Garage management system

Date	02 NOV 2025
Team ID	NM2025TMID02807
Project Name	Garage management system

Summary

The **Garage Management System** is a software application designed to streamline and automate the daily operations of a vehicle service garage. The main goal of the system is to improve efficiency by managing customer information, vehicle details, service records, billing, and staff activities in a centralized platform.

This system allows garage owners and employees to easily track ongoing repairs, manage spare parts inventory, schedule maintenance services, and generate digital invoices for customers. It also helps maintain accurate records of each vehicle's service history, enabling quick access to past data and better customer service.

By reducing manual paperwork and minimizing errors, the Garage Management System enhances productivity, saves time, and ensures smoother workflow management. Overall, it provides a smart and reliable solution for modernizing garage operations and improving customer satisfaction.

1.ABSTRACT

The Garage Management System (GMS) developed using Salesforce is a comprehensive cloud-based application designed to streamline operations in automotive repair facilities. This system automates the management of customer details, appointments, service records, and billing processes.

By leveraging Salesforce's CRM (Customer Relationship Management) capabilities, the project integrates standard and custom objects, flows, validation rules, and dashboards to deliver an efficient workflow. The system reduces manual paperwork, minimizes data entry errors, and provides managers with real-time visibility into service performance.

Salesforce's cloud infrastructure ensures data is securely stored, easily accessible, and scalable. With automation tools like Flow Builder and Apex triggers, key tasks such as calculating service costs, updating statuses, and sending email alerts are executed automatically.

Ultimately, the Garage Management System enhances customer satisfaction, ensures transparency, and demonstrates the power of low-code Salesforce development in real-world business solutions.

Thus this is the Garage Management System's Abstract that system automates the management of customer details, appointments, service records, and billing processes. The system reduces manual paperwork, minimizes data entry errors, and provides managers with real-time visibility into service performance.

2.INTRODUCTION

The Garage Management System is a valuable tool for automotive repair facilities, helping them deliver top-notch service, increase operational efficiency, and build lasting customer relationships. With its user-friendly interface and powerful features, GMS empowers garages to thrive in a competitive market while ensuring a seamless and satisfying experience for both customers and staff.

Salesforce is a customer success platform designed to help users sell, service, market, analyze, and connect with customers. It provides everything needed to run a business from anywhere, including managing relationships with prospects and customers, collaborating with employees and partners, and storing data securely in the cloud. Before Salesforce, contacts, emails, follow-up tasks, and prospective deals were often disorganized. This project utilizes Salesforce to create a developer account, define custom objects, and implement automation for garage management.

To begin, a developer org was created at https://developer.salesforce.com/signup by entering details such as name, email, role (Developer), company (College Name), country (India), postal code, and a unique username in the format username@organization.com. The account was activated via email verification, and a password was set, redirecting to the Salesforce setup page.

Salesforce objects are database tables for storing organization-specific data. Standard objects include users, contracts, reports, and dashboards, while custom objects are user-created for unique information. Custom objects form the core of the application.

In this project, custom objects were created via the Setup page > Object Manager > Create > Custom Object. The objects include:

- Customer Details: Label "Customer Details", Plural "Customer Details", Record Name "Customer Name" (Text), with options for reports, field history, and search.
- Appointment: Label "Appointment", Plural "Appointments", Record Name "Appointment Name" (Auto Number, format "app-{000}", starting 1), with reports, field history, and search.
- Service Records: Label "Service records", Plural "Service records", Record Name "Service records Name" (Auto Number, format "ser-{000}", starting 1), with reports, field history, and search.
- Billing Details and Feedback: Label "Billing details and feedback", Plural "Billing details and feedback", Record Name "Billing details and feedback Name" (Auto Number, format "bill-{000}", starting 1), with reports, field history, and search.

Tabs were created for these objects to provide user interfaces. Tabs include Custom Tabs, Web Tabs, Visualforce Tabs, Lightning Component Tabs, and Lightning Page Tabs. Custom tabs were created for each object via Setup > Tabs > New, selecting the object, style, and saving.

A Lightning App named "Garage Management Application" was created, adding navigation items for the objects, reports, and dashboards, and assigning to the System Administrator profile.

3.OBJECTIVES

The primary objectives of the Garage Management System using Salesforce are:

- To automate the management of customer details, appointments, service records, and billing in automotive garages.
- To ensure data integrity through validation rules, duplicate rules, and lookup relationships.
- To provide role-based access control for managers and salespersons.
- To implement automation using flows and Apex triggers for efficient workflow.
- To generate reports and dashboards for insights into service information and performance.
- To enhance customer satisfaction by sending automated email alerts upon payment completion.

4. SYSTEM REQUIREMENTS

4.1 HARDWARE REQUIREMENTS:

- Processor: Intel Core i5 or equivalent
- □ RAM: 8 GB minimum
- □ Hard Disk: 256 GB SSD
- □ Internet Connection: High-speed broadband for cloud access

4.2 SOFTWARE REQUIREMENTS:

- Operating System: Windows 10/11, macOS, or Linux
- Salesforce Platform: Developer Edition (free signup at developer.salesforce.com)
- No additional installations required as Salesforce is cloud-based

5. MODULES OF THE SYSTEM:

The system comprises several modules implemented through Salesforce customizations:

1. **Customer Details Module**: Manages customer information. Fields include Phone Number (Phone type) and Gmail (Email type). Matching and duplicate rules ensure uniqueness based on Gmail and Phone Number.
2. **Appointment Module**: Handles scheduling. Fields include Appointment Date (Date, required), Vehicle Number Plate (Text, length 10, required, unique), Service Amount (Currency, read-only), and checkboxes for Maintenance Service, Repairs, and Replacement Parts. Validation rule enforces vehicle number format (e.g., NOT(REGEX(Vehicle_number_plate__c, "[A-Z]{2}[0-9]{2}[A-Z]{2}[0-9]{4}")) with error "Please enter valid number").
3. **Service Records Module**: Tracks services. Fields include Service Status (Picklist: Started, Completed), Quality Check Status (Checkbox, default unchecked), and Service Date (Formula: CreatedDate). Lookup to Appointment with filter (Appointment Date < Created Date, required).
4. **Billing Details and Feedback Module**: Manages payments and feedback. Fields include Payment Status (Picklist: Pending, Completed), Payment Paid (Currency), and Rating for Service (assumed Text). Lookup to Service Records.

Validation rule for rating (NOT(REGEX(Rating_for_service__c, "[1-5]{1}")) with error "rating should be from 1 to 5").

- Lookup relationships connect modules: Appointment to Customer Details, Service Records to Appointment, Billing Details and Feedback to Service Records.
- Profiles were created: Manager (cloned from Standard User, with full access to objects, session timeout 8 hours, passwords never expire, min length 8) and Sales Person (cloned from Salesforce Platform User, with limited access).
- Roles include Manager and Sales Person under CEO. Users were created (e.g., Niklaus Mikaelson as Manager, and three Sales Persons).
- Public Group "Sales Team" includes Sales Person role.
- Sharing Settings: OWD for Service Records set to Private. Sharing rule shares Sales Person records with Manager (Read/Write).

6.TECHNOLOGIES USED

- **Salesforce Platform**: Cloud-based CRM for custom objects, tabs, apps, fields, flows, Apex, reports, and dashboards.

- **Apex Triggers and Classes:** For custom logic, e.g., AmountDistributionHandler class and trigger on Appointment for calculating Service Amount based on selected services (e.g., all three: 10000, Maintenance+Repairs: 5000).

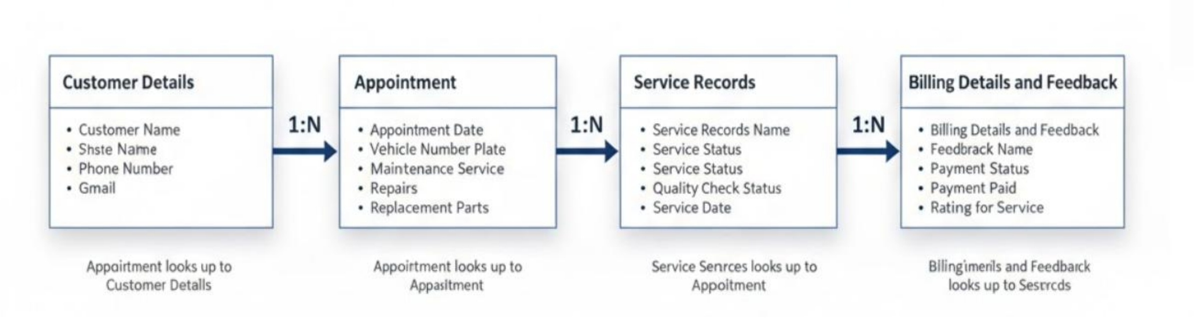
- **Flows:** Record-triggered flows for updating records (e.g., update Payment Paid on completion, send email alerts, update Service Status on quality check).

- **Validation and Matching Rules:** For data integrity.

- **Reports and Dashboards**: For analytics on service information.

7. ER DIAGRAM

The Entity-Relationship (ER) diagram represents the database structure with entities and relationships based on the custom objects created.



Relationships (Lookup, One-to-Many):

- Customer Details --(1:N)-- Appointment (Appointment looks up to Customer Details)
- Appointment --(1:N)-- Service Records (Service Records looks up to Appointment)
- Service Records --(1:N)-- Billing Details and Feedback (Billing Details and Feedback looks up to Service Records)

8. WORKFLOW DESCRIPTION

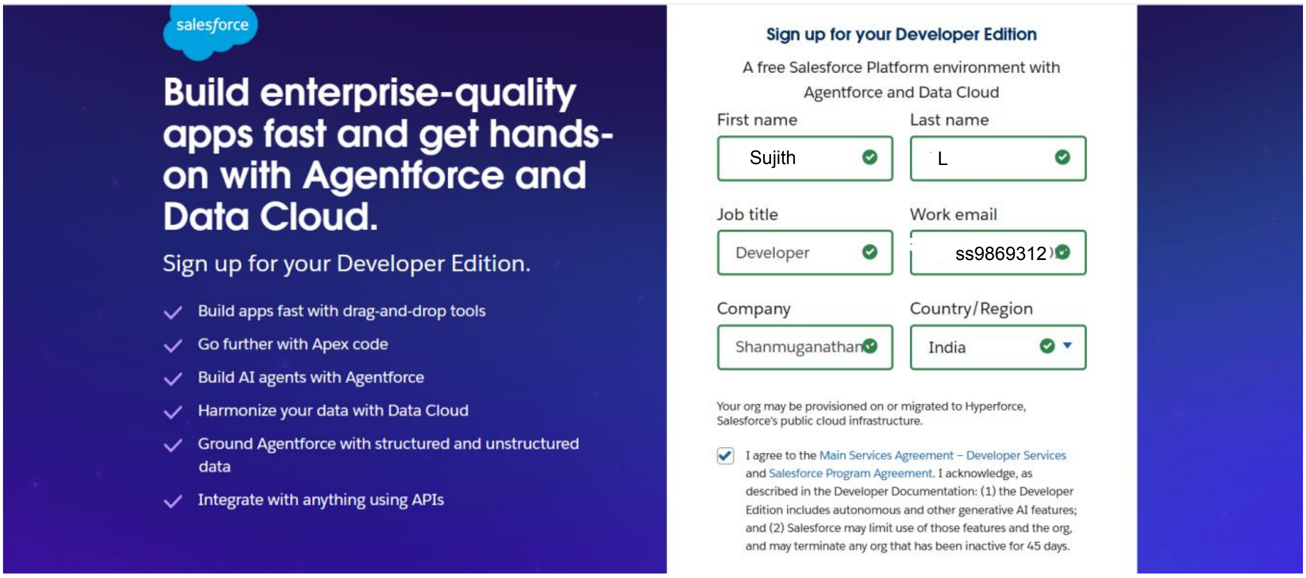
The workflow automates garage operations:

1. **Customer Registration**: Create Customer Details record with phone and email.
2. **Appointment Booking**: Create Appointment linked to Customer, enter date (must be < created date), vehicle plate (validated), select services. Apex trigger calculates Service Amount (before insert/update).
3. **Service Recording**: Create Service Records linked to Appointment. On save, if Quality Check Status is true, flow updates Service Status to Completed.
4. **Billing and Feedback**: Create Billing record linked to Service Records. On payment completion, flow updates Payment Paid from Appointment's Service Amount and sends email alert: "Dear [Customer Name], ... Amount paid: [Amount]. Thank you."
5. **Validation and Duplicates**: Rules prevent invalid data; matching rules on Customer Details prevent duplicates.
6. **Access Control**: Managers have full access; Sales Persons have limited. Sharing rules allow Managers to view/edit Sales Person records.
7. **Reports and Dashboards**: Report Type "Service Information" joins objects. Report "New Service Information Report" groups by Rating and Payment Status, with columns for Customer Name, Appointment Date, Service Status, Payment Paid. Dashboard "Service Rating Dashboard" uses line chart, subscribed weekly.

Records were created (10+ per object) to test.

9.IMPLEMENTED STEPS

1.Creating Developer Account:



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: Sujith

Last name: L

Job title: Developer

Work email: ss9869312

Company: Shanmuganathan

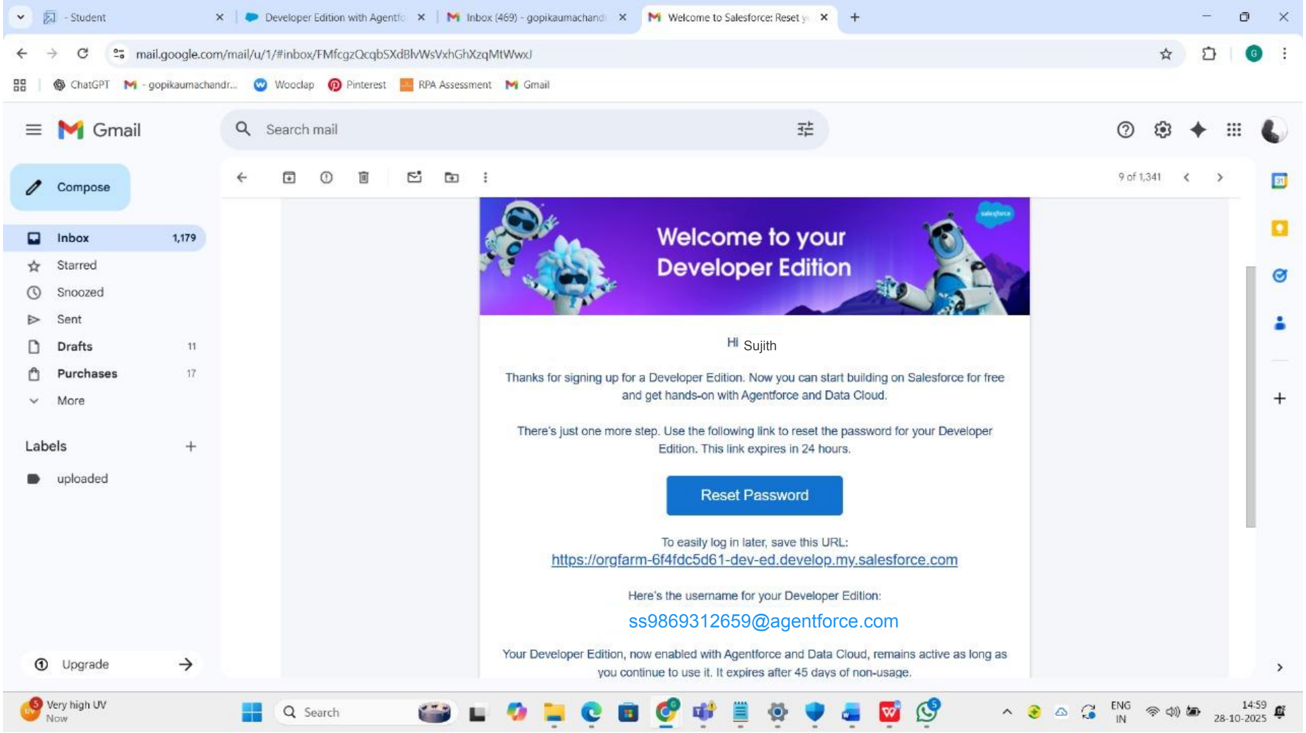
Country/Region: India

Your org may be provisioned on or migrated to Hyperforce, Salesforce's public cloud infrastructure.

☑ I agree to the Main Services Agreement – Developer Services and Salesforce Program Agreement. I acknowledge, as described in the Developer Documentation: (1) the Developer Edition includes autonomous and other generative AI features; and (2) Salesforce may limit use of those features and the org, and may terminate any org that has been inactive for 45 days.

Fig:1.1 Developer Account

2. Account Activation:



mail.google.com/mail/u/1/#inbox/FMfcgzQcqbSXdBlvWsVxhGhXzqMtWwxJ

Welcome to your Developer Edition

Hi Sujith

Thanks for signing up for a Developer Edition. Now you can start building on Salesforce for free and get hands-on with Agentforce and Data Cloud.

There's just one more step. Use the following link to reset the password for your Developer Edition. This link expires in 24 hours.

Reset Password

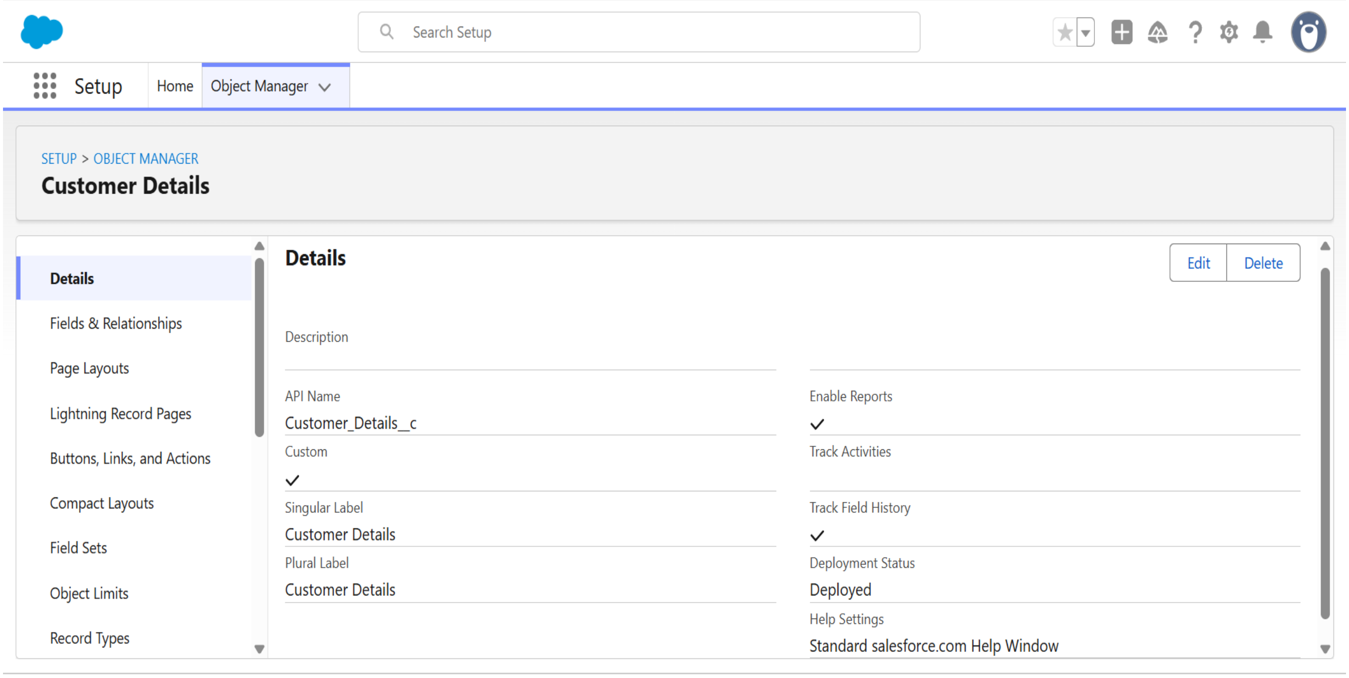
To easily log in later, save this URL:
https://orgfarm-6f4fdc5d61-dev-ed.develop.my.salesforce.com

Here's the username for your Developer Edition:
ss9869312659@agentforce.com

Your Developer Edition, now enabled with Agentforce and Data Cloud, remains active as long as you continue to use it. It expires after 45 days of non-usage.

Fig:2.1 Verifying Account

3. Object Creation:



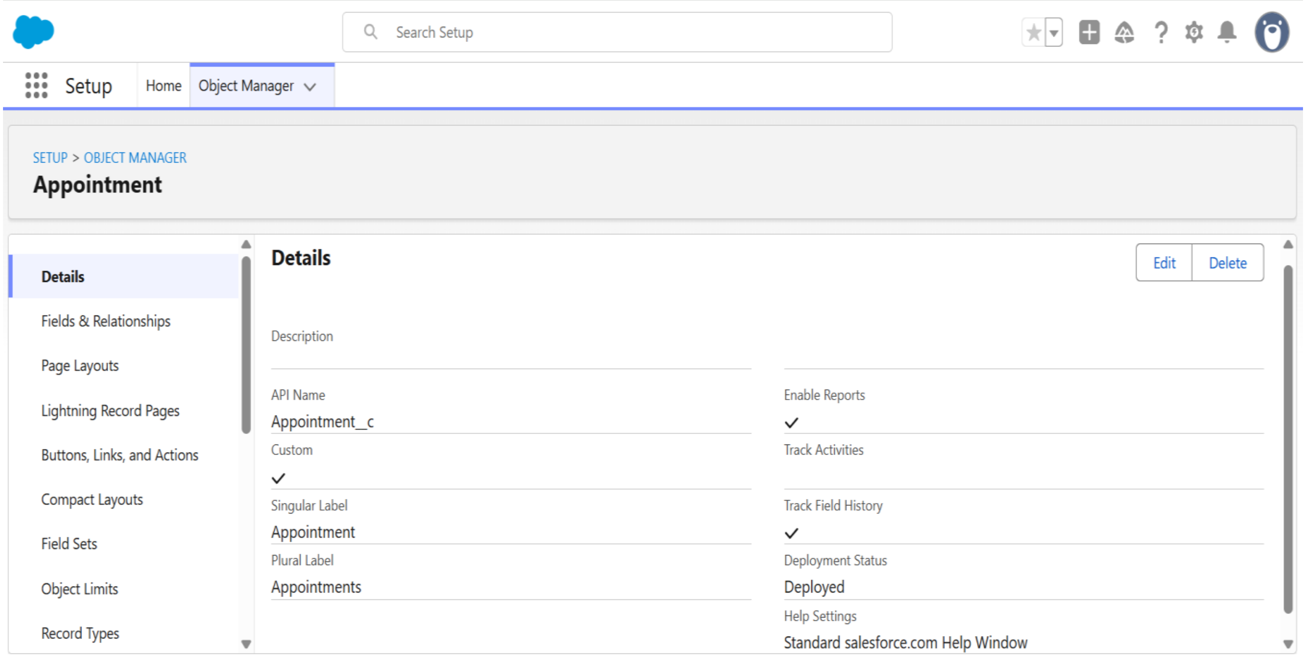
Customer Details

Details

Description

API Name Customer_Details__c	Enable Reports ✓
Custom ✓	Track Activities
Singular Label Customer Details	Track Field History ✓
Plural Label Customer Details	Deployment Status Deployed
	Help Settings Standard salesforce.com Help Window

Fig :3.1 Creation of Customer details Object



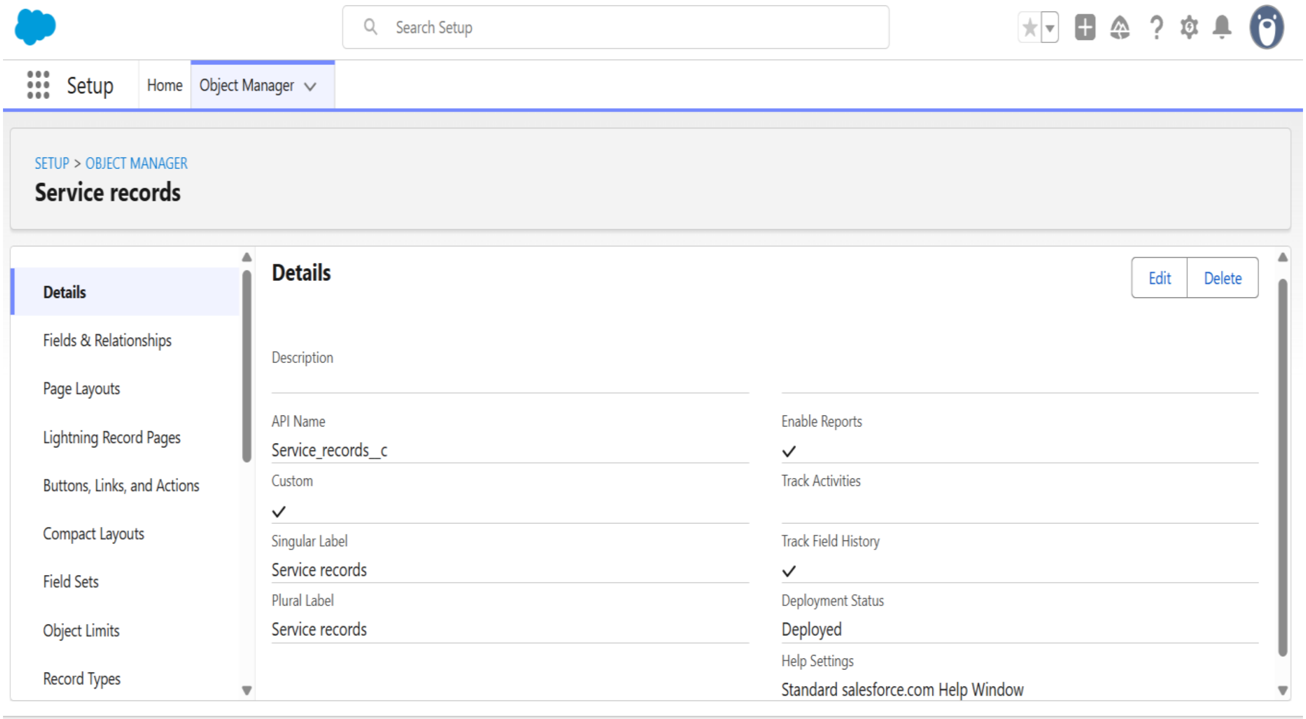
Appointment

Details

Description

API Name Appointment__c	Enable Reports ✓
Custom ✓	Track Activities
Singular Label Appointment	Track Field History ✓
Plural Label Appointments	Deployment Status Deployed
	Help Settings Standard salesforce.com Help Window

Fig :3.2 Creation of Appointment Object



Service records

Details

Description

API Name: Service_records__c

Enable Reports: ✓

Custom: ✓

Track Activities:

Singular Label: Service records

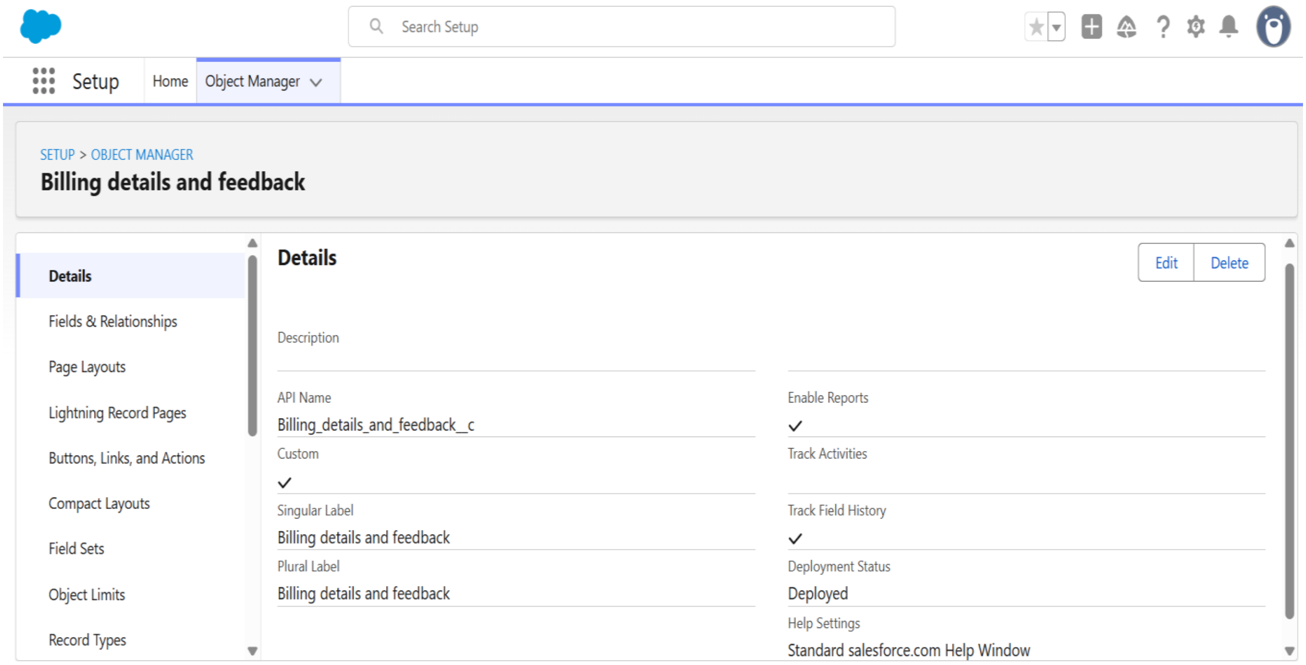
Track Field History: ✓

Plural Label: Service records

Deployment Status: Deployed

Help Settings: Standard salesforce.com Help Window

Fig :3.3 Creation of Service records Object



Billing details and feedback

Details

Description

API Name: Billing_details_and_feedback__c

Enable Reports: ✓

Custom: ✓

Track Activities:

Singular Label: Billing details and feedback

Track Field History: ✓

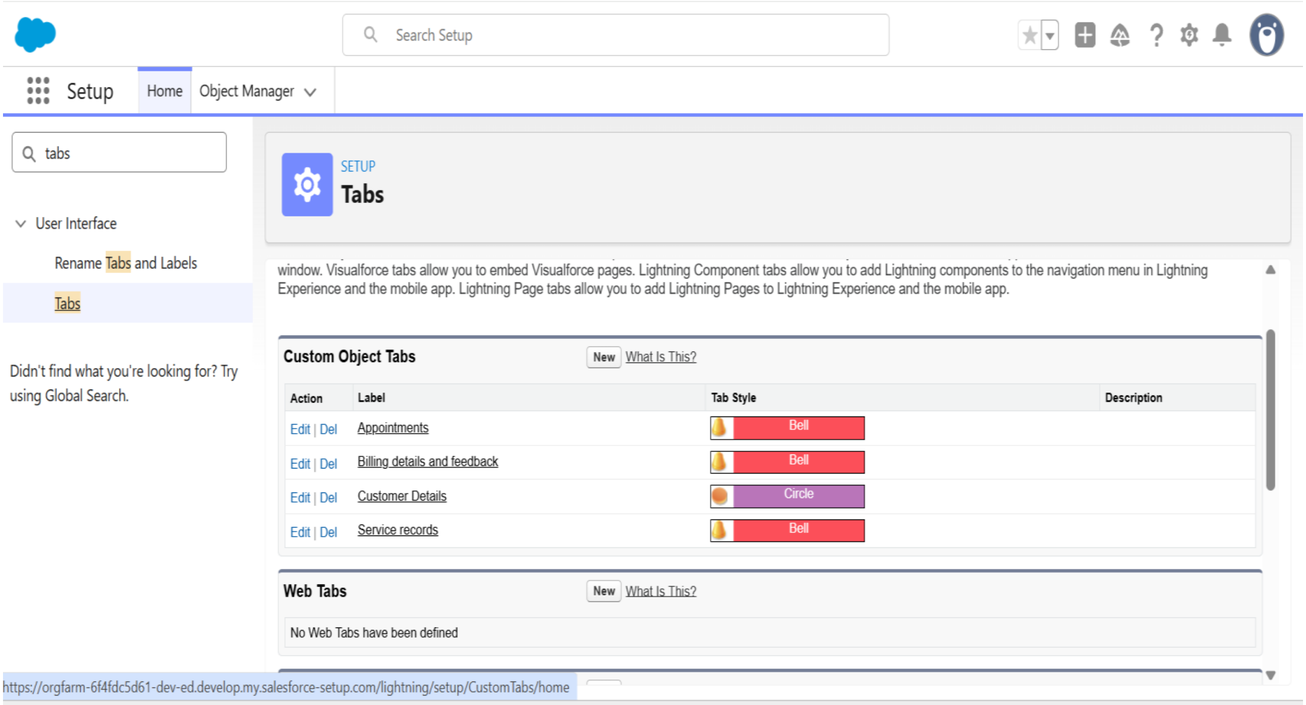
Plural Label: Billing details and feedback

Deployment Status: Deployed

Help Settings: Standard salesforce.com Help Window

Fig :3.4 Creation of Billing details and Feedback Object

4. Tabs:



Didn't find what you're looking for? Try using Global Search.

SETUP

Tabs

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 New What Is This?

Action	Label	Tab Style	Description
Edit | Del	Appointments	Bell	
Edit | Del	Billing details and feedback	Bell	
Edit | Del	Customer Details	Circle	
Edit | Del	Service records	Bell	

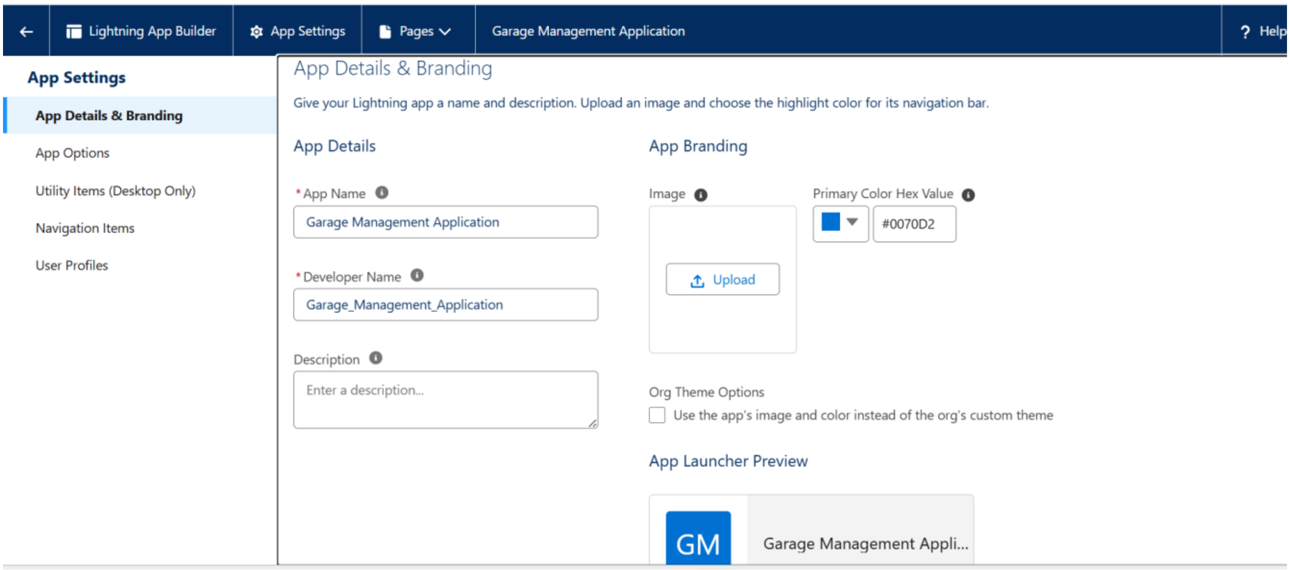
Web Tabs New What Is This?

No Web Tabs have been defined

https://orgfarm-6f4fdc5d61-dev-ed.develop.my.salesforce-setup.com/lightning/setup/CustomTabs/home

Fig :4.1 Creation of a Custom Tab

5.The Ligthning App:



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 Name: Garage Management Application

* Developer Name: Garage_Management_Application

Description: Enter a description...

App Branding

Image

Primary Color Hex Value: #0070D2

Org Theme Options

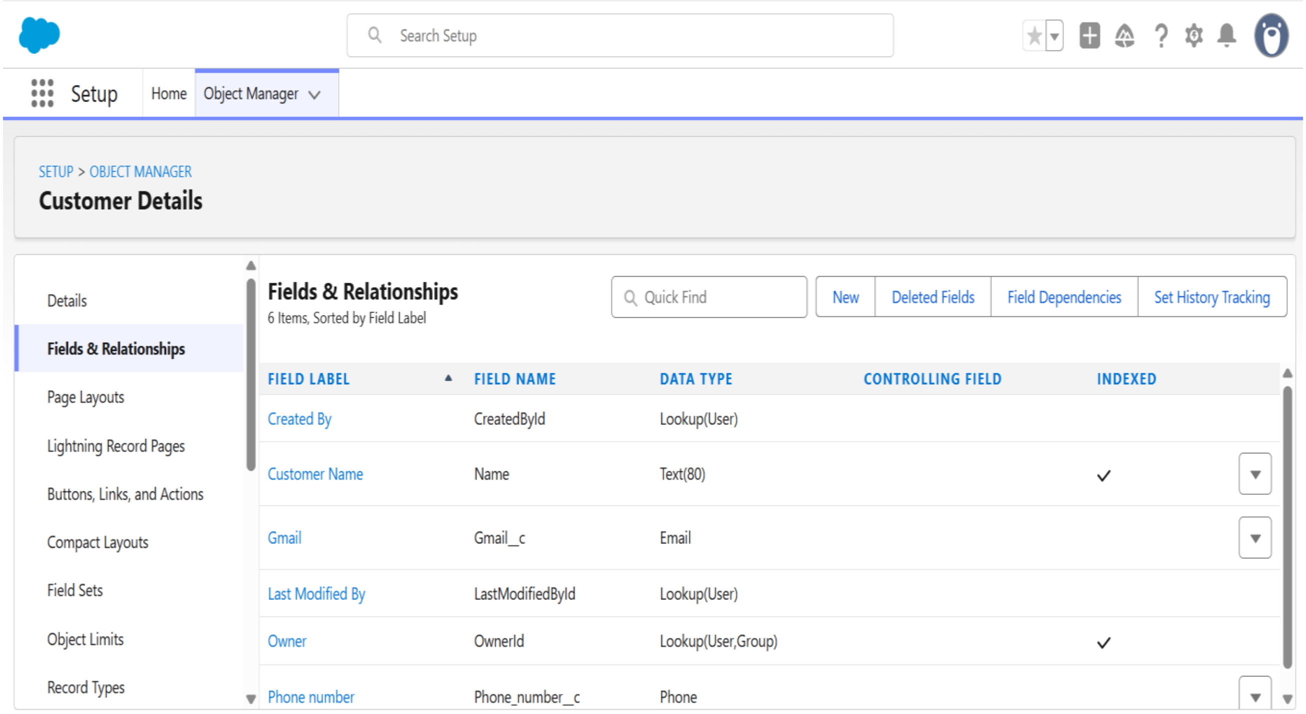
□ Use the app's image and color instead of the org's custom theme

App Launcher Preview

Garage Management Appli...

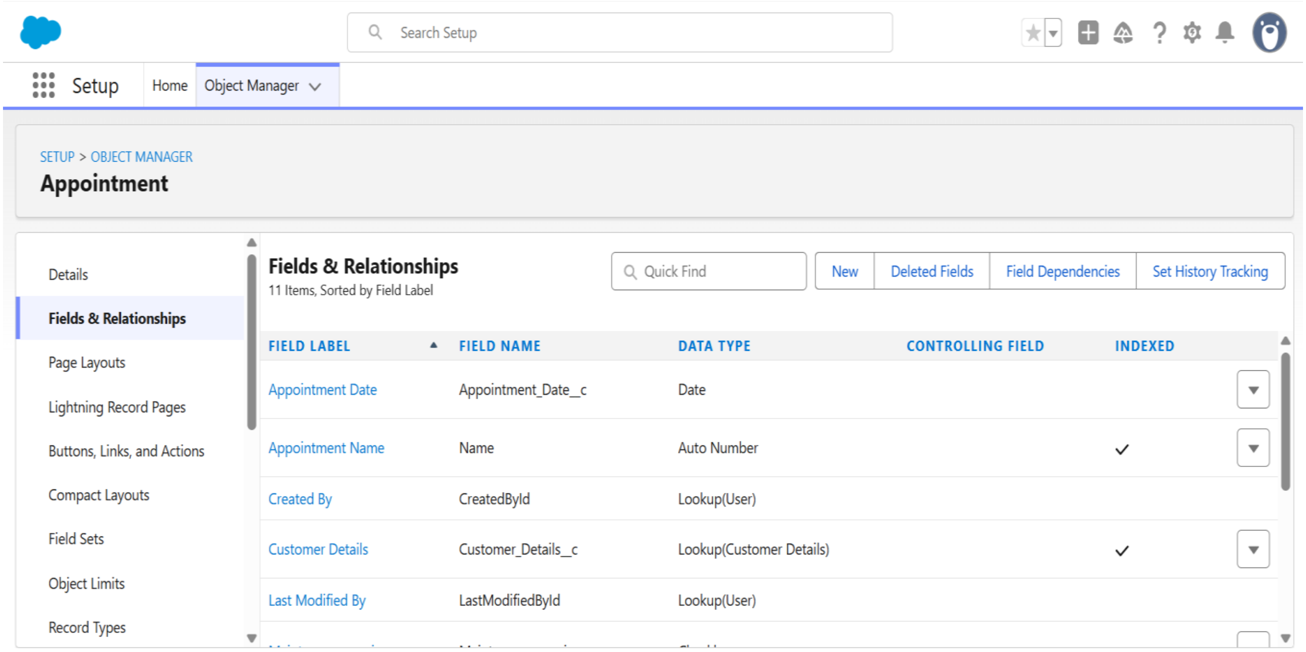
Fig :5.1 Garage Management Application

6.Fields:



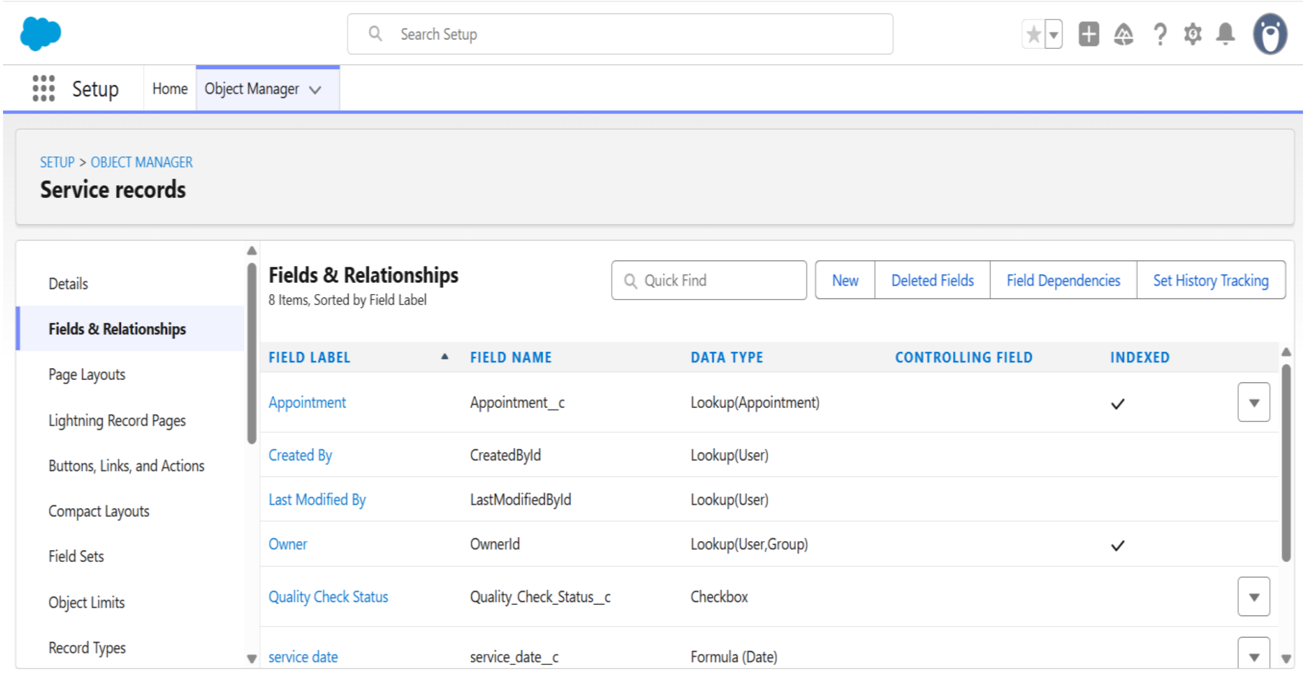
FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Created By	CreatedById	Lookup(User)		
Customer Name	Name	Text(80)		✓
Gmail	Gmail__c	Email		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		✓
Phone number	Phone_number__c	Phone		

Fig :6.1 Creation of fields for the Customer Details object



FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Appointment Date	Appointment_Date__c	Date		
Appointment Name	Name	Auto Number		✓
Created By	CreatedById	Lookup(User)		
Customer Details	Customer_Details__c	Lookup(Customer Details)		✓
Last Modified By	LastModifiedById	Lookup(User)		

Fig :6.2 Creation of fields for the Appointments object



SETUP > OBJECT MANAGER

Service records

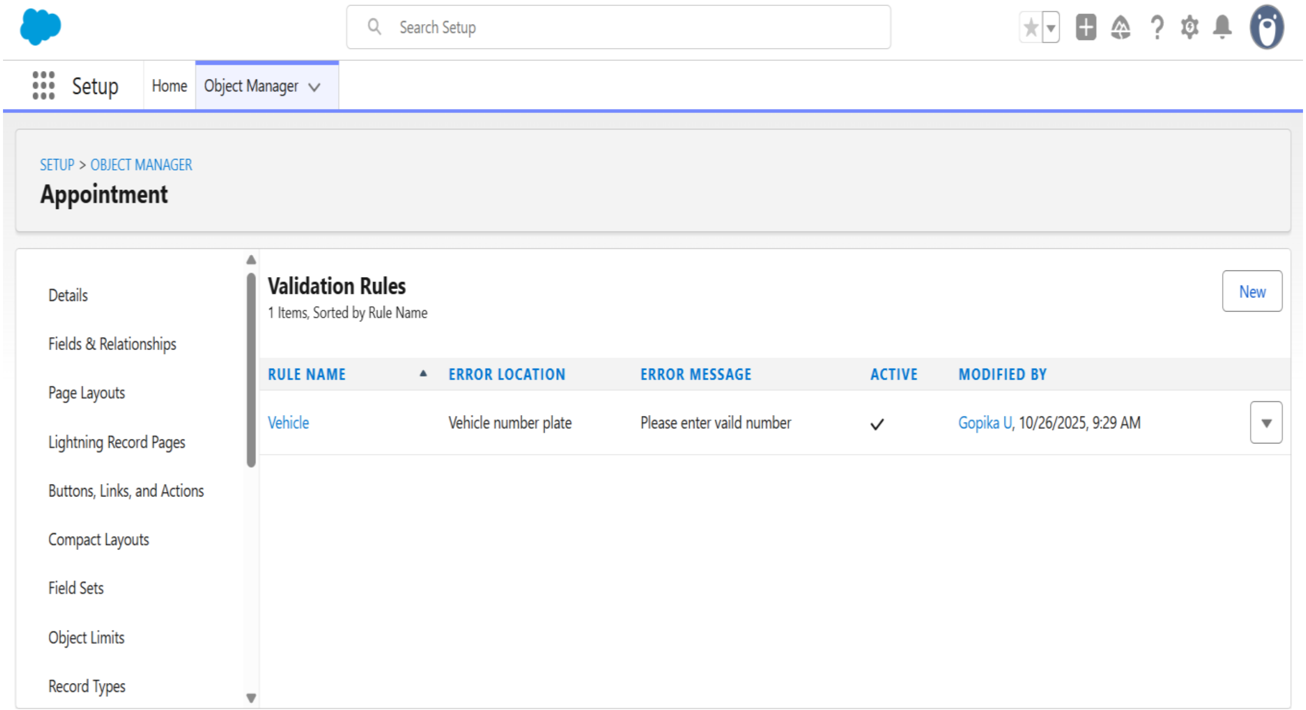
Fields & Relationships

8 Items, Sorted by Field Label

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Appointment	Appointment__c	Lookup(Appointment)		✓
Created By	CreatedById	Lookup(User)		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		✓
Quality Check Status	Quality_Check_Status__c	Checkbox		
service date	service_date__c	Formula (Date)		

Fig :6.3 Creation of fields for the Service records object

7.Validation Rules:



SETUP > OBJECT MANAGER

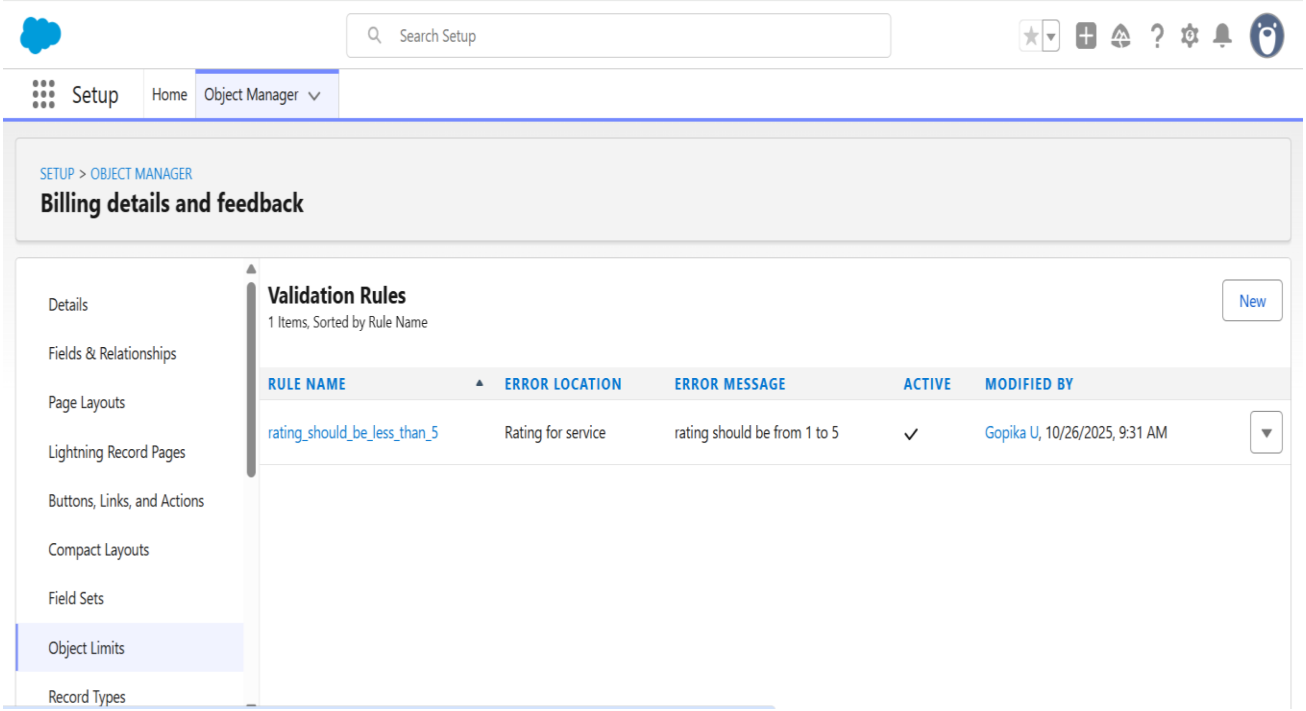
Appointment

Validation Rules

1 Items, Sorted by Rule Name

RULE NAME	ERROR LOCATION	ERROR MESSAGE	ACTIVE	MODIFIED BY
Vehicle	Vehicle number plate	Please enter vaild number	✓	Gopika U, 10/26/2025, 9:29 AM

Fig :7.1 Validation Rules for Appointment



SETUP > OBJECT MANAGER

Billing details and feedback

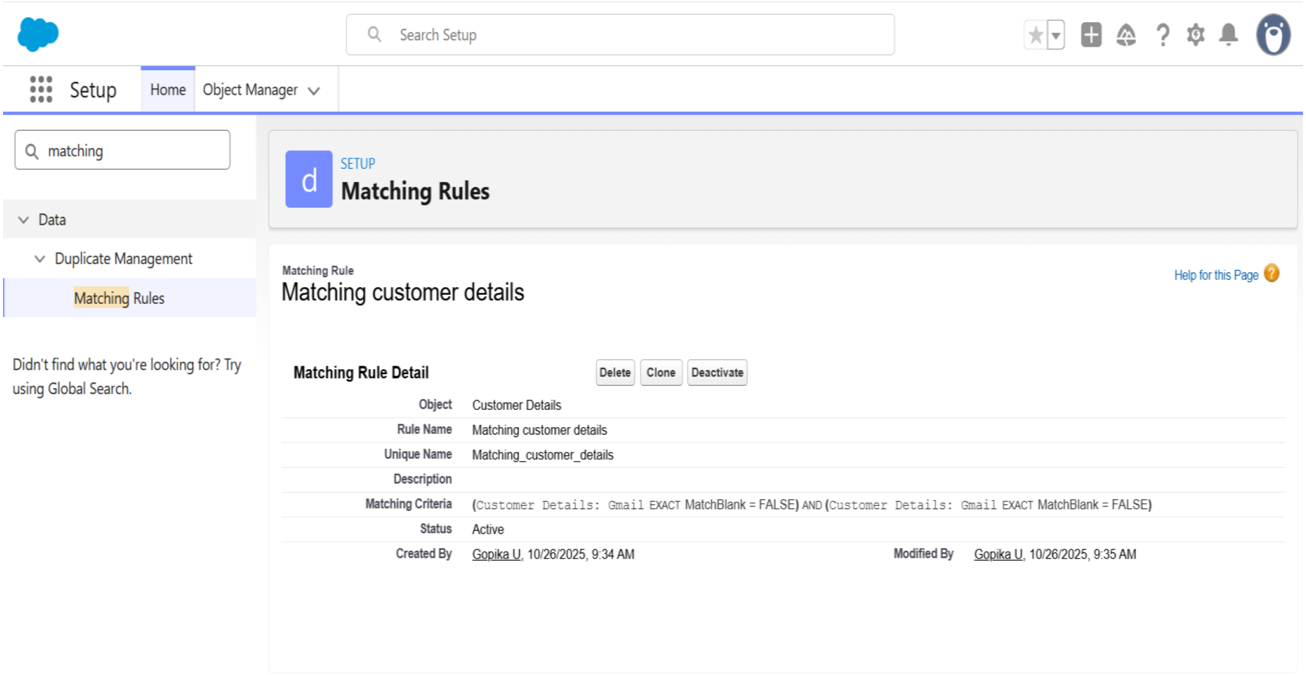
Validation Rules

1 Items, Sorted by Rule Name

RULE NAME	ERROR LOCATION	ERROR MESSAGE	ACTIVE	MODIFIED BY
rating_should_be_less_than_5	Rating for service	rating should be from 1 to 5	✓	Gopika U, 10/26/2025, 9:31 AM

Fig :7.2 Validation Rules for Billing details and feedback

8.Duplicate Rule:



SETUP

Matching Rules

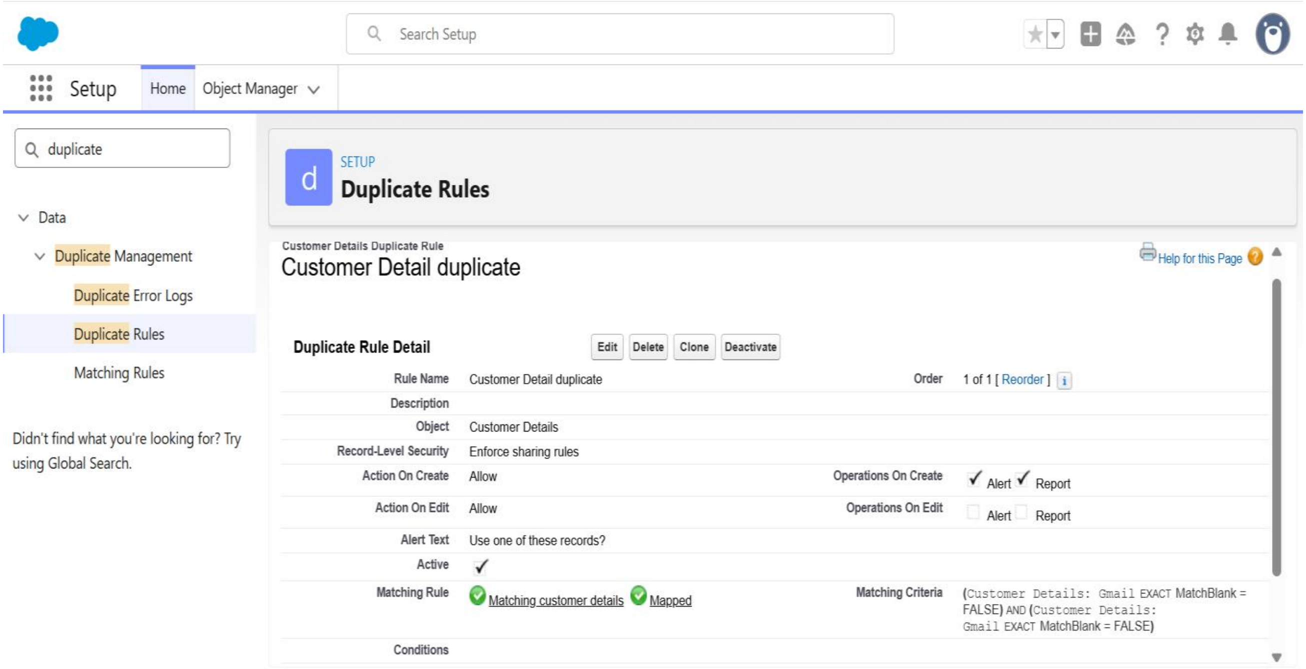
Matching Rule

Matching customer details

Matching Rule Detail

Object	Customer Details
Rule Name	Matching customer details
Unique Name	Matching_customer_details
Description	
Matching Criteria	(Customer Details: Gmail EXACT MatchBlank = FALSE) AND (Customer Details: Gmail EXACT MatchBlank = FALSE)
Status	Active
Created By	Gopika U, 10/26/2025, 9:34 AM
Modified By	Gopika U, 10/26/2025, 9:35 AM

Fig :8.1 Matching rule to an Customer details Object



Customer Details Duplicate Rule

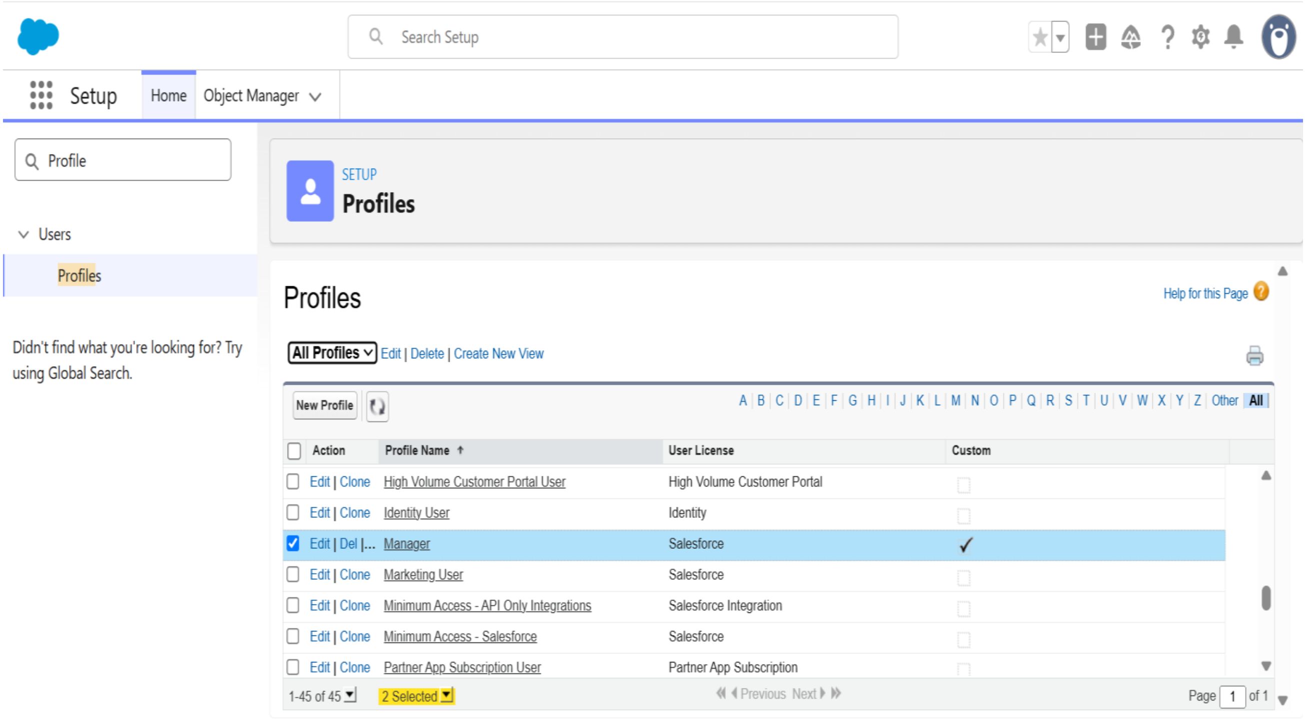
Customer Detail duplicate

Duplicate Rule Detail

Rule Name	Customer Detail duplicate	Order	1 of 1 [Reorder]
Description			
Object	Customer Details		
Record-Level Security	Enforce sharing rules		
Action On Create	Allow	Operations On Create	✓ Alert ✓ Report
Action On Edit	Allow	Operations On Edit	□ Alert □ Report
Alert Text	Use one of these records?		
Active	✓		
Matching Rule	Matching customer details Mapped	Matching Criteria	(Customer Details: Gmail EXACT MatchBlank = FALSE) AND (Customer Details: Gmail EXACT MatchBlank = FALSE)
Conditions			

Fig :8.2 Duplicate rule to an Customer details Object

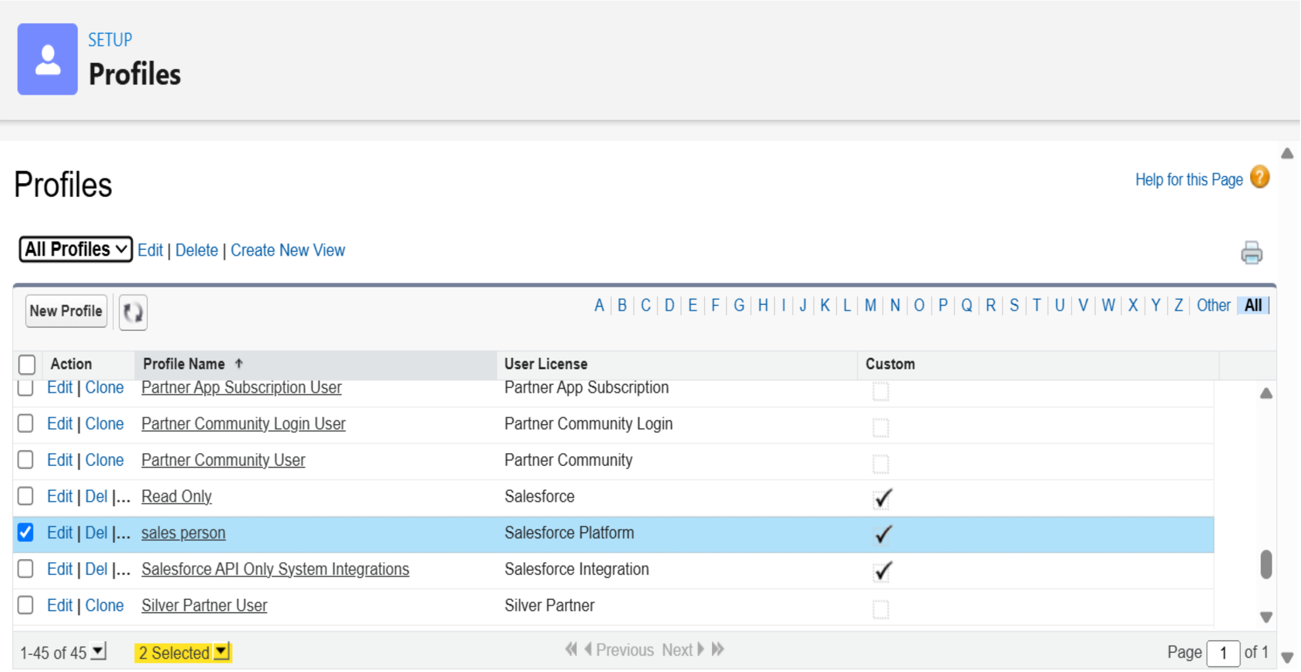
9.Profile:


Search Setup
Setup
Home
Object Manager
Profile
Users
Profiles
Didn't find what you're looking for? Try using Global Search.
SETUP
Profiles
Help for this Page
All Profiles
Edit | Delete | Create New View
New Profile
A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | Other | All
1-45 of 45
2 Selected
Previous Next
Page 1 of 1

Profiles

	Action	Profile Name ↑	User License	Custom
□	Edit | Clone	High Volume Customer Portal User	High Volume Customer Portal	□
□	Edit | Clone	Identity User	Identity	□
☑	Edit | Del |...	Manager	Salesforce	✓
□	Edit | Clone	Marketing User	Salesforce	□
□	Edit | Clone	Minimum Access - API Only Integrations	Salesforce Integration	□
□	Edit | Clone	Minimum Access - Salesforce	Salesforce	□
□	Edit | Clone	Partner App Subscription User	Partner App Subscription	□

Fig :9.1 Manager Profile



SETUP

Profiles

Profiles

Help for this Page

All Profiles | Edit | Delete | Create New View

New Profile

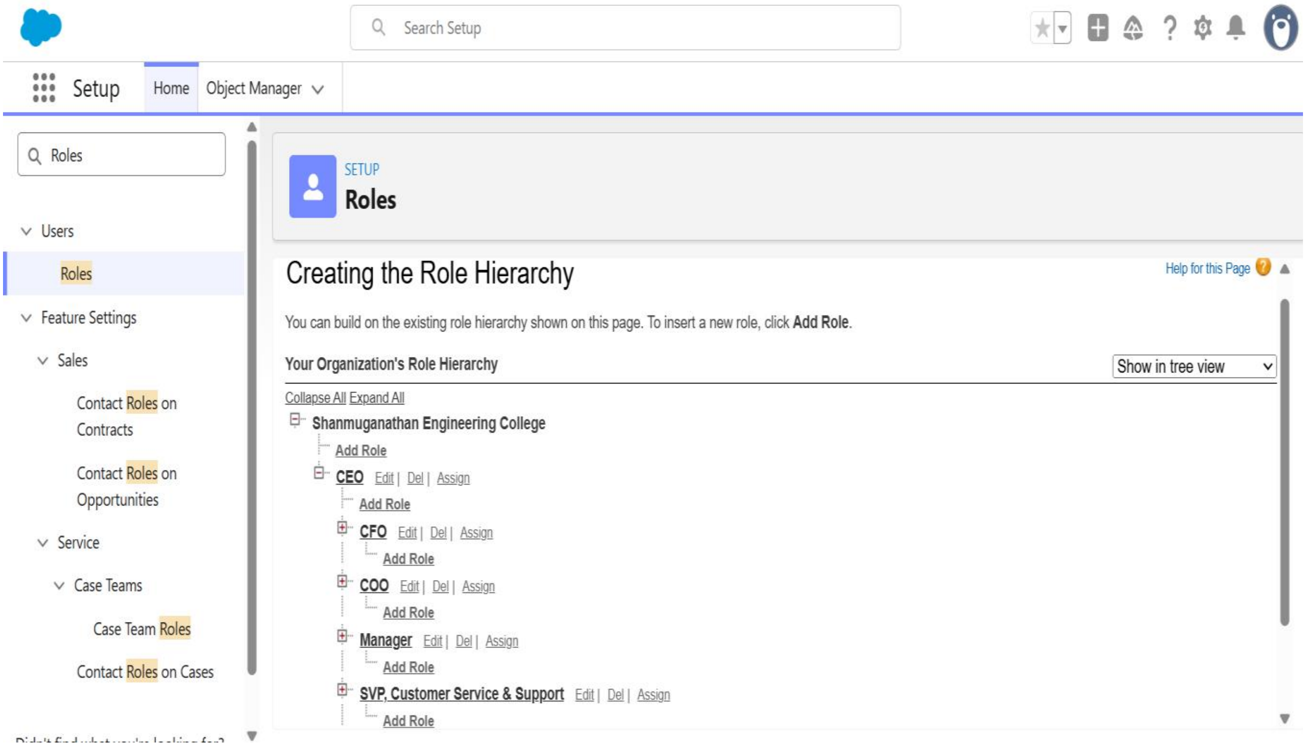
A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | Other | **All**

☐ Action	Profile Name ↑	User License	Custom
☐ Edit | Clone	Partner App Subscription User	Partner App Subscription	☐
☐ Edit | Clone	Partner Community Login User	Partner Community Login	☐
☐ Edit | Clone	Partner Community User	Partner Community	☐
☐ Edit | Del |...	Read Only	Salesforce	✓
☑ Edit | Del |...	sales person	Salesforce Platform	✓
☐ Edit | Del |...	Salesforce API Only System Integrations	Salesforce Integration	✓
☐ Edit | Clone	Silver Partner User	Silver Partner	☐

1-45 of 45   2 Selected   Previous Next   Page 1 of 1

Fig :9.2 Salesperson Profile

10.Role and Role Hierarchy:



Search Setup

Setup   Home   Object Manager

Roles

- Users
  - Roles
- Feature Settings
  - Sales
    - Contact Roles on Contracts
    - Contact Roles on Opportunities
  - Service
    - Case Teams
      - Case Team Roles
    - Contact Roles on Cases

SETUP

Roles

Creating the Role Hierarchy

Help for this Page

You can build on the existing role hierarchy shown on this page. To insert a new role, click **Add Role**.

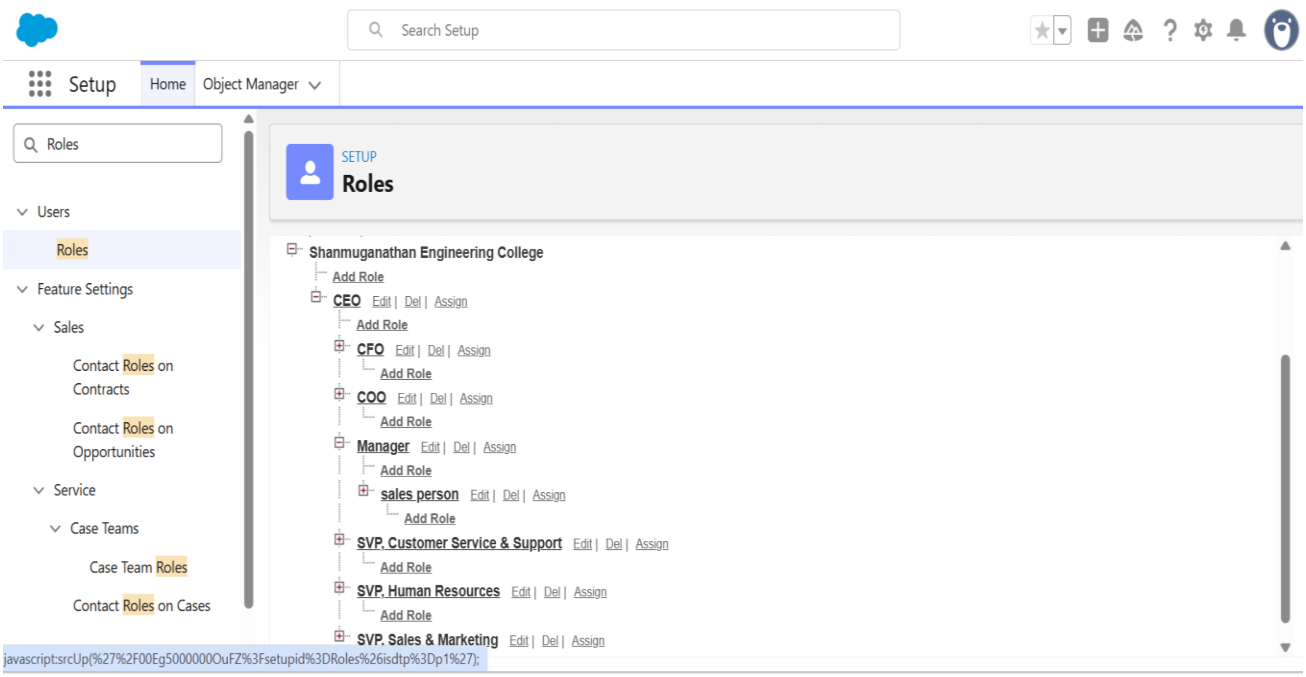
Your Organization's Role Hierarchy

Show in tree view

Collapse All Expand All

- **Shanmuganathan Engineering College**
  - Add Role
  - **CEO** Edit | Del | Assign
    - Add Role
    - **CFO** Edit | Del | Assign
      - Add Role
    - **COO** Edit | Del | Assign
      - Add Role
    - **Manager** Edit | Del | Assign
      - Add Role
    - **SVP, Customer Service & Support** Edit | Del | Assign
      - Add Role

Fig :10.1 Manager Role

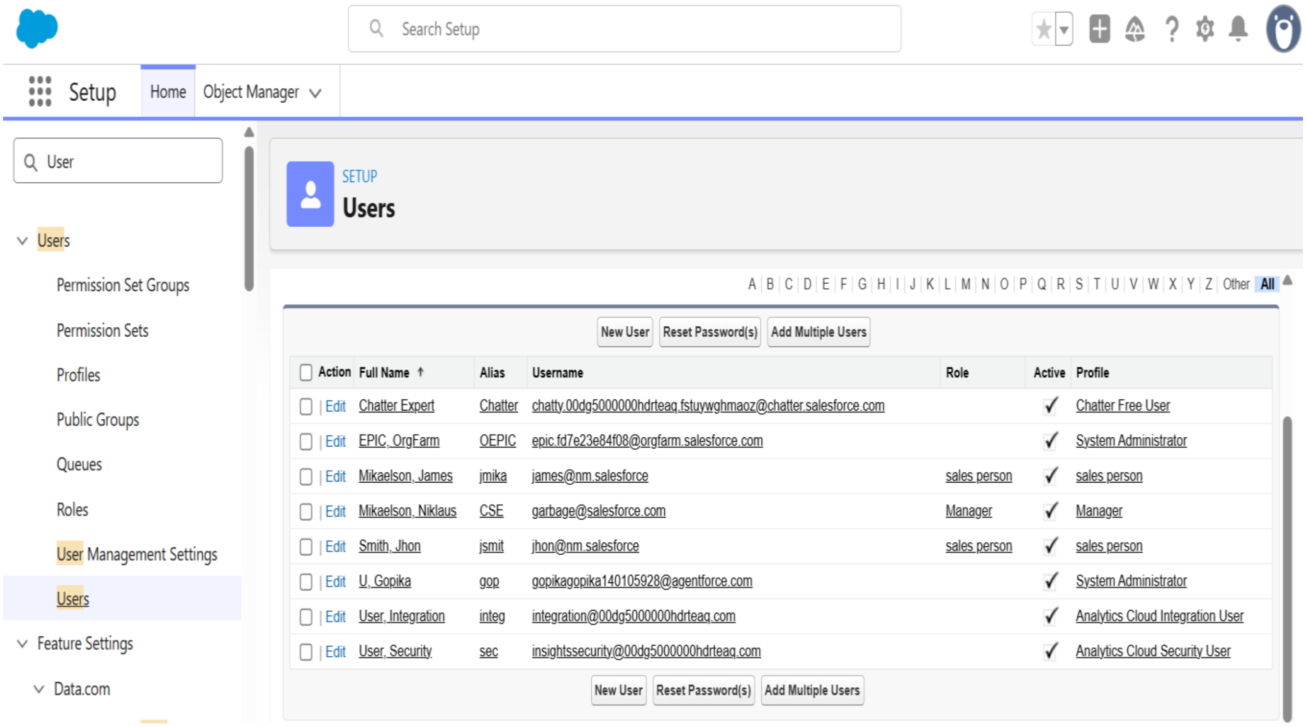


- Shanmuganathan Engineering College
  - Add Role
  - CEO Edit | Del | Assign
    - Add Role
    - CFO Edit | Del | Assign
      - Add Role
    - COO Edit | Del | Assign
      - Add Role
    - Manager Edit | Del | Assign
      - Add Role
      - sales person Edit | Del | Assign
        - Add Role
    - SVP, Customer Service & Support Edit | Del | Assign
      - Add Role
    - SVP, Human Resources Edit | Del | Assign
      - Add Role
    - SVP, Sales & Marketing Edit | Del | Assign

javascript:srcUp(%27%2F00Eg5000000OuFZ%3Fsetupid%3DRoles%26isdtp%3Dp1%27);

Fig :10.2 Salesperson Role

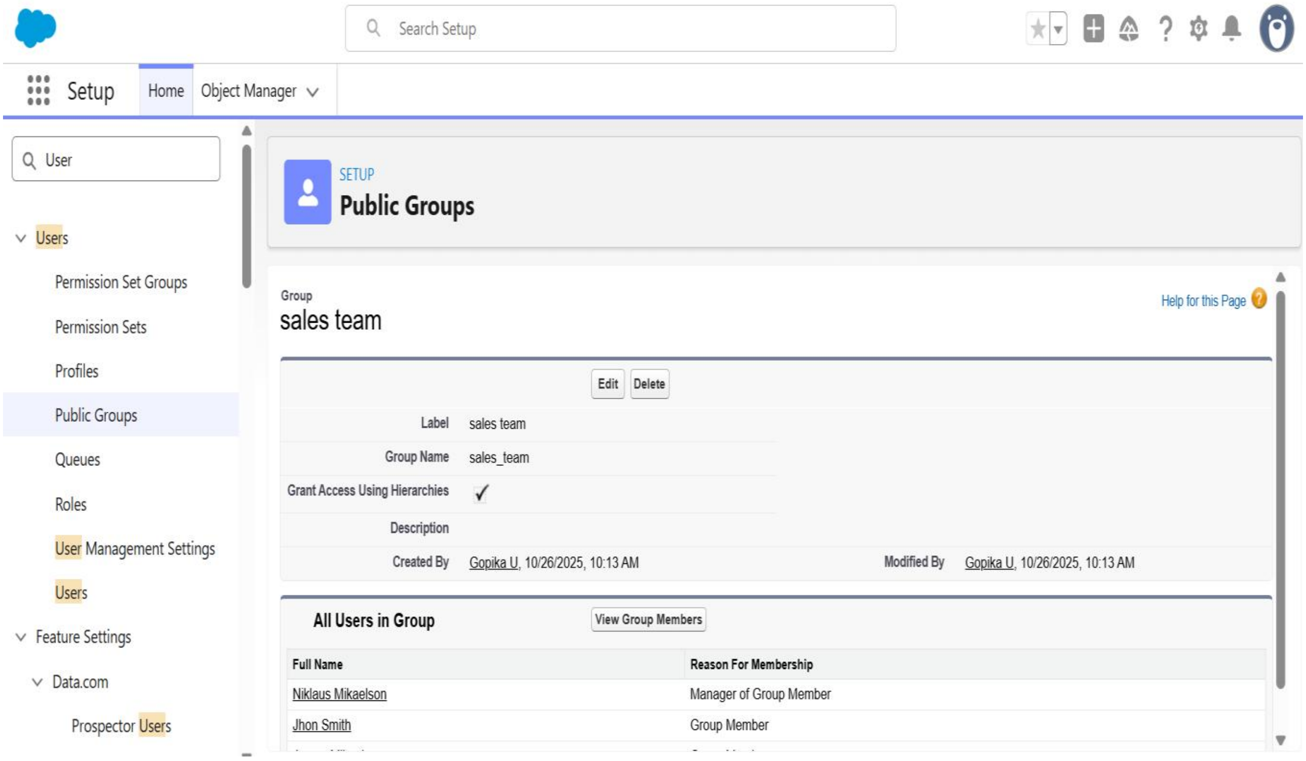
11.Users:



□ Action	Full Name ↑	Alias	Username	Role	Active	Profile
□ | Edit	Chatter Expert	Chatter	chatty.00dg5000000hdrteaq.fstuywghmaoz@chatter.salesforce.com		✓	Chatter Free User
□ | Edit	EPIC, OrgFarm	OEPIC	epic.fd7e23e84f08@orgfarm.salesforce.com		✓	System Administrator
□ | Edit	Mikaelson, James	jmika	james@nm.salesforce	sales person	✓	sales person
□ | Edit	Mikaelson, Niklaus	CSE	garbage@salesforce.com	Manager	✓	Manager
□ | Edit	Smith, Jhon	jsmit	jhon@nm.salesforce	sales person	✓	sales person
□ | Edit	U, Gopika	gop	gopikagopika140105928@agentforce.com		✓	System Administrator
□ | Edit	User, Integration	integ	integration@00dg5000000hdrteaq.com		✓	Analytics Cloud Integration User
□ | Edit	User, Security	sec	insightssecurity@00dg5000000hdrteaq.com		✓	Analytics Cloud Security User

Fig :11.1 Creating Users

12.Public Groups:



SETUP
Public Groups

Group
sales team

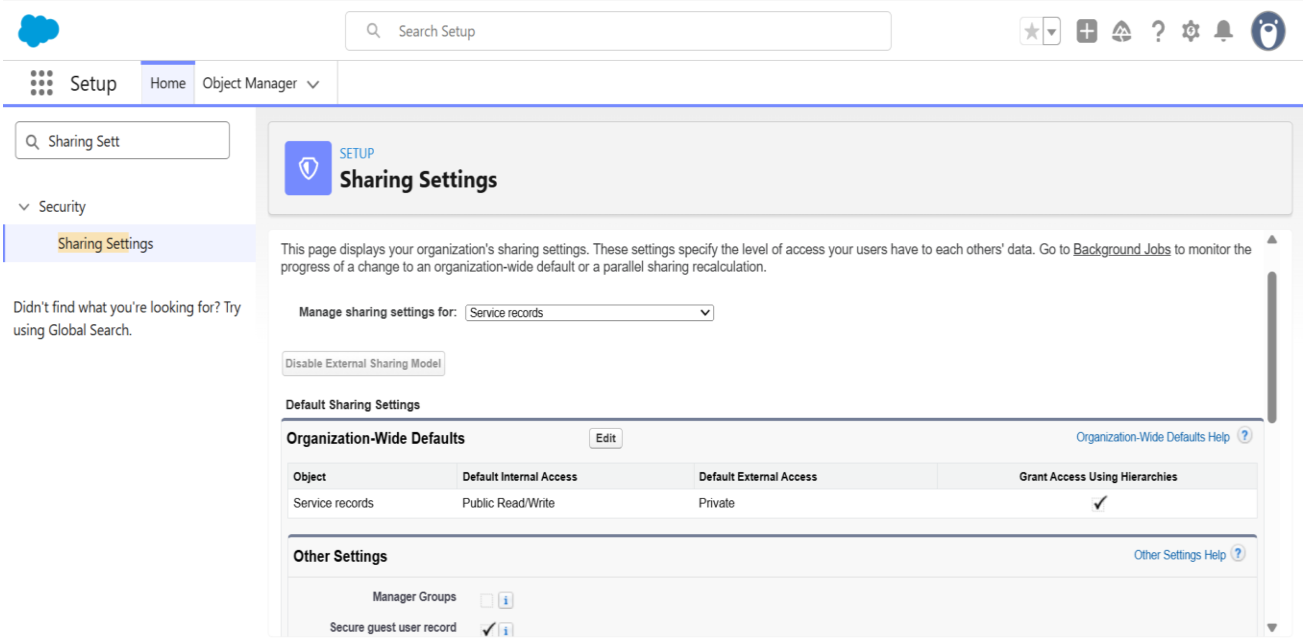
Label	sales team
Group Name	sales_team
Grant Access Using Hierarchies	✓
Description	
Created By	Gopika U, 10/26/2025, 10:13 AM
Modified By	Gopika U, 10/26/2025, 10:13 AM

All Users in Group

Full Name	Reason For Membership
Niklaus Mikaelson	Manager of Group Member
Jhon Smith	Group Member

Fig:12.1 New Public Group

13.Sharing Setting:



SETUP
Sharing Settings

Didn't find what you're looking for? Try using Global Search.

This page displays your organization's sharing settings. These settings specify the level of access your users have to each others' data. Go to Background Jobs to monitor the progress of a change to an organization-wide default or a parallel sharing recalculation.

Manage sharing settings for: Service records

Default Sharing Settings

Organization-Wide Defaults

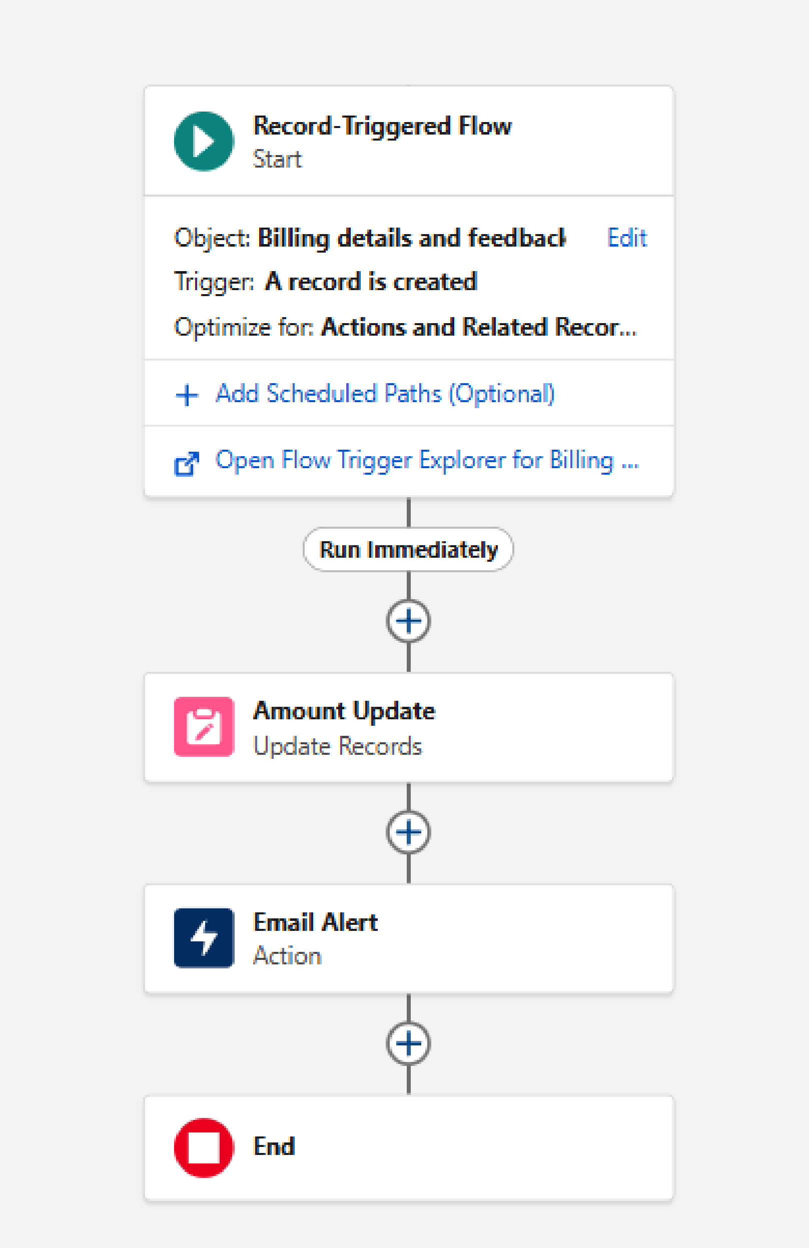
Object	Default Internal Access	Default External Access	Grant Access Using Hierarchies
Service records	Public Read/Write	Private	✓

Other Settings

Manager Groups	□
Secure guest user record	✓

Fig:13.1 Creating Sharing Setting

14.Flows:



Record-Triggered Flow
Start

Object: Billing details and feedbacl Edit

Trigger: A record is created

Optimize for: Actions and Related Recor...

+ Add Scheduled Paths (Optional)
Open Flow Trigger Explorer for Billing ...
Run Immediately

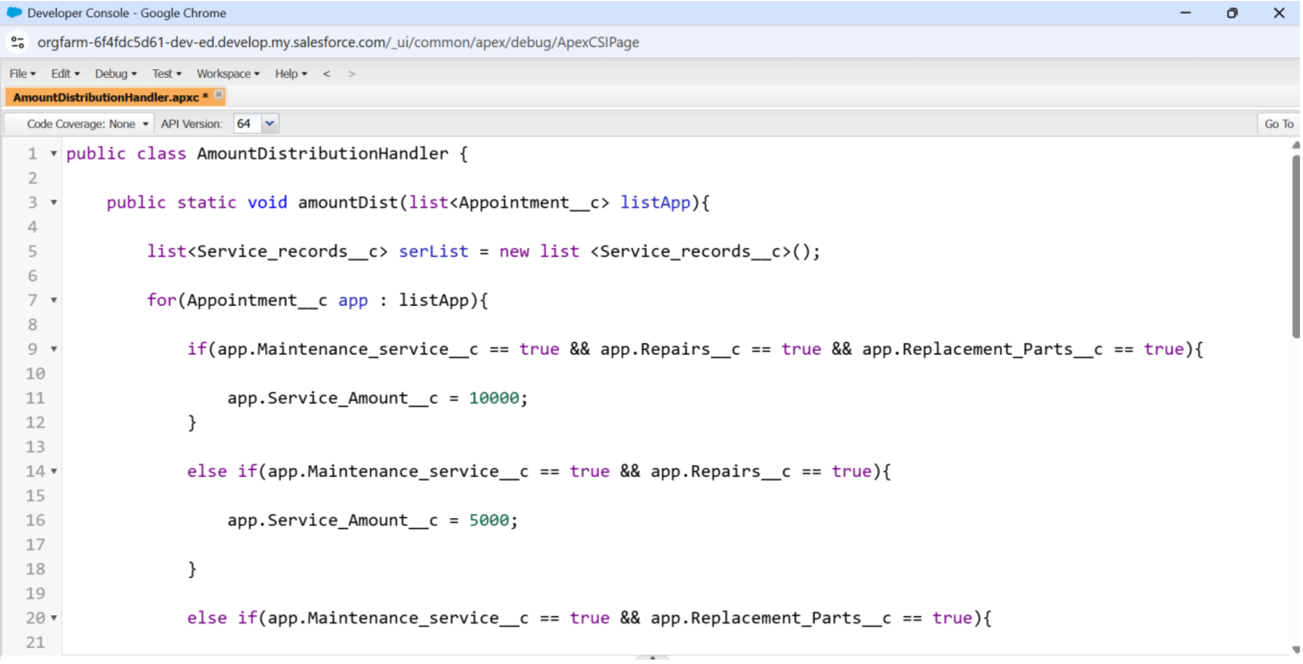
Amount Update
Update Records

Email Alert
Action

End

Fig:14.1 Creating a flow

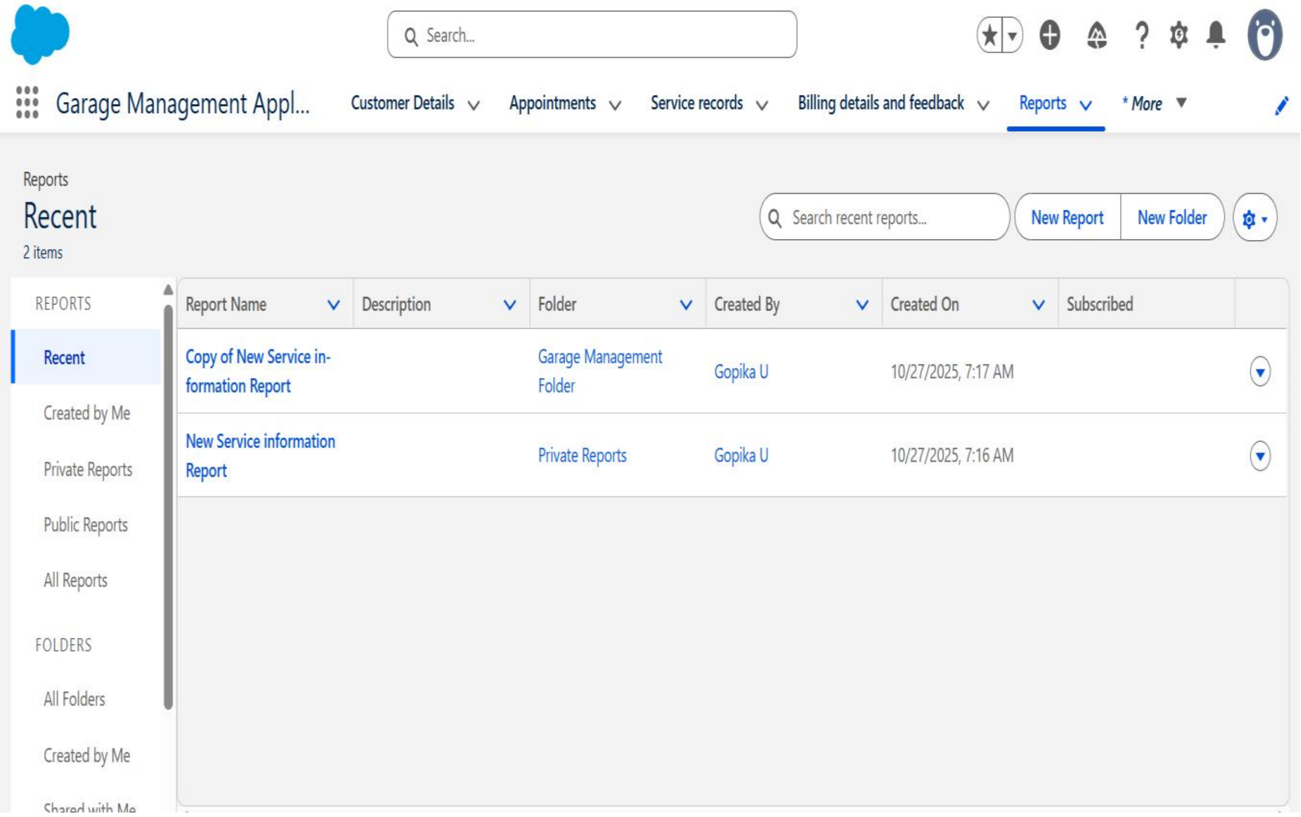
15.Apex Trigger:


Developer Console - Google Chrome
orgfarm-6f4fdc5d61-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage
File ▾ Edit ▾ Debug ▾ Test ▾ Workspace ▾ Help ▾ < >
AmountDistributionHandler.apxc *
Code Coverage: None ▾ API Version: 64
Go To

```
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.Repairs__c == 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){

```

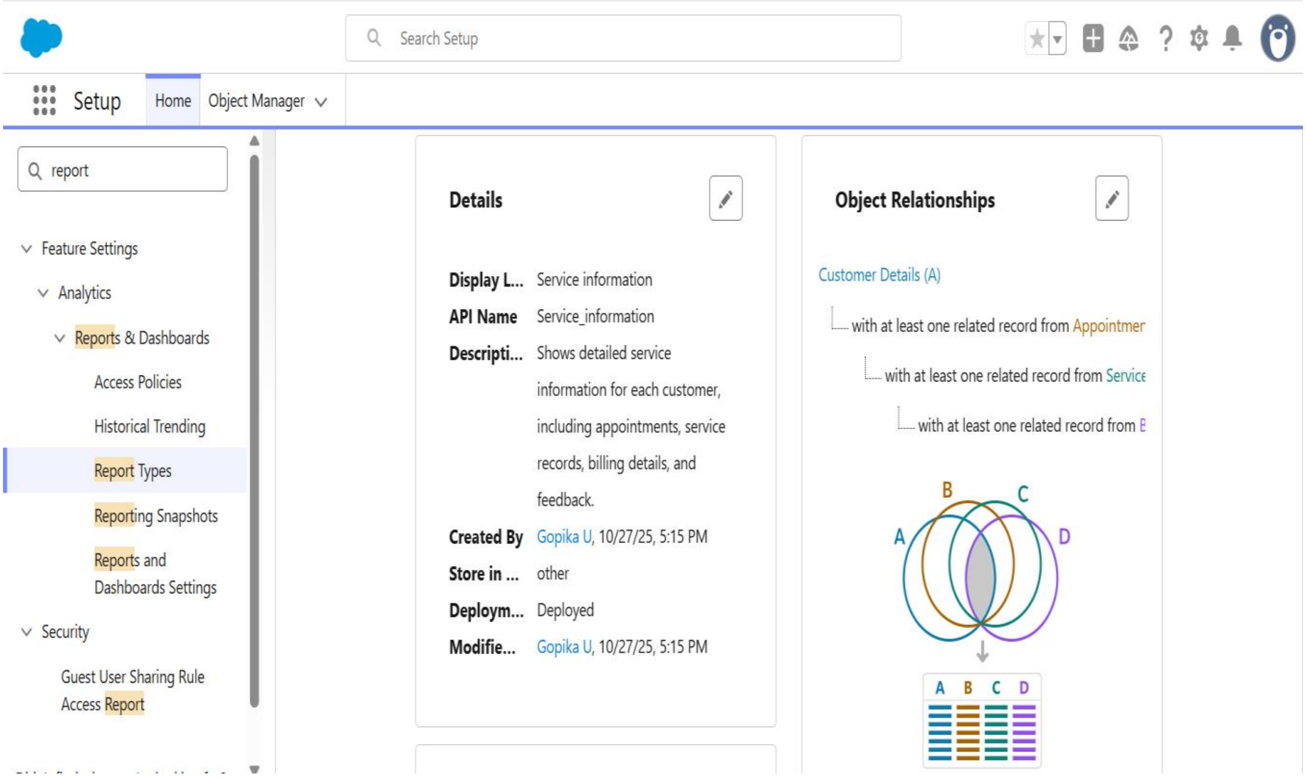
Fig:15.1 Apex Handler

16.Reports:



Report Name	Description	Folder	Created By	Created On	Subscribed
Copy of New Service information Report		Garage Management Folder	Gopika U	10/27/2025, 7:17 AM	
New Service information Report		Private Reports	Gopika U	10/27/2025, 7:16 AM	

Fig:16.1 Report Folder



Details

Display L...	Service information
API Name	Service_information
Descripti...	Shows detailed service information for each customer, including appointments, service records, billing details, and feedback.
Created By	Gopika U, 10/27/25, 5:15 PM
Store in ...	other
Deploym...	Deployed
Modifie...	Gopika U, 10/27/25, 5:15 PM

Object Relationships

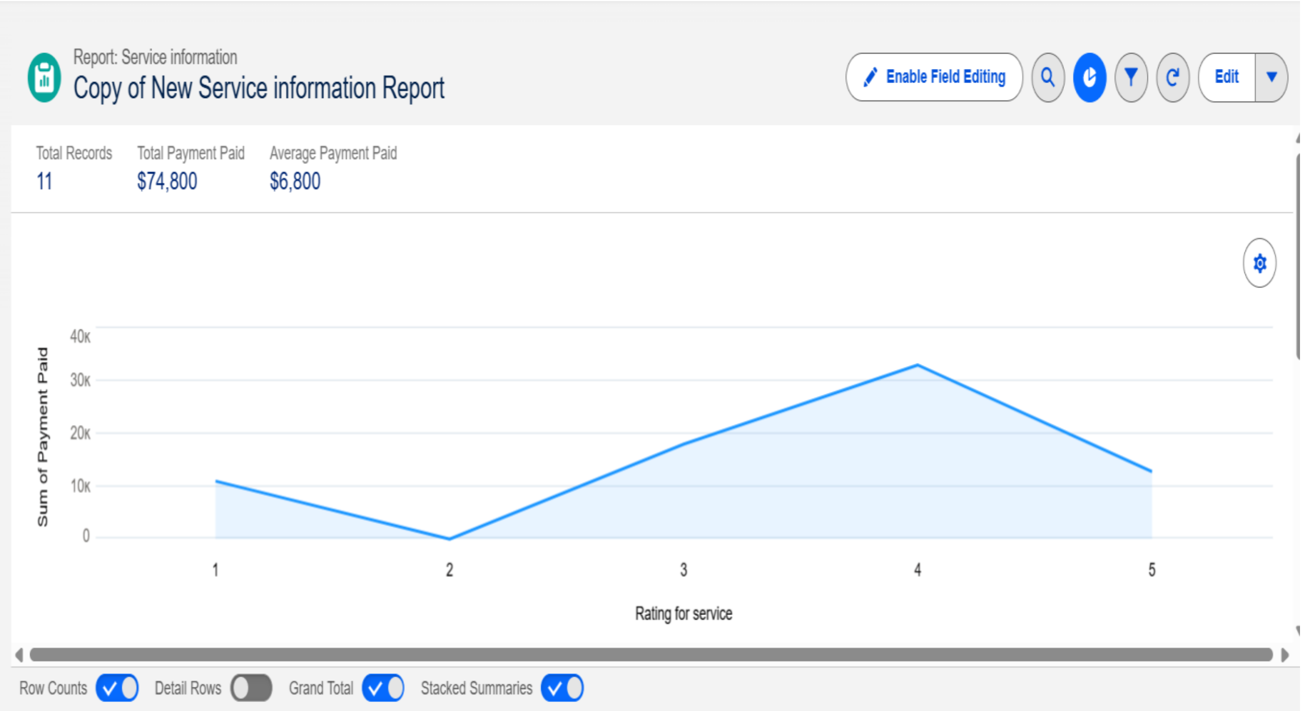
Customer Details (A)

with at least one related record from Appointmer

with at least one related record from Service

with at least one related record from E

Fig:16.2 Report Types



Report: Service information

Copy of New Service information Report

Total Records	Total Payment Paid	Average Payment Paid
11	$74,800	$6,800

Row Counts ✓ Detail Rows ○ Grand Total ✓ Stacked Summaries ✓

Fig:16.3 Creation of Report

17.Dashboard:

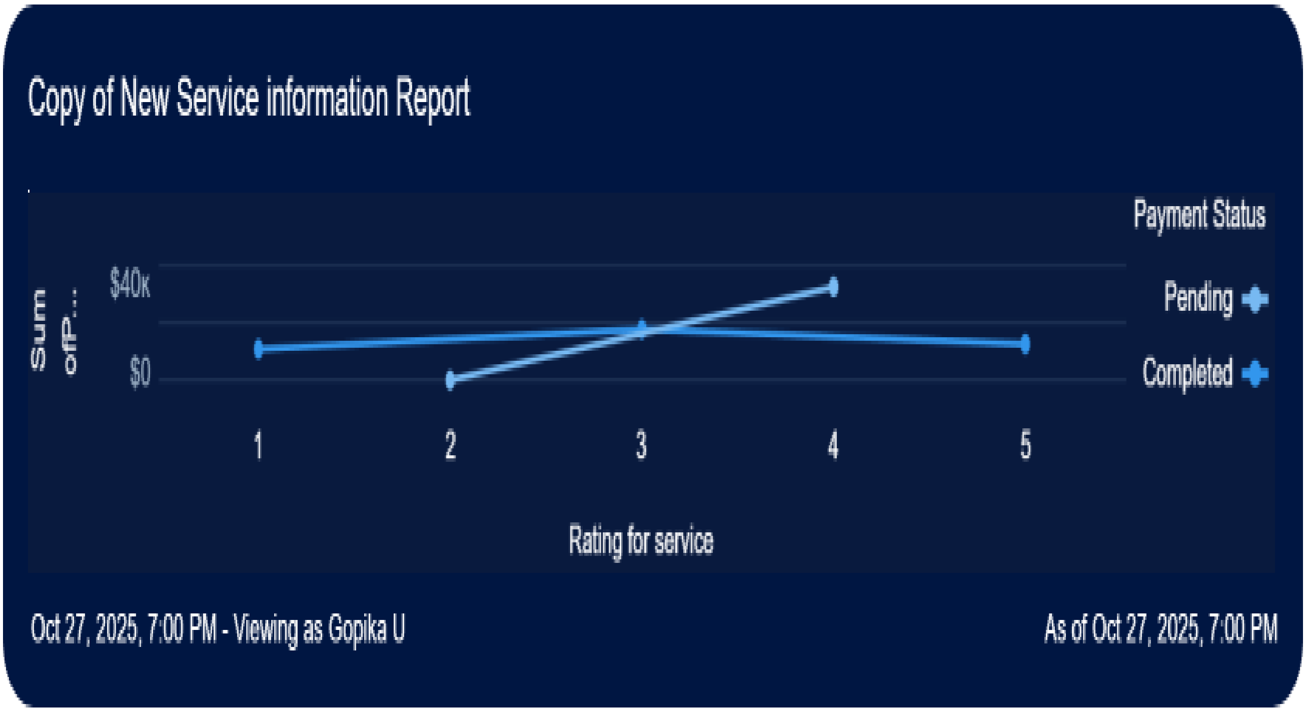
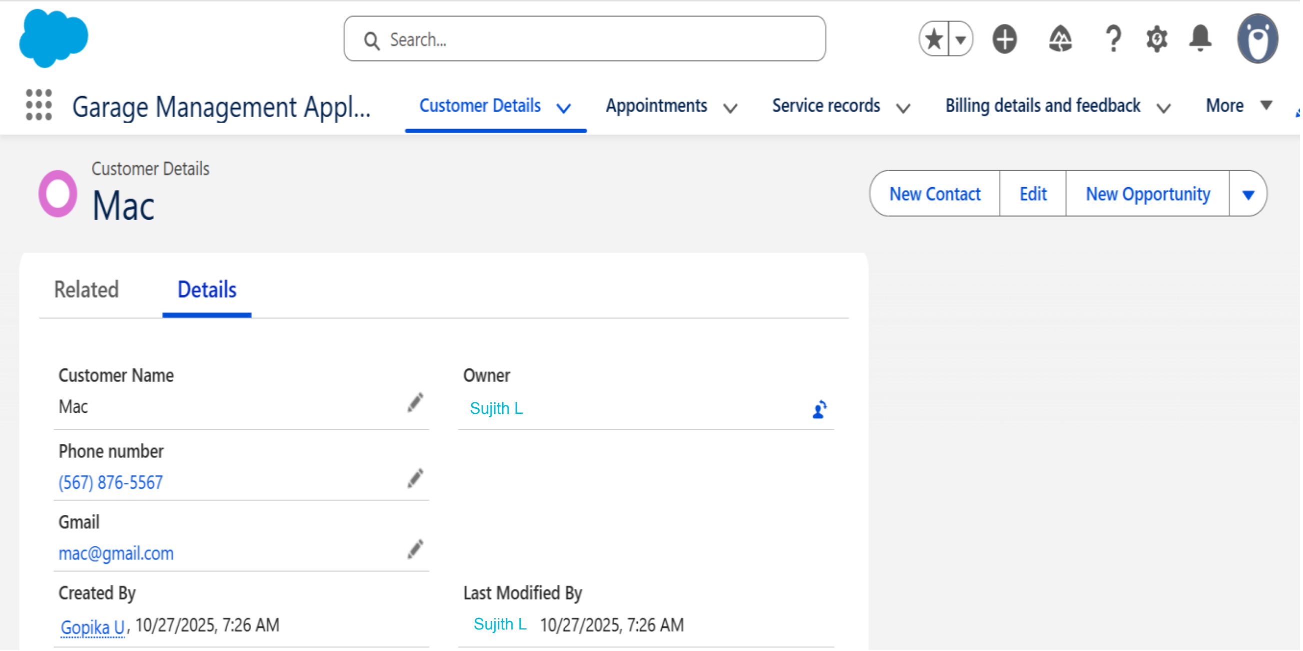


Fig:17.1 Creation of Dashboard

18.User Adoption:



Customer Details

Mac

Customer Name
Mac

Owner
Sujith L

Phone number
(567) 876-5567

Gmail
mac@gmail.com

Created By
Gopika U, 10/27/2025, 7:26 AM

Last Modified By
Sujith L 10/27/2025, 7:26 AM

Fig:18.1 Creating Records

8. EXPECTED OUTCOMES

The **Garage Management System (GMS)** developed using Salesforce is expected to deliver a range of outcomes that enhance the overall efficiency and customer experience within automotive service centers. By integrating automation, cloud technology, and CRM features, the system transforms traditional manual workflows into a digital, intelligent, and reliable process.

The first major outcome is the **automation of garage operations**. Through Salesforce Flows, Apex Triggers, and Validation Rules, all major processes — including customer registration, appointment scheduling, service tracking, billing, and feedback — are automated. This reduces manual effort, ensures faster execution, and minimizes human error across the workflow.

The second expected outcome is **improved data accuracy and transparency**. Salesforce's in-built validation and duplicate management features ensure that every piece of data stored in the system is verified and consistent. Managers can monitor service records, payments, and customer feedback in real time, promoting accountability and trust between staff and customers.

A third key result is **simplified management and faster billing**. The system automatically calculates the total service amount based on the selected services, generates invoices instantly, and updates payment records upon completion. This automation significantly reduces billing time, providing a smooth experience for both customers and garage staff.

The project also ensures **high scalability and reliability** through Salesforce's cloud infrastructure. As the business grows, new users, branches, or services can be easily integrated without additional system reconfiguration. The multi-tenant cloud model ensures secure data handling, automatic backups, and uninterrupted access from anywhere.

Finally, the system enhances **customer engagement and satisfaction** through automated notifications and feedback collection. Customers receive instant email alerts regarding appointment confirmations, service completion, and payment updates, ensuring transparency and consistent communication.

Overall, the expected outcomes of the project include improved operational efficiency, better decision-making through analytics, and an enhanced customer relationship experience. The Garage Management System thus serves as a complete, scalable, and future-ready solution for modern automotive service centers.

10. ADVANTAGES

The **Garage Management System (GMS)** built on the **Salesforce platform** provides several advantages that make it a powerful, efficient, and modern solution for managing automotive service operations. Its cloud-based nature, automation capabilities, and integrated analytics ensure improved productivity and customer satisfaction.

One of the primary advantages is that the system is **cloud-based and accessible from anywhere**. Since Salesforce operates entirely on the cloud, authorized users can access garage data securely from any device with an internet connection. This allows managers, staff, and customers to stay connected at all times, ensuring flexibility and continuity of operations without the need for physical infrastructure or local installations.

Another key advantage is **high security and data protection**. Salesforce provides multiple layers of security, including role-based access control, data encryption, and secure authentication. Each user has a defined level of access, which prevents unauthorized data manipulation. This ensures that sensitive customer and billing information is always protected.

The system also offers **extensive automation**, which significantly reduces manual work. Tasks such as calculating service charges, updating payment status, generating invoices, and sending email alerts are handled automatically through Salesforce Flows and Apex Triggers. This leads to faster service delivery, fewer human errors, and more accurate records.

A further advantage is the **ease of customization and scalability**. The system can be easily modified to meet specific business requirements—new services, fields, or workflows can be added without extensive coding. Salesforce's flexible architecture allows the same system to scale from a single small garage to a multi-branch enterprise with minimal effort.

The **reporting and dashboard features** are another major benefit. Managers can generate analytical reports and visualize real-time performance metrics such as total revenue, service ratings, and completed appointments. These insights help in data-driven decision-making and continuous process improvement.

The system also contributes to **enhanced customer engagement**. Automated notifications, timely service updates, and post-service feedback collection build strong communication between the garage and its customers. This not only improves transparency but also strengthens trust and long-term relationships.

Lastly, the system supports **environmental sustainability** by minimizing the need for paper-based records and manual documentation. All data is stored digitally in the Salesforce cloud, promoting an eco-friendly and organized working environment.

11. FUTURE ENHANCEMENT

While the **Garage Management System (GMS)** developed using Salesforce successfully automates the core operations of automotive service centers, there remains ample scope for further enhancement to make the system more intelligent, user-friendly, and connected to emerging technologies. The following future improvements can extend the project's functionality and make it even more impactful in real-world scenarios.

A key enhancement would be the **integration of online payment gateways** such as PayPal, Razorpay, or UPI-based systems. This would allow customers to make secure digital payments directly from the system, enabling real-time billing and reducing the dependency on manual cash handling. Automatic receipts and payment confirmations could also be generated and sent via email or SMS.

Another valuable improvement is the **development of a dedicated mobile application** using Salesforce Mobile or Experience Cloud. This would allow customers to book appointments, check service status, and provide feedback conveniently from their smartphones. Service advisors and mechanics could also use the mobile app to update records instantly, improving communication and workflow efficiency on the go.

The system can further evolve through **integration with Internet of Things (IoT)** devices. Modern vehicles are equipped with sensors that can transmit data about fuel levels, engine health, and maintenance requirements. Connecting these IoT signals to Salesforce could enable automatic service scheduling or alerts for preventive maintenance, thereby reducing breakdowns and enhancing customer trust.

Another major future direction is the **use of Artificial Intelligence (AI) and Predictive Analytics** through Salesforce Einstein. By analyzing service patterns, customer feedback, and vehicle history, the AI module could predict future service needs, identify potential issues before they occur, and recommend personalized offers or maintenance schedules to customers. This would add an intelligent decision-making layer to the system.

Expanding the project to include **inventory and spare parts management** would further strengthen its capabilities. Tracking parts availability, ordering new stock automatically, and linking inventory data with service records can help garages operate more efficiently while reducing downtime due to unavailable components.

In addition, **multi-language and regional support** can make the application more inclusive, allowing users from different locations and linguistic backgrounds to use the system comfortably. Salesforce's localization features can easily support this enhancement, improving accessibility for diverse customer bases.

Finally, the system could be extended to support **franchise or multi-branch operations**, where multiple garages under the same brand are managed through a centralized Salesforce org. This would enable unified reporting, performance monitoring, and customer tracking across different service centers.

In conclusion, these enhancements — including digital payment integration, mobile accessibility, AI-based predictions, IoT connectivity, and multi-branch scalability — will elevate the **Garage Management System** from a functional automation platform to a **smart, intelligent, and future-ready enterprise solution**. Implementing these features will help garages adopt a truly digital and data-driven approach, aligning with the vision of next-generation cloud-based business transformation.

12. CONCLUSION

The **Garage Management System (GMS)** developed using **Salesforce** successfully demonstrates how cloud-based technologies can transform traditional garage operations into an automated, efficient, and customer-centric process. The project integrates key Salesforce features such as **custom objects, validation rules, flows, triggers, reports, and dashboards**, enabling a complete digital solution for managing customers, appointments, services, and billing.

Through the implementation of automation tools, the system eliminates manual errors, reduces operational time, and ensures seamless coordination between different functional modules. Managers can easily monitor ongoing services, generate analytical reports, and make data-driven decisions, while customers benefit from transparency, timely updates, and accurate billing.

The project also highlights the potential of **Salesforce as a low-code platform**, showcasing its adaptability beyond conventional CRM functions. It serves as a practical example of how industry-relevant technologies can be used in real-world applications to improve productivity and service quality.

Overall, the Garage Management System stands as an innovative and scalable solution that not only simplifies business processes but also strengthens customer relationships. This project experience has enhanced technical knowledge in Salesforce development and provided valuable insights into the importance of automation and data management in modern business environments.

13. REFERENCES

1. Salesforce Developer Guide — https://developer.salesforce.com/docs
2. Salesforce Trailhead — https://trailhead.salesforce.com
3. Apex Triggers and Flow Tutorials — YouTube
4. CRM Best Practices — Salesforce Admin Guide
5. Naan Mudhalvan Project Portal — Skill Development Materials