

Garage management system

Collage name : Government arts and science
collage for womens

Collage code : brubm

Team ID : NM2025TMID26092

Team Members

Team leader name : C.Mahima

Email ID: mahimahima4540@gmail.com

Team Member 1 : R.Nivetha

Email ID: boopathynivetha1204@gmail.com

Team Member 2 : T.Sivaranjani

Email ID: theerthantheerthanthee@gmail.com

Team Member 3 : T.Boomika

Email ID: suganthithamotharan@gmail.com

1.INTRODUCTION

1.1 Project Overview

This project manages customer, vehicle, and service details in a centralized Salesforce system. It helps track service records, billing, and feedback for each garage visit. Automation features reduce manual work and improve customer experience. Reports and dashboards provide insights into garage performance and sales.

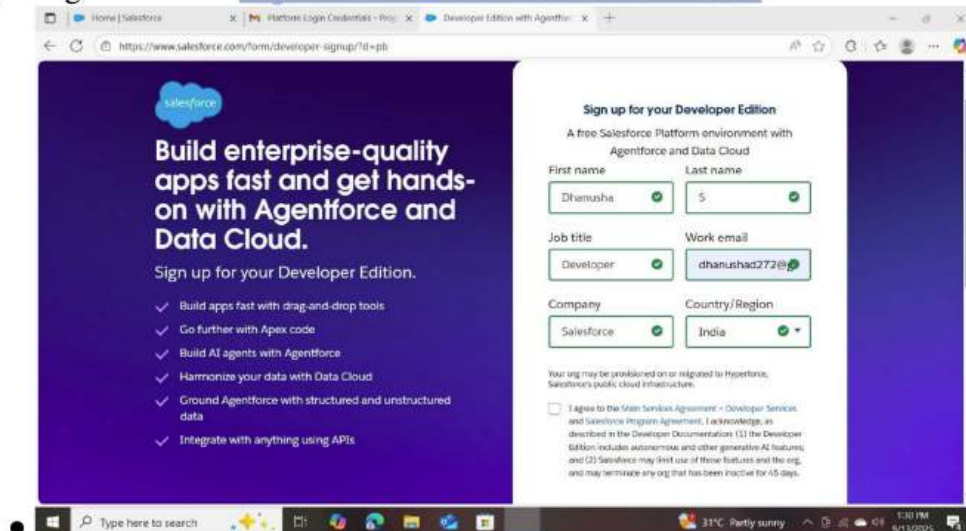
1.2 Purpose

To maintain and organize customer, vehicle, and service records efficiently. To simplify billing, payment, and feedback collection through automation. 3. To provide reports and insights for improving garage operations and customer satisfaction.

1. DEVELOPMENT PHASE

➤ Creating Developer Account:

By using this URL - <https://naanmudhalvan.smartinternz.com>

A screenshot of a web browser displaying the Salesforce Developer Edition sign-up page. The page has a dark blue header with the Salesforce logo and the text "Build enterprise-quality apps fast and get hands-on with Agentforce and Data Cloud." Below this, it says "Sign up for your Developer Edition." and lists several benefits with checkmarks: "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", and "Integrate with anything using APIs". On the right side, there is a form titled "Sign up for your Developer Edition" with the subtitle "A free Salesforce Platform environment with Agentforce and Data Cloud". The form fields include: "First name" (Dhanusha), "Last name" (S), "Job title" (Developer), "Work email" (dhanushad272@), "Company" (Salesforce), and "Country/Region" (India). There is a checkbox for "I agree to the Salesforce Terms of Service" and a note about the trial period: "Your org may be provisioned on or migrated to Hyperforce, Salesforce's public cloud infrastructure." The browser's address bar shows the URL "https://www.salesforce.com/form/developer/signup?id=pb". The Windows taskbar is visible at the bottom with the date and time "1:10 PM 10/12/2025".

- Created **Objects**: Customer Details, Appointments, Service records, Billing details and feedback.

Setup

Home

Object Manager

Search Setup

Customer Details

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

Description

API Name

Customer_Details_c

Custom

✓

Singular Label

Customer Details

Plural Label

Customer Details

Enable Reports

✓

Track Activities

✓

Track Field History

✓

Deployment Status

Deployed

Help Settings

Standard salesforce.com Help Window

Edit

Delete

Setup

Home

Object Manager

Search Setup

Appointment

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

Description

API Name

Appointment_c

Custom

✓

Singular Label

Appointment

Plural Label

Appointments

Enable Reports

✓

Track Activities

✓

Track Field History

✓

Deployment Status

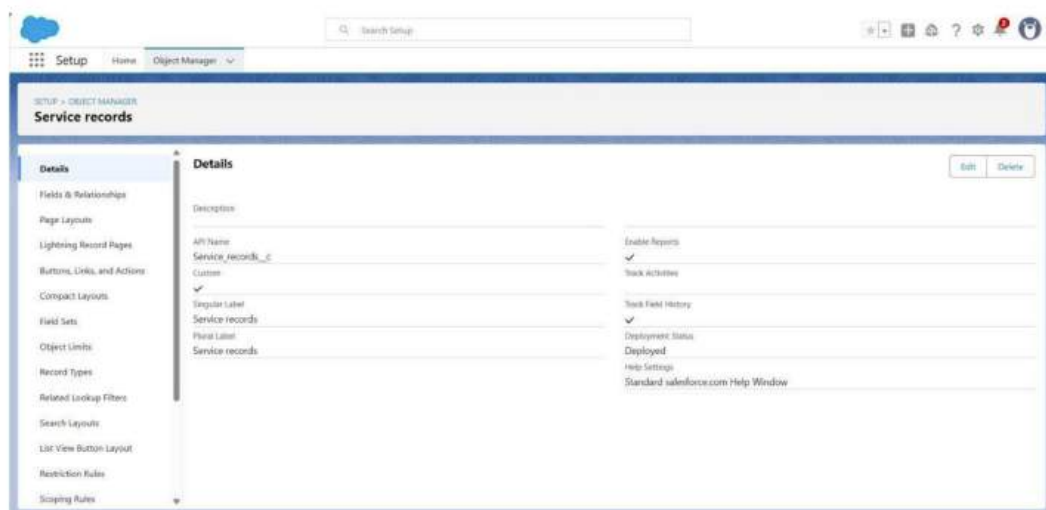
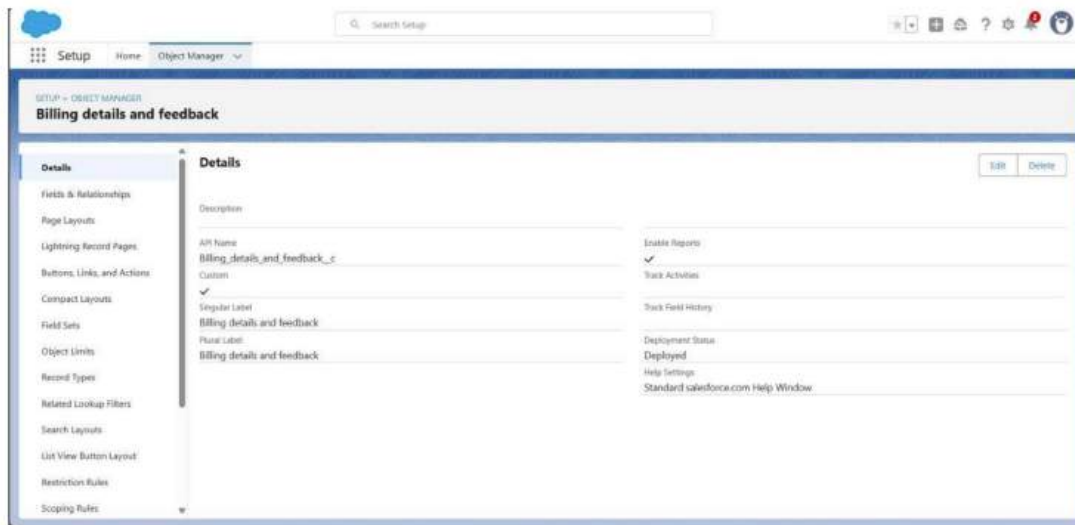
Deployed

Help Settings

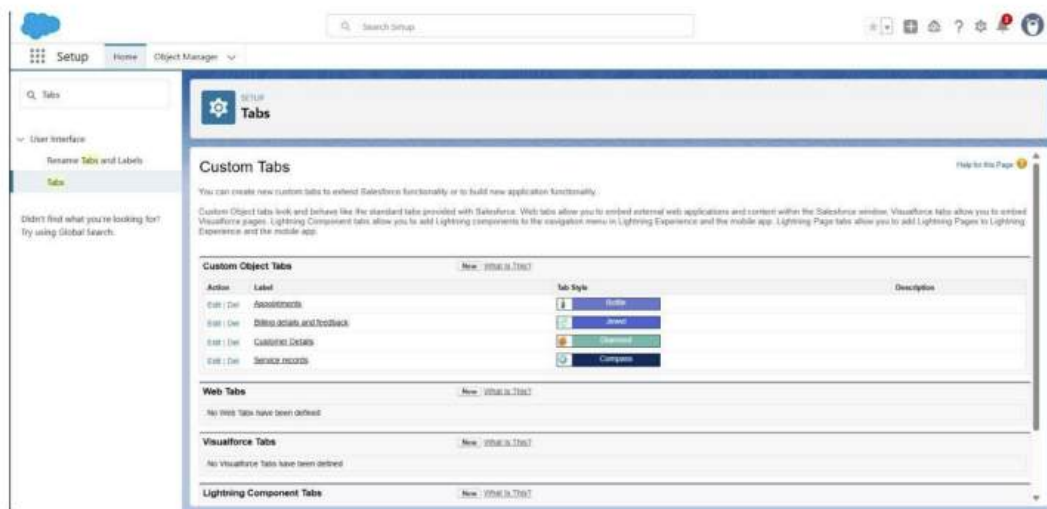
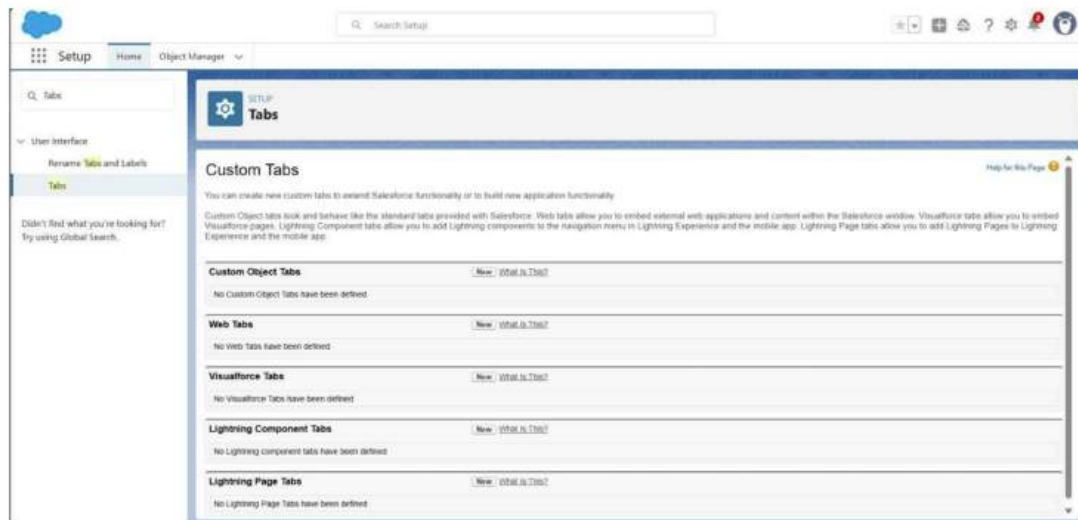
Standard salesforce.com Help Window

Edit

Delete



➤ Created Custom **Tabs** for all the Objects that is been Created



➤ Developed **Lightning App** with relevant tabs

New Lightning App

App Options

Navigation and Form Factor ●

Navigation Style

☒ Standard navigation

☐ Console navigation

Supported Form factors

☒ Desktop and phone

☐ Desktop

☐ Phone

Setup and Personalization ●

Setup Experience

☒ Setup (full set of Setup options)

☐ Service Setup

☐ Data Cloud Setup

App Personalization Settings

☐ Disable end user personalization of new items in this app

☐ Disable temporary tabs for items outside of this app

☐ Use Omni-Channel sidebar

Back

Next

New Lightning App

Utility Items (Desktop Only)


Give your users quick access to productivity tools and add background utility items to your app.

Add Utility Item

Utility Bar Alignment ⓘ

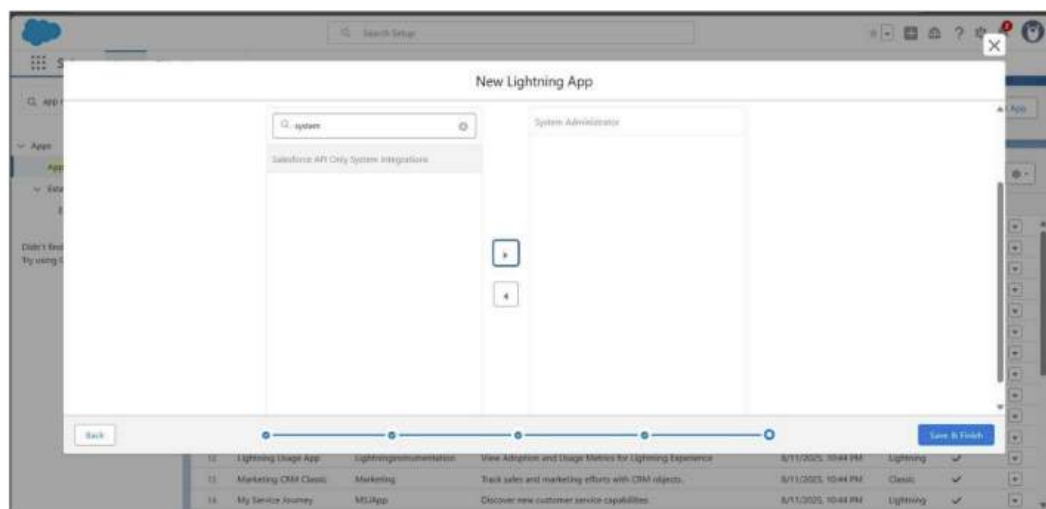
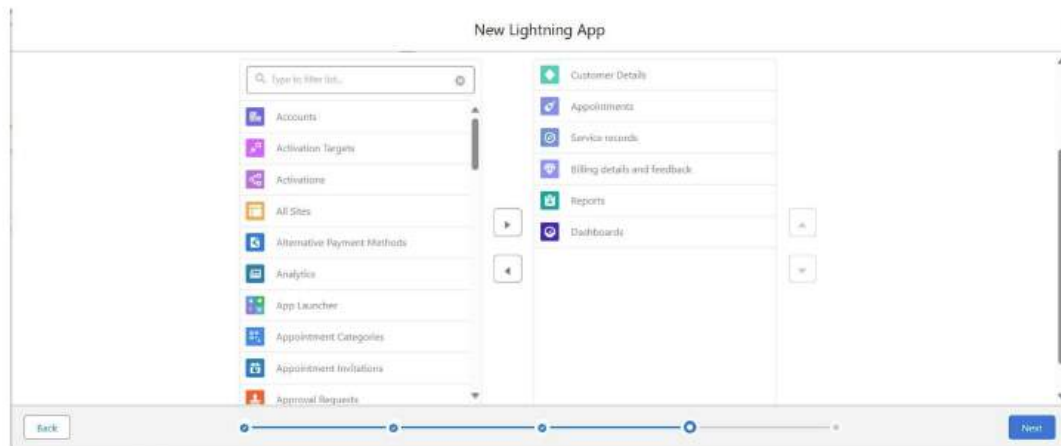
Default

The utility bar is a fixed footer that opens components in docked panels. Available only when the app is viewed in Lightning Experience on a desktop.



Back

Next



- Created the **fields** for: Customer Details Object, Lookup, Checkbox, Date Fields, Currency, Text, Picklist, Formula and Service Records Object

Setup

Home

Object Manager

Search Setup

Setup > OBJECT MANAGER

Customer Details

Details

Fields & Relationships

Quick Find

New

Deleted Fields

Field Dependencies

Set History Tracking

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Created By	CreatedById	Lookup(User)		
Customer Details Name	Name	Text(80)		✓
Email	Email_c	Email		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		✓
Phone number	Phone_number_c	Phone		

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

Setup

Home

Object Manager

Search Setup

Setup > OBJECT MANAGER

Appointment

Details

Fields & Relationships

Quick Find

New

Deleted Fields

Field Dependencies

Set History Tracking

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Appointment Name	Name	Auto Number		✓
Created By	CreatedById	Lookup(User)		
Customer Details	Customer_Details_c	Lookup(Customer Details)		✓
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		✓

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

Setup

Home

Object Manager

Search Setup

Service records

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

Description

API Name

Service_records__c

Custom

✓

Singular Label

Service records

Plural Label

Service records

Enable Reports

✓

Track Activities

Track Field History

✓

Deployment Status

Deployed

Help Settings

Standard salesforce.com Help Window

EditDelete

Setup

Home

Object Manager

Search Setup

Billing details and feedback

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

Fields & Relationships

0 Items. Sorted by Field Label

Quick Find

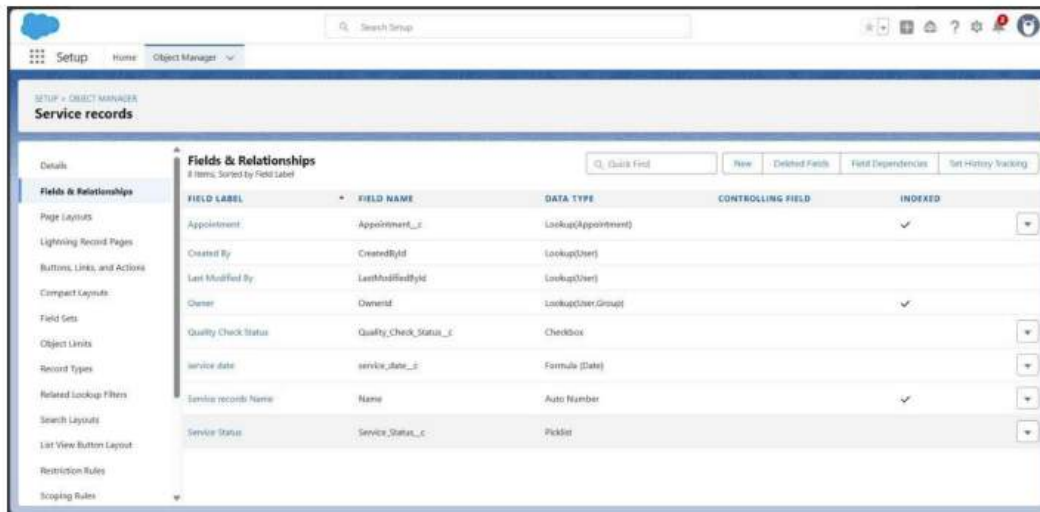
New

Define Fields

Field Dependencies

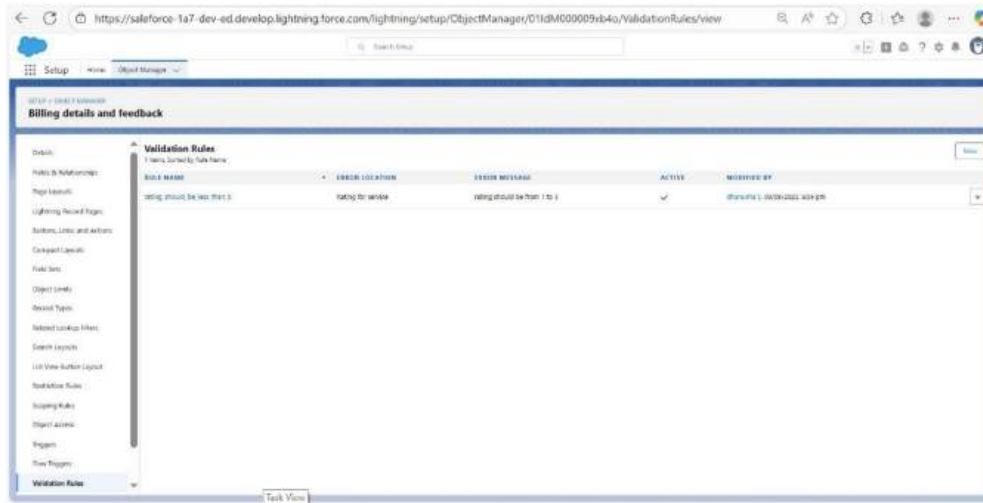
Set History Tracking

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Billing details and feedback Name	Name	Auto Number		✓
Created By	CreatedById	Lookup(User)		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		✓
Payment Paid	Payment_Paid__c	Currency(18, 0)		
Service records	Service_records__c	Lookup(Service records)		✓

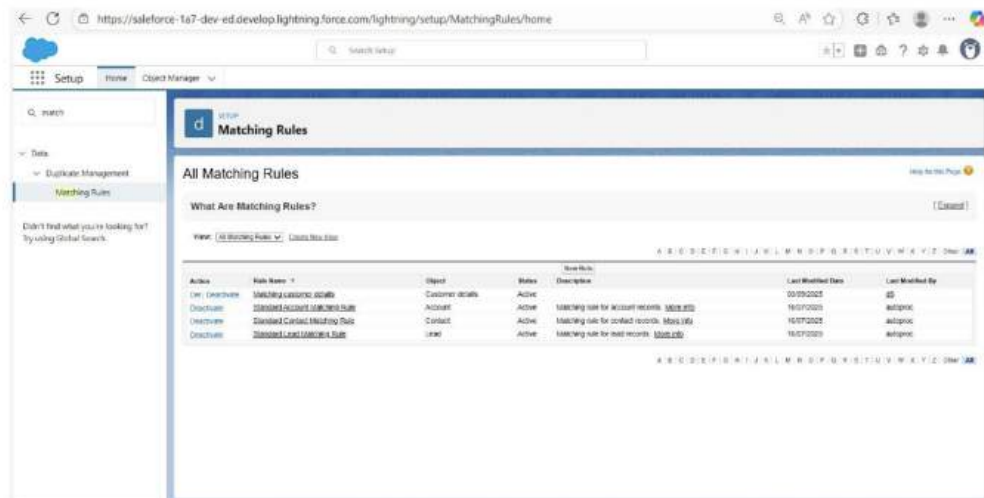


- Created the **Validation rule** for: Appointment object, Billing Details and Feedback objects.





- Create **duplicate rules** to Customer details Object for: Matching and Duplicate.





Setup

Home

Object Manager

Search Setup

Quick Find

Setup Home

Salesforce Go

Service Setup Assistant

Commerce Setup Assistant

Field Service Setup Home (Beta)

Hyperforce Assistant

Release Updates

Salesforce Mobile App

Lightning Usage

Optimizer

Sales Cloud Everywhere

ADMINISTRATION

> Users

> Data

> Email

PLATFORM TOOLS

> Subscription Management

SETUP

Sharing Settings

Note: "Roles and subordinates" includes all users in a role, and the roles below that role.

You can use sharing rules only to grant wider access to data, not to restrict access.

Step 1: Rule Name

Label: Sharing setting

Rule Name: Sharing_setting

Description:

Step 2: Select your rule type

Rule Type: ☒ Based on record owner ☐ Based on criteria

Step 3: Select which records to be shared

Service Resource: owned by members of Roles sales person

Step 4: Select the users to share with

Share with: Roles Manager

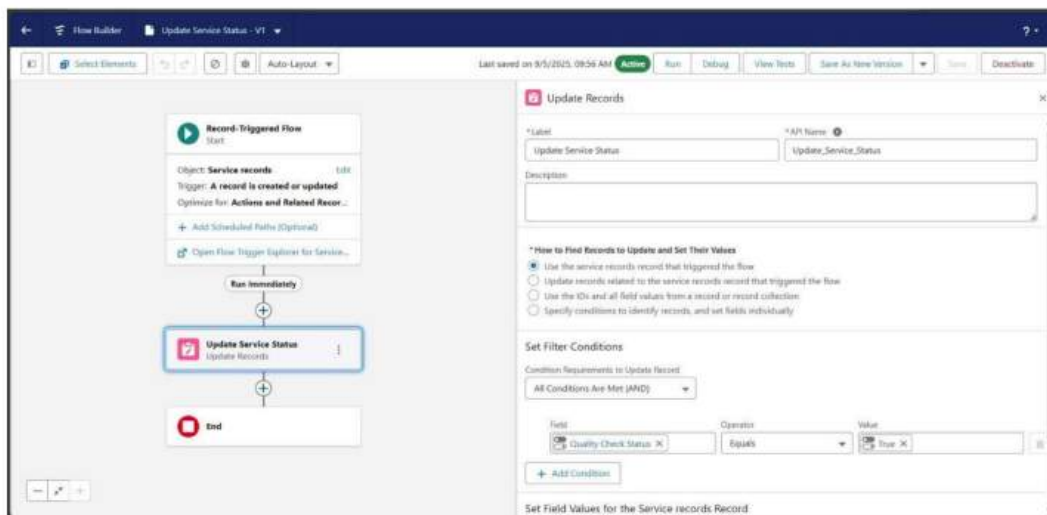
Step 5: Select the level of access for the users

Access Level: Read/Write

Save

Cancel

➤ **Flow** - Billing amount flow.



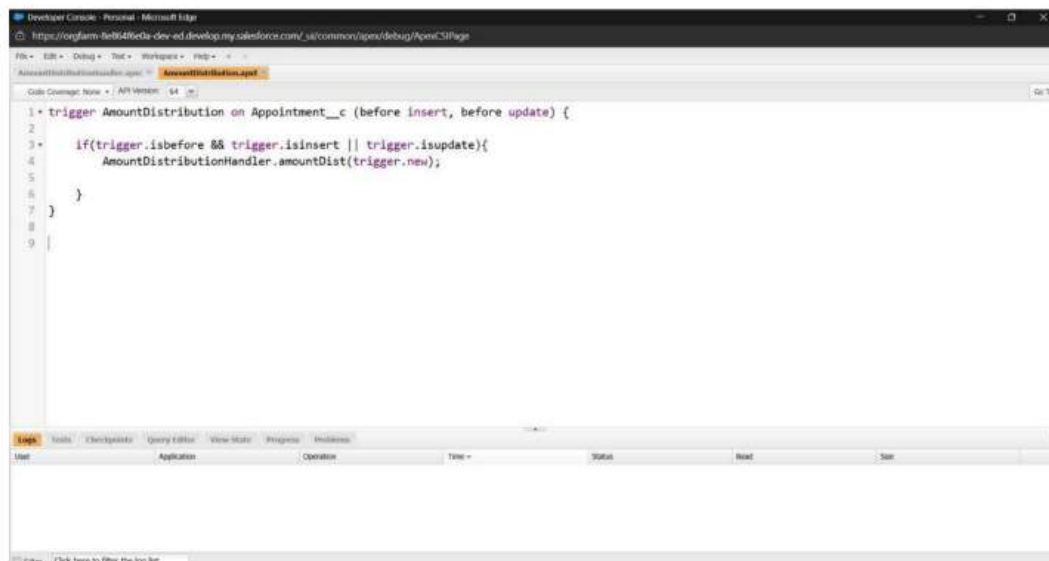
Developed **Apex Triggers**:

Created Apex **Class** called **AmountDistributionHandler**.

```
1 public class AmountDistributionHandler {
2
3     public static void amountDist(list<Appointment__c> listApp){
4
5         list<Service_records__c> serlist = new list<Service_records__c>();
6
7
8
9         for(Appointment__c app : listApp){
10             if(app.Maintenance_service__c == true && app.Repairs__c == true && app.Replacement_Parts__c == true){
11                 app.Service_Amount__c = 10000;
12             }
13
14             else if(app.Maintenance_service__c == true && app.Repairs__c == true){
15                 app.Service_Amount__c = 5000;
16             }
17
18             else if(app.Repairs__c == true && app.Replacement_Parts__c == true){
19                 app.Service_Amount__c = 5000;
20             }
21
22         }
23     }
24 }
```

Created

- Created an **Apex trigger** called **AmountDistribution**.

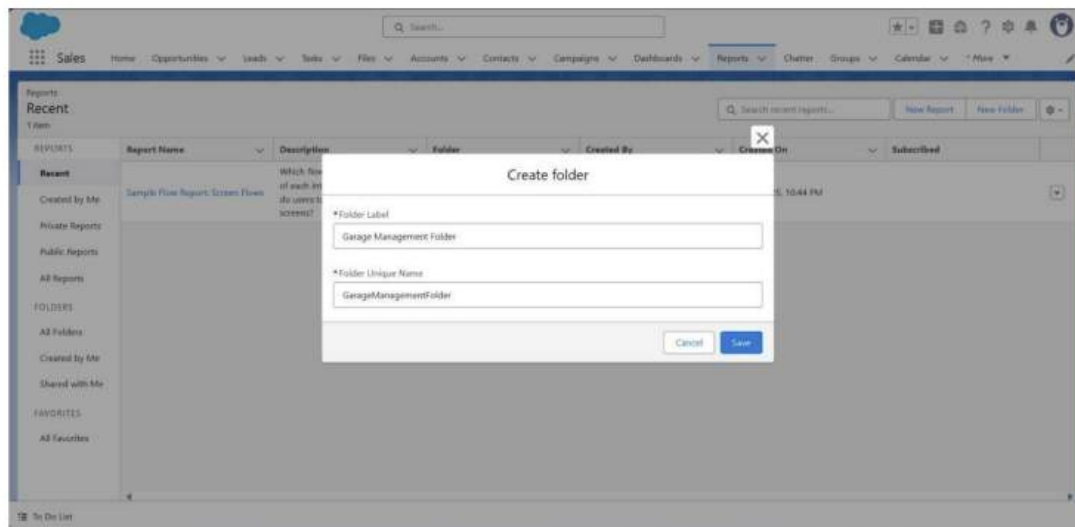


The screenshot shows the Salesforce Developer Console interface. The main editor displays the following Apex trigger code:

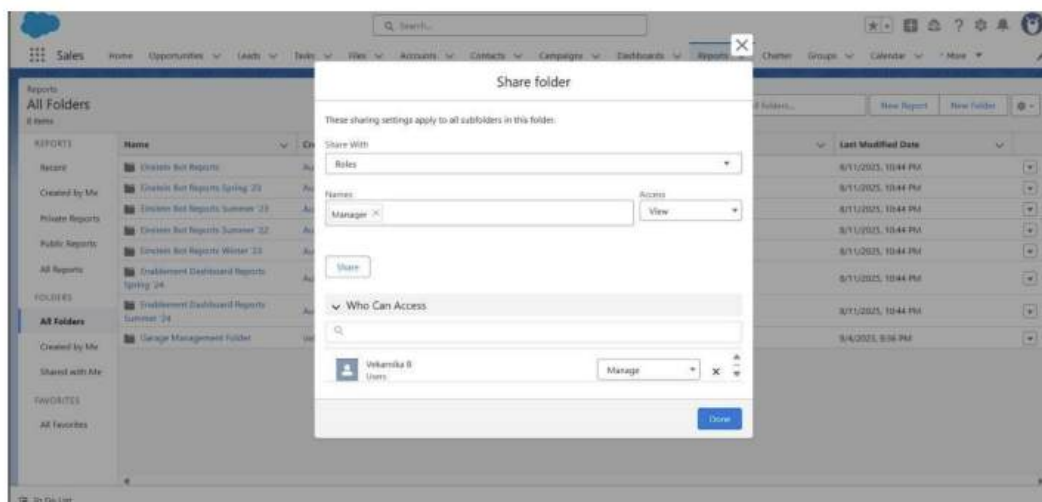
```
1 trigger AmountDistribution on Appointment__c (before insert, before update) {  
2  
3     if(trigger.isbefore && trigger.isinsert || trigger.isupdate){  
4         AmountDistributionHandler.amountDist(trigger.new);  
5     }  
6 }  
7  
8  
9
```

The interface includes a menu bar (File, Edit, Debug, Test, Workspace, Help), a toolbar, and a status bar at the bottom.

- Created **Report** folders, Report type, Report and shared Report folder.



➤ **Report** folders, Report type, Report and shared Report folder.

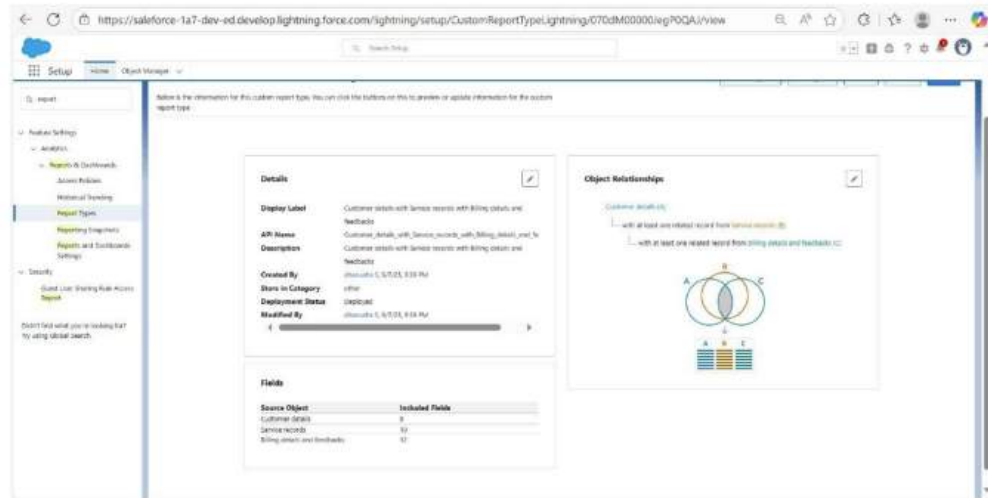


Created

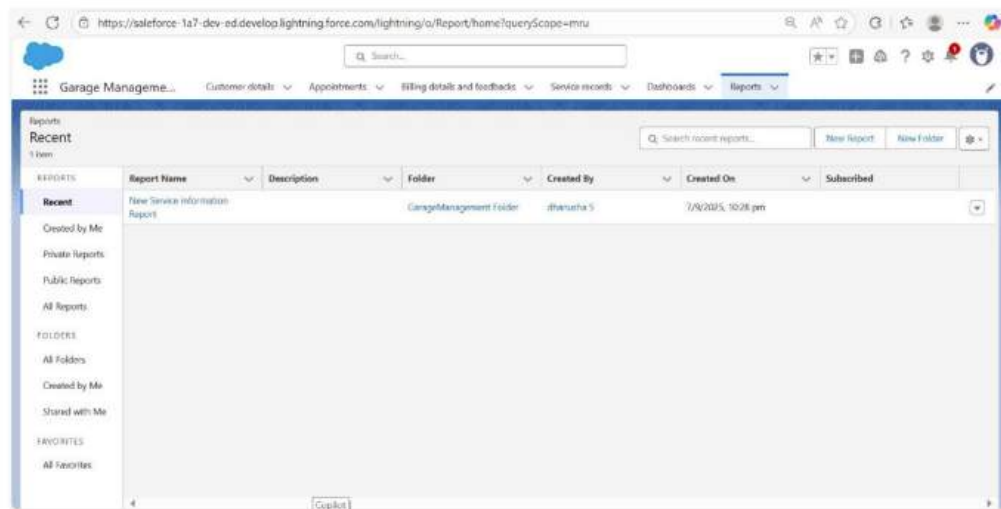
The screenshot displays the Salesforce Setup interface. The top navigation bar includes 'Setup', 'Home', and 'Object Manager'. A search bar is present. The left sidebar lists various setup options like 'Setup Home', 'Salesforce Go', 'Service Setup Assistant', etc. The main content area shows a configuration for a report type, titled 'Select related objects to define which records are included in reports using this report type.' It includes a diagram with four overlapping circles labeled A, B, C, and D. Below the diagram, there are three relationship definitions: 'A to B Relationship', 'B to C Relationship', and 'C to D Relationship'. Each relationship has a radio button to select the relationship type (e.g., 'Each "A" record must have at least one related "B" record.').

Below the configuration, there is a 'New Service information Report' section. It includes a 'Fields' sidebar with 'Groups' and 'Columns' sections. The 'Columns' section lists 'Customer Details Name', 'Appointment Date', 'Service Status', and 'Payment Paid'. The main area shows a preview of the report data, which is a table with columns: 'Rating for service', 'Payment Status', 'Customer Details Name', 'Appointment Date', 'Service Status', and 'Payment Paid'. The data is filtered to show records where 'Rating for service' is 5 and 'Payment Status' is Completed. The table shows 10 records, with the first 6 visible.

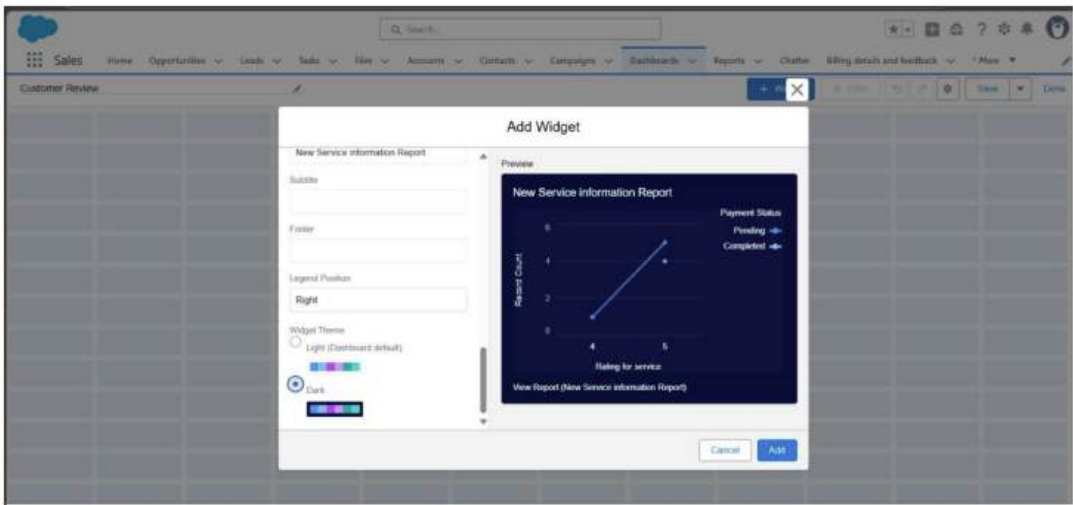
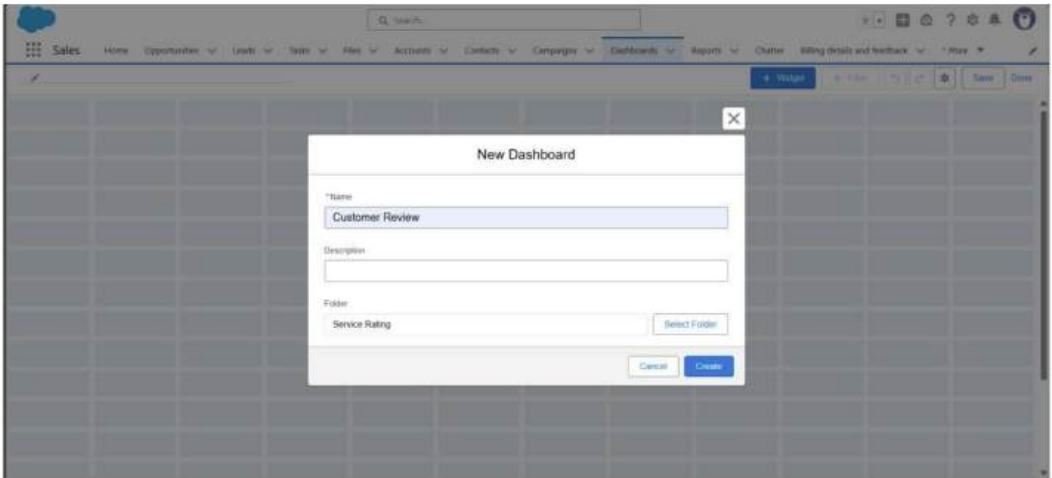
Rating for service	Payment Status	Customer Details Name	Appointment Date	Service Status	Payment Paid
5 (10)	Completed (10)	pavi	9/17/2025	Completed	\$10,000
		Subaash	9/6/2025	Completed	\$10,000
		sandhya	9/7/2025	Completed	\$10,000
		vasanthi	9/6/2025	Completed	\$10,000
		mouli	9/9/2025	Completed	\$10,000
		meera	9/10/2025	Completed	\$10,000

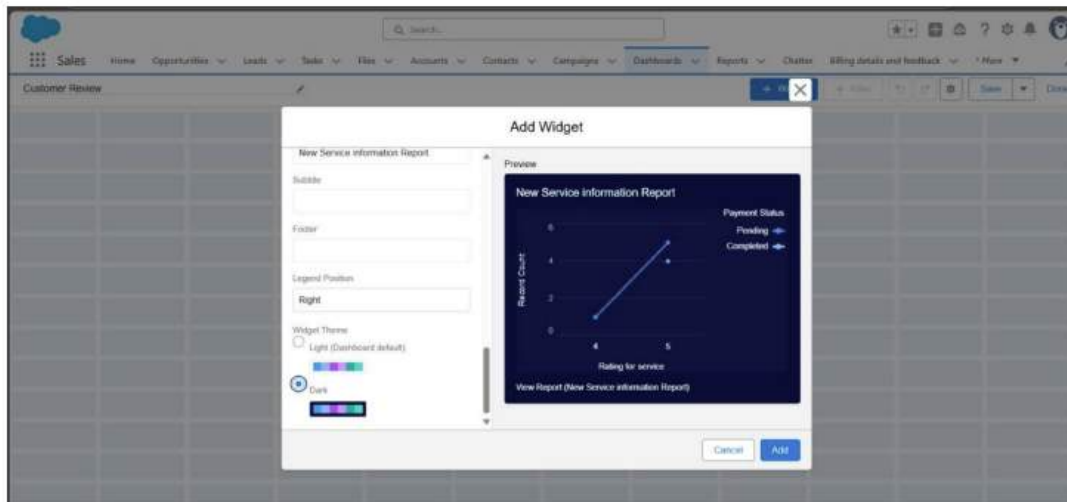


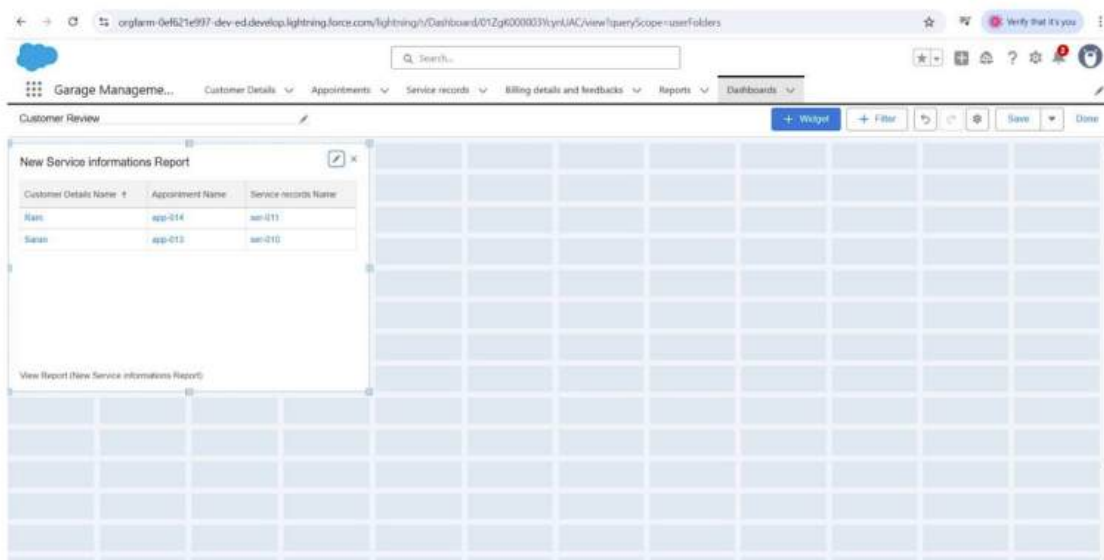
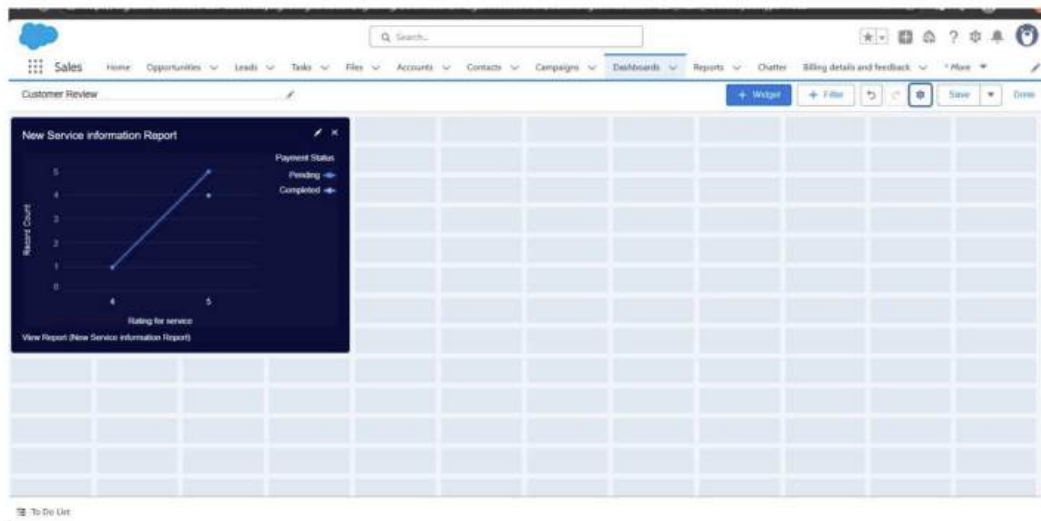
➤ Dashboards and Dashboard folders



Created







2. ADVANTAGES & DISADVANTAGES

Advantages of Garage Management System:

1. **Centralized Data Management** – All customer, vehicle, and service records are stored in one place, improving accessibility and reducing data duplication.
2. **Improved Customer Experience** – Automated service reminders, transparent billing, and quick updates enhance customer trust and satisfaction.
3. **Operational Efficiency** – Streamlines job scheduling, inventory tracking, and billing, reducing manual effort and errors.
4. **Scalability** – Built on Salesforce, the system can easily expand with additional modules or features as the business grows.
5. **Integration Capabilities** – Seamlessly integrates with CRM, payment systems, and other Salesforce apps for smooth workflows.
6. **Analytics & Reporting** – Real-time dashboards and reports help managers track performance, revenue, and customer trends.

Disadvantages of Garage Management System:

1. **Cost of Implementation** – Licensing, customization, and ongoing Salesforce subscription costs may be high for small garages.
2. **Learning Curve** – Staff may require training to adapt to Salesforce's interface and features.
3. **Customization Dependency** – Advanced requirements often need Salesforce developers/consultants, increasing dependency.
4. **Internet Reliance** – Being a cloud-based system, uninterrupted internet is essential for smooth operations.

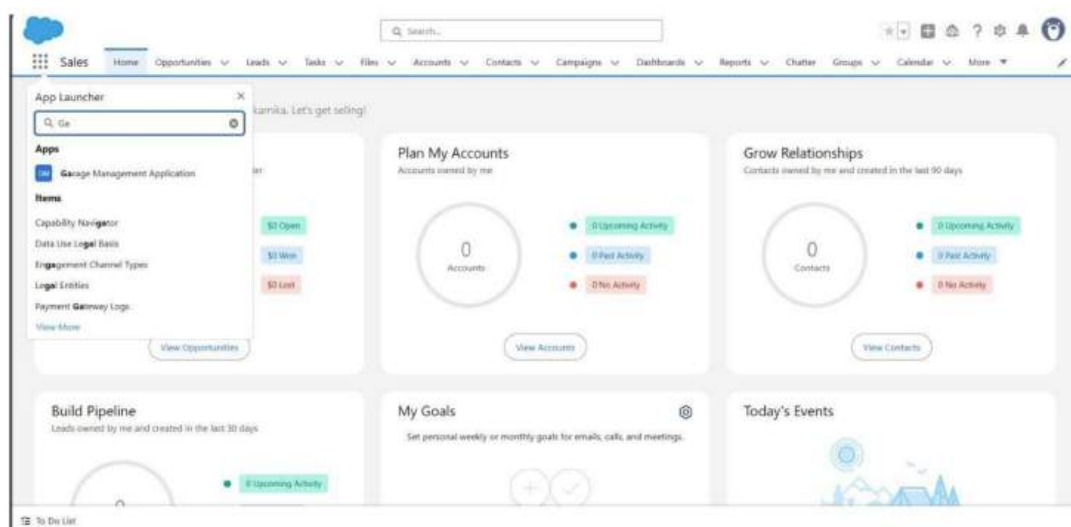
5. **Overhead for Small Businesses** – Smaller repair shops may find the system more complex than needed for basic operations.

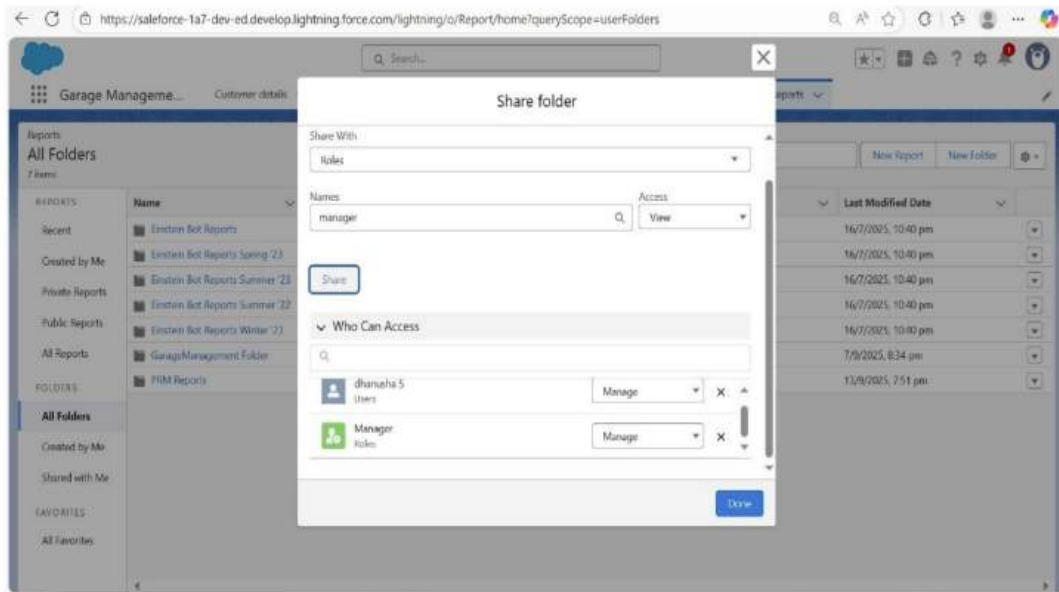
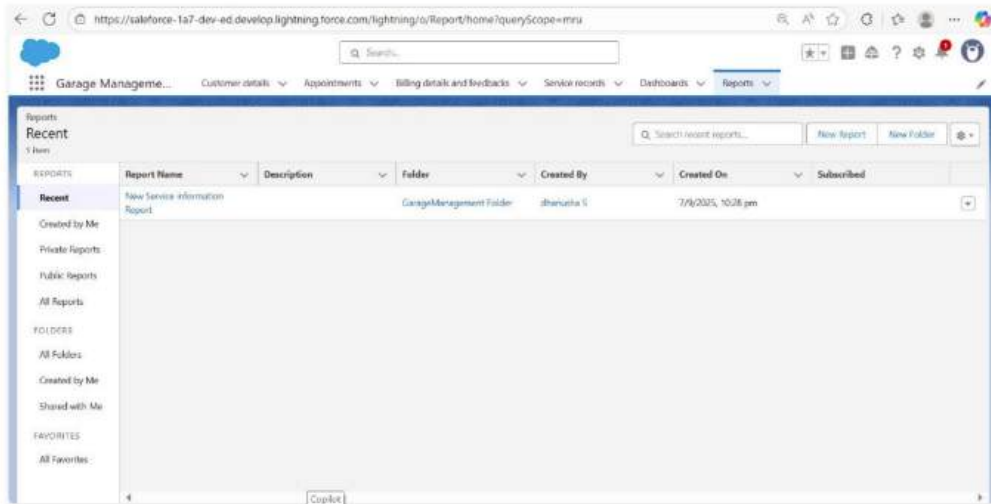
3. CONCLUSION

The Garage Management System streamlines operations, enhances customer satisfaction, and supports business growth, making it a reliable solution for modern automotive repair facilities.

4. RESULTS

Output Screenshots





5. APPENDIX

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

To create apex class (AmountDistributionHandler)

```
public class AmountDistributionHandler {  
  
    public static void amountDist(list<Appointment__c> listApp){  
  
        list<Service_records_c> serList = new list <Service_records_c>();  
  
        for(Appointment__c app : listApp){  
  
            if(app.Maintenance_service_c == true && app.Repairsc == true &&  
                app.Replacement_Parts_c == true){  
app.Service_Amount__c = 10000;  
            }  
        }  
    }  
}
```

```

    }

    else if(app.Maintenance_service_c == true && app.Repairs_c ==
true){
app.Service_Amount__c = 5000;

    }

    else if(app.Maintenance_service_c == true &&
app.Replacement_Parts_c == true){
app.Service_Amount__c = 8000;

    }

    else if(app.Repairs_c == true && app.Replacement_Parts_c ==
true){
app.Service_Amount__c = 7000;

    }

    else if(app.Maintenance_service_c == true){ app.Service_Amount
__c = 2000;

    }

    else if(app.Repairs_c == true){ app.Service_Amount_c
= 3000;

    }

```

```
else if(app.Replacement_Parts_c == true){ app.Service_Amount  
    ____c = 5000;  
}
```

```
}
```

```
}
```

```
}
```

AmountDistribution - Apex Trigger:

```
trigger AmountDistribution on Appointment_c (before insert, before update) {
```

```
    if(trigger.isbefore && trigger.isinsert || trigger.isupdate){
```

```
        AmountDistributionHandler.amountDist(trigger.new);
```

```
    }
```

```
}
```
