Phase 3 : Project Design Phase

Project Title : Medical Inventory Management System

Introduction

The **Project Design Phase** defines how the Medical Inventory Management System will be structured and implemented on the Salesforce platform. This phase focuses on designing the system's architecture, database (objects and fields), workflows, user interfaces, and automation features. The goal is to create a simple, efficient, and secure CRM-based solution that helps healthcare facilities manage their medical stock effectively.

System Architecture

The Medical Inventory Management System follows a **three-layer architecture** within Salesforce:

1. **Presentation Layer:**
   - Built using Salesforce **Lightning App Builder** and **Lightning Components**.
   - Provides a user-friendly interface for managing medicines, suppliers, and purchase records.
2. **Application Layer:**
   - Contains business logic implemented using **Apex Classes**, **Triggers**, **Validation Rules**, and **Flows**.
   - Handles operations like stock updates, expiry alerts, and automatic notifications.
3. **Data Layer:**
   - Stores all data in **Salesforce Custom and Standard Objects** such as Medicine, Supplier, and Purchase Record.
   - Data relationships ensure integrity and easy access across modules.

System Components

A. Custom Objects

1. **Medicine:** Stores details like medicine name, batch number, quantity, and expiry date.
2. **Supplier:** Contains information about vendors, contact details, and supplied items.

3. **Purchase Record:** Logs purchase transactions, quantity purchased, and supplier reference.
4. **Inventory Transaction:** Tracks medicine usage, stock-in, and stock-out details.

B. Relationships

- **Supplier ↔ Medicine:** Lookup relationship (to identify which supplier provides which medicines).
- **Medicine ↔ Purchase Record:** Master-detail relationship (each purchase is linked to a medicine).
- **Medicine ↔ Inventory Transaction:** Master-detail relationship (to track stock movement).

Automation and Workflows

Automation is a key component in Salesforce-based inventory management:

1. **Salesforce Flow / Process Builder:**
    - Sends automatic alerts for low stock or upcoming expiry.
    - Updates status fields when stock levels change.
2. **Apex Triggers:**
    - Automatically decreases stock when an item is used or increases it upon purchase.
3. **Validation Rules:**
    - Prevent entry of expired medicine records.
    - Ensure all mandatory fields are filled before saving.
4. **Approval Process (Optional):**
    - Used for approving new medicine entries or purchase requests.

User Interface Design

- **Lightning App Tabs:**
    - Home
    - Medicine
    - Supplier
    - Purchase Record

    - Reports and Dashboards

- **Dashboard View:**
  Displays key metrics like:
    - Total available stock
    - Expired and low-stock medicines
    - Purchase statistics
- **Reports:**
    - **Stock Report:** Displays all available medicines with quantity and expiry date.
    - **Purchase Summary:** Shows medicines bought and suppliers involved.
    - **Usage Report:** Tracks consumption trends.

Security Design

To maintain data privacy and control access:

1. **Profiles and Roles:** Define who can view, edit, or delete records.
2. **Field-Level Security:** Protects sensitive fields such as pricing or supplier data.
3. **Record-Level Sharing:** Ensures only authorized users can access specific inventory data.

Expected Outcome

- A fully designed Salesforce CRM application structure for managing medical inventory.
- Automated workflows that ensure real-time updates and notifications.
- User-friendly dashboards and reports for analysis and decision-making.
- Secure and scalable system suitable for healthcare environments.

Conclusion

The Project Design Phase establishes the blueprint for the **Medical Inventory Management System** on the **Salesforce platform**. It ensures that the system is functionally sound, well-organized, and ready for development. With clear data models, automated workflows, and secure access controls, the design phase lays the foundation for efficient implementation and smooth project execution.