FRESH REMINDER

PROJECT PROPOSAL

Supervisor:
Shekhar Kalra

Authors:

Matthew Soulsby s3784709@student.rmit.edu.au

Vincent Villaflores s3728807@student.rmit.edu.au

GitHub:
https://github.com/rmit-iPSE-s2-2023/a1-s3784709

TABLE OF CONTENTS

PROJECT CHARTER

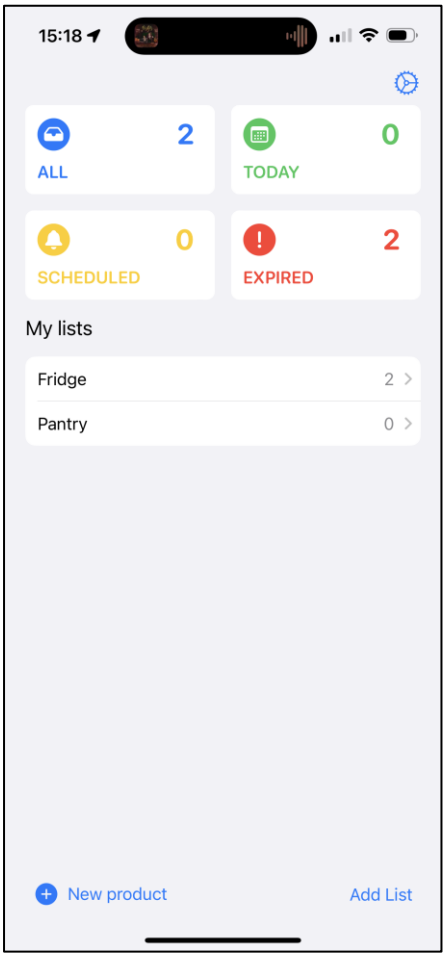
1. INTRODUCTION

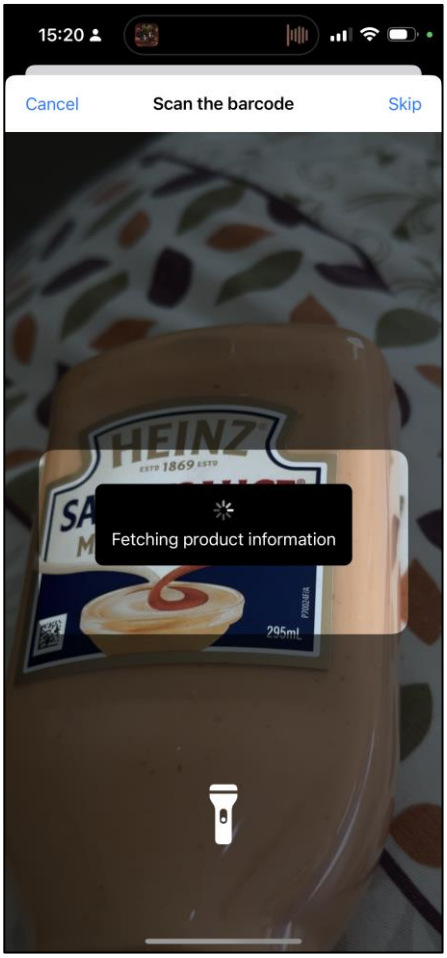
Every household goes grocery shopping. It is an important chore that feeds the entire family. The average Australian household spends $7904 annually at the supermarket (Dean 2023). It is no surprise that such a large sum is spent on such a daily necessity. Without it, we would struggle to get enough energy and nutrients to survive a full day. 2.5 million tonnes of food are wasted every year and reducing this could save the average Australian household up to $3800 annually (Clean Up Australia n.d.). Being mindful of food waste could help save a significant amount of money.

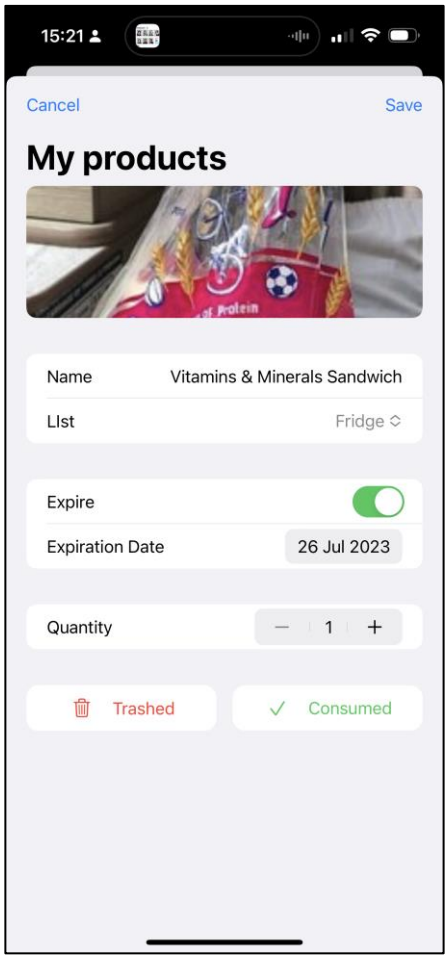
Food waste is the issue that we are trying to alleviate. It is easy to forget about what is in our fridge. With our app we aim to help reduce this waste by providing the user an easy-to-use and simple interface that will help track and notify them as the expiration date approaches. The main feature of this app will be its use of machine learning to determine what the user has taken a photo of. The app will then automatically add the item to the list with the appropriate expiration date.

2. SIMILAR APP REVIEW

2.1. FRIDGE BUDDY



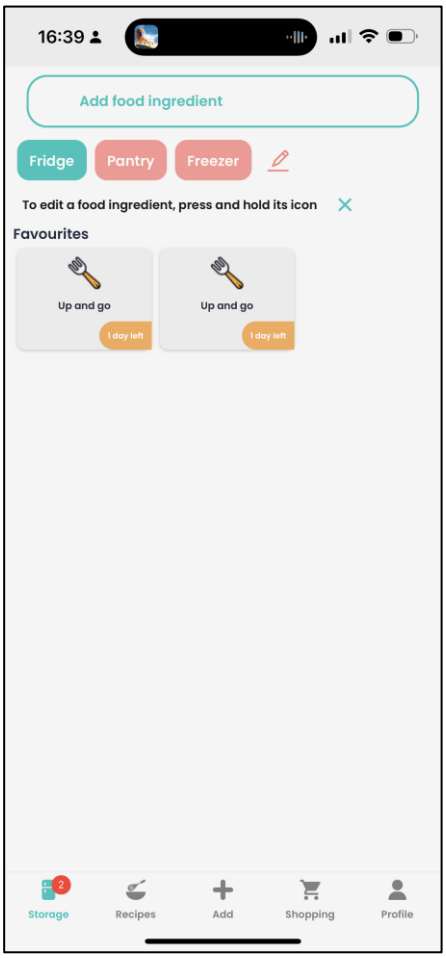


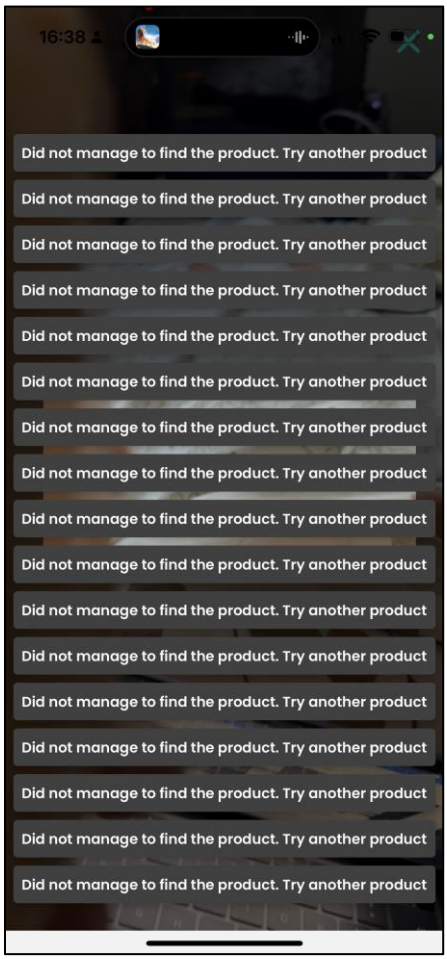


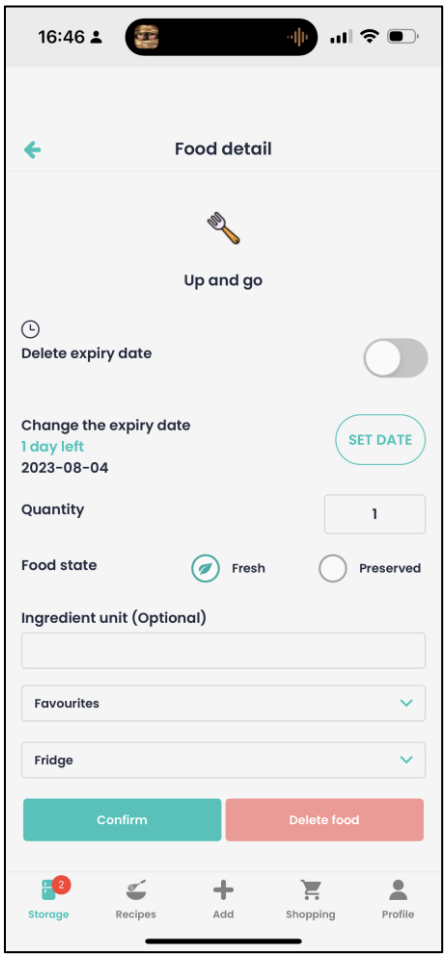
- Requires barcode scanning; automatically adds product name but not the expiration date.
- “New product” button being in the bottom left corner makes it hard to press when users are holding their phones with one hand, especially if they have a bigger phone.

- You do not get any information in the home screen; you need to press one more button to see the items in the lists.
- When scanning the barcode, the “Cancel” and “Skip” buttons are in the top left and right corners, making them hard to press.
- This looks like the Apple Reminders app.

2.2. EMPTYMYFRIDGE

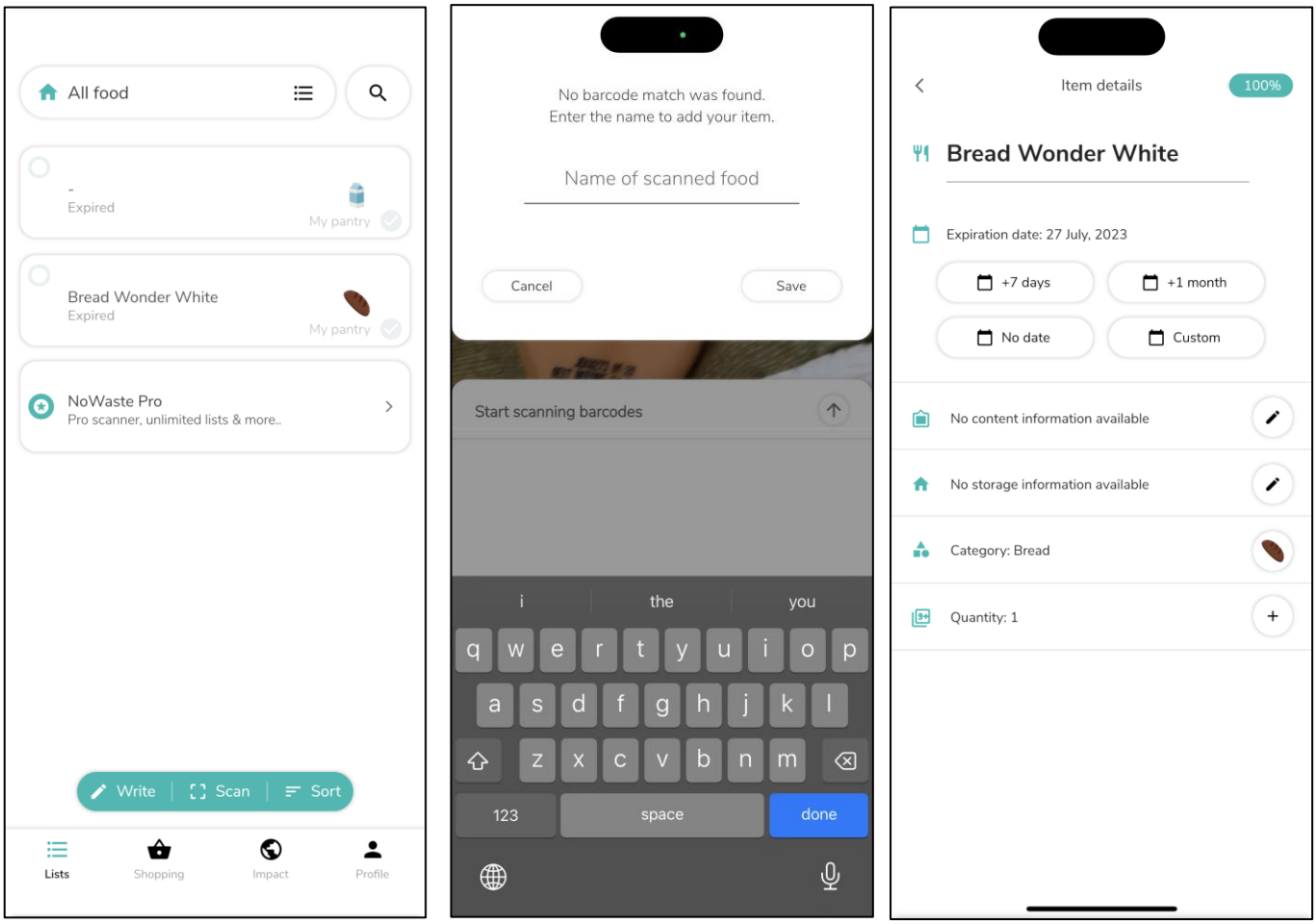






- Requires barcode scanning; automatically adds product name but not the expiration date.
- “Add” tab gives you the options to manually input the information or to scan a barcode.
- When barcode is unrecognisable, it can spam the screen with error messages.
- When scanning the barcode, the close button in the top right corner makes it hard to close the scanner and is not pressable on an iPhone 14 pro max.

2.3. NOWASTE



- In the home screen, it gives you two options to manually add products or to scan a barcode which can be easily reached in most scenarios.
- When the barcode cannot be recognised, it gives you a prompt so that you can easily add the name of the food.
- You can easily extend the expiration date of a product when you view an individual product.

2.4. COMPARISON

Consideration	**Fridge Buddy**	**emptymyfridge**	**NoWaste**
Visual Appeal	Basic and Simple	Can feel a bit cluttered, with too much information	Looks polished and information is presented neatly
UI Elements	Key buttons can be hard to press	Screens can be littered with too many elements	Most elements are reachable and intuitively placed
Navigation	Clear sections but are hard to reach since they are at the top of the screen	Sections in the bottom tab are clearly labelled but icons may be too small	Tabs are clearly labelled and given an appropriate amount of space

Consistency	All screens have consistent fonts, colours, and styles	All screens have consistent fonts, colours, and styles	All screens have consistent fonts, colours, and styles
Clarity and Readability	Texts are legible	Texts can be small in some areas; making it harder to read	Texts are legible
Gestures and Interactions	Gestures and Interactions are like native Apple apps	Gestures and Interactions are simple but are slightly different from native Apple apps	Gestures and Interactions are like native Apple apps
Feedback and Animation	Feedback and animations are consistent with native Apple apps	Minimal feedback and animations are present	Feedback and animations are consistent with native Apple apps
Error Handling	No error handling if scanner does not scan a barcode	Error handling can spam the screen and may persists indefinitely	Error handling is handled nicely, providing an easy alternative
User Flow	Provides a straightforward and minimalistic path for users to navigate	Provides an abundance of features that may provide to be challenging to traverse	Simple yet thought out, creating a seamless and intuitive experience
Learning Curve	Minimal learning curve	Minimal learning curve	Minimal learning curve
Machine Learning	NONE, need to manually enter data	NONE, need to manually enter date	NONE, need to manually enter date
Overall User Experience	Simple	Featureful but cluttered	Polished and intuitive

3. APPLE'S HIG RECOMMENDATIONS

3.1. COLOUR SCHEME

The colour scheme that we have decided to go with is Green and White. We decided to use Green because of its association with fresh food and longevity. The colour Green is associated with fresh and healthy foods, and this is emotion we are trying to communicate when we want to remind our users of the freshness and edibility of their grocery items. Green also represents well-being and vitality. We also want to subtly encourage healthy and responsible food choices. Green is also a soothing colour and since food expiration can be stressful, we want to soothe the user with a sense of control and ease when tracking their groceries.

We chose White for its cleanliness and simplicity. White exudes a sense of organization and order, which helps with the representation of our app. Having too much colour can look too cluttered, which can make communication harder and distract our users.

3.2. WRITING & TYPOGRAPHY

The typeface family that we will be using is the Apple system font San Francisco. This text is simple and legible typeface that fits the simplicity theme that we want to employ in our app. We will be avoiding thinner fonts as they can be difficult to read in certain scenarios. System fonts also support Dynamic Type which will allow our users to choose the right text size for them.

3.3. LAYOUT

Apple HIGs Best practices List:

1. Use placement to convey relative importance. People often start by viewing items in reading order.
2. Make essential information easy to find by giving it sufficient space.
3. Create visual groupings to help people find the information they want.
4. Use alignment to ease visual scanning and to communicate organization and hierarchy.
5. Consider providing visual hints to help people discover content that's currently hidden.
6. Make interactive components easy to discover by providing enough space around them.
7. Prefer displaying text in a list or table.
8. Keep item text succinct so row content is comfortable to read.
9. Choose a table or list style that coordinates with your data and platform.
10. Choose a row style that fits the information you need to display.

The Apple HIGs principle that was considered for each design choice will be contained in "(“and “)" and separated by ",".

3.3.1. CAMERA

Functionality: To be able to capture photos of grocery products

Design Choices:

- Large Camera Display: Users can easily view the product they want to capture. (1, 2)
- Black Bars: Users can easily focus on capturing the photo. (3)
- Large Shutter Button: Users can easily press the button in any situation. (6)
- Large Flash Button: Users can easily capture photos in the dark. (6)
- Wheel Menu: Users can easily navigate to other pages. (6)
- Next Expiration Date: Users can easily view vital information without being distracted from primary functionality. (1, 2)

The most important feature of this screen is that the user has enough space to ensure they can take a great photo of the grocery product. This is reflected in the screen real estate given to the camera display. Every other feature has been designed as to not distract from this primary feature. The white colour provides a nice contrast against the Black Bars whilst not being distracting.

3.3.2. CALENDAR

Functionality: To be able to gauge the proximity of expiration dates of products in the current month

Design Choices:

- Calendar Table: Users can easily scan and read the text. (4, 7, 8, 9, 10)
- Drop-down Menu: Users can filter the Calendar based on their preferences. (6)
- Grey Today Square: Users can identify the current date. (3)
- Red Expiration Count: Users can tell how many items are expiring on a given day. (2)

This screen provides a neat reminder on the expiration dates of various grocery products. The central alignment helps ease of reading and accentuates the simplicity and usefulness of this screen. The calendar portion of the screen will be a custom layout.

Initial Design:

3.3.3. FRIDGE

Functionality: To be able to view all grocery items on one page

Design Choices:

- Grocery Categories: Users can easily identify each category. (2, 3)
- Indented Items: Users can identify which products belong in which categories. (3, 4, 10)
- Negative Space: Users can associate the expiration dates with products. (3)
- Emoji Icon: Users have another way of identifying the product. (2)

Although this screen is not as neat as the Calendar screen, it is packed with information. This screen presents every single item in a simple and intuitive manner. Being mindful of having too much text, this screen can give the user a complete run down of their groceries.

3.3.4. DATE

Functionality: To be able to view all grocery items that expire on the same day on one page

Design Choices:

- Grocery Categories: Users can easily identify each category. (2, 3)
- Indented Items: Users can identify which products belong in which categories. (3, 4, 10)
- Negative Space: Users can associate the quantity with products. (3)
- Emoji Icon: Users have another way of identifying the product. (2)

This screen is like the Fridge screen except you only see the products expiring on a certain day. To substitute the expiration date for each item, we present the quantity left. Unlike the Fridge screen, this screen has a greater sense of urgency, so providing the quantity remaining fits the theme.

3.3.5. ITEM

Functionality: To be able to view details about a single grocery product

Design Choices:

- Name: Users can easily identify the grocery product. (2, 3)
- Indented Items: Users can identify which products belong in which categories. (3, 4, 10)
- Negative Space: Users can associate the quantity with products. (3)
- Emoji Icon: Users have another way of identifying the product. (2)
- Modify Buttons: Users can modify any information easily. (6)

3.3.6. LOADING

The Loading page has our logo on it in the middle of the screen.

3.4. NAVIGATION BAR

The navigation bar consists of four buttons with text labels. To prevent the buttons from looking indistinguishable, we spaced the buttons out evenly. This will ensure that the users have enough space to press each button. The current screen in the Navigation Bar will be greyed out so that the user knows what page they are on.

4. WIREFRAMES & WORKFLOW

The first screen that the user will see is Figure 1. This is the loading screen for our app and will provide us some buffer time to complete any background tasks.

The second screen that the user will see is Figure 2. This is our Camera screen. This screen provides the primary functionality of this app. The user will be able to capture an image of their grocery product and a machine learning program will automatically identify the product and create a reminder for it. Each photo the user takes will be added to a queue that the machine learning program will process. This way the user does not have to wait for the machine learning program to finish. From this screen the user can navigate to the Fridge, Calendar, or Settings screen.

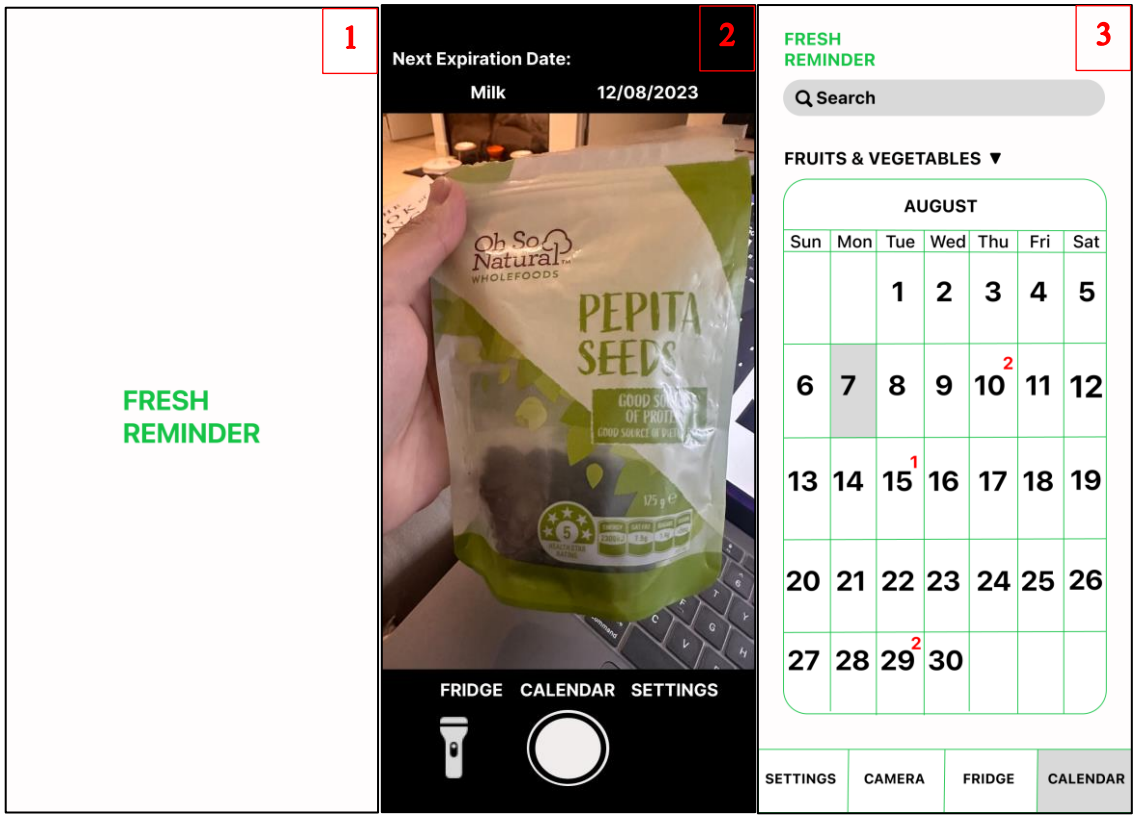
The Calendar screen, figure 3, provides a simple view of the current month's expiration dates. The user can easily tell how many of their products will expire soon and be able to plan their meals accordingly. The user can also filter the calendar based on the grocery category they want, see Figure 4 and 5. If the date has a counter the user can click on that date and open a new page that has all the items that expire on that date.

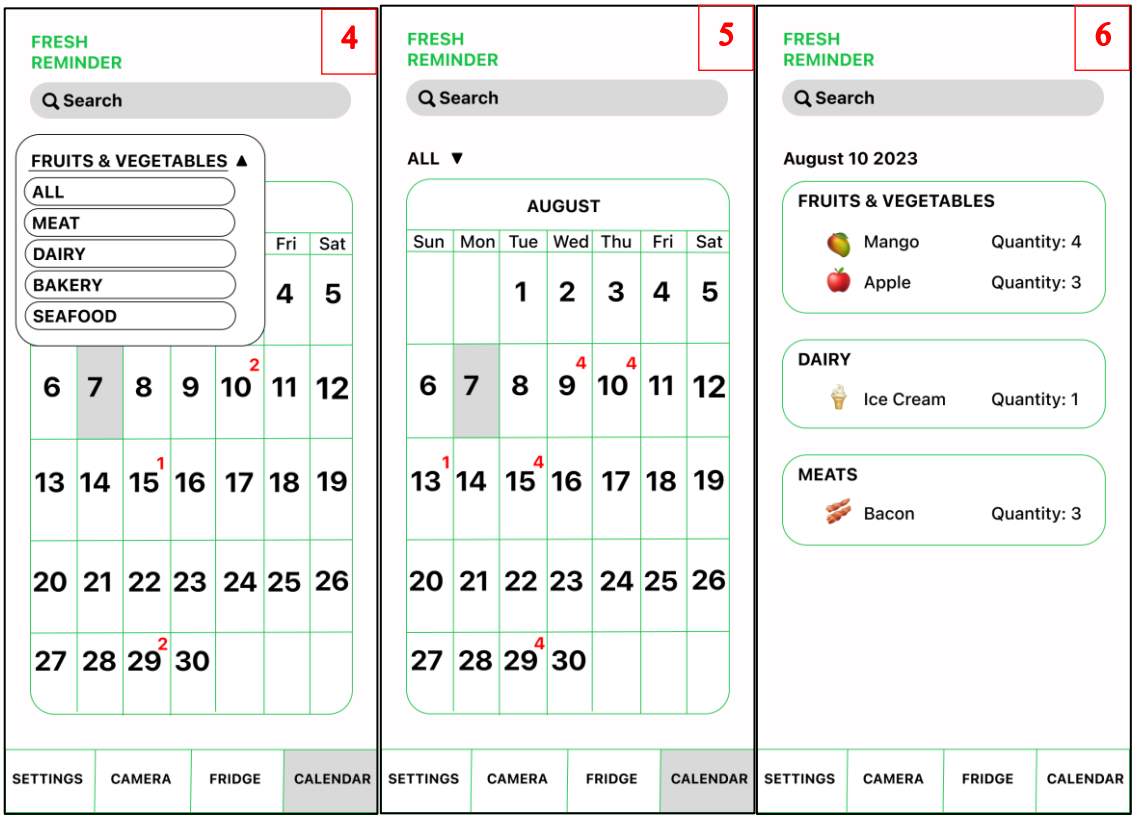
The Date Screen, figure 6, provides a list of all the grocery products that will expire on that date. These grocery products are grouped by their grocery category and the quantity of each product is listed. This

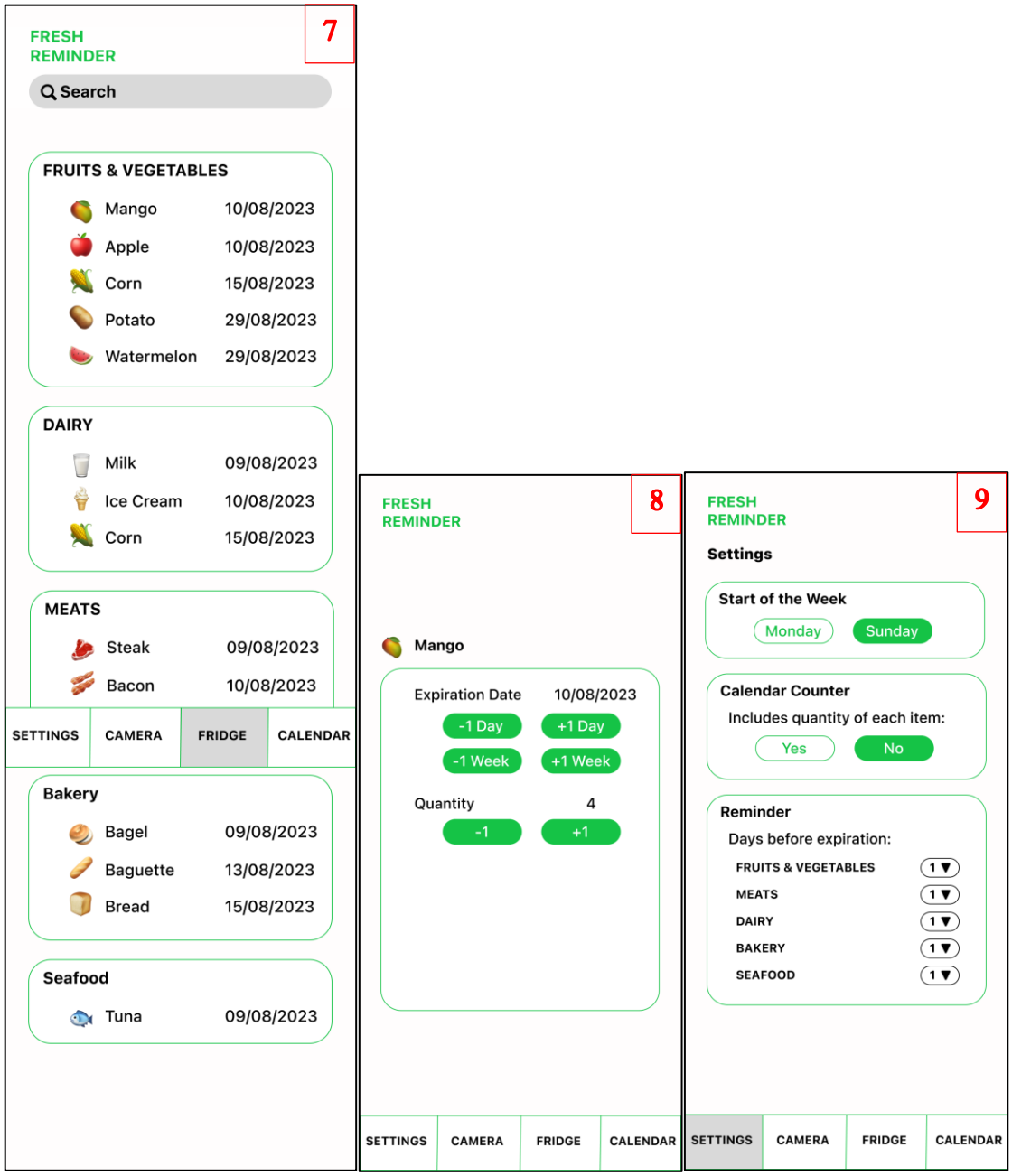
screen works with the Calendar screen to provide the user with more information about a particular date so they can plan their meals better.

The Fridge Screen, figure 7, provides a complete list of all grocery products. These grocery products are grouped by their grocery category and the expiration date of each product is listed. This gives the user a screen where they can view all their grocery products in one page.

The Item Screen, figure 8, can be accessed by clicking on the individual item on either the Fridge Screen or the Date screen. This screen provides the user with a view to be able to easily modify the expiration date or the quantity of a product.




4
FRESH REMINDER
Search
FRUITS & VEGETABLES ▲
ALL
MEAT
DAIRY
BAKERY
SEAFOOD
Fri Sat
4 5
6 7 8 9 10 11 12
13 14 15 16 17 18 19
20 21 22 23 24 25 26
27 28 29 30
SETTINGS CAMERA FRIDGE CALENDAR
5
FRESH REMINDER
Search
ALL ▼
AUGUST
Sun Mon Tue Wed Thu Fri Sat
1 2 3 4 5
6 7 8 9 10 11 12
13 14 15 16 17 18 19
20 21 22 23 24 25 26
27 28 29 30
SETTINGS CAMERA FRIDGE CALENDAR
6
FRESH REMINDER
Search
August 10 2023
FRUITS & VEGETABLES
Mango Quantity: 4
Apple Quantity: 3
DAIRY
Ice Cream Quantity: 1
MEATS
Bacon Quantity: 3
SETTINGS CAMERA FRIDGE CALENDAR



5. DATA MODELS

There are four different kinds of data that we need to keep track of:

User Profile Data: Information of the user, name, email, and preferences.

Grocery Product data: This includes details of the product, name, category, expiration date, quantity.

Notification Data: Details about upcoming expiration dates, the product information and any user preferences.

Image Queue: A queue of images that the machine learning program will process to automatically create new Grocery Product and Notification data.

The User Profile, Grocery Product, and Notification Data can all be stored locally. An SQLite database will be used because it is suitable for mobile apps and offers offline data access.

The Image Queue will be stored in an Amazon AWS S3. The images will be stored in a queue until the machine learning program has processed the information.

There will be three tables that we will create.
User Table:
- UserID (Primary Key)
- Name
- Email
- NotificationPreferences

Product Table:
- ProductID (Primary Key)
- UserID (Foreign Key)
- ProductName
- ProductCategory
- ExpirationDate
- Quantity

Notification Table:
- NotificationID (Primary Key)
- UserID (Foreign Key)
- ProductID (Foreign Key)
- NotificationPreferences

6. REST API

The rest API which we will use with the application will be the following: https://github.com/jcomo/shelf-life. This API provides expiration dates for a variety of food items, which is the most significant informational hurdle for our application. Hardcoding each of these values into a local database would be time-consuming and fraught with potential error.

Whilst the API is no longer running on Heroku (in spite of what the readme suggests), it can be run locally through building the source code. In the event that the ability to query the API from outside our local network is needed, the API can be run on an EC2 instance in AWS's free tier (with appropriate start-stop times).

7. XCODE PROJECT STRUCTURE

In terms of the project structure, the high-level thought process is to keep views, along with component and logic, as close to their relevant counterparts to ensure clarity. For example, the Fridge View would have a sub-folder containing components and logic only relevant to that view.

This logic governs the sections of the project. The project will then follow a hierarchy in which the root section (views, components, and logic) are at the top level, and sub sections (with their relevant views and components) will be contained in sub-directories within the root directory. Subsections will be named appropriately, given their intended role, and will have strict separation of duties.

Any significant functionality (in terms of buttons, data fetching and manipulation, etc) will be separated into a separate file dedicated to logic for that section of the code. This ensures that views are comprised of UI elements and layout, while other miscellaneous files house logic (unless the logic is short and easy to understand), so as to ensure clarity of a file's role in the codebase.

Dependency injection will be used as often as possible, such that structs and functions do not become too interdependent on each other, and instead have clearly separated duties. Additionally, this makes functionality simulation and mocking easier, as mock dependencies can be created where they are required for testing purposes.

REFERENCES

Clean Up Australia (n.d.) Food Waste, Clean Up Website, accessed 25 July 2023.
https://www.cleanup.org.au/foodwaste

Dean H(2023) What is the average grocery bill, Canstar Blue Website, accessed 25 July 2023.
https://www.canstarblue.com.au/groceries/average-grocery-bill/

Eriksen Translations Inc. (n.d.) Color & Culture, Eriksen Translations Inc. website, accessed 1 August 2023.
https://eriksen.com/marketing/color_culture/

Apple Inc. (n.d.) Color, Apple Developer website, accessed 1 August 2023.
https://developer.apple.com/design/human-interface-guidelines/color

Apple Inc. (n.d.) Typography, Apple Developer website, accessed 3 August 2023.
https://developer.apple.com/design/human-interface-guidelines/typography

Apple Inc. (n.d.) Layout, Apple Developer website, accessed 5 August 2023.
https://developer.apple.com/design/human-interface-guidelines/layout

Apple Inc. (n.d.) Lists and Tables, Apple Developer website, accessed 5 August 2023.
https://developer.apple.com/design/human-interface-guidelines/lists-and-tables

Apple Inc. (n.d.) Navigation Bars, Apple Developer website, accessed 7 August 2023.
https://developer.apple.com/design/human-interface-guidelines/navigation-bars