



## Practice on software engineering

---

# Recipe sharing app

---

**Instructor:**  
*Assoc. Prof. Quan Thanh Tho*  
*Master. Mai Duc Trung*

### **MEMBERS:**

*Tau Nhật Quang*      - 1852698  
*Lâm Trịnh Long*      - 1811044

Ho Chi Minh City, April, 2021

# Table of contents:

[I. Introduction](#)

[II. Functional requirements](#)

[III. Non-functional requirements](#)

[V. Use-cases:](#)

[VI. Interface mock-ups:](#)

[VII. Implementation view](#)

[VIII. Deployment view](#)

[IX. Class diagrams:](#)

[X. Sequence diagrams:](#)

[Conclusion](#)

## Work assignment:

Front-end designs and development: Tau Nhật Quang

Back-end designs and development: Lâm Trịnh Long

## I. Introduction

Our software aims to create a community of food lovers, passionate chefs or even culinary experts. A community to share cooking knowledge, tips, tricks or secrets.

Mobile app providing a platform for online communities to share their cooking methods, rate, comment and save each others' recipes, when they are using a registered account. Hosting a wide variety of recipes and interactions between users.

Guest accounts could also look at recipes, search for recipes.

On those recipes, users can also share their experience and preferences, essentially capable of blogging their life experiences.

Users would also be able to save posts, adding them to their own collection for viewing at a later time.

## II. Functional requirements

1. Create accounts, keep profile on each registered account
2. Guest system so people without registered account can:

- Search uploaded recipes
- View recipes and blog post

### Registered accounts can:

1. Upload and share recipes
2. Browse uploaded recipes
3. Search function to look up recipes:
  - User enter a word or phrase
  - Return recipes with name that include the searched word

3. Rate recipes, on a scale of 1-5 stars
4. Users' profiling to keep track of their favourites recipes:
  - Option to favourites and save other people recipes
  - Create a whole collection of recipes for other users to see, or for you to visit the recipe in later time
5. User profiling:
  - User shall be able to register an account onto the system, customize their display name, avatar.
  - Uses this profile system to track which recipe belongs to whom, or to allow each user to save different recipes into a personal collection.

### **III. Non-functional requirements**

1. Service stays up 24/7
2. Responsive:
  - load time should be fast, maximum of 2 seconds load time
  - Support 1000 simultaneous access
3. Multi platform: supports both android and IOS
4. Smooth and good looking user interface

## V. Use-cases:

General use-case diagram:

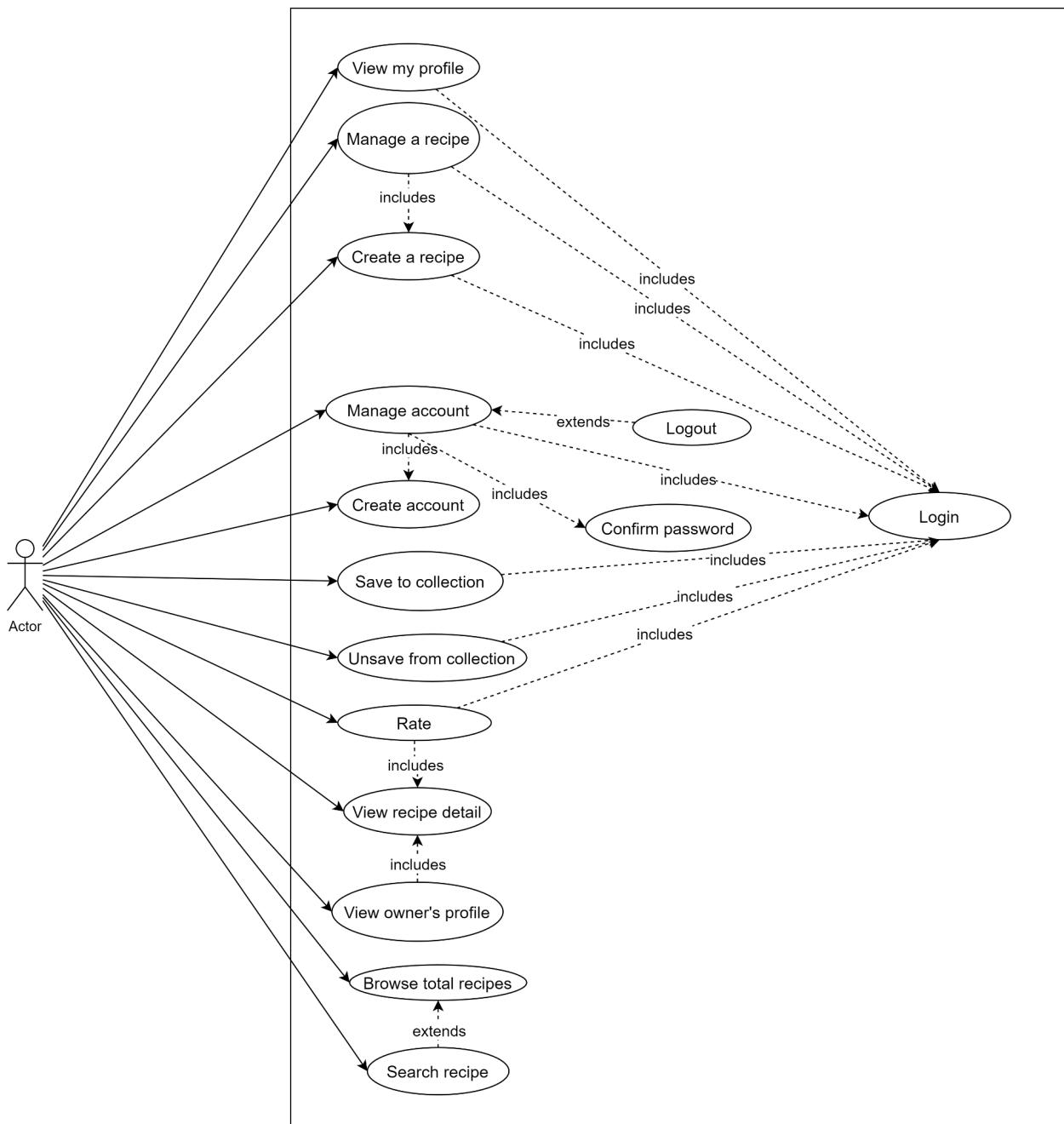


Figure 1: General use-case diagram

### 1. View user profile

Name	View user profile
Actor	User

Description	Display user profile
Precondition	User sign in successfully with account to gain access to the app
Normal flow	<ol style="list-style-type: none"> <li>1. User fill in username and password in the login screen</li> <li>2. Successful access to the application with user's account</li> <li>3. User is at the home screen</li> <li>4. User swipes to profile screen</li> <li>5. User is able to view all profile information</li> </ol>
Exception	<p>Exception on step 1:</p> <ul style="list-style-type: none"> <li>- User has not yet signed in to the application with the user's account or signed in as a guest.</li> </ul>
Alternative flow	<p>Alternatively on step 3:</p> <ul style="list-style-type: none"> <li>- 3a: User accesses a recipe that belongs to the user's account.</li> <li>- 4a: User clicks on the avatar of the recipe's creator in the recipe detail screen.</li> </ul>

## 2. Manage a recipe

Name	Manage a recipe
Actor	User
Description	Let the user modify or remove the recipes created
Precondition	User sign in successfully to gain access to the app There are recipes already created on the profile
Normal flow	<ol style="list-style-type: none"> <li>1. User fill in username and password in the login screen</li> <li>2. Successful access to the application with user's account</li> <li>3. User is at the home screen</li> <li>4. User swipes to profile screen</li> <li>5. User clicks on delete icon of a recipe that needs removed</li> <li>6. Successfully removed the chosen recipe</li> </ol>
Exception	<p>Exception on step 1:</p> <ul style="list-style-type: none"> <li>- User has not yet signed in to the application with the user's account or signed in as a guest.</li> </ul> <p>Exception on step 4:</p> <ul style="list-style-type: none"> <li>- User has no recipe created.</li> </ul>
Alternative flow	<p>Alternatively on step 5:</p> <ul style="list-style-type: none"> <li>- 5a. User clicks on the modify icon of a recipe that needs modified.</li> <li>- 6a. User refills all the information in the introduction page.</li> <li>- 7a. User swipes to the ingredient page.</li> <li>- 8a. User refills all the information in the ingredient page.</li> <li>- 9a. User swipes to the step page.</li> <li>- 10a. User refills all the information in the step page.</li> </ul>

- |  |  |
|--|--|
|  | <ul style="list-style-type: none"> <li>- 11a. User clicks 'Save recipe' to save the modification.</li> <li>- 12a. Successfully updated the chosen recipe.</li> </ul> |
|--|--|

### 3. Create a recipe

Name	Create a recipe
Actor	User
Description	Let the user create a new recipe.
Precondition	User sign in successfully to gain access to the app
Normal flow	<ol style="list-style-type: none"> <li>1. User fill in username and password in the login screen</li> <li>2. Successful access to the application with user's account</li> <li>3. User is at the home screen</li> <li>4. User swipes to profile screen</li> <li>5. User clicks to the add icon button</li> <li>6. User fills all the information in the introduction page.</li> <li>7. User swipes to the ingredient page.</li> <li>8. User fills all the information in the ingredient page.</li> <li>9. User swipes to the step page.</li> <li>10. User fills all the information in the step page.</li> <li>11. User clicks 'Save recipe' to save the modification.</li> <li>12. Successfully created a new recipe.</li> </ol>
Exception	<p>Exception on step 1:</p> <ul style="list-style-type: none"> <li>- User has not yet signed in to the application with the user's account or signed in as a guest.</li> </ul> <p>Exception on step 6,7,8:</p> <ul style="list-style-type: none"> <li>- User has not yet filled in all the information.</li> </ul>
Alternative flow	No alternatives

### 4. Manage an account

Name	Manage an account
Actor	User
Description	Let the user modify or remove the account created or sign out of the current account.
Precondition	User sign in successfully to gain access to the app
Normal flow	<ol style="list-style-type: none"> <li>1. User fill in username and password in the login screen</li> <li>2. Successful access to the application with user's account</li> <li>3. User is at the home screen</li> <li>4. User swipes to profile screen.</li> <li>5. User clicks to the edit profile button</li> </ol>

	<p>6. User clicks the sign out button.      7. The screen gets back to the login screen.</p>
Exception	<p>Exception on step 1:        - User has not yet signed in to the application with the user's account or signed in as a guest.        Exception on step 7a,7b,7c,7d:        - User entered the wrong password.</p>
Alternative flow	<p>Alternatively on step 6:        - 6a. User clicks password bar to edit password        - 6b. User clicks display name bar to edit display name        - 6c. User clicks the delete account button to remove the account.        - 6d. User clicks to the avatar frame to change the avatar image.        - 7a/7b/7c/7d. User confirms the correct password to complete the modification.</p>

## 5. Create an account

Name	Create an account
Actor	User
Description	Let the user create an account.
Precondition	App is available on the device.
Normal flow	<ol style="list-style-type: none"> <li>1. User clicks the sign up button at the login screen.</li> <li>2. User fills all the fields in the sign up form.</li> <li>3. Then the user must click the sign up button on the sign up screen to complete creating an account.</li> </ol>
Exception	<p>Exception on step 2:        - User entered the wrong format username        Exception on step 3:        - User clicked the sign up button while not completing the sign up form.</p>
Alternative flow	No alternatives

## 6. Save recipes to collection

Name	Save recipes into collection
Actor	User
Description	Let the user save favourite recipes into the personal collection
Precondition	User sign in successfully to gain access to the app Exist the recipes to save.

Normal flow	<ol style="list-style-type: none"> <li>1. User fill in username and password in the login screen</li> <li>2. Successful access to the application with user's account</li> <li>3. User is at the home screen</li> <li>4. User clicks on the favourite recipe card.</li> <li>5. User clicks on the bookmark icon.</li> </ol>
Exception	No exception
Alternative flow	Alternatively on step 4: - User clicks on the bookmark icon directly on the recipe card.

## 7. Unsave recipes from collection

Name	Unsave recipes from collection
Actor	Registered user
Description	Let the user remove the recipes out of the collection.
Precondition	User sign in successfully to gain access to the app Exist recipes in the collection.
Normal flow	<ol style="list-style-type: none"> <li>1. User fill in username and password in the login screen</li> <li>2. Successful access to the application with user's account</li> <li>3. User is at the home screen</li> <li>4. User swipes to profile screen</li> <li>5. User swipes to collection screen</li> <li>6. User clicks on the recipe card in the profile collection screen</li> <li>7. User clicks on the bookmark icon in the recipe detail screen.</li> </ol>
Exception	No exception
Alternative flow	Alternatively on step 4: - 4a. User clicks on the bookmark icon directly on the recipe card at the home screen. Alternatively on step 6: - 6a. User clicks on the bookmark icon directly on the recipe card at the profile collection screen.

## 8. View recipe detail

Name	View recipe detail
Actor	User
Description	Let the user view all the information of a recipe
Precondition	Exist the recipe to view.
Normal flow	<ol style="list-style-type: none"> <li>1. User fill in username and password in the login screen</li> <li>2. Successful access to the application with user's account</li> <li>3. User is at the home screen</li> </ol>

	4. User clicks on a recipe card to view the introduction screen. 5. Swipe to view the ingredient screen. 6. Swipe to view the instruction screen.
Exception	No exception
Alternative flow	Alternatively on step 1: - 1a. User login as guest - 2a. User is at the home screen. - 3a. User clicks on a recipe card to view the introduction screen. - 4a. Swipe to view the ingredient screen. - 5a. Swipe to view the instruction screen.

## 9. Browse total recipes

Name	Save recipes into collection
Actor	User
Description	Let the user browse all the recipes available in the application.
Precondition	App is available on the device
Normal flow	1. User fill in username and password in the login screen 2. Successful access to the application with user's account 3. User scrolls down and up at the home screen to view.
Exception	No exception.
Alternative flow	Alternatively on step 1: - 1a. User login as guest - 2a. User scrolls down and up at the home screen to view.

## 10. Rate a recipe

Name	Rate a recipe
Actor	User
Description	Let the user rate a specific recipe.
Precondition	User sign in successfully to gain access to the app Exist the recipe to rate.
Normal flow	1. User fill in username and password in the login screen 2. Successful access to the application with user's account 3. User is at the home screen 4. User clicks on a recipe card to access a specific recipe. 5. User clicks on the rate button. 6. User rates a recipe by swipe gesture.
Exception	No exception.

Alternative flow	No alternatives.
------------------	------------------

## 11. Search a recipe

Name	Save a recipe
Actor	User
Description	Let the user search a recipe based on keywords.
Precondition	App is available on the device
Normal flow	<ol style="list-style-type: none"> <li>1. User fill in username and password in the login screen</li> <li>2. Successful access to the application with user's account</li> <li>3. User is at the home screen</li> <li>4. User clicks on the recipe search bar</li> <li>5. User enters keywords to search.</li> </ol>
Exception	The keywords are not related to the recipe.
Alternative flow	Alternatively on step 1: <ul style="list-style-type: none"> <li>- 1a. User login as guest</li> <li>- 2a. User clicks on the recipe search bar</li> <li>- 3a. User enters keywords to search.</li> </ul>

## 12. View other profiles

Name	View other profiles
Actor	User
Description	Let the user view the owner of a specific recipe.
Precondition	App is available on the device Exist the recipe and the owner relationship.
Normal flow	<ol style="list-style-type: none"> <li>1. User fill in username and password in the login screen</li> <li>2. Successful access to the application with user's account</li> <li>3. User clicks on a recipe card to access a specific recipe.</li> <li>4. User clicks on the avatar of the creator.</li> </ol>
Exception	No exception.
Alternative flow	Alternatively on step 1: <ul style="list-style-type: none"> <li>- 1a. User login as guest</li> <li>- 2a. User clicks on a recipe card to access a specific recipe.</li> <li>- 3a. User clicks on the avatar of the creator.</li> </ul>

## VI. Interface mock-ups:

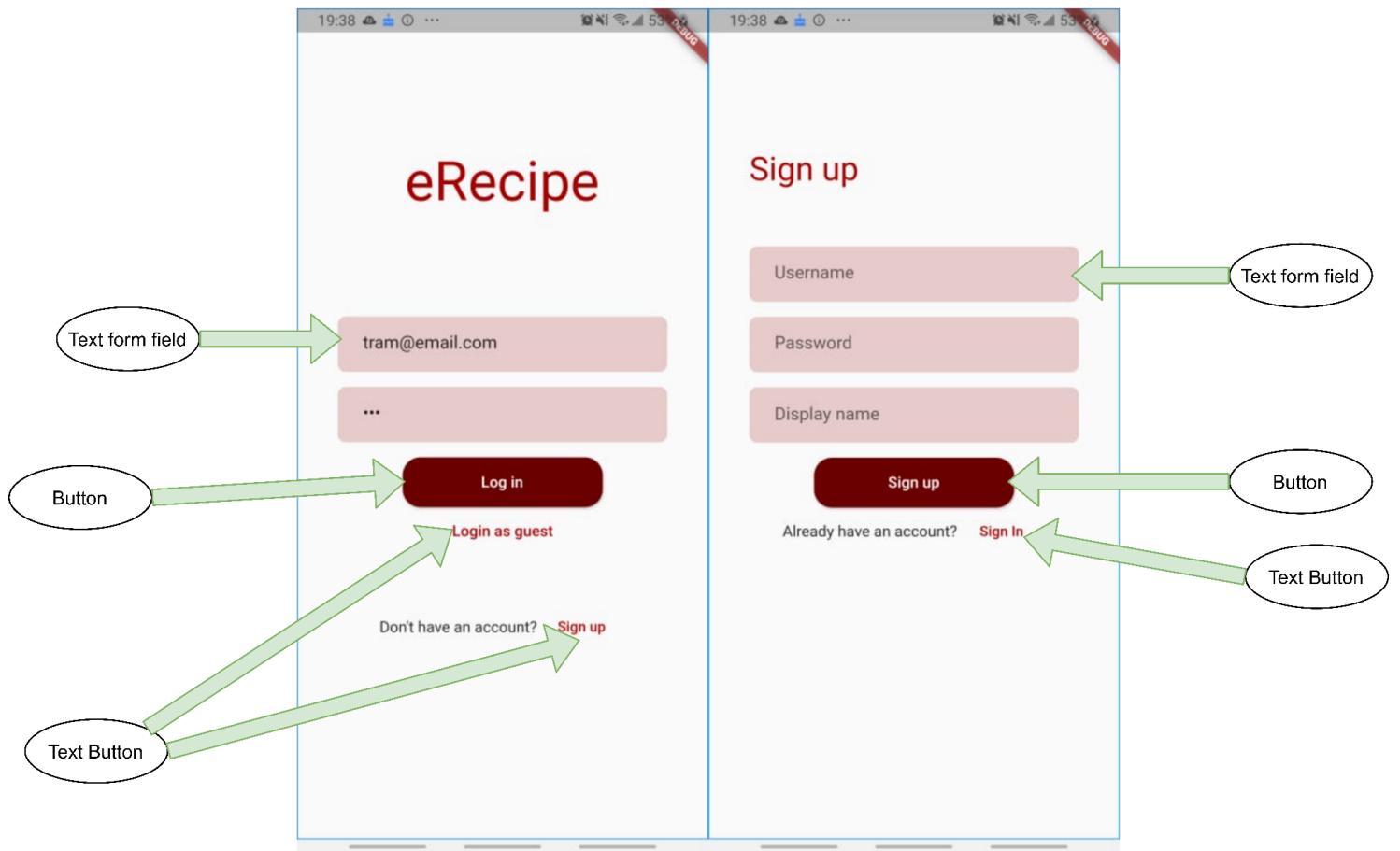


Figure 2: User interface on login and sign up (**available feature**)

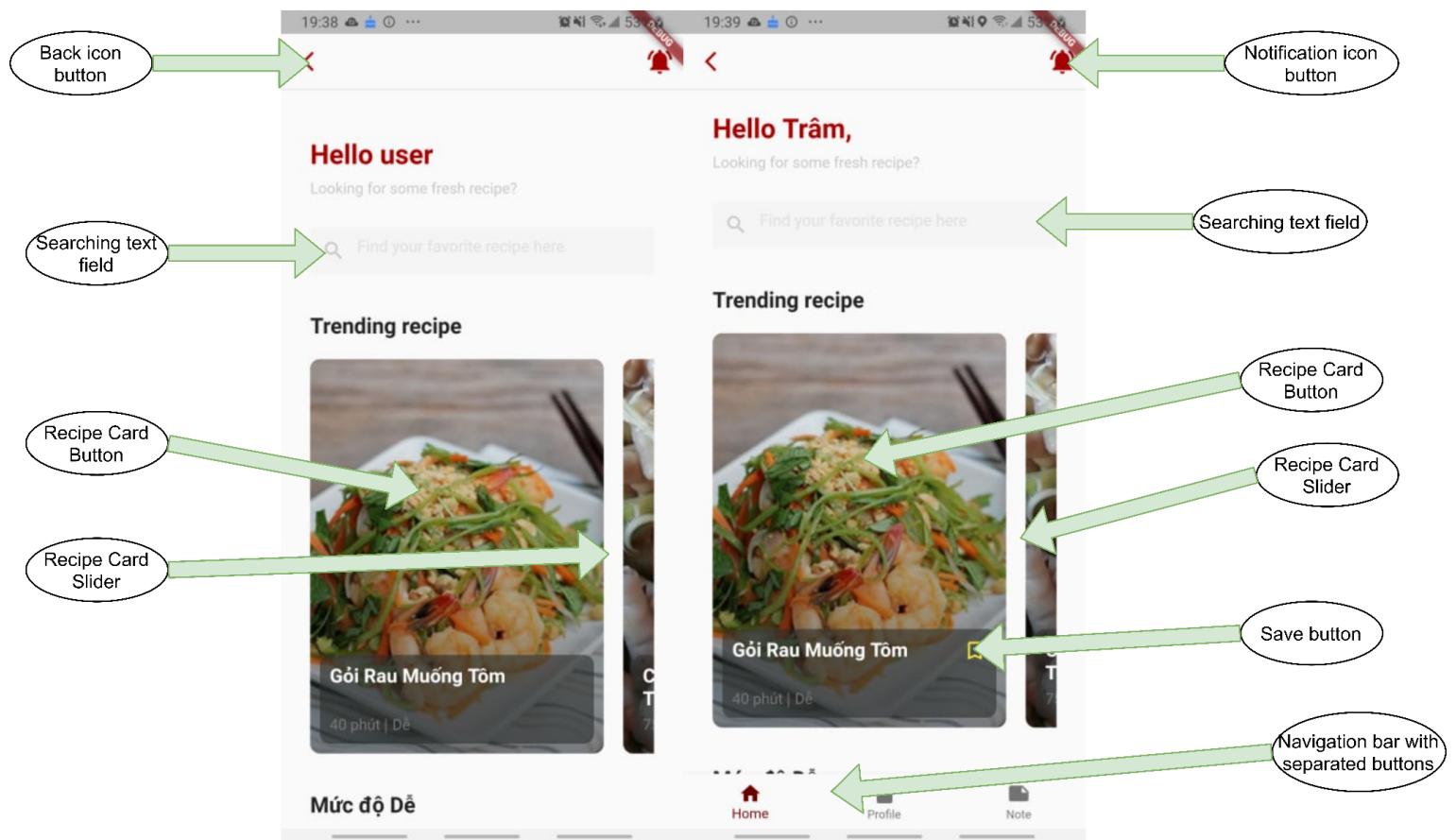


Figure 3: User interface on home screen as guest (left) and registered user (right). **(available feature)**

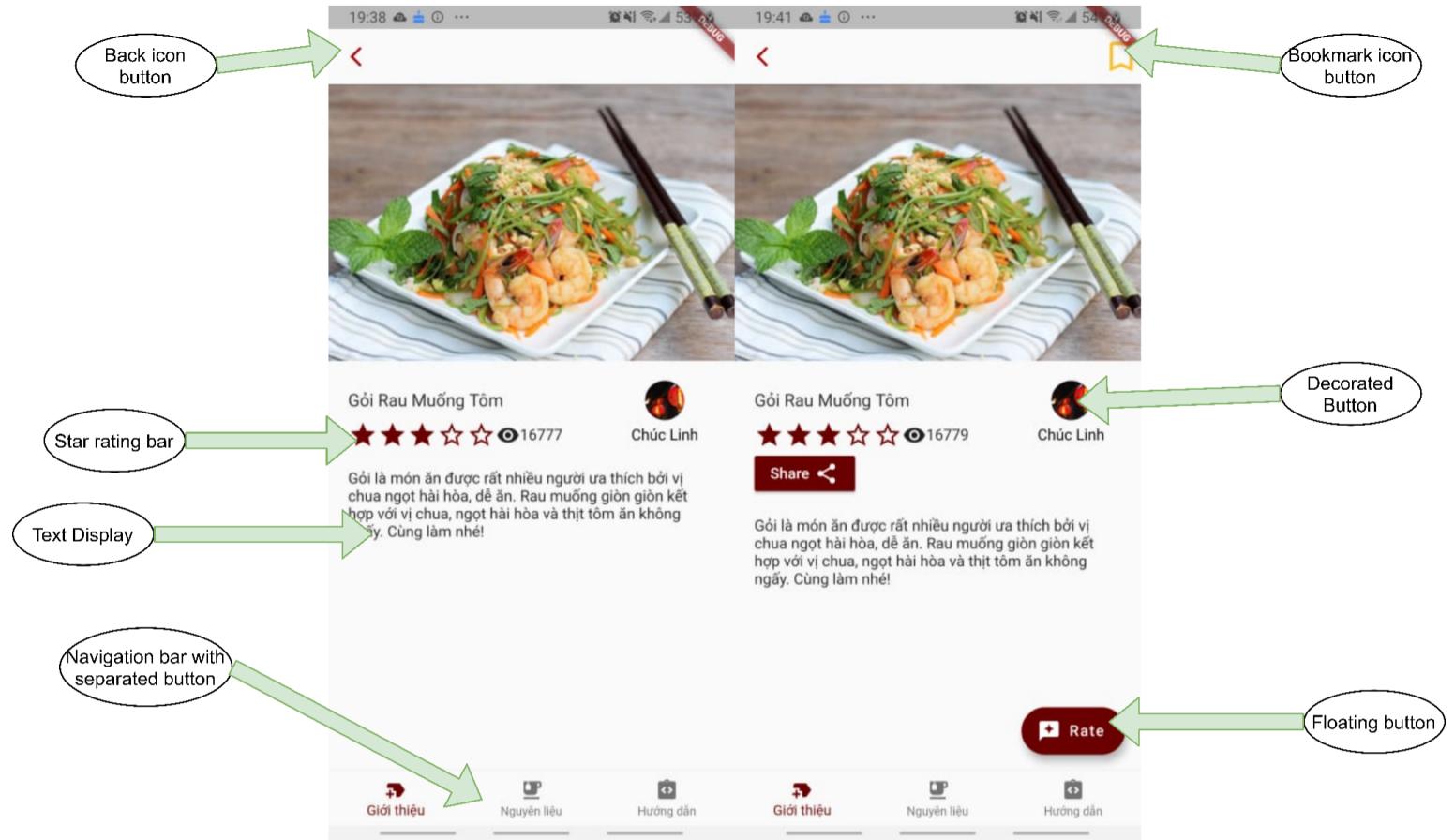


Figure 4: User interface on viewing tab introduction in a specific recipe as guest (left) and registered user. (**available feature**)

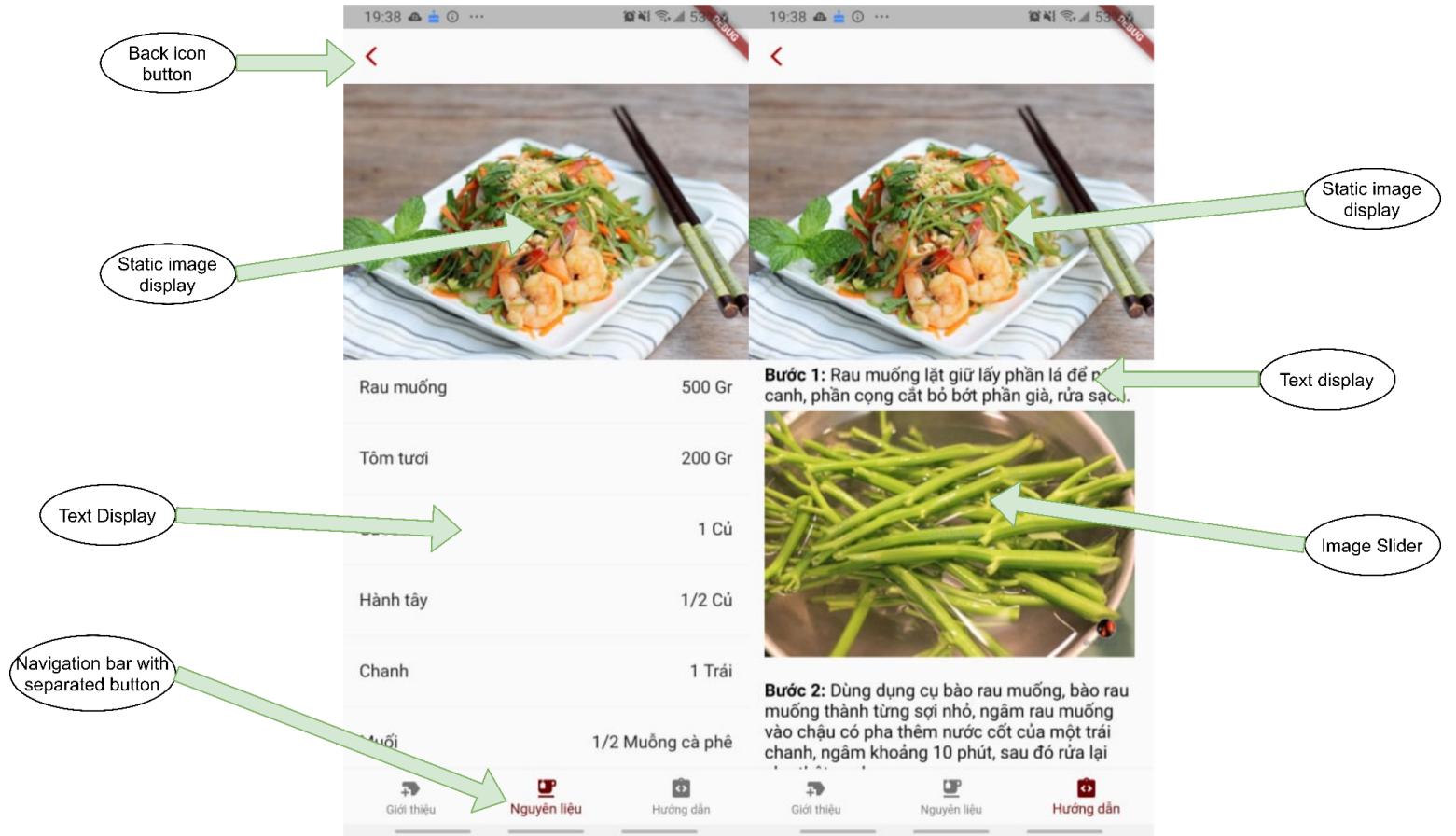


Figure 5: User interface on viewing ingredients (left) and instructions(right) in a specific recipe.  
**(available feature)**

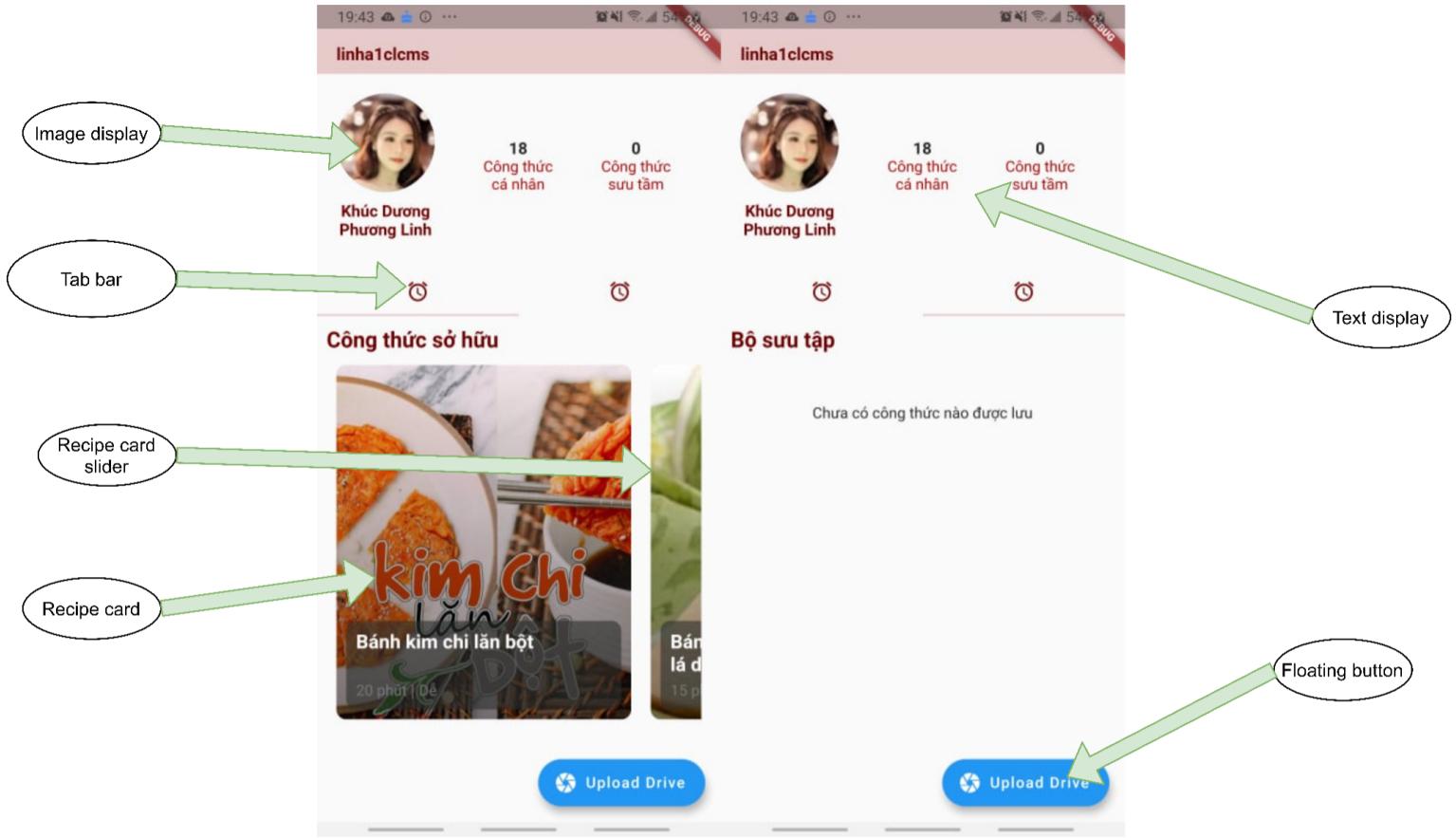


Figure 6: User interface on viewing created recipes (left) and collection (right) in a specific user's profile. **(available feature)**

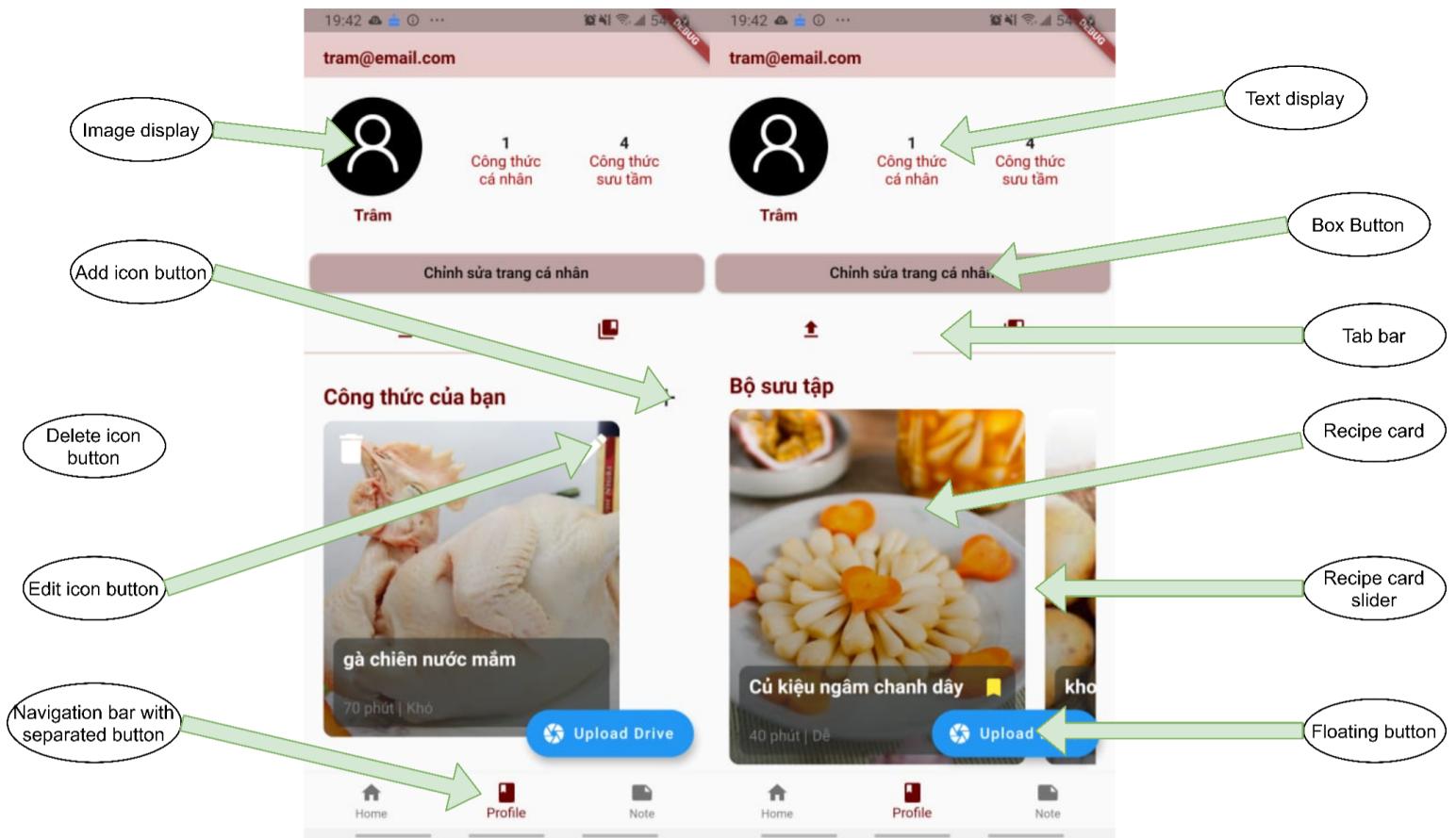


Figure 7: User interface on viewing created recipes (left) and collection (right) in a main user's profile. **(available feature)**

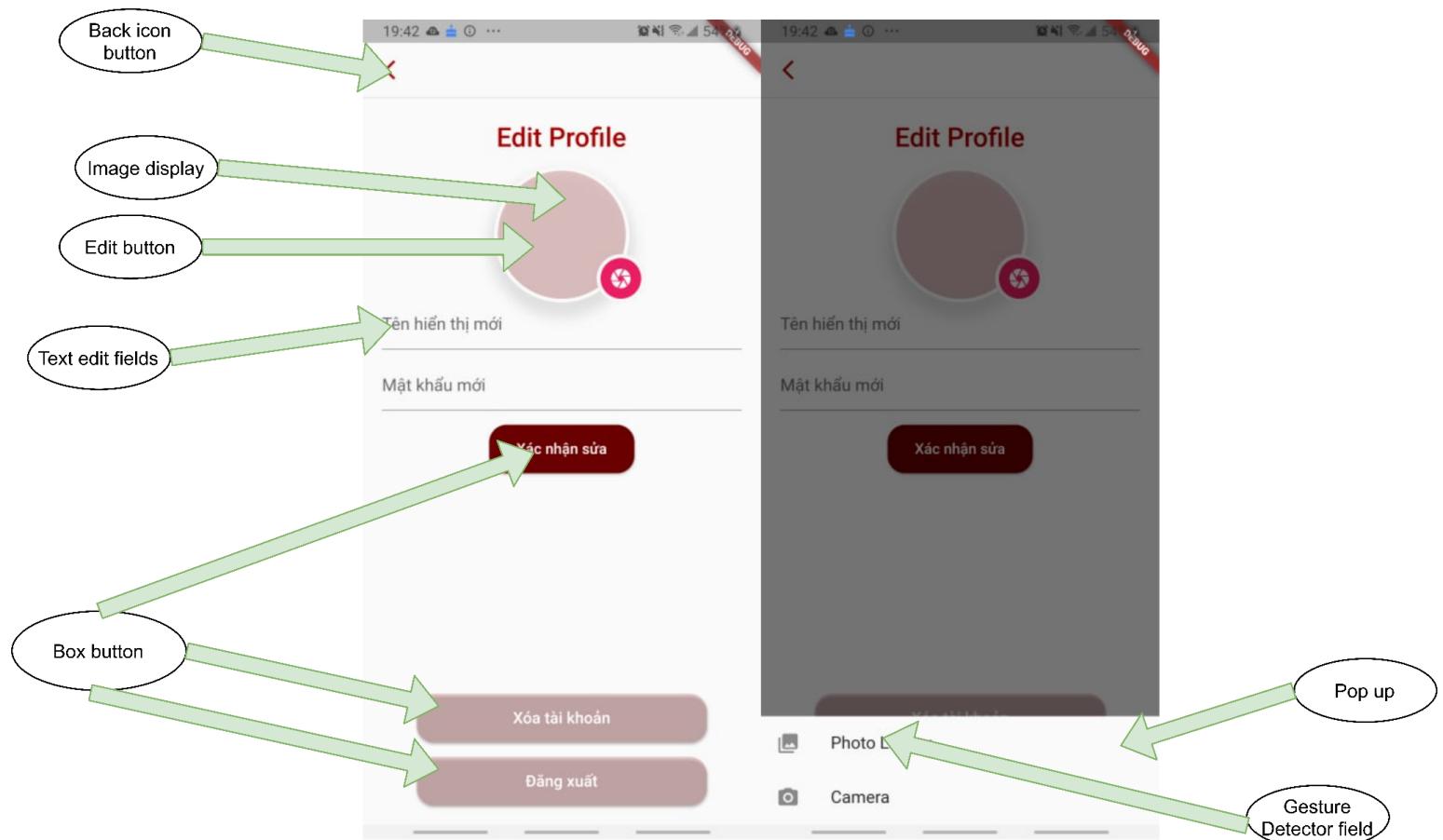


Figure 8: User interface of modifying main user's profile.(available feature)

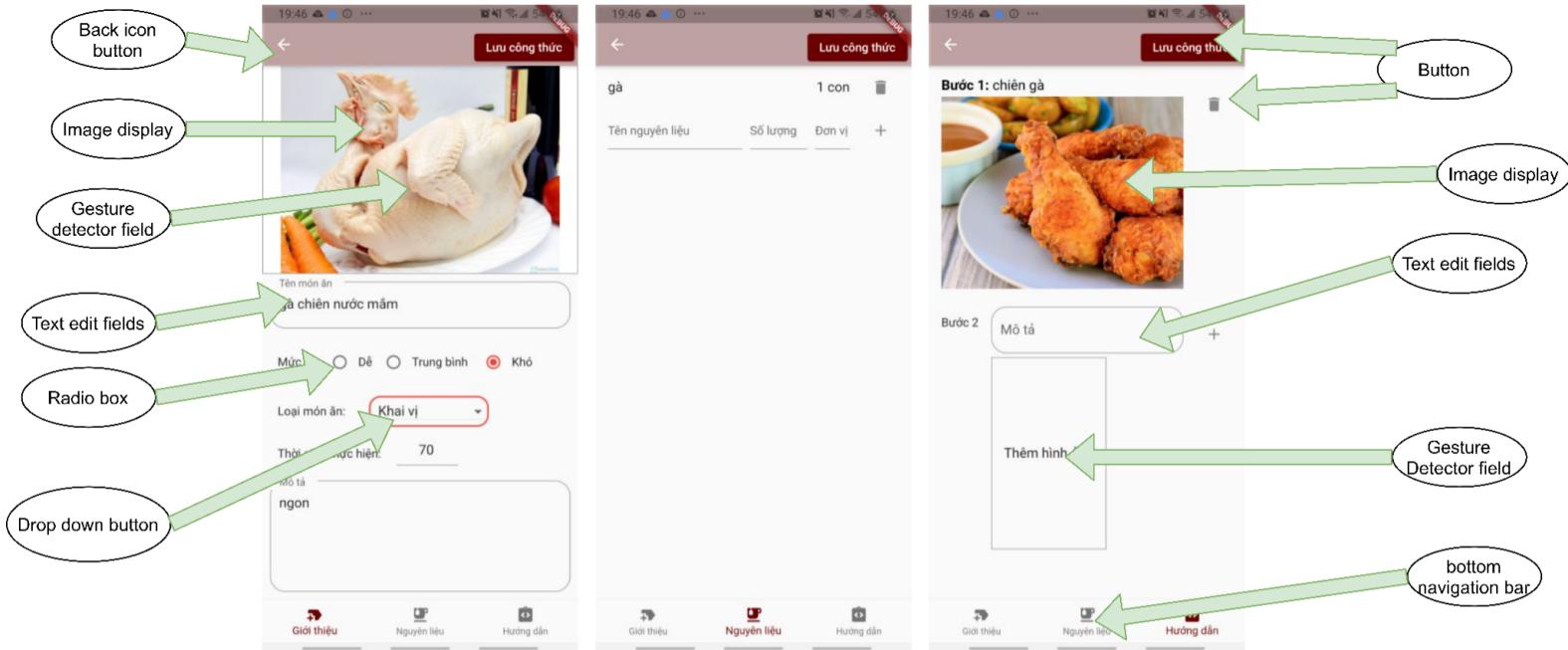


Figure 9: User interface of creating a new or editing the already created recipe(left: introduction page, middle: ingredient page, right: instruction page). **(available feature)**

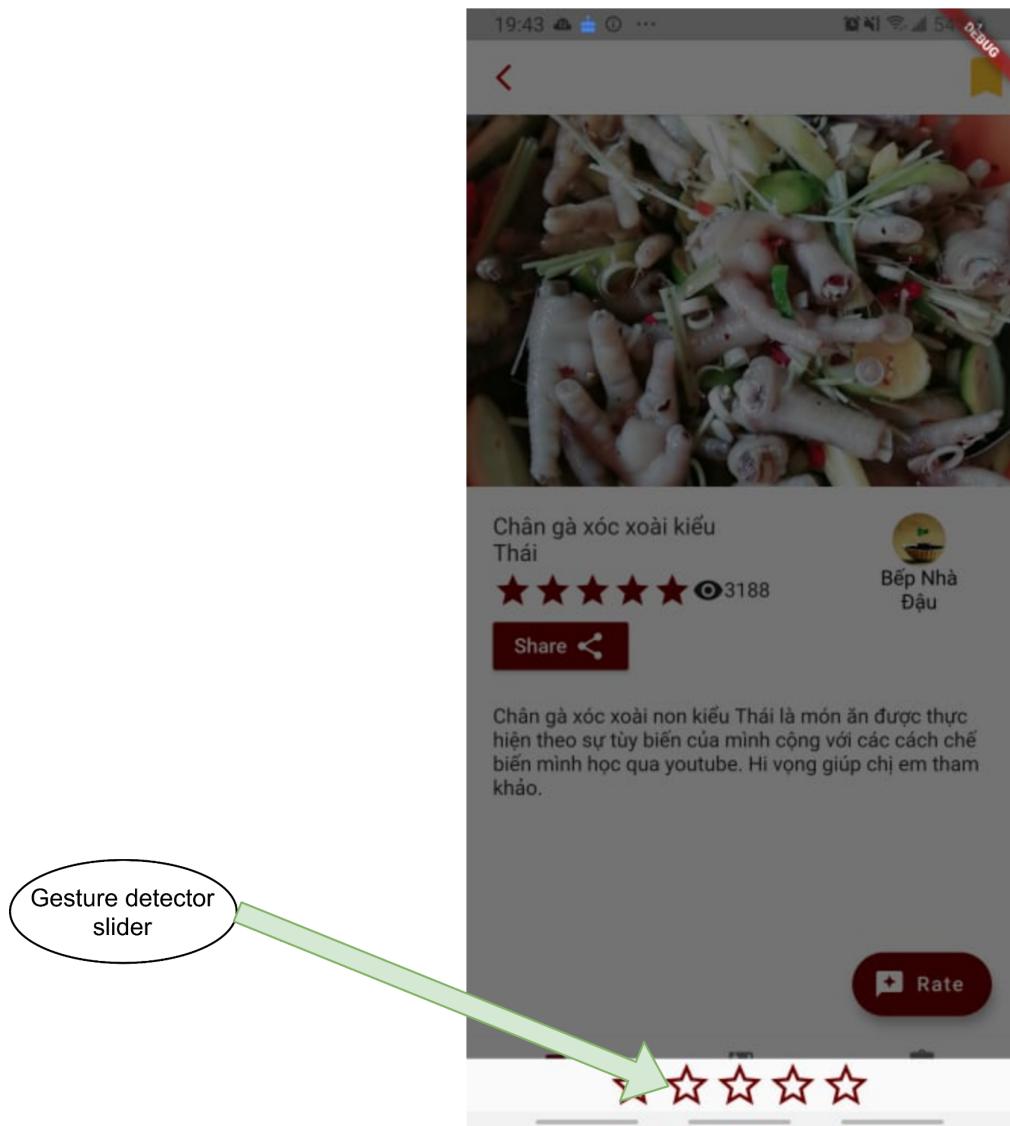


Figure 10: User interface of rating action in a specific recipe. **(available feature)**

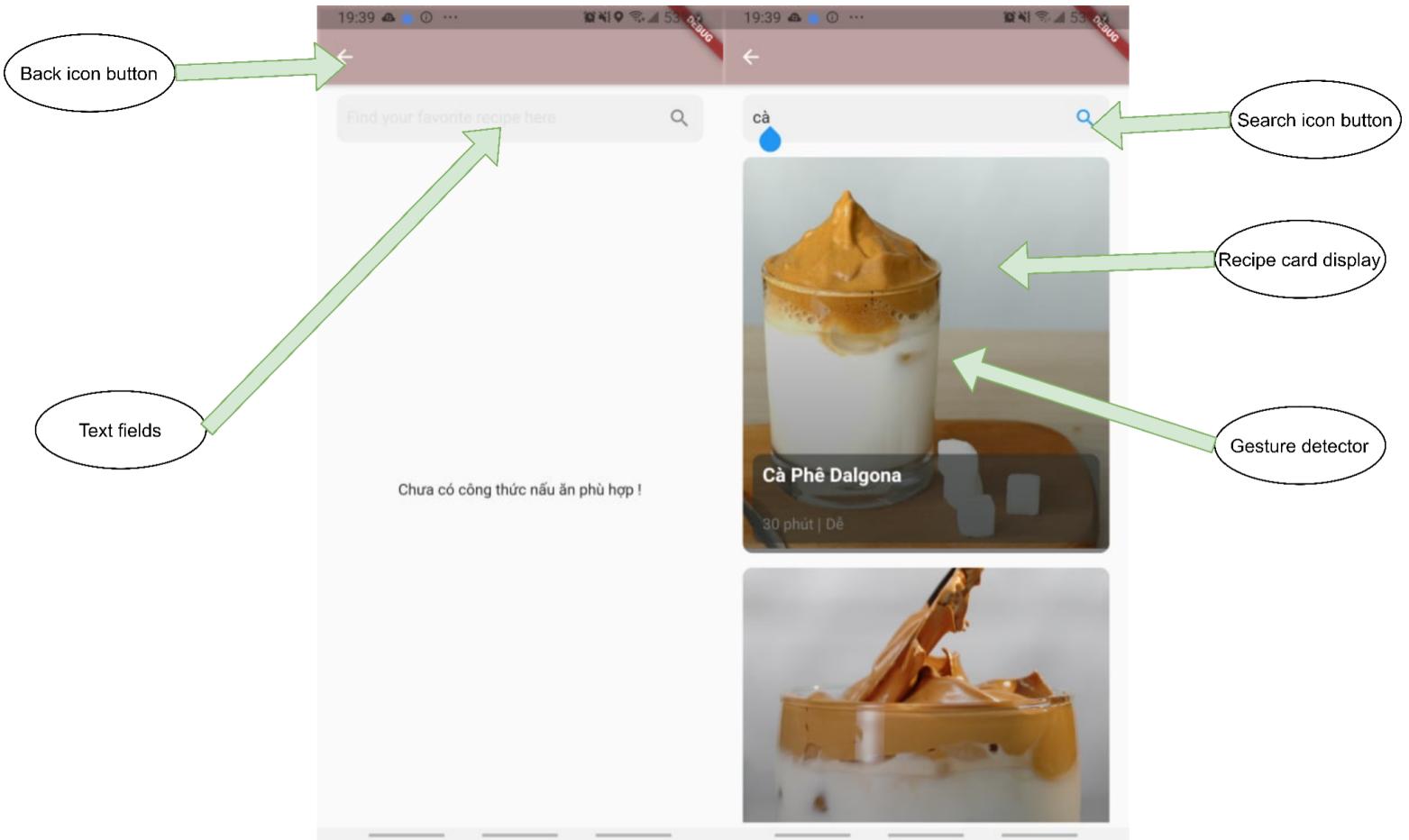


Figure 11: User interface of searching a recipe keyword title name. (**available feature**)

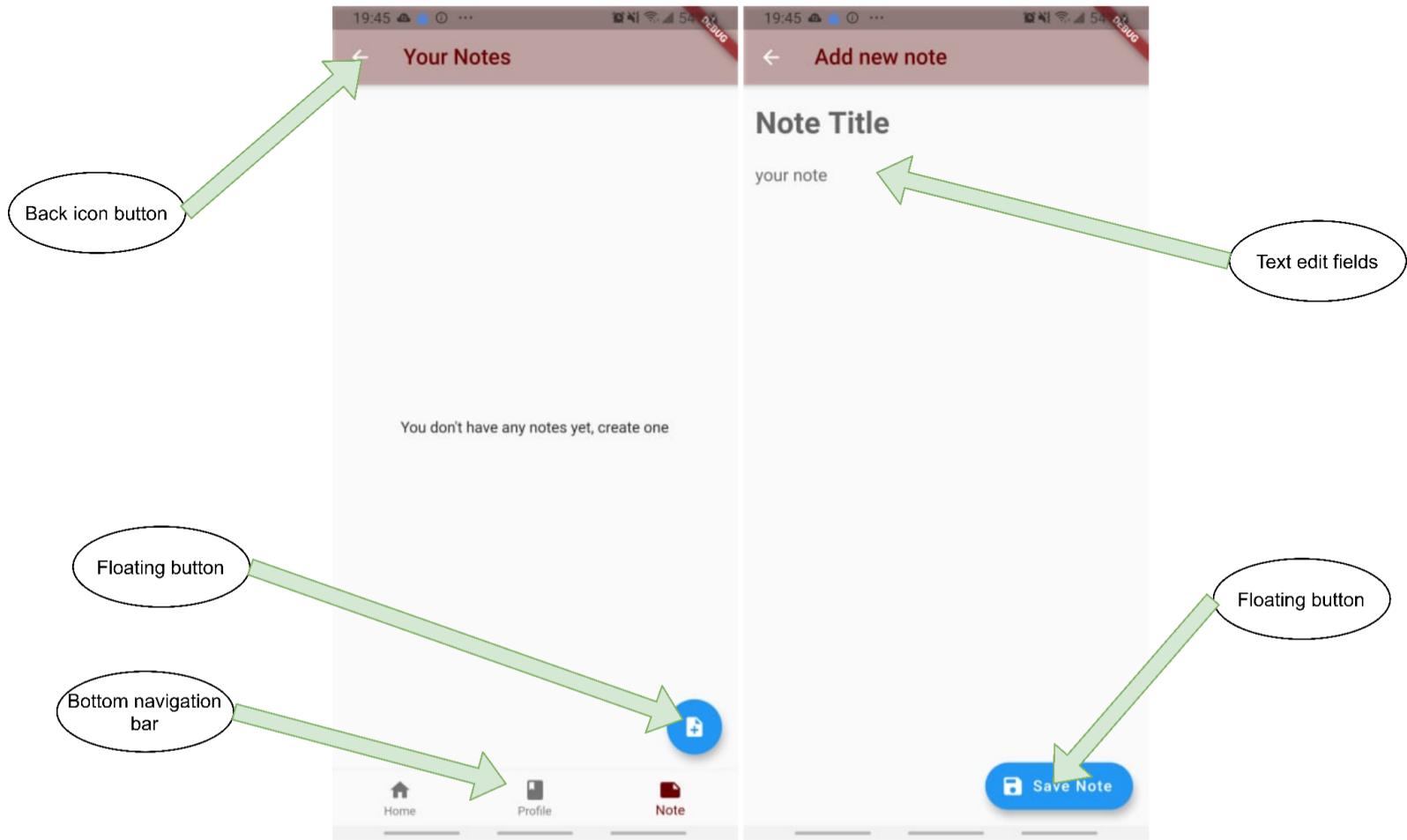


Figure 12: User interface of creating new notes. **(future updated feature)**

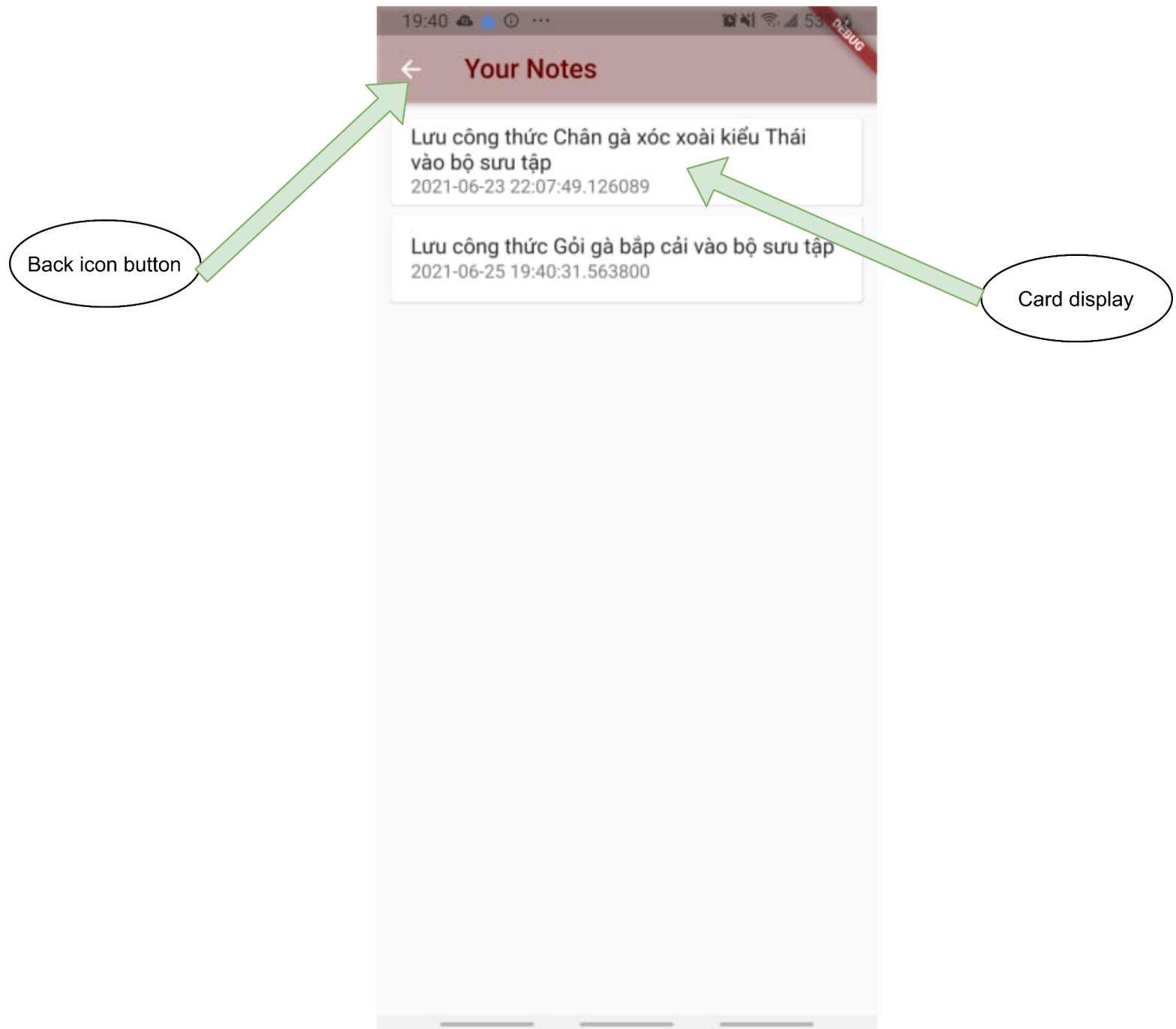


Figure 13: User interface of viewing the application history action. (**future updated feature**)

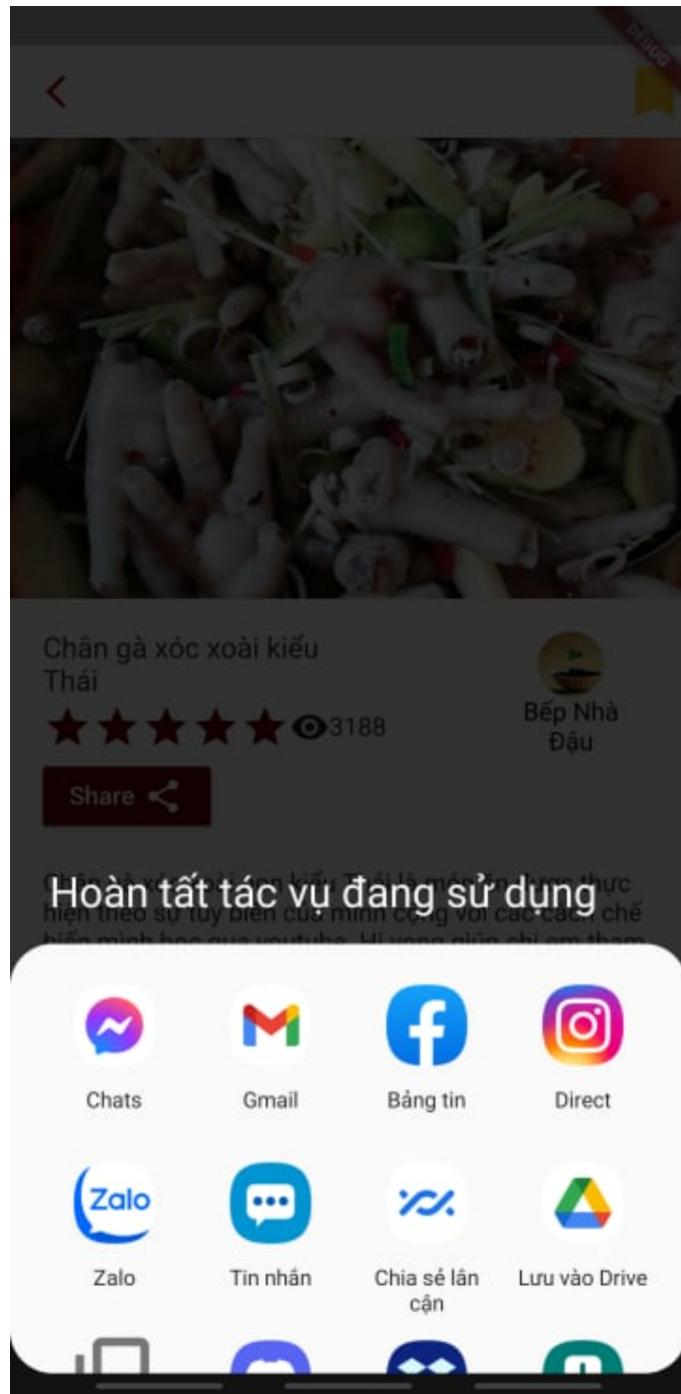
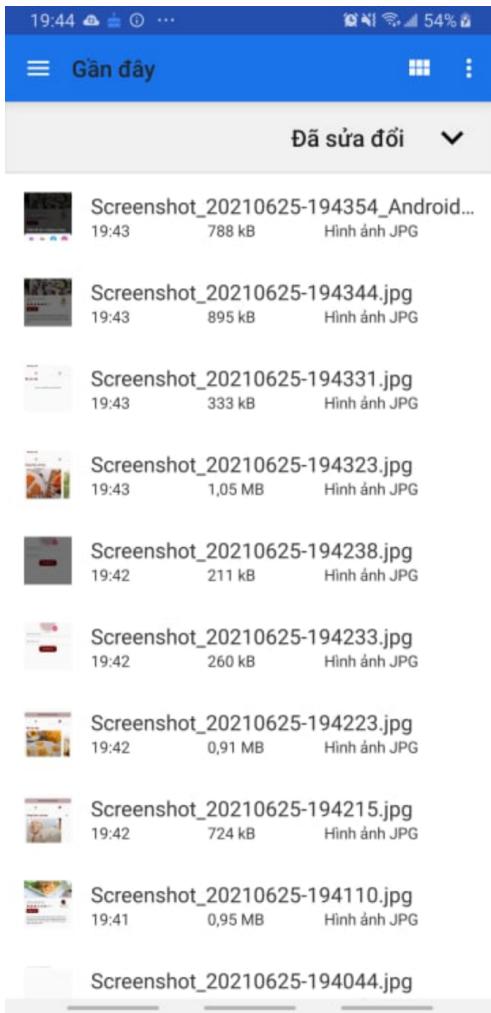


Figure 14: User interface of sharing recipe to other platforms. **(future updated feature)**



19:45 54%  
accounts.google.com/o/oaui  
Đăng nhập bằng Google

Chọn tài khoản  
để tiếp tục tới Flutter Drive

Quang Tau  
nhatquang.panthera0207@gmail.com

QUANG Tau Nhật  
quang.tau\_panthera@hcmut.edu.vn

Quang Tau  
nhatquang.carnivore0207@gmail.com

Rj Alexander  
pantheraatrox27@gmail.com

Resources ASKademyPVC  
askademy.pvc.resources@gmail.com

trem queng and  
quengiutrem.4221@gmail.com

Sử dụng một tài khoản khác

19:45 54%  
accounts.google.com/signin  
Đăng nhập bằng Google

Chọn tài khoản  
để tiếp tục tới Flutter Drive

Quang Tau  
nhatquang.panthera0207@gmail.com

QUANG Tau Nhật  
quang.tau\_panthera@hcmut.edu.vn

Cấp quyền cho Flutter Drive  
Chỉ xem, chỉnh sửa, tạo và xóa  
các tệp Google Drive bạn sử  
dụng với ứng dụng này

Từ chối Cho phép

trem queng and  
quengiutrem.4221@gmail.com

Sử dụng một tài khoản khác

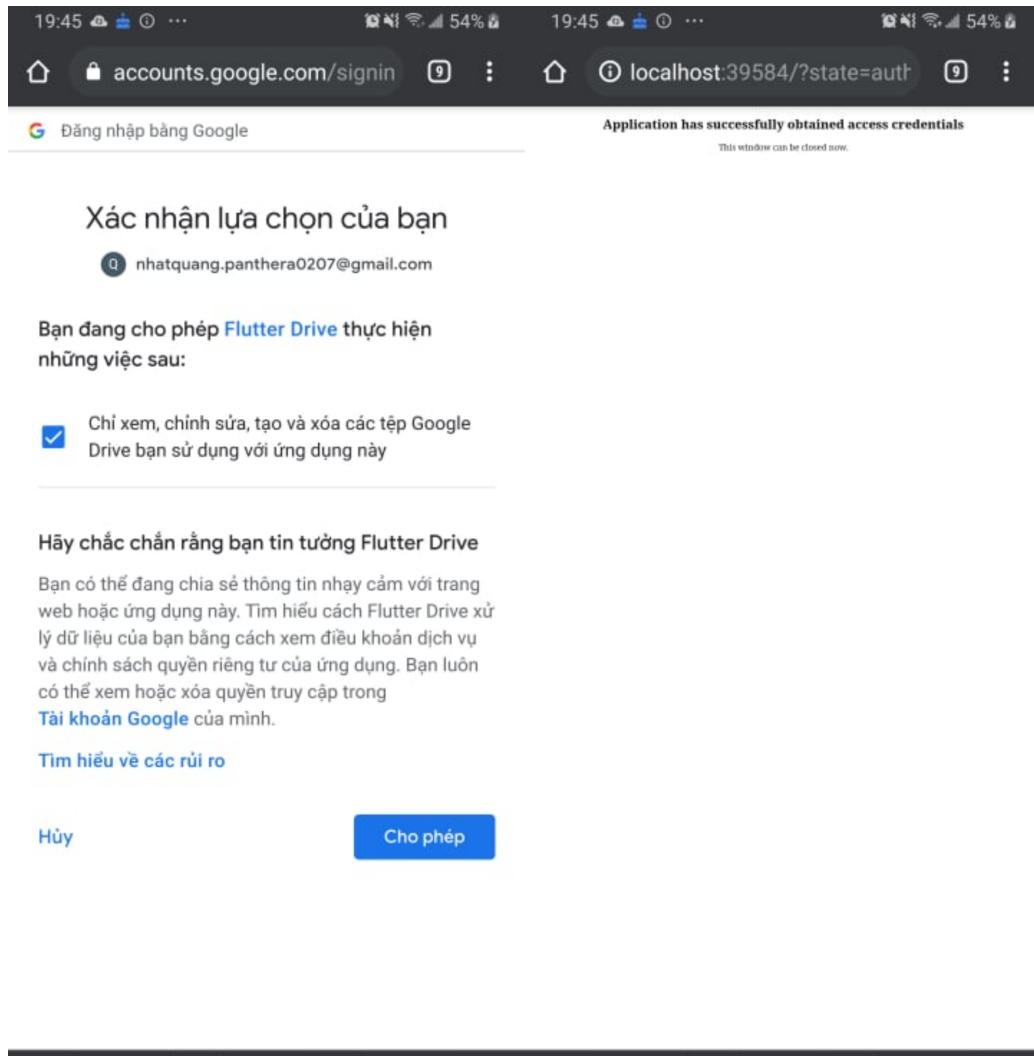
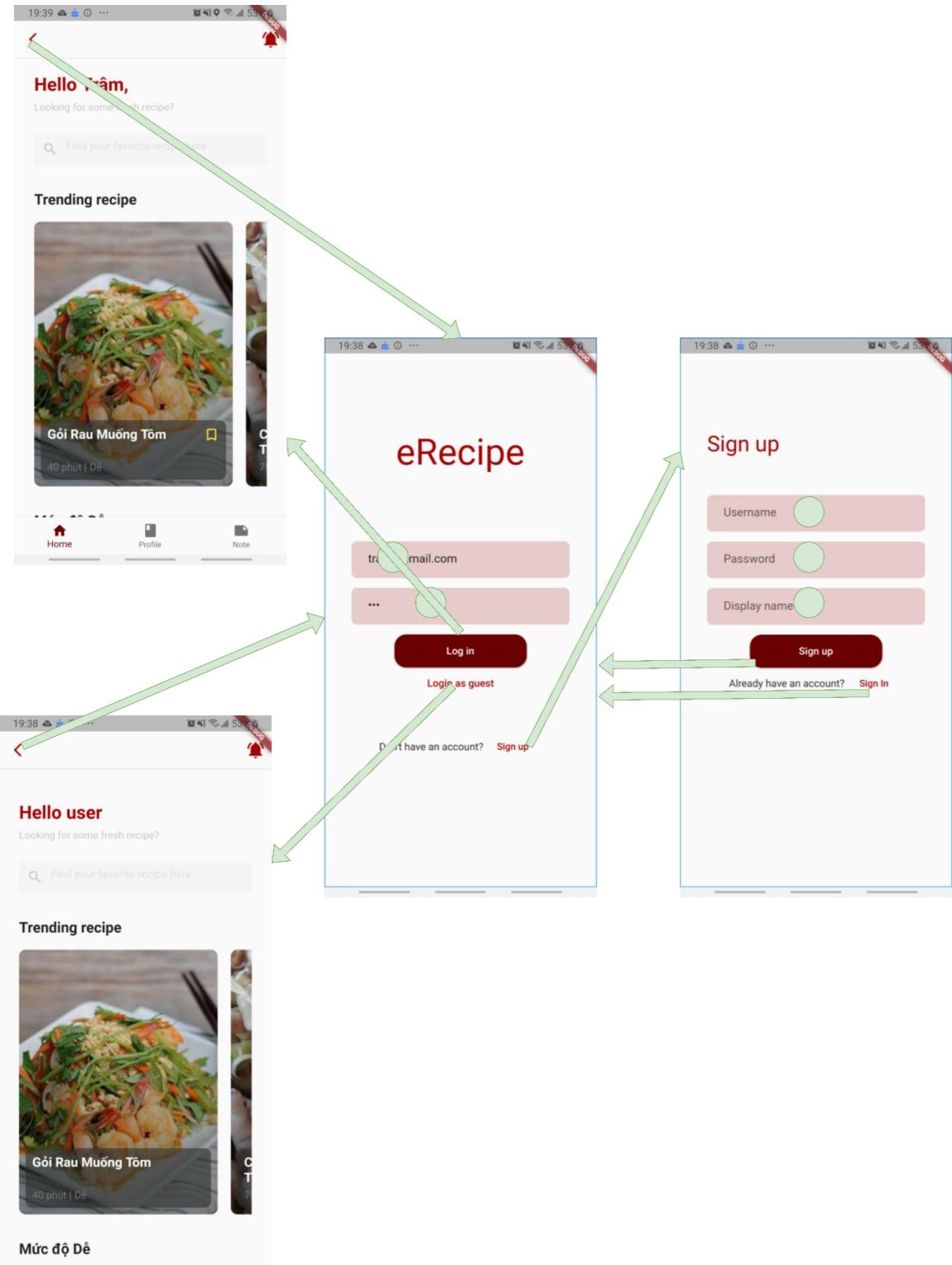
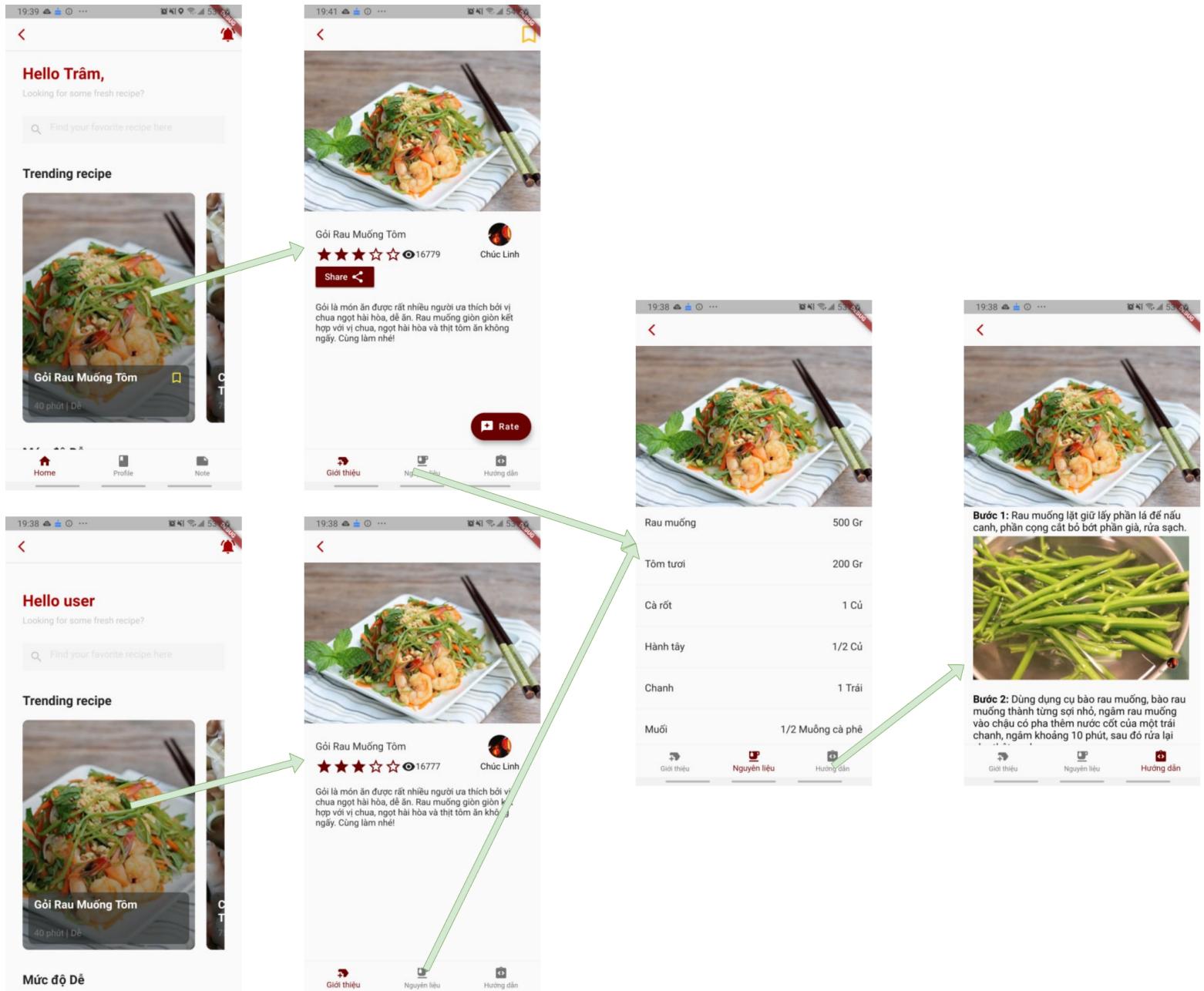


Figure 15: User interface of google drive image integration. (**available feature**)

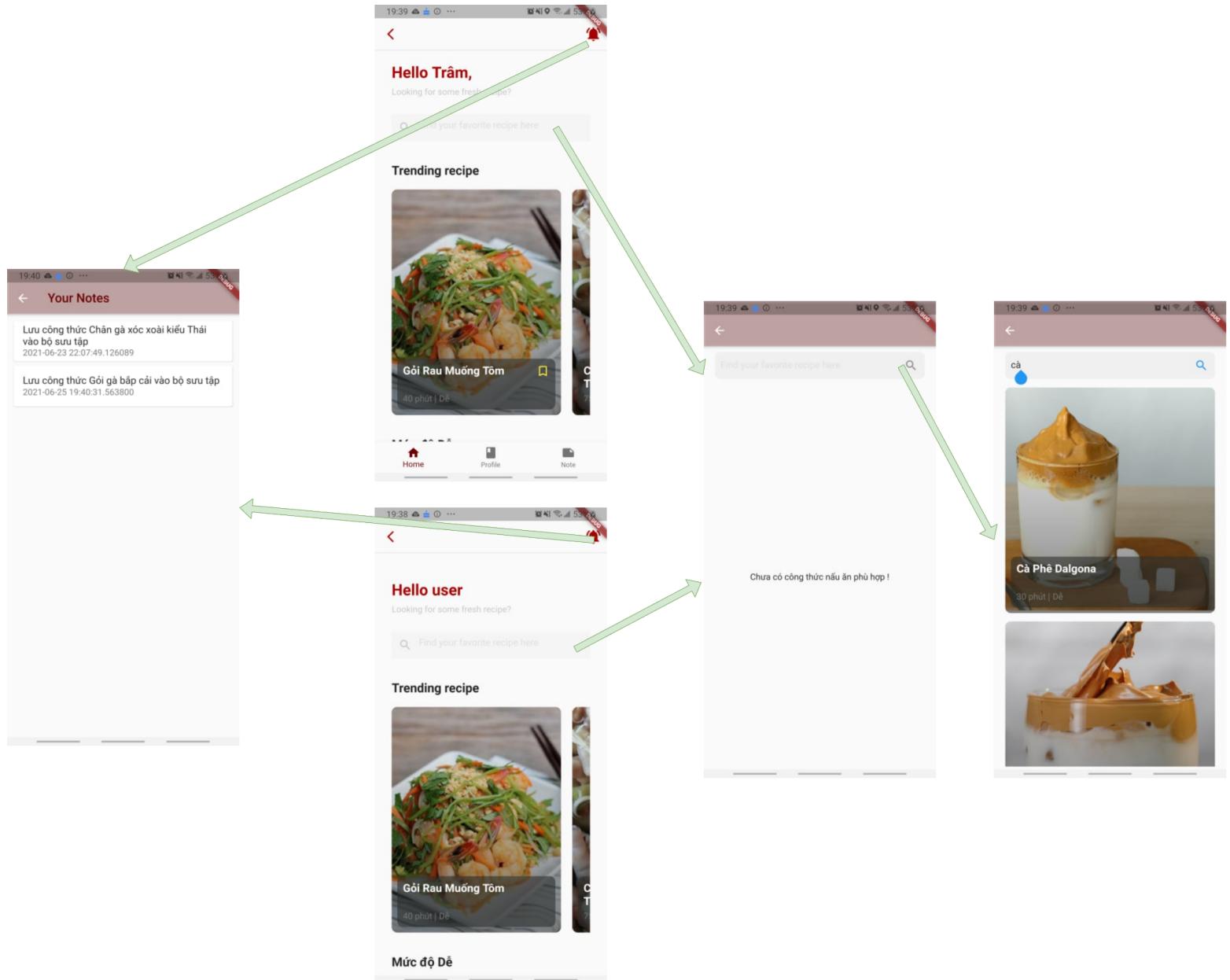
## MOCK-UP FLOWS:



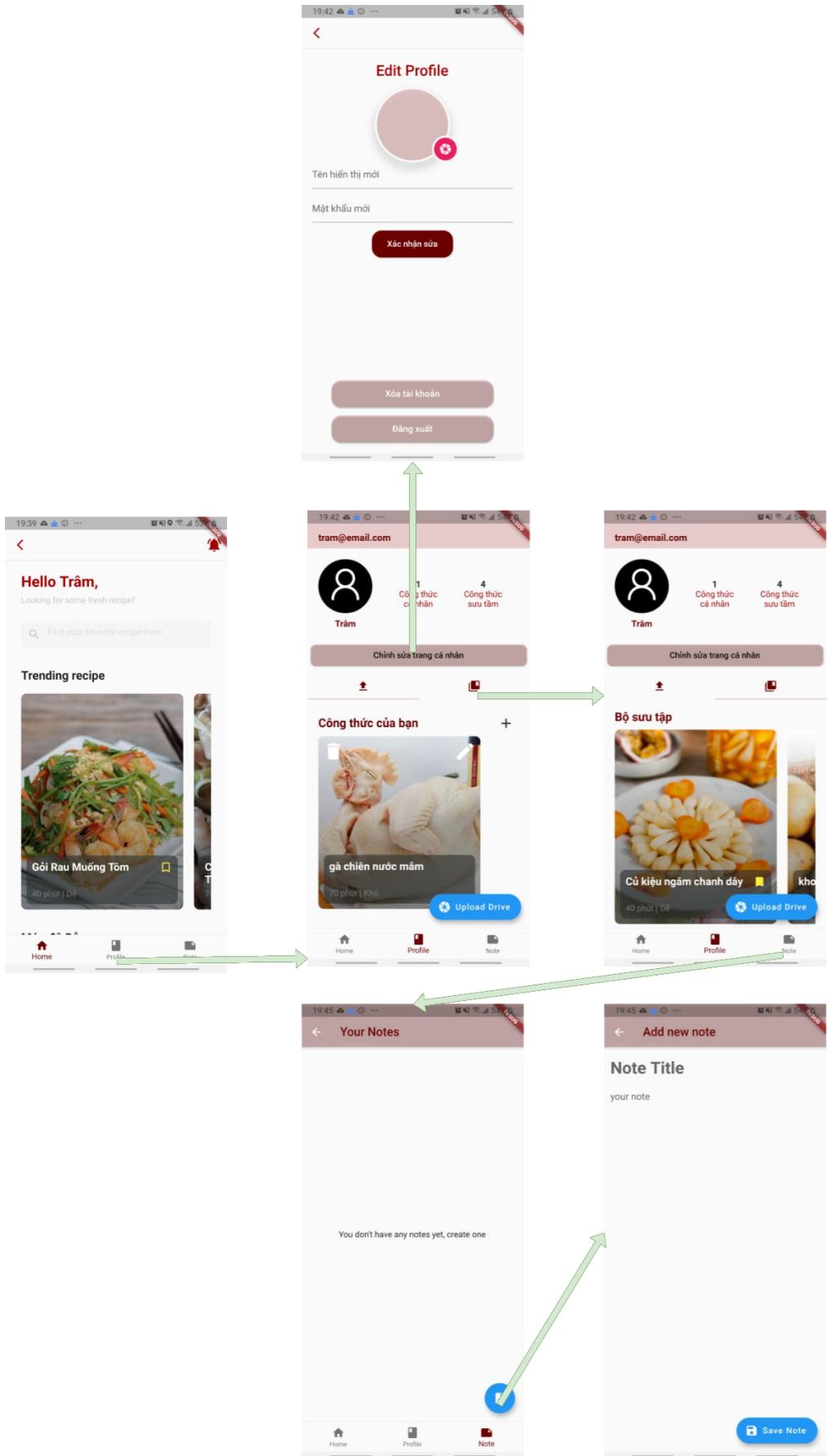
**Mock up flows between login screen, sign up screen, home screen as guest, and home screen as registered user.**



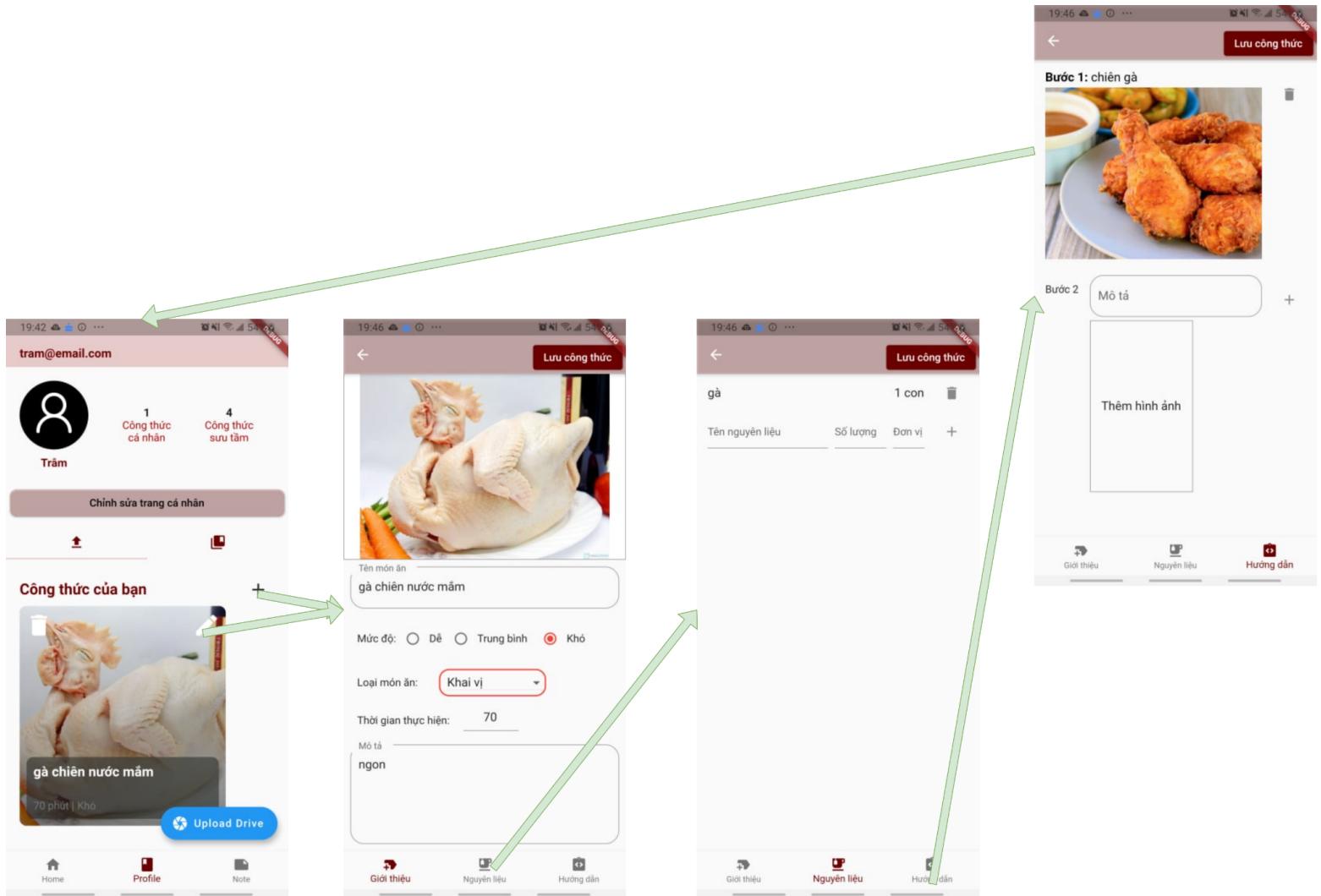
**Mock up flows between home screen and recipe detail screen with introduction , ingredients and instructions component screen.**



**Mock up flows between home screen, search screen and history action screen.**



**Mock up flows between home screen, profile screen, edit profile screen, note display screen and edit note screen.**



**Mock up flows between profile screen and add/edit recipe screen.**

## VII. Implementation view

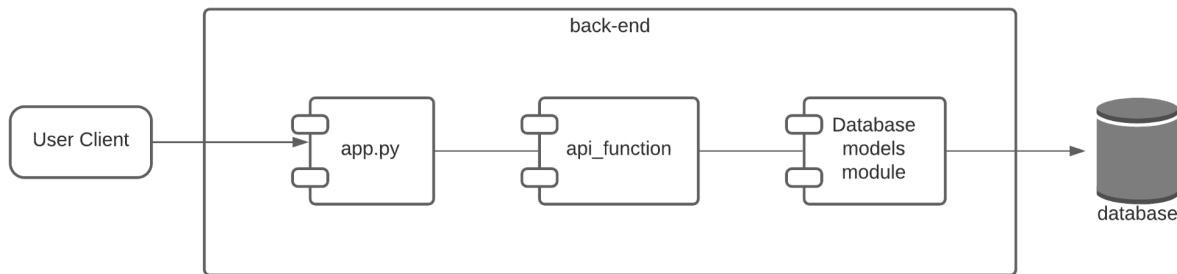


Figure: Implementation view on the backend server

Component explanation:

- App.py module: Set up the API to receive request from the client, handle the request and operate functions from api\_function module accordingly, error handling and take cares of login-authorization
- Api\_function module: call CRUD operations provided from the database model module to operate (i.e. on viewing a post, query up that post, then update the database to increment view count)
- Database models module: Model the database document fields, provide basic interface for creating, reading, updating, deleting the mentioned documents into our database.

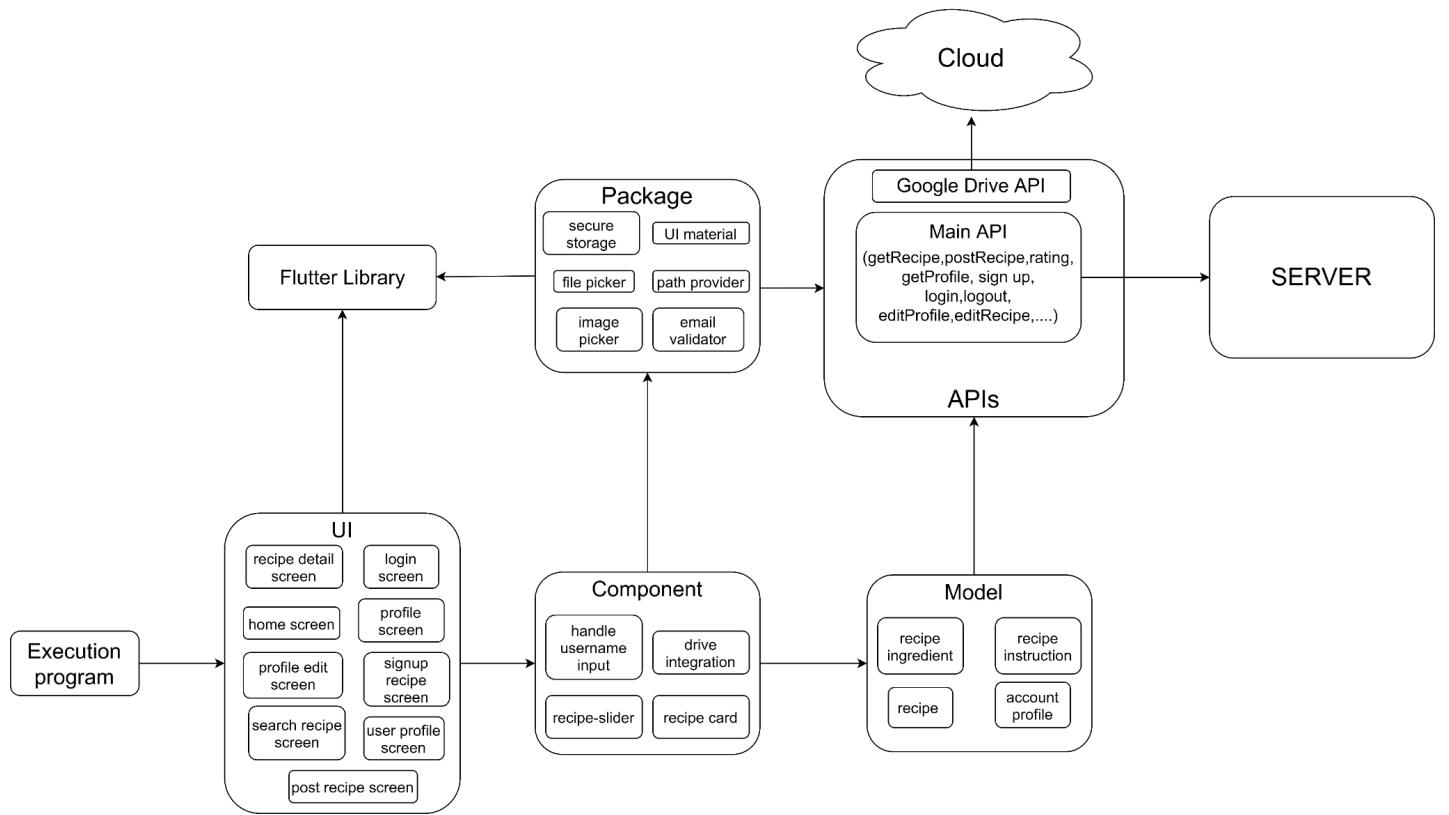


Figure: Implementation view on the frontend.

- **UI:** includes all the codes that draw and design screen from boxes, buttons and different components supported by Flutter library (material package ,....)
- **Model:** design objects and classes to pour data to the equivalent UI. Model also creates mapping methods to map from json data to variables of specific classes to pour the data.
- **Components:** initialize base objects for complicated structure of UI for example: recipe card slider,... and handles complex featured functions.
- **API:** handle API calls to get data, post data , as well as delete and edit data. It ensures the communication between front end and back end to work smoothly.

## VIII. Deployment view

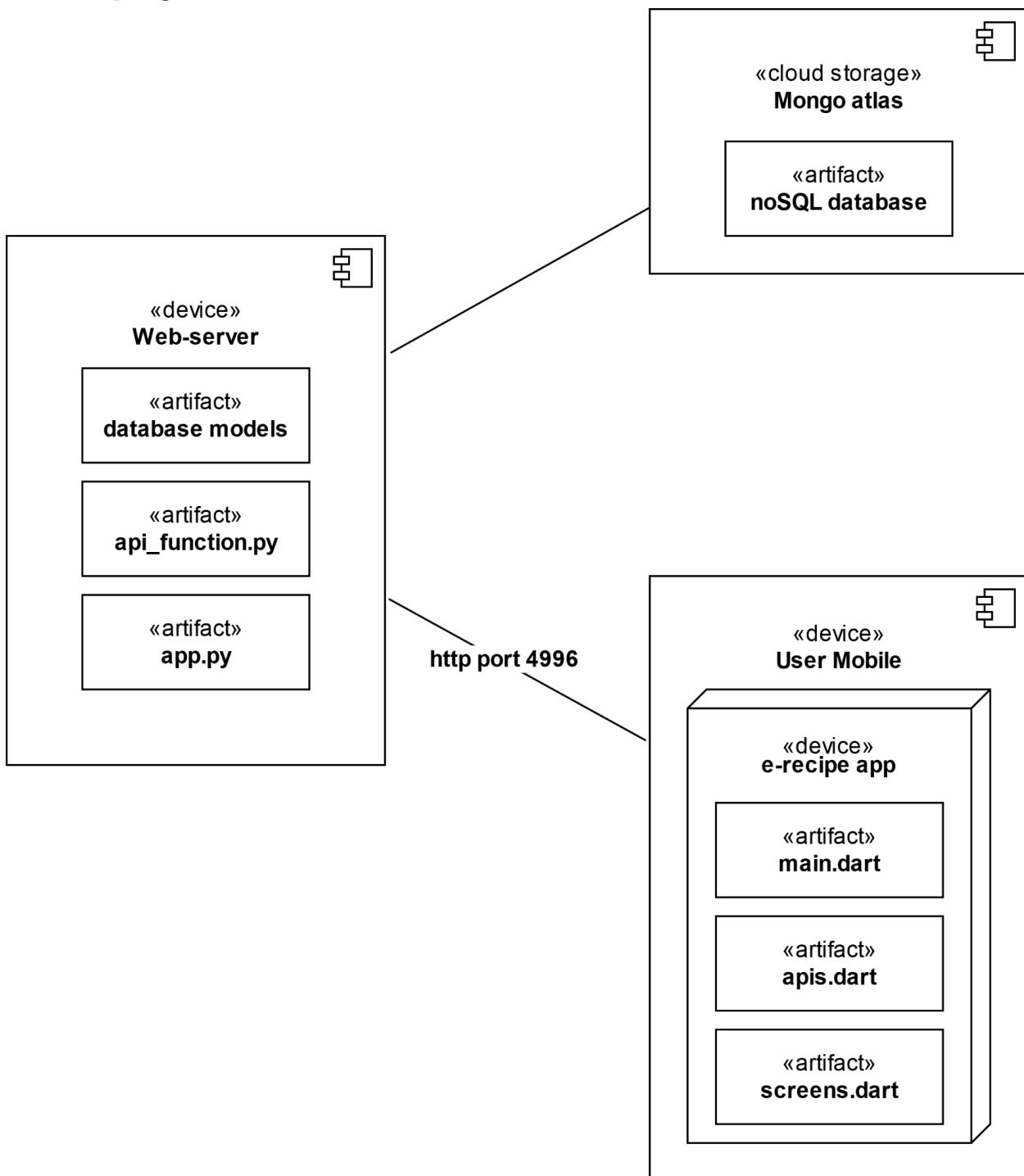


Figure: Deployment view

## Component explanation:

- E-recipe app: connect to the product service via a mobile app, the app would display a GUI for the user to use the functionalities that the software offers.  
The client-side sends https requests to the server to get data(view posts, browsing view,... ).  
Handles client-side processes.
- The client app would also be able to publish http messages to the server. (make a post, making a comment,... )
- Database: Stores data of the service(user account data, posts, ... )
- Web-server: listening over port 4996, handles all kinds of api requests, from user registration, user profile management upto recipe creation and recipe management

## IX. Class diagrams:

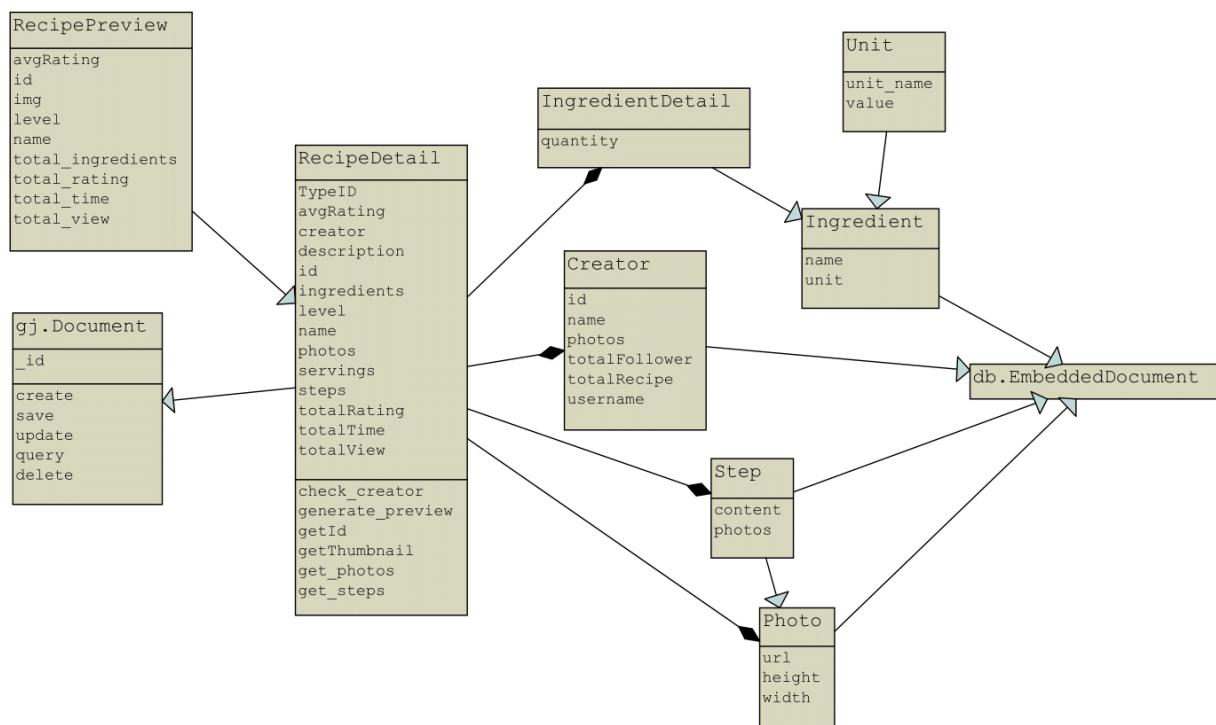


Figure: Model of the recipe stored in database

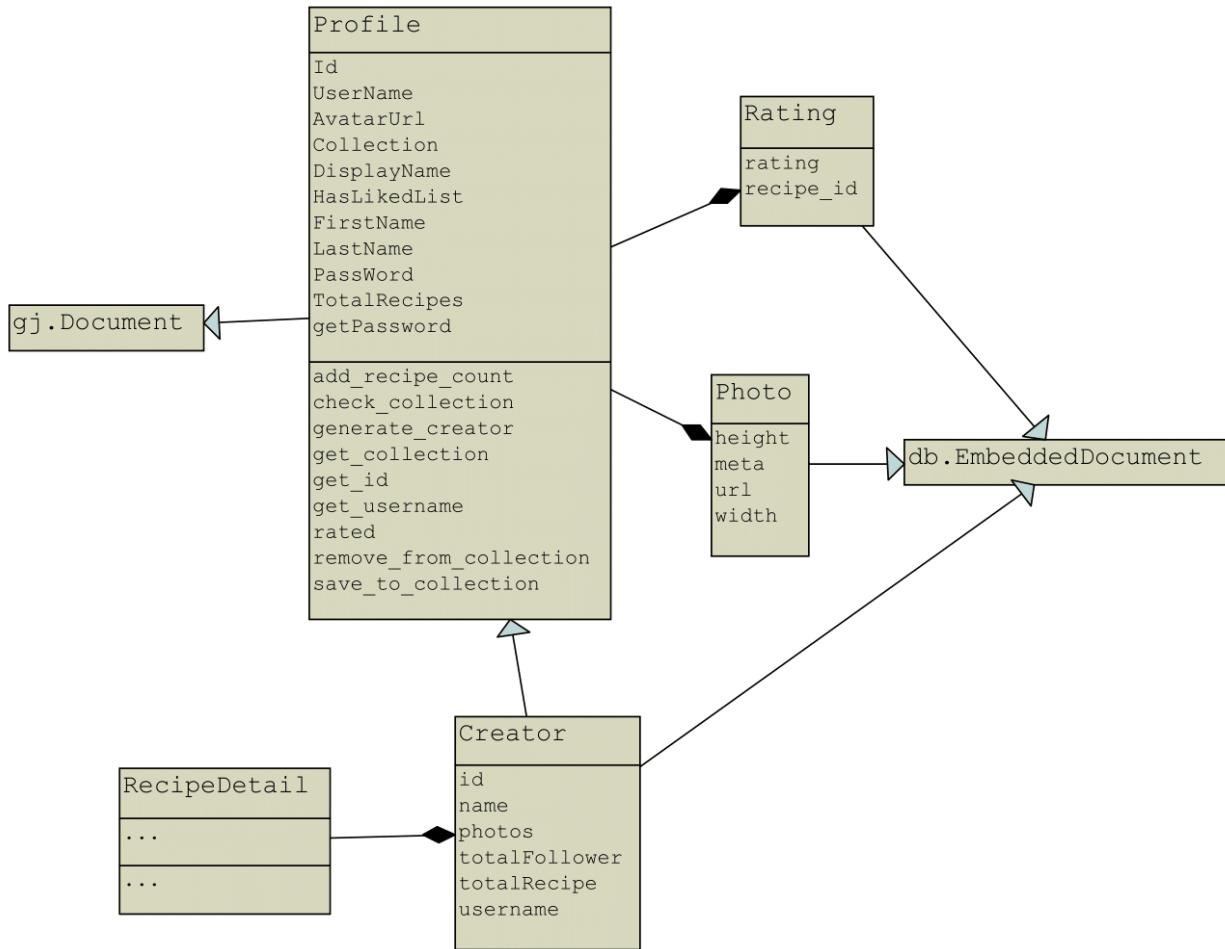


Figure: Model of the profiles stored in the database

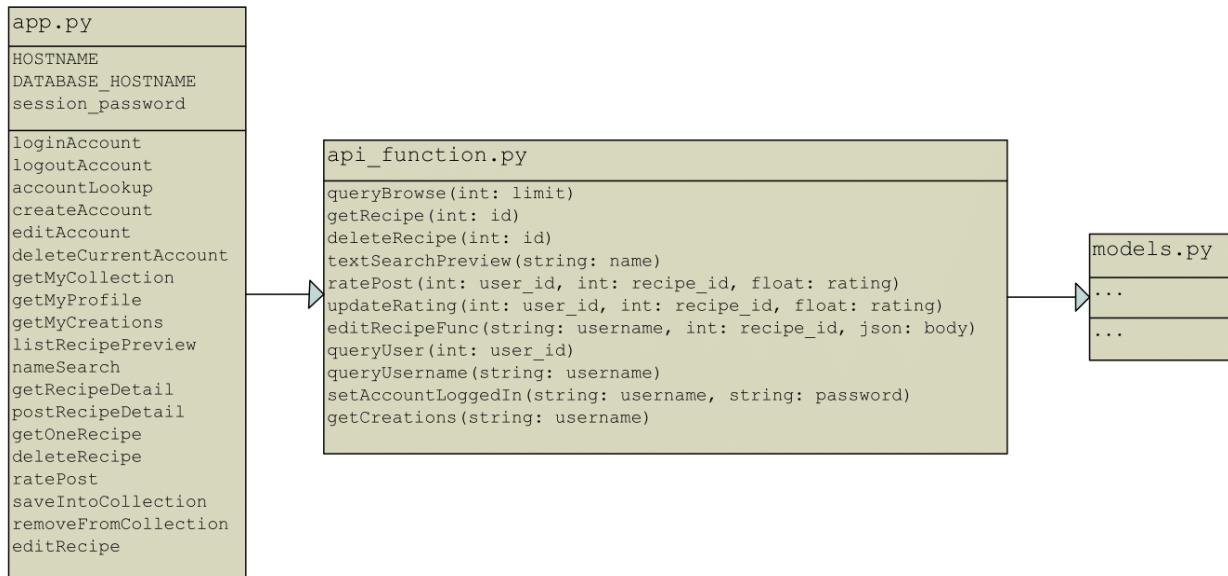


Figure: class diagram of the main module of the back-end

## X. Sequence diagrams:

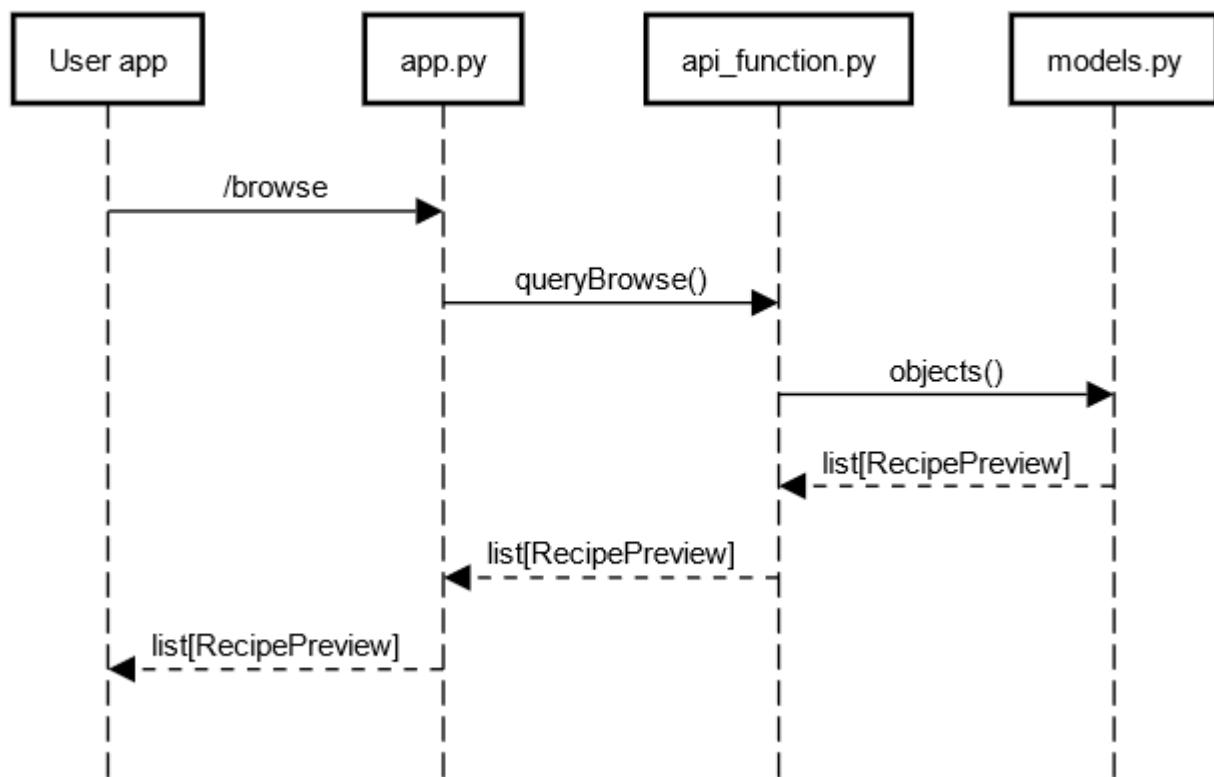


Figure: Sequence diagram on query the preview of recipes

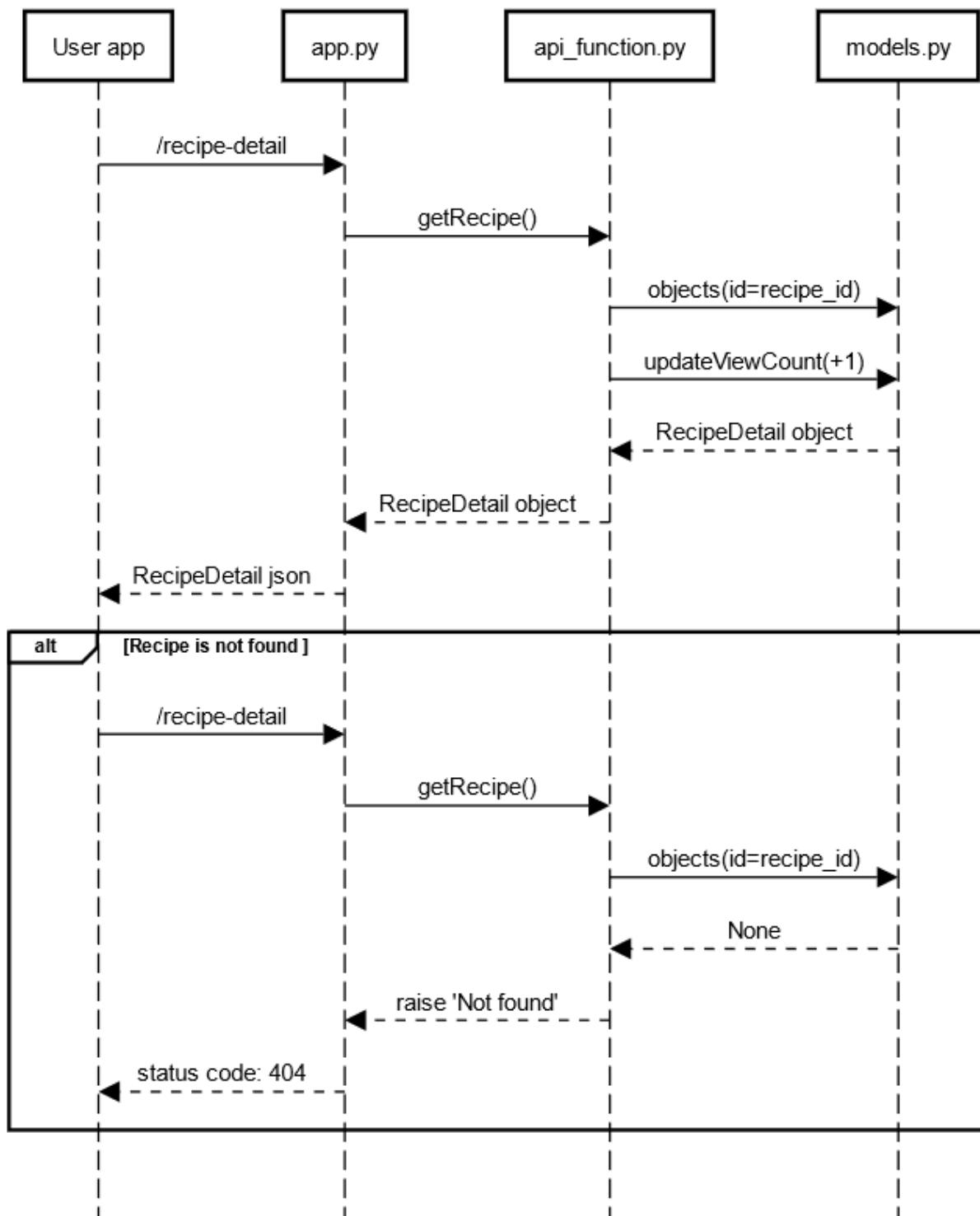


Figure: sequence diagram on getting details of a single recipe

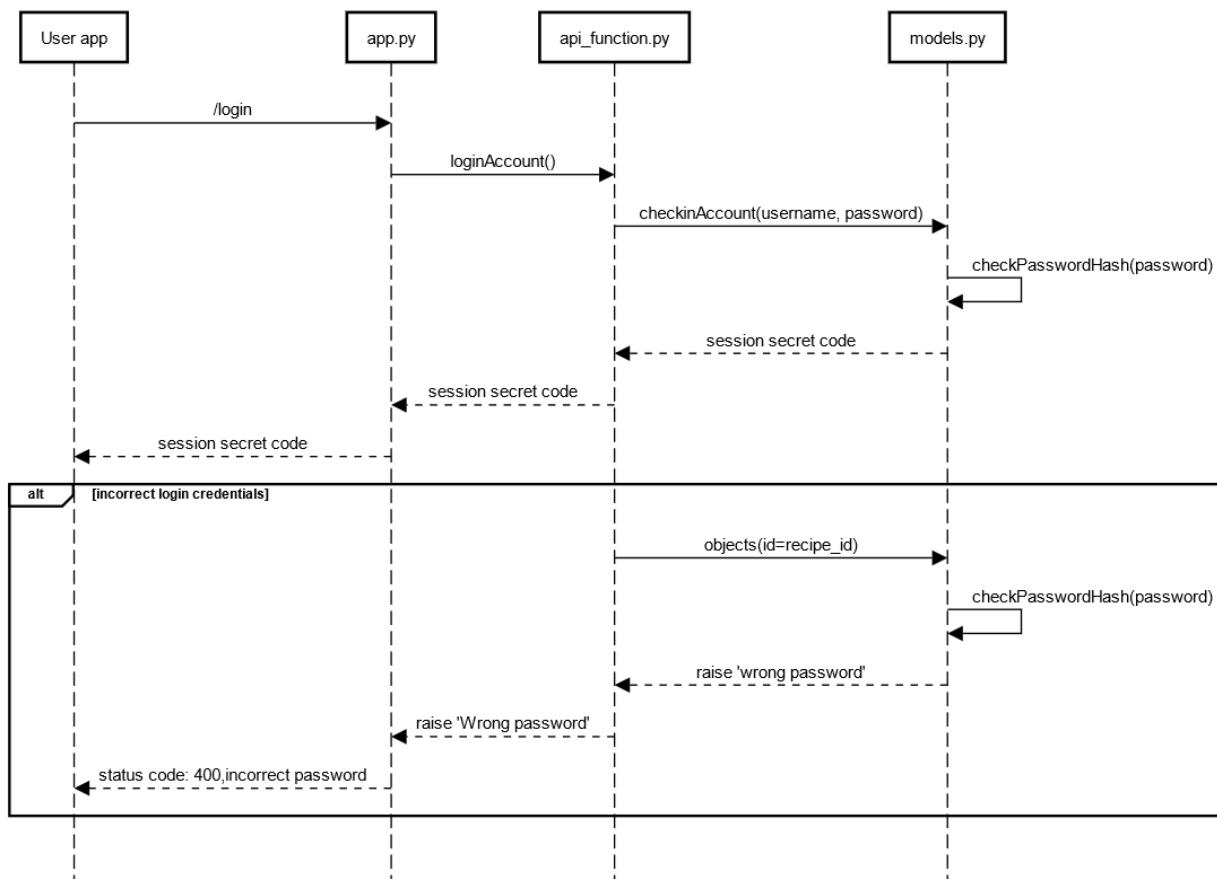


Figure: sequence diagram on logging in

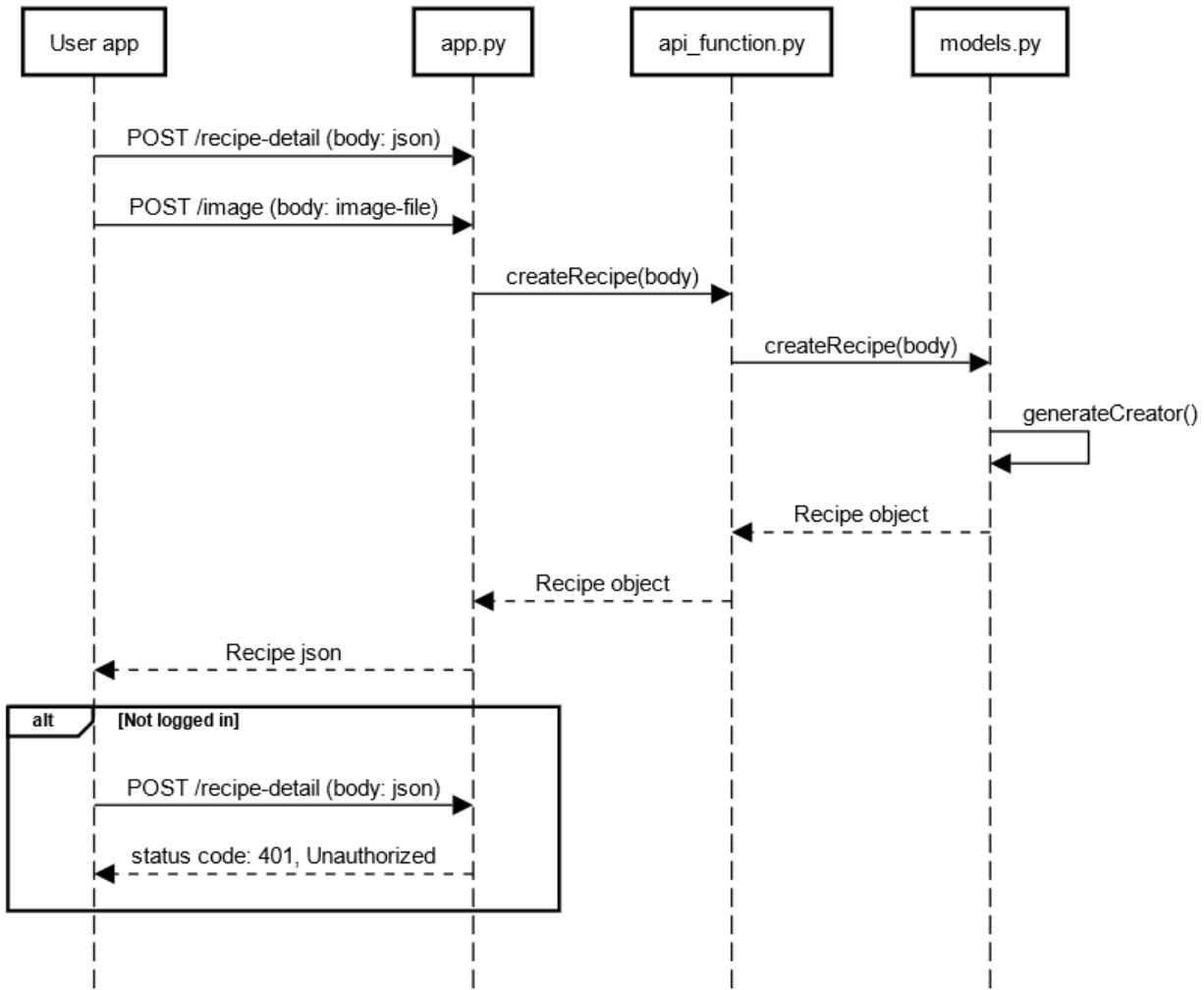


Figure: sequence diagram on posting a new recipe

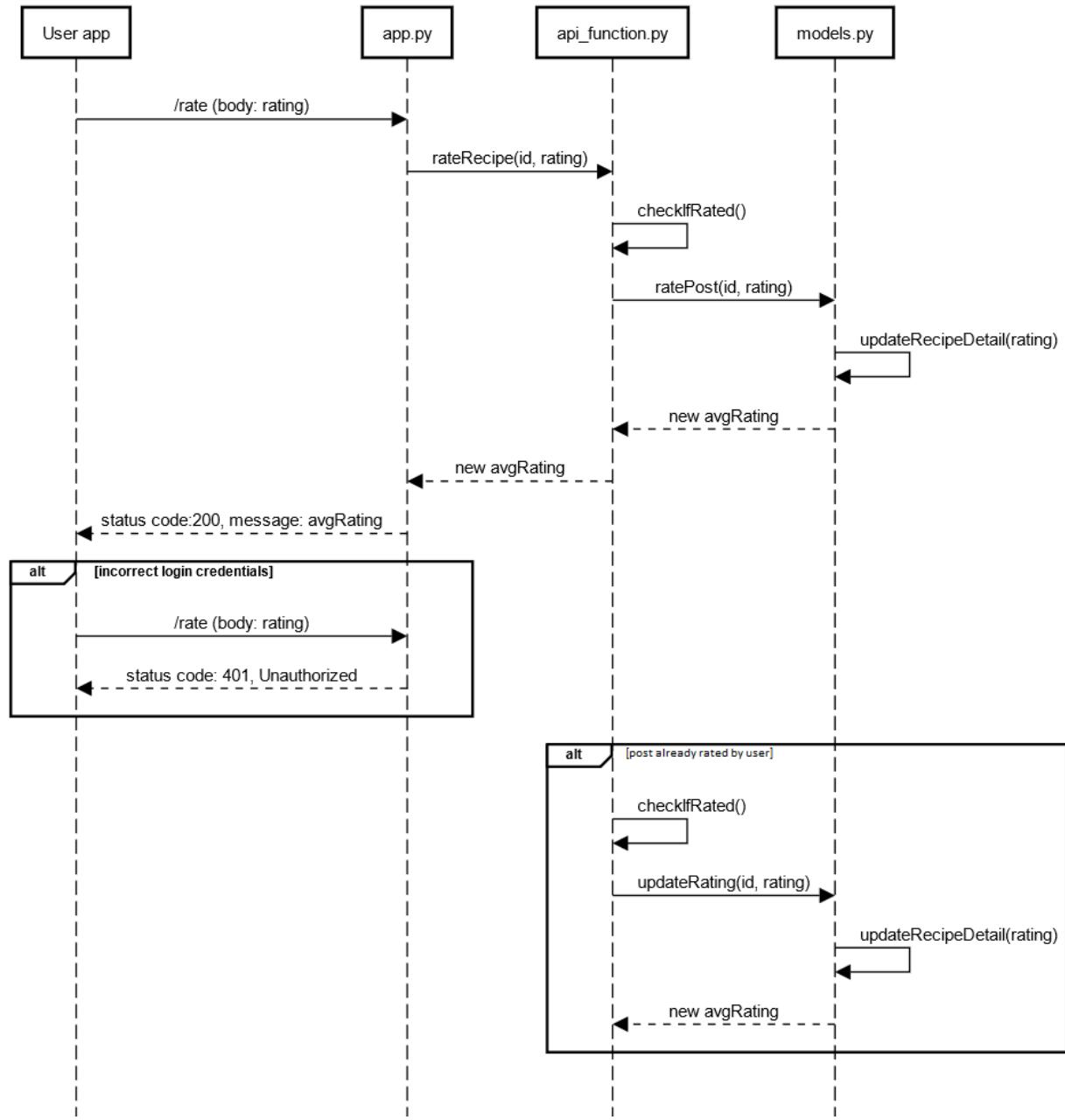


Figure: Sequence diagram on rating a recipe

## Conclusion

Github repository: <https://github.com/PantheraXpoint/Recipe-Forum-App>

The user app interface is functionally completed, along with the back-end server, ensuring that most of the main use-cases are covered and are working as intended. Although we pruned the use-cases of recipe sales, cash trading, commenting from the initial conceptual design phase.

#### Possible upgrades:

- Currently we implemented the login mechanism on the session cookies, which is still vulnerable to cookie theft. Possible upgrade would be to use an independent authorization header on each http request from the client app
- Images are now hosted locally on the host device, which is less than ideal, further plan would be to move the image storage onto the same mongo atlas cloud.
- The Take-Notes feature is now incomplete. In the future, taking notes will be more private and better supporting the user as a reminder.
- The sharing feature is now on-going as it still cannot share a specific url link to other platforms but contents only.
- Drive integration is now on-going as it can only manipulate images but not a whole recipe. In the near future, further plans will take this feature into consideration to make it more convenient for the user.