Software Requirement Specification Document For Food Studio Application

Mariam Hesham, Nour Ahmed, Samiha Hesham, Sandra Fares

December 6, 2020

SRS Version	Date	Reason for Change
1.0	08-October-2020	first version created
2.0	08-November-2020	UI created
2.1	15-November-2020	User interface' images are modified
3.0	06-December-2020	Non functional requirements are updated

Table 1: Document version history

GitHub: https://github.com/SandraFW/MobFoodRecipeApp

Contents

1 Introduction **3**
- 1.1 Purpose of this document . . . 3
- 1.2 Scope of this document . . . 3
- 1.3 Overview . . . 3
- 1.4 Business Context . . . 3

2 General description **4**
- 2.1 Product Functions . . . 4
- 2.2 Similar System Information . . . 4
- 2.3 User Characteristics . . . 5
- 2.4 User Problem Statement . . . 5

3 Functional Requirements **6**

4 Interface Requirements **19**
- 4.1 User Interfaces . . . 19
  - 4.1.1 GUI . . . 19
  - 4.1.2 Profile pages . . . 21

5 Design Constraints **23**

6 Other non-functional attributes **23**
- 6.1 Security . . . 23
- 6.2 Reliability . . . 23
- 6.3 Maintainability . . . 23
- 6.4 Portability . . . 23
- 6.5 Usability . . . 23

7 Project Plan **23**

8 Github Contribution **24**

9 References **24**

1 Introduction

1.1 Purpose of this document

The purpose of this document is to elucidate the requirements to build a food recommendations system. These requirements include adding food recipes, viewing them, and rating them based on their quality. The aim of this project is to offer an easier and more organized way for people to share a variety of food recipes with each other. Depending on the reviews given to each recipe, the application will keep the user updated with the finest recipes and the latest food trends. Moreover, the user can view a dish of his choice and choose his desired recipe based on the reviews shown.

1.2 Scope of this document

The scope of this document is to show the basic outlines of our system requirements in order to understand our system and the issues we may face . Requirements outlined in this document are subject to change.

1.3 Overview

The Food recommendation system is developed to predict the preferences of users to certain recipes. We will design a recommendation system based on a novel model that captures correlations between hidden aspects in reviews and numeric ratings [7]. It is motivated by observing the user's preference against an item is affected by different aspects discussed in reviews.The software would be managed by two the admins and the users. The admin can view all user's details and will be in charge to control everything in the system including banning any user for any misbehave comment or any problem in the system . The user can post a new recipe,rate any new post and comment on them.The system also offers the user to save any post to their favourite panel.

1.4 Business Context

The main goal of the project is to build a well-developed mobile application for managing and organizing food recipe recommendations system. The system aim to reduce time and complexities for the users, by making it much easier to find and save a desired recipe. Moreover the system saves user time by making him able to view the reviews and rates for each recipe to choose a well-cooked food.

2 General description

2.1 Product Functions

1. The system must be fully dynamic,in which the admin is fully controlling the system
2. The users must be able to control their information
3. The user must be able to post the recipe he/she want to share on his/her profile.
4. The user must be able to comment,rate and save any post.
5. The user should be able to report any post.
6. The user must be able to search for any recipe.
7. The admin must be able to view any reports that are added by users against certain comment or post for safety.
8. The system must observe a statistics that is based on the ratings that shows the most trending food recommendations to all the users.

2.2 Similar System Information

There are many similar mobile applications that provide food recommendation as Cookmate [1] , Cook Pade [2] while the most relevant application is[3] Kitchen bowl Recipes Cookbook. This application is available for installation by any user in need through the apple store. It is a platform for people who are passionate about cooking where they can create and share their favorite recipes, follow family ,friends, and the featured cooks, they can also search for dishes and discover new food ideas , just like our application is intended to do.

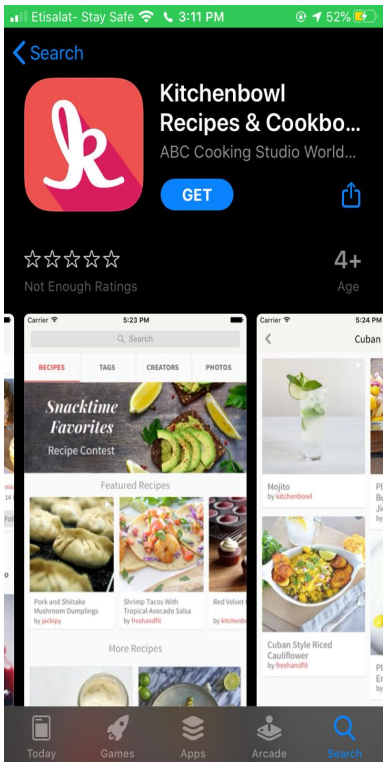

Figure 1: Kitchenbowl Application

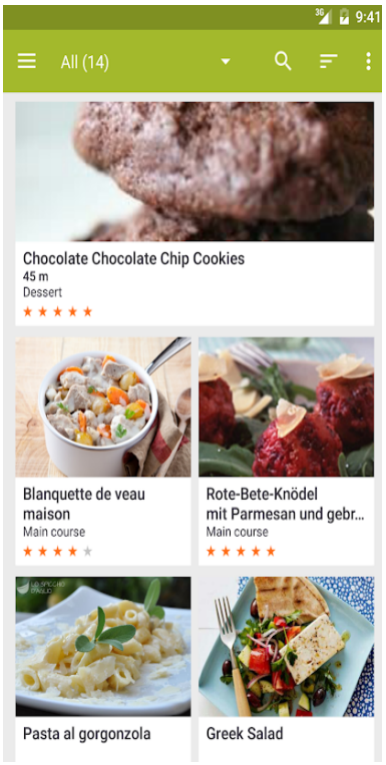

Figure 2: Cookmate Application

2.3 User Characteristics

There are two main users in the system.

Admin: The main role of the admin is controlling the whole project. He is able to view the shared recipes and delete the unwanted posts or comments. Moreover the admin can ban any user after being reported for his unacceptable behavior.

Casual users: Users can post a various recipes. They can search for a desired recipe, rate it and save it.

2.4 User Problem Statement

In various social media platforms, the user finds it difficult to determine if a recipe is worth trying, as there are a lot of recipes placed upon him. Moreover, the recipes and the information relevant to the user are sometimes unclear due to the disorganization of the system. To solve this problem, this application is user friendly and offers the information desired by the user in an effortless and more organized way.

3 Functional Requirements

Table 2

Function:	Check email validation.
ID	FR00
Description	This Function checks if an email address is valid.
Type	Boolean
Input	Email
Action	Checks if the email entered by the user is in the proper format. if it is, the function returns true.
Output	Boolean if email is valid
Precondition	User entered email into the required text field.
Post-condition	return to login function (FR03)
Dependencies	FR03.

Table 3

Function:	Check password validation
ID	FR01
Description	This function checks if the password is valid before inserting it in the database.
Type	Boolean
Input	Password
Action	Checks if the password is in a valid format. If yes, it returns true.
Output	Boolean if password valid
Precondition	User entered password into the required text field
Post-condition	Return to login function (FR03)
Dependencies	FR03

Table 4

Function:	Login.
ID	FR03
Description	This Function is for the user to login into the system using his/her account.
Type	Boolean
Input	Email and Password
Action	Checks if the data are filled correctly. If yes, the function returns true. otherwise, it returns false.
Output	Boolean acceptance login.
Precondition	The user needs to input his email and password into the Text fields. (FR00, FR01)
Post-condition	Redirect to the home page
Dependencies	FR00, FR01

Table 5

Function:	Forget Password
ID	FR04
Description	If the user forgets his password, this function gets generated. It helps the user to insert a new password so that he login successfully again.
Type	Boolean
Input	Email
Action	First, the email is checked using FR00 then if the email exists in the database, the function sends email FR14 with the new password generated.
Output	password updated
Precondition	The user enters his email in the Text field.
Post-condition	Notification is sent to the user with new password.
Dependencies	FR00, FR14

Table 6

Function:	Create User.
ID	FR05
Description	The function uses the input data to insert a new row in users table in the system database
Type	void
Input	user data
Action	checks if the user's email and password are validated. if so, it enters the user record in the database and returns true. else, it returns false and an error message will appear
Output	new user record in database
Precondition	User's information is right and validated
Post-condition	Creating a new row in database table with for a new user.
Dependencies	FR00 , FR01 , FR02

Table 7

Function:	Delete User.
ID	FR06
Description	The Function is to remove the selected user row from users table
Type	Boolean
Input	ID of the user selected to be removed
Action	check if the user id exists in database
Output	A Success Message if User is deleted
Precondition	Administrator chooses which user to be deleted
Post-condition	User Record is deleted from database and database is updated
Dependencies	FR05

Table 8

Function:	Search Recipe.
ID	FR07
Description	The Function returns the recipe searched for by the user
Type	Return Function
Input	Name of the recipe
Action	check if recipe name matches any of records in database. if a recipe exists it shows it to the user. else, returns null
Output	if recipe exists
Precondition	user enters recipe name
Post-condition	recipe is retrieved from database if it is found.
Dependencies	–

Table 9

Function:	Update User
ID	FR08
Description	The Function performs an sql query to users table to update a certain record according to a specific user ID
Type	Boolean
Input	User ID and the desired category to be updated with it's new record
Action	check if those data can be updated
Output	success message or failure message
Precondition	check if user exists in database and has the right to update
Post-condition	records in database are updated
Dependencies	–

Table 10

Function:	Comment or review
ID	FR09
Description	this function is to add a new comment on a certain post to the system in the database
Type	Boolean
Input	User's comment
Action	If the user's comment is created successfully,it gets recorded in the database else,it returns an error message.
Output	comment created successfully.
Precondition	Check if user's comment is. not empty.
Post-condition	The comment has been added successfully.
Dependencies	FR03-FR05

Table 11

Function:	Rate
ID	FR10
Description	This function is to store the rating every user will give to a any post that are ranged from one to five stars and according to this stars the trending statistics will be generated.
Type	Boolean
Input	number of the stars selected.
Action	Storing the rating number to the database.
Output	New rating is added to the database
Precondition	–
Post-condition	–
Dependencies	FR03-FR05

Table 12

Function:	save
ID	FR11
Description	This function is save a certain post to the favourite panel for every user.
Type	Returns array
Input	–
Action	if the post is added successfully it gets recorded in the favorites database table.
Output	new post is added to the favourites.
Precondition	–
Post-condition	check the existence of the saved posts.
Dependencies	FR03-FR05

Table 13

Function:	Report.
ID	FR12
Description	User can report any post or misbehaved comment in any post.
Type	Boolean
Input	report Type ID
Action	If report is updated return true, else return false and show error message.
Output	The report is successfully sent to the admin.
Precondition	There must be at least one comment Type added to the storage and action must be made by the admin.
Post-condition	The record of the user is updated in the database
Dependencies	FR03-FR05

Table 14: -

Function:	view notification.
ID	FR13
Description	viewing all the notification in the database
Type	return array
Input	
Action	fetching all the notifications in the database
Output	array of notifications
Pre condition	check the existence of notifications
Post condition	
Dependencies	FR03

Table 15

Function:	Send Notification.
ID	FR14
Description	The function for sending the notification to the user
Type	Boolean
Input	Notification object
Action	Check for receiving the notificatin
Output	If true string notification, if false there will be an error message
Precondition	The user must be existed in the database
Post-condition	The user will receive the notification, the notification is filled in the database
Dependencies	FR09

Table 16: -

Function:	Create post
ID	FR15
Description	This function is for the user to post a recipe on the application and share it with the other users.
Type	Boolean
Input	User ID. Recipe ingredients and steps to be added to the post
Action	If the post is added successfully ,it gets recorded in the database. else, it returns an error message.
Output	New post is added.
Precondition	Users must login first and create a new post to add the recipe.
Post-condition	Creating a new record in the database table with a new post ID.
Dependencies	FR03

Table 17: -

Function:	Update post
ID	FR16
Description	This function is for the user to modify the recipe and save the new post
Type	Boolean
Input	Post ID. New recipe to be updated.
Action	Check if the recipe data has been updated. If yes, it returns true. else, it returns an error message.
Output	Updated post is added.
Precondition	Check if the post exists in database.
Post-condition	Record in the database is updated.
Dependencies	FR03,FR15

Table 18: -

Function:	Delete post.
ID	FR17
Description	This function is for deleting a specific post whether by the admin or the user.
Type	Boolean
Input	Post ID to be deleted.
Action	Check if the post exists in the database. If yes, removes it. else, returns an error message.
Output	The post is deleted successfully.
Precondition	The post needed to be selected from the list.
Post-condition	Record is deleted from database.
Dependencies	FR03,FR15

Table 19: -

Function:	Delete comment.
ID	FR18
Description	This function is for deleting a specific comment whether by the admin or the user.
Type	Boolean
Input	Comment ID to be deleted.
Action	Check if the comment exists in the database. If yes, removes it. else, returns an error message.
Output	The comment is deleted successfully.
Pre condition	The comment needed to be selected from the list.
Post condition	Record is deleted from database.
Dependncies	FR03,FR09,FR15

Table 20: -

Function:	Logout.
ID	FR19
Description	This function is for all users to logout from their accounts
Type	void
Input	User ID
Action	User logs out from the application.
Output	Logout from the account is done successfully.
Pre condition	The user must login first.
Post condition	Redirect to login page.
Dependncies	FR03

4 Interface Requirements

4.1 User Interfaces

4.1.1 GUI

Figure 3: Introductory page

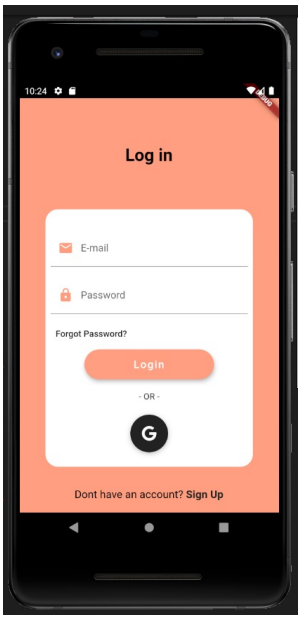

Figure 4: Log in Page[6]

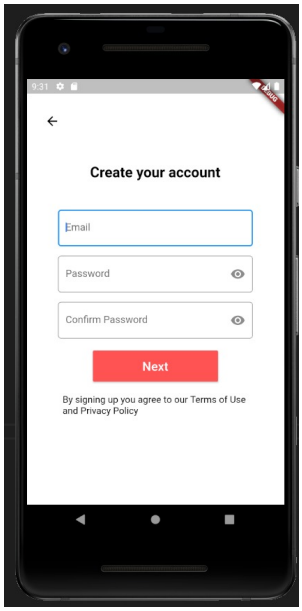

Figure 5: Sign Up

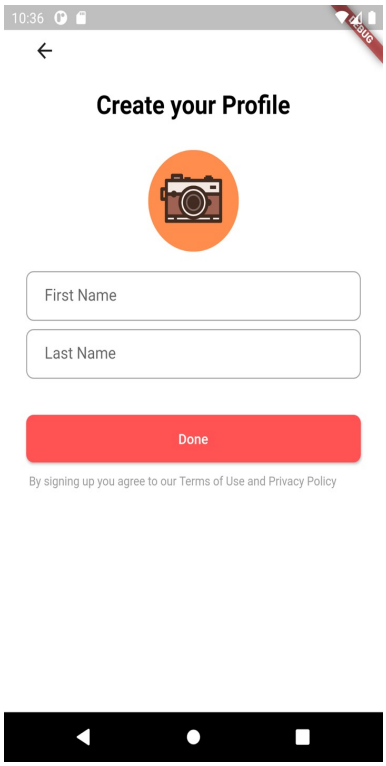

Figure 6: Complete Profile

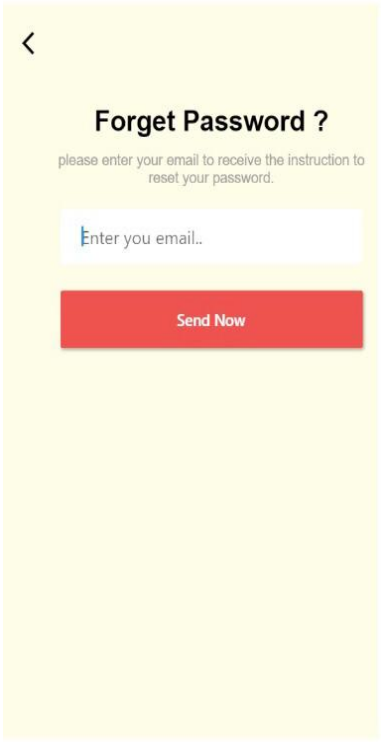

Figure 7: Forget Password

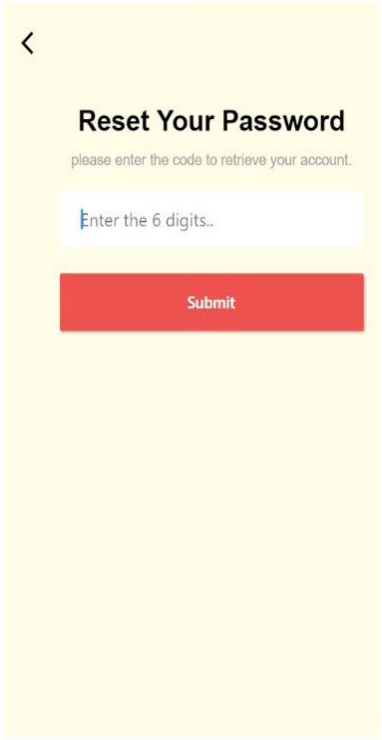

Figure 8: Reset Password

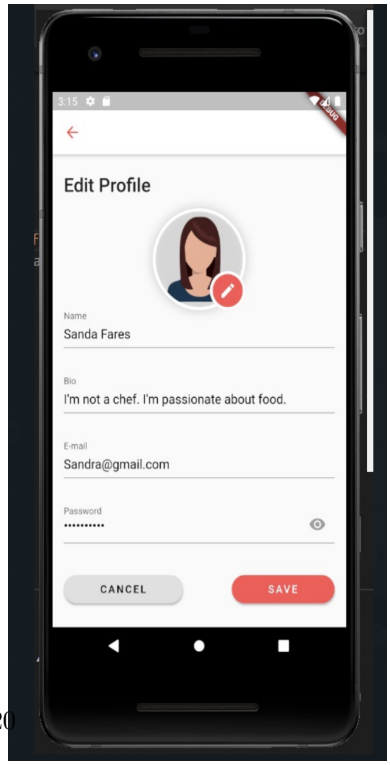

Figure 9: editing profile[5]

4.1.2 Profile pages

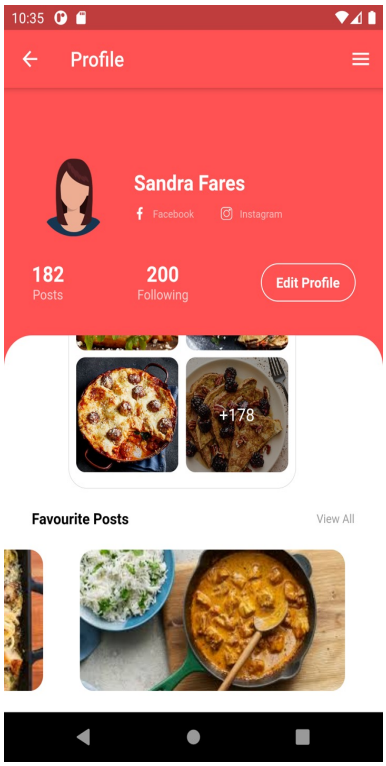

Figure 10: profile page[4]

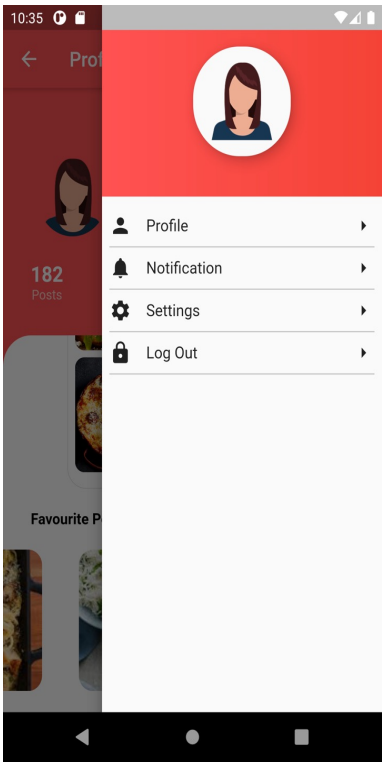

Figure 11: Drawer

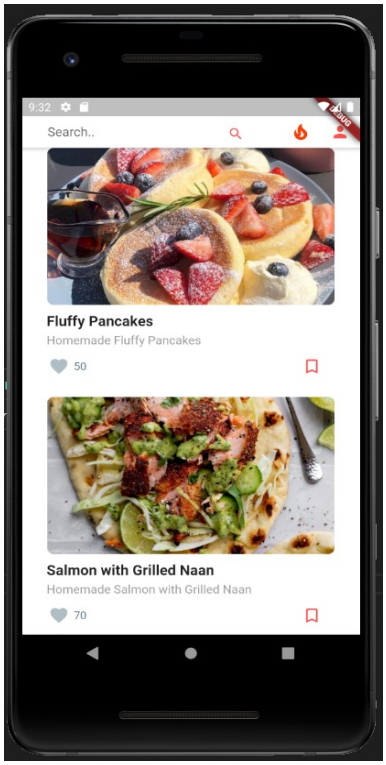

Figure 12: main page

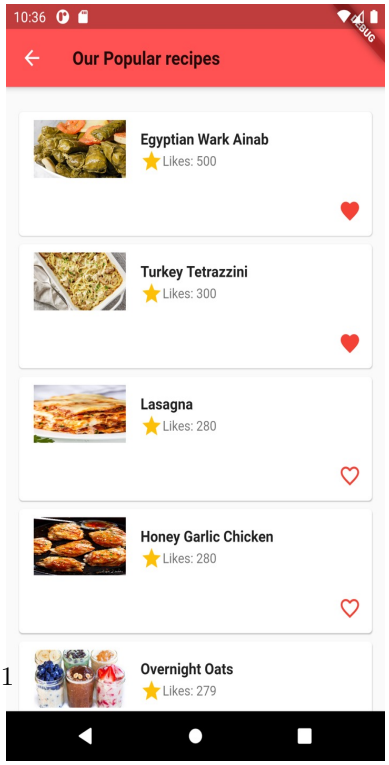

Figure 13: trending page

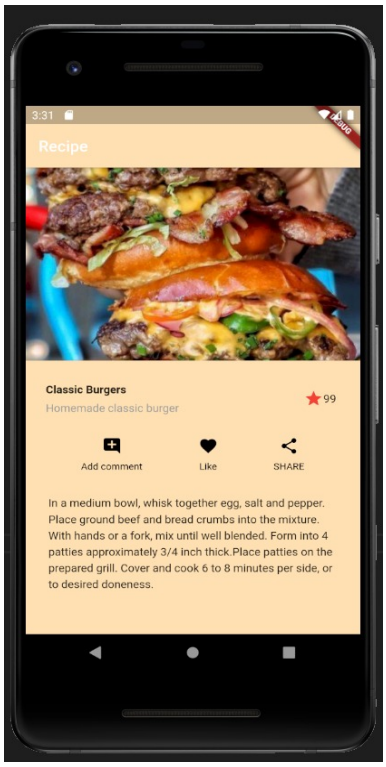

Figure 14: one of the recipes page

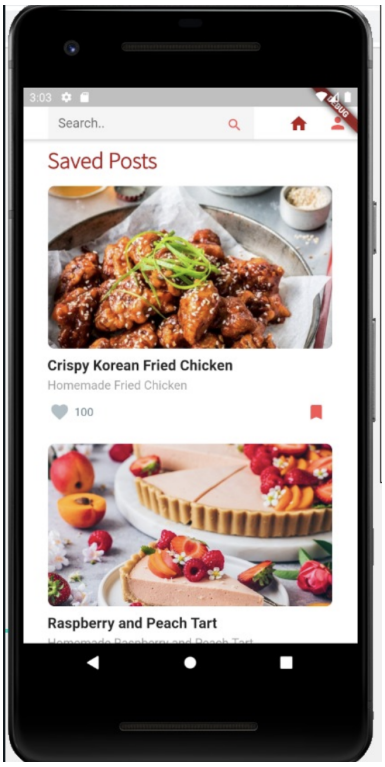

Figure 15: search results

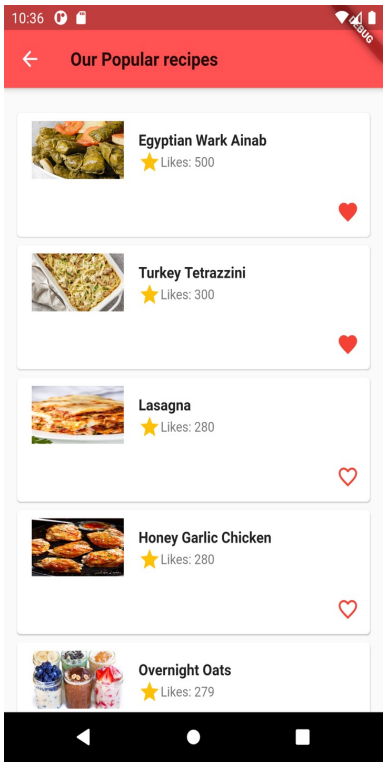

Figure 16: trending page

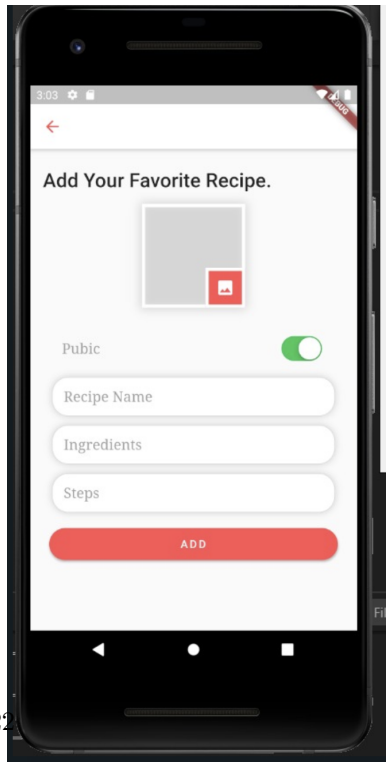

Figure 17: adding new recipe page

5 Design Constraints

The system should be adaptable to be vied from different kinds of devices either tablets, laptops... etc. Also we had a constrain with which is the internet speed it varies from one user to another.

6 Other non-functional attributes

6.1 Security

This system is provided with authentication without which no user can pass. So only the legitimate users are allowed to use the application. If the legitimate user's share the authentication information then the system is open to outsiders.

6.2 Reliability

Proper validations for user inputs will be applied to avoid incorrect storage of records or any invalid inputs.

6.3 Maintainability

The software system should be designed to be easily maintained and improved by software developers so that helps a lot in maintaining the software system, and minimize the amount of changes that will be done in the code

6.4 Portability

This system is implemented using flutter cross platform that can be installed on any smart phone supporting android and IOS software.

6.5 Usability

Food studio application will be user friendly and easy to understand.The system consists if flexible and usable User interface.

7 Project Plan

Start Date	End Date	Task
06-November-2020	17-November-2020	Main page created
06-November-2020	17-November-2020	Creating Login,signUp,forget password,reset password pages
10-November-2020	17-November-2020	Creating One recipe,create recipe,search results pages
17-November-2020	05-December-2020	Creating User profile,drawer,trending,logout pages
10-December-2020	31-December-2020	Creating database and linking it to front end
11-December-2020	02-January-2021	Creating and handling backend functionality

Table 21: Project plan

8 Github Contribution

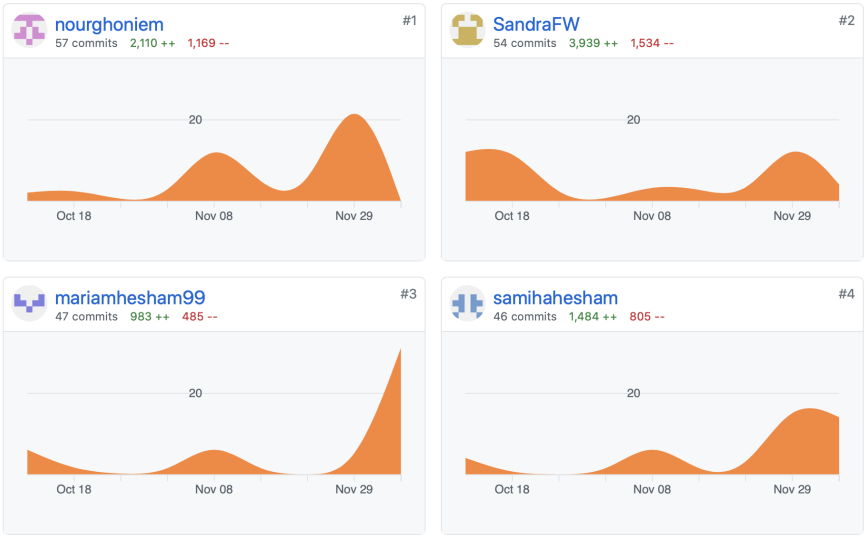

Figure 18: Github Contribution

9 References

References

[1] "Cookmate (formerly my cookbook) - recipe manager," 2010. [Online]. Available: https://play.google.com/store/apps/details?id=fr.cookbookhl=engl=US

[2] "Cookpad - create your own recipes," 2015. [Online]. Available: https://play.google.com/store/apps/details?id=com.mufumbo.android.recipe.searchhl=engl=US

[3] "Kitchenbowl recipes cookbook," 2017. [Online]. Available: https://apps.apple.com/eg/app/kitchenbowl-recipes-cookbook/id831037164

[4] "Profile page template," 2019. [Online]. Available: https://github.com/slackvishal/flutter$_t raveler_p rofile_a pp$

[5] "Creating settings page template," 2020. [Online]. Available: https://github.com/codeWithSilver/settings$_u i/blob/master/lib/pages/edit_p rofile.dart$

[6] "Login page template," 2020. [Online]. Available: https://github.com/abhishekmahajan98/Flutter$_S imple_L ogin_P age_U I_T utorial/tree/master/lib$

[7] P. C. R. Z. Yifan Gao, Wenzhe Yu, "A restaurant recommendation system by analyzing ratings and aspects in reviews," Apr 2015. [Online]. Available: https://www.researchgate.net/publication/312829358$_{A R} estaurant_R ecommendation_S ystem_b y_A nalyzing_R atin$