PROJECT TITLE: Medical Inventory Management System

College Name: KG College of Arts and Science
College Code: bru4y

TEAM ID: **NM2025TMID23491**

TEAM MEMBERS:

- **Team Leader Name:** **SRI SANJAYY S** -2322ja53@kgcas.com
- **Team Member 1:** **SUDHARSHAN MYILSAMY**– 2322ja54@kgcas.com
- **Team Member 2:** **SUMITHRA S** – 2322ja55@kgcas.com
- **Team Member 3:** **SURYANARAYANA C P** – 2322ja56@gcas.com

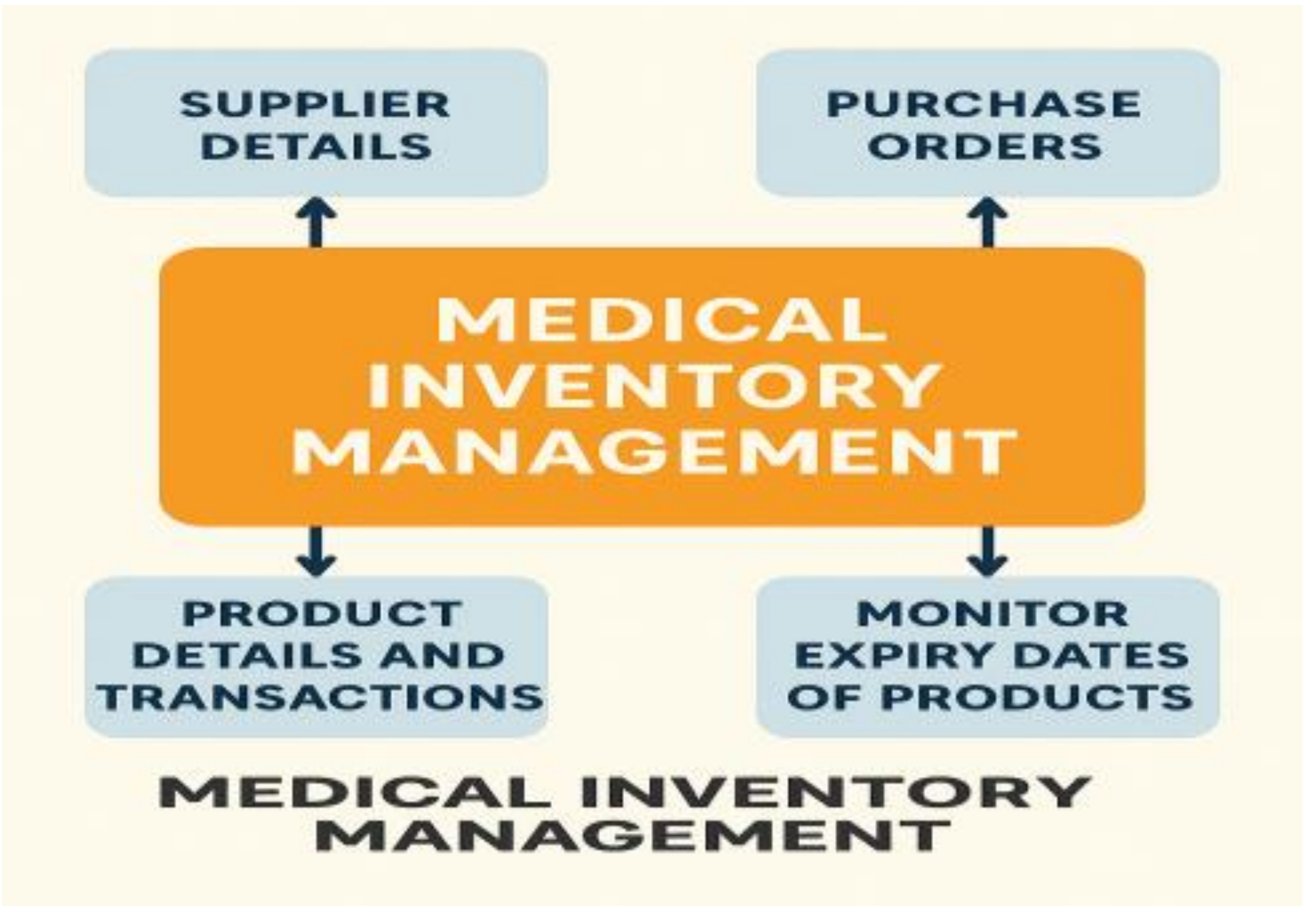
DEMO VIDEO LINK:

NM_Demo video(Medical inventory management).mp4

INTRODUCTION

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.


SUPPLIER DETAILS
PURCHASE ORDERS
MEDICAL INVENTORY MANAGEMENT
PRODUCT DETAILS AND TRANSACTIONS
MONITOR EXPIRY DATES OF PRODUCTS
MEDICAL INVENTORY MANAGEMENT

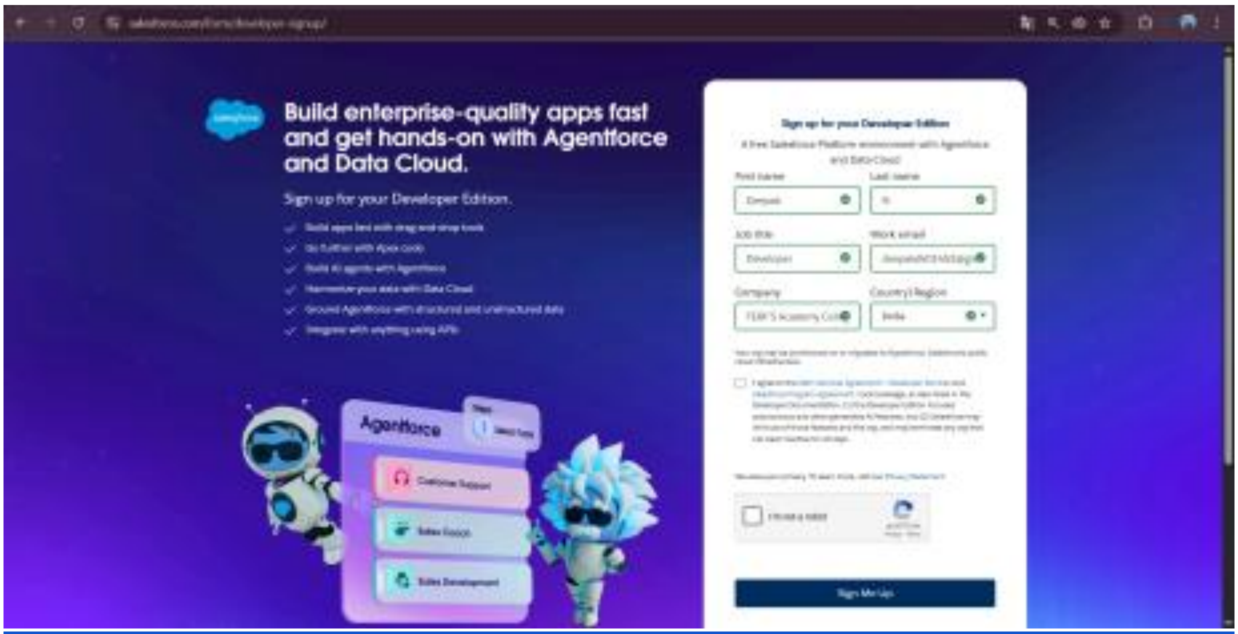
Purpose

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.

DEVELOPMENT PHASE

Creation of Developer Account

- A Salesforce Developer account was created using the signup link: https://www.salesforce.com/form/developer-signup



Build enterprise-quality apps fast and get hands-on with Agentforce and Data Cloud.

Sign up for your Developer Edition.

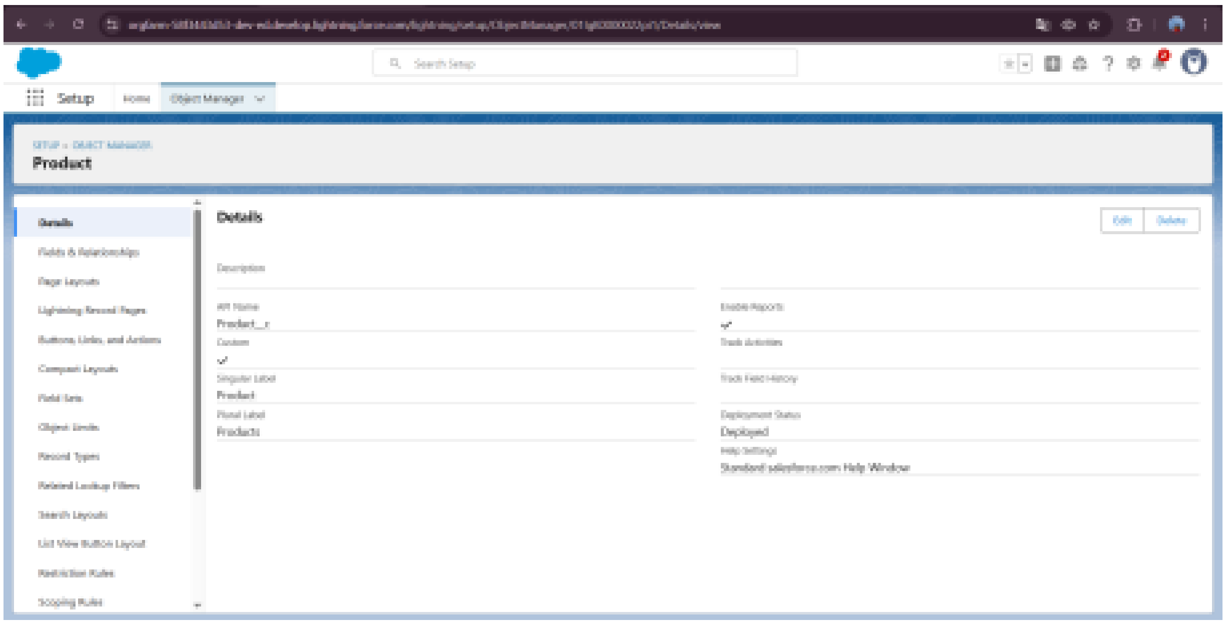
- Build apps fast with drag-and-drop tools
- Go further with Apex code
- Build AI agents with Agentforce
- Harmonize your data with Data Cloud
- Ground Agentforce with structured and unstructured data
- Integrate with anything using APIs

Sign up for your Developer Edition

A free Salesforce Platform environment with Agentforce and Data Cloud

First name: Deepak	Last name: R
Job title: Developer	Work email: [illegible]
Company: [illegible]	Country/Region: India

Creating a Product Object



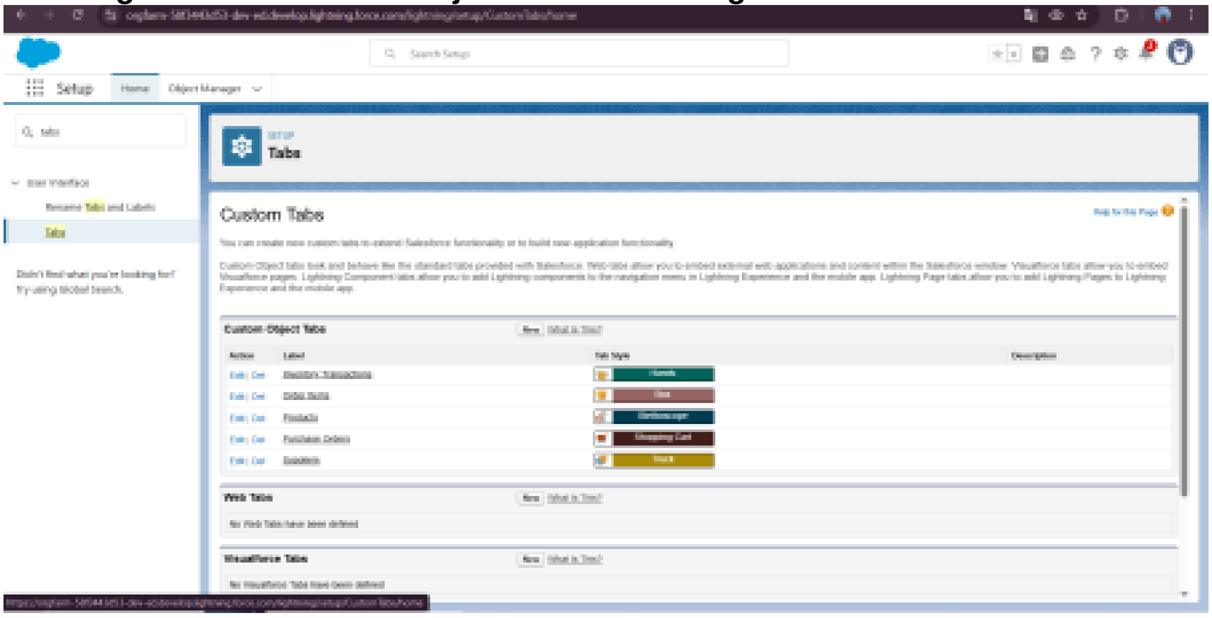
SETUP > OBJECT MANAGER

Product

Details

Description	
API Name: Product__c	Enable Reports: ✓
Custom: ✓	Track Activities
Singular Label: Product	Track Field History
Plural Label: Products	Deployment Status: Deployed
	Help Settings: Standard salesforce.com Help Window

Creating a tab for Product Object and Remaining Tabs



SETUP

Tabs

Custom Tabs

You can create new custom tabs to extend Salesforce functionality or to build new application functionality.

Custom Object tabs look and behave like the standard tabs provided with Salesforce. Web tabs allow you to embed external web applications and content within the Salesforce window. 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	[illegible]	[illegible]	
Edit | Del	Order Items	[illegible]	
Edit | Del	Products	[illegible]	
Edit | Del	Purchase Orders	Shopping Cart	
Edit | Del	Suppliers	[illegible]	

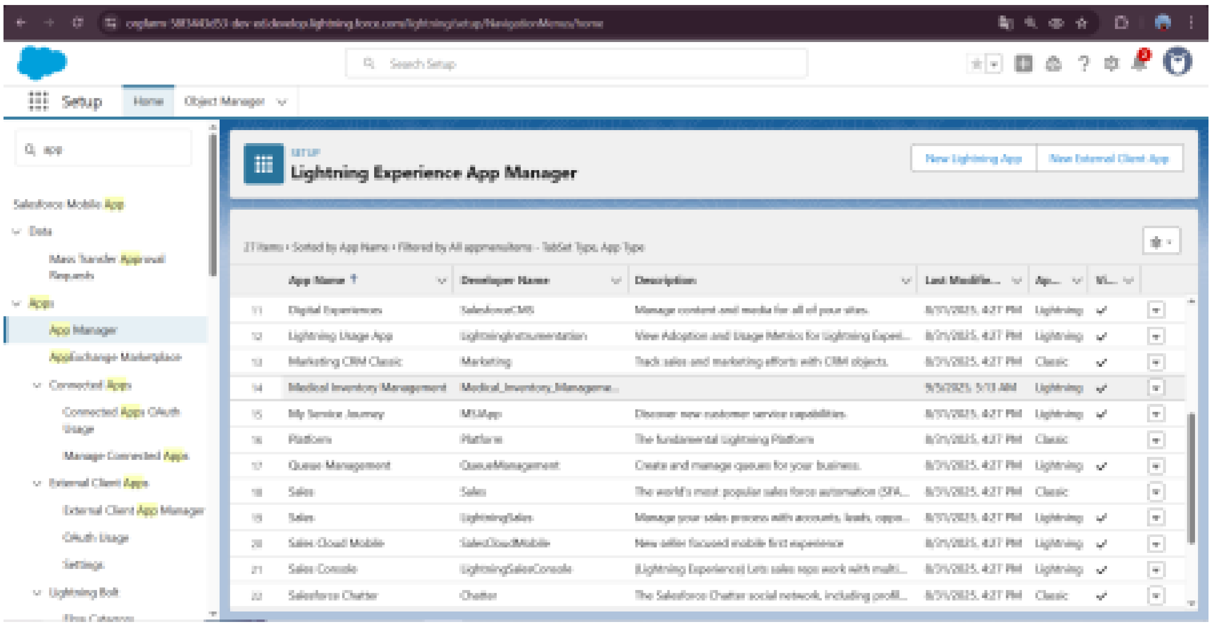
Web Tabs

No Web Tabs have been defined

Visualforce Tabs

No Visualforce Tabs have been defined

Create a Lightning App for Medical Inventory Management

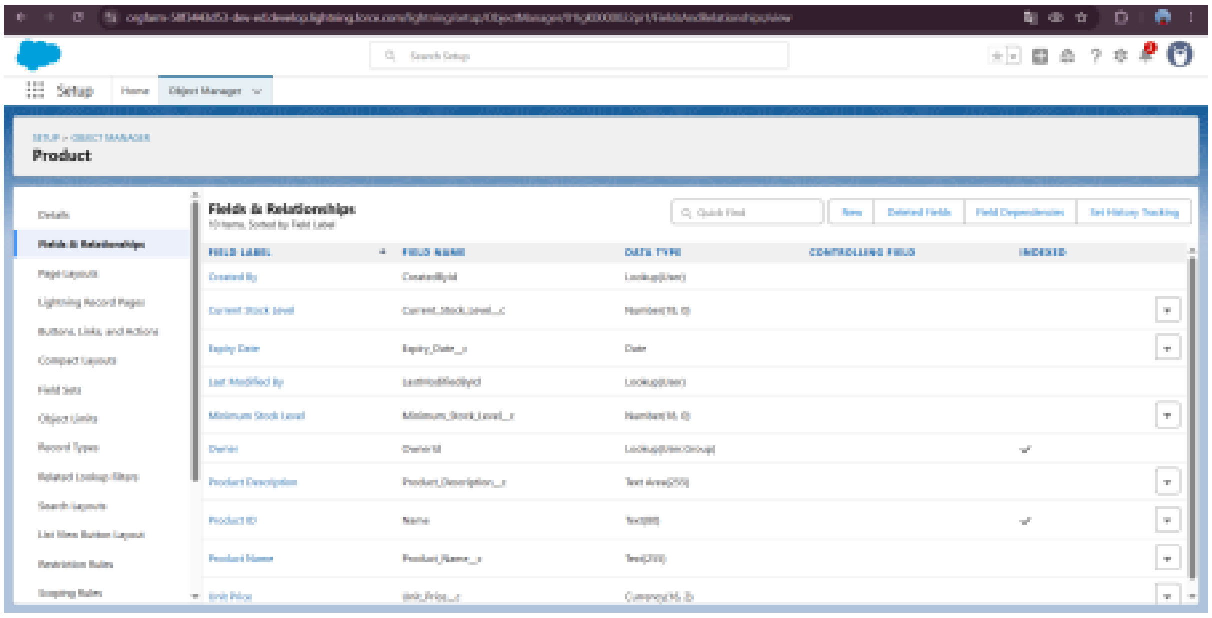


Lightning Experience App Manager

27 items • Sorted by App Name • Filtered by All appmenuitems - TabSet Type, App Type

	App Name	Developer Name	Description	Last Modifie...	Ap...	Vi...
11	Digital Experiences	SalesforceCMS	Manage content and media for all of your sites.	8/31/2025, 4:27 PM	Lightning	✓
12	Lightning Usage App	LightningInstrumentation	View Adoption and Usage Metrics for Lightning Experi...	8/31/2025, 4:27 PM	Lightning	✓
13	Marketing CRM Classic	Marketing	Track sales and marketing efforts with CRM objects.	8/31/2025, 4:27 PM	Classic	✓
14	Medical Inventory Management	Medical_Inventory_Manageme...		9/5/2025, 5:13 AM	Lightning	✓
15	My Service Journey	MSJApp	Discover new customer service capabilities.	8/31/2025, 4:27 PM	Lightning	✓
16	Platform	Platform	The fundamental Lightning Platform	8/31/2025, 4:27 PM	Classic	
17	Queue Management	QueueManagement	Create and manage queues for your business.	8/31/2025, 4:27 PM	Lightning	✓
18	Sales	Sales	The world's most popular sales force automation (SFA...	8/31/2025, 4:27 PM	Classic	
19	Sales	LightningSales	Manage your sales process with accounts, leads, oppo...	8/31/2025, 4:27 PM	Lightning	✓
20	Sales Cloud Mobile	SalesCloudMobile	New seller focused mobile first experience	8/31/2025, 4:27 PM	Lightning	✓
21	Sales Console	LightningSalesConsole	(Lightning Experience) Lets sales reps work with multi...	8/31/2025, 4:27 PM	Lightning	✓
22	Salesforce Chatter	Chatter	The Salesforce Chatter social network, including profil...	8/31/2025, 4:27 PM	Classic	✓

Creating a Text Field in Product Object



SETUP > OBJECT MANAGER

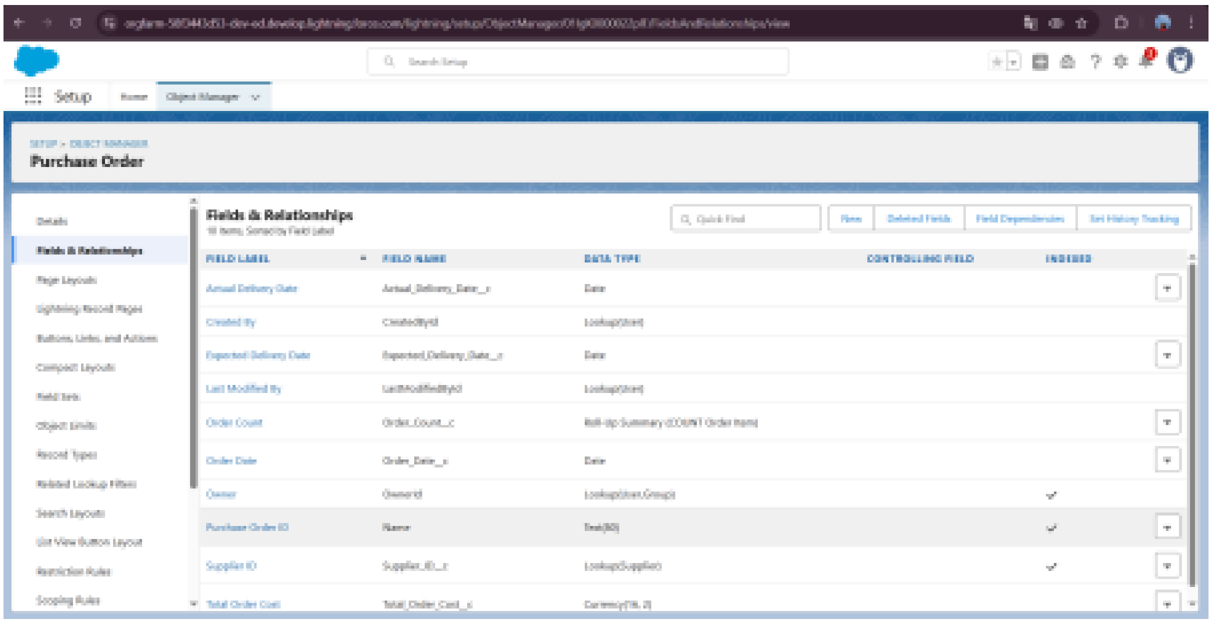
Product

Fields & Relationships

10 Items, Sorted by Field Label

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Created By	CreatedById	Lookup(User)		
Current Stock Level	Current_Stock_Level__c	Number(18, 0)		
Expiry Date	Expiry_Date__c	Date		
Last Modified By	LastModifiedById	Lookup(User)		
Minimum Stock Level	Minimum_Stock_Level__c	Number(18, 0)		
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)		

Creating Lookup Relationship in Purchase Order Object



SETUP > OBJECT MANAGER

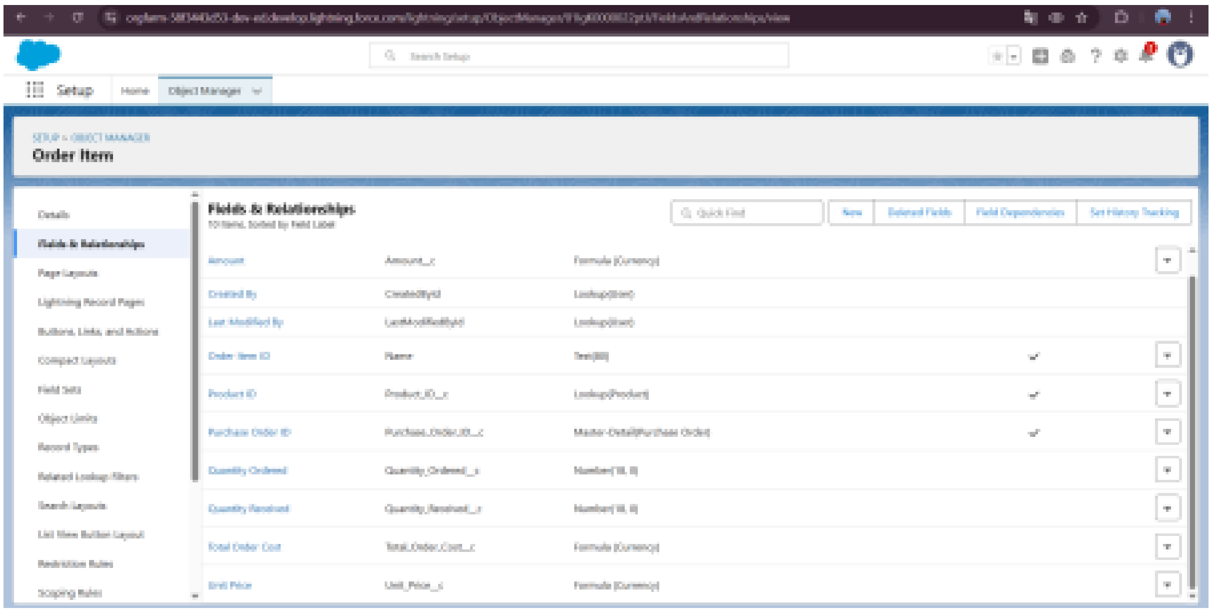
Purchase Order

Fields & Relationships

11 Items, Sorted by Field Label

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Actual Delivery Date	Actual_Delivery_Date__c	Date		
Created By	CreatedById	Lookup(User)		
Expected Delivery Date	Expected_Delivery_Date__c	Date		
Last Modified By	LastModifiedById	Lookup(User)		
Order Count	Order_Count__c	Roll-Up Summary (COUNT Order Item)		
Order Date	Order_Date__c	Date		
Owner	OwnerId	Lookup(User,Group)		✓
Purchase Order ID	Name	Text(80)		✓
Supplier ID	Supplier_ID__c	Lookup(Supplier)		✓
Total Order Cost	Total_Order_Cost__c	Currency(16, 2)		

Creating a Unit Price Formula Field in Order Item object



SETUP > OBJECT MANAGER

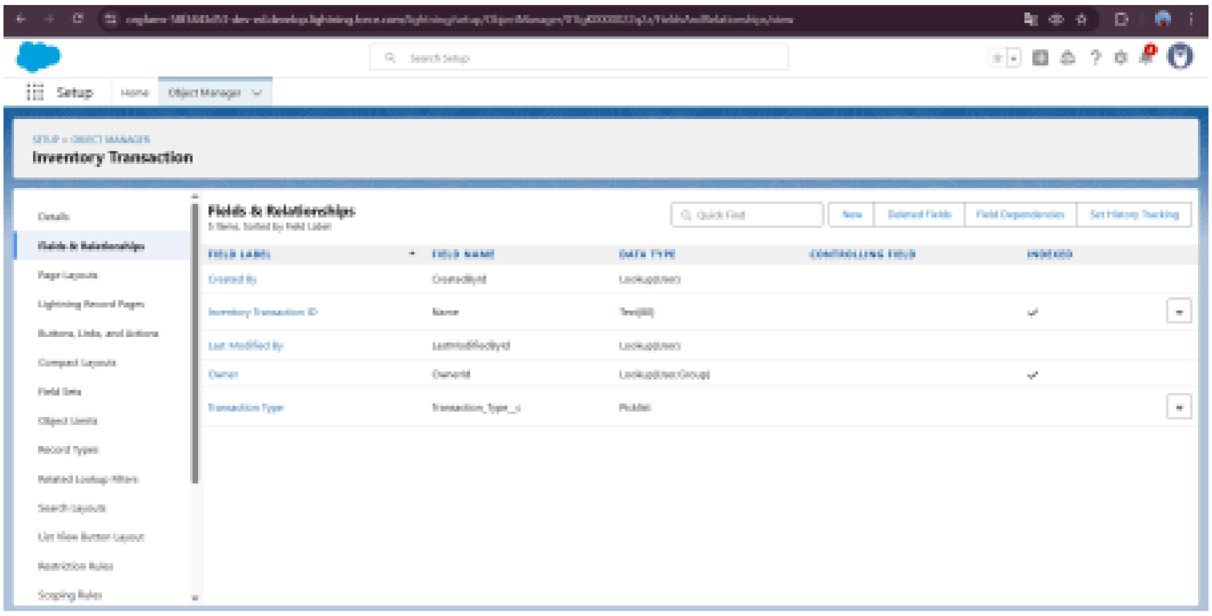
Order Item

Fields & Relationships

10 Items, Sorted by Field Label

Amount	Amount__c	Formula (Currency)	
Created By	CreatedById	Lookup(User)	
Last Modified By	LastModifiedById	Lookup(User)	
Order Item ID	Name	Text(80)	✓
Product ID	Product_ID__c	Lookup(Product)	✓
Purchase Order ID	Purchase_Order_ID__c	Master-Detail(Purchase Order)	✓
Quantity Ordered	Quantity_Ordered__c	Number(18, 0)	
Quantity Received	Quantity_Received__c	Number(18, 0)	
Total Order Cost	Total_Order_Cost__c	Formula (Currency)	
Unit Price	Unit_Price__c	Formula (Currency)	

Creating a Picklist Field in Inventory Transaction Object



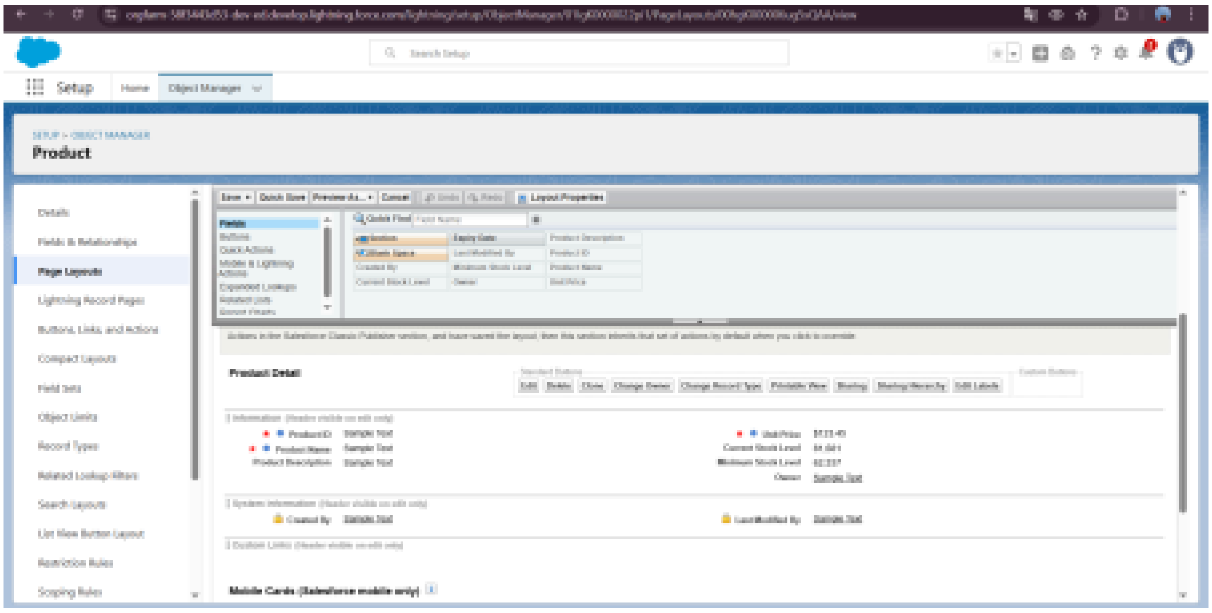
SETUP > OBJECT MANAGER

Inventory Transaction

Fields & Relationships
5 Items, Sorted by Field Label

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Created By	CreatedById	Lookup(User)		
Inventory Transaction ID	Name	Text(80)		✓
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		✓
Transaction Type	Transaction_Type__c	Picklist		

To edit a Page Layout in Product Object



SETUP > OBJECT MANAGER

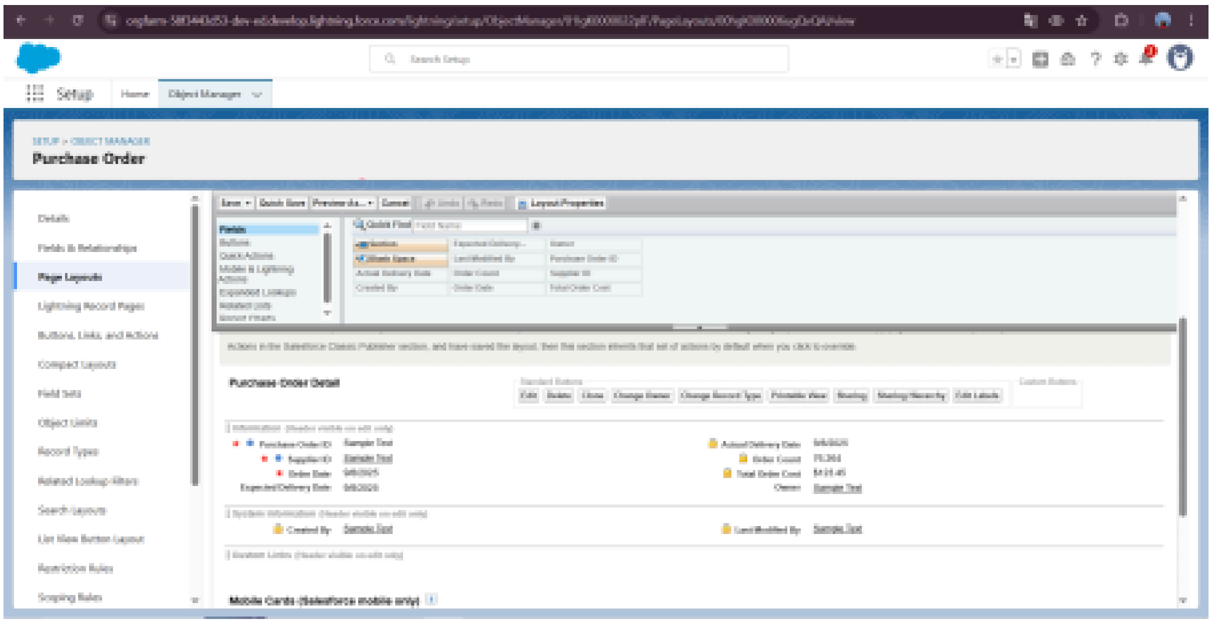
Product

Product Detail

Product ID	Sample Text
Product Name	Sample Text
Product Description	Sample Text
Unit Price	[illegible]
Current Stock Level	[illegible]
Minimum Stock Level	[illegible]
Owner	Sample Text
Created By	Sample Text
Last Modified By	Sample Text

Mobile Cards (Salesforce mobile only)

To edit a Page Layout in Purchase Order Object



SETUP > OBJECT MANAGER

Purchase Order

Details
Fields & Relationships
Page Layouts
Lightning Record Pages
Buttons, Links, and Actions
Compact Layouts
Field Sets
Object Limits
Record Types
Related Lookup Filters
Search Layouts
List View Button Layout
Restriction Rules
Scoping Rules

Save ▾ | Quick Save | Preview As... ▾ | Cancel | Undo | Redo | Layout Properties

Fields
Buttons
Quick Actions
Mobile & Lightning Actions
Expanded Lookups
Related Lists
Report Charts

Quick Find: Field Name

Section	Expected Delivery...	Status
Blank Space	Last Modified By	Purchase Order ID
Actual Delivery Date	Order Count	Supplier ID
Created By	Order Date	Total Order Cost

Purchase Order Detail

Standard Buttons: Edit | Delete | Clone | Change Owner | Change Record Type | Printable View | Sharing | Sharing Hierarchy | Edit Labels

Custom Buttons

Information (Header visible on edit only)

Purchase Order ID	Sample Text	Actual Delivery Date	9/6/2025
Supplier ID	Sample Text	Order Count	73,264
Order Date	9/6/2025	Total Order Cost	$415.45
Expected Delivery Date	9/6/2025	Owner	Sample Text

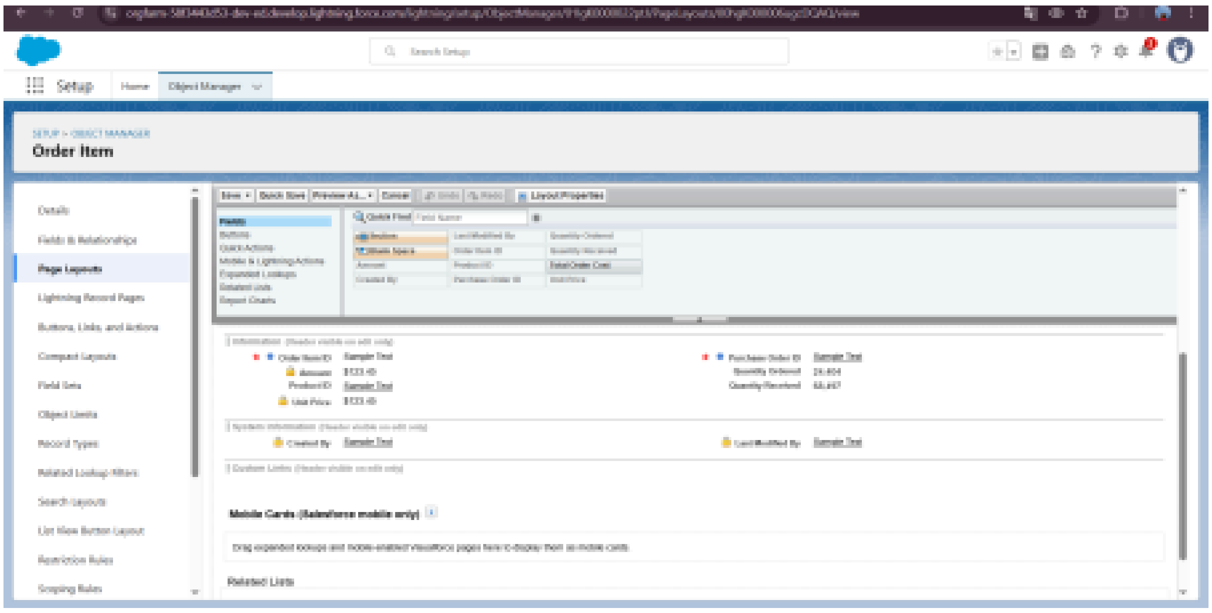
System Information (Header visible on edit only)

Created By	Sample Text	Last Modified By	Sample Text

Custom Links (Header visible on edit only)

Mobile Cards (Salesforce mobile only)

To edit a Page Layout in Order Item Object



SETUP > OBJECT MANAGER

Order Item

Details
Fields & Relationships
Page Layouts
Lightning Record Pages
Buttons, Links, and Actions
Compact Layouts
Field Sets
Object Limits
Record Types
Related Lookup Filters
Search Layouts
List View Button Layout
Restriction Rules
Scoping Rules

Save ▾ | Quick Save | Preview As... ▾ | Cancel | Undo | Redo | Layout Properties

Fields
Buttons
Quick Actions
Mobile & Lightning Actions
Expanded Lookups
Related Lists
Report Charts

Quick Find: Field Name

Section	Last Modified By	Quantity Ordered
Blank Space	Order Item ID	Quantity Received
Amount	Product ID	Total Order Cost
Created By	Purchase Order ID	Unit Price

Information (Header visible on edit only)

Order Item ID	Sample Text	Purchase Order ID	Sample Text
Amount	$123.45	Quantity Ordered	24,404
Product ID	Sample Text	Quantity Received	63,437
Unit Price	$123.45		

System Information (Header visible on edit only)

Created By	Sample Text	Last Modified By	Sample Text

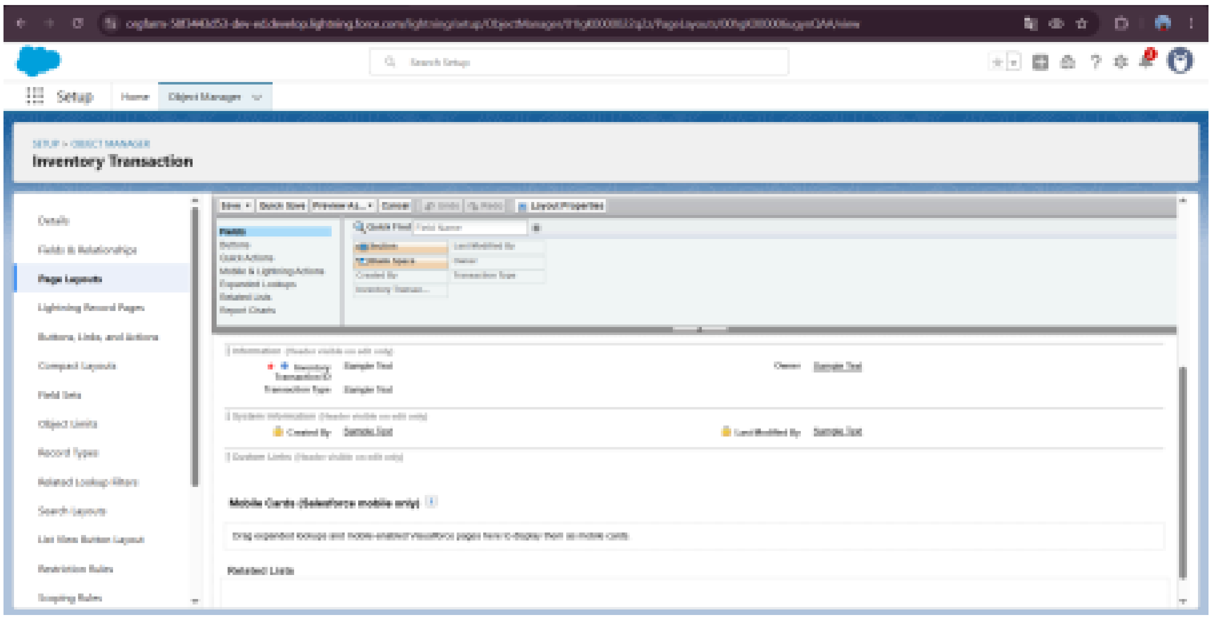
Custom Links (Header visible on edit only)

Mobile Cards (Salesforce mobile only)

Drag expanded lookups and mobile-enabled Visualforce pages here to display them as mobile cards.

Related Lists

To edit a Page Layout in Inventory Transaction Object



SETUP > OBJECT MANAGER

Inventory Transaction

Details
Fields & Relationships
Page Layouts
Lightning Record Pages
Buttons, Links, and Actions
Compact Layouts
Field Sets
Object Limits
Record Types
Related Lookup Filters
Search Layouts
List View Button Layout
Restriction Rules
Scoping Rules

Information (Header visible on edit only)
Inventory Transaction ID: Sample Text
Transaction Type: Sample Text
Owner: Sample Text

System Information (Header visible on edit only)
Created By: Sample Text
Last Modified By: Sample Text

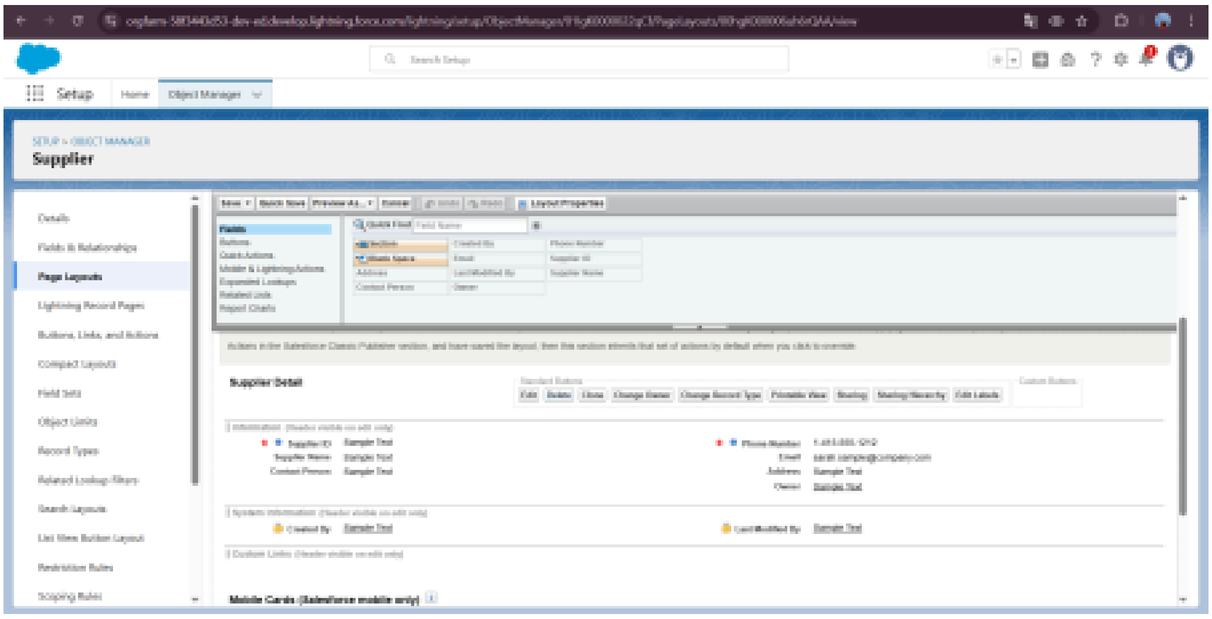
Custom Links (Header visible on edit only)

Mobile Cards (Salesforce mobile only)

Drag expanded lookups and mobile-enabled Visualforce pages here to display them as mobile cards.

Related Lists

To edit a Page Layout in Supplier Object



SETUP > OBJECT MANAGER

Supplier

Supplier Detail

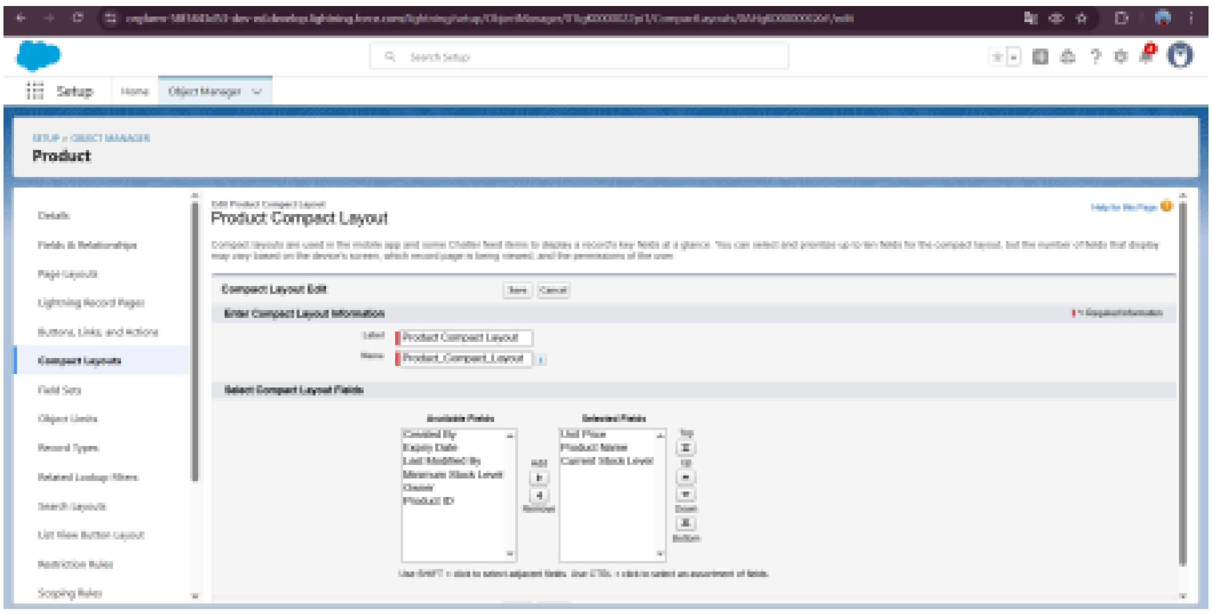
Information (Header visible on edit only)
Supplier ID: Sample Text
Supplier Name: Sample Text
Contact Person: Sample Text
Phone Number: 1-415-555-1212
Email: sarah.sample@company.com
Address: Sample Text
Owner: Sample Text

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)

To create a Compact Layout to a Product Object



SETUP > OBJECT MANAGER

Product

Edit Product Compact Layout

Product Compact Layout

Compact layouts are used in the mobile app and some Chatter feed items to display a record's key fields at a glance. You can select and prioritize up to ten fields for the compact layout, but the number of fields that display may vary based on the device's screen, which record page is being viewed, and the permissions of the user.

Compact Layout Edit

Enter Compact Layout Information

Label: Product Compact Layout
Name: Product_Compact_Layout

Select Compact Layout Fields

Available Fields	Selected Fields
Created By	Unit Price
Expiry Date	Product Name
Last Modified By	Current Stock Level
Minimum Stock Level	
Owner	
Product ID	

Use SHIFT + click to select adjacent fields. Use CTRL + click to select an assortment of fields.

To create a Compact Layout to a Purchase Order Object

To create an Expected Delivery Date Validation rule to a Employee Object

To create an Inventory Manager Profile

To create an Purchase Manager Profile

Create a Purchasing Manager Role.

Create Flow to update the Actual Delivery Date.

Create a Trigger to Calculate total amount on Order Item.

Choose Apex Class: Name it as CalculateTotalAmountHandler

Create a Purchase Orders based on Suppliers(Summary) Report

Create a Complete Purchase Details Report

View Dashboard

RESULTS

● Tabs for Product, Supplier, Purchase Order, Inventory.

● Reports for Expired Products and Supplier Performance. ●

Dashboard showing Stock Levels and Purchase Order Summary. ●

Trigger execution results (auto-calculated total order amount). ●

Validation Rule error messages (when wrong data is entered).

ADVANTAGES & DISADVANTAGES

Advantages

● Accurate tracking of products and expiry dates.

● Easy management of supplier and purchase orders.

● Reduced manual work with automation (flows and triggers).

● Visual dashboards for quick decision-making.

Disadvantages

● Requires Salesforce knowledge for customization.

● Limited offline functionality.

- Integration with external systems (e.g., hospital management software) not implemented yet.

APPENDIX

Create an Apex Trigger:

```
trigger CalculateTotalAmountTrigger on Order_Item__c (after insert, after update, after
delete, after undelete) {
   // Call the handler class to handle the logic
    CalculateTotalAmountHandler.calculateTotal(Trigger.new, Trigger.old,
Trigger.isInsert, Trigger.isUpdate, Trigger.isDelete, Trigger.isUndelete);
}
```

Create Apex Class:

```
public class CalculateTotalAmountHandler {

    // Method to calculate the total amount for Purchase Orders based on related Order
Items
   public static void calculateTotal(List<Order_Item__c> newItems, List<Order_Item__c>
oldItems, Boolean isInsert, Boolean isUpdate, Boolean isDelete, Boolean isUndelete) {

      // Collect Purchase Order IDs affected by changes in Order_Item__c records
      Set<Id> parentIds = new Set<Id>();

      // For insert, update, and undelete scenarios
      if (isInsert || isUpdate || isUndelete) {
         for (Order_Item__c ordItem : newItems) {
            parentIds.add(ordItem.Purchase_Order_Id__c);
         }
      }

      // For update and delete scenarios
      if (isUpdate || isDelete) {
         for (Order_Item__c ordItem : oldItems) {
            parentIds.add(ordItem.Purchase_Order_Id__c);
         }
      }
      // Calculate the total amounts for affected Purchase Orders
      Map<Id, Decimal> purchaseToUpdateMap = new Map<Id, Decimal>();

      if (!parentIds.isEmpty()) {
         // Perform an aggregate query to sum the Amount__c for each Purchase Order
         List<AggregateResult> aggrList = [
            SELECT Purchase_Order_Id__c, SUM(Amount__c) totalAmount
            FROM Order_Item__c
            WHERE Purchase_Order_Id__c IN :parentIds
            GROUP BY Purchase_Order_Id__c
        ];

        // Map the result to Purchase Order IDs
        for (AggregateResult aggr : aggrList) {
            Id purchaseOrderId = (Id)aggr.get('Purchase_Order_Id__c');
            Decimal totalAmount = (Decimal)aggr.get('totalAmount');
            purchaseToUpdateMap.put(purchaseOrderId, totalAmount);
        }

        // Prepare Purchase Order records for update
                                        List<Purchase_Order__c> purchaseToUpdate = new
List<Purchase_Order__c>();
        for (Id purchaseOrderId : purchaseToUpdateMap.keySet()) {
                        Purchase_Order__c purchaseOrder = new Purchase_Order__c(Id =
purchaseOrderId, Total_Order_cost__c =
purchaseToUpdateMap.get(purchaseOrderId)); purchaseToUpdate.add(purchaseOrder);
        }

        // Update Purchase Orders if there are any changes
        if (!purchaseToUpdate.isEmpty()) {
            update purchaseToUpdate;
        }
      }
    }
}
```

Future Enhancements

● Add **barcode scanning** for products to make stock entry faster.

● Implement **email or SMS alerts** for products nearing expiry. ●

Create **mobile-friendly pages** for quick access by staff. ● Add **AI**

predictions for stock demand and reordering.

● Integrate with **external hospital systems** for real-time updates.

CONCLUSION

The Medical Inventory Management System successfully streamlines the operations of managing medical supplies using Salesforce. It ensures better accuracy, reduces errors, and improves efficiency in handling suppliers, purchase orders, and products. With features like validation rules, flows, triggers, reports, and dashboards, the project demonstrates the practical use of Salesforce in real-time business scenarios.