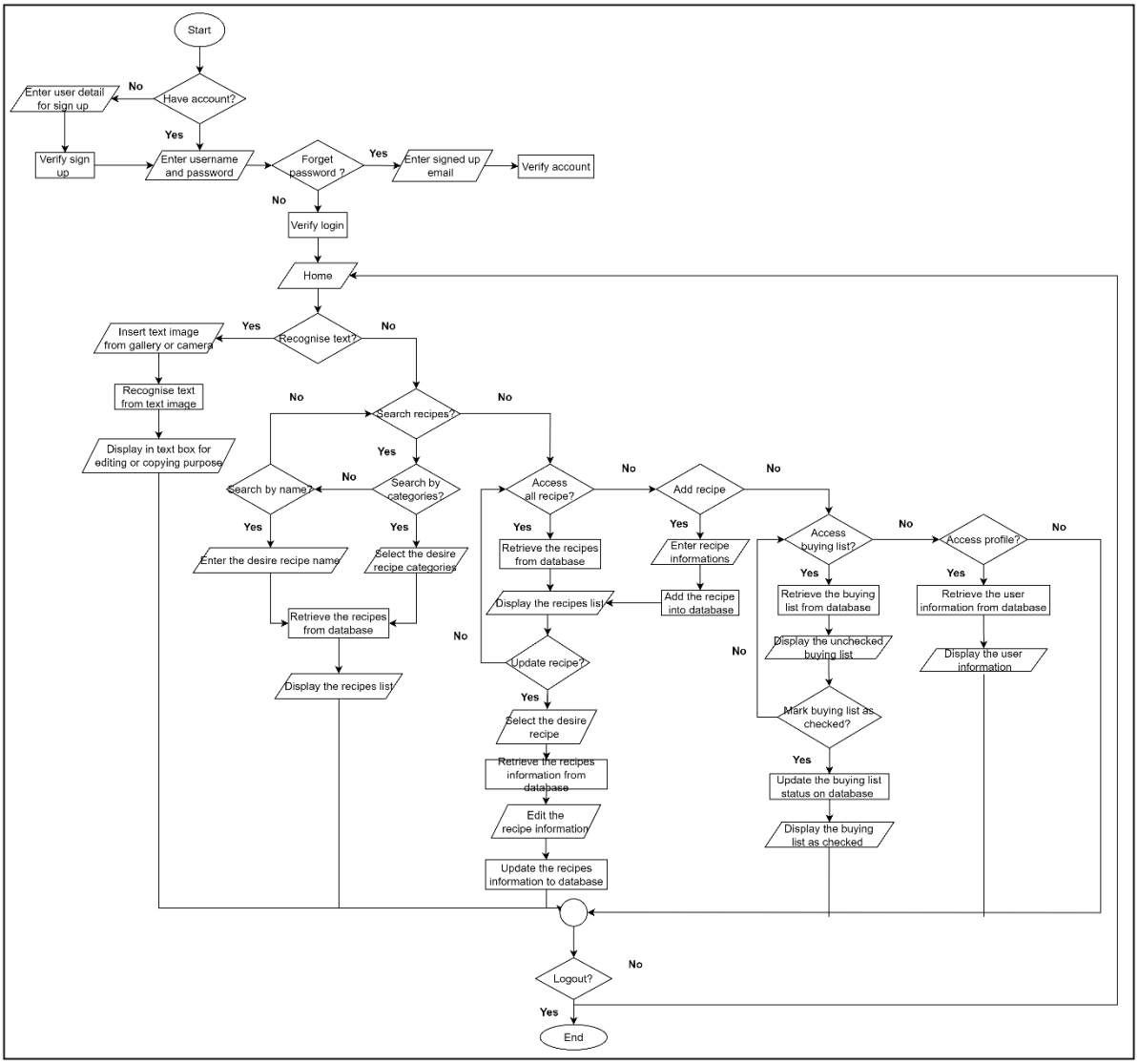


Figure 14: Flowchart of Recipes Collector

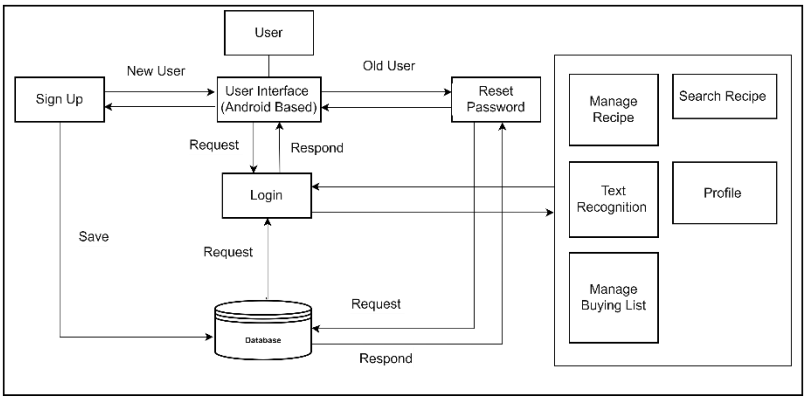


Figure 15: System architecture diagram

References

[1] A. Borghini, "What Is a Recipe?" *Journal of Agricultural and Environmental Ethics*, vol. 28, no. 4, pp. 719–738, Jun. 2015, doi: 10.1007/s10806-015-9556-9

[2] J. Reis, M. Amorim, N. Melão, and P. Matos, "Digital Transformation: A Literature Review and Guidelines for Future Research," Advances in Intelligent Systems and Computing, vol. 745, pp. 411–421, 2018, doi: https://doi.org/10.1007/978-3-319-77703-0_41.

[3] T. Wang et al., "Decoupled Attention Network for Text Recognition," Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, no. 07, pp. 12216–12224, Apr. 2020, doi: https://doi.org/10.1609/aaai.v34i07.6903.

[4] T. C. Wei, U. U. Sheikh, and A. A.-H. A. Rahman, "Improved optical character recognition with deep neural network," in IEEE Xplore, Mar. 2018, pp. 245–249. doi: https://doi.org/10.1109/CSPA.2018.8368720.

[5] (2020) Cook Book – Tasty Recipe – All (Version 1.0.0) [Mobile App]. Retrieved from Google Play Store. https://rb.gy/qu4x1

[6] (2017) Recipe book App (Version 3.10) [Mobile App]. Retrieved from Google Play Store. https://play.google.com/store/apps/details?id=ua.com.meinrezeptbuch.freecookbook&hl=en&gl=US

[7] (2020) My Recipe Book (Version 1.4.6) [Mobile App]. Retrieved from Google Play Store. https://play.google.com/store/apps/details?id=de.bo.recipe.recipes&hl=enUS&gl=US.

[8] F. M. Li, J. Dorst, P. Cederberg, and P. Carrington, "Non-Visual Cooking: Exploring Practices and Challenges of Meal Preparation by People with Visual Impairments," The 23rd International ACM SIGACCESS Conference on Computers and Accessibility, Oct. 2021, doi: https://doi.org/10.1145/3441852.3471215.

[9] H. Notaker, A history of cookbooks: from kitchen to page over seven centuries. Oakland, California: University of California Press, 2017.

[10] D. Bischof, J. Sieck, J. Fransman, C. Kassung, and E. Klingner, "Interactive recipe book - enhance your traditional recipe book," Proceedings of the Second African Conference for Human Computer Interaction: Thriving Communities, Dec. 2018, doi: https://doi.org/10.1145/3283458.3283513.

[11] C. Arindam, "An Efficient End-to-End Neural Model for Handwritten Text Recognition," in *Computation and Language*, Jul. 2018, vol. 2

[12] (2022) CamScanner: Scanner app to scan (Version 6.31.5). [Mobile app]. Retrieved from Google Play Store. https://play.google.com/store/apps/details?id=com.intsig.camscanner&hl=en-NZ

[13] (2022) Adobe Scan: PDF Scanner, OCR (Version 22.12.16). (2022). [Mobile app]. Retrieved from Google Play Store. https://play.google.com/store/apps/details?id=com.adobe.scan.android&hl=en-NZ

[14] S. Manoharan, "A SMART IMAGE PROCESSING ALGORITHM FOR TEXT RECOGNITION, INFORMATION EXTRACTION AND VOCALIZATION FOR THE VISUALLY CHALLENGED," *Journal of Innovative Image Processing*, vol. 1, no. 01, pp. 31–38, Oct. 2019, doi: 10.36548/jiip.2019.1.004.

[15] R. qiya, N. Islam, A. Rai, and N. Khan, "Usability Analysis of Android and iOS Operating Systems," *International Journal of Engineering Trends and Technology*, vol. 68, no. 10, pp. 105–111, Oct. 2020, doi: 10.14445/22315381/ijett-v68i10p218

[16] APPLE, "iPhone," *Apple*, 2022. https://www.apple.com/iphone/

[17] P. Praksh, "A Guide to Material Design, a Modern Software Design Language," *A Guide to Material Design, a Modern Software Design Language*, Apr. 2016.

[18] H. F. El-Sherbiny, "REALISM AND 3D GRAPHICS IN UI DESIGNS AND SOCIAL MEDIA PLATFORMS (TREND OF 2020/21)," Journal of Arts & Architecture Research Studies, vol. 3, no. 5, pp. 138–153, Jun. 2022, doi: https://doi.org/10.47436/jaars.2022.98880.1050.

[19] I. G. Clifton, *Android User Interface Design.* Addison-Wesley Professional, 2015.

[20] (2022) Just the Recipe (Version 1.3.4). [Mobile app]. Retrieved from Google Play Store. https://play.google.com/store/apps/details?id=com.streamline.justtherecipe&hl=en_US&gl=US

[21] K.E. Kendall and J. E. Kendall, *Systems analysis and design*. Essex, England: Pearson Education, 2014.

[22] J. Brooke, "SUS: A 'Quick and Dirty' Usability Scale," Usability Evaluation In Industry, pp. 207–212, Jun. 1996, doi: https://doi.org/10.1201/9781498710411-35.