INVENTORY MANAGEMENT SYSTEM

1. Inventory Management System Using Python Flask

This project is a web-based inventory management system developed using Python Flask. It focuses on automating warehouse operations, including product management, location tracking, and stock movement recording. The system provides real-time inventory tracking and a user-friendly interface, which helps warehouse staff manage stock efficiently without relying on manual records.

2. Abstract

The Inventory Management System allows users to add, edit, and delete products and warehouse locations, record stock movements, and generate real-time inventory reports. Built using Python Flask, SQLAlchemy, and Jinja2, it ensures accurate tracking of available, incoming, and outgoing stock. The application replaces manual tracking, reduces errors, and provides a secure and intuitive web interface for warehouse management. Users can view stock balances per location, monitor product movements, and maintain up-to-date inventory data efficiently.

3. Introduction

Warehouses require accurate inventory management to prevent stock-outs or overstocking. Manual systems are error-prone and slow. This project automates warehouse operations by allowing administrators to manage products, create warehouse locations, record stock movements, and generate detailed stock reports. The system ensures that each product is tracked across multiple locations, enabling users to monitor current stock levels, incoming shipments, and outgoing movements. It provides a web interface accessible via browsers, reducing the need for paperwork and improving operational efficiency.

4. Objectives

- Provide a Product Management Module for adding, editing, and deleting products.
- Implement a Location Management Module for managing warehouse locations.
- Record Product Movements between locations with optional from/to fields.
- Generate real-time stock reports showing available quantities per product per location.
- Ensure a user-friendly interface that minimizes manual errors and improves inventory visibility.

5. Tools and Technology used

Category	Tools and Technologies
Backend Framework	Python Flask
Frontend	HTML, CSS, Bootstrap
Database	SQLite
Forms	Flask-WTF
IDE/Editor	VS code

6. Modules Implemented

6.1 Product Management

- Add, edit, and delete products through a web interface.
- Each product has a unique ID, name, and description.
- Database integration ensures product records are linked to product movements.
- Provides a searchable and sortable product list in the interface.

6.2 Location Management

- Add, edit, and delete warehouse locations.
- Each location has a unique ID, name, and address.
- Locations are used in product movement records to track stock transfers.
- Provides a list view with edit and delete options, similar to the product module.

6.3 Product Movement Module

- Records stock movements between locations or as incoming/outgoing stock.
- Each movement contains product, from-location (optional), to-location (optional), quantity, and timestamp.
- Validations ensure numeric quantities and correct selection of product and locations.

- Movements are displayed in a table format with all relevant details.

6.4 Reporting

- Generates stock balance reports for all products across locations.
- Calculates incoming, outgoing, and current stock per location.
- Provides an overview of product availability and helps analyze warehouse operations.
- Enables administrators to make data-driven decisions for stock replenishment.

7. Database Design

7.1 Entities and Fields:

- **Product** – `product_id`, `name`, `description`, `created_at`
- **Location** – `location_id`, `name`, `address`, `created_at`
- **ProductMovement** – `movement_id`, `product_id`, `from_location`, `to_location`, `qty`, `timestamp`

7.2 Relationships:

- One Product → multiple ProductMovements
- One Location → source or destination in multiple movements

The database design ensures **referential integrity**, preventing inconsistent movement records and enabling accurate reporting.

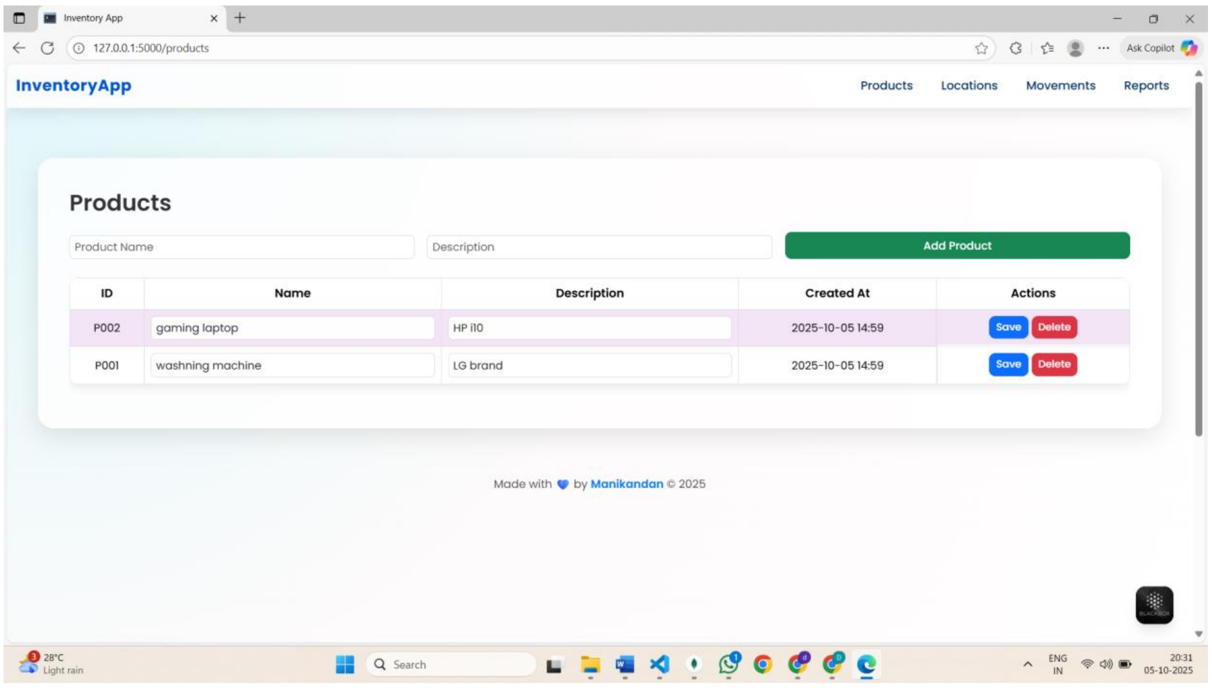
8. Flow of the Application

- Users log in and access the dashboard.
- Navigate to **Products** → add, edit, or delete products.
- Navigate to **Locations** → add, edit, or delete warehouse locations.
- Navigate to **Movements** → record product movements between locations.
- Navigate to **Report** → view real-time stock balances per product per location.

9. Flow of the Application

- **Warehouse Management:** Track stock for multiple products across multiple locations.
- **Stock Monitoring:** Monitor available stock in each location in real-time.
- **Error Reduction:** Simplify manual record-keeping, reducing human errors.
- **Operational Efficiency:** Speed up inventory tracking and reporting.
- **Decision Making:** Generate accurate reports to plan stock replenishment.

10. Screenshots



127.0.0.1:5000/products

InventoryApp

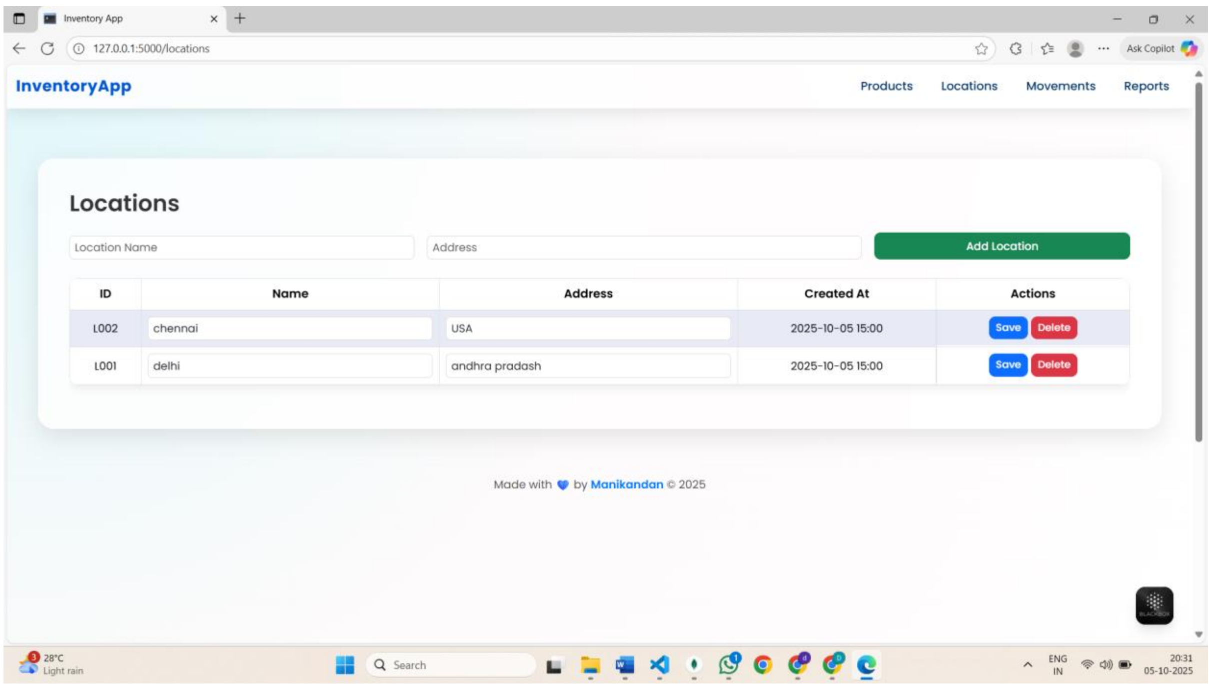
Products Locations Movements Reports

Products

ID	Name	Description	Created At	Actions
P002	gaming laptop	HP i10	2025-10-05 14:59	Save Delete
P001	washning machine	LG brand	2025-10-05 14:59	Save Delete

Made with 💙 by Manikandan © 2025

Figure 1: Add products



127.0.0.1:5000/locations

InventoryApp

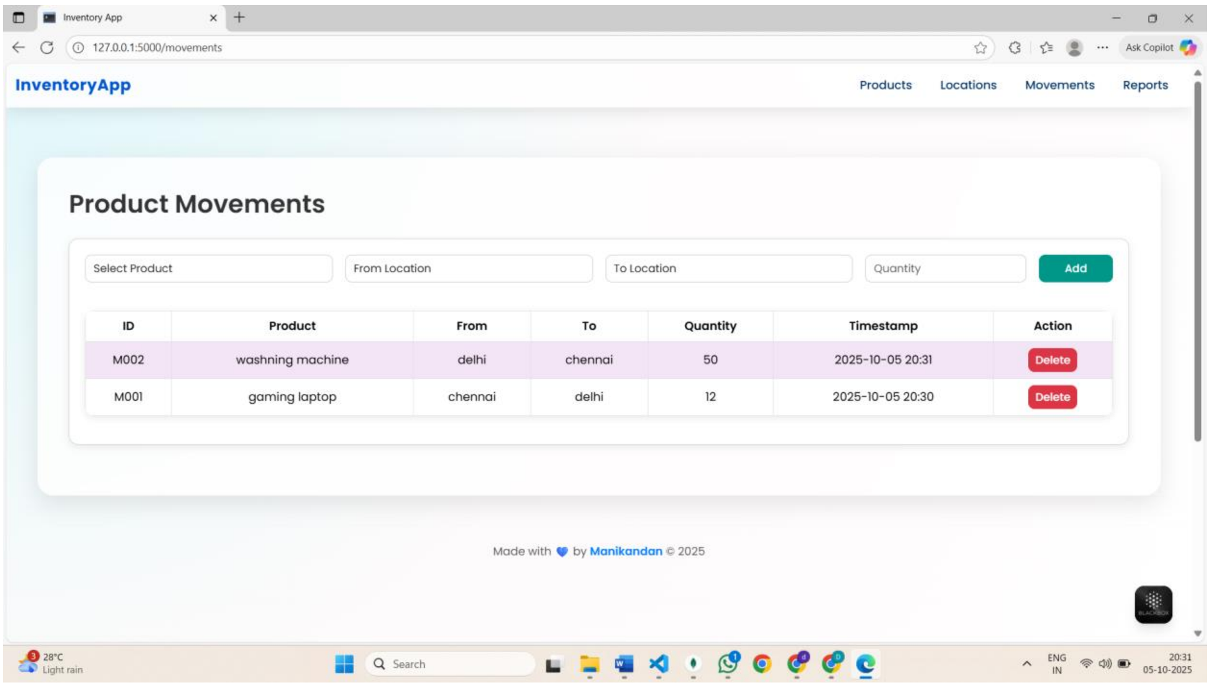
Products Locations Movements Reports

Locations

ID	Name	Address	Created At	Actions
L002	chennai	USA	2025-10-05 15:00	Save Delete
L001	delhi	andhra pradash	2025-10-05 15:00	Save Delete

Made with 💙 by Manikandan © 2025

Figure 2: Location adding



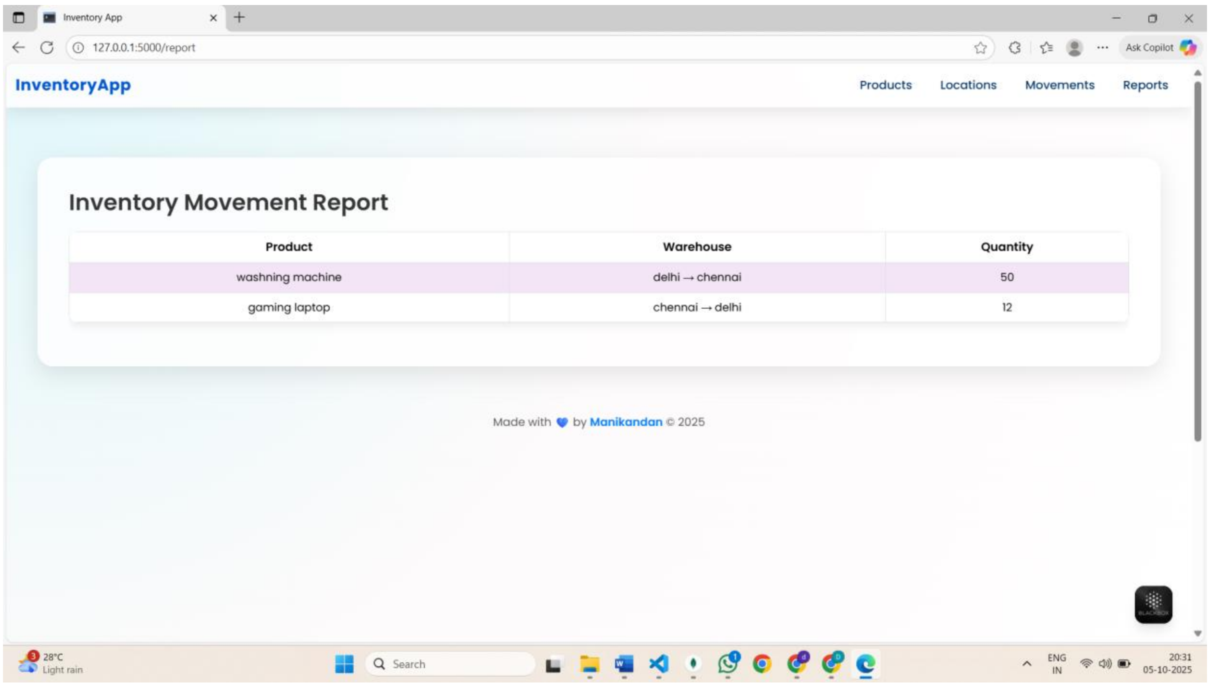
Product Movements

Select Product	From Location	To Location	Quantity	Add

ID	Product	From	To	Quantity	Timestamp	Action
M002	washning machine	delhi	chennai	50	2025-10-05 20:31	Delete
M001	gaming laptop	chennai	delhi	12	2025-10-05 20:30	Delete

Made with 💙 by Manikandan © 2025

Figure 3: Product Movement



Inventory Movement Report

Product	Warehouse	Quantity
washning machine	delhi → chennai	50
gaming laptop	chennai → delhi	12

Made with 💙 by Manikandan © 2025

Figure 4: Movement Report

11. Conclusion

The Inventory Management System provides a complete solution for warehouse inventory management, automating product tracking, location management, stock movements, and reporting. The system ensures accuracy, efficiency, and ease of use, reducing reliance on manual records. Real-time reporting helps administrators make informed decisions, track stock effectively, and manage warehouse operations smoothly. This project demonstrates how web-based tools can optimize inventory processes and improve operational efficiency, providing a scalable solution for small and medium-sized warehouses.