

Foody Moody: How to Get Food Recommendation Through Mood

Building an Android Application to Expand User' Food Choices

1st Estella, T.
School of Computer Science
Bina Nusantara University
Jakarta, Indonesia
tricia.estella@binus.ac.id

2nd Levina, A.
School of Computer Science
Bina Nusantara University
Jakarta, Indonesia
audrey.levina@binus.ac.id

3rd Ghayatrie, N. A. I.
School of Computer Science
Bina Nusantara University
Bekasi, Indonesia
nadzla.ghayatrie@binus.ac.id

4th Dr. Anggreainy, M. S.
School of Computer Science
Bina Nusantara University
Jakarta, Indonesia
maria.anggreainy@binus.edu

5th Kurniawan, A.
School of Computer Science
Bina Nusantara University
Jakarta, Indonesia
anton.akurniawan@binus.edu

Abstract—This abstract will be updated when we finish in developing our project (*Abstract*)

Keywords—Software Engineering; Food; Mood; Well-being; Healthy Eating;

I. INTRODUCTION

The idea for our project came when we had a difficult time determining food. The problem may seem simple, but so many variables can affect our wants and decisions when choosing food. For example, cold weather makes us crave warm food and vice versa. Or when we're sad, we need sweet foods to mend our emotions. The notion of food intake and the types of foods available have evolved over the last ten years as the diversity and accessibility of food options have increased. These modifications may have both beneficial and bad consequences [1].

The link between mood and food is complex, since our mood may impact the foods we eat. For example, positive mood stimulates distal, abstract construal and raises the relevance of long-term objectives such as health, according to Meryl P. Gardner et al. [2]. This leads to a stronger choice for healthy diets over decadent meals. The findings also suggest that a bad mood activates the proximal construal and raises the relevance of immediate, tangible goals like mood management, resulting in a higher desire for decadent meals over healthy foods. With varying choices, it is difficult to determine which food we will eat at the moment, especially when we must adjust our preferences to our moods.

These days, the question of food selection is already a crucial one. With technological science, psychology, and food growing, many people have begun to match the foods they consume with the body's condition they are experiencing. Some also make food or drink a refuge when healing their soul or body. Therefore, we would like to create an app called Foody Moody, which would later display a recommendation of foods consistent with the user's mood input and current weather data.

In realizing this Foody Moody application, we will need some data or surveys to find a connection between mood and

food. For user mood, it will be obtained through the user input application, which is a category or scale from happy to sad. We hope that the Foody Moody application can help users determine their food at an efficient time, without a lot of thought. Foody Moody can display appropriate food and possibly new food that users have never tasted. We also hope it will help the culinary business, primarily small businesses, promote food, thus being known by the large media. Ease of Use

A. Problem Formulation

- How to save time in choosing a dish?
- How to make users find their desired food easier?
- How can the application recommend foods suitable for users based on the user's behavior and mood?
- How can the application help culinary businesses around the user?

B. Paper Objectives

- How Users can find their preferred food efficiently.
- Users can easily find their preferred food.
- The application can give food recommendations based on the users' behavior and mood.
- The application can help culinary businesses to get customers.

II. METHODOLOGY

In developing this Foody Moody application, we approach the software development process with the waterfall model. The Waterfall Model is the oldest paradigm for software engineering [3]. There are five phases inside it, which are communication, planning, mocking, construction, and deployment which could be found in Fig. 1. This model helps us to manage and break our project into a logical, linear sequential flow with a straightforward analysis and testing process.

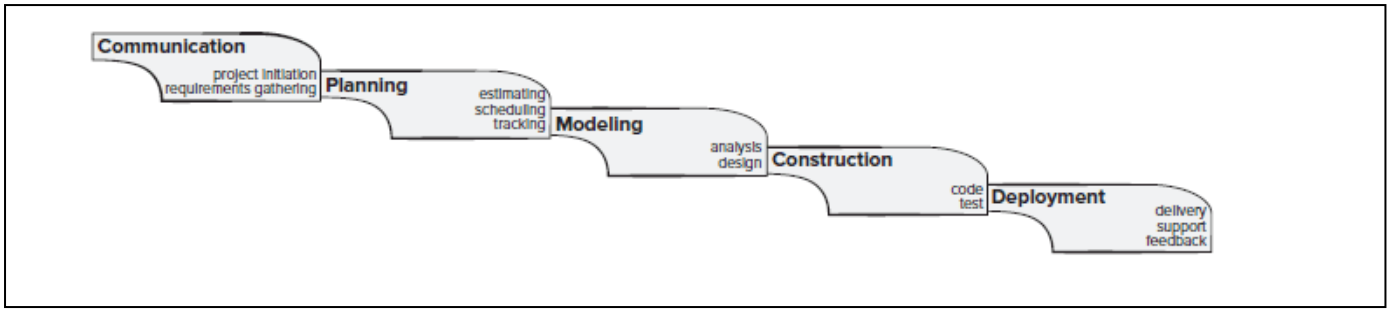


Fig. 1. The Waterfall Model Illustration (Sommerville, 2011)

A. Communication

In this phase, we discuss project initiation requirements gathering. The theme of our project, problem formulation for this project, and our background information will be the main topics for our discussion. At the end of the discussion, Foody Moody has been the theme for the project. Foody Moody is a mood-tracker application recommended for food or drink according to the user input. The Foody Moody application will recommend according to the mood that the user is having, and the user can also choose whether the food or drink is what he wants or not, using a similar swiping system as the dating app that already exists.

B. Planning

In this phase, we began to estimate how long it took to finish the project (estimating) and list the activities within the project (scheduling). To know which team/member has done/hasn't done their tasks, progress tracking is needed (tracking). We timeline our work from the beginning to the end of the project and begin to categorize what has been done, is being done, and what will be done. The final target we have specified is that the Foody Moody application can be used by a large community of Android users in July.

C. Modelling

In this phase, we began to analyze data and design the application. Because the application needs to direct users to their preferred food, it needs to have accurate food data. In the past, there was some research about how mood can determine one's diet. For example, feelings of anger, and depression can be caused by a low carbohydrate diet, while fatigue and depressed mood (which might be caused by iron deficiency) can be countered by taking food with good sources of iron [5]. To further ensure what types of food can counter/enhance a certain mood, we will be giving out surveys to some people.

The application must make a friendly user interface (UI) and has a calm and mood-boosting color palette. User Interface (UI) is a part of an information system that requires user interaction to make input and output [6]. To identify users' needs and tasks, we will be using the Task Centered System Design (TCSD), which consists of:

- 1) *Identification*: This stage will be used to identify users' problems, and tasks that can counter those problems.
- 2) *User-Centered Requirement Analysis*: This stage will analyze the existing problems and determine if the existing results from the identification stage should be implemented or not.

3) *Design the U.S. Scenario*: In this stage, developers will determine the system design process, and data required by the new system. Developers will create a new design through simulation.

4) *Evaluate Walkthrough*: Developers will evaluate the design they have made.

We also made a storyboard which can be found in Fig. 2., which is an interaction design methodology, to demonstrate an end-to-end solution for user scenarios [7]. This helps us to illustrate the flow of our software, as well as get feedback on the early development of Foody Moody.



Fig. 2. Storyboard of Foody Moody

D. Construction

In this construction phase, we began to start developing the application into a physical version—which is an android-based application. Android is a popular mobile operating system that is based on a modified version of Linux and is freely available to be customized and configured [8]. In order to create the app, we use Android Studio which provides a unified environment and structured code modules, allowing us to divide this project into units of functionality. We are using Java as the programming language and Gradle to automatically build the software. We also use Firebase for managing user data in the application database, which is Google-backed development software for creating mobile or web applications. The developed application targets SDK (Software Development Kit) version 32, with minimal SDK of version 21. We also do testing with the prototype every time we add or alter the activity of Foody Moody, to find and fix bugs and errors.

E. Deployment

When it comes to this Deployment phase, we started getting feedback from the people about the Foody Moody apps we had created. They perform several user inputs and try to run a swipe left feature for food or drinks that they don't like and swipe right for food or drink that they like. When they have clicked on the food they want, the application will directly provide the food description and the nearby restaurant that provides the food. They also try other features, including creating accounts in this application, logging in, and managing their passwords. The feedback they gave us eventually became one of the points for us to fix and develop the Foody Moody app.

III. UML DIAGRAMS

The Unified Modeling Language (UML) is a standard specification language for documenting, specifying, and building software systems. In addition, UML is used to create good language models that humans and computers can use.

UML also serves as a bridge in communicating several aspects of the software. Therefore, to make it easier for us to set up and build the Foody Moody application, we designed several UMLs, such as Use Case Diagrams, Class Diagrams, and Sequence Diagrams.

A. Use Case Diagrams

Use Case diagram is a diagram that shows the part of the system where users can interact. The diagram will visually represent various use cases and users of the system.

Use Case functions are as follows [9]:

1. Shows the system's activity process in order.
2. Shows the sequence of activities of a process.
3. Makes it easier for manufacturers to describe how the system works to consumers.

The system in our use case diagram is the Foody Moody Application System itself as shown in Fig. 3, and can also be accessed through the [link](#). There are two actors that interact with the system, which are the user and the admin. Both of the actors need to complete the login and validation in the application, to access further interactions in the system. However, users need to register or create an account first before logging in, which adds the user data into Foody Moody's database. On the other hand, the admin doesn't need to register their account since the admin account is already written in the database and cannot be changed. The actors, both the admin and user, can also log out from the Foody Moody Application System once they are done with their tasks.

First, let's discuss the admin's interactions with the system. The first activity or use case that they can handle is managing the food data, which includes three scenarios: adding food, updating food, or deleting food from the application's data. Hence those three use cases are inherited from their parent class. The admin can also browse or view the handled food data once they're launched to the system. Besides managing the foods data, they can also handle and manage the user database, making it easier to maintain all of the user accounts and data.

Next is the user interactions with the systems. After signing up and logging in, the user can browse and view the food lists according to their mood input, therefore browsing food choices includes the use case of input mood. When the

user finds their corresponding food choice, they can add the food to their favorites by swiping right in the application. Once the food is added to their favorite food list, they have choice to remove the food from the list, and remove favorite food is extended from add favorite food use case. Last, the user can interact with the app to change their email or change their password, which this two-use case are generalized into manage user account use case.

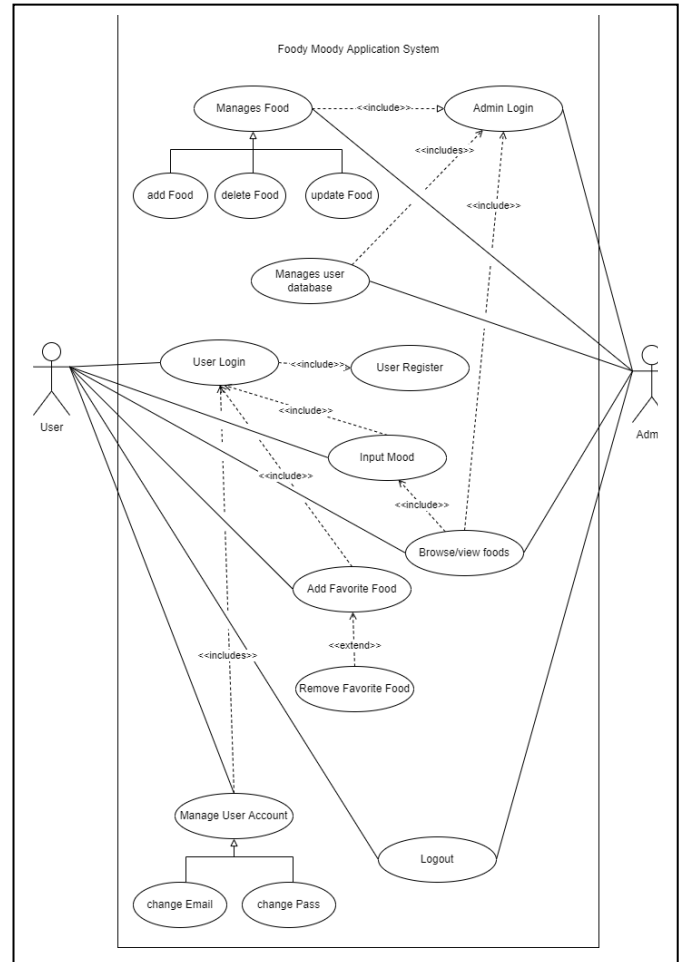


Fig. 3. Use Case Diagram for Foody Moody. [Link HERE](#)

B. Class Diagrams

The class diagram is a diagram that shows the system's structure by providing its classes and their attributes, operation, and relationships.

Class diagram functions are as follows [10]:

1. Clearly show the structure of a system.
2. Can provide an overview of the system and the relationships contained.
3. Improve understanding of the system's general description/program.

The admin class consists of 'adminUsername' and 'adminPassword', which are needed to login into their account. Admin has 3 methods to change the foods' data (addFood to insert new food, updateFood to update a food, and deleteFood to delete existing food).

The food class is an aggregation of the User class and consists of 'foodName', 'foodDescription' and 'foodImageUrl'. The food class has some methods, which are 'setFoodName' (to set food name), 'setFoodDesc' (to set food description), 'setFoodURL' (to set food's image through

URL), 'getFoodName' (to get food name), 'getFoodDesc' (to get food's description), and 'getFoodURL' (to get food's image through URL). 'set' is used to create new/update existing food, while 'get' is used to retrieve the foods' data.

The user class consists of 'userEmail', 'userPassword', 'userMood' and 'userFavorites'. The user has some methods, which are 'Register' (to register new account by inputting email and password), 'Login' (to log into an existing account), changeEmail (to change their registered email), changePass (to change their old password), inputMood (to input the mood to the application), browseFood (to browse for foods), and addFavoriteFood (to bookmark their favorite food).

AngryFood, SadFood, MehFood, and HappyFood are generalized classes of the Food class. Each of them contains an arrayList of their respective foods and has a method to get data from their own arrayList. The Class Diagrams is available in Fig. 4., also can be accessed through the [link](#).

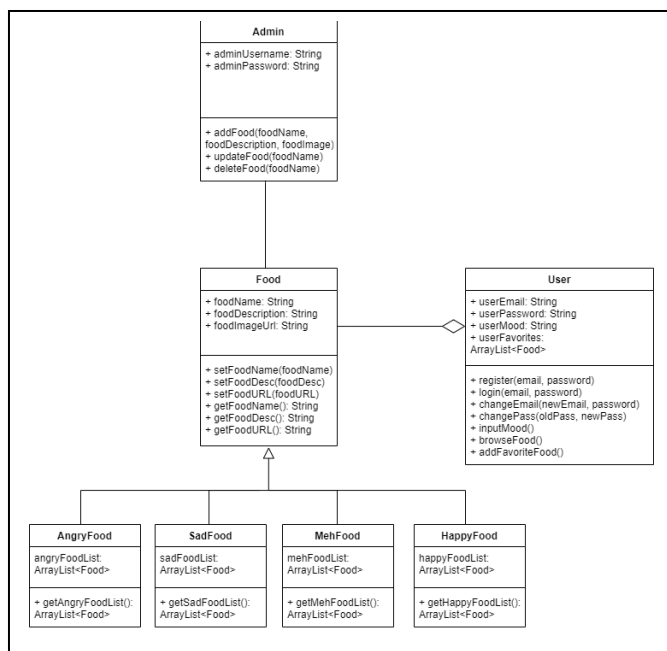


Fig. 4. Class Diagram for Foody Moody. [Link HERE](#)

C. Sequence Diagram

The sequence diagram is a diagram that is used to describe and display the interactions between objects in a detailed system. The sequence diagram also displays the message or command sent, along with the execution time. The process of related objects runs from left to right.

In the Sequence diagram that we have designed in Fig. 5. or through this [link](#), one actor is the user. And five objects, namely Registration, Login, InputMood, showFood, and showFoodDetail. First of all, users register their accounts by filling in the registration form contained in the application. After that, the system will send an email to the user, and the user can verify the account. After successful account verification, the user logs in according to the registered email. The system will validate the user login first, and if successful, the system will show the mood input feature. The user can input the mood according to what he is feeling. Then, the system will provide the results from the user input and display a list of foods according to these results. Users can directly choose the food they like or don't like. When the user has

swiped right, which is the food they like, the system will provide the details of the food.

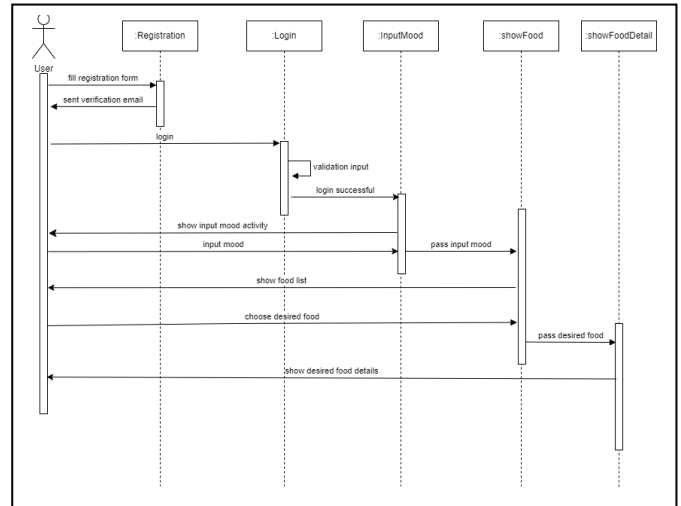


Fig. 5. Sequence Diagram for Foody Moody. [Link HERE](#)

IV. APPLICATION DEVELOPMENT

A. UI/UX Development

For appearance and UI/UX, we develop it together with application development, so that we can determine which features can be realized and which are not so that we can customize the UI/UX. Here is what our application looks like:

1. Register Page

This page is a page where users can register their account by filling in the email, password, and confirm password fields. If the user already has an account, the user can immediately click 'Sign in', and will be directed to the login page.

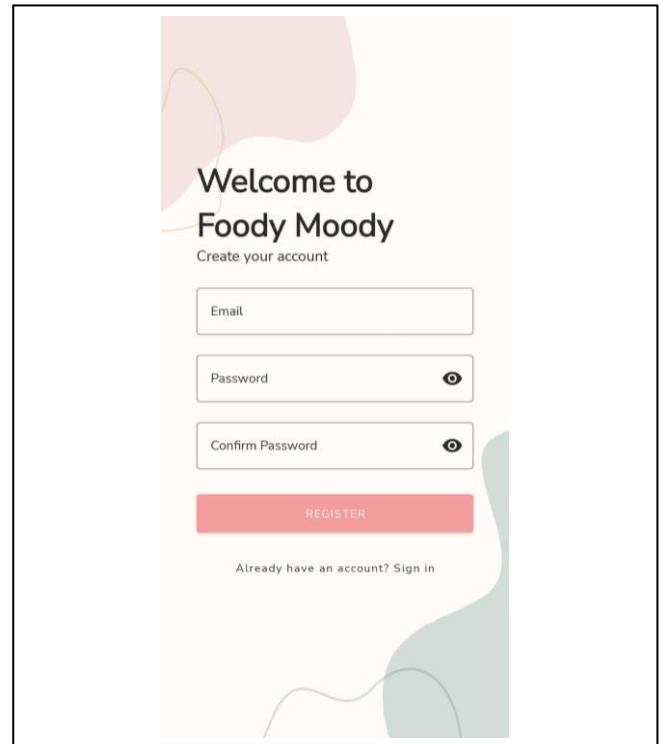


Fig. 6. Register Page

2. Login Page

On this page, users can directly login into their registered account. If the user forgets the password that has been created, the user can reset the password by directly clicking 'Forgot Password'.

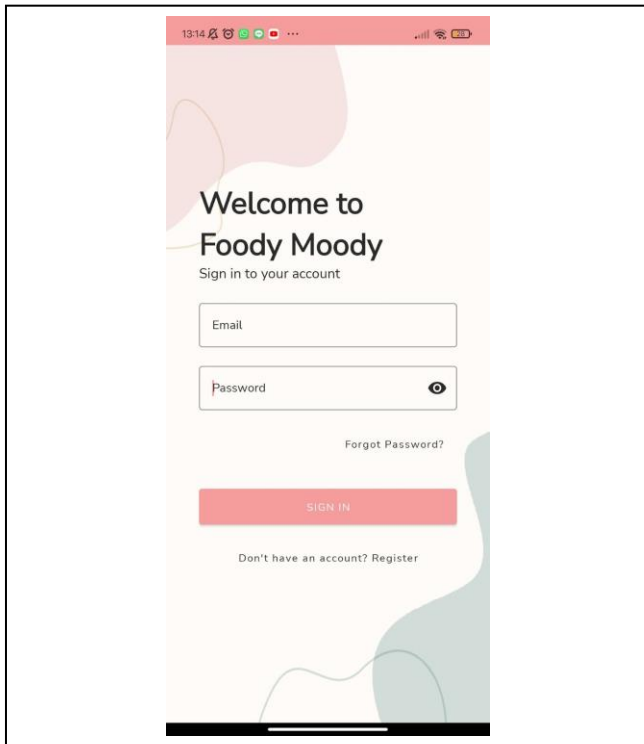


Fig. 7. Login Page

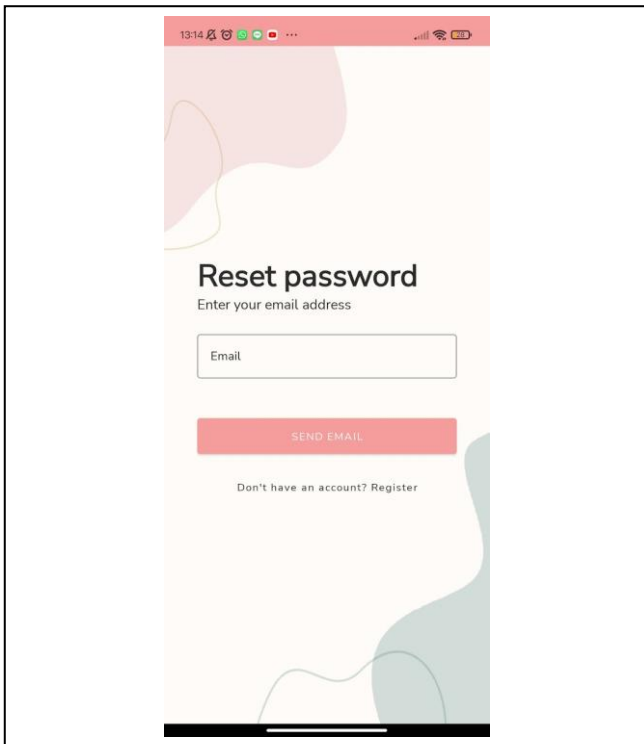


Fig. 8. Forget Password Page

3. Forget Password Page

On this page, users can reset their password by entering their registered email. The password can be reset directly via the email that will be sent by the application.

4. Home Page

On the main page of the Foody Moody application, there will be a user input pop up to detect the mood that the user is currently feeling. There are currently four moods in our database; happy, sad, angry, and normal.

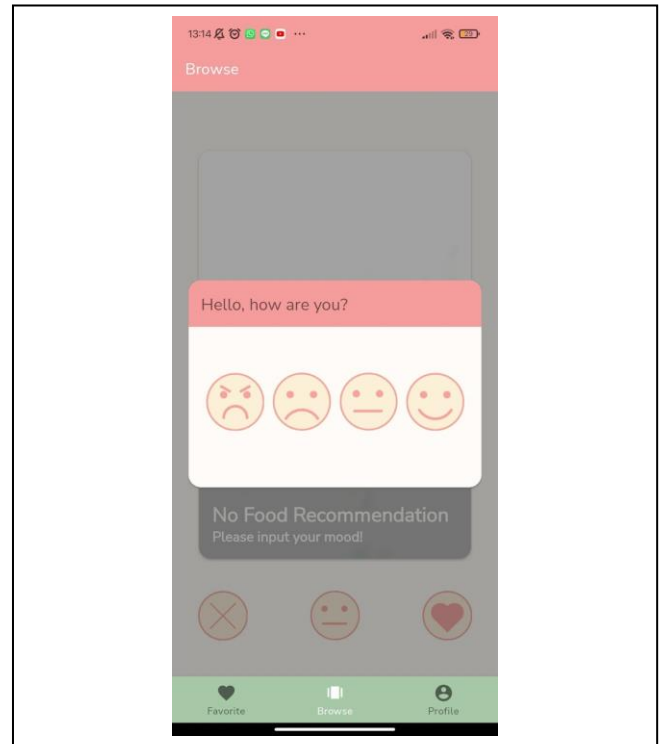


Fig. 9. Home Page

5. Browse Page

After the user inputs their mood, the application will start recommending the appropriate food. There is also an explanation about the food. The swipe left and swipe right features can also be done now; when the user doesn't like the food/drink, the user can swipe left. On the other hand, swipe right when the user likes the food/drink recommended by the app.

In this feature, we provide an opportunity for restaurants to advertise their food on our application, so that it can be displayed regularly.

6. Like Page

This page will appear when the user swipes right on the recommended food/drink. On this page, the application will provide a complete description of the food they like, explanations, calories, and also potential allergies. After that, we will also show the closest restaurants that provide these foods. As for the paid Google Places API, the restaurant data displayed is only dummy data.

With this feature, we open up opportunities for restaurants to advertise their restaurants on this page so that they can gain more recognition among Foody Moody users.

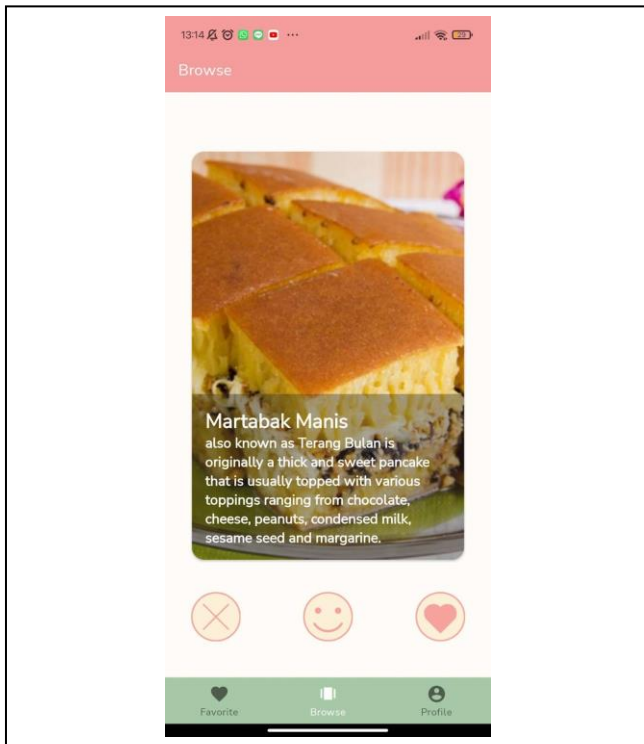


Fig. 10. Browse Page

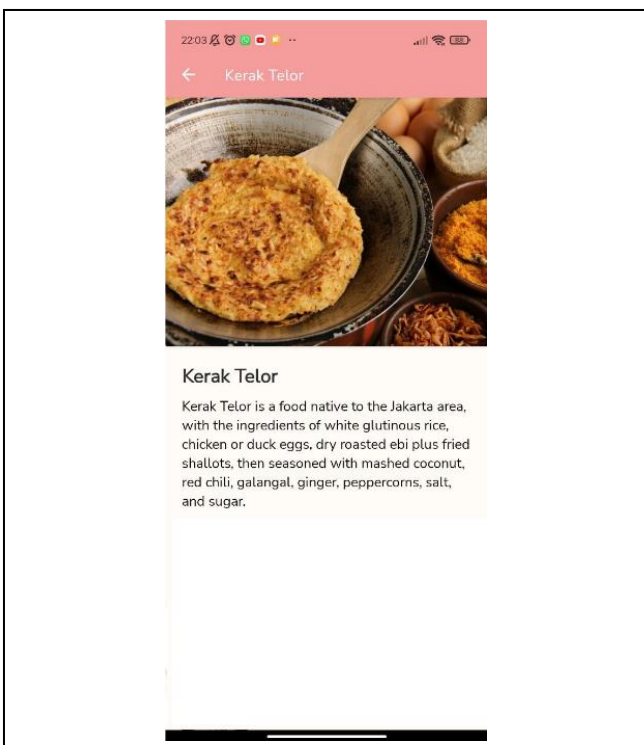


Fig. 11. Like Page

7. Profile Page

The figure in Fig 12. is a user profile page where users can change their email and password, contact

Foody Moody, promote food, and fill out an assessment for the application.

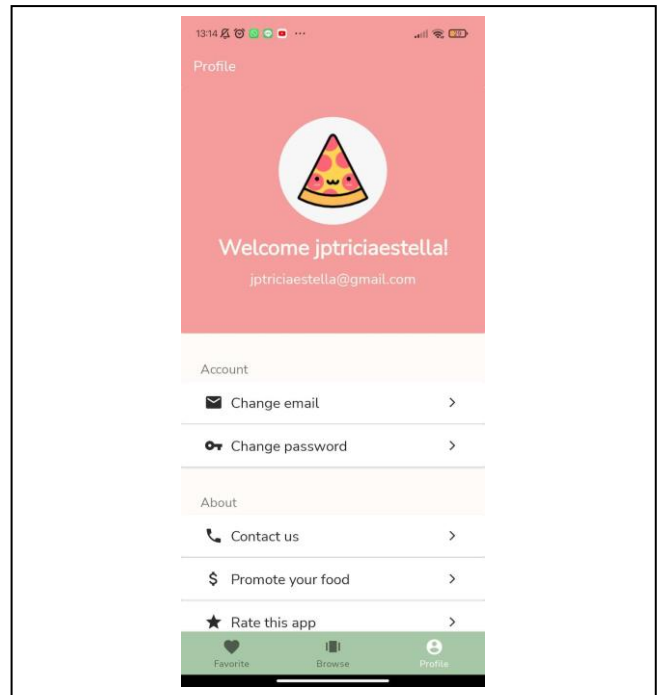


Fig. 12. Profile Page

8. Reset Password Page

This page will appear when the user wants to reset their password after successfully logging in. If the user forgot the current password, the user can click 'Forgot Password' and will be directed to the start page where they need to enter the registered email.

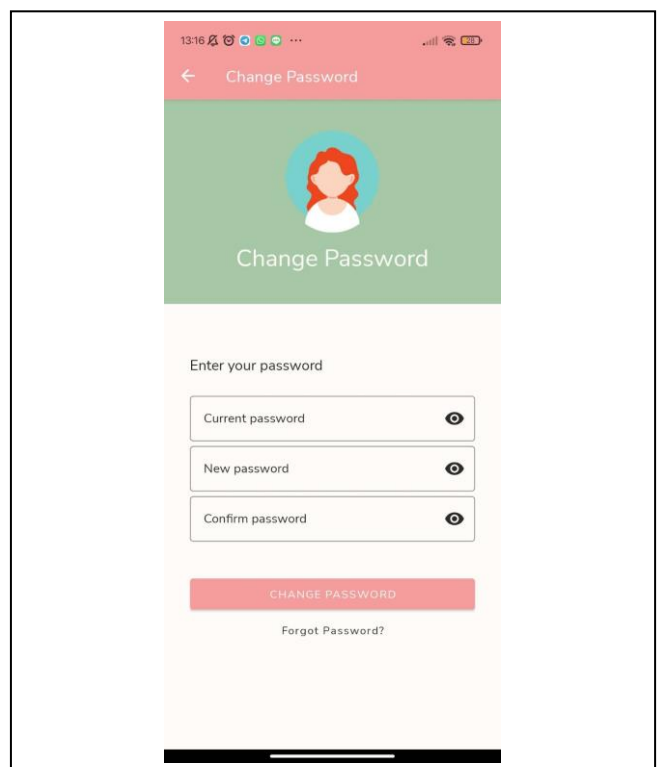


Fig. 13. Reset Password

9. Change Email Page

To change the registered email, the user needs to input the correct password. If the password does not match, the email cannot be changed.

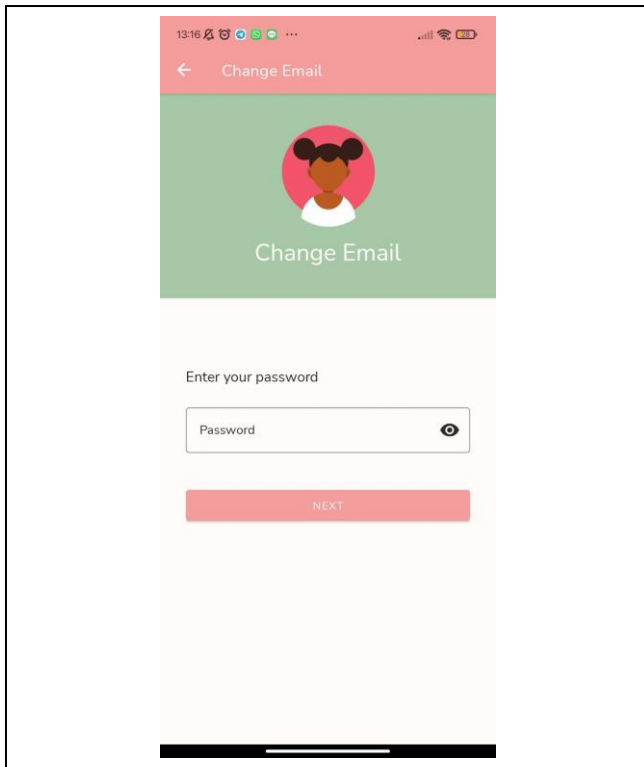


Fig. 14. Change Email Page

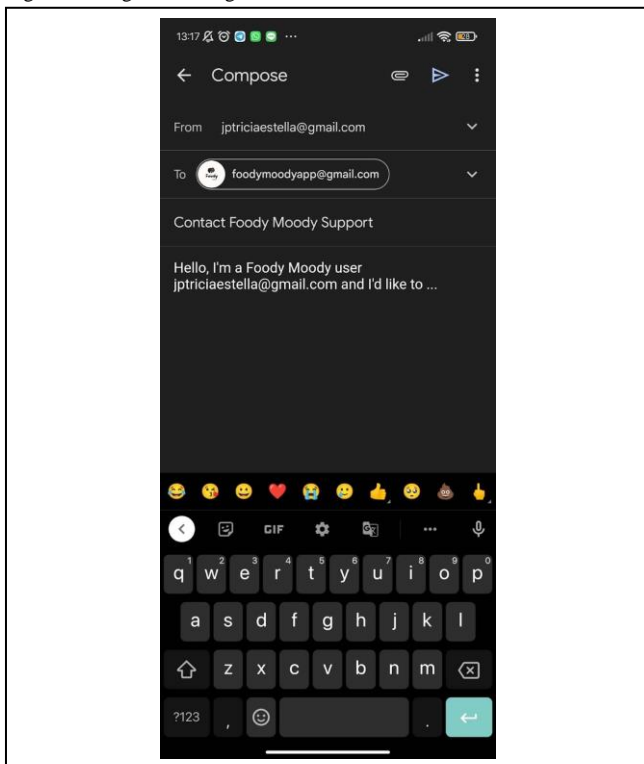


Fig. 15. Contact Us Page

10. Contact Us and Promote Your Food Page

On this page, the user will be redirected to their email application. When they press the appropriate button

on the profile page (send), they will send an email to our Foody Moody email (foodymoodyapp@gmail.com).

11. Logout Page

As shown on Fig. 16., logout page is the page where the user wants to log out from their account in the Foody Moody application.

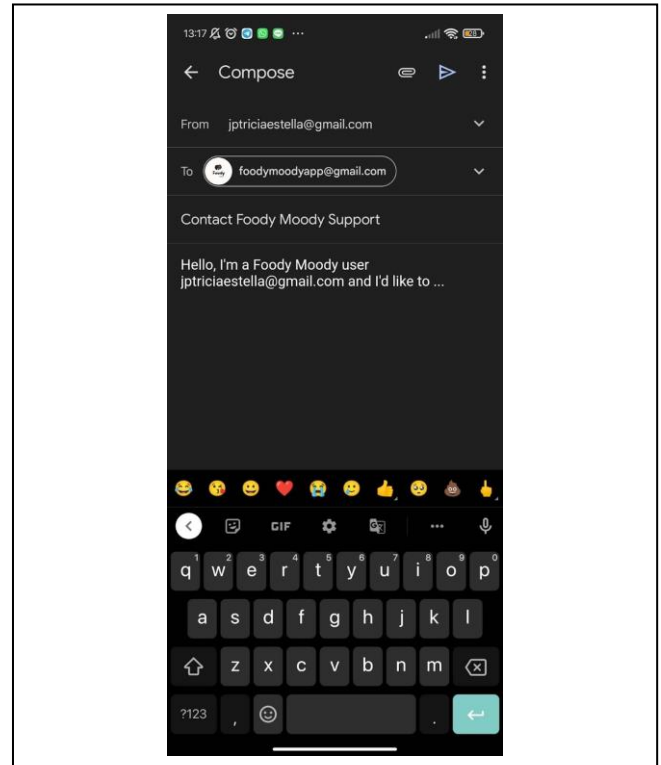


Fig. 16. Logout Page

12. Favorite Page

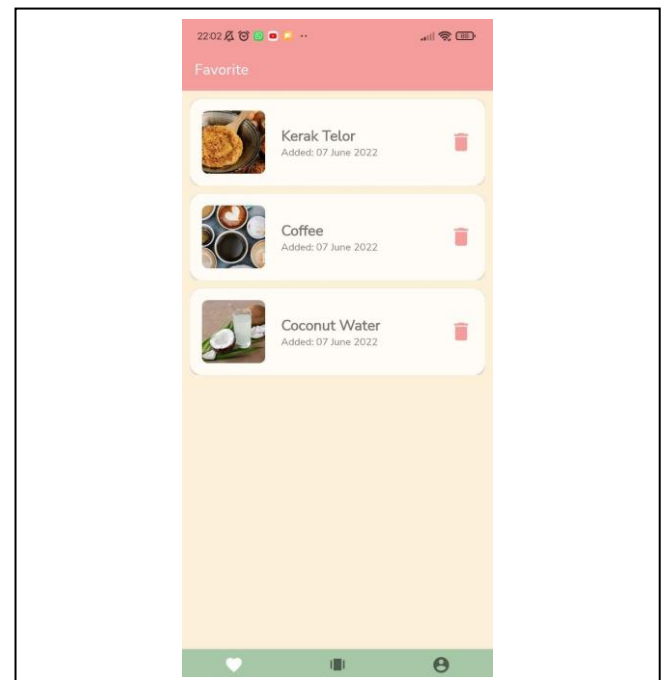


Fig. 17. Favorite Page

This page will display every food that the user has liked (swipe right) so that the user can revisit their food history. Users can delete food from their data, or view food information.

B. Building the prototype

As stated before, Foody Moody is an Android-based application, thus we use Android Studio as the platform in developing our application. Java is the main programming language that we use, and the developed app targets SDK (Software Development Kit) version 32, with a minimum SDK version of 21.

1. Android Studio

Because we use Java as a programming language, we created this application using the object-oriented programming method. This application consists of data and model classes, namely what objects are in the application, a ui controller class to control each existing view/page, and the layout of the application itself (views).

Our data consists of classes Food and CardStackAdapter, where food has attributes such as name, description, and image. In the future, we will add attributes such as calories and allergens, according to the input of our respondents in the previous prototype development. CardStackAdapter is a model that we use to adapt an object into a card on the browse page, where we use the CardStackView repository from Yuyakaido's Github to implement swiping cards such as dating apps.

For the model itself, our application consists of AngryFood, HappyFood, MehFood, and SadFood, where "Angry", "Happy", "Meh", and "Sad" are the moods that we provide as options in the application. This model will be used to store food data according to the selected mood.

The UI and view controller parts consist of Browse, ChangeEmailActivity, ChangePasswordActivity, Favorite, ForgotPasswordActivity, HomeActivity, LoginActivity, MainActivity, Profile, RegisterActivity, and VerifyAccountActivity, it is already running well and can be seen in the explanation in the UI/UX section above.

2. Firebase Database

In the Foody Moody application that we have created, our backend or database is running and usable, whereas we use the Firebase service provided by Google for small-scale projects. Firebase helps us save user accounts, and services related to other accounts, such as "Forgot Password", "Verify Email", or "Reset Email or Password". In the future, unfortunately, we cannot use Firebase to store user favorite data, due to configuration and payments that must be made. So, we only save the user's favorite food locally for our prototype now.

V. FEEDBACK

After having a working prototype of the application, we ask for input from users to find out what features are still lacking

or need to be added. We distributed a complete questionnaire with installation and usage guides, then we were able to measure user satisfaction based on the responses from their questionnaires. Here is our feedback grid after getting a sufficient number of respondents:

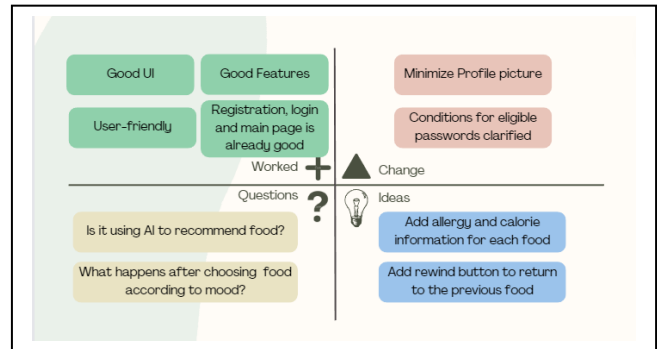


Fig. 18. Feedback Grid

VI. FINAL PRODUCT

Through the feedback we received, we know what needs to change and improve in the final version. This includes adding a rewind feature to card swiping, adding calorie information, reducing the profile picture, and so on. The following is an example of how the application looks after getting feedback, such as adding a rewind button and complete information on food and its calories and restaurant:

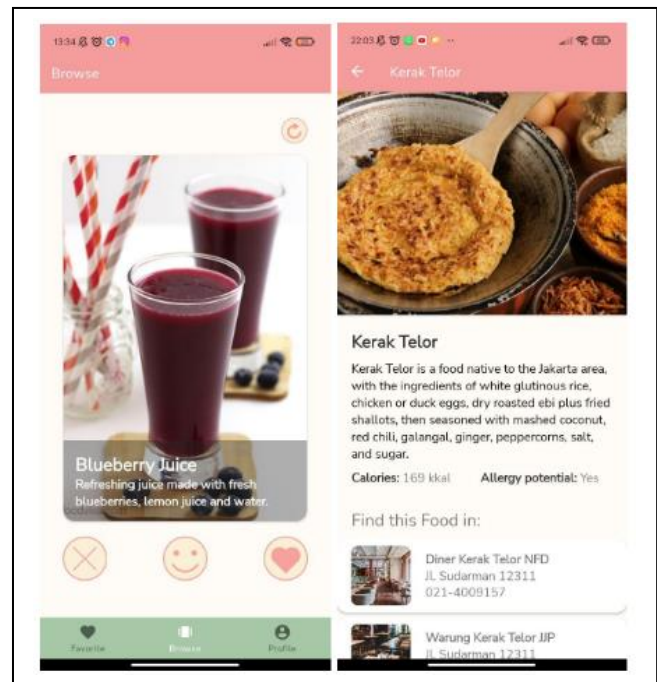


Fig. 19. Fixed Prototype after feedback

When we feel that our prototype is complete both in function and appearance, we are ready to deploy our final prototype as a product from Foody Moody.

Here is the link to our prototype application: <https://bit.ly/FoodyMoodyApp>. Applications can be downloaded and installed into Android devices with a minimum SDK version 21. For the code itself, you can find the Foody Moody Project on <https://github.com/jptriciaestella/FoodyMoodyApp.git>

Here is the complete video link for the Foody Moody application demo: <https://youtu.be/s9n8j66XUeU>.

VII. CONCLUSION

Although some of the features are still locked behind a paywall, the main features in the app are already working well in the finalization stage. Foody Moody application will be the latest solution in helping to solve the problem of improving the mood or mood of its users. With features that will continue to develop and UI/UX that will always be evaluated following technological developments, this application will survive in its market. We hope that through this application, we can help users who have difficulty in choosing their desired food, and help restaurants by promoting their food.

REFERENCES

- [1] AlAmmar, W. A., Albeesh, F. H., & Khattab, R. Y. (2020). Food and Mood: the Corresponsive Effect. *Current Nutrition Reports*. doi:10.1007/s13668-020-00331-3 (<https://doi.org/10.1007/s13668-020-00331-3>)
- [2] Gardner, M. P., Wansink, B., Kim, J., & Park, S.-B. (2014). Better moods for better eating?: How mood influences food choice. *Journal of Consumer Psychology*, 24(3), 320–335. doi:10.1016/j.jcps.2014.01.002 (<https://doi.org/10.1016/j.jcps.2014.01.002>)
- [3] Sommerville, I. (2011). *Software Engineering* 9th ed, Pearson, Addison Wesley United States of America, ISBN 13: 978 0 13 705346 9.
- [4] Petersen, K., Wohlin, C., & Baca, D. (2009). The Waterfall Model in Large-Scale Development. 386. doi: 10.1007/978-3-642-02152-7_29 (https://doi.org/10.1007/978-3-642-02152-7_29).
- [5] Ottley, C. (2000). Food and Mood. *Royal College of Nursing*. doi: 10.7748/mhp2000.12.4.4.32.c1684 (<https://doi.org/10.7748/mhp2000.12.4.4.32.c1684>).
- [6] Indriana, M., Adzani, M. (2017). UI/UX Analysis & Design For Mobile E-Commerce Application Prototype on Gamedia.com. Universitas Multimedia Nusantara. Doi: 10.1109/CONMEDIA.2017.8266051
- [7] Andriole, S. J. (1989). *Storyboard prototyping: a new approach to user requirements analysis*. QED Information Sciences, Inc.
- [8] DiMarzio, J. (2016). *Beginning Android Programming with Android Studio*. John Wiley & Sons, ISBN-13: 978-1118705599.
- [9] Juliarto, R. (2021, May 9). Contoh use case diagram Lengkap Dengan penjelasannya. *Dicoding Blog*. Retrieved April 16, 2022, from <https://www.dicoding.com/blog/contoh-use-case-diagram/>
- [10] Setiawan, R. (2021, September 16). Memahami class diagram Lebih Baik. *Dicoding Blog*. Retrieved April 16, 2022, from <https://www.dicoding.com/blog/memahami-clas>