# **Medical Store Inventory System**

### **Deliverable 1**

### **Team: IntelliWare**

### **Team Members:**

* Amaar Khan 22i-0759– Scrum Master
* Saad Nadeem 22i-1030– Developer
* Mustafa Iqbal 22i1139– Team Lead

### **Submission Date: 28 February**

## Introduction

The Medical Store Inventory System is designed to manage the stock of medicines efficiently. It helps pharmacists, store managers, and staff track inventory levels, update stock, remove expired medicines, and generate reports. The system ensures the seamless management of medicine records, reducing errors and improving operational efficiency.

## Project Vision

Our vision is to develop a robust, user-friendly, and scalable Medical Store Inventory System that enhances stock management, prevents medicine shortages, and ensures the availability of essential medicines. The system will streamline pharmacy operations and improve service quality.

## Intended Use of the System

The system will be used by medical store personnel, including pharmacists, store managers, and cashiers, to efficiently manage inventory, update stock, and track expiry dates.

## Stakeholders & Their Needs

|  |  |
| --- | --- |
| Stakeholder | Needs |
| Pharmacist | Add, update, and delete medicines; track expiry dates. |
| Store Manager | Monitor inventory levels and ensure timely restocking. |
| Cashier | Verify stock availability before processing sales. |

## Features & Overall Functionality

- Add, update, and delete medicines in stock.

- Track medicine expiry dates and remove expired medicines.

- Generate inventory reports.

- Notify store managers of low stock levels.

- User authentication and role-based access control.

## User Stories

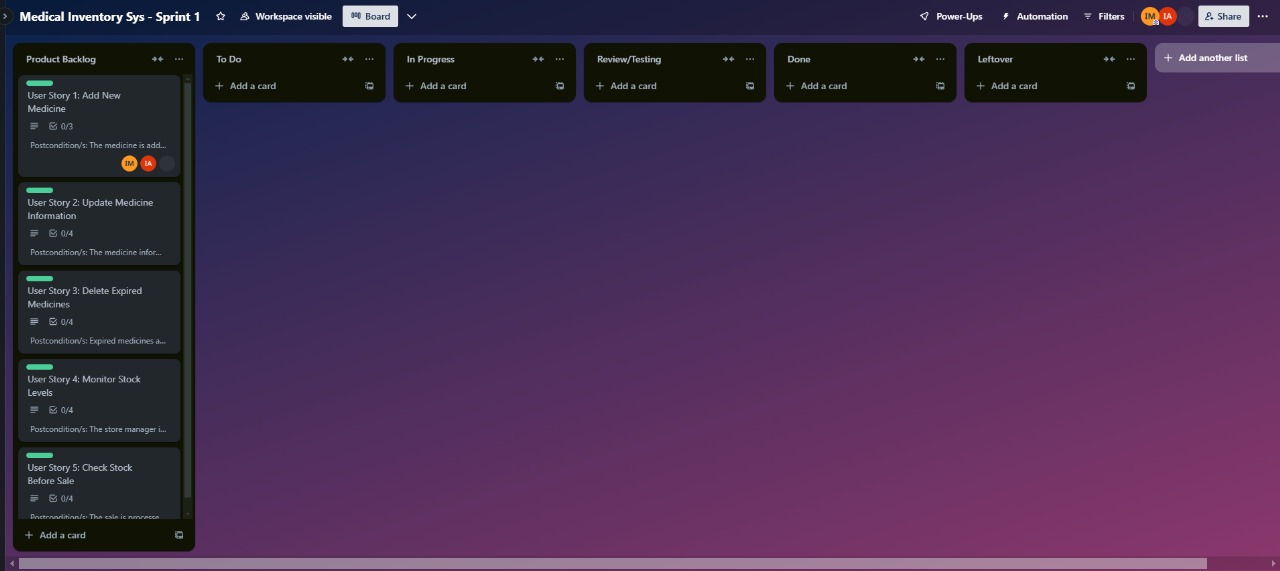
1. As a \*\*pharmacist\*\*, I want to \*\*add new medicines to the inventory\*\*, so that I can \*\*keep track of available stock\*\*.

2. As a \*\*store manager\*\*, I want to \*\*monitor stock levels\*\*, so that I can \*\*restock before medicines run out\*\*.

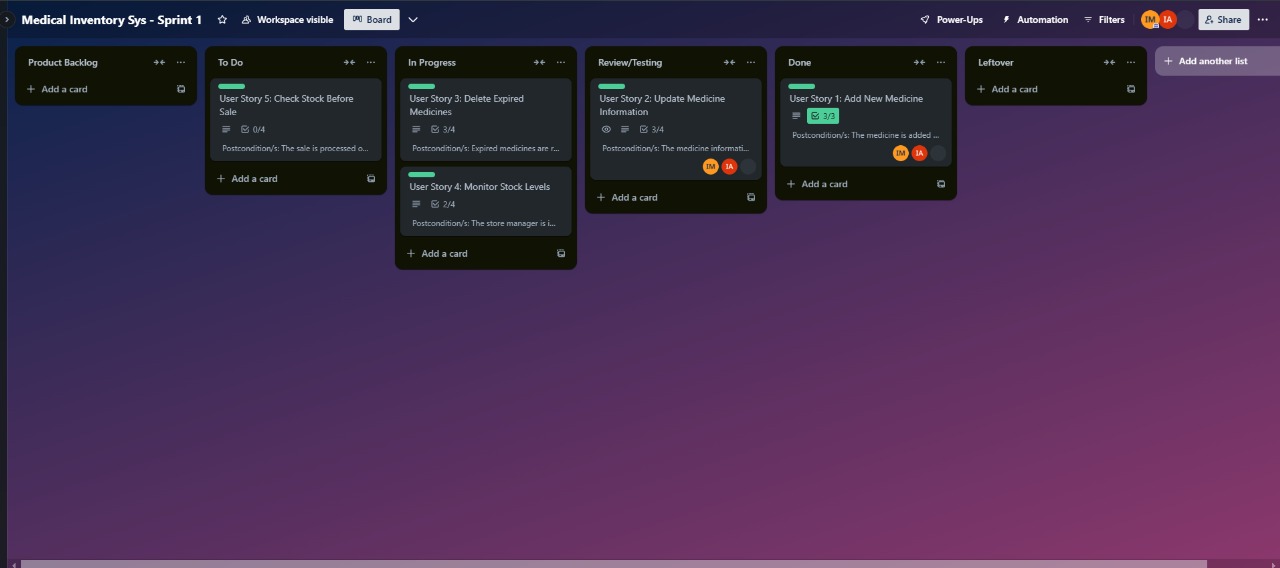
3. As a \*\*cashier\*\*, I want to \*\*check stock availability before processing a sale\*\*, so that I can \*\*ensure the medicine is in stock\*\*.

## Scrum Board Snapshots

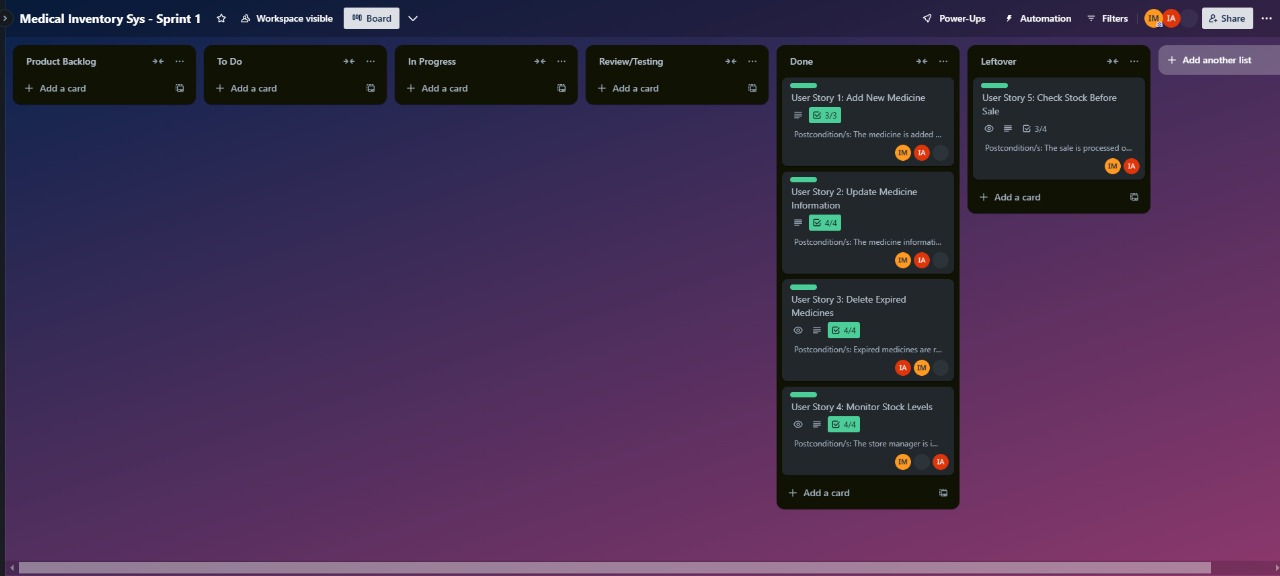
- \*\*Snapshot 1\*\*: Initial backlog uploaded.

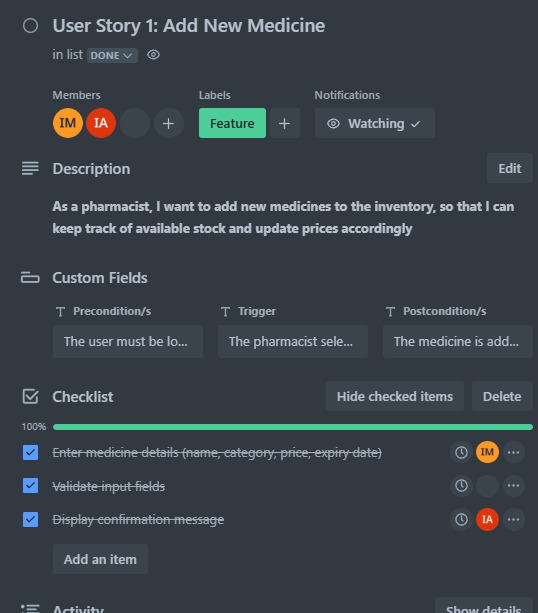


- \*\*Snapshot 2\*\*: Mid-sprint progress showing half-completed tasks.



- \*\*Snapshot 3\*\*: End of sprint with completed and leftover tasks.

Card example



## Non-Functional Requirements (NFRs)

- \*\*Performance\*\*: The system should handle at least 1000 inventory records efficiently.

- \*\*Security\*\*: Only authorized users can add, update, or delete medicines.

- \*\*Usability\*\*: The UI should be user-friendly and accessible for non-technical users.

- \*\*Availability\*\*: The system should be available 24/7 without major downtimes.

- \*\*Scalability\*\*: The system should allow the addition of more users and medicines without performance issues.

## Work Division

|  |  |
| --- | --- |
| Team Member | Task Assigned |
| Mustafa Iqbal (Team Lead) | Project Management, System Design. |
| Amaar Khan (Scrum Master) | Ensuring Agile Scrum Methodology, Sprint Planning. |
| Saad Nadeem (Developer) | Development and implementation of core functionalities. |

# Structured Specifications for All User Stories

## User Story 1: Add New Medicine

As a pharmacist, I want to add new medicines to the inventory, so that I can keep track of available stock and update prices accordingly.

\*\*Precondition:\*\* The relevant user must be logged into the system.

\*\*Trigger:\*\* The user accesses the respective feature from the inventory system menu.

\*\*Normal Flow:\*\*

1. The system displays the corresponding interface for the action.

2. The user performs the necessary actions (adding, updating, deleting, monitoring).

3. The system validates inputs and executes the task accordingly.

\*\*Postcondition:\*\* The action is successfully completed and updated in the inventory system.

## User Story 2: Update Medicine Information

As a pharmacist, I want to update existing medicine details, so that I can keep the stock information accurate and up to date.

\*\*Precondition:\*\* The relevant user must be logged into the system.

\*\*Trigger:\*\* The user accesses the respective feature from the inventory system menu.

\*\*Normal Flow:\*\*

1. The system displays the corresponding interface for the action.

2. The user performs the necessary actions (adding, updating, deleting, monitoring).

3. The system validates inputs and executes the task accordingly.

\*\*Postcondition:\*\* The action is successfully completed and updated in the inventory system.

## User Story 3: Delete Expired Medicines

As a store manager, I want to remove expired medicines from the system, so that I can ensure no expired products are sold to customers.

\*\*Precondition:\*\* The relevant user must be logged into the system.

\*\*Trigger:\*\* The user accesses the respective feature from the inventory system menu.

\*\*Normal Flow:\*\*

1. The system displays the corresponding interface for the action.

2. The user performs the necessary actions (adding, updating, deleting, monitoring).

3. The system validates inputs and executes the task accordingly.

\*\*Postcondition:\*\* The action is successfully completed and updated in the inventory system.

## User Story 4: Monitor Stock Levels

As a store manager, I want to monitor stock levels, so that I can ensure timely restocking of medicines before they run out.

\*\*Precondition:\*\* The relevant user must be logged into the system.

\*\*Trigger:\*\* The user accesses the respective feature from the inventory system menu.

\*\*Normal Flow:\*\*

1. The system displays the corresponding interface for the action.

2. The user performs the necessary actions (adding, updating, deleting, monitoring).

3. The system validates inputs and executes the task accordingly.

\*\*Postcondition:\*\* The action is successfully completed and updated in the inventory system.

## User Story 5: Check Stock Before Sale

As a cashier, I want to check stock availability before processing a sale, so that I can ensure that the requested medicine is in stock before billing the customer.

\*\*Precondition:\*\* The relevant user must be logged into the system.

\*\*Trigger:\*\* The user accesses the respective feature from the inventory system menu.

\*\*Normal Flow:\*\*

1. The system displays the corresponding interface for the action.

2. The user performs the necessary actions (adding, updating, deleting, monitoring).

3. The system validates inputs and executes the task accordingly.

\*\*Postcondition:\*\* The action is successfully completed and updated in the inventory system.