

Fooducate App

Development Team

Yousef Mardrosyan (Product Owner)

Allie Offer

Nigel Gary

Harsirat Grewal (Scrum Master)

Danny Al Najjar

Project Theme

Health and Safety

Summary: Our project primarily focuses on promoting health and well-being by providing users with tools to track their nutrition and maintain a healthy lifestyle.

User Story

As a health-conscious user, I want to track my daily nutrient intake with meal logging & meal planning, find and save healthy recipes, create a grocery list, and set a budget for the ingredients of the recipe. I also want to have a nutritional chatbot that can answer any nutrition related questions with a feature to get nutritional information from a picture of the food.

Computer Science / Software Concepts

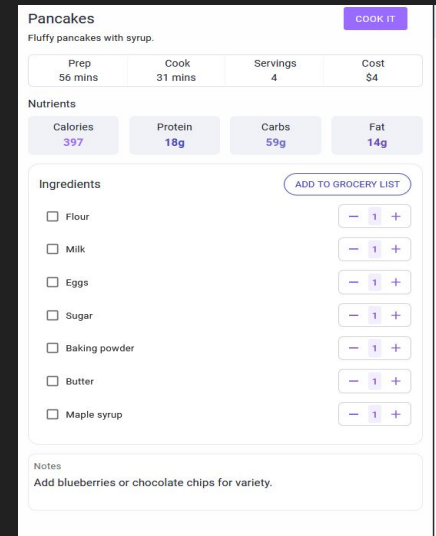
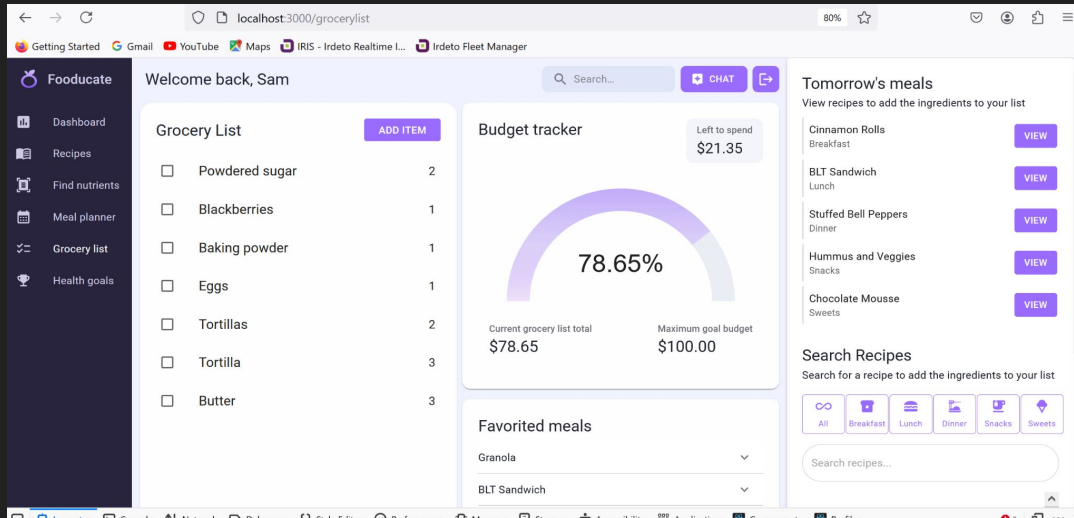
- Database management - Implemented Firebase, designed the database using concepts learned from database design & implementation (csi 3450).
- Web Development - Used React and Material UI for user-friendly UI design, building from fundamental concepts in sophomore project.
- Version Control - Collaborating in a team environment, practicing scrum development policies.
- API Integration - Created personalized OpenAI Chatbot, using concepts learned from Software Engineering.

Life Balance Objectives/Goals Page

- Create Todos/Goals: Users can set goals across different life categories (nutritional, health, fitness, personal development, etc.).
- Track Progress: Visual indicators to show progress in each category.
- Journal Entries: Option to create and maintain journal entries for personal reflection.

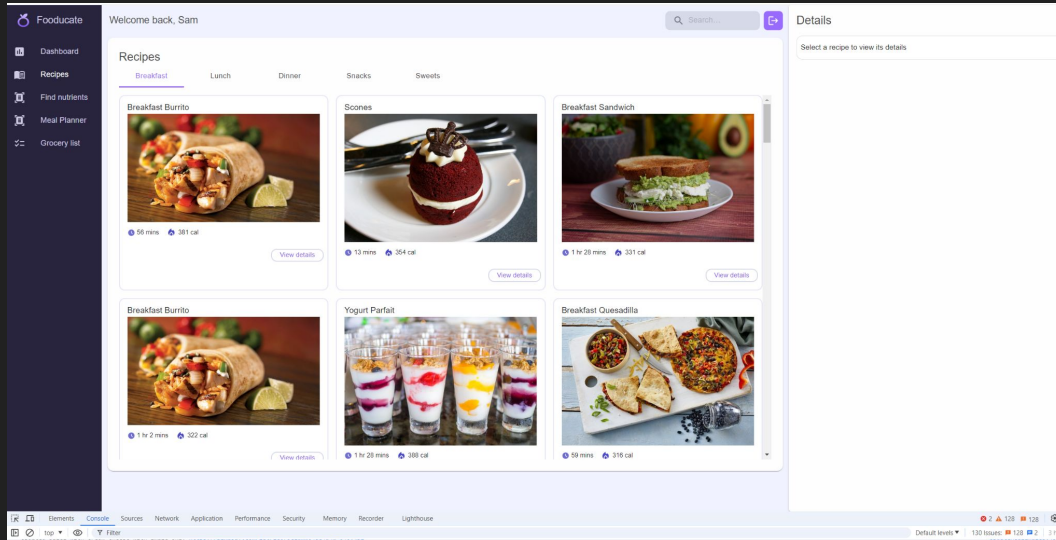
Ingredient to Grocery List

- Meal Integration: Users can add ingredients from any meal in the database to their grocery list.
- Convenience: Simplifies grocery shopping by linking meals to grocery needs.



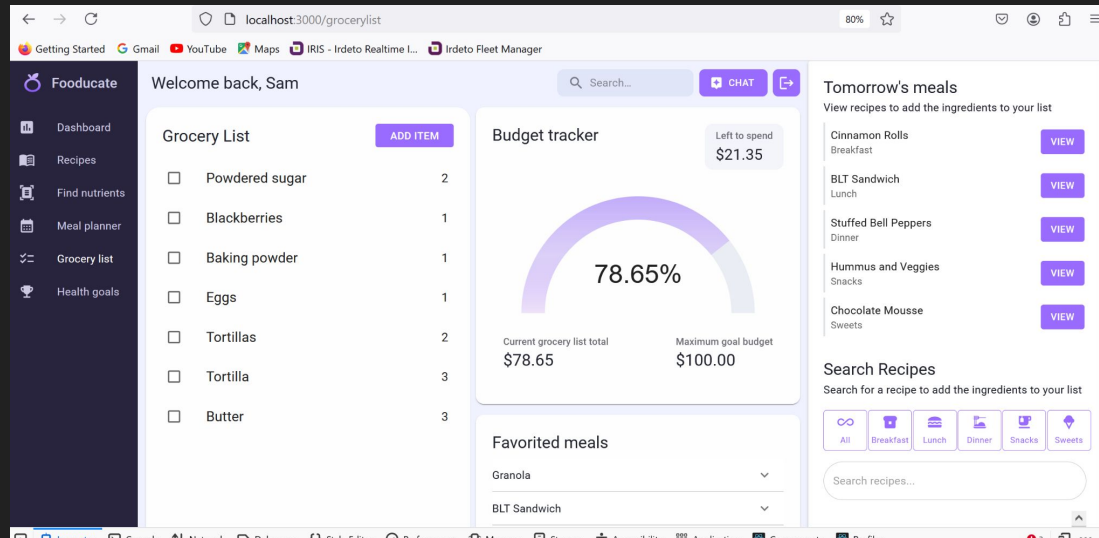
Favorite Meals Feature

- Meal Favorites: Users can mark and store their favorite meals for easy access.
- Enhanced Experience: Quick access to preferred meals improves user experience.



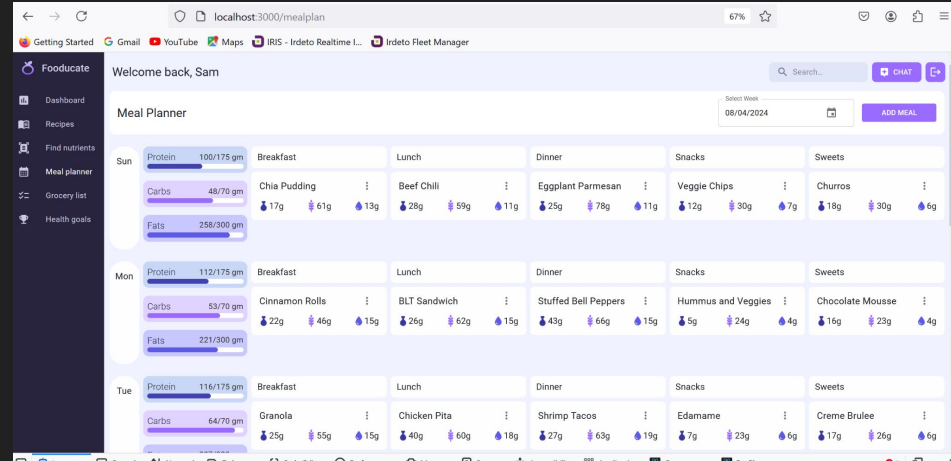
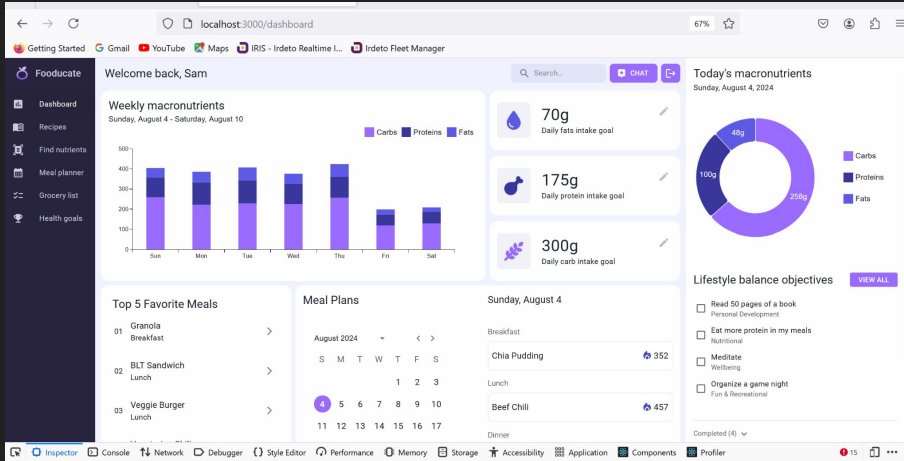
Budget Calculator

- Set Budget: Users enter their maximum budget for groceries.
- Track Spending: Real-time updates on how much has been spent and remaining budget.
- Total Calculation: Displays current total amount spent and remaining budget.



Macronutrient Intake Display

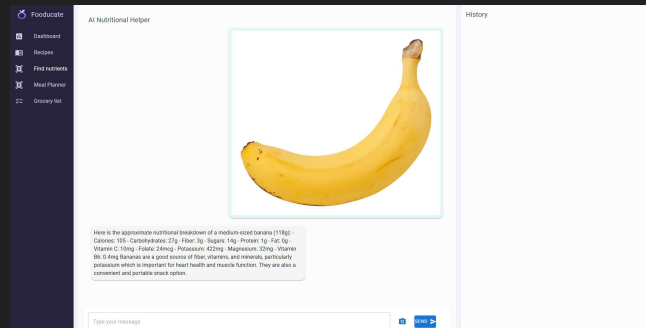
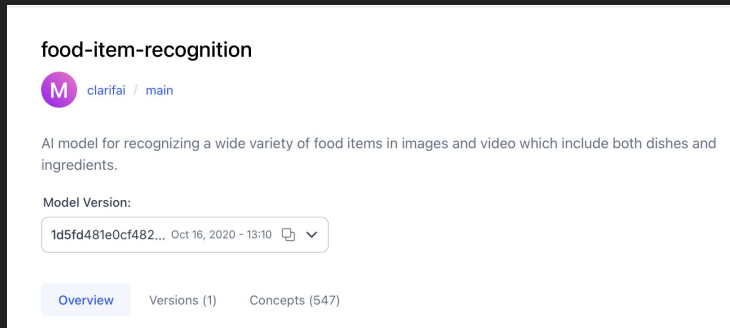
- Daily/Weekly Intake: Calculates and displays users macronutrient intake.
- Nutritional Tracking: Helps users monitor their nutritional balance.



Nutritional Chatbot AI Coach

AI Coach: Provides personalized nutritional advice and support.

Functionality: Offers recommendations and answers nutrition-related questions.

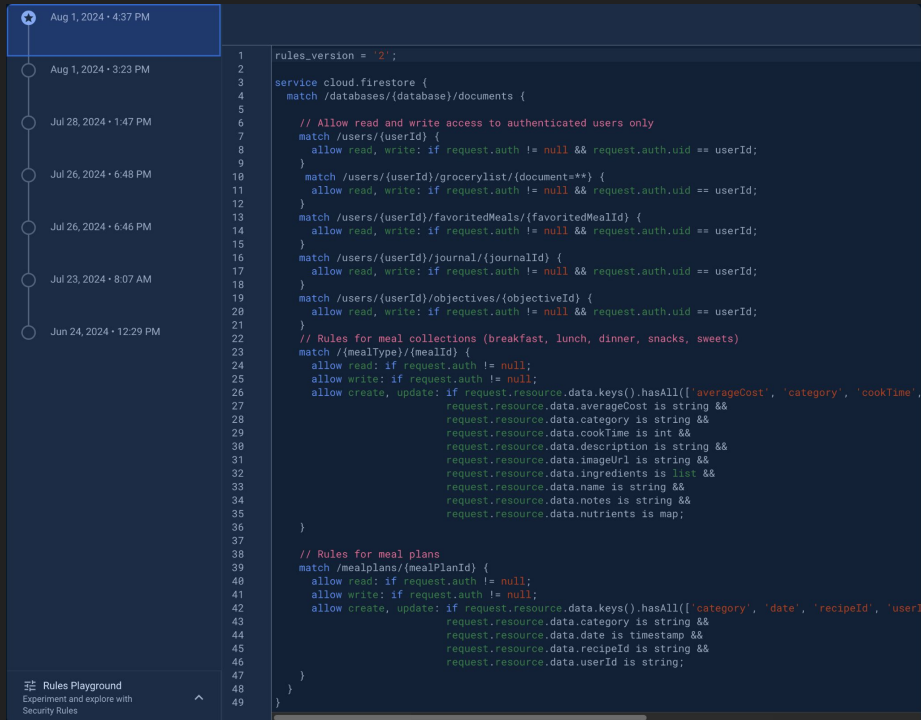


Demo!



User Access Management

- Integrated security rules
- Rules for protecting user Information and database access.



The screenshot displays the Google Cloud Rules Playground interface. On the left, a vertical timeline shows a series of rule updates with timestamps: Aug 1, 2024 - 4:37 PM (highlighted), Aug 1, 2024 - 3:23 PM, Jul 28, 2024 - 1:47 PM, Jul 26, 2024 - 6:48 PM, Jul 26, 2024 - 6:46 PM, Jul 23, 2024 - 8:07 AM, and Jun 24, 2024 - 12:29 PM. At the bottom left, the text 'Rules Playground' is visible, along with the subtitle 'Experiment and explore with Security Rules'. On the right, the rule code is displayed in a dark-themed editor. The code defines a Firestore security rule for the 'cloud.firestore' service, specifically for the path '/databases/{database}/documents'. It includes several match clauses for different document paths and their associated permissions (allow read, allow write, allow create, allow update). The rules are organized into sections: general authenticated user access, meal collections (breakfast, lunch, dinner, snacks, sweets), and meal plans. The code uses conditional logic to restrict access based on the request's authentication and the resource's data fields.

```
1 rules_version = '2';
2
3 service cloud.firestore {
4   match /databases/{database}/documents {
5
6     // Allow read and write access to authenticated users only
7     match /users/{userId} {
8       allow read, write: if request.auth != null && request.auth.uid == userId;
9     }
10    match /users/{userId}/grocerylist/{documentId} {
11      allow read, write: if request.auth != null && request.auth.uid == userId;
12    }
13    match /users/{userId}/favoritedMeals/{favoritedMealId} {
14      allow read, write: if request.auth != null && request.auth.uid == userId;
15    }
16    match /users/{userId}/journal/{journalId} {
17      allow read, write: if request.auth != null && request.auth.uid == userId;
18    }
19    match /users/{userId}/objectives/{objectiveId} {
20      allow read, write: if request.auth != null && request.auth.uid == userId;
21    }
22    // Rules for meal collections (breakfast, lunch, dinner, snacks, sweets)
23    match /mealType/{mealId} {
24      allow read: if request.auth != null;
25      allow write: if request.auth != null;
26      allow create, update: if request.resource.data.keys().hasAll(['averageCost', 'category', 'cookTime',
27        request.resource.data.averageCost is string &&
28        request.resource.data.category is string &&
29        request.resource.data.cookTime is int &&
30        request.resource.data.description is string &&
31        request.resource.data.imageUrl is string &&
32        request.resource.data.ingredients is list &&
33        request.resource.data.name is string &&
34        request.resource.data.notes is string &&
35        request.resource.data.nutrients is map;
36    }
37
38    // Rules for meal plans
39    match /mealplans/{mealPlanId} {
40      allow read: if request.auth != null;
41      allow write: if request.auth != null;
42      allow create, update: if request.resource.data.keys().hasAll(['category', 'date', 'recipeId', 'userId',
43        request.resource.data.category is string &&
44        request.resource.data.date is timestamp &&
45        request.resource.data.recipeId is string &&
46        request.resource.data.userId is string;
47    }
48  }
49 }
```

Work in Progress

- Personal Journal - Will help the user securely write down personal nutritional goals.
- Chatbot History - It will provide the user to look back at nutritional chatbot history instead of querying again.
- Health Goals - Users will be able to set weekly/monthly health goals and reminders.

Q&A

Questions and Answers:

- Open floor for questions from the audience.