Hi-Fi Prototype Midway

Culture through Cuisine

Defne Genç, Janet Zhong,
Amrita Palaparthi, Kyla Guru

Meet Our Dishcover-ers

Defne

Dish of choice:

Kyla

Dish of choice:

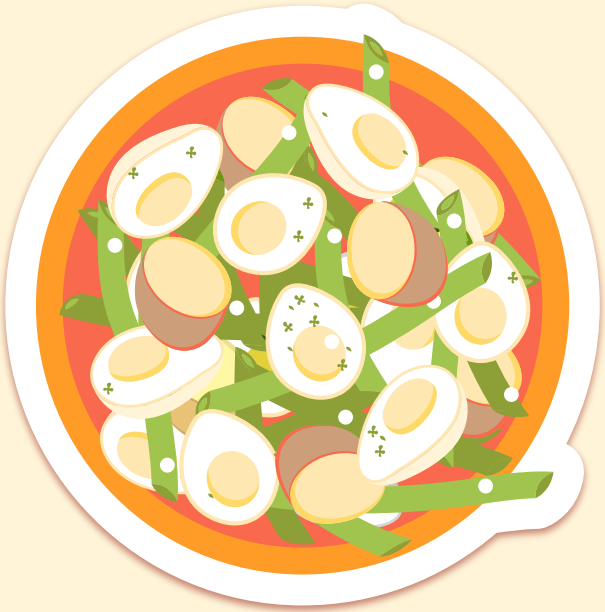
Janet

Dish of choice:

Amrita

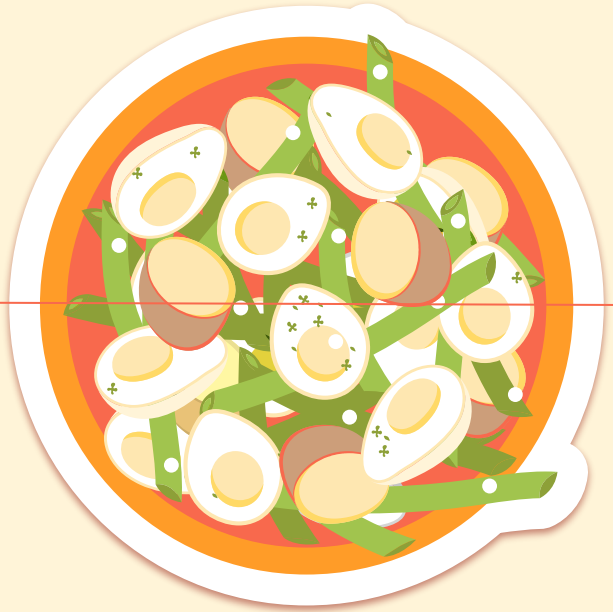
Dish of choice:

Problem Statement



Learning about and creating food with **ingredients** from other cultures can be an **intimidating experience**.

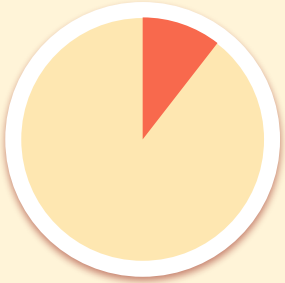
Our Solution



Dishcovery helps you recognize, learn about, and cook with foods from around the world. We use **image recognition** to identify new ingredients and enable you to delve into their **cultural context** and authentic recipes using them, providing a **cultural culinary companion** from the grocery store to your kitchen.

Table of Contents

Heuristic Evaluation Summary



6

Severity - 4

Our highest priority violations.

22

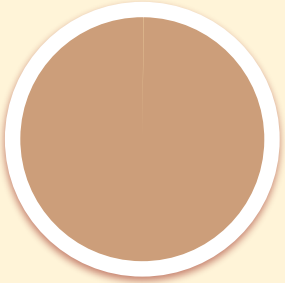
Severity - 3

Second priority violations.

31

Severity - 2

The most numerous amount of violations — .



82

Violations Total

With the remaining Severity 1.

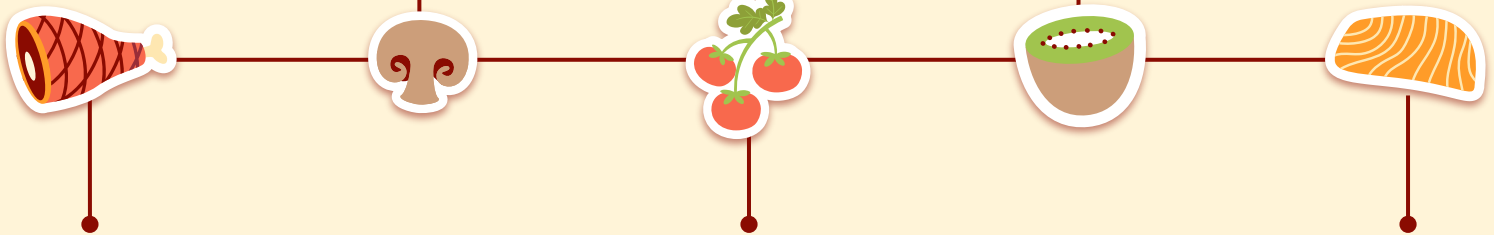
Heuristic Evaluation Summary

- **Consistency**
  - Consistent look and feel across pages
  - Layout, element placement within pages
- **Aesthetics and Minimalism**
  - Colors used in our app with meaning and intention
  - Only necessary animation between pages
  - Clear differentiation between different components in our UI
- **Signposting + Handling Failure Paths**
  - Notifications upon success and failure for tasks like scanning
  - Paths to re-scan an element when a scan fails
  - Clearer instructions for scanning and searching through dishes

Unaddressed Revisions

- Some qualitative recommendations and questions, i.e. **which cultures will be addressed**, where to get the data, and how to present foods that belong to multiple cultures
  - Will be left to decide for when we have real datasets, beyond implementation
- Some **button flows** that don't work aren't all resolved in the Figma (i.e. scan screens because of Figma limitations), more so left for the implementation prototype.
- **Mentioning cultural significance** in the context
  - Again, this depends on the datasets we plan on using, so it would be premature at this stage to try and reconcile what we will mention regarding the cultural significance of an ingredient
- We didn't **remove the match percentage (%90),** but rather added a "match" text to clarify.

Heuristic Revision Methodology



Separate

Violations by task

Rank

Violations by severity and colour-code them.

Prioritise

Higher severity to be resolved sooner.

Revise

Design to execute changes.

Mark

Changes made in green.

Encoding Violations to Prioritize

Red: Severity=4, Orange:Severity=3

Task 1: Search for an unfamiliar food item

Violations:

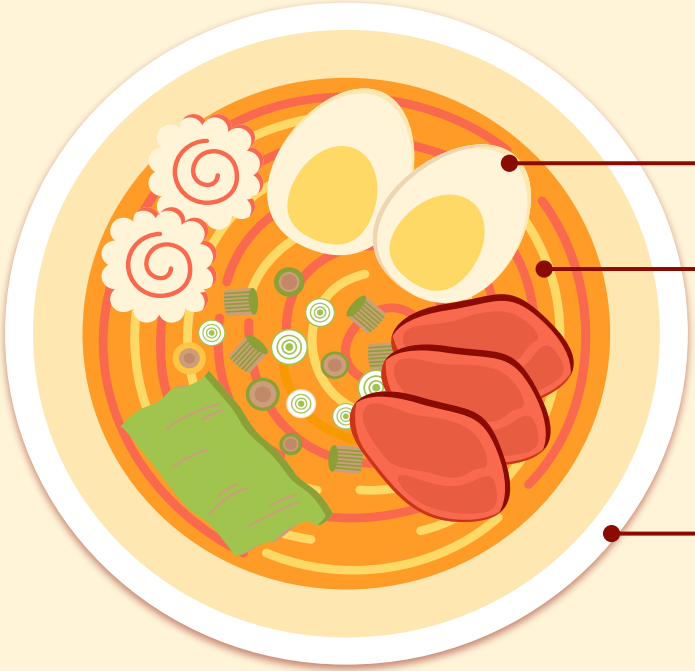
- There is no indication/confirmation on whether or not an item has been successfully scanned by the application.
- The interface's back button behaves differently based on the place in a task flow: sometimes it intuitively takes me back one screen, but sometimes it takes me back 2-3 screens (e.g. scanning bitter melon → more info → back)
- Description: navigation buttons do not work when I am about to scan an item / other parts of the app. [edit] - they work every other time for some reason.
- Description: After clicking on the Camera icon to start a scan (and before seeing the image recognition results), there isn't a way to go back to the previous screen.
- The image-recognition results of an ingredient doesn't have a mechanism for closing that tab.
- What happens if the user searches for an ingredient that doesn't exist?
- The Scan screen has instructional text saying "Place subject within focus!"
- You can check multiple user reviews for stars. Right now, you can click 5 star and

Task 1: Search for an unfamiliar food item

Violations:

- There is no indication/confirmation on whether or not an item has been successfully scanned by the application.
- The interface's back button behaves differently based on the place in a task flow: sometimes it intuitively takes me back one screen, but sometimes it takes me back 2-3 screens (e.g. scanning bitter melon → more info → back)
- Description: navigation buttons do not work when I am about to scan an item / other parts of the app. [edit] - they work every other time for some reason.
- Description: After clicking on the Camera icon to start a scan (and before seeing the image recognition results), there isn't a way to go back to the previous screen.
- The image-recognition results of an ingredient doesn't have a mechanism for closing that tab.
- What happens if the user searches for an ingredient that doesn't exist?
- The Scan screen has instructional text saying "Place subject within focus!"
- You can check multiple user reviews for stars. Right now, you can click 5 star and up, 3 star and up, and exclude 4 stars.
- In the Scan screen, there is nothing indicating that the subject is correctly in focus.

Recap of our Task Flows



1 **Search**

For an unfamiliar food item through scanning.

2 **Contextualise**

Using the information provided in the app.

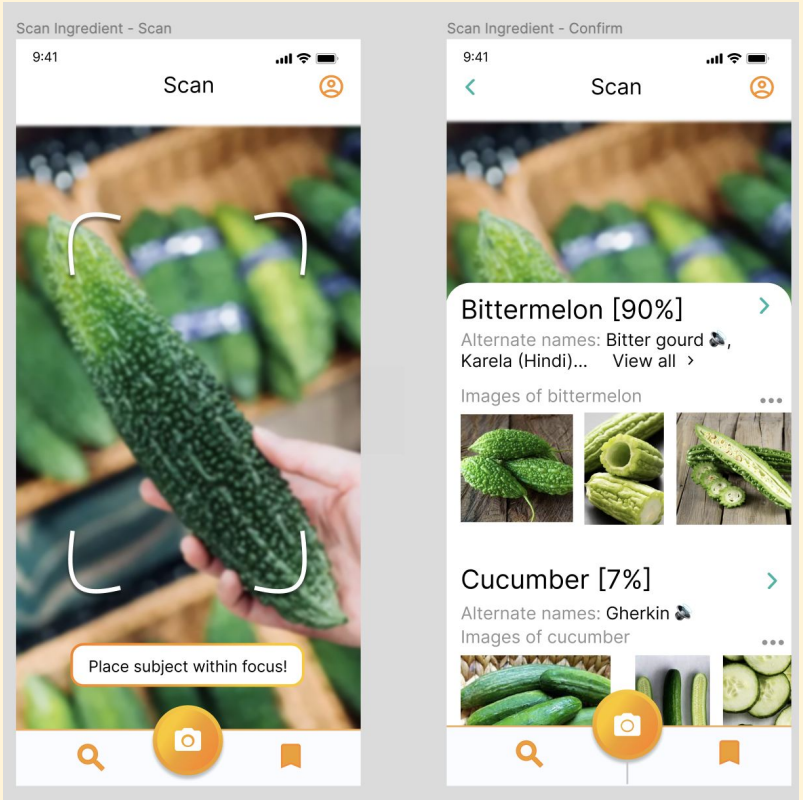
3 **Authentic Cooking**

Through recipes the user finds and/or saves.

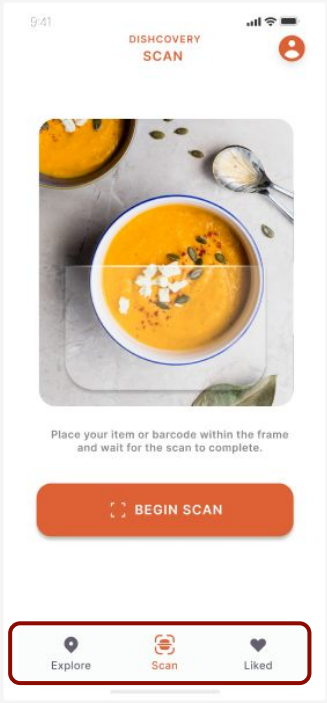
Task 1: Search for an unfamiliar food item

Severity	Description	Solution/Rationale
4	No confirmation when item scans or when the item isn't recognised.	• Use colours, confirmation, and warning as appropriate.
4	Some buttons not working properly in prototype.	• Addressed in implementation.
3	No warning for when subject isn't in focus.	• Added a warning to place subject in focus.
3	Colour scheme hard on the eyes.	• Changed to a single accent colour and removed gradients.

Task 1: Search for an unfamiliar food item (V1)



Task 1: Search for an unfamiliar food item (V2+Rationale)



S3: Removal of gradients to introduce a cleaner aesthetic.

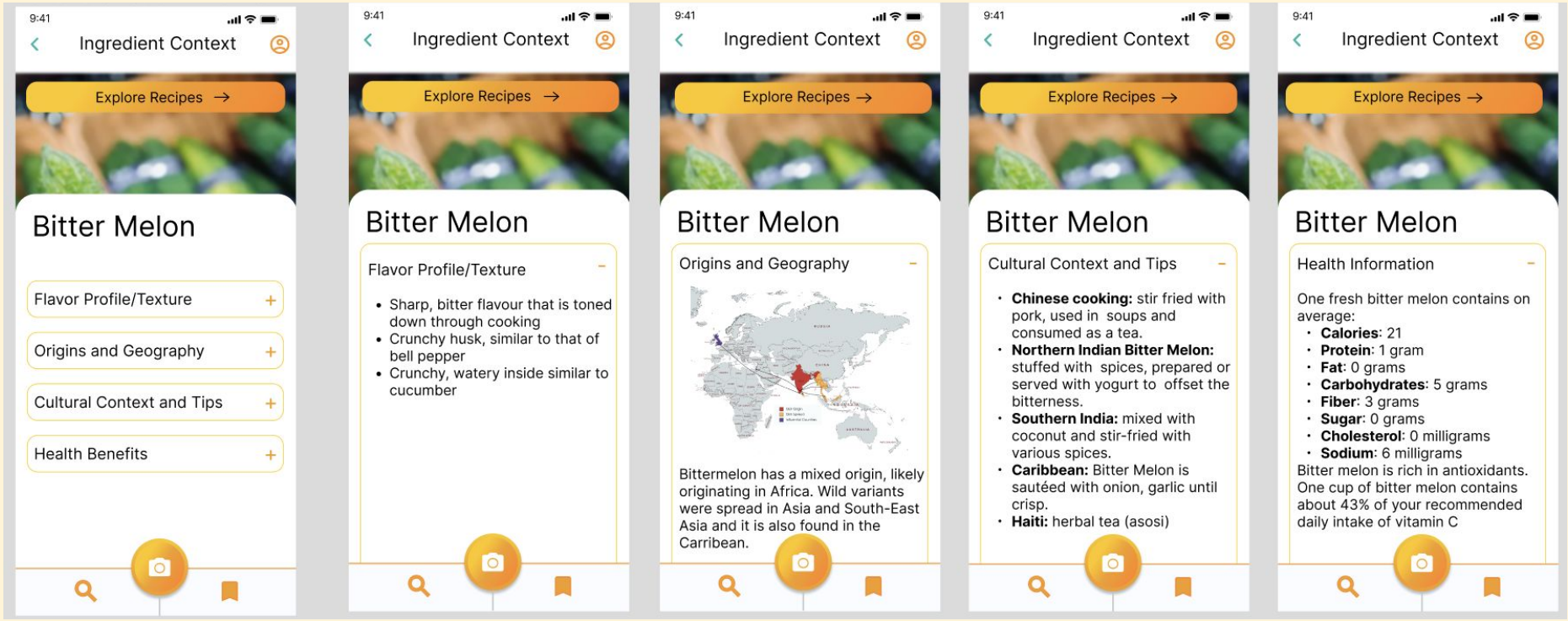
S3: Clear instructions for placing item in focus and search in progress.

S4: Clear errors to guide user towards accurate scan results.

Task 2: Contextualization

Severity	Description	Solution
4	"View all" button not visible because it's in the same colour as the text.	• Changed to accent colour.
4	Need to re-scan to go back to the context from recipes.	• Added a back button.
3	Inconsistent tab names, spacing and punctuation.	• Standardised using components.
3	Which recipes and cultures to include?	• "Request recipe" option added for inclusivity.

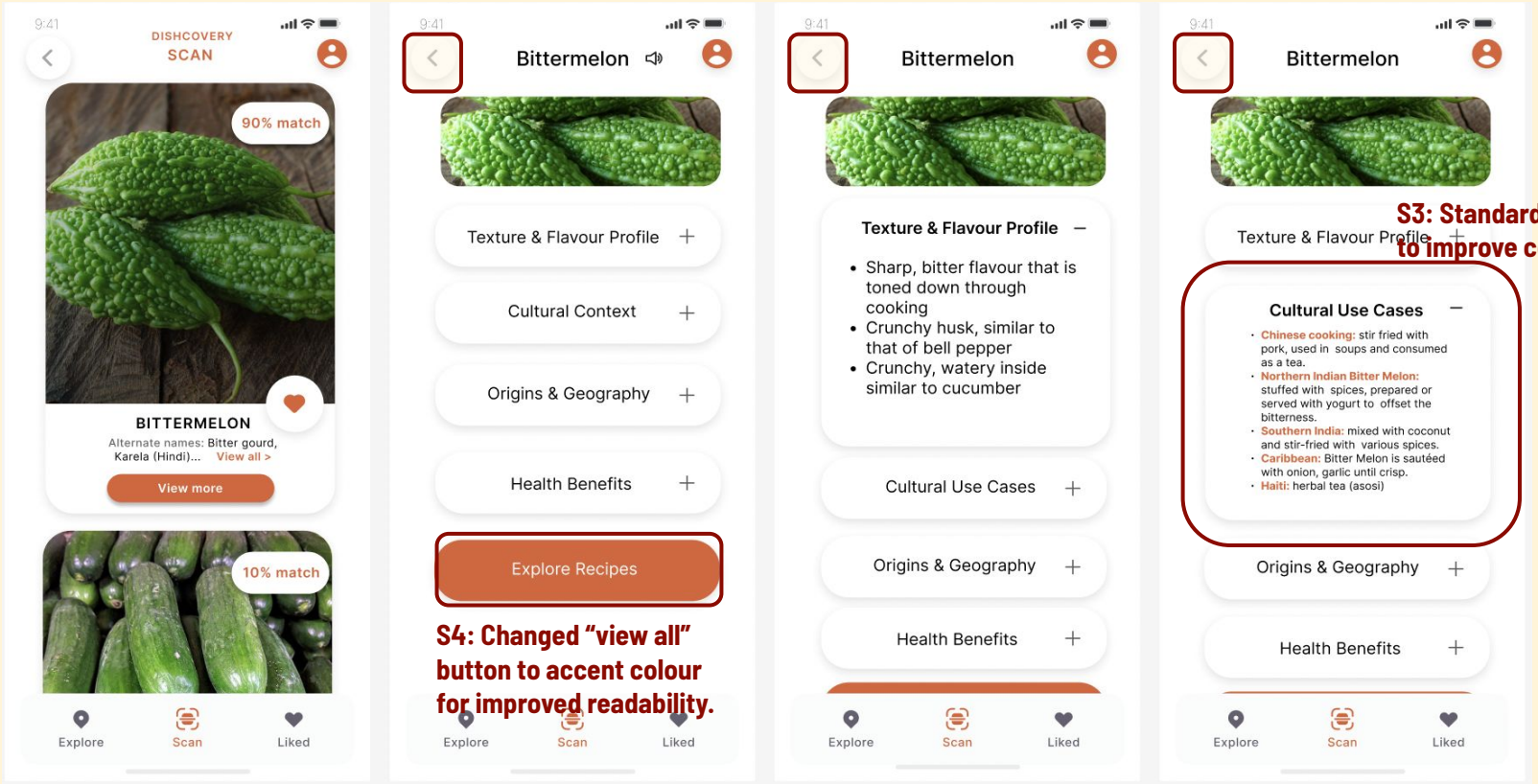
Task 2: Contextualization (V1)



Task 2: Contextualisation (V2+Rationale)

Usability goals: increasing learnability!

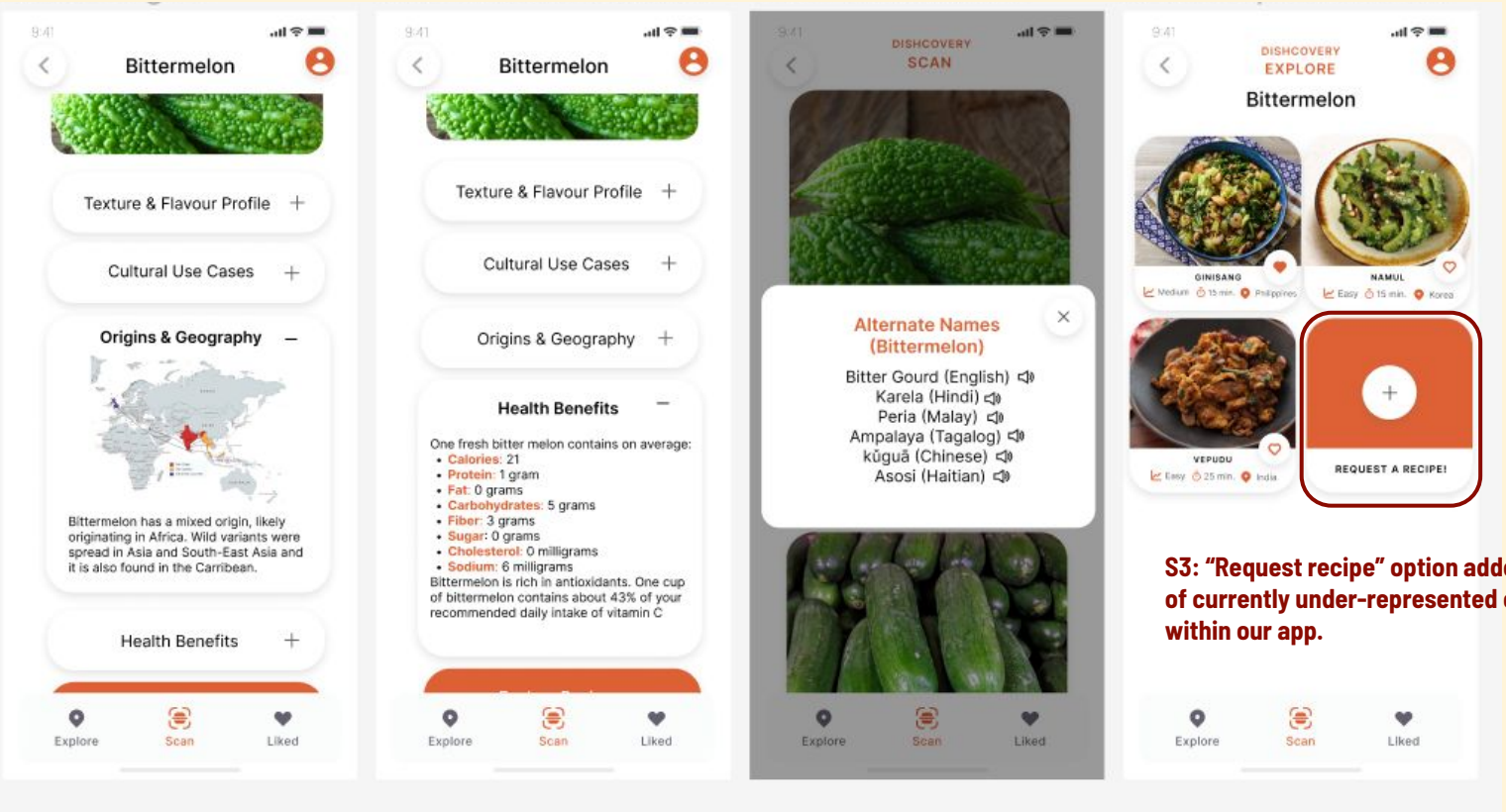
S4: Added a back button.



S4: Changed "view all" button to accent colour for improved readability.

S3: Standardized styles to improve consistency.

Task 2: Contextualisation (V2+Rationale)

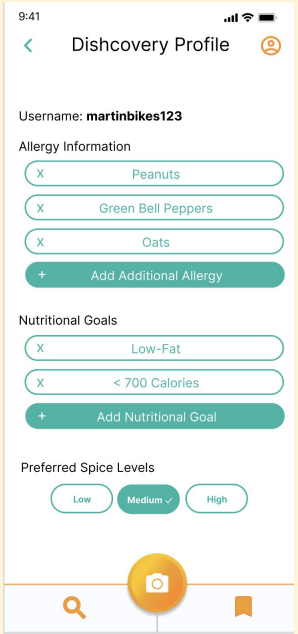


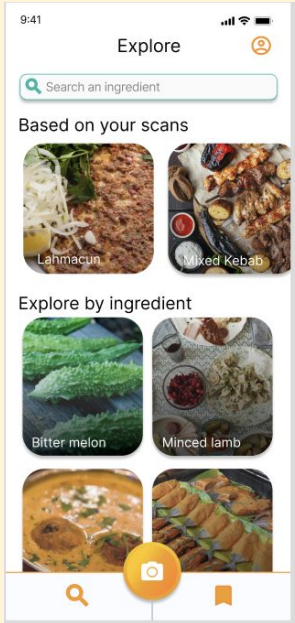
S3: "Request recipe" option added for inclusivity of currently under-represented cultural foods within our app.

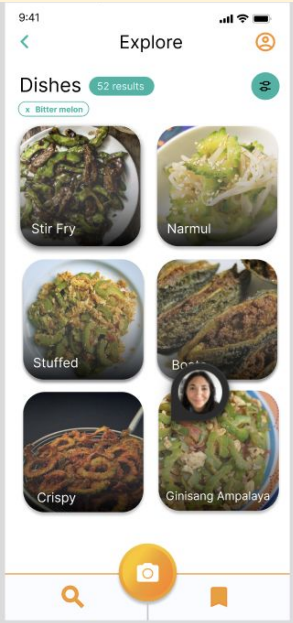
Task 3: Authentic Cooking

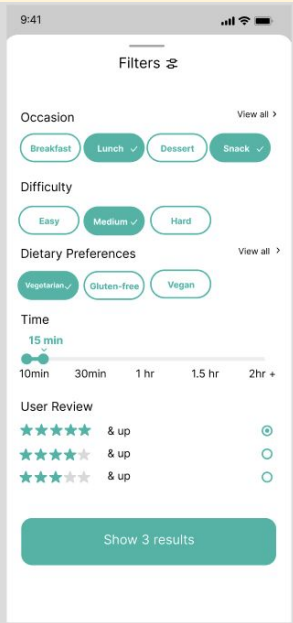
Severity	Description	Solution
4	Some buttons and texts take user to wrong destination.	• Addressed in prototype and implementation.
3	Inconsistent fonts and headers.	• Addressed in prototype and implementation.
3	No confirmation before un-saving.	• Confirmation added.
3	Filters buttons move when changed.	• Used auto layout for filters.

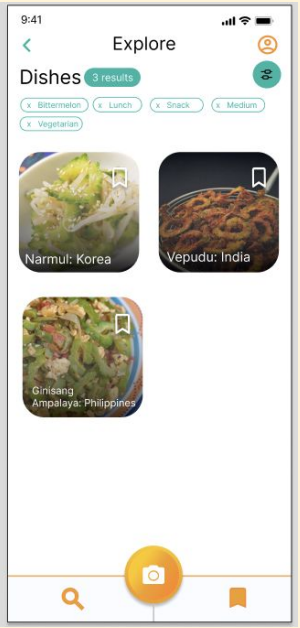
Task 3: Authentic Cooking (V1)





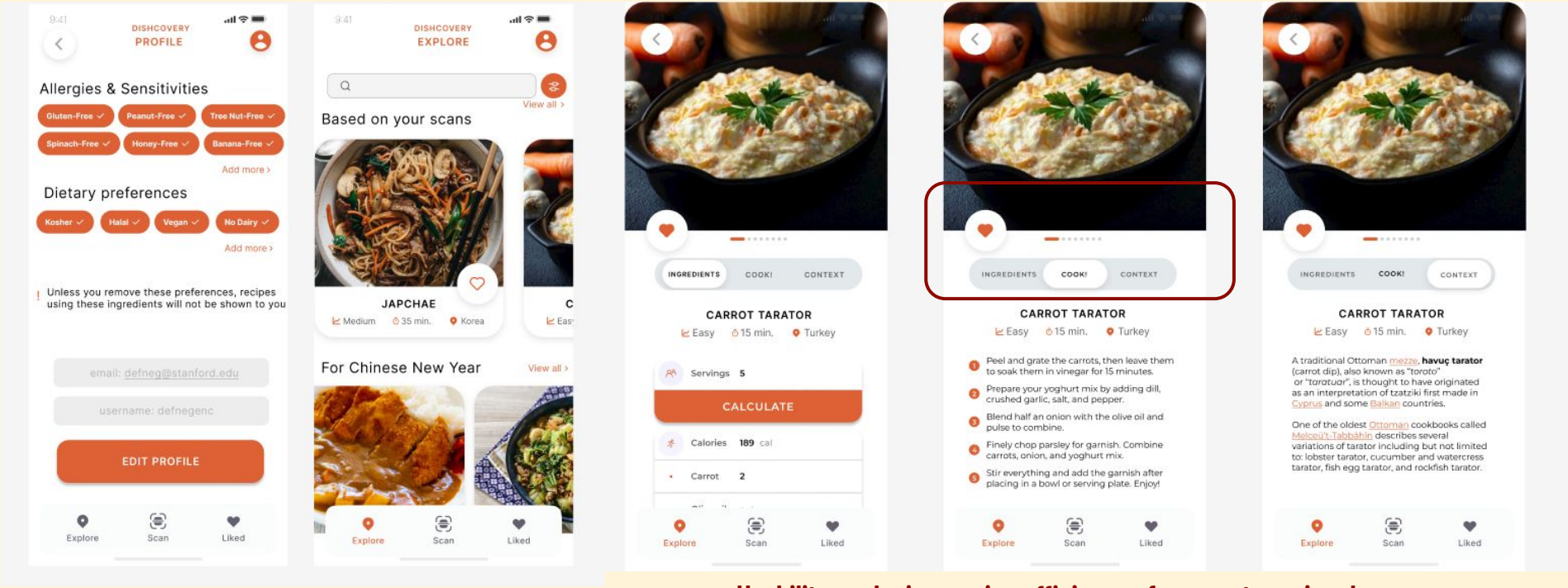






Task 3: Authentic Cooking (V2+Rationale)

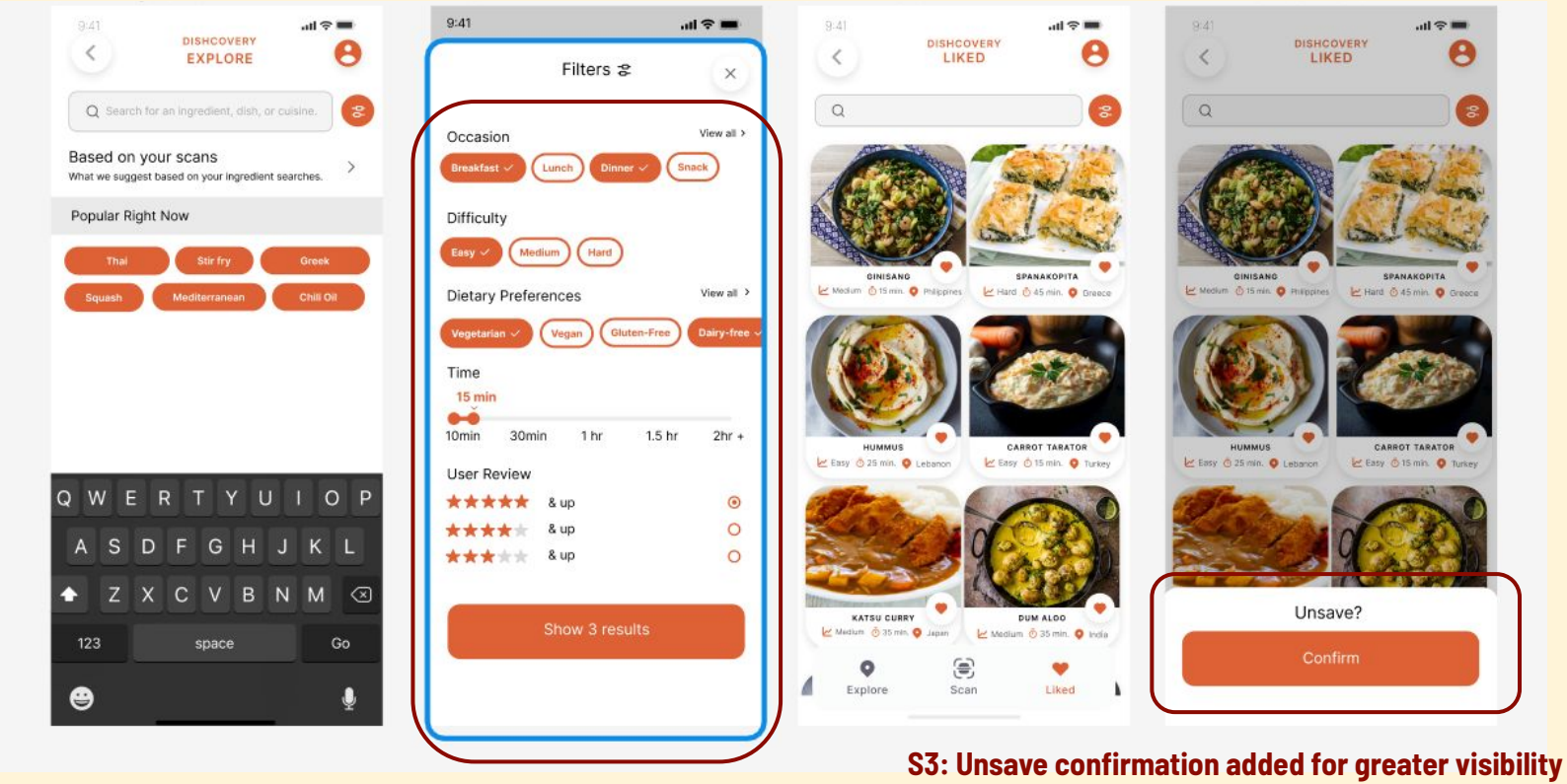
S3: more consistent fonts and headers.



Usability goals: increasing efficiency of access to recipes!

Task 3: Authentic Cooking

S3: Auto-layout for filters to improve readability



S3: Unsave confirmation added for greater visibility of status

Progress Towards Usability Goals

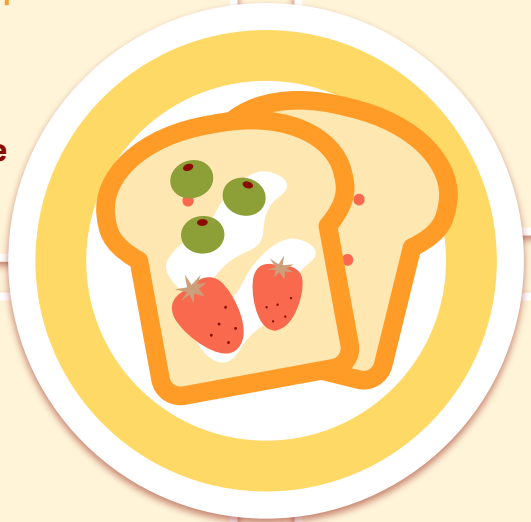
Efficiency

Increasing the relevance of recipes

To help the user quickly find a recipe that suits their preferences, we improved the options provided in preferences and the filtering flow. The user can also calculate ingredients required depending on serving size.

Guiding the user's eyes quicker through design

Ease of use in navigating recipe with consistent fonts/coloring, added bar navigation inside the recipe itself.



Learnability

Scan results organization

Instead of having several options for what the ingredient may be, the prototype presents the most likely ingredient prominently and *feeds* the user consumable and well-formatted cultural content.

Easy access to cultural context

The user can access cultural context of a specific ingredient after scanning, as well as the cultural context of the selected dish after recipe selection.

Design Revisions Recap

H8 and H11: Accessible and Minimalist Design

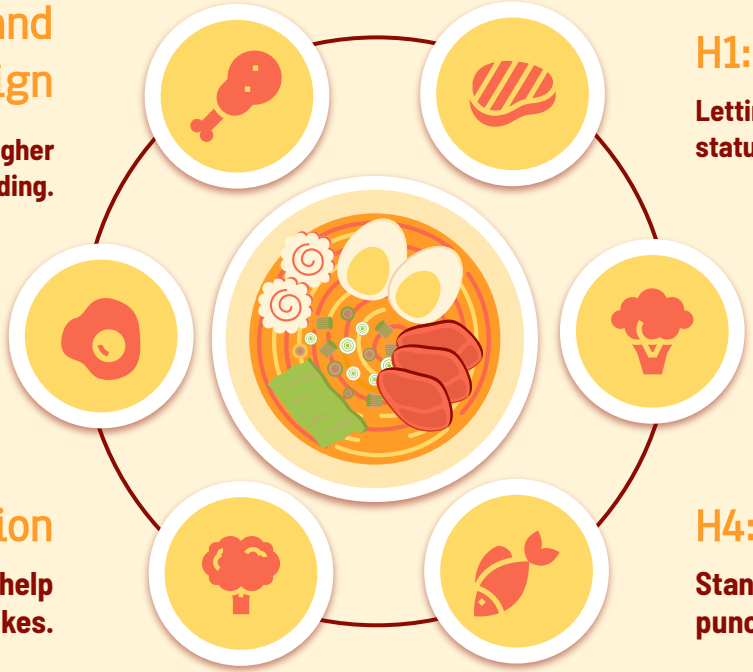
Fewer colours and higher contrast for ease of reading.

H12: Inclusion

More nutrition settings to be more inclusive of religious preferences.

H5: Error Prevention

Confirmations add to help users avoid mistakes.



H1: Visibility

Letting the user know the status and result of the scan.

H7: Efficiency

Making all buttons fully functional.

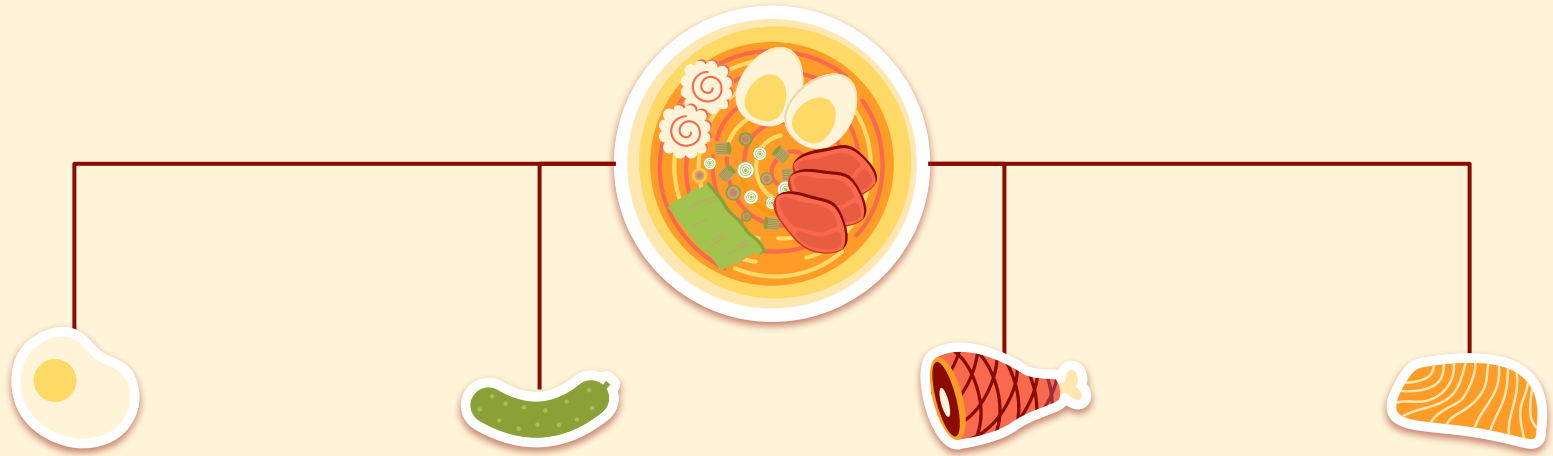
H4: Consistency

Standardising fonts, sizes, punctuation and names.

Prototyping

Converting design to code

Framework and Tools Used for Dev



GitHub

We have a central repo where we push updates to.

VSCode

VSCode allows us to produce code files and debug errors from Expo

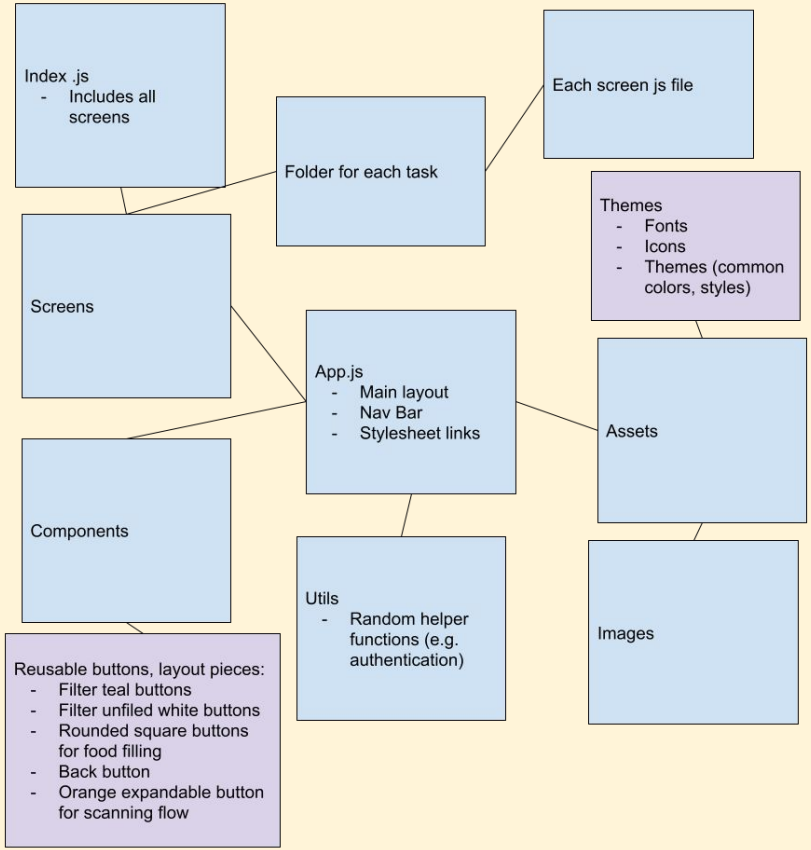
LiveShare

Especially when we are remote, LiveShare allows us to code collaboratively.

Expo

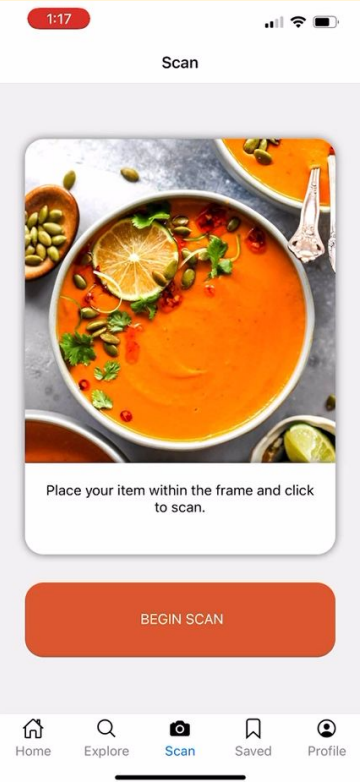
Expo lets us see our code changes (or code errors) come to life.

Code Organization



- Assets
  - Fonts, icons, images styles
- Constants (assets made into variables)
  - Fonts, icons, images, styles
- Screens (organized by task flow)
  - HomeScreen
  - Scan (**completed**)
    - ScanIntroScreen
    - ScanScreen
    - ScanCompleteScreen
  - Additional Context Screen
  - Explore
  - SavedScreen
  - ProfileScreen
- Components (reused in different screens)
  - DishCard

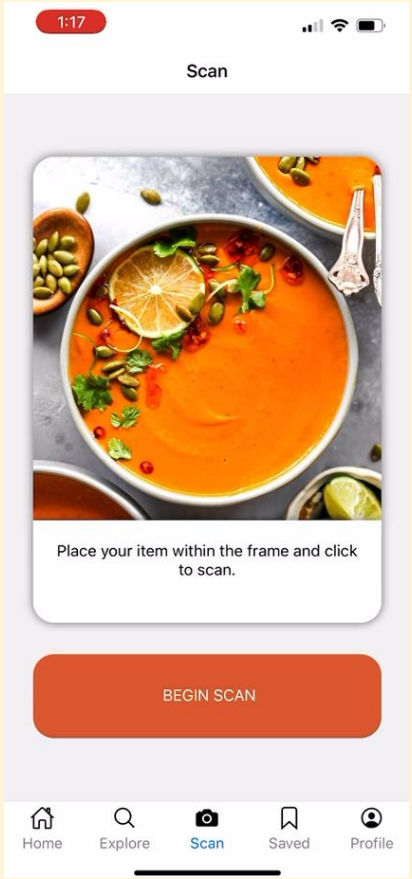
Implemented Features

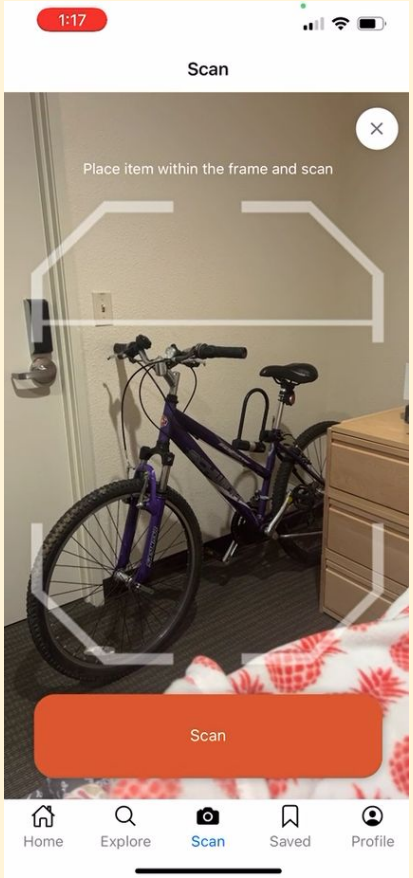


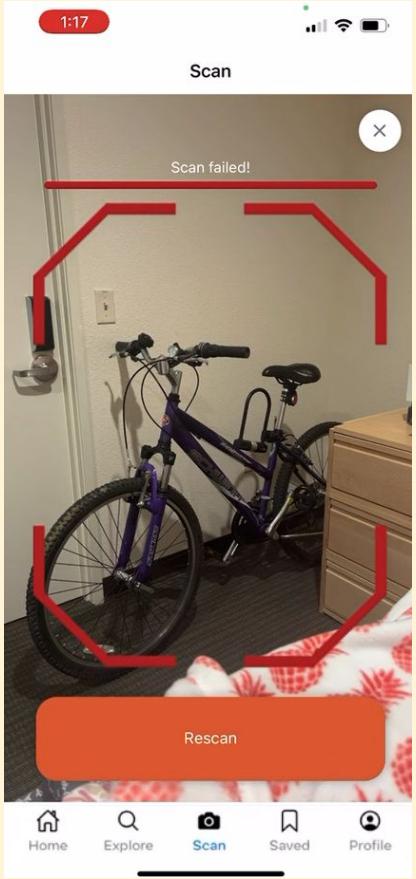
- ✓ Navigation for all main pages has been coded (Tab Navigator for Home, Explore, Scan, Saved, Profile and nested Stack Navigator for Scan Flow screen)
- ✓ **Functionality of Task 1:** Scan an Unfamiliar Ingredient has been completed
  - ✓ Connected to camera
  - ✓ Linked to Clarifai image recognition API
  - ✓ Created a library of cultural ingredients in the assets folder
  - ✓ Can recognize the items from the cultural ingredients (within some accuracy)
  - ✓ If ingredient is found, links to additional context page
  - ✓ If ingredient is not found, button to rescan
- ✓ **UI for Task 1:** Scan an Unfamiliar Ingredient has also been completed
  - ✓ Begin Scan UI
  - ✓ Animated white scanning aimer gif on top of camera
  - ✓ Progress bar as image recognition is searching
  - ✓ Scan box and progress bar turn red or green if image recognition is successful or not successful

Implemented Features

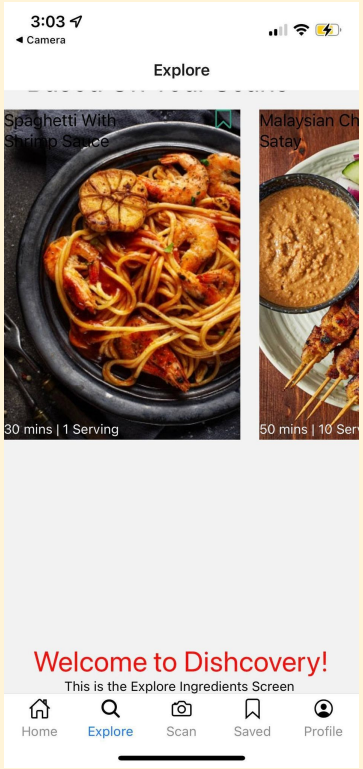
***UI is slightly different to Figma because in the code you have to click to scan, and it doesn't scan if you just hover like a QR code**







Unimplemented Features and Plans to Finish



- **Task flow 2 and 3**
  - We have started the explore page!
- Search by recipe function
- Filter recipe function
- Profile page
- Pre-filtering of searching recipes using profile data
- Connect app to Firebase/ Supabase to save user's data if they want to save recipes
- Saved recipe page
- (We no longer have a home page - it is now the explore page)
- We also have to write the hard-coded cultural context information
- Unify the UI (common fonts, button sizes, boxes etc)

We will have a weekend "hackathon" to finish!

Wizard of Oz or Hard-Coded Elements

Surprisingly, our image recognition is actually fully functional (woohoo!) and not wizard of oz'd. However, the perceived accuracy is a wizard of oz technique as it only recognizes three ingredients from a list that we create.

Wizard of Oz - Accuracy of Image Recognition

Image recognition only uses guesses that are in both the Clarifai API food-item library (around 500 generic ingredients) as well as our custom library of cultural food items. Currently we only have three items, bitter melon, cardamom and lemongrass. So if one scanned a cucumber, it would probably come up as bitter melon.

Hard-coded

The cultural context of our selected items is currently hard-coded and stored locally on the app. In practice, this could potentially be sourced by searching the internet and interviewing individuals who prepare traditional cultural recipes

Issues/Questions

We haven't had significant roadblocks in our code so far but we do raise the following issues if this were to become a real app:

- **The Clarifai AI library only recognizes fairly generic food items.** If our aim is to recognize cultural / uncommon ingredients, then using Clarifai library is actually biased against this. One would need to create their own annotated data set of cultural ingredients and train their own computer vision model to truly achieve Discovery's aim. This is actually doable though - it could potentially be sourced by scraping/ downloading foreign food items from ethnic online grocery websites.
- In a real app, sourcing the cultural context and **making it very authentic is also a difficult** and hard to scale problem if this cultural context is manually sourced.

Progress and Trajectory

We are here!

Design

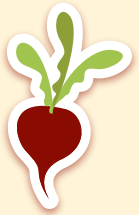
Revise and prepare to code.

Dev pt. 1

Code our first task flow, which is scanning.

Dev pt. 2

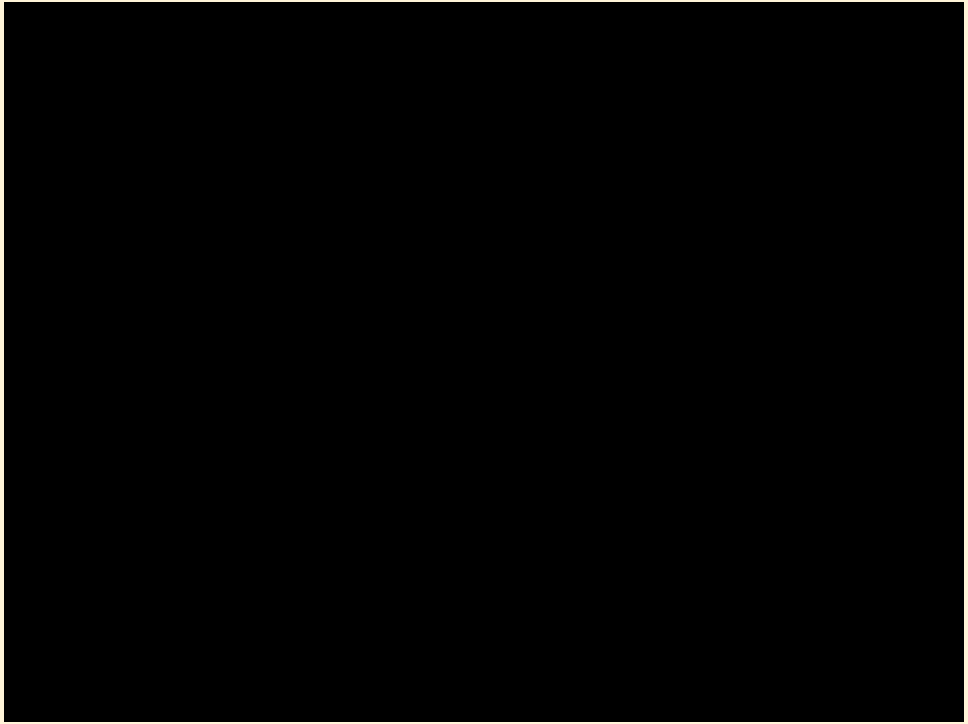
Code up and refine the rest of our implementation in accordance with the design.



Complete!

Project expo and video

Demo Video!



Appen-dish