Medical Inventory Management

SANTHOSH KUMAR N	**911222104308**
VENUGOPAL P	**911222104046**
SIVAKUMAR P	**911222104037**
MATHAN M	**911222104022**

Project Overview:

The Medical Inventory Management System is a comprehensive Salesforce application designed to streamline and manage various operational aspects of the medical inventory. It can efficiently maintain supplier details, manage purchase orders, track product details and transactions, and monitor expiry dates of products, thereby improving operational efficiency, data accuracy, and reporting capabilities.

This project is a comprehensive Salesforce application to streamline and manage various operational aspects of medical inventory. The system aims to efficiently maintain supplier details, manage purchase orders, track product details and transactions, and monitor the expiry dates of products. Maintain detailed records of suppliers, including contact information. Catalog product information, including descriptions, stock levels. Monitor and track product expiry dates to avoid using expired items. Comprehensive reports to track supplier performance, and purchase orders.

Objectives:

Efficient Inventory Control

To streamline and automate the management of medical inventory, ensuring accurate tracking of stock levels and minimizing manual errors.

Supplier and Purchase Order Management

To maintain detailed supplier records and manage purchase orders effectively for timely procurement and supplier performance tracking.

Product Tracking and Expiry Monitoring

To monitor product details, including batch numbers and expiry dates, to prevent the use of expired or low-quality medical items.

Data Accuracy and Centralization

To ensure all supplier, product, and transaction data is accurately recorded and stored within Salesforce for easy access and reliability.

Comprehensive Reporting and Decision Support

To provide real-time reports and dashboards for analyzing inventory trends, supplier performance, and purchase efficiency, supporting informed decision-making.

Detailed Steps to Solution Design:-

Ideation Phase:-

Step 1: Developer Account Setup

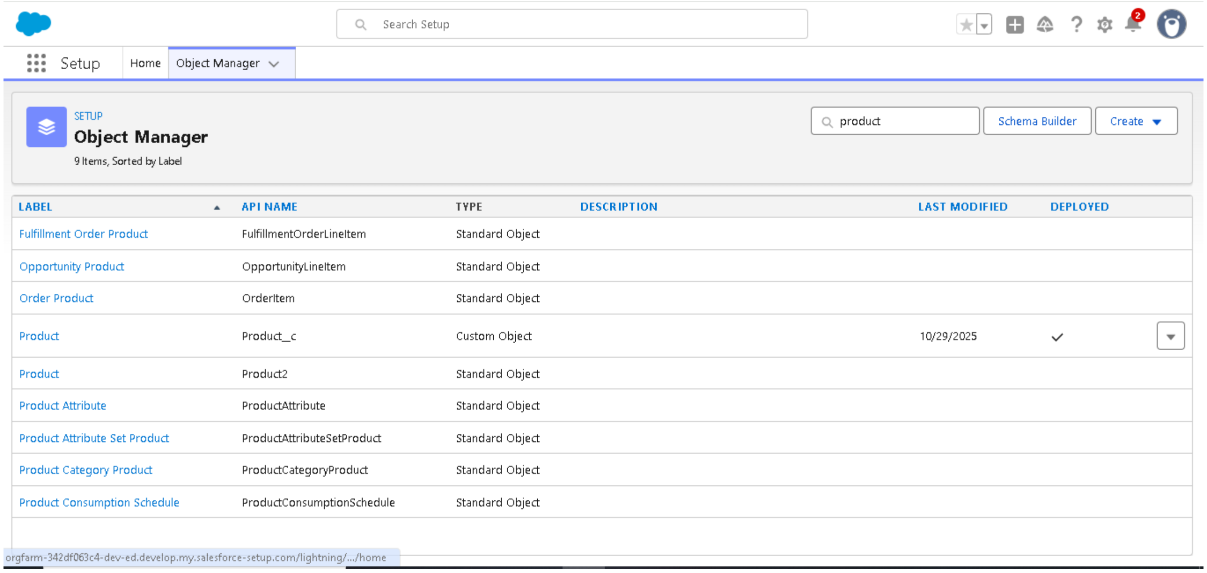
- Registered for a Salesforce Developer account to create a dedicated environment for development and testing.

- Verified the account to unlock full access to Salesforce features, ensuring a smooth setup process.

Step 2: Custom Object Creation

- Used **Salesforce Object Manager** to create custom objects for **Product**, **Purchase Order**, **Order Item**, **Inventory Transaction**, and **Supplier** to manage various aspects of medical inventory.

- Configured fields such as **text**, **numbers**, **dates**, and **relationships** to capture essential details, ensuring each object accurately represents the operational requirements of medical inventory management.



SETUP
Object Manager
9 Items, Sorted by Label

LABEL	API NAME	TYPE	DESCRIPTION	LAST MODIFIED	DEPLOYED
Fulfillment Order Product	FulfillmentOrderLineItem	Standard Object			
Opportunity Product	OpportunityLineItem	Standard Object			
Order Product	OrderItem	Standard Object			
Product	Product__c	Custom Object		10/29/2025	✓
Product	Product2	Standard Object			
Product Attribute	ProductAttribute	Standard Object			
Product Attribute Set Product	ProductAttributeSetProduct	Standard Object			
Product Category Product	ProductCategoryProduct	Standard Object			
Product Consumption Schedule	ProductConsumptionSchedule	Standard Object			

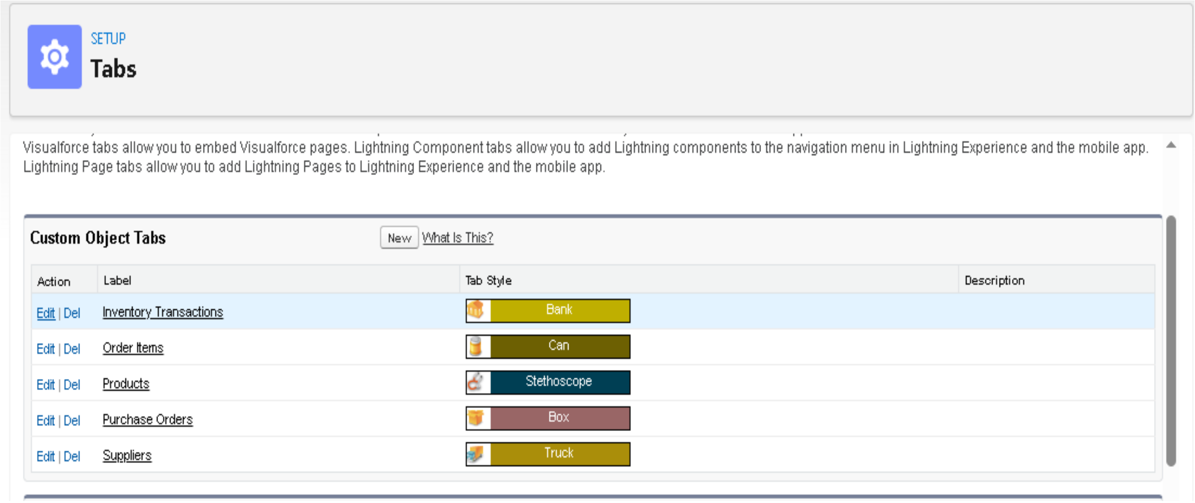
orgfarm-342df063c4-dev-ed.develop.my.salesforce-setup.com/lightning/.../home

Project Planning Phase:-

Step 3: Custom Tabs for Navigation

- Created custom tabs for objects — **Product, Purchase Order, Order Item, Inventory Transaction,** and **Supplier** using **Salesforce Setup** → Tabs, enhancing navigation and data accessibility.

- Configured tab styles and visibility settings to ensure easy access and seamless management of each object within the application.



SETUP
Tabs

Visualforce tabs allow you to embed Visualforce pages. Lightning Component tabs allow you to add Lightning components to the navigation menu in Lightning Experience and the mobile app. Lightning Page tabs allow you to add Lightning Pages to Lightning Experience and the mobile app.

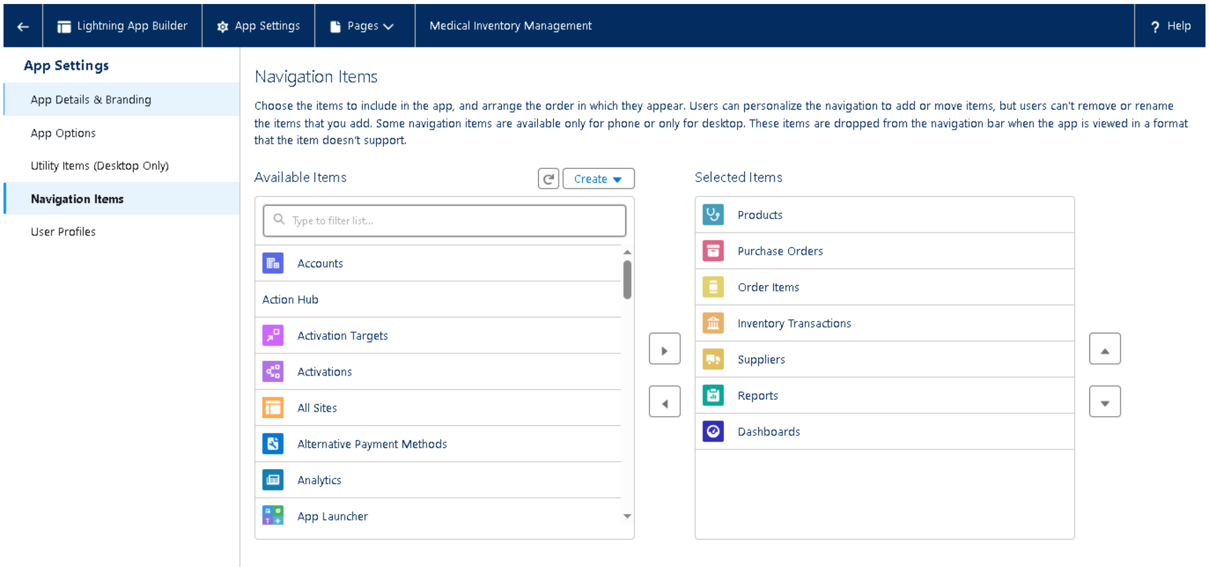
Custom Object Tabs

Action	Label	Tab Style	Description
Edit | Del	Inventory Transactions	Bank	
Edit | Del	Order Items	Can	
Edit | Del	Products	Stethoscope	
Edit | Del	Purchase Orders	Box	
Edit | Del	Suppliers	Truck	

Step 4: Lightning App Development

- Created a **Lightning App** named **Medical Inventory Management** in **Salesforce App Manager**, uploading a relevant image and configuring default app options.

- Added **Products, Purchase Orders, Order Items, Inventory Transactions, Suppliers, Reports, and Dashboards** tabs, assigning access to the **System Administrator** profile for efficient management.



Navigation Items

Choose the items to include in the app, and arrange the order in which they appear. Users can personalize the navigation to add or move items, but users can't remove or rename the items that you add. Some navigation items are available only for phone or only for desktop. These items are dropped from the navigation bar when the app is viewed in a format that the item doesn't support.

Available Items

- Accounts
- Action Hub
- Activation Targets
- Activations
- All Sites
- Alternative Payment Methods
- Analytics
- App Launcher

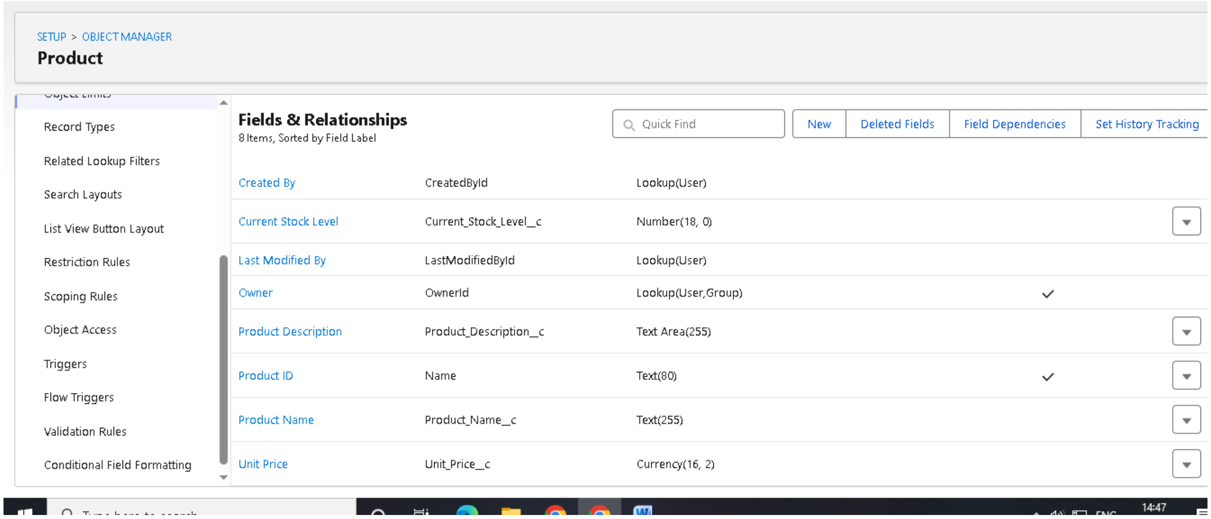
Selected Items

- Products
- Purchase Orders
- Order Items
- Inventory Transactions
- Suppliers
- Reports
- Dashboards

Step 5: Field Configuration

- Created **custom fields** for objects — **Product**, **Purchase Order**, **Order Item**, **Inventory Transaction**, and **Supplier** in **Salesforce Object Manager**, defining appropriate data types and relationships.

- Configured fields such as **Text, Number, Currency, Date, Lookup, Master-Detail, Formula, Roll-Up Summary, Picklist, and Text Area** to capture detailed information for efficient management of medical inventory operations.



SETUP > OBJECT MANAGER

Product

Fields & Relationships

8 Items, Sorted by Field Label

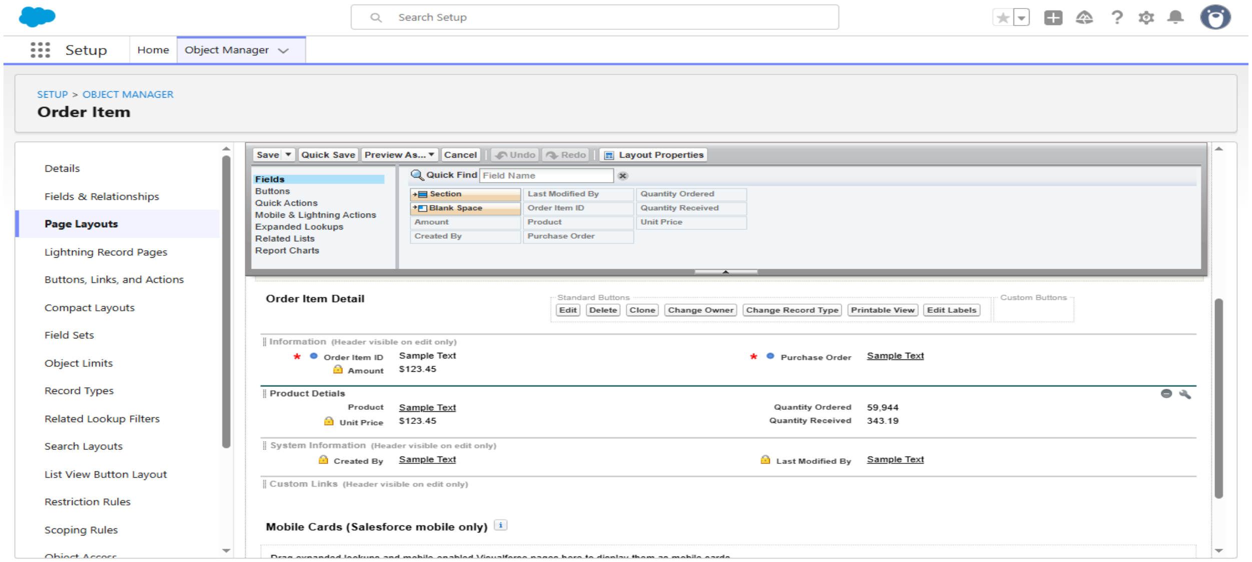
Created By	CreatedById	Lookup(User)	
Current Stock Level	Current_Stock_Level__c	Number(18, 0)	
Last Modified By	LastModifiedById	Lookup(User)	
Owner	OwnerId	Lookup(User,Group)	✓
Product Description	Product_Description__c	Text Area(255)	
Product ID	Name	Text(80)	✓
Product Name	Product_Name__c	Text(255)	
Unit Price	Unit_Price__c	Currency(16, 2)	

Project Design Phase:-

Step 6: Page Layout Configuration

- Edited **page layouts** for all custom objects — **Product, Purchase Order, Order Item, Inventory Transaction, and Supplier** to organize fields and improve usability.

- Customized layouts to ensure important details are easily accessible, enhancing user efficiency and data visibility within the Salesforce app.



SETUP > OBJECT MANAGER

Order Item

Order Item Detail

Information (Header visible on edit only)

* Order Item ID	Sample Text	* Purchase Order	Sample Text
Amount	$123.45		

Product Detials

Product	Sample Text	Quantity Ordered	59,944
Unit Price	$123.45	Quantity Received	343.19

System Information (Header visible on edit only)

Created By	Sample Text	Last Modified By	Sample Text

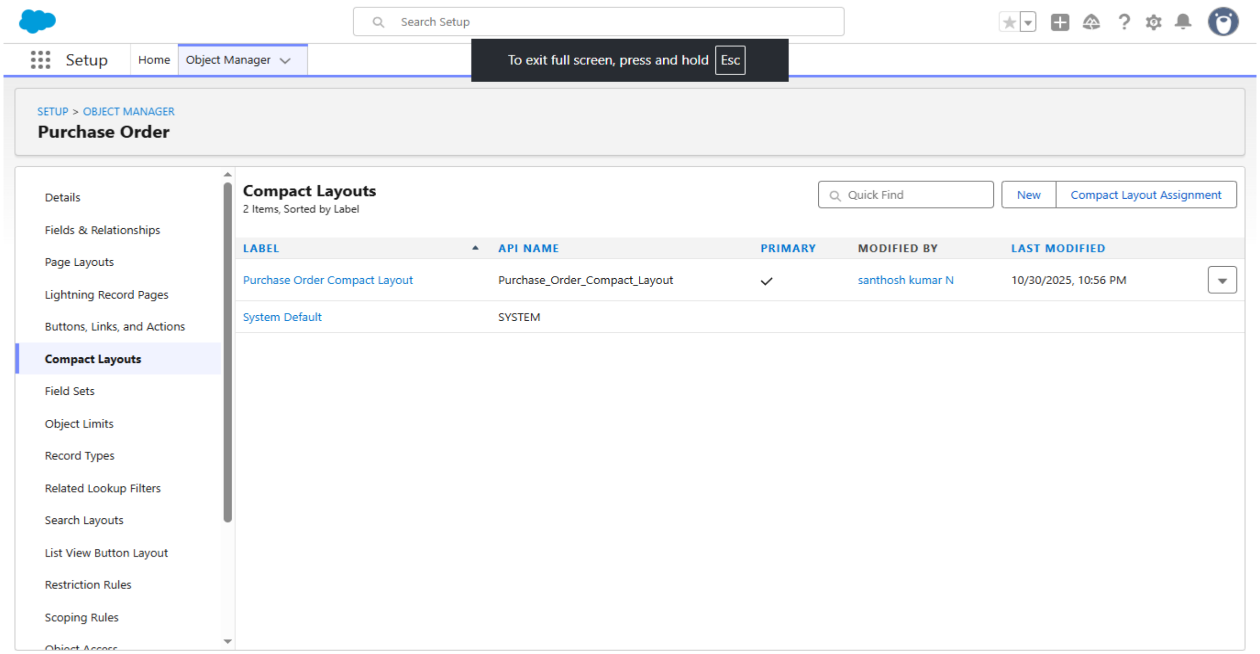
Custom Links (Header visible on edit only)

Mobile Cards (Salesforce mobile only)

Step 7: Compact Layout Configuration

- Created **Compact Layouts** for **Product** and **Purchase Order** objects in Salesforce to display key information in record highlights.

- Configured fields — **Product Name, Unit Price, Current Stock Level** for Product and **Purchase Order ID, Order Date, Total Order Cost, Supplier ID** for Purchase Order, then assigned each as the default compact layout.



SETUP > OBJECT MANAGER

Purchase Order

Compact Layouts

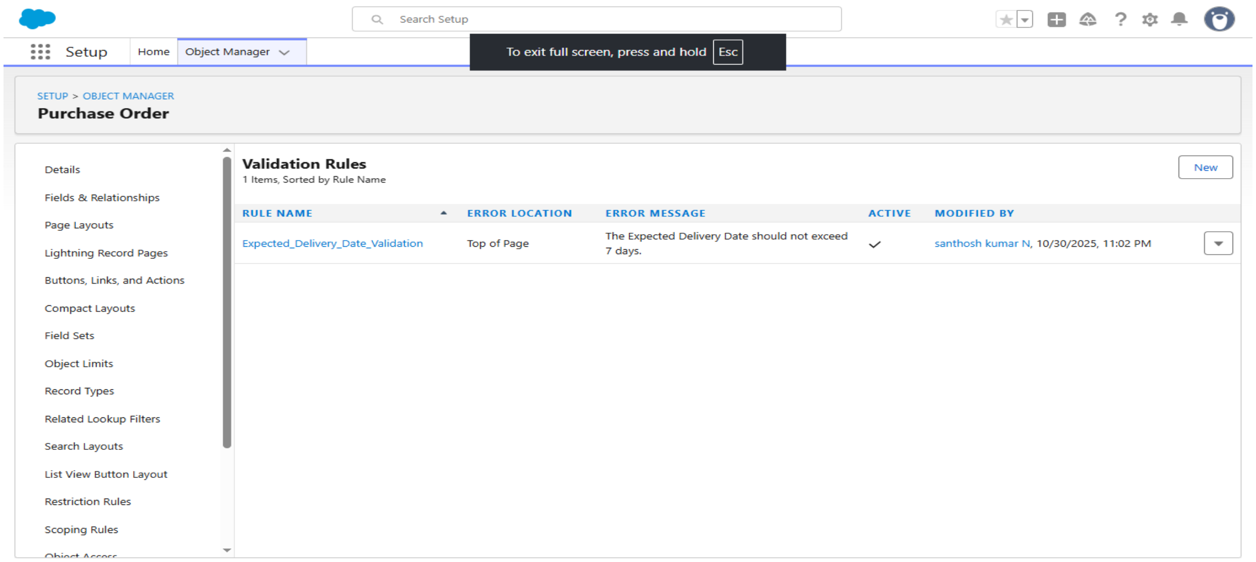
2 Items, Sorted by Label

LABEL	API NAME	PRIMARY	MODIFIED BY	LAST MODIFIED
Purchase Order Compact Layout	Purchase_Order_Compact_Layout	✓	santhosh kumar N	10/30/2025, 10:56 PM
System Default	SYSTEM			

Step 8: Validation Rules

- **Navigate and create validation rule:** Go to **Setup** → **Object Manager** → **Purchase Order** → **Validation Rules** → **New**, enter **Rule Name** as *Expected Delivery Date Validation*, select **Active**, and set **Error Condition Formula** to (Expected_Delivery_Date__c - Order_Date__c) > 7.

- **Set error details and save:** Enter **Error Message** as *"The Expected Delivery Date should not exceed 7 days."*, choose **Error Location** as *Top of Page*, and click **Save**.



Validation Rules

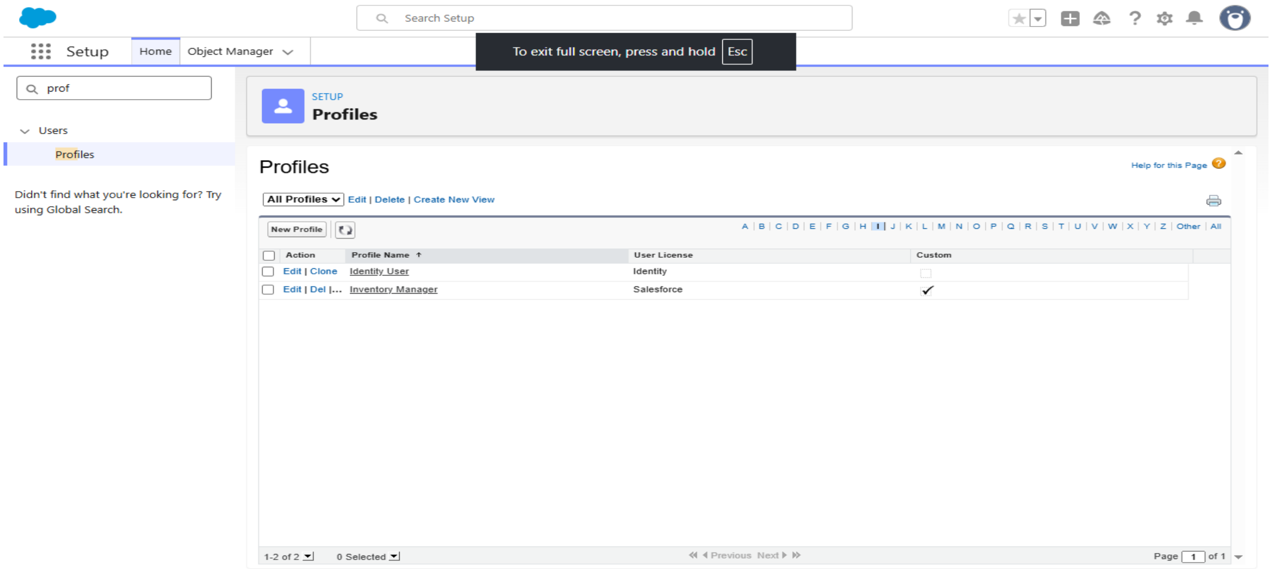
1 Items, Sorted by Rule Name

RULE NAME	ERROR LOCATION	ERROR MESSAGE	ACTIVE	MODIFIED BY
Expected_Delivery_Date_Validation	Top of Page	The Expected Delivery Date should not exceed 7 days.	✓	santhosh kumar N, 10/30/2025, 11:02 PM

Step 9: Profile Creation

- Go to **Setup** → **Profiles** (search in Quick Find), clone the desired profile (Standard User), enter **Profile Name** as Inventory Manager, and click **Save**. Scroll down to **Custom Object Permissions** and assign access as per the diagram.

- Change password policies by setting **User Passwords Expire In** to Never *Expires* and **Minimum Password Length** to *8*, then click **Save**.



Search Setup
Setup
Home
Object Manager
To exit full screen, press and hold Esc
prof
Users
Profiles
Didn't find what you're looking for? Try using Global Search.
SETUP
Profiles
Help for this Page
New Profile
A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | Other | All
1-2 of 2
0 Selected
Previous Next
Page 1 of 1

Profiles

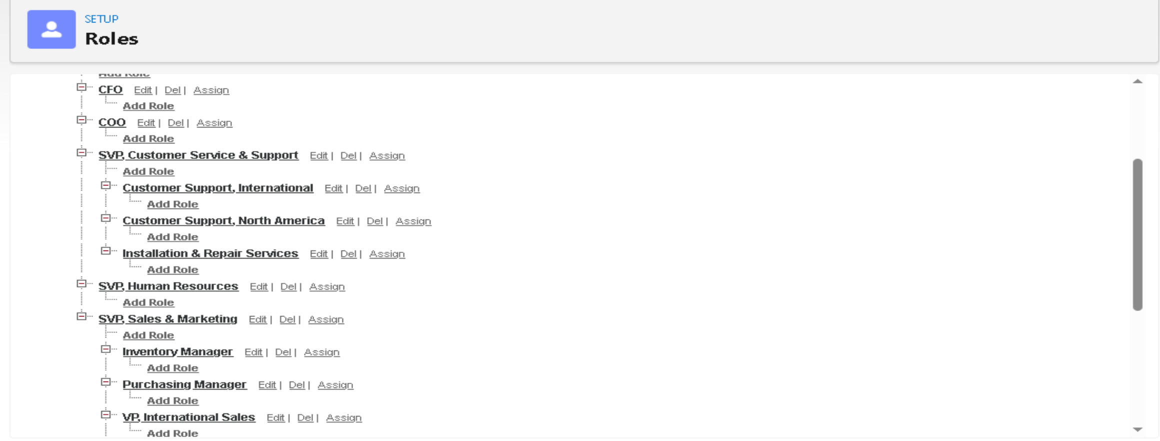
All Profiles | Edit | Delete | Create New View

□ Action	Profile Name ↑	User License	Custom
□ Edit | Clone	Identity User	Identity	□
□ Edit | Del |...	Inventory Manager	Salesforce	✓

Requirement Analysis:-

Step 10: Role Creation

- Create a Purchasing Manager Role.
- Create an additional Purchasing Manager Role.



SETUP

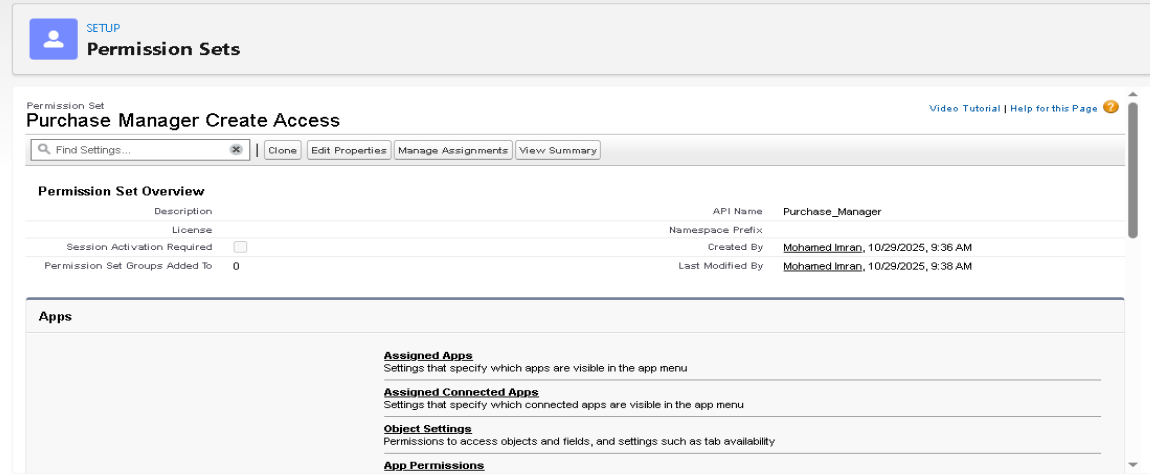
Roles

- CFO Edit | Del | Assign
  - Add Role
- COO Edit | Del | Assign
  - Add Role
- SVP, Customer Service & Support Edit | Del | Assign
  - Add Role
  - Customer Support, International Edit | Del | Assign
    - Add Role
  - Customer Support, North America Edit | Del | Assign
    - Add Role
  - Installation & Repair Services Edit | Del | Assign
    - Add Role
- SVP, Human Resources Edit | Del | Assign
  - Add Role
- SVP, Sales & Marketing Edit | Del | Assign
  - Add Role
  - Inventory Manager Edit | Del | Assign
    - Add Role
  - Purchasing Manager Edit | Del | Assign
    - Add Role
  - VP, International Sales Edit | Del | Assign
    - Add Role

Step 11: Permission set Creation

- Go to **Setup → Permission Sets → New**, create the Purchase Manager Create Access permission set, enable **Order Item** tab and **Read/Create** permissions, then save.

- Assign this permission set to user John PurchaseM with **No Expiration Date** via **Manage Assignments → Add Assignments**



Permission Set

Purchase Manager Create Access

Permission Set Overview

Description		API Name	Purchase_Manager
License		Namespace Prefix	
Session Activation Required	□	Created By	Mohamed Imran, 10/29/2025, 9:36 AM
Permission Set Groups Added To	0	Last Modified By	Mohamed Imran, 10/29/2025, 9:38 AM

Apps

Assigned Apps
Settings that specify which apps are visible in the app menu

Assigned Connected Apps
Settings that specify which connected apps are visible in the app menu

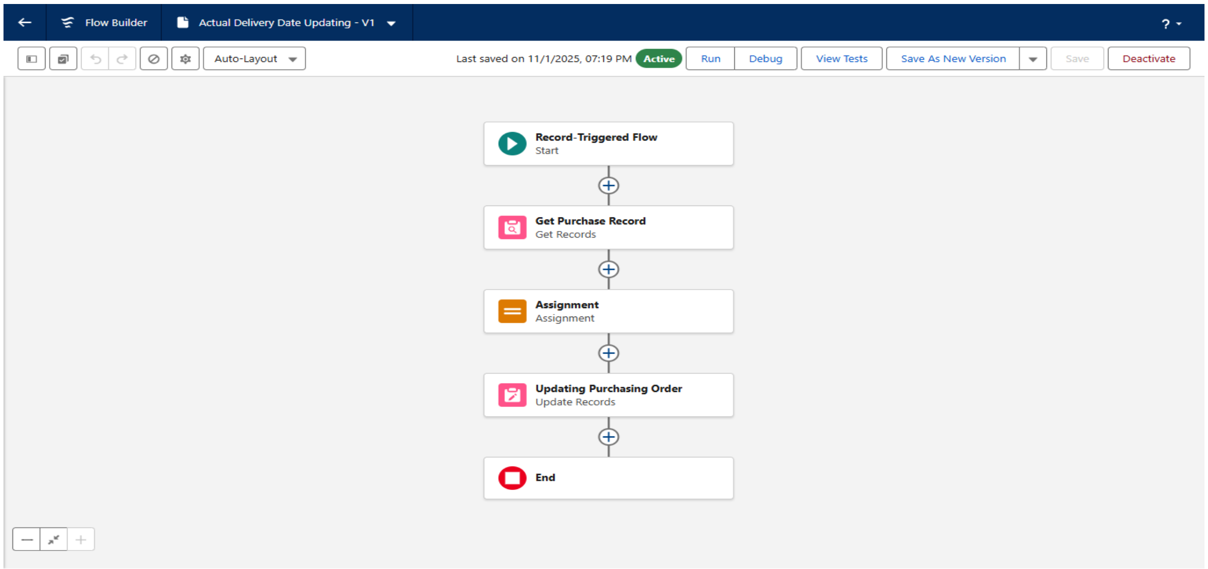
Object Settings
Permissions to access objects and fields, and settings such as tab availability

App Permissions

Step 12: Flow Creation

- Go to **Setup** → **Flow** → **New Flow** → **Record-Triggered Flow**, select **Purchase Order** object with trigger **A record is created or updated**, set fast field updates, then add **Get Records** to fetch the purchase order by Id retrieving the **Order_Date__c** field.

- Create a **Date variable** (ActualDeliveryDate), use an **Assignment** element to set it to Order_Date__c plus 3 days, save the flow as *Actual Delivery Date Updating*, and activate it.



Performance Testing:-

Step 13: Triggers

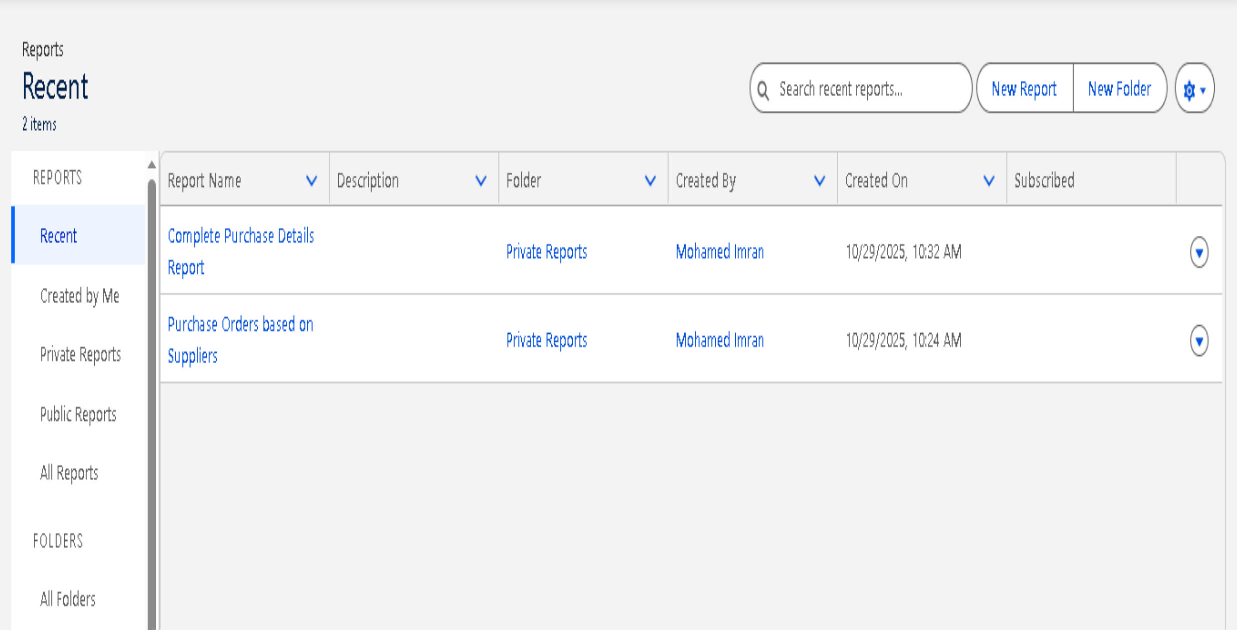
- Login to Salesforce, open **Developer Console** → **File** → **New** → **Apex Trigger**, create CalculateTotalAmountTrigger on Order_Item__c to call the handler class.

- Then create **Apex Class** → **CalculateTotalAmountHandler** to calculate and update Total_Order_cost__c on related Purchase_Order__c records based on Order_Item__c changes, and save both.

Step 14: Reports Creation

- Create a **Purchase Orders Summary Report** by selecting **Purchase Orders with Order Items**, applying filters, grouping by Supplier ID, Delivery Date, and Purchase Order ID, then adding Product ID, Name, Order Count, Quantity, and Amount.

- Create a **Complete Purchase Details Report** by including all relevant columns and saving it as **Complete Purchase Details Report**.

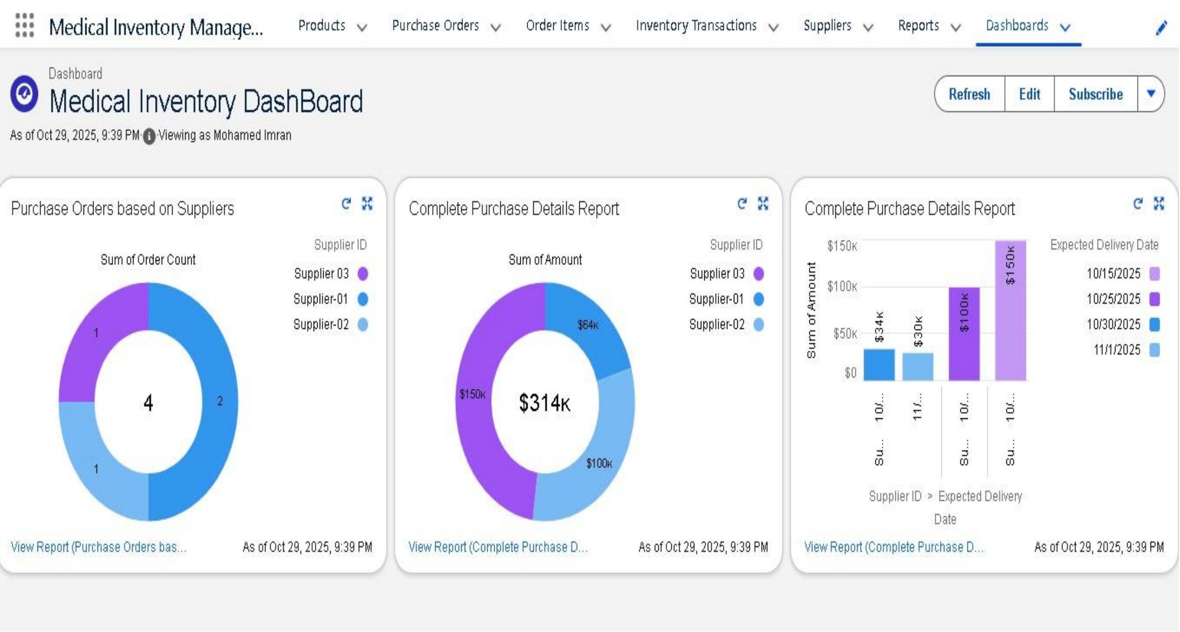


Report Name	Description	Folder	Created By	Created On	Subscribed
Complete Purchase Details Report		Private Reports	Mohamed Imran	10/29/2025, 10:32 AM	
Purchase Orders based on Suppliers		Private Reports	Mohamed Imran	10/29/2025, 10:24 AM	

Step 15: Dashboard Creation

- Create a **Medical Inventory Dashboard** by clicking **Dashboards → New Dashboard**, naming it, and clicking **Create**.

- Add a widget using the **Purchase Orders based on Suppliers** report, choose a chart or table for visualization, then click **Add** and **Save**.


Medical Inventory Manage...
Products
Purchase Orders
Order Items
Inventory Transactions
Suppliers
Reports
Dashboards
Dashboard
Medical Inventory DashBoard
As of Oct 29, 2025, 9:39 PM · Viewing as Mohamed Imran
Refresh
Edit
Subscribe
Purchase Orders based on Suppliers
Sum of Order Count
Supplier ID
Supplier 03
Supplier-01
Supplier-02
1
2
1
4
View Report (Purchase Orders bas...
As of Oct 29, 2025, 9:39 PM
Complete Purchase Details Report
Sum of Amount
Supplier ID
Supplier 03
Supplier-01
Supplier-02
$64K
$150K
$314K
$100K
View Report (Complete Purchase D...
As of Oct 29, 2025, 9:39 PM
Complete Purchase Details Report
Sum of Amount
$150K
$100K
$50K
$0
$34K
$30K
$100K
$150K
10/...
11/...
10/...
10/...
Su...
Su...
Su...
Supplier ID > Expected Delivery Date
Expected Delivery Date
10/15/2025
10/25/2025
10/30/2025
11/1/2025
View Report (Complete Purchase D...
As of Oct 29, 2025, 9:39 PM

Key Scenarios Addressed by Salesforce in the Implementation Project:-

Automated Stock Replenishment – Automatically generates purchase orders when inventory levels fall below thresholds.

Supplier Performance Analytics – Tracks supplier reliability, delivery timelines, and quality metrics.

Real-time Inventory Visibility – Provides instant updates on stock levels across all locations.

Expiry Alerts and Notifications – Sends alerts for products nearing expiry to prevent losses.

Integrated Procurement Workflow – Streamlines the entire purchase process from order creation to receipt.

Customizable Reports & Dashboards – Allows stakeholders to generate tailored reports and visual dashboards for decision-making.

Conclusion:

The Medical Inventory Management System implemented in Salesforce streamlines inventory operations, ensures data accuracy, and enhances decision-making. By automating stock management, monitoring supplier performance, tracking product expiry, and providing real-time reports, the system improves efficiency, reduces errors, and supports effective procurement and inventory control. It enhances accuracy, accountability, and operational speed, ensuring that healthcare facilities can deliver uninterrupted and high-quality patient care.