

UAT Testing for Database Enhancements

1. Initial Setup and Validation

*** For set up you can create new login information or use the ones I created: username: Manager password: test**

A. Install and Launch the Application

- ☐ Install the App: Ensure the application is installed on an Android device or emulator.
- ☐ Launch the App: Open the application to confirm that it runs without any crashes or errors.

2. Testing Inventory Entry Addition

A. Add New Inventory Items

- ☐ Navigate to Inventory Management: Go to the section where inventory items can be added.
- ☐ Add Items with Categories and Dates: Add several new inventory items, ensuring to fill in the item name, quantity, category, and verifying that the date is automatically added.

Example: Add an item named "Ice Cream", quantity 100, category "Dairy".

B. Verify Database Entry

- ☐ Check Display: Ensure that the new items appear in the inventory list with the correct category and date added.
- ☐ Database Inspection (Optional): If possible, use a database inspection tool (e.g., SQLite browser) to verify that the entries are correctly added to the database with the new columns.

3. Testing Inventory Trend Tracking

A. View Inventory Trends

- ☐ Navigate to Trend Section: Go to the Manager Dashboard section of the app that displays inventory trends.
- ☐ Check Trends Display: Verify that the trends are displayed correctly, showing the total quantity of items added over time.

Example: Ensure the dates and corresponding total quantities are accurately shown in a graph or list.

4. Testing Total Inventory by Category

A. View Inventory by Category

- ☐ **Navigate to Category Section:** Go to the section of the app that displays total inventory by category.
- ☐ **Check Categories Display:** Verify that the total quantities for each category are correctly calculated and displayed.

Example: If you added 100 items to the "Dairy" category and 5 to "Meats", ensure that the total for each category is accurate.

Files to check for database enhancements

1. Database Schema and Helper

A. DatabaseHelper.java

Location: `com.cs360.project3.DatabaseHelper`

Key Sections:

Database Schema Definition: Look at the `COLUMN_CATEGORY` and `COLUMN_DATE_ADDED` constants.

Table Creation: Check the `INVENTORY_TABLE_CONSTRAINTS` for the addition of the new columns.

onUpgrade Method: Ensure the upgrade logic correctly handles the addition of new columns without data loss.

Insert Methods: Review `insertInventoryEntry` to see how new columns are populated during insertion.

2. Database Queries

B. DatabaseQueries.java

Location: `com.cs360.project3.DatabaseQueries`

Key Sections:

getTotalInventoryByCategory: This method contains the SQL query to calculate the total inventory by category.

getInventoryTrend: This method contains the SQL query to track inventory trends over time, using the date_added column.

3. User Interface and Data Display

C. InventoryTrendFragment.java

Location: com.cs360.project3.InventoryTrendFragment

Key Sections:

loadInventoryTrendData: Method to load and display inventory trend data, integrating with the DatabaseQueries class.

D. CategoryDisplayFragment.java (or equivalent)

Location: com.cs360.project3.CategoryDisplayFragment

Key Sections:

loadCategoryData: Method to load and display total inventory by category, integrating with the DatabaseQueries class.

4. Data Models

E. InventoryEntry.java

Location: com.cs360.project3.InventoryEntry

Key Sections:

Constructor and Fields: Ensure the category and date_added fields are properly added and handled.

F. InventoryTrend.java

Location: com.cs360.project3.InventoryTrend

Key Sections:

Constructor and Fields: Ensure the class is designed to handle date and total quantity fields.