

NAAN MUDHALVAN PROJECT REPORT

SB8067- SALESFORCE DEVELOPER

“MEDICAL INVENTORY MANAGEMENT “



Submitted by:

DIVYA V (REG.NO:912022243005)

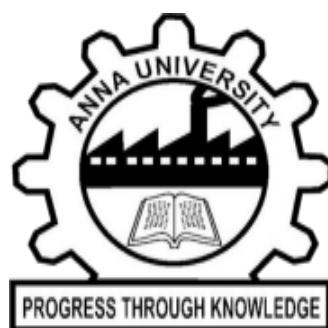
KHUSHI M (REG.NO:912022243009)

KEERTHANA R (REG.NO: 912022243008)

SHALINI K (REG.NO:912022243013)

JAI SRIRAM S (REG.NO:912022243007)

**PANDIAN SARASWATHI YADAV ENGINEERING COLLEGE,
SIVAGANGAI**



ANNA UNIVERSITY: CHENNAI - 600 025 NOV-DEC 2025

BONAFIDE CERTIFICATE

Certified that this Naan Mudhalvan report “**MEDICAL INVENTORY MANAGEMENT**” is the Bonafide work of “DIVYA V(REG.NO:912022243005)KHUSHI.M(REG.NO:912022243009)KEERTHAN A.R(REG.NO:912022243008)SHALINI.K(REG.NO:912022243013)JAISRIRA M.S(REG.NO:912022243007)” who carried out the mini project work under supervision.

COURSE COORDINATOR,
MS.D.SWETHA,ME.,
ASSISTANT PROFESSOR,
ARTIFICIAL INTELLIGENCE AND DATA
SCIENCE
PANDIAN SARASWATHI YADAV
ENGINEERING COLLEGE

HEAD OF THE DEPARTMENT ,
MS.D.VANATHI, M.TECH.,
(PH.D.)
ASSISTANT PROFESSOR,
ARTIFICIAL INTELLIGENCE
AND DATA SCIENCE
PANDIAN SARASWATHI YADAV
ENGINEERING COLLEGE

STAFF IN- INCHARGE

HEAD OF THE DEPARTMENT

Submitted for Viva-Voice Examination held on_____

INTERNAL EXAMINER**EXTERNAL EXAMINER****ACKNOWLEDGEMENT**

First of all, I thank God Almighty for His wisdom and His substantial blessings by which I have been able to complete my project successfully.

I would like to express my sincere thanks and gratitude to our beloved Founder “Hind Rattan Mr. Malaysia S.Pandian” Of our college for his support.

I express my thanks to our beloved Managing Director Er.S.P.Varadharajan,B.E., of our college, for his support. My special thanks to our principal Dr.P.Raveendran.,M.E.,Ph.D., for permitting me to undertake this project.

I am also grateful to our Head of the Department Ms.D.Vanathi, M.Tech., (Ph.D.)Department of Artificial Intelligence and Data Science and my project supervisor Ms.D.Swetha.M.E, Department of Artificial Intelligence and Data Science for making available excellent infrastructure that has enabled me the completion of the project and also for providing the great opportunity to pursue this project and guided me at every stage of my project to make it a successful one.

I am grateful to express my gratitude to my family members and friends for their support and effort in encouraging me to complete this project in an efficient manner.

TABLE OF CONTENTS:

S.NO	TITLE
1.	Introduction
2.	Problem Statement
3.	Objective
4.	Problem Description
5.	Proposed Solution
6.	Methodology
7.	Workflow
8.	Steps
9.	Implementation Details
10.	System Design
11.	Modules Description
12.	Output
13.	Advantages
14.	Features
15.	Use Cases
16.	Challenges Faced
17.	Future Enhancements
18.	Learning Outcomes
19.	Conclusion

1. INTRODUCTION:

The **Medical Inventory Management System** is a Salesforce-based cloud application developed to streamline all aspects of medical stock control—product cataloging, supplier coordination, purchase-order tracking, and expiry surveillance.

Effective inventory administration is essential in healthcare to guarantee the timely availability of medicines and consumables. This system integrates Salesforce's declarative capabilities (Flows, Validation Rules, Reports, Dashboards) with programmatic tools such as Apex Triggers to deliver an intelligent, automated, and centralized framework.

Through this implementation, the project demonstrates the capacity of Salesforce Lightning to enhance reliability, reduce manual workload, and improve decision-making across hospital supply operations.

2. PROBLEM STATEMENT:

Many healthcare institutions still depend on manual record-keeping or outdated on-premise systems for inventory control.

Such methods create persistent issues including:

- Inaccurate stock counts and inconsistencies between physical and digital records.
- Poor visibility of expiry dates, leading to wastage of medicines.
- Lack of structured supplier data and accountability.
- Procurement delays caused by human intervention in repetitive ordering tasks.
- Incomplete or unreliable analytical reports.

These deficiencies result in higher operational costs, inefficiency, and risk to patient safety. An automated, cloud-driven solution is therefore essential to ensure precision, accountability, and real-time monitoring.

3. OBJECTIVE:

The primary objective is to design and deploy an automated Salesforce application that monitors medical inventory and optimizes resource flow.

Specific aims:

1. Build a centralized data model for suppliers, products, and purchase orders.
2. Automate inventory updates and expiry tracking using Flows and Apex Triggers.
3. Maintain data integrity through validation rules and controlled relationships.
4. Provide an intuitive interface through Lightning Apps and Page Layouts.
5. Offer real-time analytical insights via Reports and Dashboards.
6. Strengthen managerial decision-making and procurement efficiency.

4. PROBLEM DESCRIPTION:

Manual inventory systems are time-consuming, error-prone, and unscalable for hospitals managing thousands of SKUs.

Without automation, staff face challenges such as:

- Overstocking or shortage of critical items.
- Human errors in data entry and stock reconciliation.
- Absence of proactive alerts for near-expiry products.
- Communication lags with suppliers.
- Difficulty generating accurate performance metrics.

Adopting Salesforce CRM as a digital backbone allows real-time synchronization of records, automated validation, and structured workflow management—thereby minimizing human error and optimizing resource allocation

5. PROPOSED SOLUTION:

The proposed Salesforce-based Medical Inventory Management System addresses these limitations through end-to-end automation.

It utilizes custom objects, tailored relationships, and Lightning Components to manage all data entities. The application enables users to:

- Register suppliers and add product details.
- Create and process purchase orders.
- Track quantities and expiry timelines automatically.
- Receive alerts for low stock or impending expirations.
- Visualize operational metrics through dashboards and analytical reports.

The screenshot shows the Salesforce Object Manager page. At the top, there's a navigation bar with 'Setup' and 'Object Manager'. Below it is a sidebar titled 'Object Man' with a 'RECENT RECORDS' section containing 'Supplier' and other standard objects like 'Account', 'Order Item', etc. The main area displays a table of custom objects. The columns are labeled 'LABEL', 'ME', 'TYPE', 'DESCRIPTION', 'LAST MODIFIED', and 'DEPLOYED'. The table includes rows for 'Inventory Transaction', 'Alternative Payment Method', 'API Anomaly Event Store', 'Appointment Category', 'Appointment Invitation', 'Appointment Invitee', 'Appointment Topic Time Slot', 'Approval Submission', 'Approval Submission Detail', 'Approval Work Item', 'Asset', 'Asset Action', and 'AssetActionSource'. The 'TYPE' column consistently shows 'Standard Object' for most entries, except for 'Alternative Payment Method' which is 'AlternativePaymentMethod'.

Technology Used:

- Salesforce Platform (Lightning Experience)
- Apex (for automation logic)
- Process Builder and Flow Builder
- Validation Rules
- Reports and Dashboards

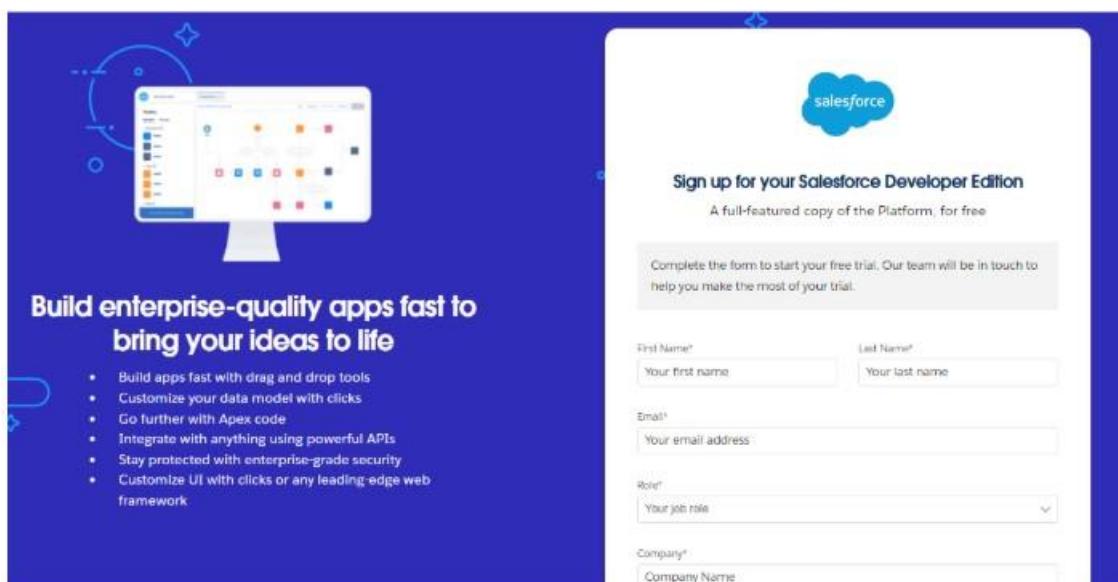
The Salesforce environment ensures **security**, **scalability**, and **accessibility**, making it ideal for healthcare organizations.

6. METHODOLOGY:

The project was executed through a structured and iterative approach within the Salesforce Developer environment to ensure accuracy and maintainability.

Phase 1 – Environment Setup

- Creation of a Salesforce Developer Account.
- Enabling the Lightning Experience for enhanced UI.
- Configuring profiles and permission sets for user access.



Phase 2 – Object Design

- Creation of four main custom objects:
 - **Supplier** (Supplier Name, Contact, Email, Address)
 - **Product** (Product Name, Batch Number, Expiry Date, Stock Quantity)
 - **Purchase Order** (Order ID, Date, Supplier, Status)
 - **Transaction** (Purchase Order ID, Product, Quantity, Delivery Date)

The screenshot shows the Salesforce Object Manager interface. On the left, there's a sidebar with 'RECENT RECORDS' and a list of objects. The main area displays a table with columns: ME, TYPE, DESCRIPTION, LAST MODIFIED, and DEPLOYED. The table lists various standard and custom objects.

ME	TYPE	DESCRIPTION	LAST MODIFIED	DEPLOYED
Account	Standard Object			
Order Item	Standard Object			
Purchase Order	Standard Object			
Product	Standard Object			
Alternative Payment Method	Standard Object			
API Anomaly Event Store	Standard Object			
Appointment Category	Standard Object			
Appointment Invitation	Standard Object			
Appointment Invitee	Standard Object			
Appointment Topic Time Slot	Standard Object			
Approval Submission	Standard Object			
Approval Submission Detail	Standard Object			
Approval Work Item	Standard Object			
Asset	Standard Object			
Asset Action	Standard Object			

Phase 3 – Relationships Mapping

- Supplier → Purchase Order: Lookup Relationship
- Purchase Order → Product: Master-Detail Relationship

The top screenshot shows the 'Custom Tabs' section of the Salesforce Tabs setup page. It includes sections for 'Custom Object Tabs', 'Web Tabs', and 'Visualforce Tabs', each with a 'What is This?' link. The bottom screenshot shows 'Step 1. Enter the Details' for creating a new custom tab. It asks to choose a custom object ('Product') and a tab style ('Stitchscope'). There are fields for a 'Short Description' and an optional 'Home Page Custom Link'.

Phase 4 – Business Logic Implementation

- Validation Rules: Ensure data accuracy and prevent duplication.
- Flows: Automate the stock update and expiry tracking process.
- Apex Triggers: Automatically populate “Actual Delivery Date” once order status changes to “Delivered.”

Step 2: Choose output type

Field Label: 5

Field Name: Cancel

Add this field to existing custom report types that contain this entry Help

Formula Return Type

- None Selected
- Checkbox
- Currency 6
- Date
- Date/Time
- Number
- Percent

Select one of the data types below.

Calculate a boolean value.
Example: `TODAY() > OWDODATE`.

Calculate a dollar or other currency amount and automatically format the field as a currency amount.
Example: `Gross Margin = Amount - Cost_C`.

Calculate a date, for example, by adding or subtracting days to other dates.
Example: `Reminder Date = CloseDate - 7`.

Calculate a datetime, for example, by adding a number of hours or days to another datetime.
Example: `New = NOW() + 1`.

Calculate a numeric value.
Example: `Percentile = 1.0 * Octile_2 + 32`.

Calculate a percent and automatically add the percent sign to the number.
Example: `Percent = 10 * Octile_2 + 32`.

Order Item
New Custom Field

Step 3: Enter formula

Enter your formula and click Check Syntax to check for errors. Click the Advanced Formula tab to use additional fields, operators, and functions.
Example: `Gross Margin = Amount - Cost_C`. More Examples

Simple Formula Advanced Formula

Insert Field Insert 7

Insert Operator Insert

Functions All Function Categories

- ABS
- ACOS
- ADDMONTHS
- AND
- ASCII
- ASIN

Insert Selected Function

Previous Next Cancel

Step 3 of 5

Help for this Page Help

Quick Tips 8

- Getting Started
- Operators & Functions

Phase 5 – Reports and Dashboards

Constructed dashboards to visualize product stock summary, expiry count, and supplier performance.

Phase 6 – Testing and Deployment

Performed unit testing for all components to verify functionality, reliability, and integration performance.

8. STEPS:

Milestone	Description
1	Create Developer Account
2	Add Custom Objects
3	Add Object Tabs
4	Build Lightning App
5	Add Fields & Relations
6	Customize Layouts
7	Create Compact Layouts
8	Add Validation Rules
9	Set Profiles and Roles
10	Create Users
11	Define Permission sets
12	Build Flows For Updates
13	Add Apex Triggers
14	Generate Reports
15	Create Dashboards
16	Test and Deployment
17	Finalize Documentation

m

9. IMPLEMENTATION DETAILS:

Each module within the system was configured through a combination of Salesforce administrative tools and developer customization.

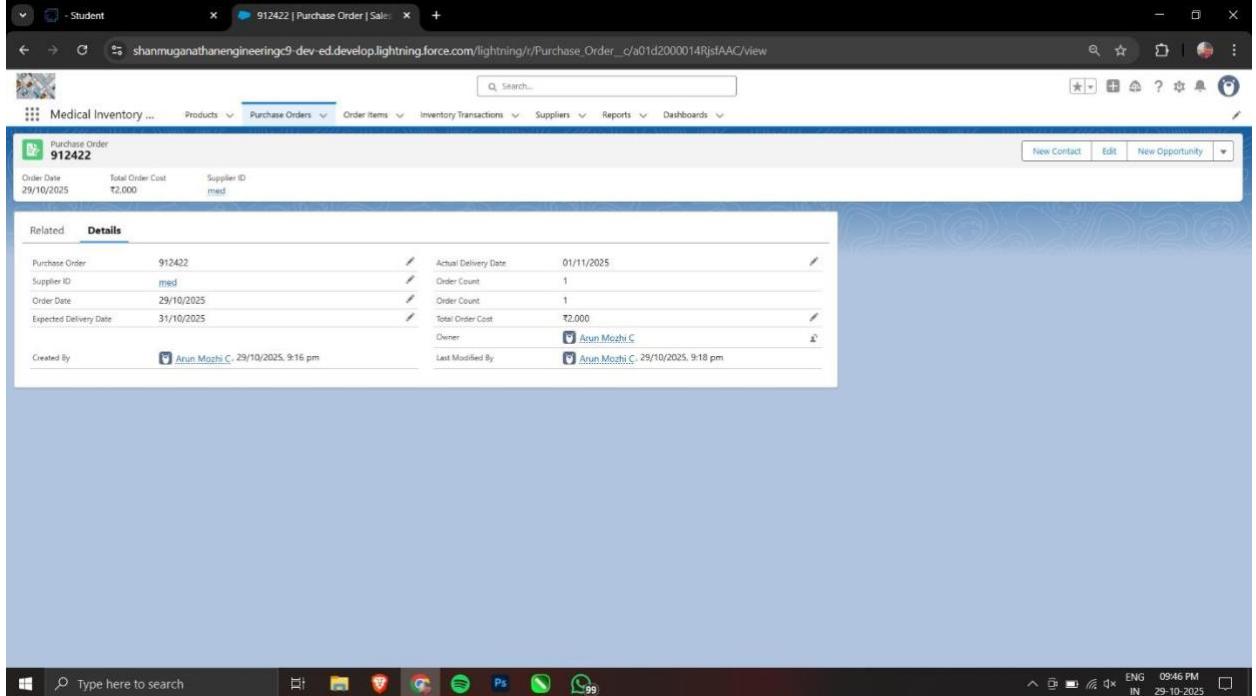
Example – Apex Trigger:

Automatically updates the “Actual Delivery Date” field whenever the status of a purchase order changes to Delivered.

Example – Flow Automation:

A scheduled Flow checks all active products daily; if a product's expiry date is within 30 days, an alert email is automatically sent to the admin and concerned staff.

These automations ensure minimal manual oversight while maintaining high data accuracy.



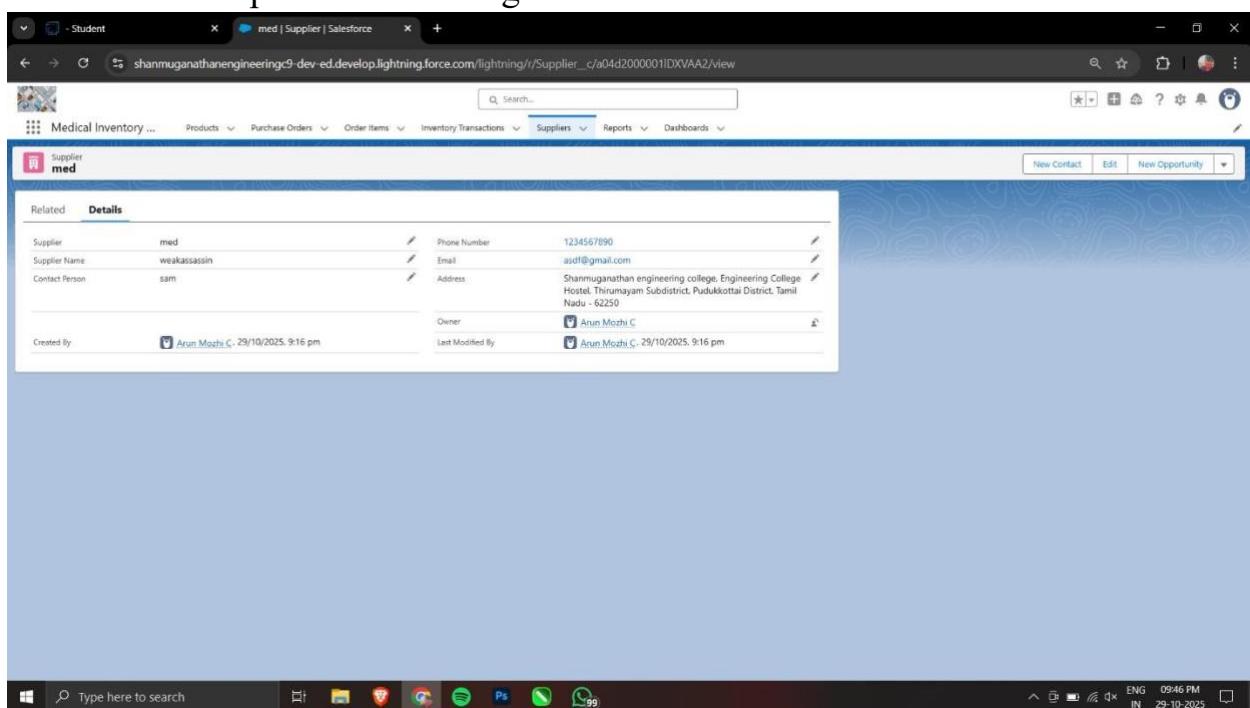
10. SYSTEM DESIGN:

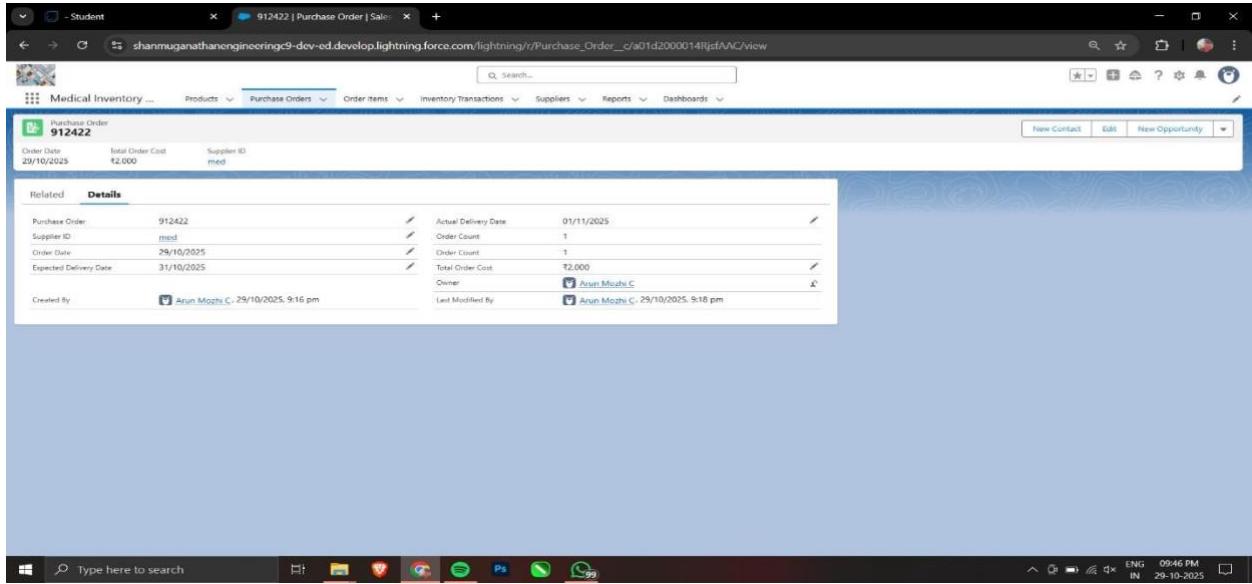
The Medical Inventory Management System is structured in multiple operational layers to ensure logical separation of tasks and maintain data integrity:

- User Layer: Involves hospital personnel, administrators, and pharmacists interacting with the Lightning app.
- Application Layer: Comprises the Salesforce Lightning Components, Process Builders, and Flow Automations responsible for business logic.
- Data Layer: Maintains structured records through custom Salesforce objects, relationships, and field dependencies.
- The architecture guarantees seamless communication between layers, ensuring real-time synchronization and secure data handling across the platform.

11. MODULES DESCRIPTION:

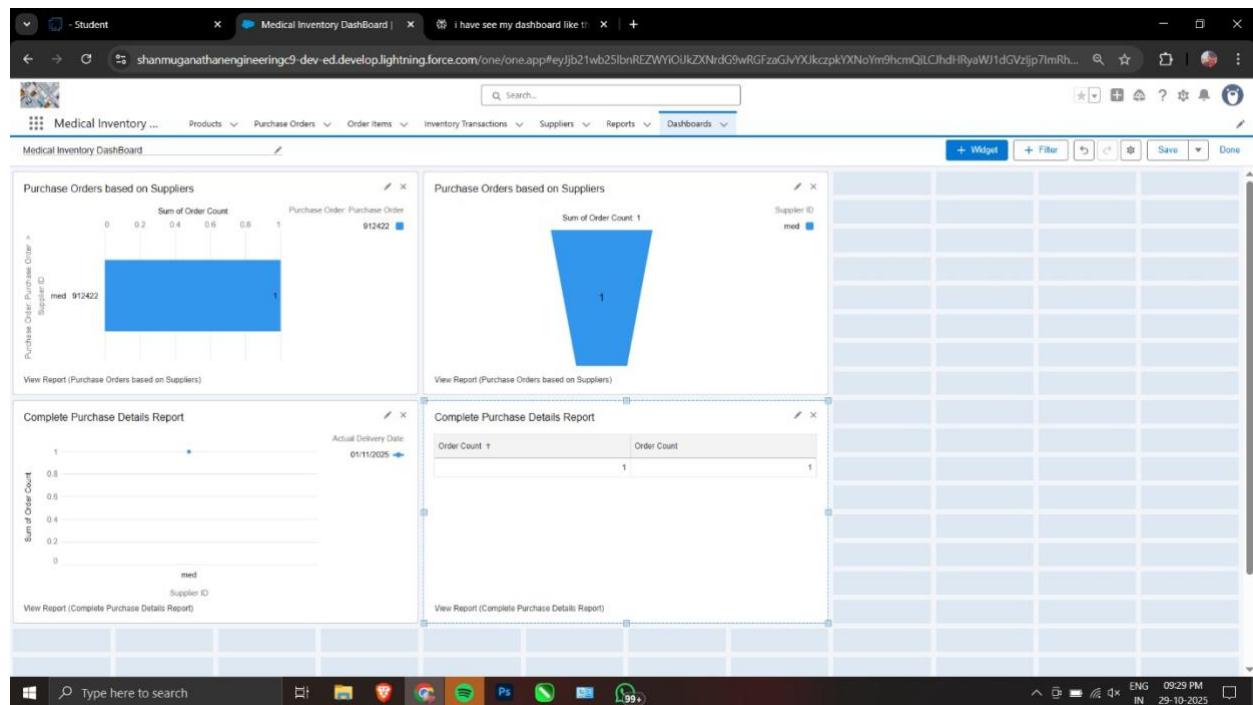
The system is divided into several modules that work together to maintain smooth medical inventory operations. The Supplier Management module stores supplier details and communication records. The Product Management module maintains product information such as name, batch number, expiry, and quantity. The Purchase Order module tracks each order, its supplier, and delivery status. The Expiry Tracking module automatically monitors expiry dates and alerts users when products near expiration. Finally, the Reporting Module compiles all data into dashboards for quick and clear performance insights.





12. OUTPUT:

Once implemented, the system achieved complete automation of inventory monitoring. Stock quantities and expiry data updated automatically, and dashboards provided real-time overviews of supplier activity and product status. This automation reduced manual work, improved accuracy, and simplified the management of medical resources across departments.



13. ADVANTAGES:

The application ensures real-time visibility of stock levels and expiry timelines, minimizing wastage and delays. It centralizes all records in a secure cloud environment accessible through authorized accounts. Automated alerts and validation rules enhance data reliability, while analytical dashboards support faster and more informed decision-making.

14. FEATURES:

- Custom Salesforce Objects
- Lightning Record Pages
- Automation through Flows and Triggers
- Data Validation Rules
- Reports and Dashboards
- Expiry Notifications
- User Management and Permissions

The screenshot shows a Salesforce Lightning Report titled "Complete Purchase Details Report". The report displays a table with the following data:

Total Records	Total Order Count	Total Order Count		
Supplier ID	Actual Delivery Date	Purchase Order: Purchase Order	Order Count	Order Count
1	01/11/2025	912422	1	1
		Subtotal	1	1
		Subtotal	1	1
		Total (1)	1	1

At the bottom of the report, there are checkboxes for "Row Counts", "Detail Rows", "Subtotals", and "Grand Total". The status bar at the bottom right shows "ENG 09:25 PM IN 29-10-2025".

The image consists of two vertically stacked screenshots of a Salesforce Lightning application. Both screenshots show a modal dialog box over a background dashboard.

Screenshot 1: New Inventory Transaction

This modal is titled "New Inventory Transaction". It contains the following fields:

- * Transaction ID: 12333
- * Transaction Date: 29/10/2025
- Purchase Order: 912422
- Transaction Type: Receipt
- * Inventory Transaction: 912422

Owner: Arun Mozhi C

Buttons at the bottom: Cancel, Save & New, Save

Screenshot 2: New Order Item

This modal is titled "New Order Item". It contains the following fields:

- * Order Item: med
- * Purchase Order: 912422
- Product: 912422
- Quantity Received: 10
- Quantity Ordered: 20

Buttons at the bottom: Cancel, Save & New, Save

15. USE CASES:

1. Pharmacy Inventory Control

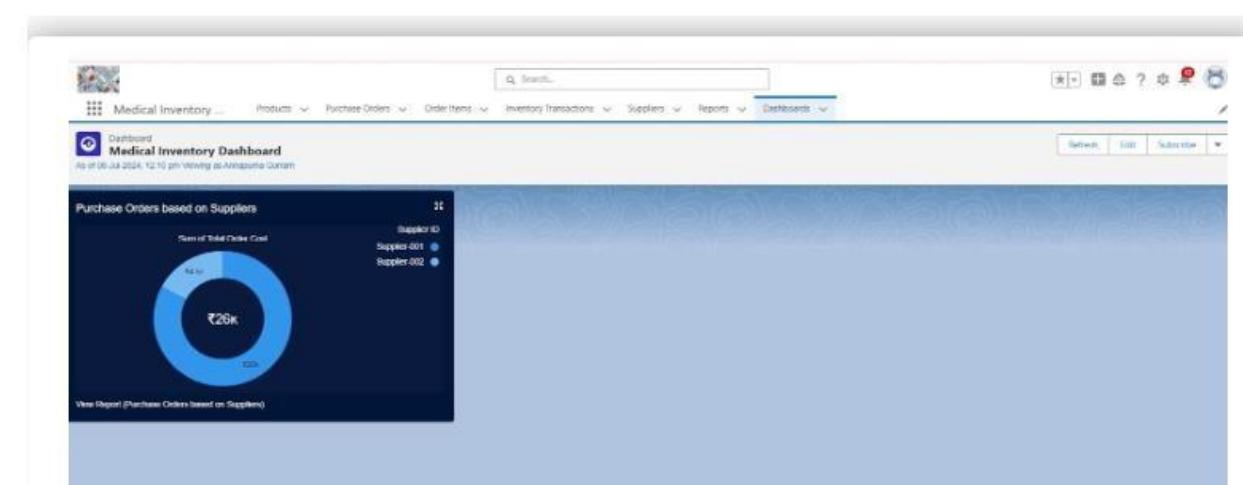
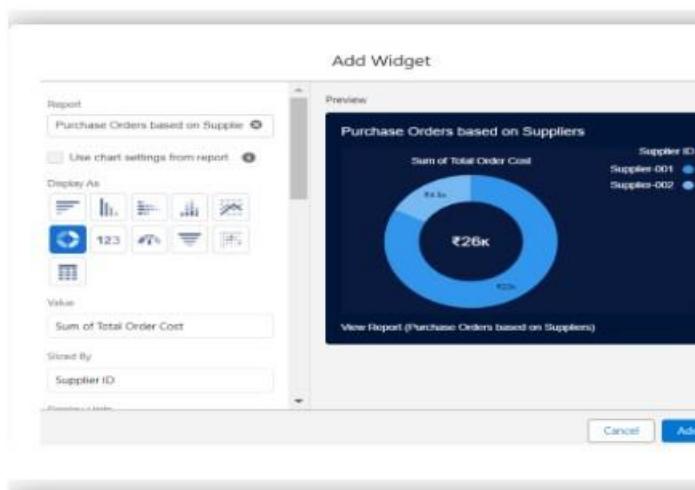
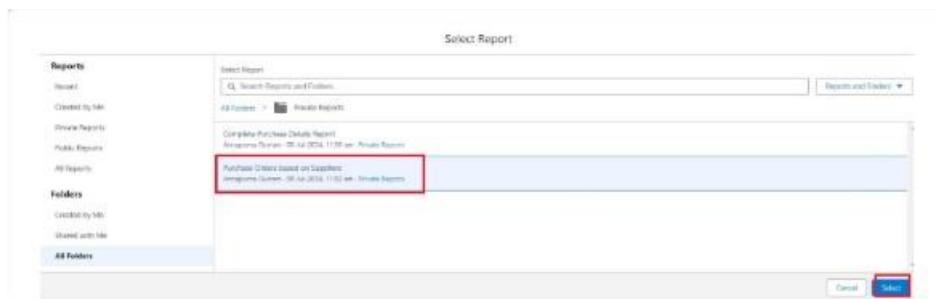
Tracks medicine availability and expiry.

2. Hospital Procurement

Generates and manages purchase orders.

3. Supplier Analytics

Evaluates supplier performance using dashboards.



16. CHALLENGES FACED:

The main challenges during development were resolving conflicts between automation flows and triggers, maintaining consistent data during bulk updates, and designing relationships between multiple custom objects without breaking dependencies. Each issue was addressed through testing and iteration.

17. FUTURE ENHANCEMENTS:

Future improvements may include barcode scanning for faster product entry, AI-based demand forecasting, automatic reordering through supplier APIs, and a dedicated mobile interface to enable on-site inventory tracking.

18. LEARNING OUTCOMES:

Through this project, the team gained practical experience in Salesforce administration, data modeling, automation with Flows and Apex, and report generation. It also strengthened understanding of applying CRM-based solutions to real-world healthcare management scenarios.

19. CONCLUSION:

The Medical Inventory Management System demonstrates how Salesforce can automate healthcare stock control effectively. By integrating validation, automation, and analytics, it enhances accuracy, reduces manual errors, and supports efficient decision-making. The project establishes a strong framework for scalable digital management in medical environments.

20. REFERENCES

- Salesforce Developer Documentation
- Trailhead Modules on Flows, Apex, and Reports
- Naan Mudhalvan Salesforce Developer Program Resources
- Healthcare Inventory Management Case Studies