FreshAI: An AI-Powered Sustainability App

Overview

FreshAI is an AI-powered app designed to help individuals (and potentially restaurants) reduce food waste and make the most of their groceries. Developed by a team of three using Jupyter Notebook and OpenAI, the app empowers users to track what they buy, get notified when items are close to spoiling, and discover creative, personalized recipes to use those ingredients before they go to waste.

Problem

Food waste is a serious global challenge - households and restaurants often throw away food due to poor tracking, forgotten items, and lack of meal ideas. This not only wastes money but also has a significant environmental impact.

Solution

FreshAI gives users visibility and actionable suggestions to minimize waste:

- Track groceries easily by scanning receipts or manually entering items.

- Optionally input expiration dates for perishable items.

- Get notifications when items are close to spoiling.

- Receive creative, personalized recipes based on soon-to-expire ingredients, tailored to user preferences such as allergies, dietary restrictions, and preferred cuisines.

- Suggestions for proper storage techniques to extend freshness.

By focusing only on ingredients that are about to expire, FreshAI keeps the experience simple and goal-oriented.

Features

- Grocery Input: Scan grocery receipts or manually add items.

- Expiration Tracking: User-provided expiration dates or estimated based on item type.

- Spoilage Notifications: Alerts about what needs to be used soon.

- Recipe Suggestions: AI-generated recipes using ingredients that are about to spoil, filtered by dietary needs and cuisine preferences.

- Storage Tips: Best practices for storing specific items to extend their shelf life.

- Mobile-First Vision: Though currently built in Jupyter Notebook for development, the long-term goal is a seamless mobile app experience.

Target Users

- Individuals and families who want to save money and reduce waste.

- Individuals with dietary restrictions or specific cuisine preferences.

- Restaurants and food service establishments seeking to minimize waste and costs.

Technology Stack

- Jupyter Notebook - current development and testing platform.

- OpenAI API - for recipe generation, natural language understanding, and tailored suggestions.

- Python - for data handling, notifications logic, and integration with receipt parsing libraries.

Current Stage

We are starting from scratch, currently building a proof-of-concept in Jupyter Notebook to implement the core functionality:

- Receipt scanning and grocery logging.

- Spoilage detection and notifications.

- Recipe generation tailored to user preferences.

Future Vision

We envision FreshAI as a widely-used, mobile-first application that empowers users to shop smarter, eat better, and waste less. Potential next steps include:

- Developing a clean mobile app interface.

- Adding push notification functionality.

- Offering sustainability metrics over time, such as "You've saved $XX and reduced your waste by XX% this month."

- Exploring partnerships with grocery stores, meal kit services, or recipe websites to expand value to users.

Roles & Responsibilities for Building FreshAI

1. Backend Developer / Data & AI Lead

Responsible for the technical backbone of the app - data collection, AI integration, and notifications logic.

Step-by-Step Tasks:

1. Set up the Jupyter Notebook environment and project repository.

2. Write Python code to:

- Parse grocery receipt data (e.g., using OCR or text extraction libraries).

- Store and manage grocery item records, expiration dates, and timestamps.

3. Implement expiration tracking:

- Estimate shelf life for common items if expiration date isn't provided.

- Schedule notifications for items approaching expiration.

4. Integrate OpenAI API:

- Generate recipe suggestions based on soon-to-expire ingredients.

- Tailor recipes to user preferences (allergies, dietary needs, cuisine).

5. Develop storage tips module to recommend best storage practices per item.

6. Prepare backend logic to support eventual mobile app integration.

2. Frontend Developer / UI/UX Designer

Responsible for designing and implementing the user interface (currently conceptualized, later for mobile).

Step-by-Step Tasks:

1. Draft user flows:

- From grocery input to notifications and recipe viewing.

- Include optional settings for preferences and storage tips.

2. Design mockups / wireframes of the future mobile app:

- Clean dashboard showing current pantry and upcoming expirations.

- Simple input screen for adding groceries (manual + receipt scanning).

- Notifications screen with alerts and suggested recipes.

3. Plan user-friendly forms for entering allergies, cuisine preferences, and expiration dates.

4. Work with Backend Developer to test how data will be displayed in the interface.

5. Prepare for later implementation as a mobile app (choose framework, e.g., React Native, Flutter, or Swift/Kotlin).

3. Project Manager / Tester / Documentation Lead

Responsible for planning, coordinating, and ensuring the quality and usability of the app.

Step-by-Step Tasks:

1. Define project timeline and milestones:

- Prototype completion, testing, feedback rounds, and mobile app development phases.

2. Manage task assignments, track progress, and facilitate team communication.

3. Research user needs and best practices around food tracking and sustainability to inform feature development.

4. Test the app at each stage:

- Input flow, notification timing, recipe quality, and user experience.

- Document and report bugs to the development team.

5. Write clear documentation:

- Setup instructions for the codebase.

- README file for the repository.

- User guide describing how to use the prototype.

6. Coordinate feedback from potential users and suggest improvements.

Summary Table

Role	Focus Areas
Backend Developer / AI Lead	Data storage, expiration logic, OpenAI integration, notifications
Frontend Developer / UI/UX	User flows, wireframes, input forms, visual design, future app framework
Project Manager / Tester / Docs	Timeline, testing, user research, documentation, feedback loop