Functional Specifications

Project title: Grocery Inventory Manager
Author: James Kirk
Platform: Java, SwingUI

1. Introduction

The Grocery Inventory Manager is a desktop application designed to help users manage and organize food items within their household. The application allows users to assign items to storage locations such as a pantry, refrigerator, or freezer, and organize items by user-defined categories. Users may optionally assign expiration dates to items and filter items based on freshness. Additional features include searching for items by name, sorting items alphabetically or by category, and creating a printable grocery list. The goal of this system is to provide an easy-to-use tool for tracking food inventory and reducing waste.

2. Functional Requirements

2.1 Item Management

• The system shall allow the user to add new food items.

• Each food item shall have:

  • A name

  • A quantity

  • An optional expiration date

  • An optional category

  • An assigned storage location

• The system shall allow the user to edit existing food items.

• The system shall allow the user to delete food items.

2.2 Location Management

• The system shall allow users to create storage locations (pantry, fridge…)

- The system shall allow users to delete existing storage locations.

- The system shall display all food items assigned to a selected location.

2.3 Category Management

- The system shall allow users to create food categories.

- The system shall allow users to delete food categories.

- The system shall allow users to assign categories to food items.

- The system shall allow users to filter food items by category.

2.4 Expiration Tracking & Freshness Filtering

- The system shall allow users to optionally assign expiration dates to food items.

- The system shall determine the freshness status of an item based on the current date.

- The system shall allow users to filter items by:

  - Fresh

  - Expiring Soon

  - Expired

2.5 Sorting and Searching

- The system shall allow users to sort food items alphabetically by item name.

- The system shall allow users to sort food items alphabetically by category name.

- The system shall allow users to search for food items by name.

2.6 Grocery List Feature (Bonus Feature)

- The system shall allow users to create a grocery list.

- The system shall allow users to save the grocery list.

- The system shall allow users to print the grocery list.

3. Use Cases

3.1 Add Food Item

- Actor: User

- Description: The user adds a new food item to the inventory.

- Steps:

    1. User enters item name, quantity, category (optional), expiration date (optional), and location.

    2. User clicks the "Add Item" button.

    3. The system validates the input.

    4. The item is added to the selected location.

3.2 Remove Food Item

- Actor: User

- Description: The user removes a selected food item from the inventory.

- Steps:

    1. User selects an existing food item.

    2. User clicks the "Delete" button.

    3. The system removes the item from the inventory.

3.3 Filter Items by Freshness

- Actor: User

- Description: The user filters items based on expiration status.

- Steps:

    1. User selects a freshness filter option.

    2. The system displays only items matching the selected freshness level.

3.4 Search for Food Item

- Actor: User

- Description: The user searches for an item by name.

- Steps:

    1. User enters the item name in the search field.

    2. The system displays matching items.

3.5 Create Grocery List

- Actor: User

- Description: The user creates and saves a grocery list.

- Steps:

    1. User adds items to the grocery list.

    2. User saves the grocery list.

    3. The user prints the grocery list (optional).

4. Data Requirements

4.1 Food Item Data

Each food item shall store the following:

- Item ID

- Item Name

- Quantity (1 by default)

- Category (optional)

- Storage Location

- Expiration Date (optional)

- Freshness Status (Calculated)

4.2 Category Data

Each category shall store:

- Category ID

- Category Name

4.3 Location Data

Each location shall store:

- Location ID

- Location Name

4.4 Grocery List Data

Each grocery list shall store:

- Grocery List ID

- Date Created

- List of Grocery Items

5. User Interface

- The main window shall display:

  - A list of storage locations.

  - A list of food items for the selected location.

- The interface shall include:

  - Buttons for adding, editing, and deleting food items.

  - Controls for sorting and filtering.

  - A search bar for item name searches.

- A grocery list screen shall allow:

  - Viewing and editing the current grocery list.

  - Saving and printing the grocery list.

- The interface shall be designed to be clean, intuitive, and easy to navigate.

6. Glossary

- **Food Item**: A product stored in the user's household inventory.

- **Location**: A storage area such as a pantry, refrigerator, or freezer.

- **Category**: A classification for food items such as dairy, produce, or snacks.

- **Freshness Status**: A calculated classification indicating whether a food item is fresh, expiring soon, or expired.

- **Grocery List**: A user-created list of items to be purchased.