**Group 3**

Title: Smart Fridge Application

Students:

Madhur Rajput

Willahelm Wan

**Top level description of the project**:

High Level Functionality: The application is used to track the inventory of the fridge.

Data Domain: Any items that can be kept inside of a fridge.

|  |  |
| --- | --- |
| Users interacting with database | Use Cases/Functionalities |
| **1. Owner of the fridge** |  |
| Inventory management | 1. Add an item to the fridge 2. Remove an item from the fridge 3. Pin an item 4. Display all items by Category |
| Provider management | 1. Add a provider 2. Remove a provider 3. Notification from the providers |
| Promotion management | 1. Display promotions 2. Add promotion to the promotion table 3. Remove promotion from the promotion table |
| Shopping List management | 1. Add an Item to Shopping list 2. Remove an item from shopping list 3. Clear shopping list 4. Display all items in shopping list 5. Price comparison among all providers 6. For each specific item in shopping list, by comparing price from different vendors, automatically cheapest provider per item would be selected for placing an order. 7. Manual ordering items from providers |
| Dynamic Ordering | 1. For all Pinned items whenever the quantity is below the threshold we will decide, those items will be automatically ordered and removed from the shopping list if they are in there. 2. Expiration display will be refreshed accordingly. |
| Processing items management | 1. Track ordered items 2. Add items upon manual ordering or automatic ordering 3. Delete items upon placing the items in the fridge |
| Processed items management | 1. Add items upon placing the items in the fridge 2. Maintain purchase history 3. Delete purchase history |
| Expiration Items management | 1. Retrieve expiration date of all items from the vendor database 2. Items already expired 3. Items that are about to expire (2 or 3 days tolerance) 4. Expiration date would be dynamically fetched from a central database managed by the provider instead of typing in manually. |
| **2. Vendors (for e.g. Amazon, Walmart)** |  |
| Promotion management | 1. Add a promotion 2. Manual removal a promotion 3. Automatic removal of a promotion upon the end date 4. Available items details |
| Product detail management | 1. Add product detail 2. Remove product detail |

**Tools:**

- Relational DB, MySQL

- Java, JSP for UI, Tomcat Apache Web Server, Javascript

- Alternative: Python Tkinter for UI

- No machine restrictions for this project

**Interest**:

This is a real-world example of an application that we can use in our daily life.