Fridge Manager: Requirements, Vision, and Scope

Product Vision Statement

Motivation / Opportunity

Our goal is to implement an android app to solve the problem of not knowing what food is currently at home and when it expires. In addition, our app should solve the problem of a missing overview of the spendings and food consumption. The advantage of our product over pre-existing apps is that users are able to scan the grocery bill with their smartphone and the app will provide them with a list of all foods the users purchased and automatically add them to their local food database.

Problem Statement

The problem of	Reduce the food waste caused by expired food
affects	Consumers looking to reduce waste and maintain an overview of spendings on groceries without having to spend time on tracking grocery purchases.
The impact of which is	Solves the problem of food waste by reminding users of the expiry date for each product. An example case would be meat that spoils quickly and is expensive which might have been forgotten by the user.
A successful solution would be	A program which gives you an overview of your groceries currently at home, in the fridge or pantry and then provides information on nutritional value and sets reminders for the expiry dates of the said items.

Product Position Statements

For	All consumers who make regular grocery purchases.
Who	Leave food in the fridge/pantry without being aware of whether it is still in good state.
Our System	Android app
That	Automatically scans receipts, adds food to virtual fridge manager, and reminds the user of food about to expire.
Unlike	Other fridge manager apps like fresh box, fridge pal, best before or grocery hero.
Our Product	Allows users to scan receipts and automatically adds the food products and provides updates to expiry date

	information on their current state.

Users: User of our app is everybody who wants to have a clear overview of current contents in the food stock (fridge / freezer / pantry / etc.), and food expenditures.
Feature List:

- Ability to take a picture of grocery receipts and recognize and analyze text
- Search through database for expiry date of food item and input groceries onto list and display expiry date through UI
- Visual interface of food stock; allows user to add and remove groceries from food stock to keep track of quantities remaining
- Ability to remove items and manually add items
- Categorize items by physical location they might be placed (items that might go in the fridge, freezer, or pantry) or food type (fish, poultry, beef, vegetables, etc.)
- Ability to track weekly/monthly spendings on food
- Ability to notify soon-to-expire foods in stock

Constraints: App has to be running on Android devices, version 4.0.3, with built-in camera support
Scope and Limitations: App doesn't provide any recipe suggestions for the food in the food stock or suggestions which food should be bought. It also does not provide the ability to scan barcodes and provide information about the item.
Assumptions and Dependencies: We expect that the text recognition service, Google Mobile Vision, allow us to scan grocery bills and output text, including blocks, lines and words.

Use Cases

1. **Analyze and Read Text from Receipt**
    a. Primary Actor : End user
    b. Stakeholders : End user
    c. Precondition : App must be running on an Android device with Camera support and the receipt must be in English, with minimum physical damage.
    d. Postcondition : All food items will be extracted from the receipt.
    e. Main Success Scenario :
        1. User takes a picture of the receipt
        2. App receives the image and processes it using OCR Reader code from Google Mobile Vision
        3. Google Mobile Vision processes the image to retrieve a list of lines on the image
        4. App receive the lines and retrieves one food item per line if exists
        5. App displays the list of food items it was able to identify from the receipt
        6. User confirms that recognized items are linked with the right food items.
    f. Extensions and Alternative Flows :
        1. Image was unreadable or some food was not recognized
            1. App prompts the user to make a picture again or manually enter items
        2. Text was unrecognized

1. App prompts the user which items it couldn't recongnize.
2. User has to enter the corresponding food manually.
3. App saves the llink between unknown item and entered name

g. Open Issues : Compatibility with cameras on different phones

2. Search Expiry Date Database for Expiry Date

a. Primary Actor : App
b. Stakeholders : End user
c. Precondition : Food item must be valid and contained in the library database.
d. Postcondition : Returns an expiry date for the corresponding food.
e. Main Success Scenario :
    1. App access the expiry data database
    2. App searches through the database for expiry date corresponding to specified food item
    3. App retrieves the expiry date from database and mapps it to specifed food
    4. User confirms that the found expiry dates are correct
f. Extensions and Alternative Flows :
    1. Finds the wrong expiry date for the food item
        1. User enters exipry date manually

3. Manually Add Food to Food Item Database

a. Primary Actor : End user
b. Stakeholders : End user
c. Precondition : The food item must be composed of valid ASCII characters
d. Postcondition : Food item is added into food item database, and can be viewed by the end user through the graphing interface
e. Main Success Scenario:
    1. User selects a food from manually enters a food name. The app can suggest auto-complete word from the library.
    2. User enters quantity, selects food location and units for the food.
    3. App verifies the name so that it contains valid characters
    4. App adds the food item to the food item database
    5. App retrieves expiry date of the food item if exists, and links it with the food
    6. App refreshes its user interface so the new changes to the database is visible to user
f. Extensions and Alternative Flows
    1. Food item already exists in the food item database bought on the same day with same expiry
        1. App increases the quantity of the food item
        2. App retrieves expiry date if exists, and links it with the food
    2. Quantity entered was 0 or smaller.
        1. App prompts a message suggesting to enter a valid quantity

4. Manually Enter Expiry Date of Food Item

a. Primary Actor : End user
b. Stakeholders : End user

c. Precondition : Food item already exists in the fridge
d. Postcondition : Expiry date for the food item is updated or created
e. Main Success Scenario :
    1. User selects a food from food stock user interface
    2. User manually enter the expiry date for that food item through a calendar interface
    3. App verifies that the date is a valid day
    4. App links the entered expiry date with the food item
    5. App adds the expiry date specified from the user to the corresponding food, and updates new change on the user interface
f. Extensions and Alternative Flows :
    1. The expiry date is in the past
        1. App alerts the user of past expiry date
        2. App prompts the user to allow automatic expiry date reset (expiry countdown from present date) or have the user input new expiry date

5. Remove Food from Food Item Database

a. Primary Actor : End user
b. Stakeholders : End user
c. Precondition : Food item already exists in the food stock database, and the quantity to remove is not bigger than the quantity in stock.
d. Postcondition : Quantity of the food item decreases by amount entered by the user. IF the quantity reaches 0, corresponding information about the food is removed.
e. Main Success Scenario :
    1. User enters the foodstock user interface
    2. User selects some quantity to remove from the food stock
    3. App decrements by that quantity
    4. App removes corresponding food and food information from the database if the quantity reaches 0 upon decrementing
f. Extensions and Alternative Flows :
    1. None

6. Alert User of Food About to Expire

a. Primary Actor : App
b. Stakeholders : End user
c. Precondition : A food item exist in the database and is about to reach its expiry date in 3 days.
d. Postcondition : User receives alert containing the name(s) of the food that is about the expiry
e. Main Success Scenario :
    1. App finds in the database a food item that expires in 3 days.
    2. App displays a push message to the user at specified time of the day (by default 18:00)
    3. The user confirms the message with the OK button
f. Extensions and Alternative Flows

1. The smartphone was off, when the message should have been displayed and the food already expired.
    1. The user gets informed when the food item expired once the phone is turned back on

Non-functional Requirements

Performance requirements:

- Finishes processing each photo and adding food items under 4 seconds
- Any additional buttons (eg. Undo) should have an instantaneous-like (0.1s) response time

Software Quality Attributes:

- Ease of use over ease of learning
- Reliability, availability, scalability, and recoverability (RASR) due to database usage

Security Requirement:

- Personal information from receipts (store location/credit card information) processing should be kept private to users only

User Demographics

End Users: Only Android users will have access to this mobile app. They are expected to have experience with an android UI. They aren't expected to have any programming skills.