Medical Inventory Management

College Name: KG COLLEGE OF ARTS AND SCIENCE

TEAM ID: NM2025TMID23760

TEAM MEMBERS: 4

Team LeaderName: SUSHANTH S
Email:2326ka53@kgcas.com

Team Member: SUSMITHA S
Email:2326ka54@kgcas.com

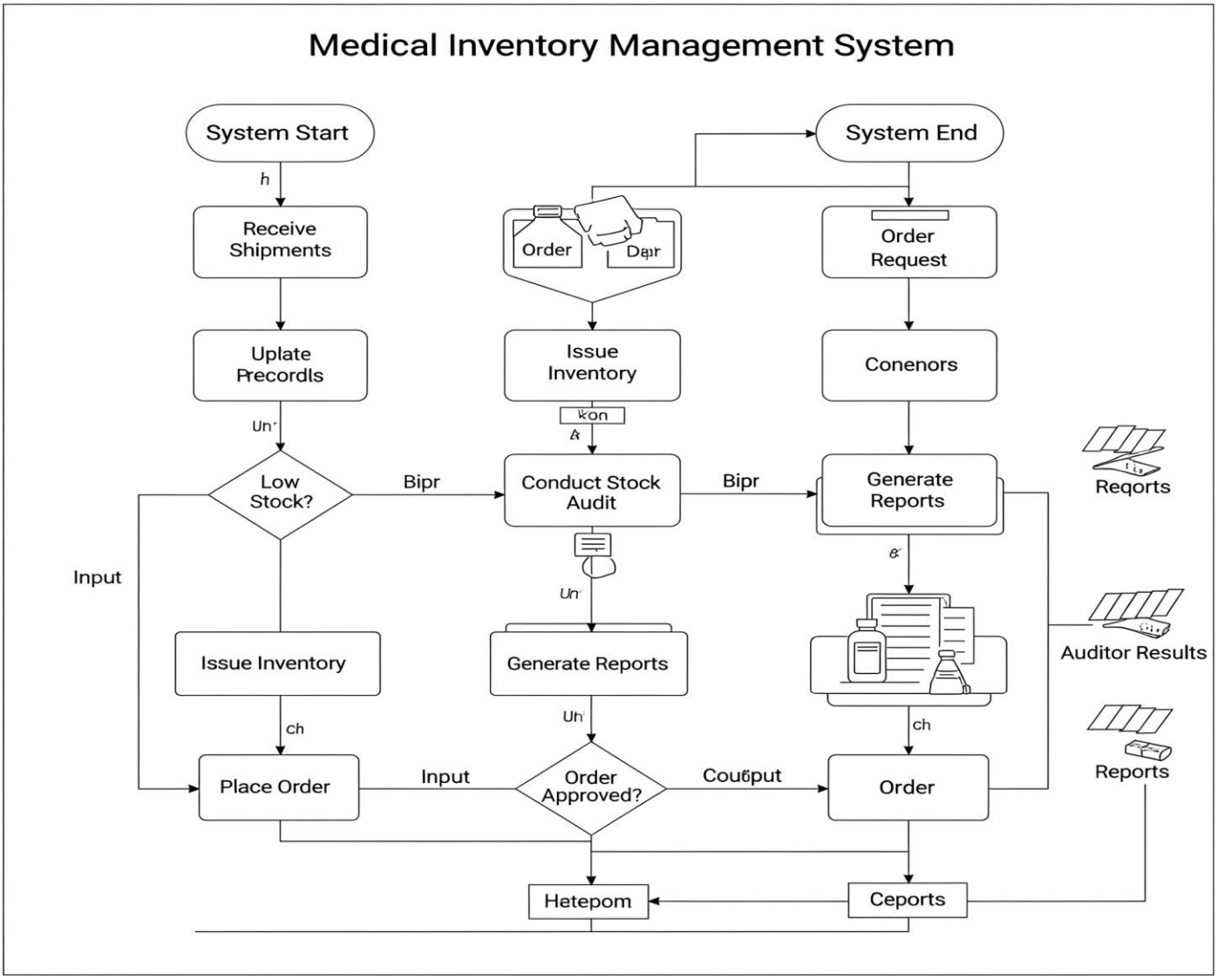
Team Member: THARUNIKA R
Email:2326ka55@kgcas.com

Team Member: VAISHNAVI M
Email: 2326ka56@kgcas.com

1.INTRODUCTION

1.1 Project Overview

The Medical Inventory Management system is designed to streamline the tracking, storage, and distribution of medical supplies. It helps healthcare facilities maintain accurate stock levels, reduce wastage, and ensure timely availability of critical items.



.

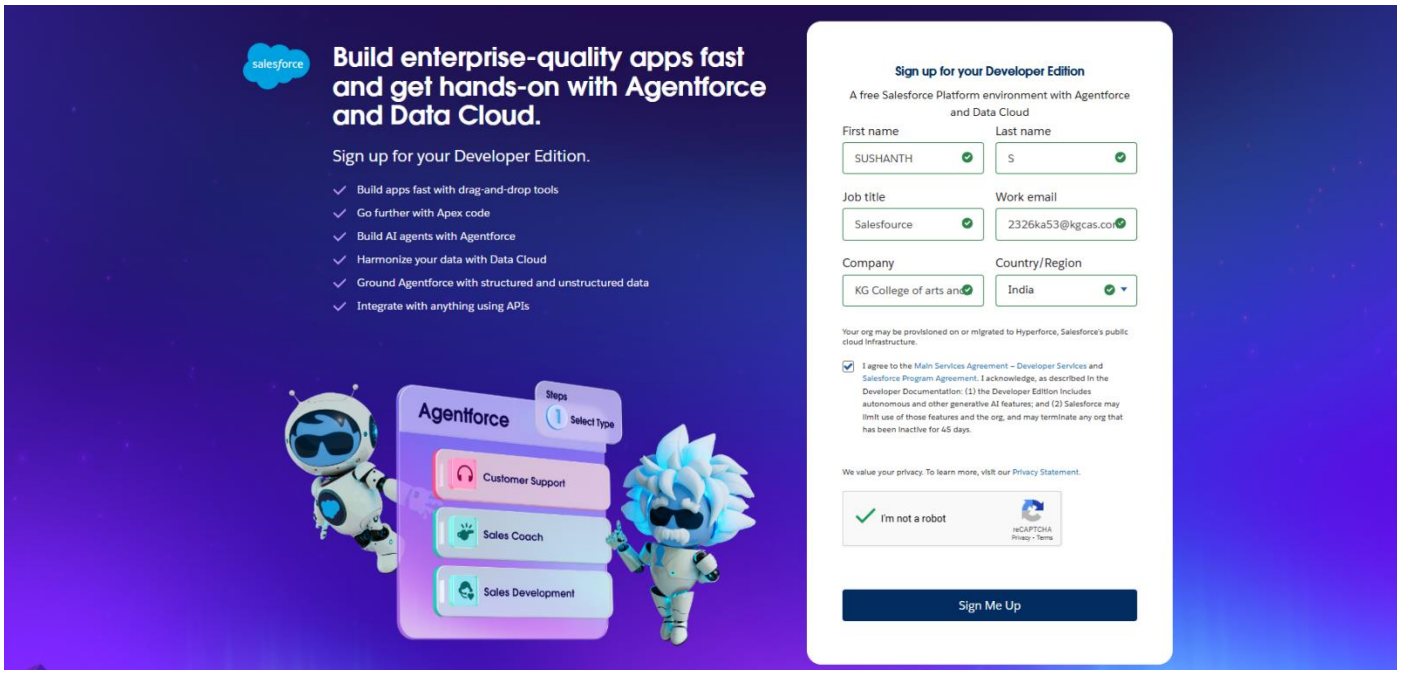
1.2 Purpose

The purpose of a Medical Inventory Management system is to ensure the efficient tracking, control, and distribution of medicines and medical supplies. It helps healthcare organizations maintain optimal stock levels, prevent shortages or overstocking, reduce wastage due to expiry, and provide timely availability of essential items for patient care.

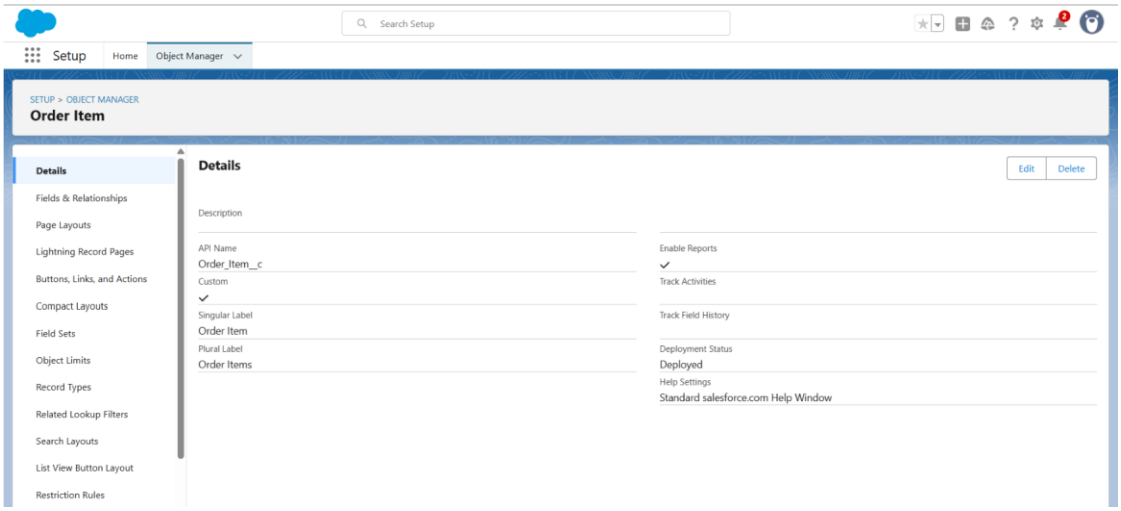
DEVELOPMENT PHASE

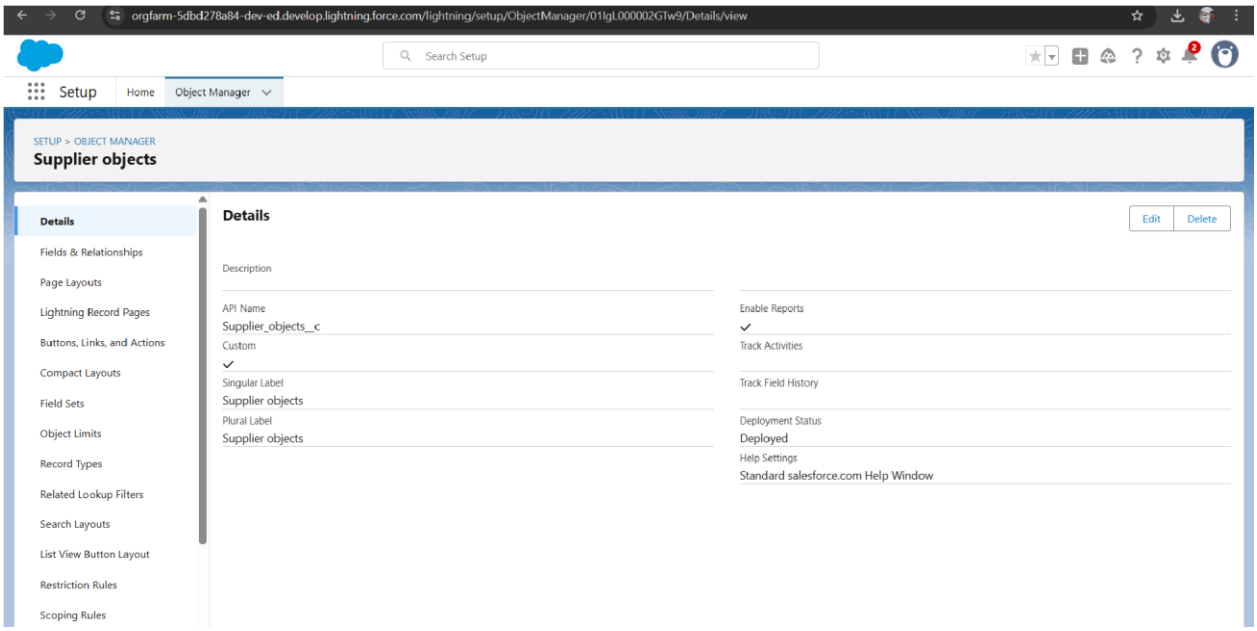
Creating Developer Account:

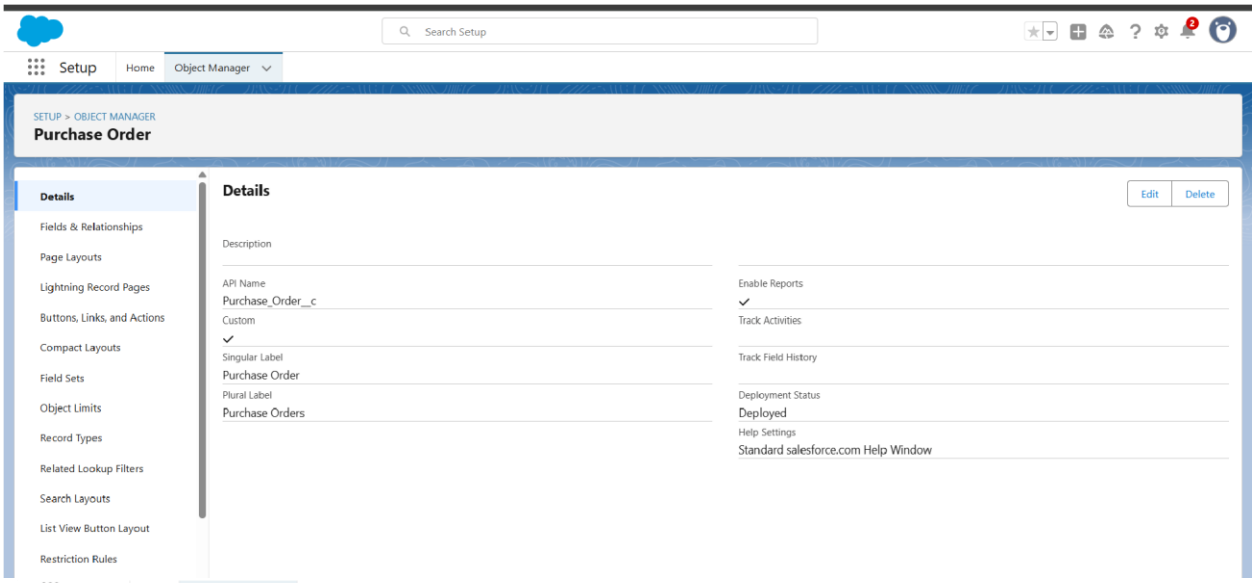
By using this URL - **https://www.salesforce.com/form/developer-signup/?d=pb**

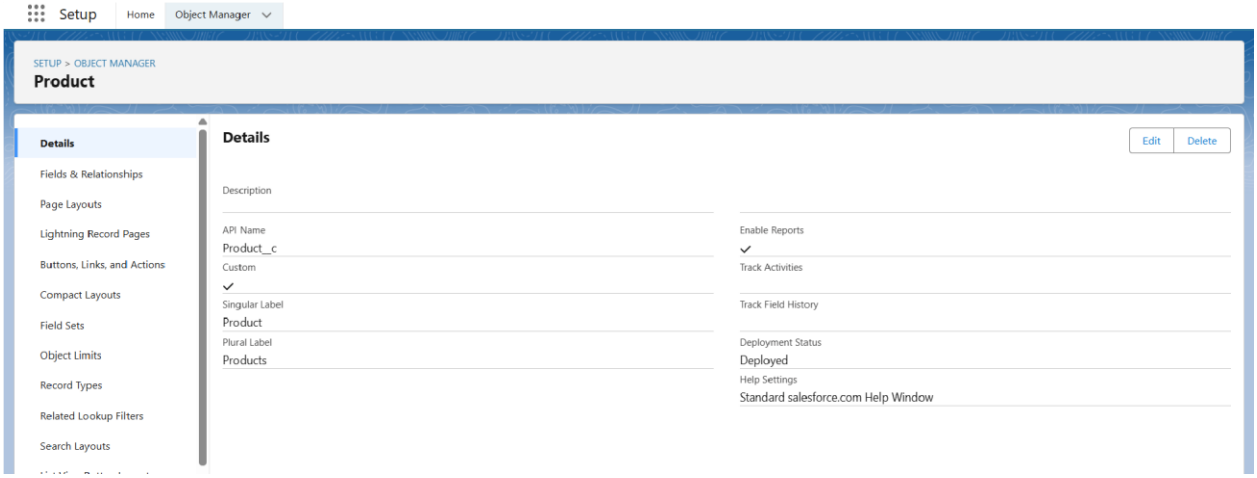


- Created objects: Order Item, Supplier, Purchase Order, Product

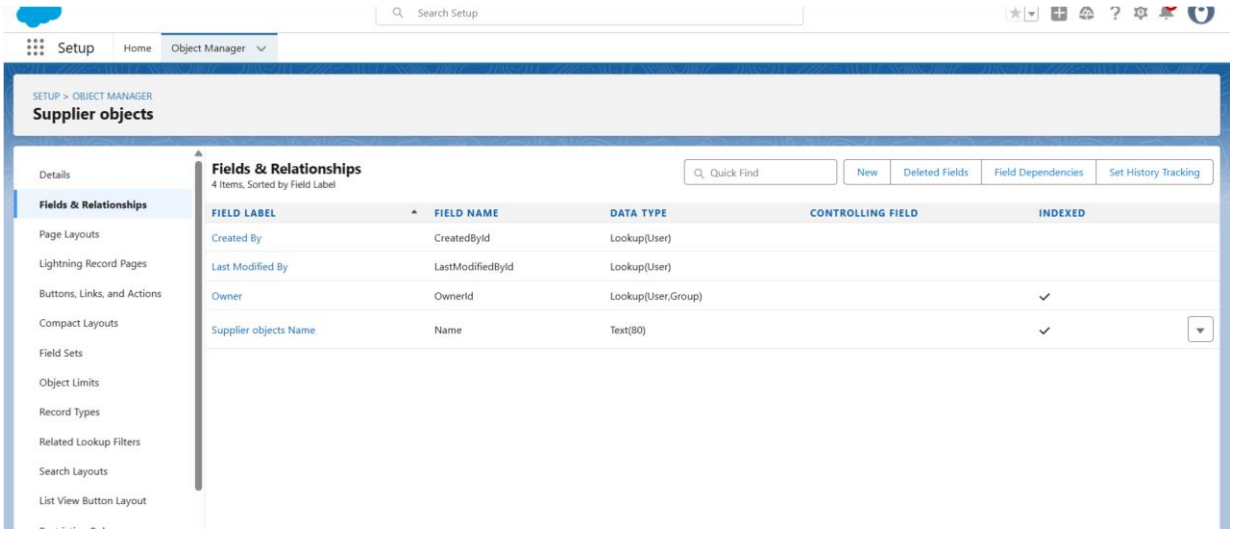


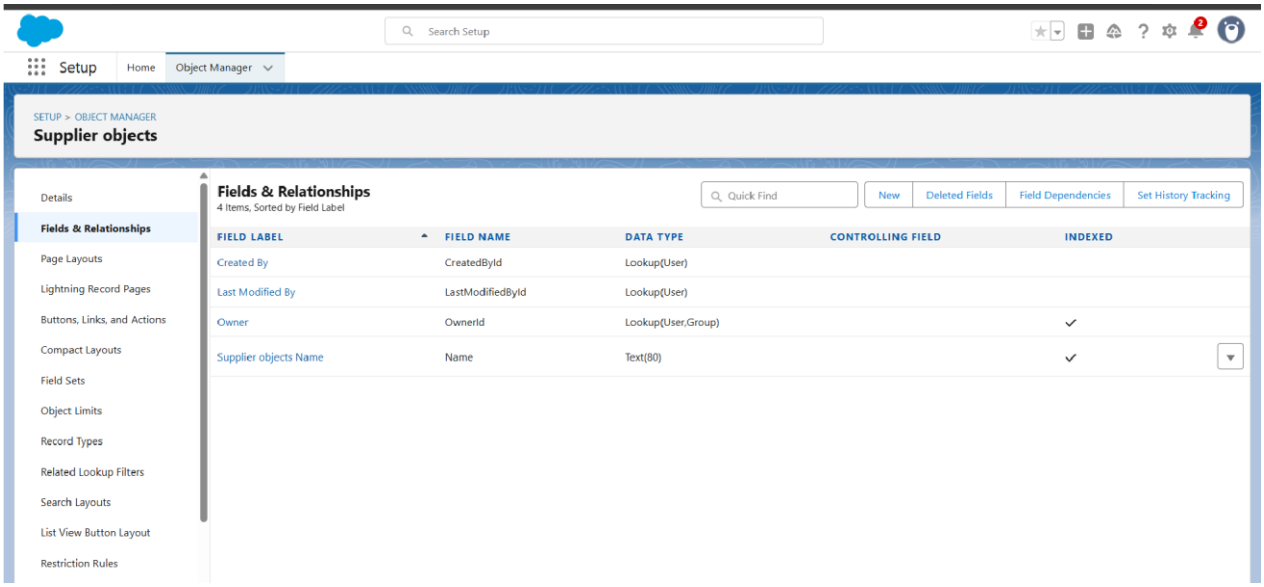

orgfarm-5dbd278a84-dev-ed.develop.lightning.force.com/lightning/setup/ObjectManager/01IgL000002GTw9/Details/view
Search Setup
Setup
Home
Object Manager
SETUP > OBJECT MANAGER
Supplier objects
Details
Fields & Relationships
Page Layouts
Lightning Record Pages
Buttons, Links, and Actions
Compact Layouts
Field Sets
Object Limits
Record Types
Related Lookup Filters
Search Layouts
List View Button Layout
Restriction Rules
Scoping Rules
Details
Edit
Delete
Description
API Name
Supplier_objects__c
Custom
Singular Label
Supplier objects
Plural Label
Supplier objects
Enable Reports
Track Activities
Track Field History
Deployment Status
Deployed
Help Settings
Standard salesforce.com Help Window

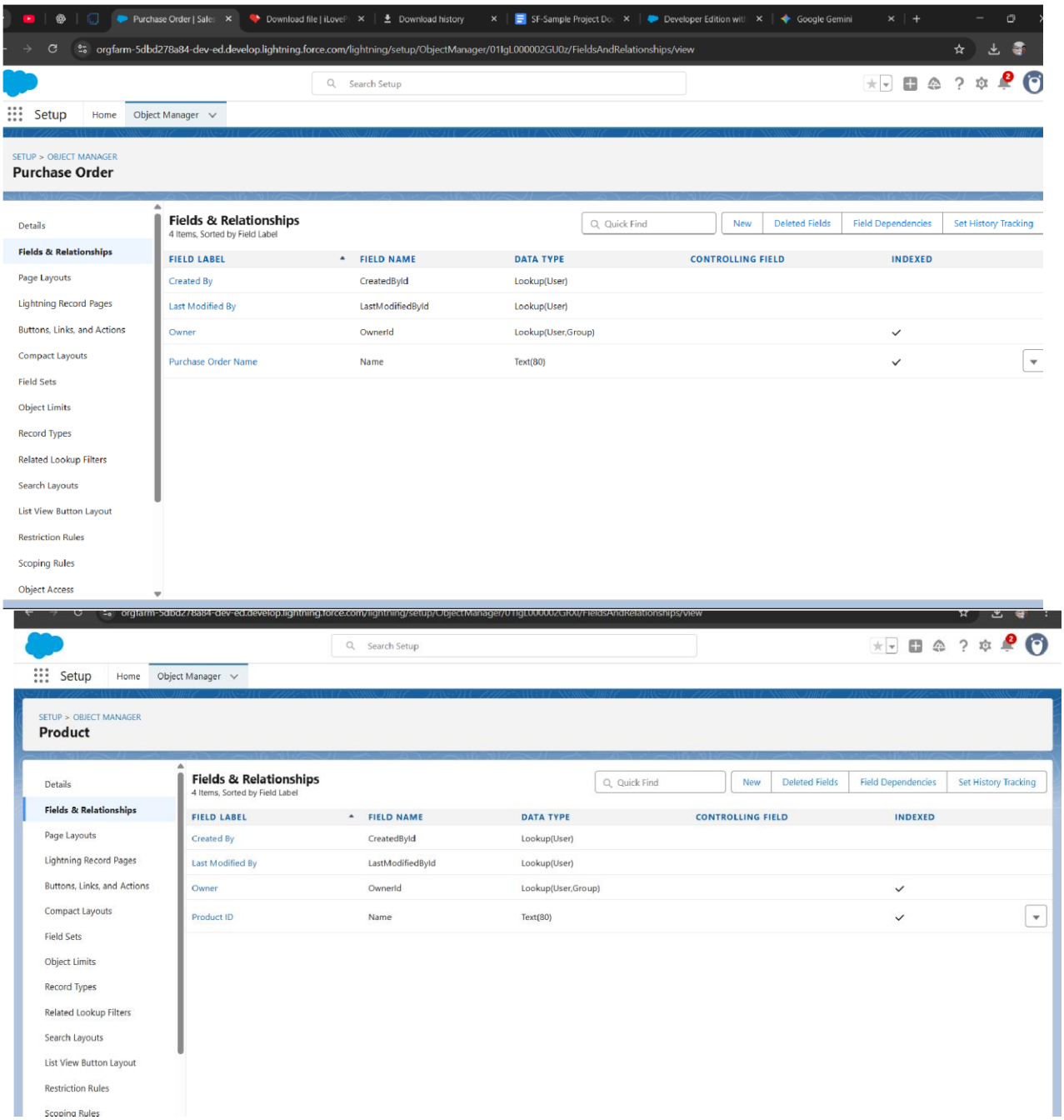

Search Setup
Setup
Home
Object Manager
SETUP > OBJECT MANAGER
Purchase Order
Details
Fields & Relationships
Page Layouts
Lightning Record Pages
Buttons, Links, and Actions
Compact Layouts
Field Sets
Object Limits
Record Types
Related Lookup Filters
Search Layouts
List View Button Layout
Restriction Rules
Details
Edit
Delete
Description
API Name
Purchase_Order__c
Custom
Singular Label
Purchase Order
Plural Label
Purchase Orders
Enable Reports
Track Activities
Track Field History
Deployment Status
Deployed
Help Settings
Standard salesforce.com Help Window


Setup
Home
Object Manager
SETUP > OBJECT MANAGER
Product
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
Details
Edit
Delete
Description
API Name
Product__c
Custom
Singular Label
Product
Plural Label
Products
Enable Reports
Track Activities
Track Field History
Deployment Status
Deployed
Help Settings
Standard salesforce.com Help Window

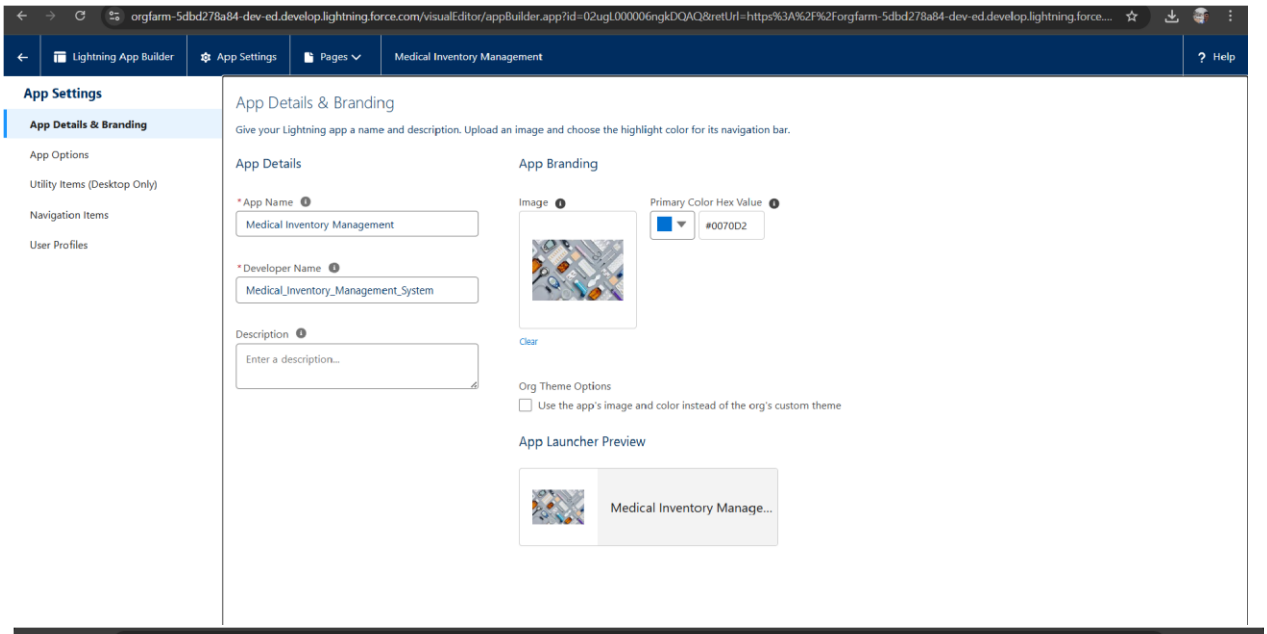
- Configured fields and relationships

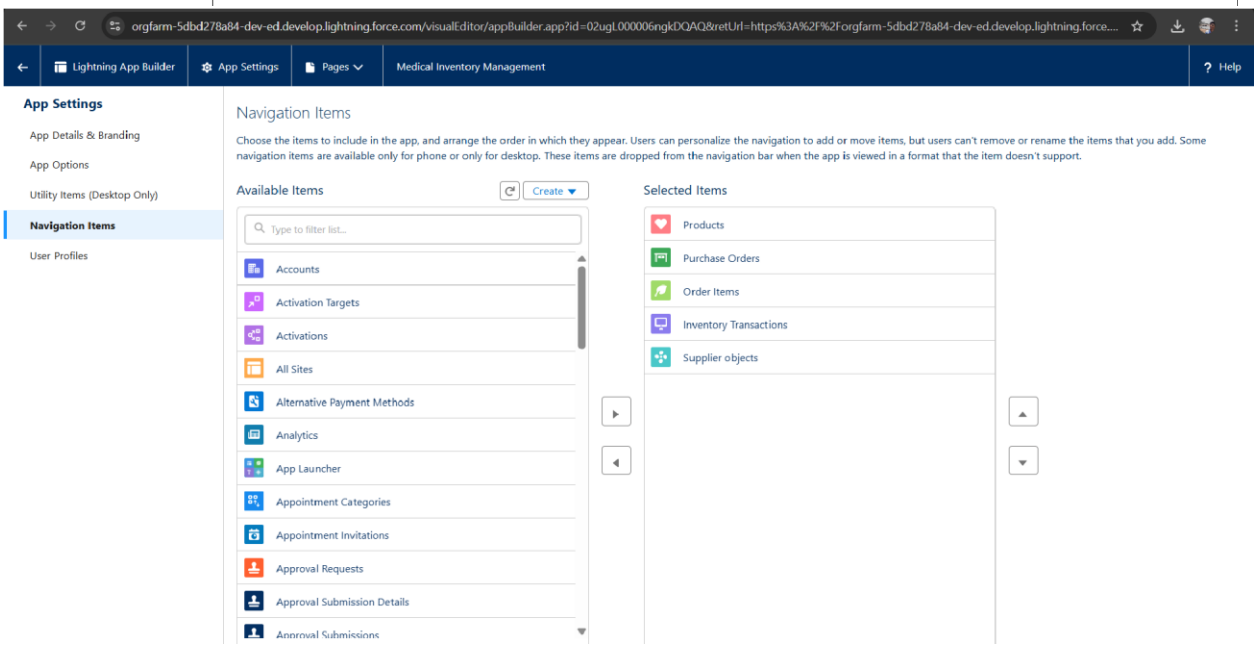


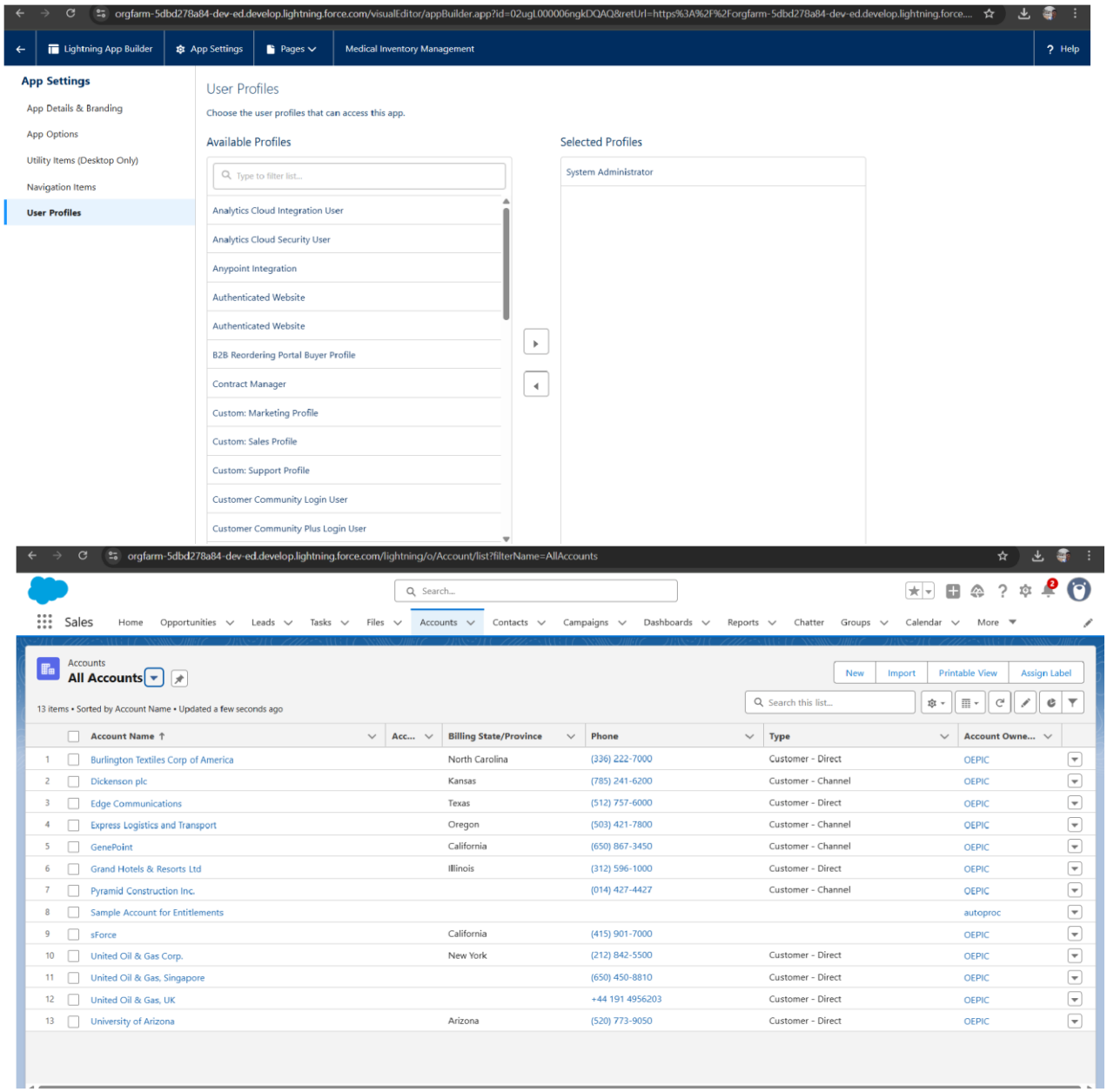




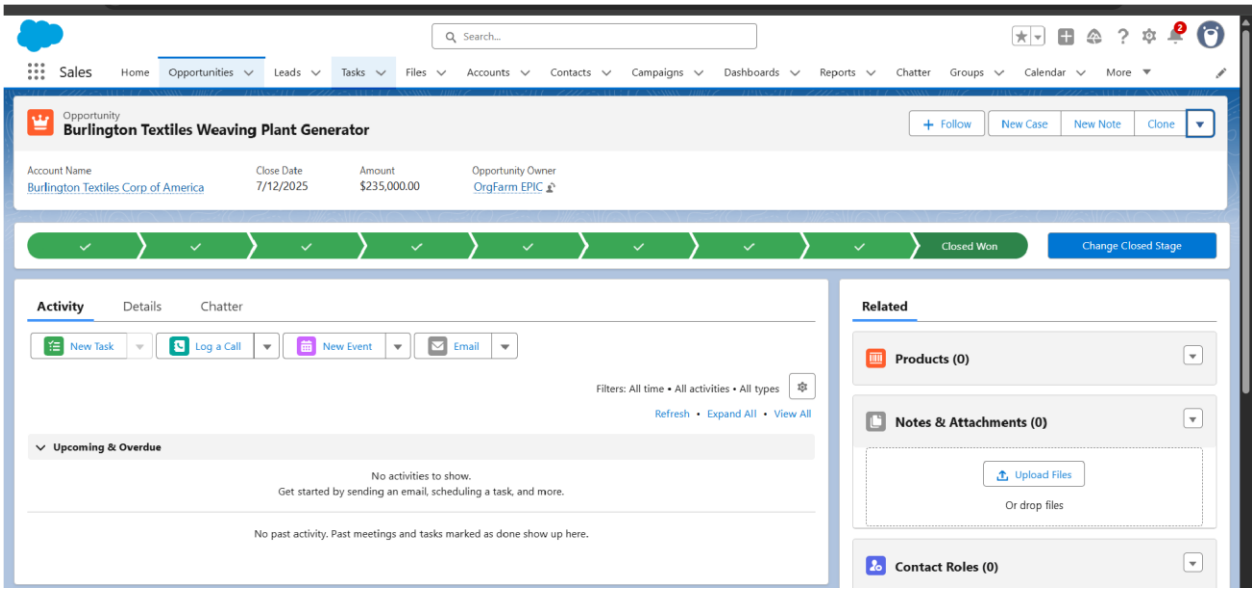
- Developed Lightning App with relevant tabs


orgfarm-5dbd278a84-dev-ed.develop.lightning.force.com/visualEditor/appBuilder.app?id=02ugL000006ngkDQAQ&retUrl=https%3A%2F%2Forgfarm-5dbd278a84-dev-ed.develop.lightning.force....
Lightning App Builder
App Settings
Pages
Medical Inventory Management
Help
App Settings
App Details & Branding
App Options
Utility Items (Desktop Only)
Navigation Items
User Profiles
App Details & Branding
Give your Lightning app a name and description. Upload an image and choose the highlight color for its navigation bar.
App Details
App Branding
*App Name
Medical Inventory Management
*Developer Name
Medical_Inventory_Management_System
Description
Enter a description...
Image
Primary Color Hex Value
#0070D2
Clear
Org Theme Options
Use the app's image and color instead of the org's custom theme
App Launcher Preview
Medical Inventory Manage...

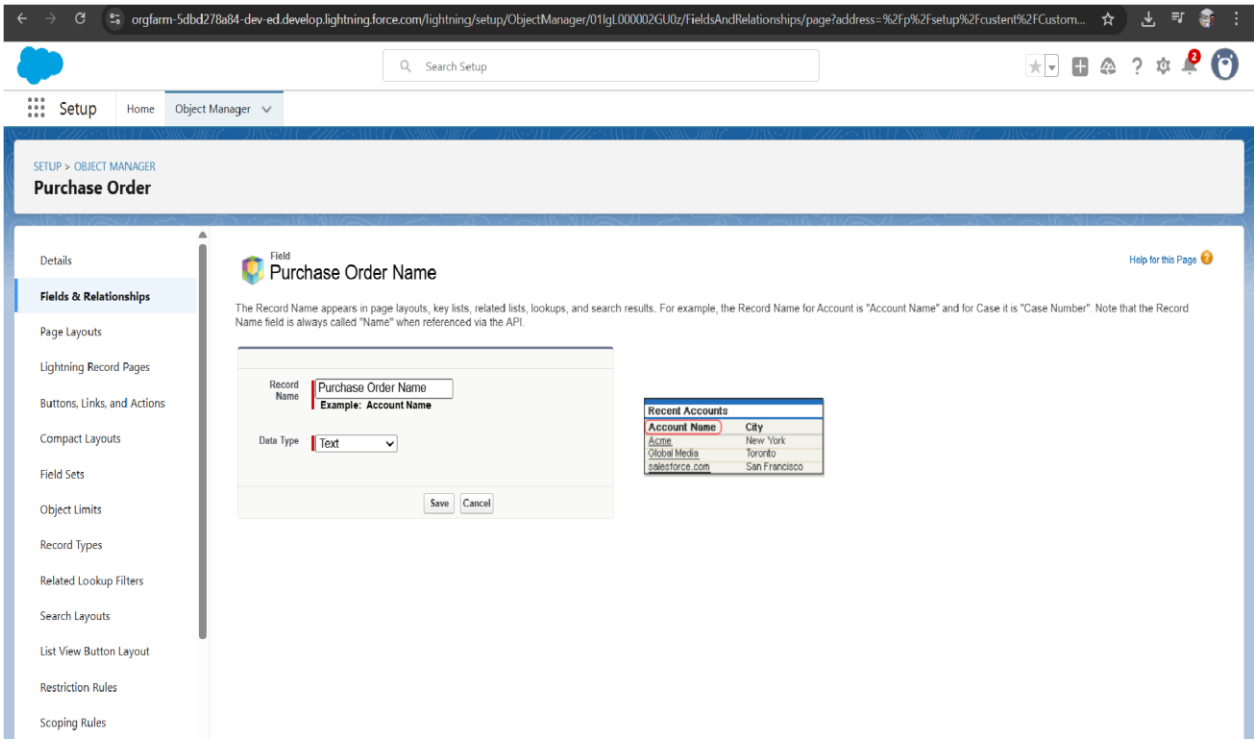

orgfarm-5dbd278a84-dev-ed.develop.lightning.force.com/visualEditor/appBuilder.app?id=02ugL000006ngkDQAQ&retUrl=https%3A%2F%2Forgfarm-5dbd278a84-dev-ed.develop.lightning.force....
Lightning App Builder
App Settings
Pages
Medical Inventory Management
Help
App Settings
App Details & Branding
App Options
Utility Items (Desktop Only)
Navigation Items
User Profiles
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
Create
Type to filter list...
Accounts
Activation Targets
Activations
All Sites
Alternative Payment Methods
Analytics
App Launcher
Appointment Categories
Appointment Invitations
Approval Requests
Approval Submission Details
Approval Submissions
Selected Items
Products
Purchase Orders
Order Items
Inventory Transactions
Supplier objects

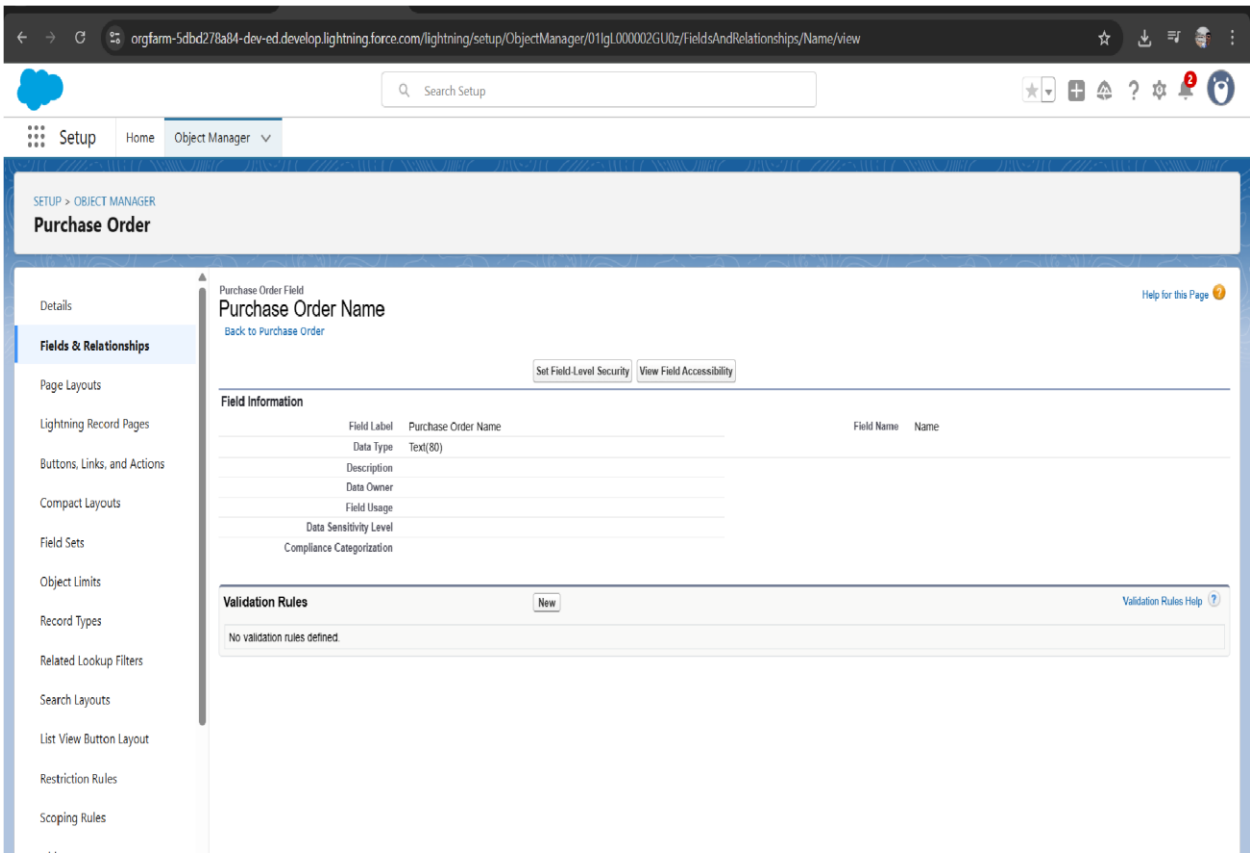


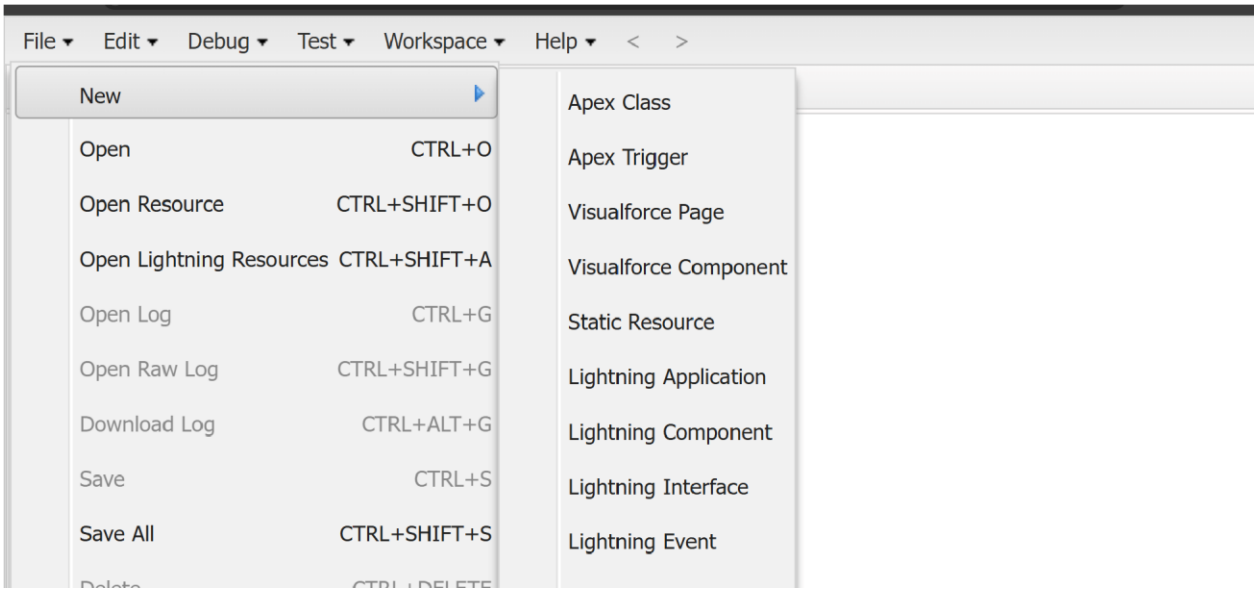
- Automates the calculation and tracking of monthly rent for medical equipment or facilities, ensuring timely reminders and accurate records.



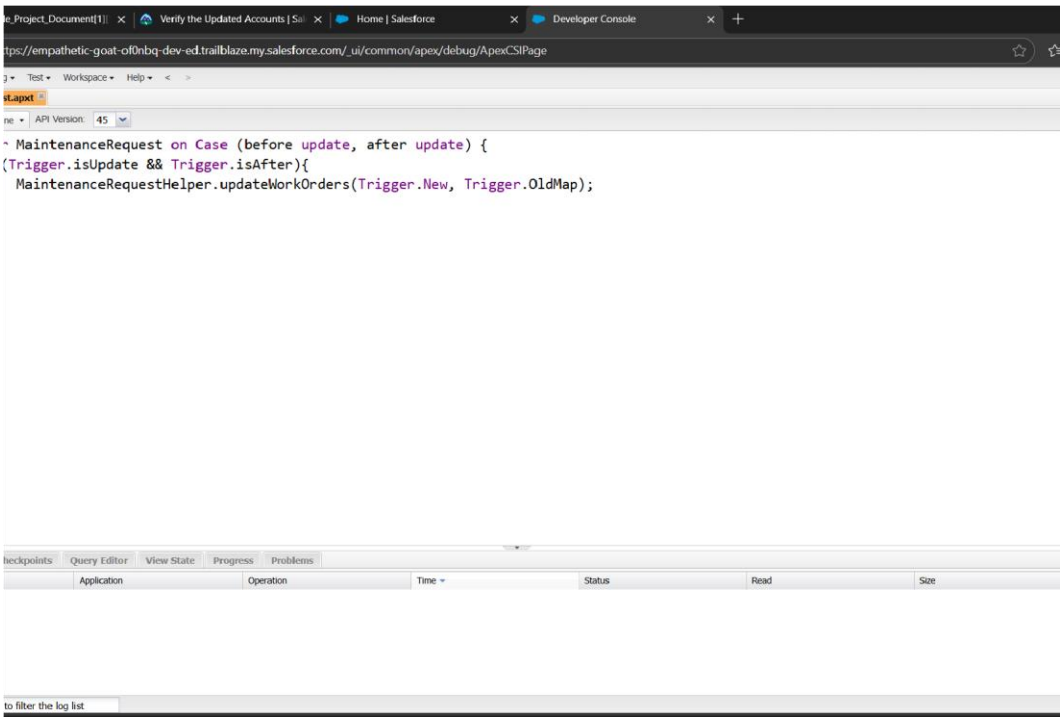
- Ensure that a lease record is valid only if the leased medical equipment has not expired (i.e., medicine/equipment expiry date is later than lease end date).



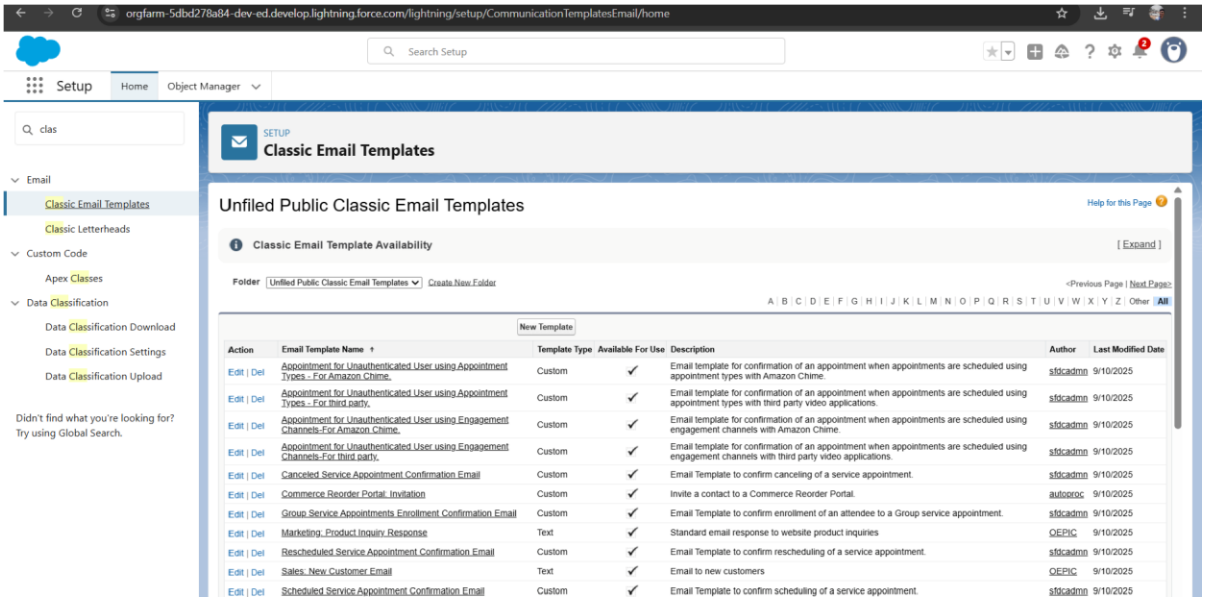


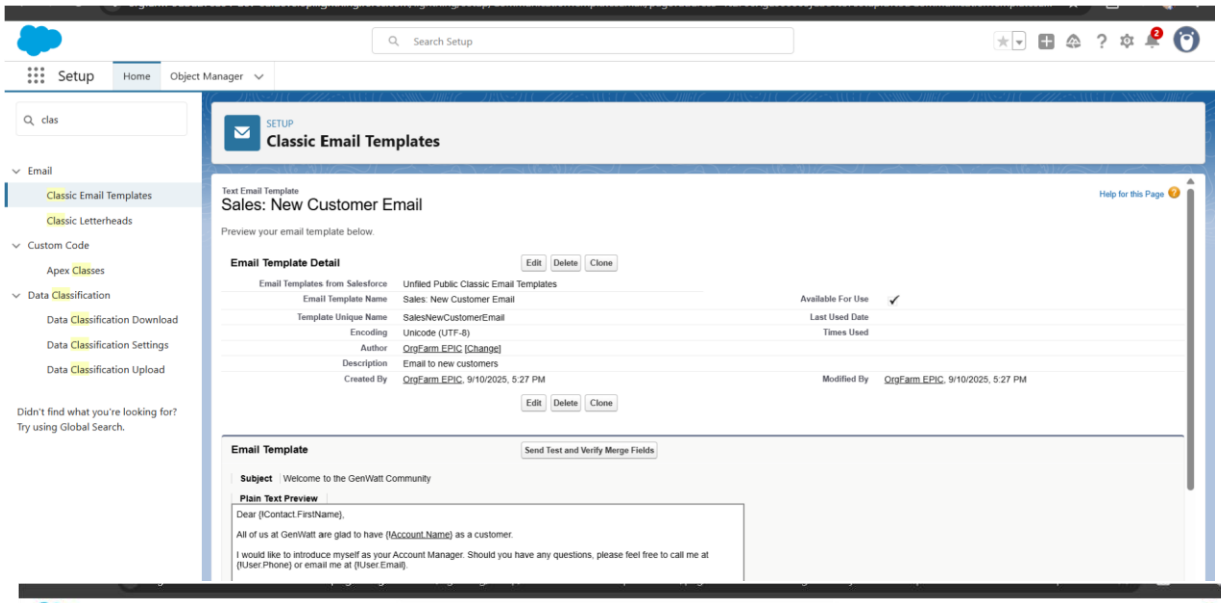


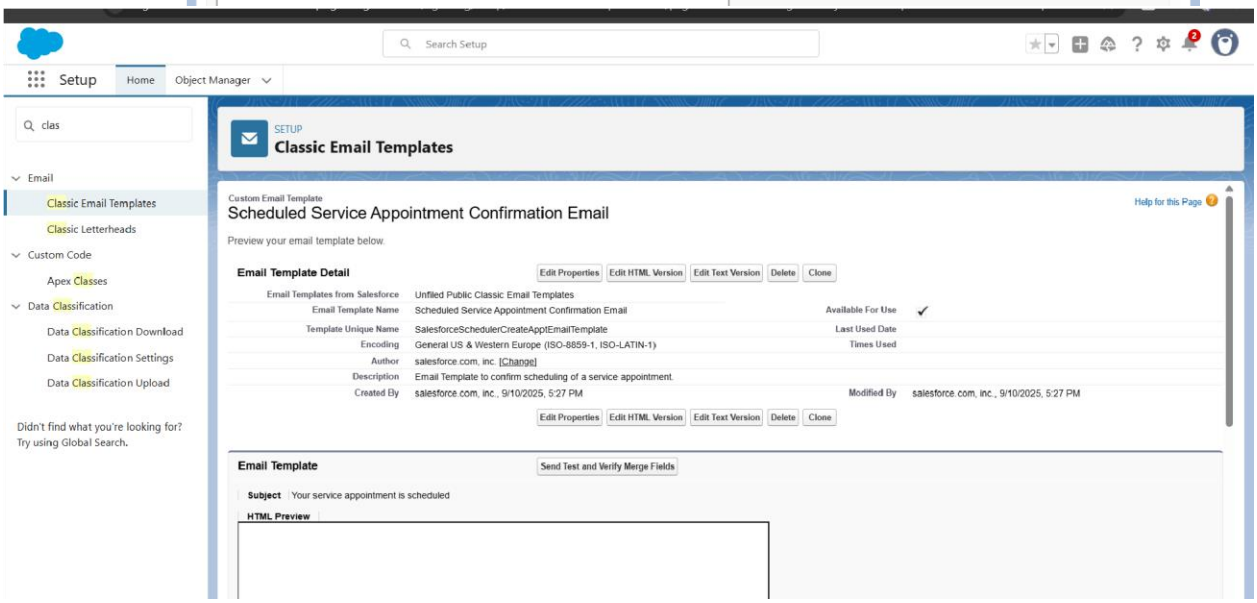
- Scheduled monthly reminder emails using Apex class

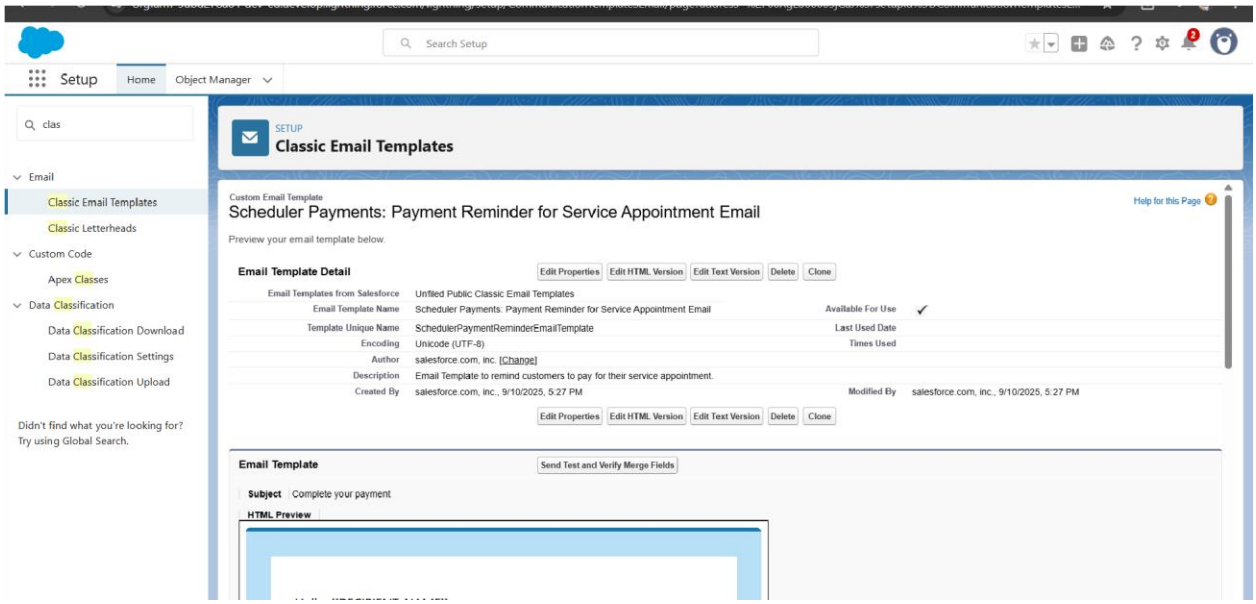


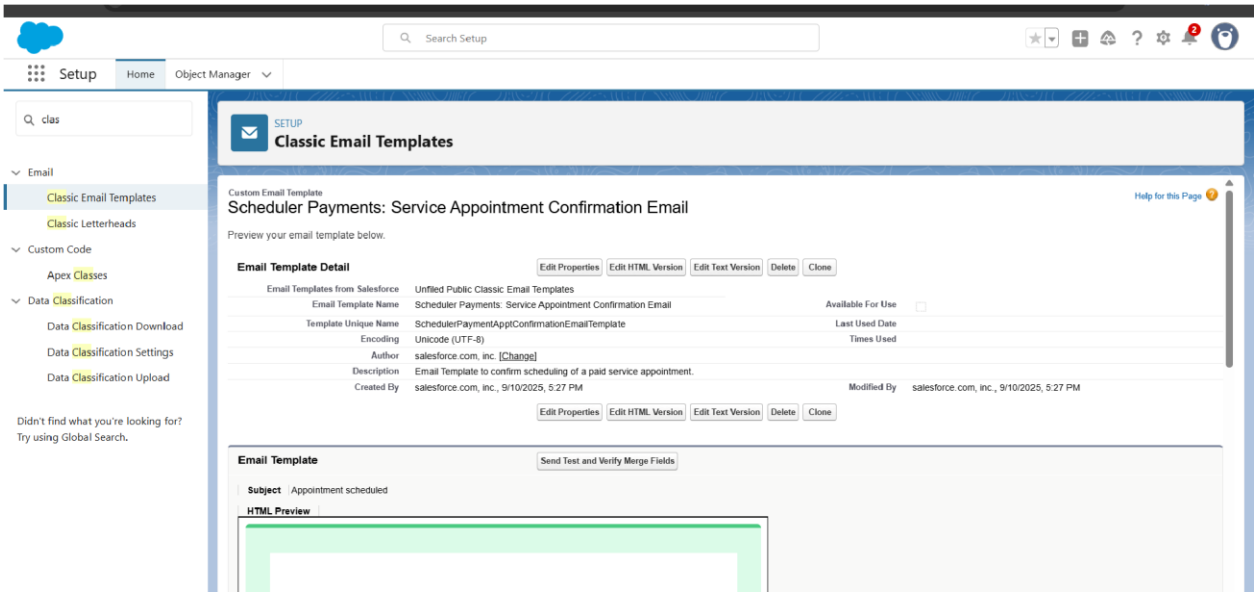
- Built and tested email templates for Product,Purchase Name,Order Item and Supplier.




Search Setup
Setup
Home
Object Manager
clas
Email
Classic Email Templates
Classic Letterheads
Custom Code
Apex Classes
Data Classification
Data Classification Download
Data Classification Settings
Data Classification Upload
Didn't find what you're looking for?
Try using Global Search.
SETUP
Classic Email Templates
Text Email Template
Sales: New Customer Email
Help for this Page
Preview your email template below.
Email Template Detail
Edit
Delete
Clone
Email Templates from Salesforce
Unfiled Public Classic Email Templates
Email Template Name
Sales: New Customer Email
Available For Use
Template Unique Name
SalesNewCustomerEmail
Last Used Date
Encoding
Unicode (UTF-8)
Times Used
Author
OrgFarm EPIC [Change]
Description
Email to new customers
Created By
OrgFarm EPIC, 9/10/2025, 5:27 PM
Modified By
OrgFarm EPIC, 9/10/2025, 5:27 PM
Email Template
Send Test and Verify Merge Fields
Subject
Welcome to the GenWatt Community
Plain Text Preview
Dear {!Contact.FirstName},
All of us at GenWatt are glad to have {!Account.Name} as a customer.
I would like to introduce myself as your Account Manager. Should you have any questions, please feel free to call me at {!User.Phone} or email me at {!User.Email}.

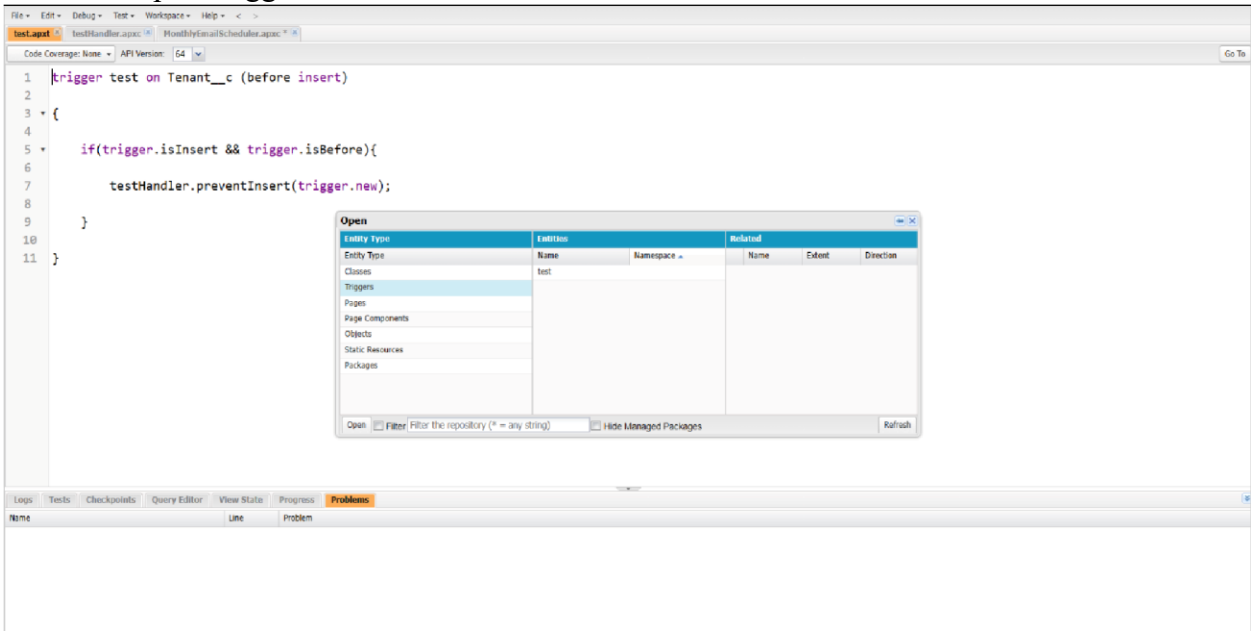

Search Setup
Setup
Home
Object Manager
clas
Email
Classic Email Templates
Classic Letterheads
Custom Code
Apex Classes
Data Classification
Data Classification Download
Data Classification Settings
Data Classification Upload
Didn't find what you're looking for?
Try using Global Search.
SETUP
Classic Email Templates
Custom Email Template
Scheduled Service Appointment Confirmation Email
Help for this Page
Preview your email template below.
Email Template Detail
Edit Properties
Edit HTML Version
Edit Text Version
Delete
Clone
Email Templates from Salesforce
Unfiled Public Classic Email Templates
Email Template Name
Scheduled Service Appointment Confirmation Email
Available For Use
Template Unique Name
SalesforceSchedulerCreateApptEmailTemplate
Last Used Date
Encoding
General US & Western Europe (ISO-8859-1, ISO-LATIN-1)
Times Used
Author
salesforce.com, inc. [Change]
Description
Email Template to confirm scheduling of a service appointment.
Created By
salesforce.com, inc., 9/10/2025, 5:27 PM
Modified By
salesforce.com, inc., 9/10/2025, 5:27 PM
Email Template
Send Test and Verify Merge Fields
Subject
Your service appointment is scheduled
HTML Preview


Search Setup
Setup
Home
Object Manager
clas
Email
Classic Email Templates
Classic Letterheads
Custom Code
Apex Classes
Data Classification
Data Classification Download
Data Classification Settings
Data Classification Upload
Didn't find what you're looking for?
Try using Global Search.
SETUP
Classic Email Templates
Custom Email Template
Scheduler Payments: Payment Reminder for Service Appointment Email
Help for this Page
Preview your email template below.
Email Template Detail
Edit Properties
Edit HTML Version
Edit Text Version
Delete
Clone
Email Templates from Salesforce
Unfiled Public Classic Email Templates
Email Template Name
Scheduler Payments: Payment Reminder for Service Appointment Email
Available For Use
Template Unique Name
SchedulerPaymentReminderEmailTemplate
Last Used Date
Encoding
Unicode (UTF-8)
Times Used
Author
salesforce.com, inc. [Change]
Description
Email Template to remind customers to pay for their service appointment.
Created By
salesforce.com, inc., 9/10/2025, 5:27 PM
Modified By
salesforce.com, inc., 9/10/2025, 5:27 PM
Email Template
Send Test and Verify Merge Fields
Subject
Complete your payment
HTML Preview

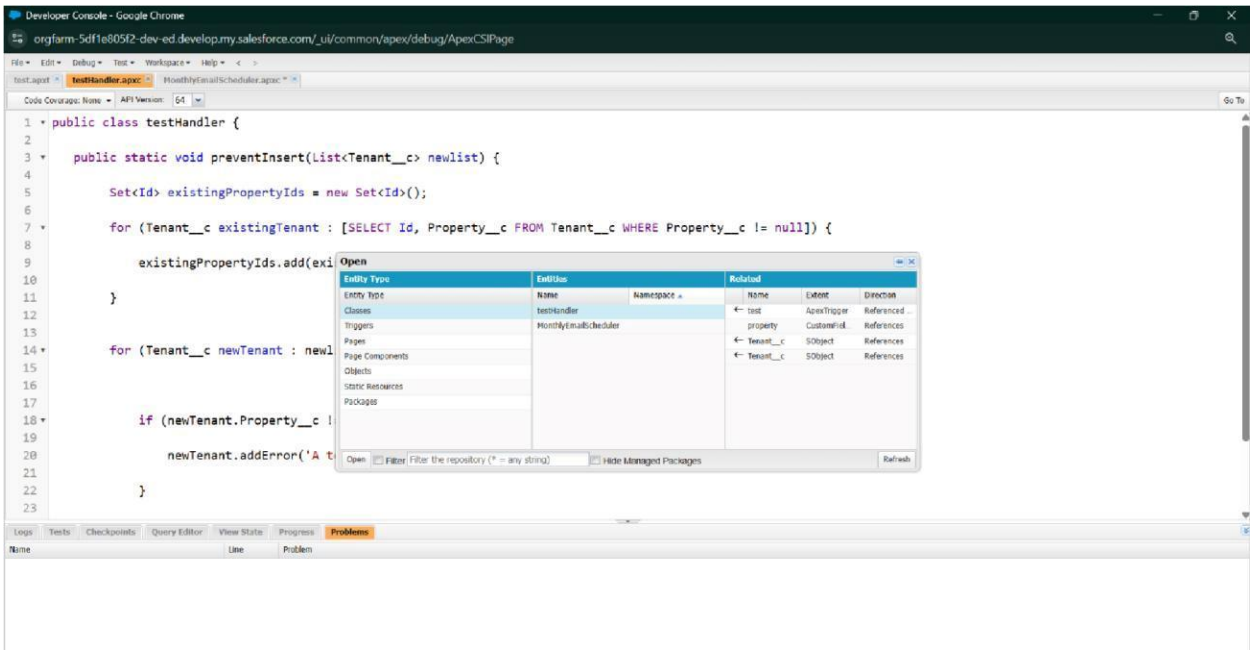


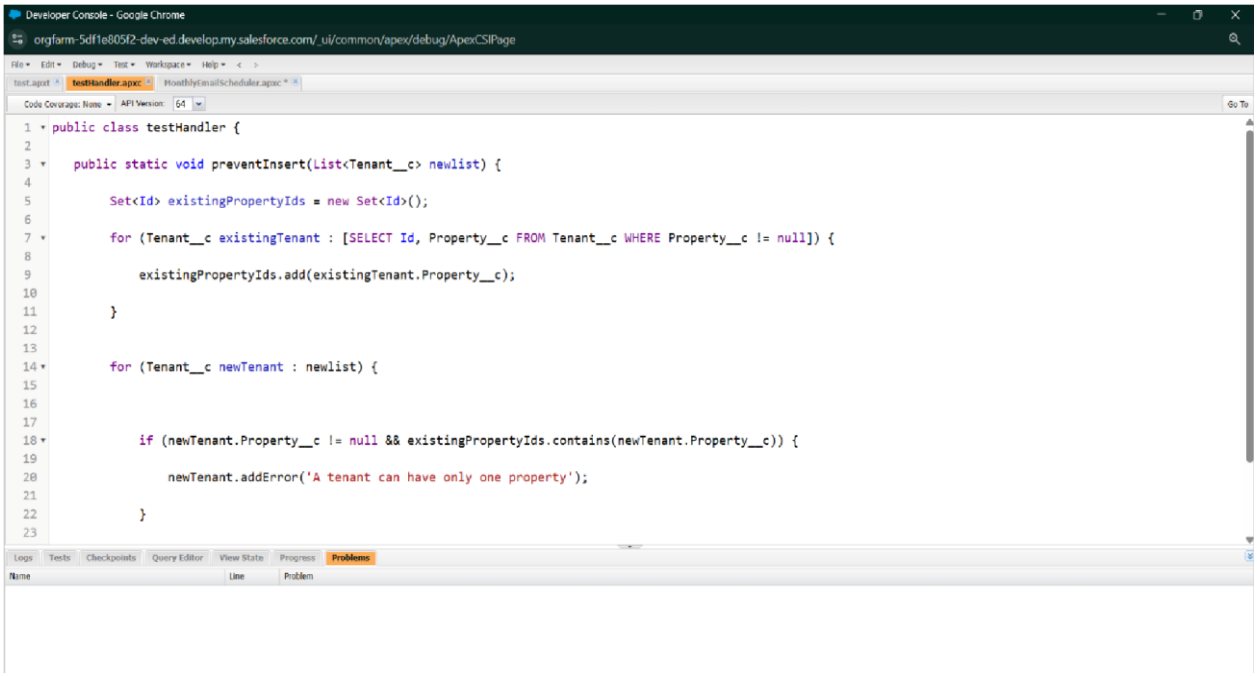
- Apex Trigger

Create an Apex Trigger

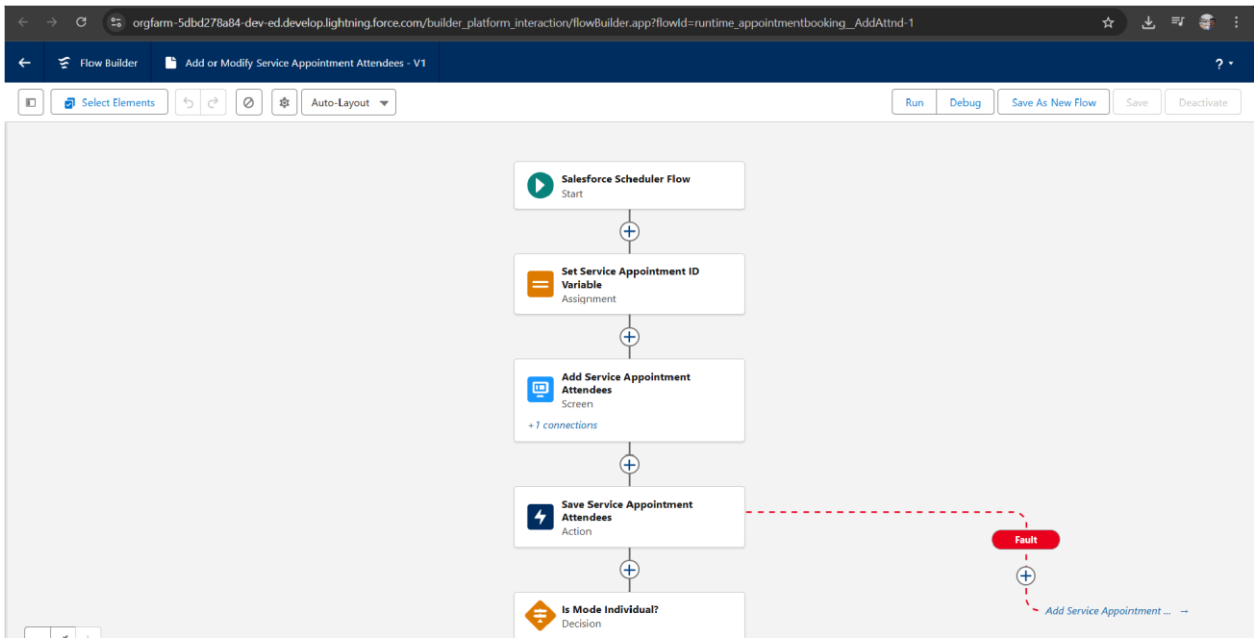


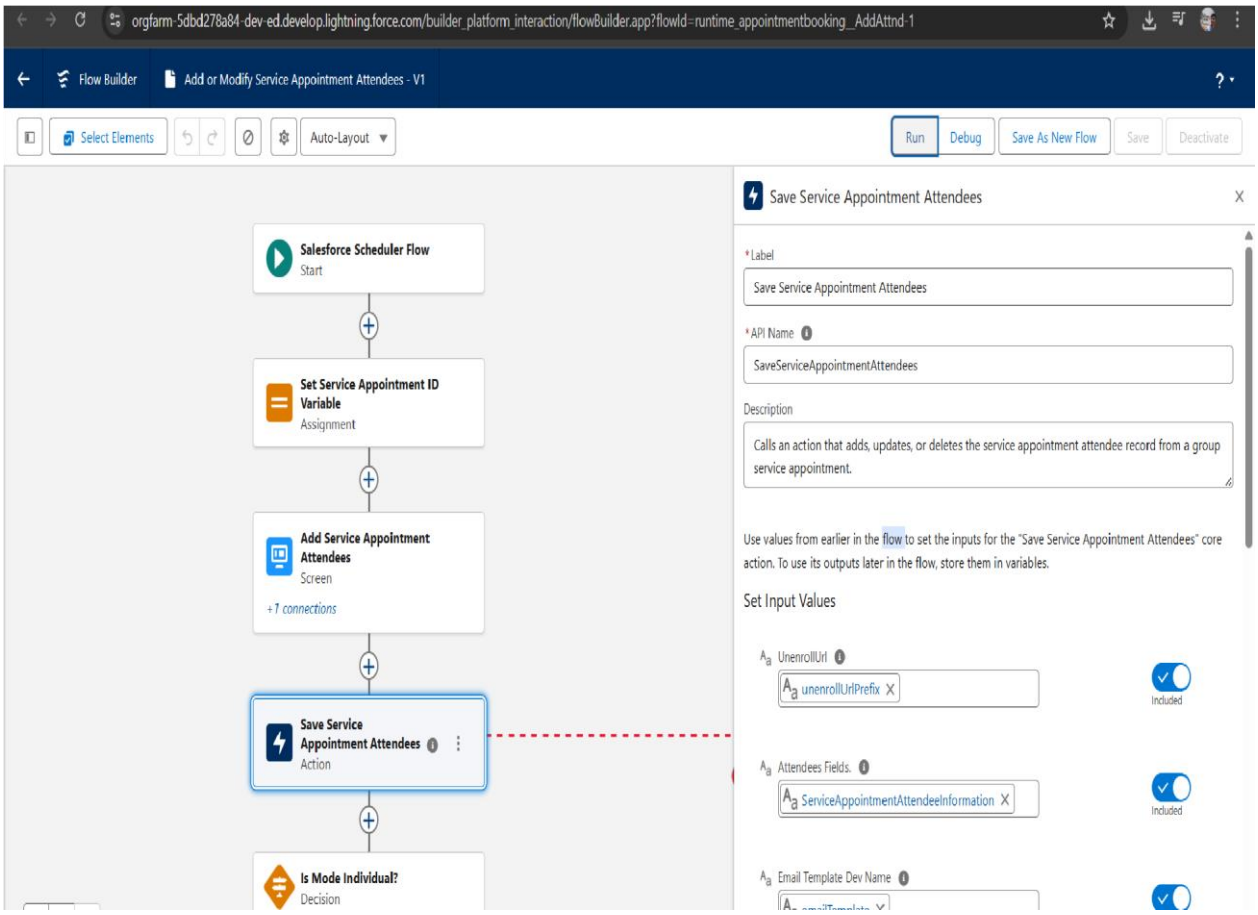
Create an Apex Handler class


Developer Console - Google Chrome
orgfarm-5df1e805f2-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage
public class testHandler {
public static void preventInsert(List<Tenant__c> newlist) {
Set<Id> existingPropertyIds = new Set<Id>();
for (Tenant__c existingTenant : [SELECT Id, Property__c FROM Tenant__c WHERE Property__c != null]) {
existingPropertyIds.add(exi
}
for (Tenant__c newTenant : newl
if (newTenant.Property__c !
newTenant.addError('A t
}
Open
Entity Type
Classes
Triggers
Pages
Page Components
Objects
Static Resources
Packages
Entities
Name
Namespace
testHandler
MonthlyEmailScheduler
Related
Name
Extent
Direction
test
property
Tenant__c
Tenant__c
Hide Managed Packages
Refresh
Logs
Tests
Checkpoints
Query Editor
View State
Progress
Problems
Name
Line
Problem

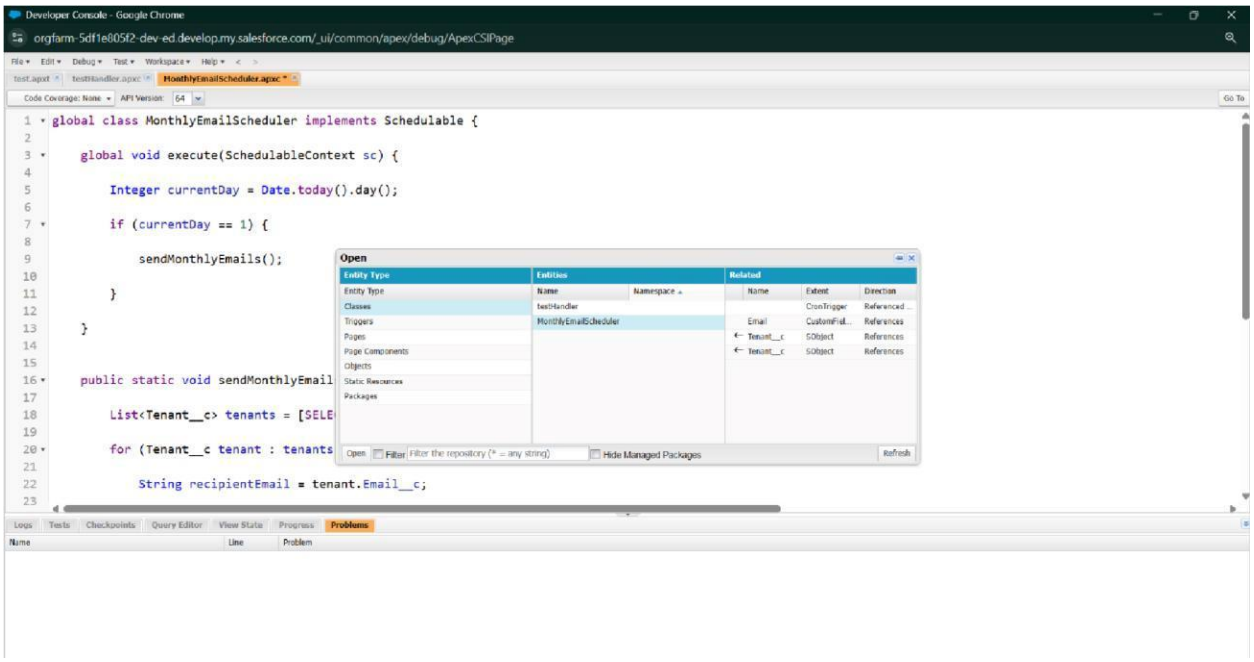

Developer Console - Google Chrome
orgfarm-5df1e805f2-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage
public class testHandler {
public static void preventInsert(List<Tenant__c> newlist) {
Set<Id> existingPropertyIds = new Set<Id>();
for (Tenant__c existingTenant : [SELECT Id, Property__c FROM Tenant__c WHERE Property__c != null]) {
existingPropertyIds.add(existingTenant.Property__c);
}
for (Tenant__c newTenant : newlist) {
if (newTenant.Property__c != null && existingPropertyIds.contains(newTenant.Property__c)) {
newTenant.addError('A tenant can have only one property');
}
Logs
Tests
Checkpoints
Query Editor
View State
Progress
Problems
Name
Line
Problem

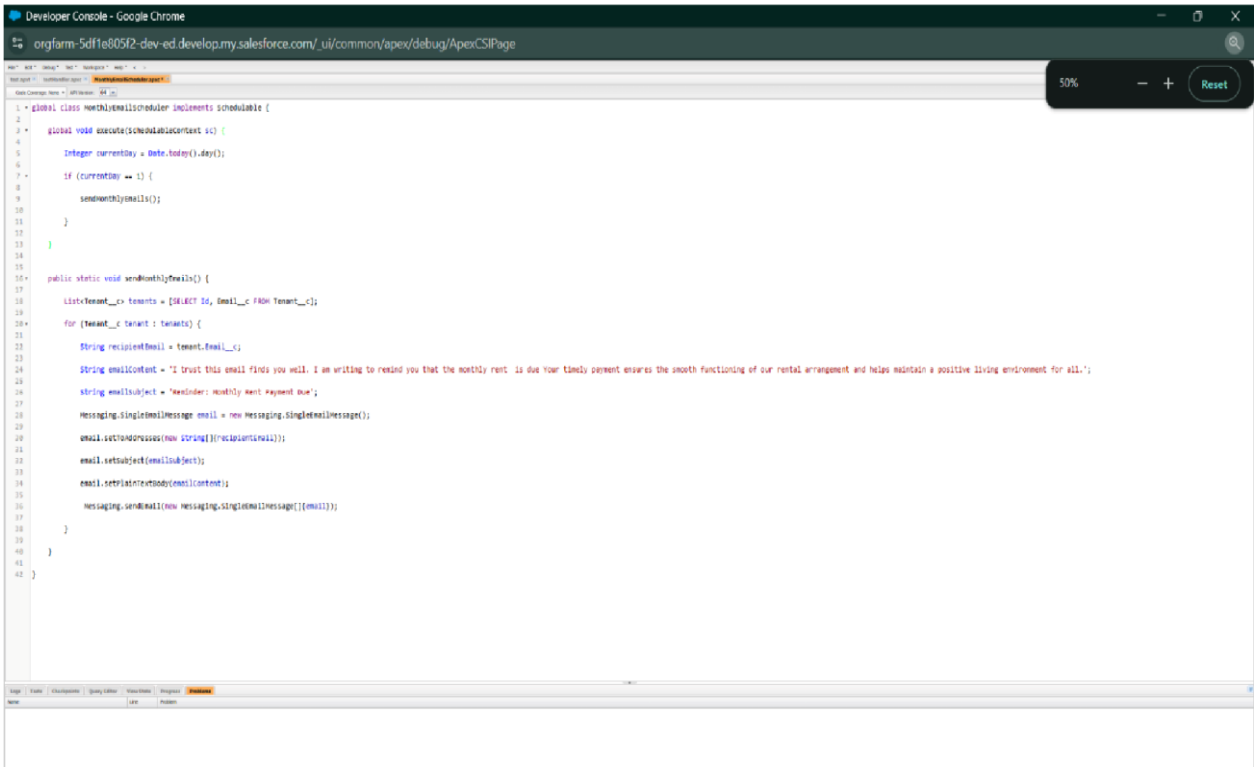
- FLOWS





- Schedule class: Create an Apex Class





ADVANTAGES

.

Accurate Tracking – Monitors medicines, suppliers, leases, and patients in real time.

Automation – Flows handle monthly rent, payment alerts, and stock updates automatically.

Error Prevention – Validation rules ensure data integrity (e.g., lease dates, rent values).

Reduced Wastage – Tracks expiry dates to avoid expired stock usage.

Improved Efficiency – Saves time by reducing manual record-keeping and errors.

DISADVANTAGES

Implementation Cost – Requires investment in Salesforce setup, licenses, or tools.

Training Need – Users must learn how to operate the system properly.

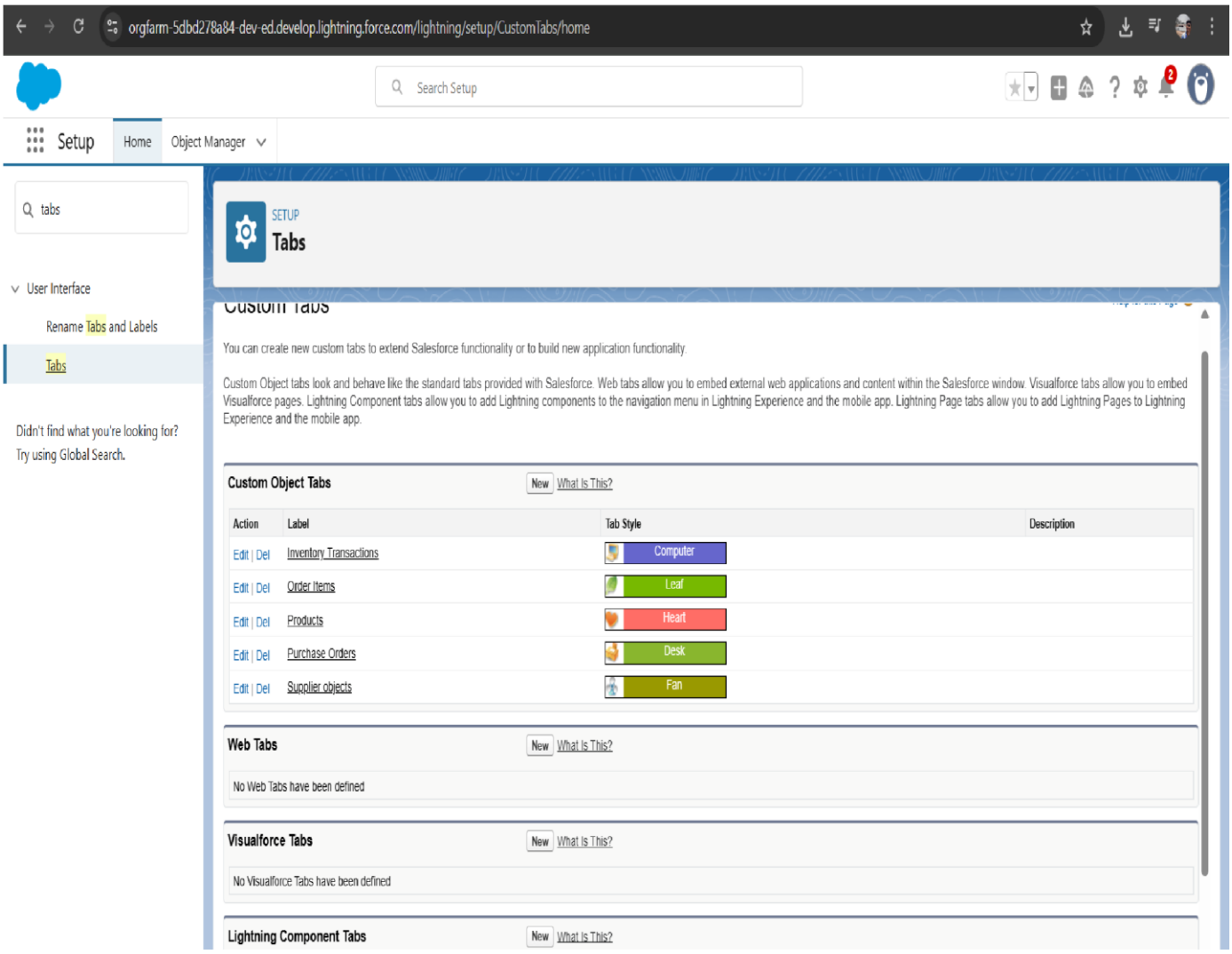
Customization Complexity – Creating flows, validation rules, and roll-ups may need admin or developer expertise.

Data Dependency – System is only as good as the accuracy of the data entered.

Maintenance – Requires regular updates, monitoring, and troubleshooting.

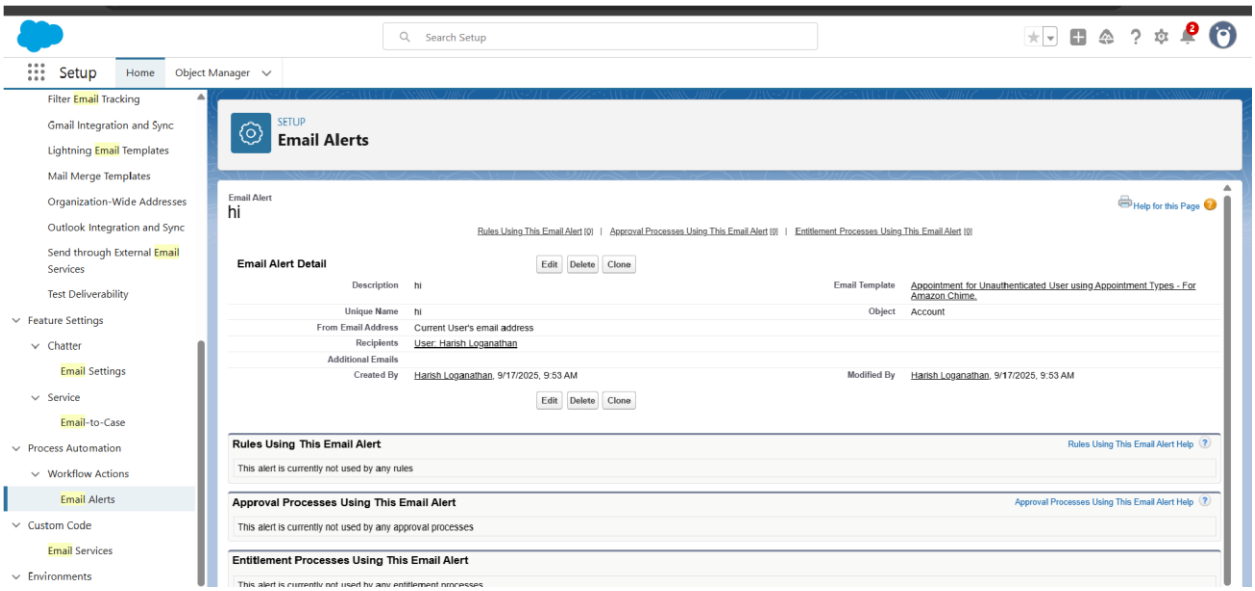
RESULTS

Output Screenshots



Tabs for Product,Purchase Order, Order ID,Supplier

Email alerts



CONCLUSION

The Medical Inventory Management system ensures efficient handling of medicines, suppliers, purchase orders, patients, and lease records. By implementing automation through flows (monthly rent tracking and payment alerts) and enforcing validation rules on the Lease object (such as preventing expired equipment leases), the system enhances accuracy, safety, and compliance. Overall, it reduces manual errors, improves inventory control, and supports timely availability of critical medical supplies.

APPENDIX

- **Source Code:** Provided in Apex Classes and Triggers

```
public with sharing class MaintenanceRequestHelper {     public static void
updateworkOrders(List<Case> updWorkOrders, Map<Id,Case> nonUpdCaseMap)
{
        Set<Id> validIds = new Set<Id>();
```

```
    For (Case c : updWorkOrders){            if (nonUpdCaseMap.get(c.Id).Status
!= 'Closed' && c.Status == 'Closed'){              if (c.Type == 'Repair' || c.Type ==
'Routine Maintenance'){                  validIds.add(c.Id);



            }
        }
    }

    if (!validIds.isEmpty()){
        List<Case> newCases = new List<Case>();
        Map<Id,Case> closedCasesM = new Map<Id,Case>([SELECT Id, Vehicle__c,
Equipment__c, Equipment__r.Maintenance_Cycle__c,(SELECT Id,Equipment__c,Quantity__c
FROM Equipment_Maintenance_Items__r)
                                        FROM Case WHERE Id IN :validIds]);
        Map<Id,Decimal> maintenanceCycles = new Map<ID,Decimal>();
        AggregateResult[] results = [SELECT Maintenance_Request__c,
MIN(Equipment__r.Maintenance_Cycle__c)cycle FROM Equipment_Maintenance_Item__c
WHERE Maintenance_Request__c IN :ValidIds GROUP BY Maintenance_Request__c];

    for (AggregateResult ar : results){              maintenanceCycles.put((Id)
ar.get('Maintenance_Request__c'), (Decimal) ar.get('cycle'));
    }

        for(Case cc : closedCasesM.values()){
            Case nc = new Case (
                ParentId = cc.Id,
            Status = 'New',
                Subject = 'Routine Maintenance',
                Type = 'Routine Maintenance',
                Vehicle__c = cc.Vehicle__c,
                Equipment__c =cc.Equipment__c,
Origin = 'Web',
                Date_Reported__c = Date.Today()

            );

            If (maintenanceCycles.containskey(cc.Id)){                  nc.Date_Due__c =
Date.today().addDays((Integer) maintenanceCycles.get(cc.Id));
            }

            newCases.add(nc);
        }
```

```
            insert newCases;

            List<Equipment_Maintenance_Item__c> clonedWPs = new
        List<Equipment_Maintenance_Item__c>();           for (Case nc :
        newCases){              for (Equipment_Maintenance_Item__c wp :
        closedCasesM.get(nc.ParentId).Equipment_Maintenance_Items__r){
        Equipment_Maintenance_Item__c wpClone = wp.clone();
        wpClone.Maintenance_Request__c = nc.Id;
                    ClonedWPs.add(wpClone);

                }
              }
              insert ClonedWPs;
            }
          }
        }
                trigger MaintenanceRequest on Case (before update,
after update) {              if(Trigger.isUpdate && Trigger.isAfter){
         MaintenanceRequestHelper.updateWorkOrders(Trigger.New, Trigger.OldMap);
         }
         }
```