

Medical Inventory Management System

Salesforce Project Documentation Report

TEAM ID: NM2025TMID08962

Team Size: 3

Team Leader: Yamunadevi R

Team Members: Ebinezer M , Durai Pandi K

GitHub Repository: <https://github.com/Ebinez004/Medical-Inventory-Management-System-Salesforce-Project-Documentation-Report>

Executive Summary

- The Medical Inventory Management System is a custom Salesforce application built to streamline inventory operations in a medical facility. It enables efficient tracking of suppliers, purchase orders, products, and transactions while ensuring data accuracy through validations, automations (Flows & Triggers), and secure access controls. Key features include real-time roll-up summaries for order totals, automated delivery date calculations, and insightful reports/dashboards for decision-making.
- This project demonstrates end-to-end Salesforce development, from object creation to analytics, using declarative tools (e.g., Flows, Validation Rules) and code (Apex Triggers). The system supports role-based access, preventing expired stock usage and optimizing procurement.
- Project Status: 100% Complete (All 17 Milestones Achieved).

CHAPTER 1 – INTRODUCTION

1.1 Overview

The Medical Inventory Management System is designed to simplify and automate the management of medicines and medical supplies using Salesforce. It helps maintain accurate records of suppliers, products, purchase orders, order items, and inventory transactions, all interconnected for seamless data flow.

The system also provides real-time insights into stock levels, expiry tracking, and supplier performance using Salesforce reports and dashboards.

1.2 Objectives

- To automate and digitize medical inventory operations.
- To create and interconnect multiple Salesforce objects for managing suppliers, products, and transactions.
- To use flows, triggers, and validation rules to automate processes and improve accuracy.
- To generate meaningful reports and dashboards for better decision-making.

1.3 Scope

The project covers:

- Creation of supplier and product databases.
- Purchase order and transaction management.
- Stock level and expiry date monitoring.
- Role-based access for different users.
- Analytical reporting and visualization.

1.4 Existing vs Proposed System

Existing System	Proposed Salesforce System
Manual entry and tracking	Automated record management
High chance of errors	Validation and automation reduce
Difficult report generation	One-click real-time dashboards
No expiry alerts	Automatic expiry tracking
No supplier analysis	Comprehensive supplier

CHAPTER 2 – PROJECT IDEATION (Phase 1)

2.1 Problem Identification

Traditional medical inventory systems face problems like missing data, manual errors, expired stock usage, and delayed ordering. There is no centralized platform to manage all operations efficiently.

2.2 Idea Description

The idea is to build a centralized Salesforce-based Medical Inventory Management System that automates stock management, order tracking, and supplier coordination through custom objects, triggers, and reports.

2.3 Feasibility

Salesforce provides built-in CRM, automation, and reporting features without additional infrastructure. It's cost-effective, scalable, and ideal for real-time medical data management.

2.4 Innovation

- Record-triggered flows for automatic delivery updates.
- Trigger-based total cost calculation.
- Dashboards visualizing supplier performance and purchase trends.

CHAPTER 3 – REQUIREMENT ANALYSIS (Phase 2)

3.1 Functional Requirements

- Create and manage suppliers, products, and purchase orders.
- Automate field updates using validation rules and triggers.
- Generate reports for supplier-wise purchases and product tracking.
- Create dashboards summarizing overall inventory performance.

3.2 Non-Functional Requirements

- Secure Salesforce cloud data environment.
- User-friendly interface.
- Highly scalable for future data and new modules.

3.3 User Roles

- System Administrator: Has full access to manage app setup and configuration.
- Purchase Manager: Manages suppliers and purchase orders.
- Inventory Manager: Tracks stock, expiry, and product flow.

3.4 Data Flow

Objects:

Supplier → Purchase Order → Order Item → Product → Inventory Transaction

Relationships include Lookup and Master-Detail connections to ensure linked data tracking.

CHAPTER 4 – PROJECT DESIGN (Phase 3)

4.1 Schema Builder

Visual representation of all custom objects and their relationships (Supplier, Product, Purchase Order, Order Item, Inventory Transaction).

4.2 Process Flow

1. Supplier and product data created.
2. Purchase orders generated.
3. Order items and transactions updated.
4. Expiry and stock level tracked automatically.
5. Reports and dashboards summarize results.

4.3 UI Design

A Lightning App with tabs for each object (Products, Orders, Suppliers, etc.), providing a clean and accessible layout.

CHAPTER 5 – PROJECT DEVELOPMENT (Phase 4)

5.1 Milestones Summary

The development process followed 17 milestones covering setup to conclusion:

- M1: Salesforce Account Creation
- M2: Object Creation
- M3: Tabs
- M4: Lightning App
- M5: Fields
- M6: Page Layouts
- M7: Compact Layouts
- M8: Validation Rules
- M9: Profiles
- M10: Roles
- M11: Users
- M12: Permission Sets
- M13: Flows
- M14: Triggers
- M15: Reports
- M16: Dashboards
- M17: Conclusion

5.2 Key Apex Code (Handler and Trigger)

A trigger was implemented to calculate total order cost in the Purchase Order object whenever Order Items are inserted or updated.

5.3 Flows

Flows were created to auto-update Actual Delivery Date and ensure dynamic updates across related records.

5.4 Reports

Two reports were designed:

1. Purchase Orders based on Suppliers
2. Complete Purchase Details Report

CHAPTER 6 – PROJECT DOCUMENTATION (Phase 5)

6.1 Execution Steps

- Created Salesforce developer account.
- Built custom objects, fields, and relationships.
- Implemented validation rules, triggers, and flows.
- Designed reports and dashboards for analysis.
- Uploaded deliverables to GitHub.

6.2 GitHub Repository

Link: <https://github.com/Ebinez004/Medical-Inventory-Management-System-Salesforce-Project-Documentation-Report>

6.3 Fixes & Adjustments

Issue	Fix Implemented
Report showing no results	Added sample data for Purchase Orders and Products
Validation rule errors	Corrected API references and syntax
Layout not showing all fields	Rearranged and saved Page Layouts properly

CHAPTER 7 – PROJECT DEMONSTRATION & RESULTS

7.1 Demo Process

- Created and linked records for Supplier, Product, and Purchase Order.
- Added data for Order Items and Inventory Transactions.
- Flows and triggers executed automatically to update related fields.
- Reports and dashboards reflected real-time data.

7.2 Outputs

- Real-time update of order costs and delivery dates.
- Reports generated successfully with summarized supplier information.
- Dashboards visualized purchase costs and supplier trends effectively.

CHAPTER 8 – CONCLUSION & FUTURE ENHANCEMENT

8.1 Conclusion

The Medical Inventory Management System successfully demonstrates how Salesforce can automate and optimize medical inventory operations. It ensures efficient data management, timely ordering, and better visibility into stock and supplier activities.

8.2 Limitations

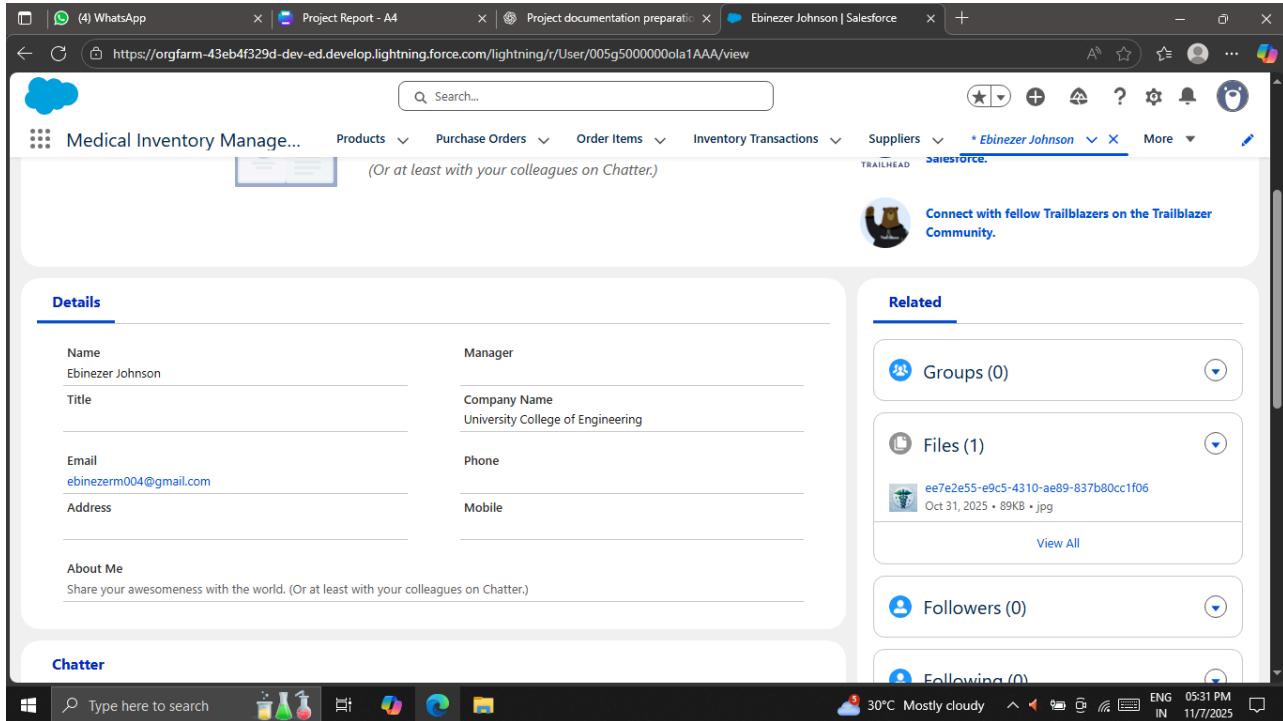
- The system is limited to Salesforce's environment.
- Requires internet access for all operations.

8.3 Future Enhancements

- Integrate real-time expiry notifications via email.
- Add mobile app version for supplier tracking.
- Implement AI-based forecasting for demand and stock prediction.

APPENDIX – SCREENSHOTS

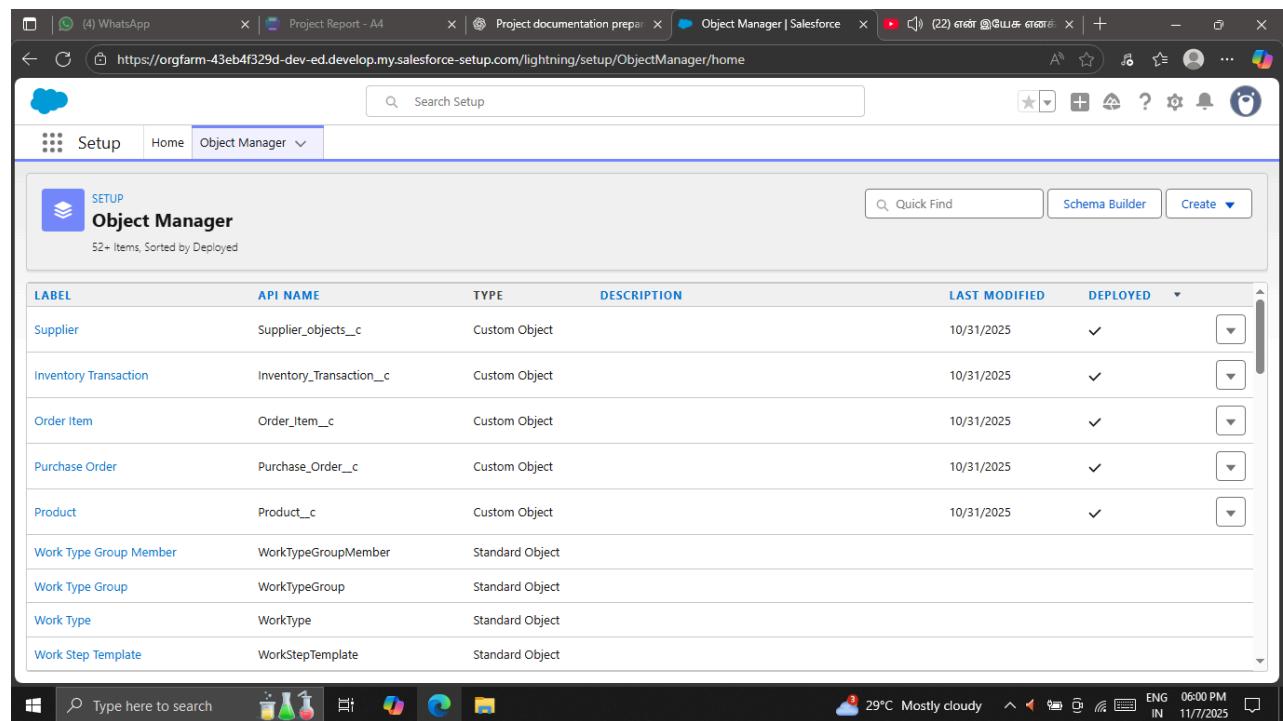
1) Salesforce Developer Account Setup



The screenshot shows a Salesforce developer account setup page. The top navigation bar includes tabs for WhatsApp, Project Report - A4, Project documentation preparation, Ebinezer Johnson | Salesforce, and more. The main content area displays a user profile for Ebinezer Johnson, including fields for Name, Title, Email, Address, Manager, Company Name, Phone, and Mobile. Below this is a "About Me" section with a placeholder for sharing awesomeness. To the right, a "Related" sidebar lists Groups (0), Files (1) containing a file named "ee7e2e55-e9c5-4310-ae89-837b80cc1f06" (Oct 31, 2025, 89KB), Followers (0), and Following (0). The bottom of the screen shows a Windows taskbar with various icons and system status.

Salesforce Developer Account successfully created and activated.

2) Custom Objects Created



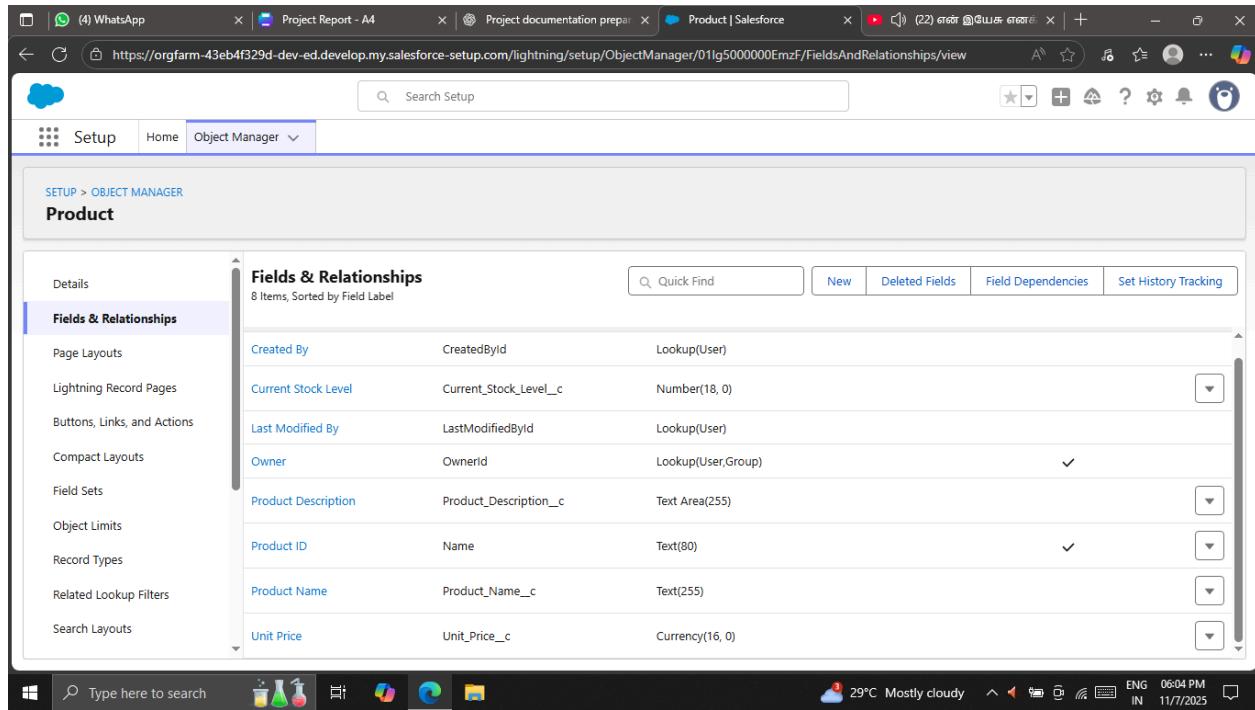
The screenshot shows the Salesforce Object Manager. The top navigation bar includes tabs for Setup, Home, and Object Manager. The main content area displays a table of custom objects, each with a Label, API Name, Type, Description, Last Modified, and Deployed status. The objects listed are Supplier, Inventory Transaction, Order Item, Purchase Order, Product, Work Type Group Member, Work Type Group, Work Type, and Work Step Template. All objects were last modified on 10/31/2025 and are deployed. The bottom of the screen shows a Windows taskbar with various icons and system status.

LABEL	API NAME	TYPE	DESCRIPTION	LAST MODIFIED	DEPLOYED
Supplier	Supplier_objects_c	Custom Object		10/31/2025	✓
Inventory Transaction	Inventory_Transaction_c	Custom Object		10/31/2025	✓
Order Item	Order_Item_c	Custom Object		10/31/2025	✓
Purchase Order	Purchase_Order_c	Custom Object		10/31/2025	✓
Product	Product_c	Custom Object		10/31/2025	✓
Work Type Group Member	WorkTypeGroupMember	Standard Object			
Work Type Group	WorkTypeGroup	Standard Object			
Work Type	WorkType	Standard Object			
Work Step Template	WorkStepTemplate	Standard Object			

Custom Objects created for Medical Inventory Management System.

APPENDIX – SCREENSHOTS

3) Product Object – Fields & Relationships

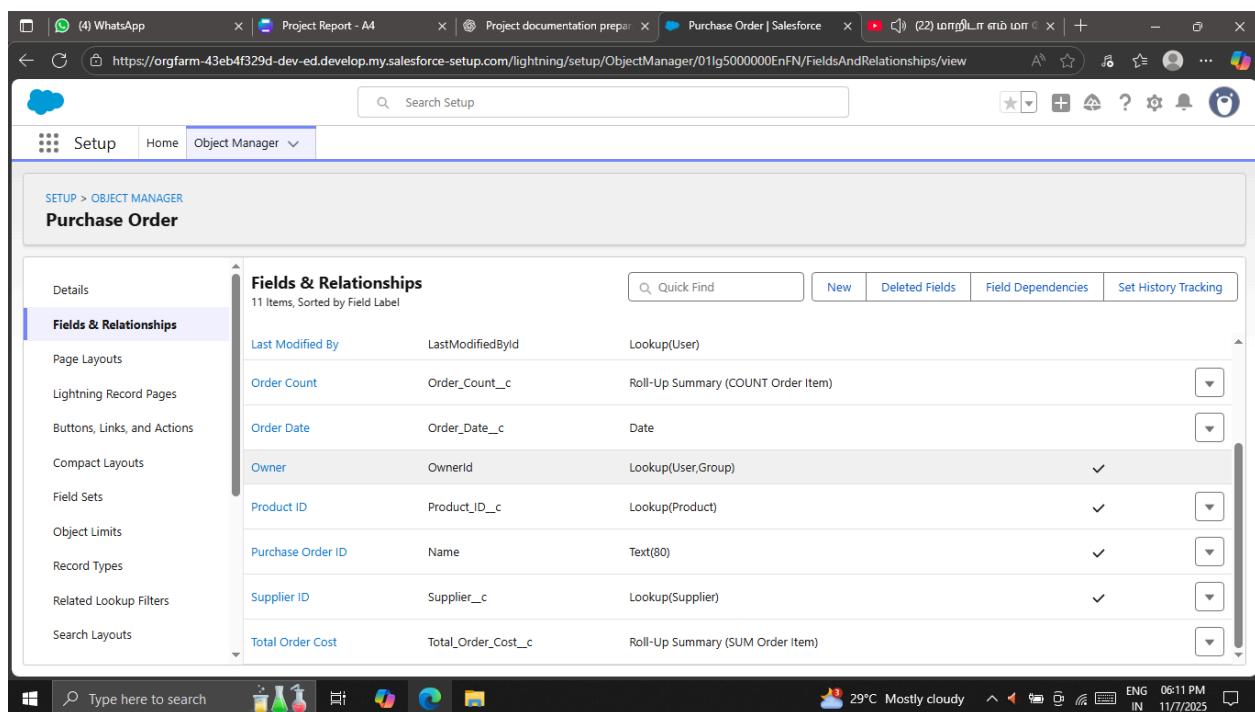


The screenshot shows the Salesforce Object Manager interface for the 'Product' object. The left sidebar lists various setup options like Page Layouts, Lightning Record Pages, and Field Sets. The main area is titled 'Fields & Relationships' and displays 8 items, sorted by Field Label. The fields listed are:

Field Name	Field Type	Description
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, 0)

Fields and Relationships defined for Product Object.

4) Purchase Order Object – Lookup & Roll-Up Fields



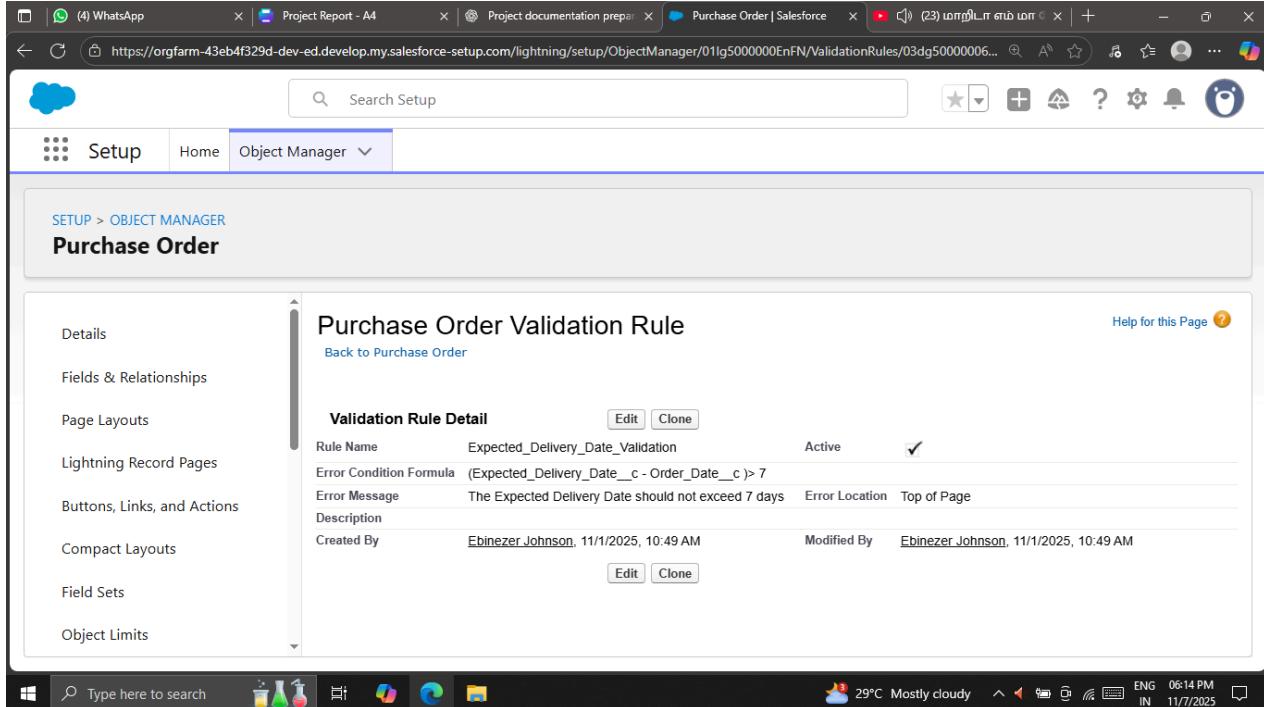
The screenshot shows the Salesforce Object Manager interface for the 'Purchase Order' object. The left sidebar lists various setup options. The main area is titled 'Fields & Relationships' and displays 11 items, sorted by Field Label. The fields listed are:

Field Name	Field Type	Description
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)
Product ID	Product_ID_c	Lookup(Product)
Purchase Order ID	Name	Text(80)
Supplier ID	Supplier_c	Lookup(Supplier)
Total Order Cost	Total_Order_Cost_c	Roll-Up Summary (SUM Order Item)

Lookup and Roll-Up Summary fields configured in Purchase Order Object.

APPENDIX – SCREENSHOTS

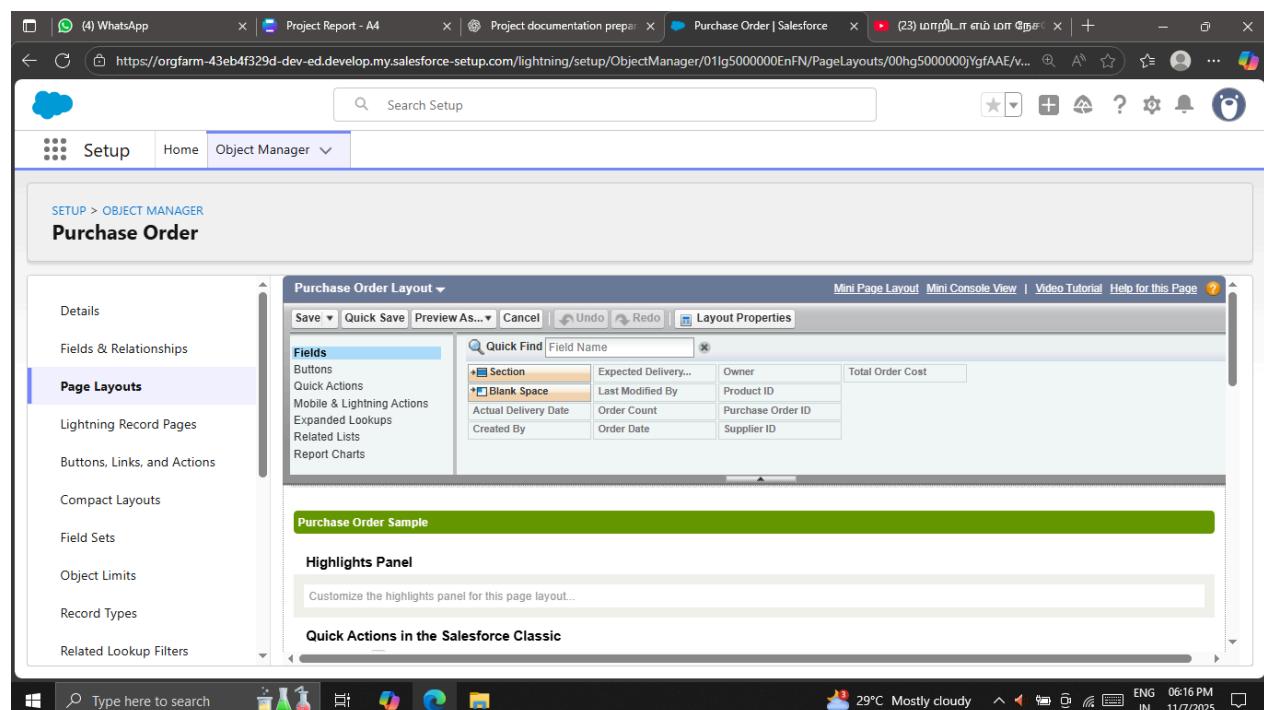
5) Validation Rule



The screenshot shows the 'Purchase Order Validation Rule' configuration in the Salesforce Object Manager. The validation rule is named 'Expected_Delivery_Date_Validation'. The error condition formula is '(Expected_Delivery_Date__c - Order_Date__c) > 7'. The error message is 'The Expected Delivery Date should not exceed 7 days'. The rule is active and was created by Ebinezer Johnson on 11/1/2025 at 10:49 AM.

Validation Rule to restrict Expected Delivery Date beyond 7 days.

6) Page Layout Editing

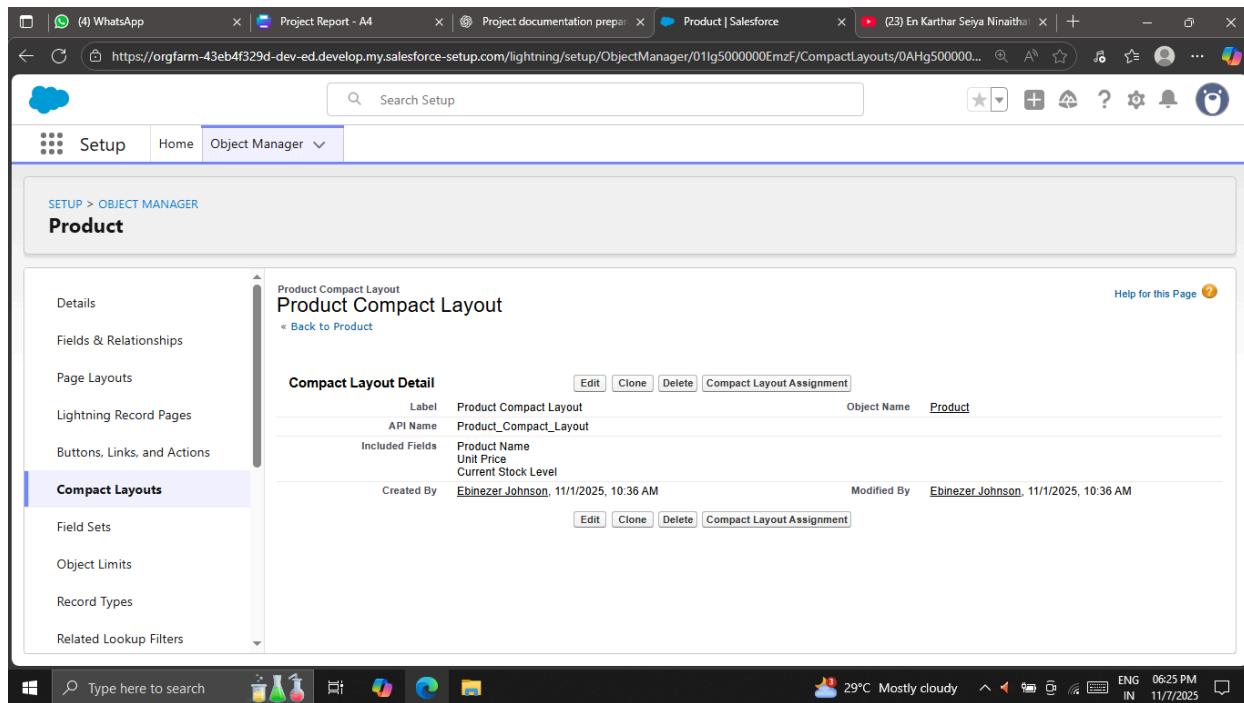


The screenshot shows the 'Purchase Order Layout' configuration in the Salesforce Object Manager. A new page layout is being created, featuring fields for Expected Delivery Date, Last Modified By, Product ID, Order Count, Purchase Order ID, and Order Date. The layout also includes sections for 'Purchase Order Sample' and 'Highlights Panel'.

Customized Page Layout for Purchase Order Object with required fields.

APPENDIX – SCREENSHOTS

7) Compact Layout



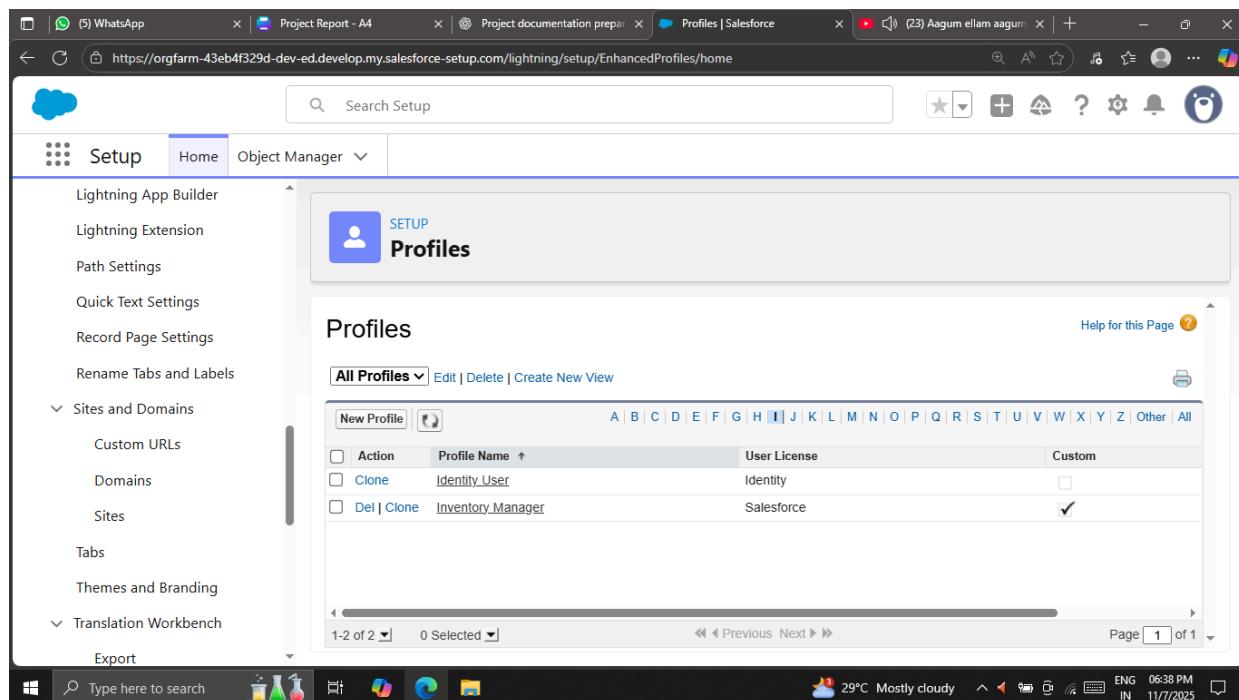
The screenshot shows the Salesforce Setup interface for creating a new Compact Layout. The left sidebar is titled 'Object Manager' and has a 'Compact Layouts' section selected. The main content area is titled 'Product Compact Layout' and displays the 'Compact Layout Detail' for 'Product Compact Layout'. The details include:

Label	Product Compact Layout	Object Name	Product
API Name	Product_Compact_Layout		
Included Fields	Product Name, Unit Price, Current Stock Level		
Created By	Ebenezer Johnson, 11/1/2025, 10:36 AM	Modified By	Ebenezer Johnson, 11/1/2025, 10:36 AM

The status bar at the bottom shows it's 29°C, Mostly cloudy, and the date is 11/7/2025.

Compact Layout created to display key fields in record highlights panel.

8) Profiles and Roles



The screenshot shows the Salesforce Setup interface for managing Profiles. The left sidebar has a 'Sites and Domains' section expanded, showing 'Custom URLs', 'Domains', and 'Sites'. The main content area is titled 'Profiles' and shows a table of existing profiles:

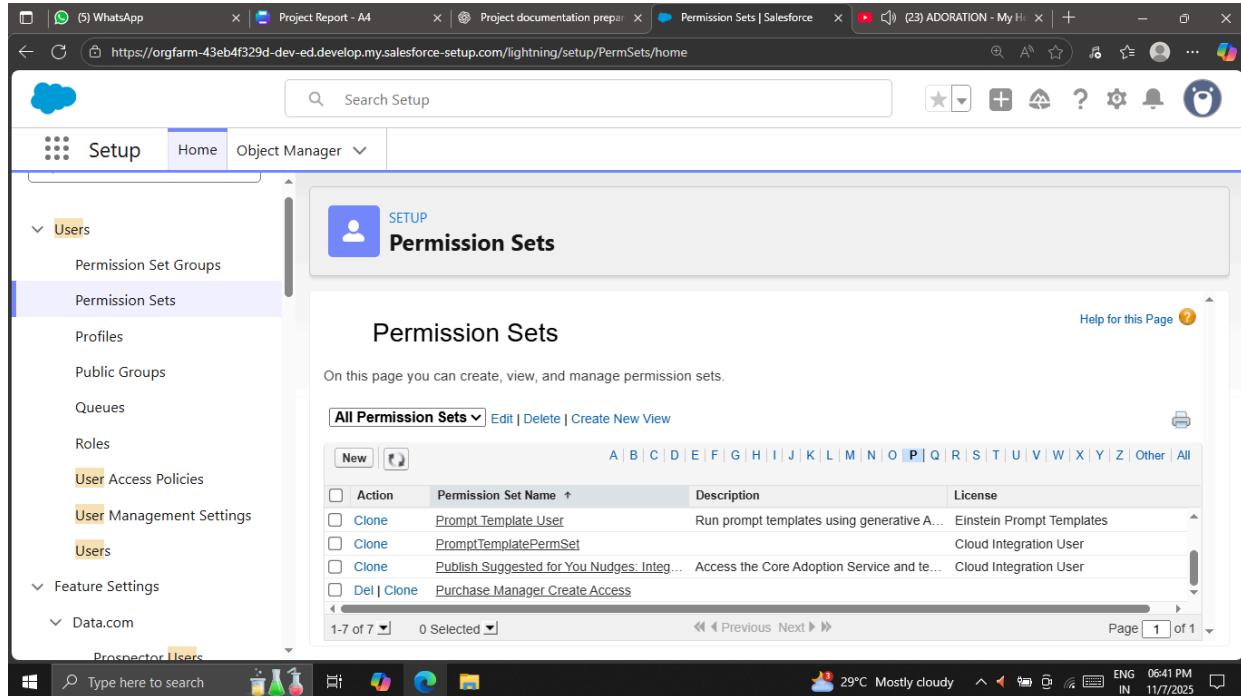
Action	Profile Name	User License	Custom
<input type="checkbox"/>	Identity_User	Identity	<input type="checkbox"/>
<input type="checkbox"/>	Inventory_Manager	Salesforce	<input checked="" type="checkbox"/>

The status bar at the bottom shows it's 29°C, Mostly cloudy, and the date is 11/7/2025.

Custom Profiles and Roles created for user access management.

APPENDIX – SCREENSHOTS

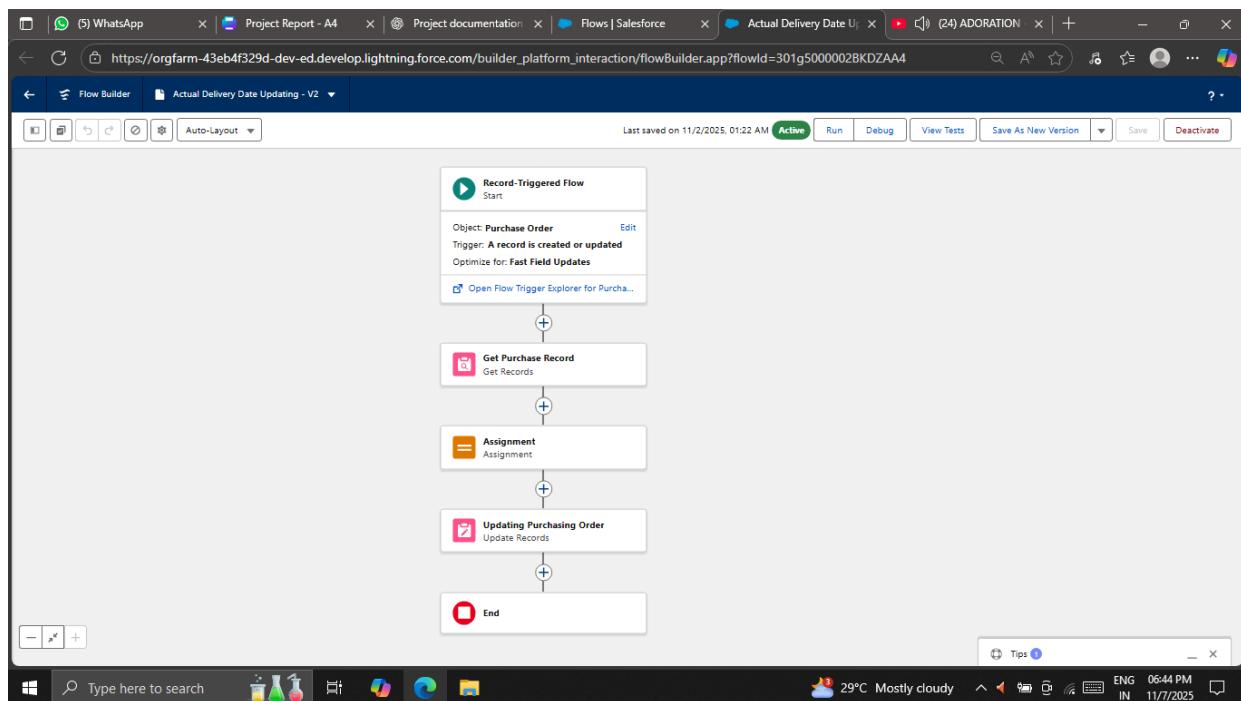
9) Permission Set



The screenshot shows the Salesforce Setup interface under the 'Permission Sets' section. The left sidebar includes options like 'Users', 'Profiles', 'Public Groups', 'Queues', 'Roles', 'User Access Policies', 'User Management Settings', and 'Users'. The main area displays a table of permission sets with columns for Action, Permission Set Name, Description, and License. The table lists several entries, including 'Prompt Template User' (Einstein Prompt Templates), 'PromptTemplatePermSet' (Cloud Integration User), and 'Purchase Manager Create Access' (Cloud Integration User). A navigation bar at the bottom indicates 1-7 of 7 results.

Permission Set assigned for creating and accessing Order Item records.

10) Flow Builder

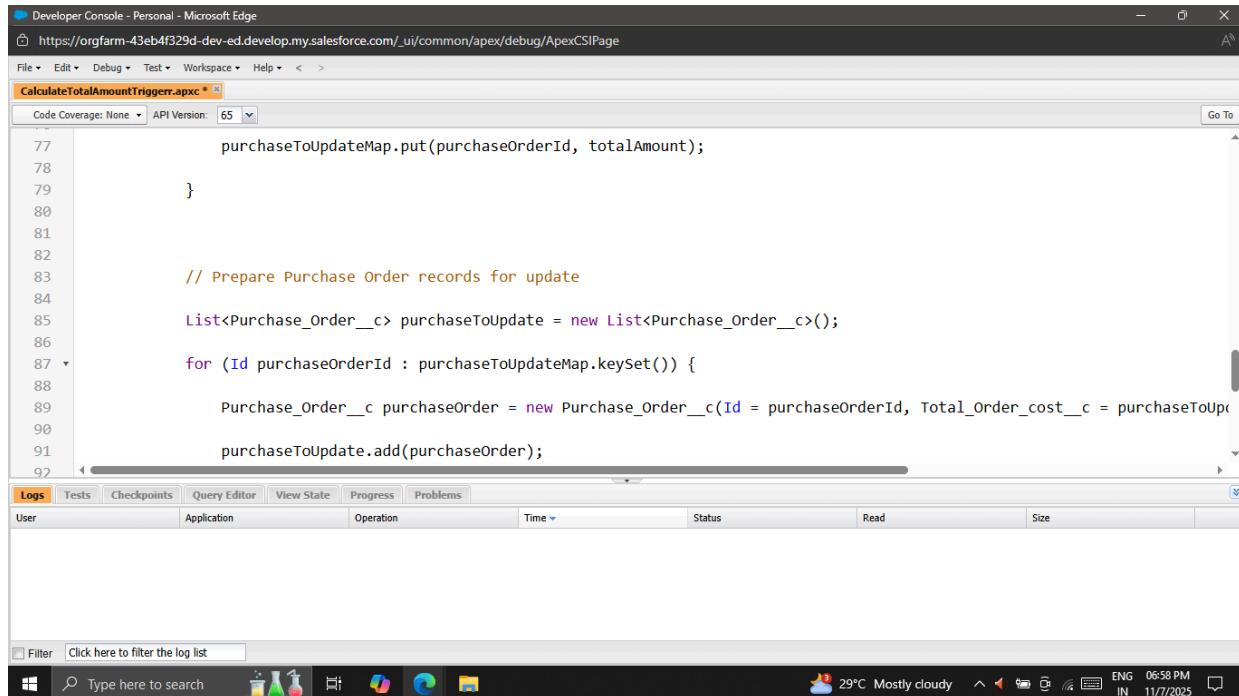


The screenshot shows the Salesforce Flow Builder interface for a 'Record-Triggered Flow'. The flow starts with a 'Record-Triggered Flow Start' step for 'Purchase Order' objects triggered by a record creation or update. This is followed by a 'Get Purchase Record' step, an 'Assignment' step, and an 'Updating Purchasing Order' step. The flow concludes with an 'End' step. The status bar at the bottom indicates the flow was last saved on 11/2/2025 at 01:22 AM and is currently active.

Record-Triggered Flow created to auto-update Actual Delivery Date in Purchase Orders.

APPENDIX – SCREENSHOTS

11) Apex Trigger and Handler Class



```

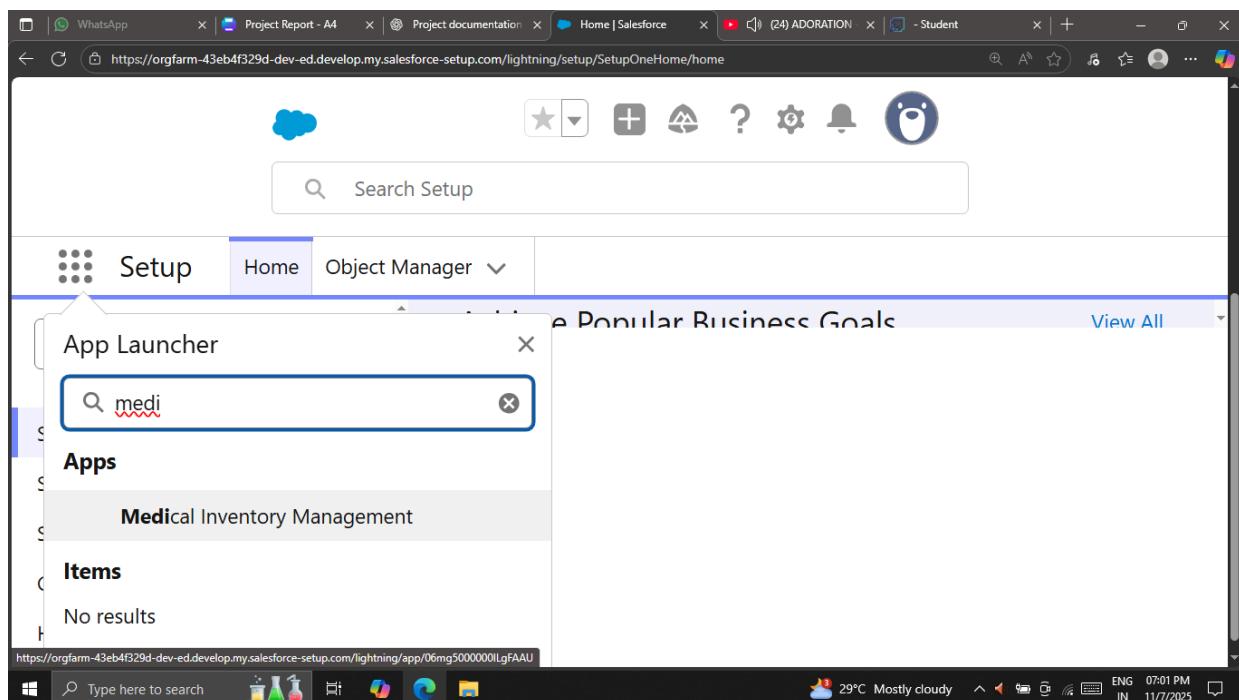
77     purchaseToUpdateMap.put(purchaseOrderId, totalAmount);
78 }
79
80
81
82
83     // Prepare Purchase Order records for update
84
85     List<Purchase_Order__c> purchaseToUpdate = new List<Purchase_Order__c>();
86
87     for (Id purchaseOrderId : purchaseToUpdateMap.keySet()) {
88
89         Purchase_Order__c purchaseOrder = new Purchase_Order__c(Id = purchaseOrderId, Total_Order_cost__c = purchaseToUpdateMap.get(purchaseOrderId));
90
91         purchaseToUpdate.add(purchaseOrder);
92

```

The screenshot shows the Salesforce Developer Console interface. The code editor displays an Apex trigger named "CalculateTotalAmountTrigger.apxc". The trigger logic involves a map to store purchase IDs and their total amounts, then iterates through the map to create new Purchase_Order__c records with the calculated total amount. The developer console also includes tabs for Logs, Tests, Checkpoints, Query Editor, View State, Progress, and Problems.

Apex Trigger and Handler implemented to calculate Total Order Cost.

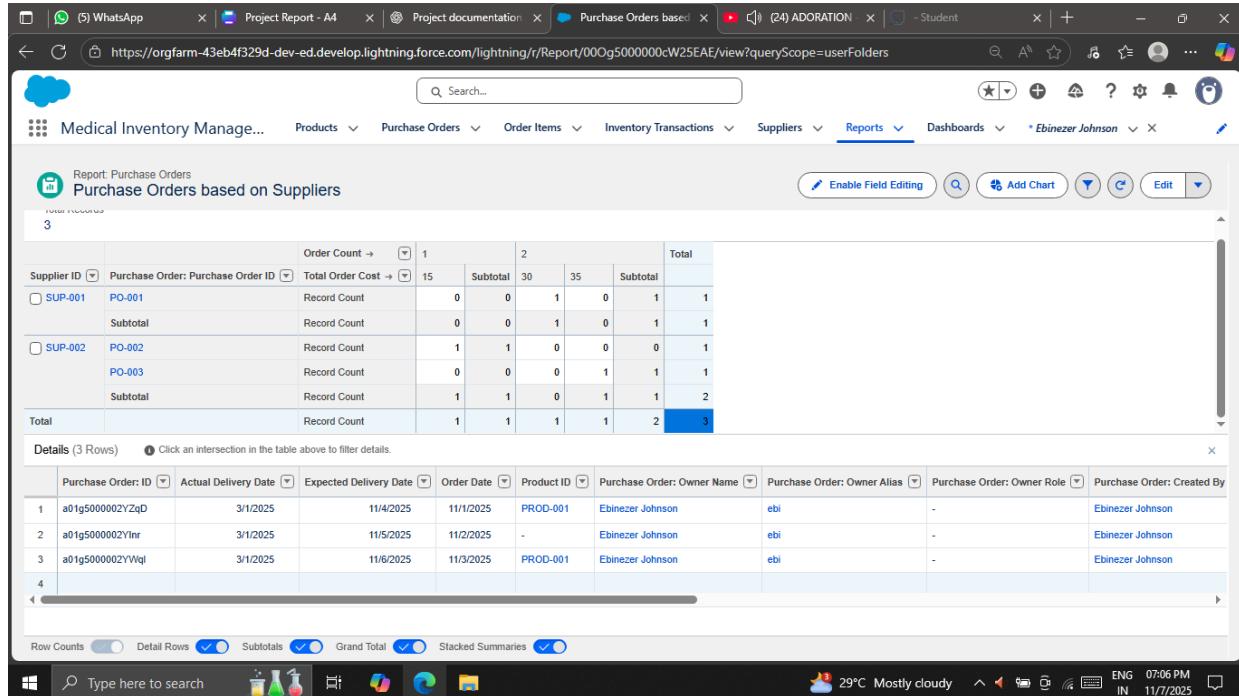
12) Lightning App Creation



Lightning App created to manage all medical inventory modules.

APPENDIX – SCREENSHOTS

13) Reports



The screenshot shows a report titled "Purchase Orders based on Suppliers". The report displays a summary table and a detailed table below it.

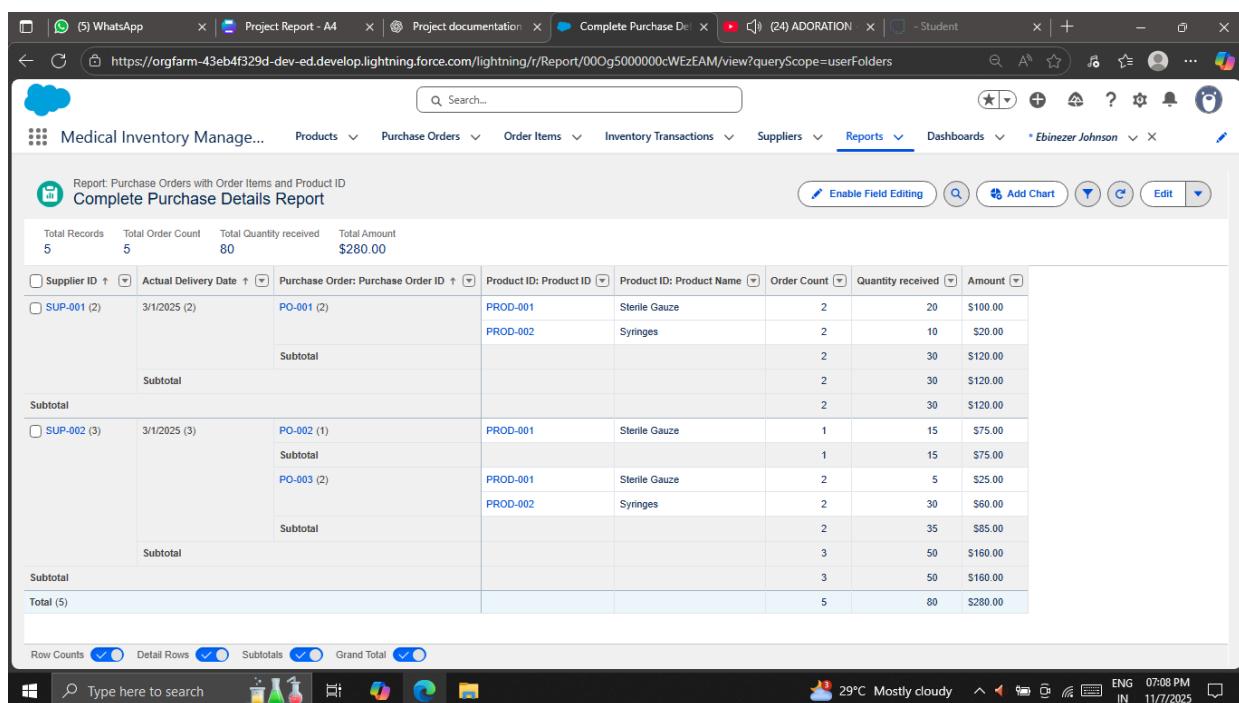
Summary Table:

Supplier ID	Purchase Order: Purchase Order ID	Order Count	Total		
			Total Order Cost	Subtotal	35
SUP-001	PO-001	15	0	1	0
	Subtotal		0	1	0
SUP-002	PO-002	1	1	0	0
	PO-003	0	0	1	1
	Subtotal	1	1	0	1
		Record Count	1	1	2
		Total			3

Detailed Table:

Purchase Order: ID	Actual Delivery Date	Expected Delivery Date	Order Date	Product ID	Purchase Order: Owner Name	Purchase Order: Owner Alias	Purchase Order: Owner Role	Purchase Order: Created By
1 a01g5000002YzQD	3/1/2025	11/4/2025	11/1/2025	PROD-001	Ebinez Johnson	ebl	-	Ebinez Johnson
2 a01g5000002Ylnr	3/1/2025	11/5/2025	11/2/2025	-	Ebinez Johnson	ebl	-	Ebinez Johnson
3 a01g5000002YYql	3/1/2025	11/6/2025	11/3/2025	PROD-001	Ebinez Johnson	ebl	-	Ebinez Johnson
4								

Purchase Orders based on Suppliers



The screenshot shows a report titled "Complete Purchase Details Report". The report displays a summary table and a detailed table below it.

Summary Table:

Total Records	Total Order Count	Total Quantity received	Total Amount
5	5	80	\$280.00

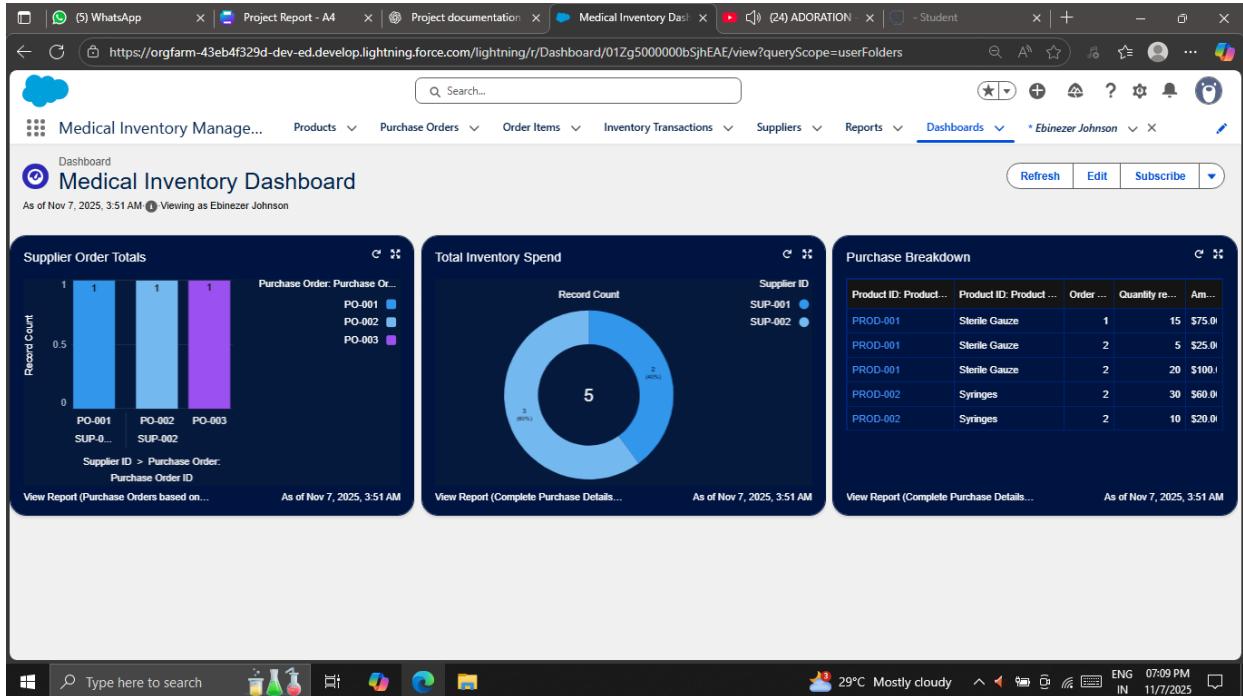
Detailed Table:

Supplier ID	Actual Delivery Date	Purchase Order: Purchase Order ID	Product ID: Product ID	Product ID: Product Name	Order Count	Quantity received	Amount
SUP-001 (2)	3/1/2025 (2)	PO-001 (2)	PROD-001	Sterile Gauze	2	20	\$100.00
			PROD-002	Syringes	2	10	\$20.00
					2	30	\$120.00
					2	30	\$120.00
					2	30	\$120.00
Subtotal							
SUP-002 (3)	3/1/2025 (3)	PO-002 (1)	PROD-001	Sterile Gauze	1	15	\$75.00
					1	15	\$75.00
		PO-003 (2)	PROD-001	Sterile Gauze	2	5	\$25.00
			PROD-002	Syringes	2	30	\$60.00
					2	35	\$85.00
					3	50	\$160.00
					3	50	\$160.00
Subtotal							
Total (5)					5	80	\$280.00

Complete Purchase Details Report

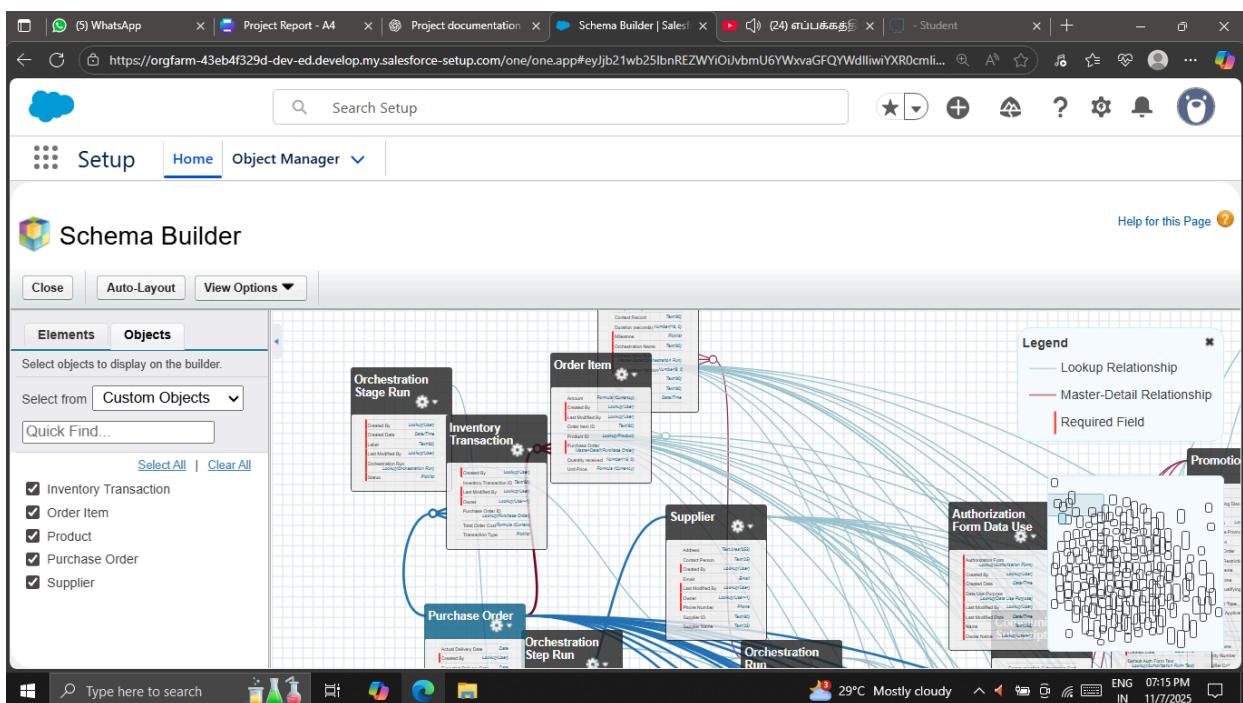
APPENDIX – SCREENSHOTS

14) Dashboard



Dashboard visualizing overall purchase performance and supplier data.

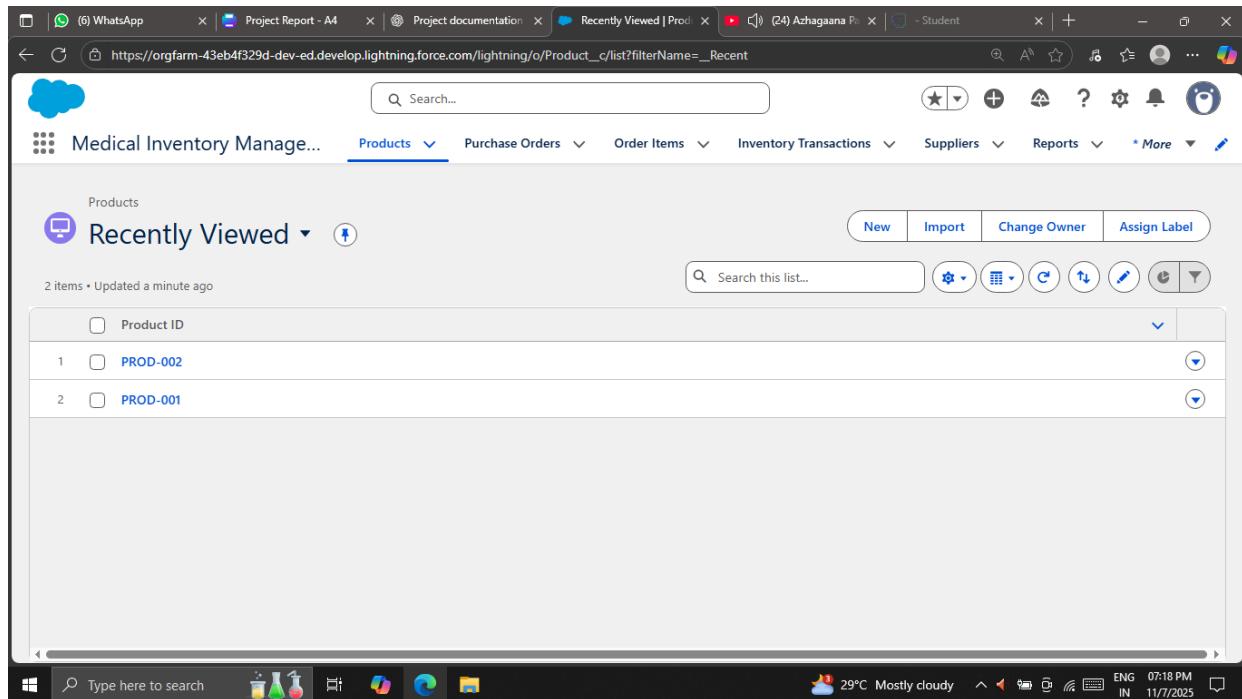
15) Schema Builder



Entity Relationship Diagram of Medical Inventory Management System.

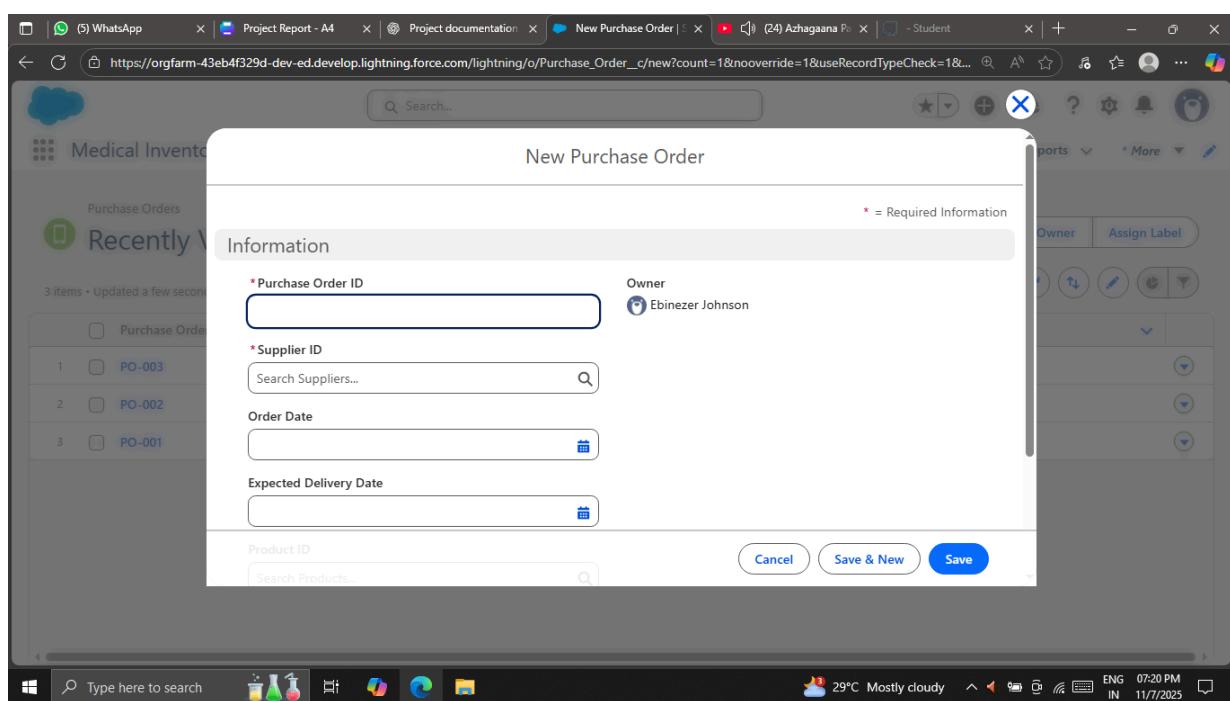
APPENDIX – SCREENSHOTS

16) Application Home Page (Final UI)



Medical Inventory Management Application Interface in Salesforce.

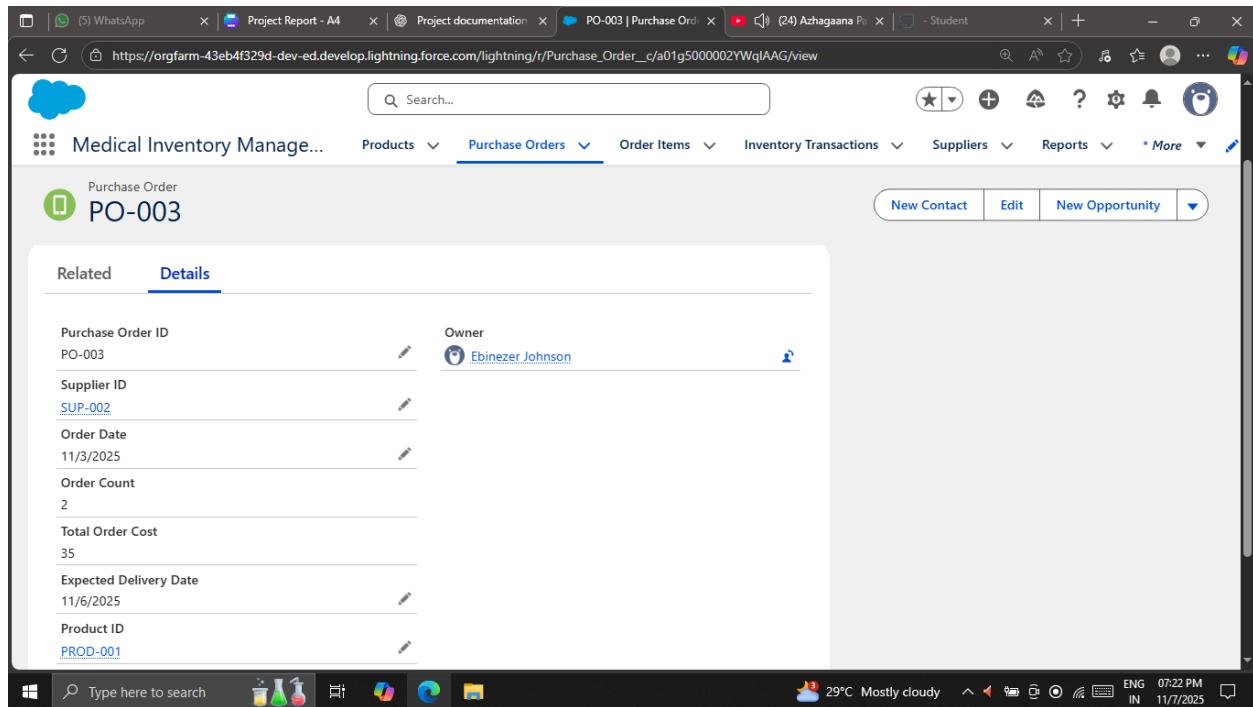
17) Record Creation Example



Sample record creation demonstrating system functionality.

APPENDIX – SCREENSHOTS

18) Sample Data Display



The screenshot shows a Salesforce Lightning interface for a Purchase Order. The top navigation bar includes links for WhatsApp, Project Report - A4, Project documentation, PO-003 | Purchase Ord, (24) Azhagaana Po, Student, and others. The main header displays "Medical Inventory Manage..." and the specific Purchase Order "PO-003". The "Purchase Orders" tab is currently selected. The "Details" section contains the following data:

Field	Value
Purchase Order ID	PO-003
Supplier ID	SUP-002
Order Date	11/3/2025
Order Count	2
Total Order Cost	35
Expected Delivery Date	11/6/2025
Product ID	PROD-001

Below the details, there are tabs for "Related" and "Purchase Order ID". The "Owner" field is listed as Ebinezer Johnson. The bottom of the screen shows the Windows taskbar with a search bar, pinned icons for File Explorer, File History, Task View, and Edge, and system status indicators for weather (29°C, Mostly cloudy), battery (ENG IN 07:22 PM 11/7/2025), and network.

Sample data added for testing Reports and Dashboards.

REFERENCES

- **Salesforce Trailhead Documentation —**
<https://trailhead.salesforce.com>
- **SmartInternz Naan Mudhalvan Portal —**
<https://nme.smartinternz.com>
- **Salesforce Developer Guide —**
<https://developer.salesforce.com>
- **GitHub Repository —**
<https://github.com/Ebinezero04/Medical-Inventory-Management-System-Salesforce-Project-Documentation-Report>