FoodScanner

-Jiaxuan Cao,
ChiYai Cheung

AGENDA

01 Purpose

02 Overview & Demo

03 Road Block & Solution

04 Differently developing next time?

05 Version 2.0

06 Closing

FoodScannner

Overview: Simplify food identification and provide nutritional information.
Purpose: The purpose of the food scanner is to help users identify food items and access detailed nutritional information efficiently.

User Authentication

- Log In / Sign Up

Scan food

- Hold, Take, Get!
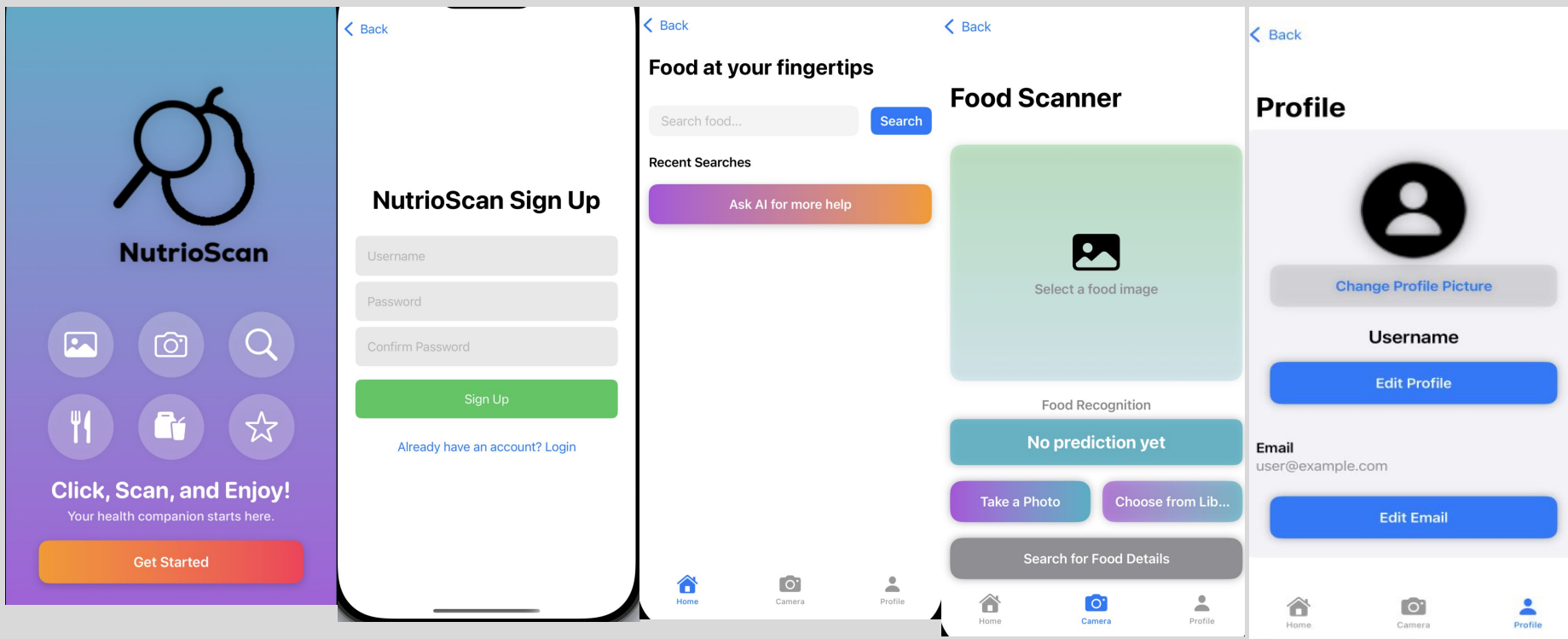- Makes it easier to get direct nutritional information about the food you have

ChatBot

- Don't know what to cook? Ask ChatBot

Direct Search

- A quick and easy way to get all information of each food item

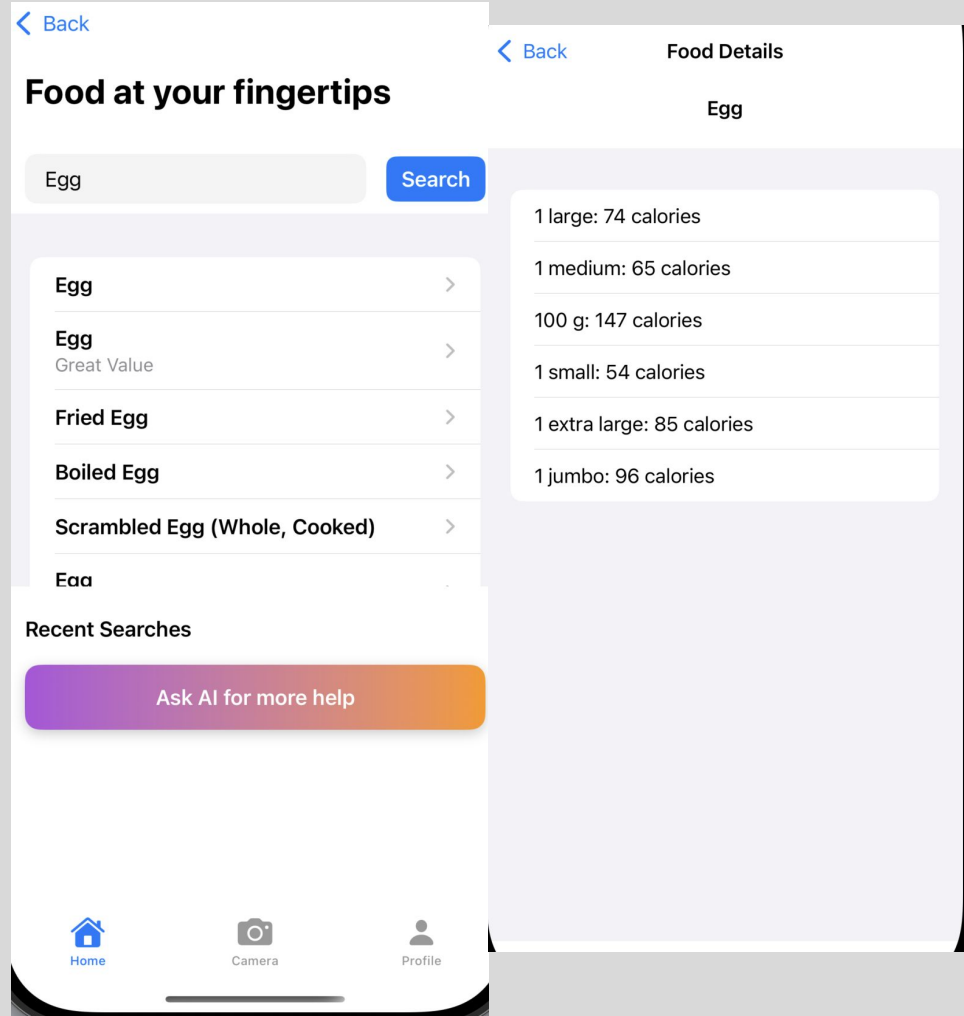# Layout

User Authentication

Chosen Approach:
- Local user authorization via JSON
- The app stores user credentials in a local JSON file on the device. The app loads the JSON data, checks for matching credentials, and provides access if authenticated when the user attempts to log in.

Local authentication vs. Cloud Based authentication (FireBase)
- Cloud based authentication such as FireBase is too computationally taxing.
- Incorporating FireBase includes packages such as FireStore which increases the number of dependencies and computational overhead.
- JSON reduces computational overhead and is a lot "cheaper" to run

Direct Search

- [Fatsecret](#) API
- Quickly retrieves food information by name.
- Provides detailed nutritional insight.

Back

Food at your fingertips

Egg   Search

Egg   >

Egg
Great Value   >

Fried Egg   >

Boiled Egg   >

Scrambled Egg (Whole, Cooked)   >

Egg

Recent Searches

Ask AI for more help

Home   Camera   Profile

Back   **Food Details**

Egg

1 large: 74 calories

1 medium: 65 calories

100 g: 147 calories

1 small: 54 calories

1 extra large: 85 calories

1 jumbo: 96 calories

Scanner

- API not working, switch to CoreML
- CoreML: Use to integrate machine learning model to project
- See Food: It's a trained CoreMl model to recognize food items.
- How it work: Input Image and converted into a format suitable for ML. Get the Prediction Result.
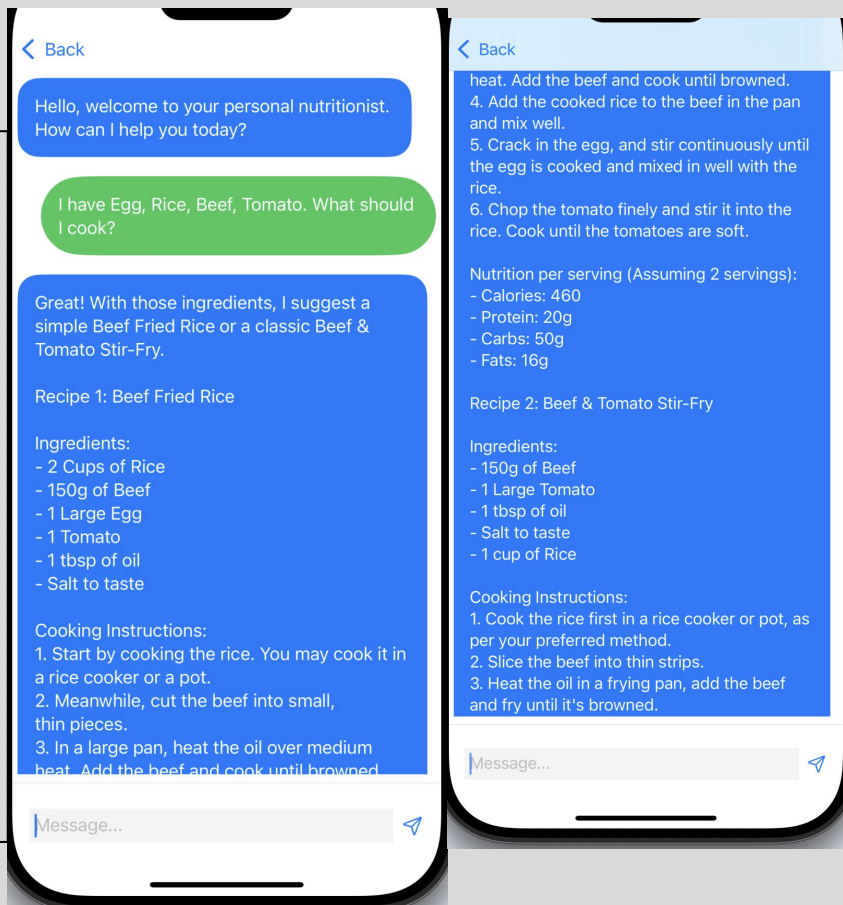
ChatBot

- Using API from ChatGPT
- Provide users with personalized advice and recipes suggestions.
- Set prompt to let ChatBot focus on nutrition or food. Not unrelated things.
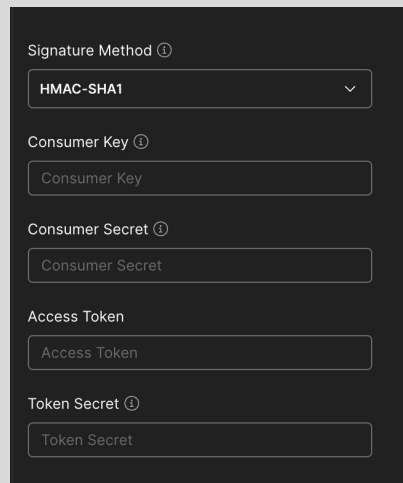
```
let prompt = """
You are a specialized nutritionist assistant. When users provide ingredients:
- Suggest 1-2 possible recipes using those ingredients.
- For each recipe provide:
  * List of all ingredients with measurements.
  * Step-by-step cooking instructions.
  * Nutrition facts per serving (calories, protein, carbs, fats).
If the user asks about anything unrelated to nutrition or food, politely refuse
  and guide them back to providing food items.
"""
```

- Set rule for input: is it from user ?

Back

Hello, welcome to your personal nutritionist. How can I help you today?

I have Egg, Rice, Beef, Tomato. What should I cook?

Great! With those ingredients, I suggest a simple Beef Fried Rice or a classic Beef & Tomato Stir-Fry.

Recipe 1: Beef Fried Rice

Ingredients:
- 2 Cups of Rice
- 150g of Beef
- 1 Large Egg
- 1 Tomato
- 1 tbsp of oil
- Salt to taste

Cooking Instructions:
1. Start by cooking the rice. You may cook it in a rice cooker or a pot.
2. Meanwhile, cut the beef into small, thin pieces.
3. In a large pan, heat the oil over medium heat. Add the beef and cook until browned

Message...

Back

heat. Add the beef and cook until browned.
4. Add the cooked rice to the beef in the pan and mix well.
5. Crack in the egg, and stir continuously until the egg is cooked and mixed in well with the rice.
6. Chop the tomato finely and stir it into the rice. Cook until the tomatoes are soft.

Nutrition per serving (Assuming 2 servings):
- Calories: 460
- Protein: 20g
- Carbs: 50g
- Fats: 16g

Recipe 2: Beef & Tomato Stir-Fry

Ingredients:
- 150g of Beef
- 1 Large Tomato
- 1 tbsp of oil
- Salt to taste
- 1 cup of Rice

Cooking Instructions:
1. Cook the rice first in a rice cooker or pot, as per your preferred method.
2. Slice the beef into thin strips.
3. Heat the oil in a frying pan, add the beef and fry until it's browned.

Message...

RoadBlocks

- Image recognition. We tried serval API with built in image recognition. But they are not working very well. Convert input image in to API readle thing is very difficult.
    - Solution: Use Core MI as we talked before.
- OAuth 1.0, which is a encrypt way to secure API requests. (Ex: see right) Otherwise it will not give result back.
    - Solution: Built a helper for signing requests.
- Package with OpenAI. In order to call API from Chatgpt with swift. I used a Github call OpenAI. It makes easier to send the request and get outcome back.

Signature Method ⓘ

HMAC-SHA1 ˅

Consumer Key ⓘ

Consumer Key

Consumer Secret ⓘ

Consumer Secret

Access Token

Access Token

Token Secret ⓘ

Token Secret

Next Time?

- Build Step By Step.
    - In the playground or another new Project.
    - Mark any updates, what's new? What's been changed? For better communication
    - Just like github.

- Build Upon What we Have
    - Our app has a lot of components that we can make better and expand upon its uses

📁 foodcoreml

📁 foodscanner2.1

📁 foodscanner2.2

📁 foodscanner2.3

📁 FoodScanner2.3.1

📁 FoodScanner2.3.1 2

📄 FoodScanner2.3.1.zip

📁 foodscannerbarcode

Version 2.0

- Secure User Information, accounts and passwords.
- Memory user history.
- Faster ChatBot response.
- Allowed user build record of their body situation and combine with chatbots to provide healthier recipes.
- Multiple items detection in camera part.
    - Combine direct search with camera scanner without click search button. (Pull out all information within one page)
- Improve on the recent searches to include links of the search results

Thank You!