

Wants a proper user interface / GUI

Testing Plan:

Unit Tests:

1. **Add Product Functionality:** Inspect to see if the add function adds a new product to the database, double check to see if the product is saved correctly
2. **Delete Product Functionality:** Inspect to see if the delete/remove function removes a new product from the database, double check to see if the product is gone from the list
3. **Update/Edit Product Functionality:** Ensure updating existing item details works correctly.
4. **Search Functionality:**
 - a. Validate that the search function can successfully find an item when given the correct search parameters.
 - b. Confirm that the search function works regardless of the letter casing (case-insensitive search).
 - c. An empty search query returns no results or handles the edge case properly.
5. **Sort/Filter Functionality:**
 - a. Verify that the sort/filter function works correctly
 - b. Determine that filtering products by relevance is working as expected
6. **Login Functionality:**
 - a. Confirm that the login function works as anticipated, correctly authenticates valid users
 - b. Test that the username and password provided is valid
 - c. Make sure that edge cases are handled correctly whether it be a wrong username or password
7. **Overview of Inventory:**
 - a. Check that the inventory overview correctly displays the list of products in stock.
 - b. Ensure that the inventory overview includes key product details like name, price, and quantity.
 - c. Test that the inventory overview updates when items are added or removed from the database.

Integration Tests:

1. Login System & Inventory System - Valid user credentials

- a. Enter valid credentials (username and password) in the login form.
- b. Submit login request.
- c. Verify that the login system fetches user data and grants access to the inventory system.

2. Login System & Database - Invalid user credentials exist

- a. Enter incorrect credentials (wrong username/password combination).
- b. Submit the login form three times with the wrong credentials.
- c. Verify that the login system interacts with the database to check credentials and deny access.
- d. Check if the system displays an error message and exits after three failed login attempts.

3. Inventory System & File I/O - Inventory file exists

- a. Ensure that the inventory file is available.
- b. Trigger the system to load the inventory data from the file.
- c. Verify that the inventory system receives and correctly displays the data from the file.
- d. Ensure that the inventory system interacts with the file system to read the data and display it in the UI.

4. Inventory System & File I/O - The inventory File is missing

- a. Start the system with the inventory file missing.
- b. Check that the system handles this error by displaying an appropriate message.
- c. Verify that the system continues running despite the missing file, without crashing.

5. Inventory System & File-based Add Function - Valid inventory file exists

- a. Ensure a valid inventory file with products is available.
- b. Trigger the system to load products from the file.
- c. Verify that the inventory system correctly adds the products to the inventory list.
- d. Check that the products are integrated properly into the system without data loss.

6. Inventory System & File-based Add Function - Invalid inventory file exists

- a. Ensure an invalid inventory file with incorrect data is available.
- b. Trigger the system to load products from the incorrect file.
- c. Verify that the system detects the issue and displays an appropriate error message.
- d. Ensure the system continues to operate normally after the error.

System Tests:

- **Inventory stock alert - Product stock is low**
 - Check the inventory status report after some products have low stock levels.
 - Review the stock alert in the inventory overview.
 - Verify that the system generates and displays a clear stock alert for low-stock products.
 - Ensure the stock alert is shown in a user-friendly format on the UI.
- **Bulk product import - A valid inventory file exists**
 - Upload a valid inventory file that contains product data.
 - Trigger the bulk import process.

- Observe the product addition process from the user interface (UI).
- Check the inventory to ensure that all products are added successfully and that existing products have their stock levels increased accordingly.
- **Data persistence after restart - The system has active data**
 - Start the system with an active inventory.
 - Restart the system (close and reopen the application).
 - After the restart, check the inventory overview to confirm the data (inventory) is still intact.
 - Verify that all the product information is still present after the restart.
- **A Full run through of the entire system using every feature available**

Box Testing Responsibilities: Show charts from Phase 3

Let the Client run and test the build script

Show

Ask the customer what the final feature list they require by the time of our last customer meeting

Our Proposal for a feature:

User Roles and Permissions: Set up different user roles (e.g., admin, store manager, stock clerk) with specific permissions to access and modify different parts of the inventory. This adds security and control over who can make changes.

We want to propose an implementation of a user roles and permissions function, this function will set up different user roles, like admin, store manager, and stock clerk, with specific permissions.

This will add an extra level of security on who can make changes

Documentation Plan

We have structured our documentation into two key categories: Developer Documentation and User Documentation.

1. Developer Documentation

- **Responsible Team Members:** Abdullah & Tony
- **Purpose:** Provides technical insights, API references, system architecture, and code documentation.
- **Implementation:**
 - Stored in the project repository (GitHub Wiki / Markdown files).
 - Includes setup guides, API usage, and system architecture diagrams.
 - Maintained as part of the development cycle.

2. User/In-Application Documentation

- **Responsible Team Members:** Omar, Tisha, Vithunan
- **Purpose:** Provides users with instructions on how to use the system.
- **Implementation:**
 - Available as a PDF user manual.
 - Includes a quick start guide and troubleshooting steps if issues occur.
 - Accessible from the application's help menu.

Initial Documentation

User Manual: Step-by-step instructions for key features such as product management, search, and login.

Testing Documentation: Includes test cases and procedures for validation.

Finalization Plan: Regular updates until completion before the final customer meeting (updated in each lab document.

Show customer completed iteration 2 for approval:

- Login
- Sort/Filter
- Overview of inventory