

An Internship Report
On
Salesforce AI Developer Virtual Internship
Submitted In accordance with the requirement for the degree of
BACHELOR OF TECHNOLOGY
INFORMATION TECHNOLOGY
Submitted by
TAMMA SREYA (218X1A1260)
BADUGU RENUKA (218X1A1203)
SAMPATHI KALYAN CHAKRAVARTHI (218X1A1231)
NAKKALA CHAITANYA KRISHNA (228X5A1204)



DEPARTMENT OF INFORMATION TECHNOLOGY
KALLAM HARANADHAREDDY INSTITUTE OF TECHNOLOGY
(AUTONOMOUS)

Approved by (AICTE, New Delhi; Permanently Affiliated to JNTU KAKINADA)

Accredited by NAAC with an 'A' Grade

NH – 16, Chowdavaram, Guntur – 522019 (A.P)

ACADEMIC YEAR: 2024-2025

Project Title : Garage Management System.

Project Overview : The Garage Management System (GMS) is designed to streamline daily operations in automotive repair shops, providing a comprehensive solution for managing customer interactions, vehicle repairs, inventory, and billing. This system offers a central platform where staff can easily track service requests, schedule appointments, and maintain detailed records of repairs, ensuring nothing slips through the cracks.

With features like automated reminders, parts tracking, and job status updates, GMS not only enhances operational efficiency but also improves customer satisfaction by providing transparency and timely communication. Furthermore, the system integrates with accounting tools, enabling seamless invoicing and payment processing, which simplifies financial management for the business.

By providing real-time insights into repair status, resource utilization, and inventory levels, GMS empowers garage owners and managers to make informed decisions, optimize workflow, and reduce downtime. This all-in-one solution helps garages focus on what matters most: delivering high-quality service, fostering customer loyalty, and growing the business. With its scalability, GMS is suitable for both small shops and large automotive service chains, making it an essential tool for garages aiming for success in a fast-paced industry.

Salesforce

Introduction:

Are you new to Salesforce? Not sure exactly what it is, or how to use it? Don't know where you should start on your learning journey? If you've answered yes to any of these questions, then you're in the right place. This module is for you.

Welcome to Salesforce! Salesforce is game-changing technology, with a host of productivity-boosting features, that will help you sell smarter and faster. As you work toward your badge for this module, we'll take you through these features and answer the question, "What is Salesforce, anyway?".

What Is Salesforce?

Salesforce is your customer success platform, designed to help you sell, service, market, analyze, and connect with your customers.

Salesforce has everything you need to run your business from anywhere. Using standard products and features, you can manage relationships with prospects and customers, collaborate and engage with employees and partners, and store your data securely in the cloud.

Objectives:

Goals : The GMS project aims to enhance operational efficiency, improve customer service, optimize inventory management, provide real-time data insights, ensure scalability, promote collaboration, and increase transparency and accountability.

Expected Outcomes : Expected outcomes include a 30% decrease in manual tasks, 20% reduction in errors, 25% increase in customer retention, 15% increase in repeat business, 20% reduction in service delays, 15% reduction in inventory costs, 25% improvement in decision-making accuracy, 10% increase in profitability, support for a 50% increase in business volume, 30% reduction in communication issues, 20% increase in team efficiency, 25% reduction in service disputes, and 15% increase in employee satisfaction.

Key Features :

1. User Management:

- **Multiple User Roles:** Systems like Garage Buddy support multiple user accounts, allowing mechanics and employees to have their own personalized access. This is crucial for managing different roles such as managers, mechanics, receptionists, and supervisors.

2. Service Tracking:

- **Detailed Service Logs:** GMS solutions like Garage Buddy help maintain a complete history of each vehicle's services, from routine maintenance to complex repairs. This ensures better customer service and helps in tracking service dates.

3. Inventory Management:

- **Vehicle Spare Parts:** GMS allows for efficient management of vehicle spare parts inventory. Supervisors can check inventory levels and track which vehicles are being serviced.

4. Appointments:

- **Appointment Management:** GMS enables users to book appointments in advance, making it easier to schedule services. This helps in managing the workflow and reducing wait times.

5. Reports and Dashboards:

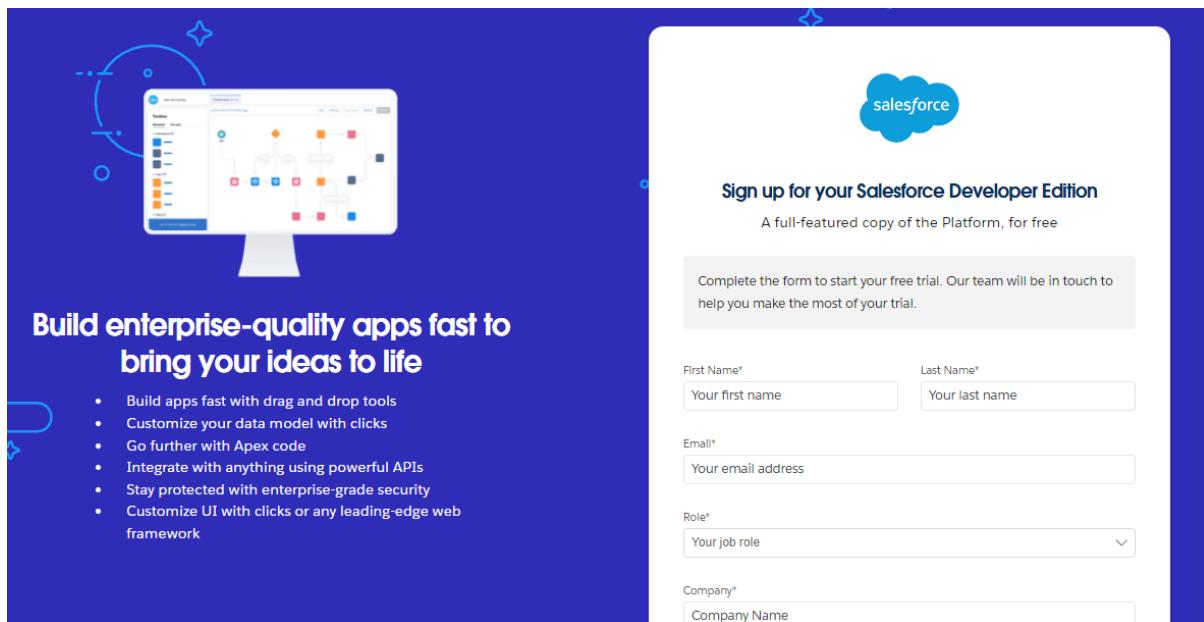
- **Data Visualization:** GMS solutions provide detailed reports and dashboards for real-time tracking of business operations. These tools help in analyzing service performance metrics and making informed decisions.

Milestone 1: Creating Developer Account

Creating a developer org in salesforce.

1. Go to <https://developer.salesforce.com/signup>

2. On the sign up form, enter the following details :



Then Fill the Details

1. First name & Last name : Internship Project

2. Email : 218x1a1260@khitguntur.ac.in

3. Role : Developer

4. Company : KHIT

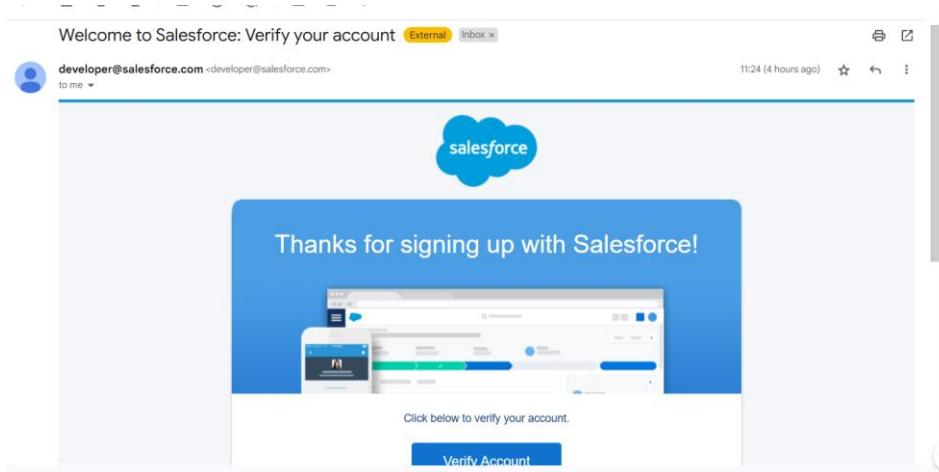
5. County : India

6. Postal Code : 522211

7. Username : garagemanagementsystem@khit.com

Account Activation

1. Go to the inbox of the email that you used while signing up. Click on the verify account to activate your account. The email may take 5-10mins.



2. Click on Verify Account
3. Give a password and answer a security question and click on change password.

Milestone 2 : Object

Create Customer DetailsObject

To create an object:

1. From the setup page >> Click on Object Manager >> Click on Create >> Click on Custom Object.
 - a) Enter the label name >> Customer Details
 - b) Plural label name >> Customer Details
 - c) Enter Record Name Label and Format
 - Record Name >> Customer Name
 - Data Type >> Text
2. Click on Allow reports and Track Field History,
3. Allow search >> Save.

The screenshot shows the Salesforce Setup interface with the 'Customer Details' object selected. The left sidebar lists various configuration tabs: Fields & Relationships, Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, Restriction Rules, Scoping Rules, and Object Access. The main panel is titled 'Details' and contains fields for API Name ('Customer_Details__c'), Singular Label ('Customer Details'), and Plural Label ('Customer Details'). On the right, there are sections for 'Enable Reports' (checked), 'Track Activities', 'Track Field History' (checked), 'Deployment Status' (set to 'Deployed'), and 'Help Settings'. At the bottom right are 'Edit' and 'Delete' buttons.

Create Appointment Object

To create an object:

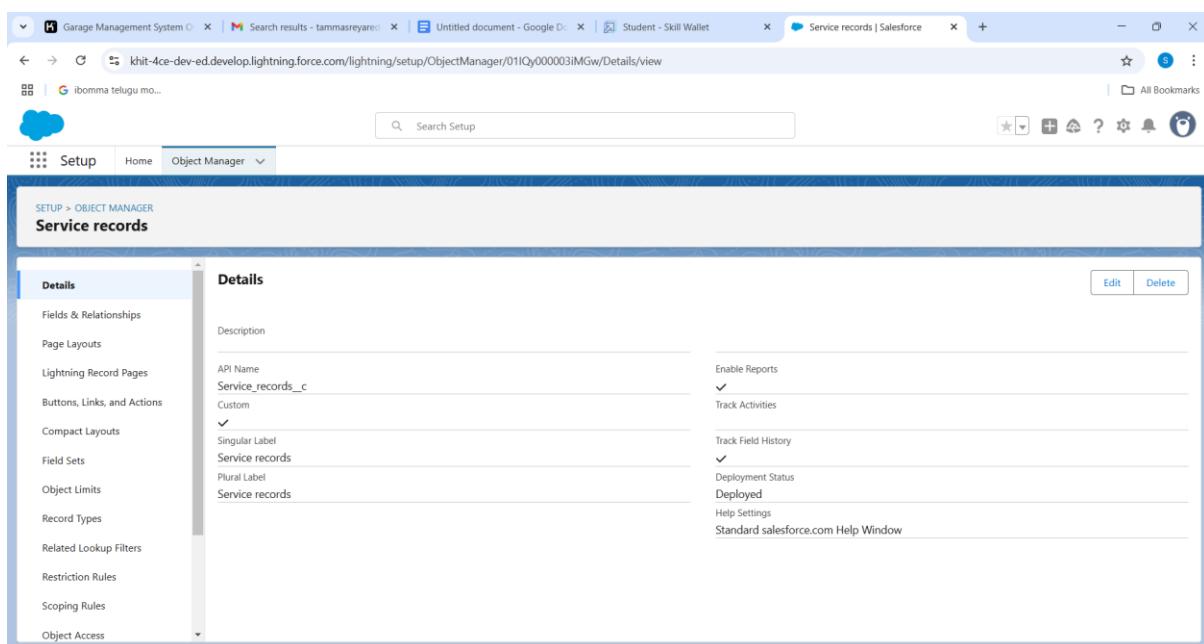
1. From the setup page >> Click on Object Manager >> Click on Create >> Click on Custom Object.
 - a) Enter the label name >> Appointment
 - b) Plural label name >> Appointments
 - c) Enter Record Name Label and Format
 - Record Name >> Appointment Name
 - Data Type >> Auto Number
 - Display Format >> app-{000}
 - Starting number >> 1
2. Click on Allow reports and Track Field History,
3. Allow search >> Save.

The screenshot shows the Salesforce Setup interface with the 'Appointment' object selected. The left sidebar lists various configuration tabs: Fields & Relationships, Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, Restriction Rules, Scoping Rules, and Object Access. The main panel is titled 'Details' and contains fields for API Name ('Appointment__c'), Singular Label ('Appointment'), and Plural Label ('Appointments'). On the right, there are sections for 'Enable Reports' (checked), 'Track Activities', 'Track Field History' (checked), 'Deployment Status' (set to 'Deployed'), and 'Help Settings'. At the bottom right are 'Edit' and 'Delete' buttons.

Create Service records Object

To create an object:

1. From the setup page >> Click on Object Manager >> Click on Create >> Click on Custom Object.
 - a) Enter the label name >> Service records
 - b) Plural label name >> Service records
 - c) Enter Record Name Label and Format
 - Record Name >>Service records Name
 - Data Type >> Auto Number
 - Display Format >> ser-{000}
 - Starting number >> 1
2. Click on Allow reports and Track Field History,
3. Allow search >> Save



Create Billing details and feedback Object

To create an object:

1. From the setup page >> Click on Object Manager >> Click on Create >> Click on Custom Object.
 - a) Enter the label name >> Billing details and feedback
 - b) Plural label name >> Billing details and feedback
 - c) Enter Record Name Label and Format
 - Record Name >> Billing details and feedback Name
 - Data Type >> Auto Number
 - Display Format >> bill-{000}
 - Starting number >> 1
2. Click on Allow reports and Track Field History,

3. Allow search >> Save

The screenshot shows the Salesforce Setup interface with the 'Object Manager' selected. A specific object, 'Billing details and feedback', is being edited. The 'Details' tab is active, showing various configuration options such as API Name ('Billing_details_and_feedback_c'), Singular Label ('Billing details and feedback'), and Plural Label ('Billing details and feedback'). On the left, a sidebar lists other setup categories like 'Fields & Relationships', 'Page Layouts', and 'Lightning Record Pages'. At the top right, there are 'Edit' and 'Delete' buttons.

Milestone 3 : Tabs

Creating a Custom Tab

To create a Tab:(Customer Details)

1. Go to setup page >> type Tabs in Quick Find bar >> click on tabs >> New (under custom object tab)
2. Select Object(Customer Details) >> Select the tab style >> Next (Add to profiles page) keep it as default >> Next (Add to Custom App) uncheck the include tab .
3. Make sure that the Append tab to users' existing personal customizations is checked.
4. Click save.

The screenshot shows the Salesforce Setup interface under the 'Tabs' section. It includes three main sections: 'Custom Object Tabs', 'Web Tabs', and 'Visualforce Tabs'. In the 'Custom Object Tabs' section, there is a table with columns for Action, Label, Tab Style, and Description. One entry for 'Customer Details' is shown with a 'Hexagon' tab style. The 'Web Tabs' and 'Visualforce Tabs' sections both indicate that no tabs have been defined.

Action	Label	Tab Style	Description
Edit Del	Appointments	Hexagon	
Edit Del	Billing details and feedback	Desk	
Edit Del	Customer Details	Chess piece	
Edit Del	Service records	Globe	

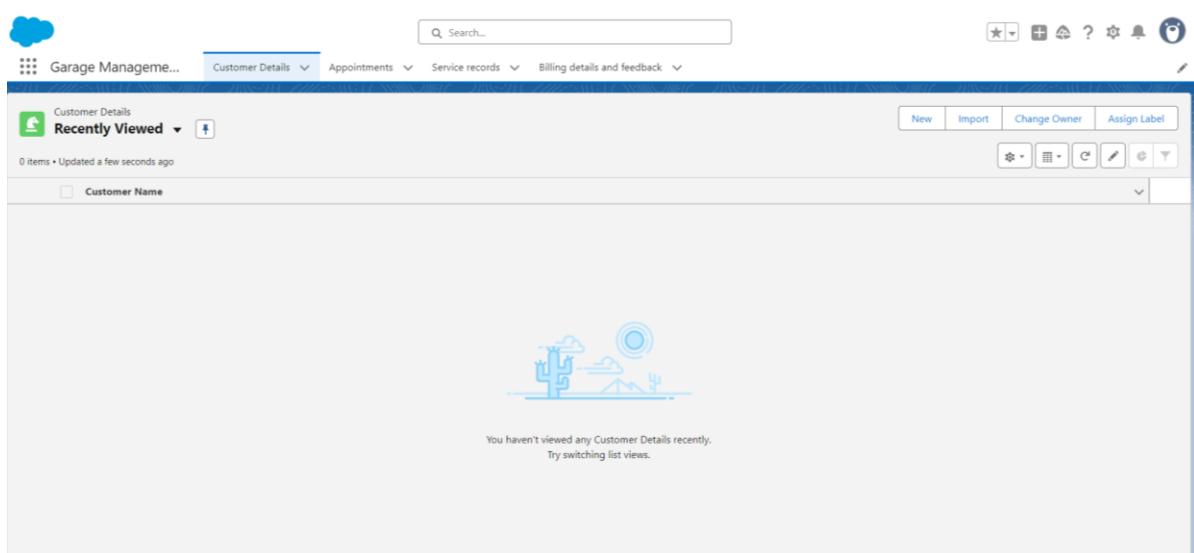
Creating Remaining Tabs

1. Now create the Tabs for the remaining Objects, they are “ Appointments, Service records,Billing details and feedback”.
2. Follow the same steps as mentioned in Activity -1.

Milestone 4 : The Lightning App

To create a lightning app page:

1. Go to setup page >> search “app manager” in quick find >> select “app manager” >> click on New lightning App.
2. Fill the app name in app details as Garage Management Application >> Next >> (App option page) keep it as default >> Next >> (Utility Items) keep it as default >> Next.
3. To Add Navigation Items:
4. Select the items (Customer Details,Appointments, Service records, Billing details and feedback, Reports and Dashboards) from the search bar and move it using the arrow button >> Next.
5. To Add User Profiles.



Milestone 5 : Fields

Creation of fields for the Customer Details object

To create fields in an object:

1. Go to setup >> click on Object Manager >> type object name(Customer Details) in search bar >> click on the object.
2. Now click on “Fields & Relationships” >> New
3. Select Data Type as a “Phone”
4. Click on next.

- Fill the Above as following:
 - Field Label: Phone number.
 - Field Name : gets auto generated.
 - Click on Next >> Next >> Save and new.

Note: Follow the above steps for the remaining field for the same object.

- To create another fields in an object:
 - Go to setup >> click on Object Manager >> type object name(Customer Details) in search bar >> click on the object.
 - Now click on “Fields & Relationships” >> New
 - Select Data type as a “Email” and Click on Next
 - Fill the Above as following:
 - Field Label : Gmail
 - Field Name : gets auto generated
 - Click on Next >> Next >> Save and new.

Customer Details Custom Field: Phone number

Custom Field Definition Detail

Field Information

Field Label	Phone number
Field Name	Phone_number
API Name	Phone_number_c
Description	
Help Text	
Data Owner	
Field Usage	
Data Sensitivity Level	
Compliance Categorization	
Created By	internship.project 03/03/2025, 4:11 pm
Modified By	internship.project 03/03/2025, 4:11 pm

General Options

Required	Yes
Default Value	

Validation Rules

Creation of Lookup Field on Appointment Object :

Appointment Field: Appointment Name

Custom Field Definition Detail

Field Information

Field Label	Appointment Name
Field Type	Auto Number
Description	
Data Owner	
Field Usage	
Data Sensitivity Level	
Compliance Categorization	
Display Format	app-(00)

General Options

Required	Yes
----------	-----

Creation of Checkbox Field on Appointment Object :

The screenshot shows the Salesforce Setup interface for creating a custom field. The left sidebar is titled 'Fields & Relationships' under 'Appointment'. The main area is titled 'Custom Field Definition Detail' for 'Appointment Custom Field Replacement Parts'. The 'Field Information' section shows the field label 'Replacement Parts', field name 'Replacement_Parts', and API name 'Replacement_Parts__c'. The 'Object Name' is 'Appointment' and the 'Data Type' is 'Checkbox'. The 'General Options' section shows 'Default Value' as 'Unchecked'. The 'Validation Rules' section indicates 'No validation rules defined'. The 'Help for this Page' link is visible at the top right.

Creation of Date Field on Appointment Object :

The screenshot shows the Salesforce Setup interface for creating a custom field. The left sidebar is titled 'Fields & Relationships' under 'Appointment'. The main area is titled 'Custom Field Definition Detail' for 'Appointment Custom Field Appointment Date'. The 'Field Information' section shows the field label 'Appointment Date', field name 'Appointment_Date', and API name 'Appointment_Date__c'. The 'Object Name' is 'Appointment' and the 'Data Type' is 'Date'. The 'General Options' section shows 'Required' checked and 'Default Value' as 'None'. The 'Validation Rules' section indicates 'No validation rules defined'. The 'Help for this Page' link is visible at the top right.

Creation of Currency Field on Appointment Object :

The screenshot shows the Salesforce Setup interface for creating a custom field. The left sidebar is titled 'Fields & Relationships' under 'Appointment'. The main area is titled 'Custom Field Definition Detail' for 'Appointment Custom Field Service Amount'. The 'Field Information' section shows the field label 'Service Amount', field name 'Service_Amount', and API name 'Service_Amount__c'. The 'Object Name' is 'Appointment' and the 'Data Type' is 'Currency'. The 'General Options' section shows 'Required' checked and 'Default Value' as 'None'. The 'Currency Options' section shows 'Length' as 18 and 'Decimal Places' as 0. The 'Validation Rules' section indicates 'No validation rules defined'. The 'Help for this Page' link is visible at the top right.

Creation of Text Fields :

The screenshot shows the Salesforce Setup interface for creating a custom field. The object selected is 'Appointment'. The field being created is named 'Vehicle number plate' with the API name 'Vehicle_number_plate_c'. The field type is set to 'Text'. Other details like description, help text, and data sensitivity level are also visible.

Creation of Picklist Fields :

The screenshot shows the Salesforce Setup interface for creating a custom picklist field named 'Service Status' on the 'Service records' object. The field type is set to 'Picklist'. It includes options for restricting the picklist to a specific value set and specifying a controlling field.

Creating Formula Field in Service records Object :

The screenshot shows the Salesforce Setup interface under the 'Object Manager'. A custom field named 'service date' has been created for the 'Service records' object. The field is defined as a formula type, with the formula being 'CreatedDate'. Other details include the field label 'service date', API name 'service_date_c', and object name 'Service records'. The field is active and was created by 'internship_project' on 03/03/2025 at 10:11 am.

Milestone 6 : Validation rule

To create a validation rule to an Appointment Object

1. Go to the setup page >> click on object manager >> From drop down click edit for Appointment object.
2. Click on the validation rule >> click New.
3. Enter the Rule name as “ Vehicle ”.
4. Insert the Error Condition Formula as :-
5. NOT(REGEX(Vehicle_number_plate__c , "[A-Z]{2}[0-9]{2}[A-Z]{2}[0-9]{4}"))
6. Enter the Error Message as “Please enter valid number”, select the Error location as Field and select the field as “Vehicle number plate”, and click Save.

The screenshot shows the Salesforce Setup interface under the 'Object Manager'. An appointment validation rule named 'Vehicle' has been created. The rule uses the formula 'NOT(REGEX(Vehicle_number_plate__c , "[A-Z]{2}[0-9]{2}[A-Z]{2}[0-9]{4}"))' to validate the 'Vehicle number plate' field. The error message is 'Please enter valid number'. The rule is active and was created by 'internship_project' on 04/03/2025 at 10:53 am.

To create a validation rule to an Billing details and feedback Object :

The screenshot shows the Salesforce Object Manager interface. The top navigation bar includes tabs for 'Inbox (1,288)', 'Your temporary password has...', 'Student - Skill Wallet', 'REGEX', and 'Object Manager | Salesforce'. Below the navigation is a search bar labeled 'Search Setup'.

The main content area is titled 'Billing details and feedback Validation Rule'. It displays a 'Validation Rule Detail' record with the following fields:

Validation Rule Detail	
Rule Name	rating_should_be_less_than_5
Error Condition Formula	NOT(REGEX(TEXT(Rating_for_Service__c), "[1-5]{1}"))
Error Message	rating should be from 1 to 5
Description	
Created By	internship.project, 04/03/2025, 11:41 am
Active	<input checked="" type="checkbox"/>
Error Location	Rating for Service
Modified By	internship.project, 04/03/2025, 11:41 am

At the bottom of the page, there is a footer bar with the text 'Billing details and feedback Validation Rule ~ Salesforce - Developer Edition'.

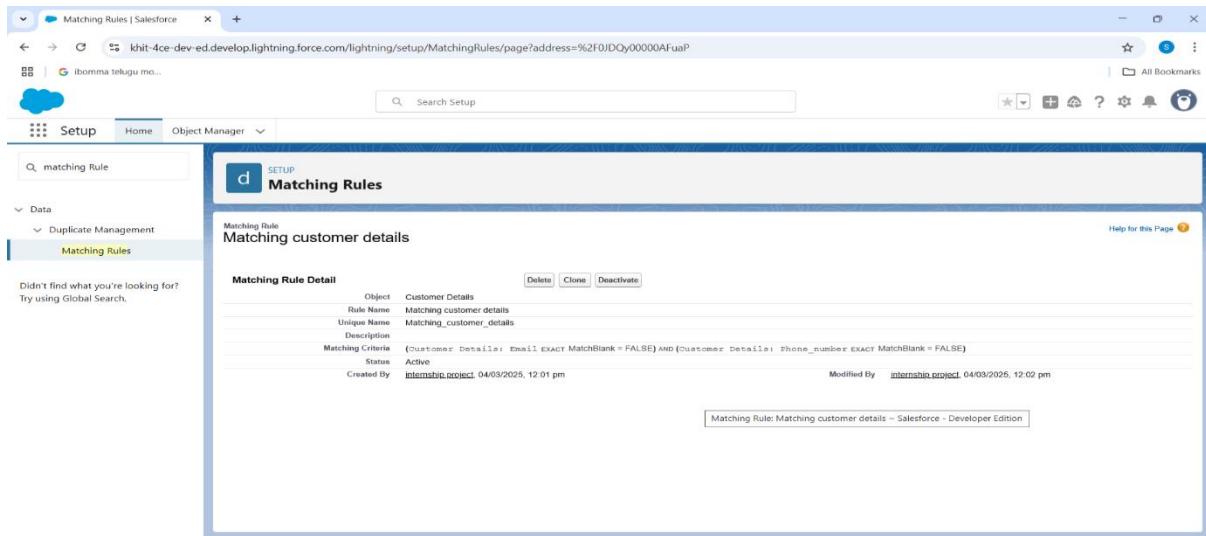
Milestone 7 : Duplicate rule

To create a matching rule to an Customer details Object

1. Go to quick find box in setup and search for matching Rule.
2. Click on matching rule >> click on New Rule.

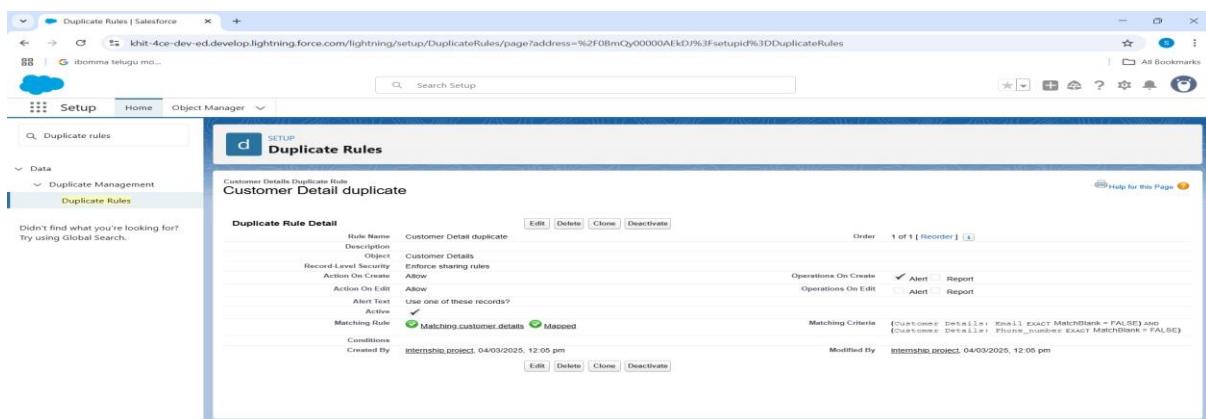
3. Select the object as Customer details and click Next.

4. Give the Rule name : Matching customer details
5. Unique name : is auto populated
6. Define the matching criteria as
 7. Field Matching Method
 - a. Gmail Exact
 - b. Phone Number Exact
8. Click save.
9. After Saving Click on Activate



To create a Duplicate rule to an Customer details Object :

1. Go to quick find box in setup and search for Duplicate rules.
2. Click on Duplicate rule >> click on New Rule >> select customer details object.
3. Give the Rule name as : Customer Detail duplicate
4. Scroll a little in Matching rule section
5. Select the matching rule : Matching customer details
6. And Click on save.
7. After saving the Duplicate Rule, Click on Activate.



Milestone 8 : Profiles Manager Profile

To create a new profile:

1. Go to setup >> type profiles in quick find box >> click on profiles >> clone the desired profile (Standard User) >> enter profile name (Manager) >> Save.
2. While still on the profile page, then click Edit.
3. Select the Custom App settings as default for the Garage management.

4. Scroll down to Custom Object Permissions and Give access permissions for Appointments,Billing details and feedback , service records and customer details objects as mentioned in the below diagram.
5. Changing the session times out after should be “ 8 hours of inactivity”.
6. Change the password policies as mentioned :
7. User passwords expire in should be “ never expires ”.
8. Minimum password length should be “ 8 ”, and click save.

	Custom Object Permissions						Session Settings						Password Policies											
	Basic Access		Edit		Delete		View All Records		Modify All Records		View All Fields		Basic Access		Edit		Delete		View All Records		Modify All Records		View All Fields	
	Read	Create	Read	Customer Details	Read	Create																		
Appointments	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Service records	<input checked="" type="checkbox"/>	<input type="checkbox"/>																		
Billing details and feedback	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>																					

Session Settings: Session Times Out After: 8 hours of inactivity

Password Policies: User passwords expire in: Never expires
Enforce password history: 3 passwords remembered
Minimum password length: 8
password complexity requirement: Must include alpha and numeric characters

Sales person Profile

1. Go to setup >> type profiles in quick find box >> click on profiles >> clone the desired profile (Salesforce Platform User) >> enter profile name (sales person) >> Save.
2. While still on the profile page, then click Edit.
3. Select the Custom App settings as default for the GArage management.
4. Scroll down to Custom Object Permissions and Give access permissions for Appointments,Billing details and feedback , service records and customer details objects as mentioned in the below diagram.
5. And click save

Profiles

Custom Field-Level Security

Appointment	[View]	Customer Details	[View]
Billing details and feedback	[View]	Service records	[View]

Custom App Settings

	Visible	Default		Visible	Default
Analytics Studio (standard_Insights)	<input type="checkbox"/>	<input type="checkbox"/>	Platform (standard_Platform)	<input checked="" type="checkbox"/>	<input type="checkbox"/>
App Launcher (standard_AppLauncher)	<input type="checkbox"/>	<input type="checkbox"/>	WDC (standard_Work)	<input type="checkbox"/>	<input type="checkbox"/>
Garage Management Application (Garage_Management_Application)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			

Service Provider Access

Tab Settings

Standard Tab Settings	Home	Default On	Inventory Operations	Tab Hidden
Accounts			Inventory Reservations	Tab Hidden
All Sites	Tab Hidden		Invoices	Tab Hidden
Alternative Payment Methods	Tab Hidden		IT Services	Tab Hidden
			Confirmed Items	Tab Hidden

Milestone 9 : Role & Role Hierarchy

Creating Manager Role:

1. Go to quick find >> Search for Roles >> click on set up roles.

Understanding Roles

Sample Role Hierarchy

Executive Staff

- CEO President
- CFO VP. Sales
- Marketing Director

Western Sales Director

- Western Sales Rep
- Central Sales Rep
- East Sales Rep

Eastern Sales Director

- North Sales Rep
- South Sales Rep

International Sales Director

- Europe Sales Rep
- Asia Sales Rep

Notes for Executive Staff:

- * View & edit data, not up to date with latest assignments for all users below
- Can view & edit data of other Executive Staff

Notes for Western Sales Director:

- * View & edit data, not up to date with latest assignments for all users directly below
- Can't edit assignments for all users above at same level

Notes for International Sales Director:

- * View & edit data, not up to date with latest assignments only for own data at same level or at same level

2. Click on Expand All and click on add role under whom this role works.

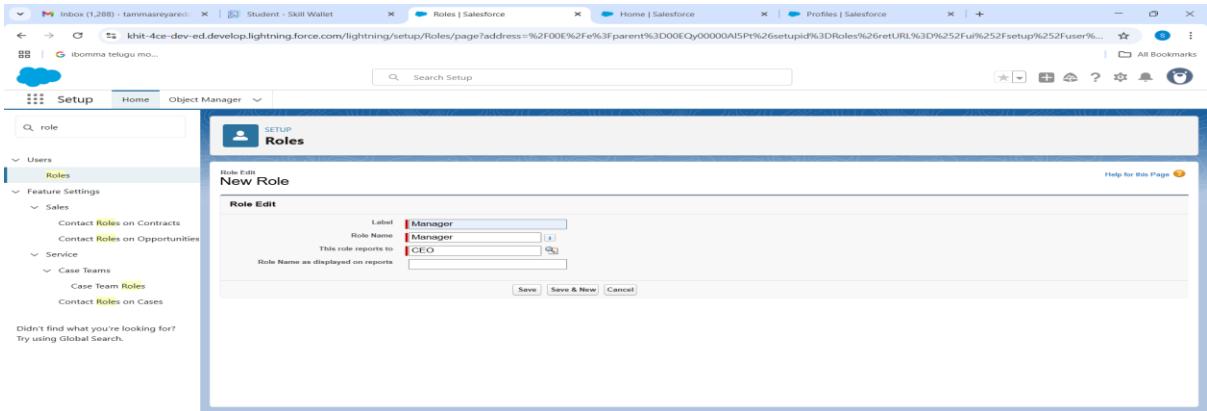
Creating the Role Hierarchy

Your Organization's Role Hierarchy

- KHR
 - CEO
 - CFO
 - COO
 - Manager
 - SVP, Customer Service & Support
 - SVP, Human Resources
 - SVP, Sales & Marketing

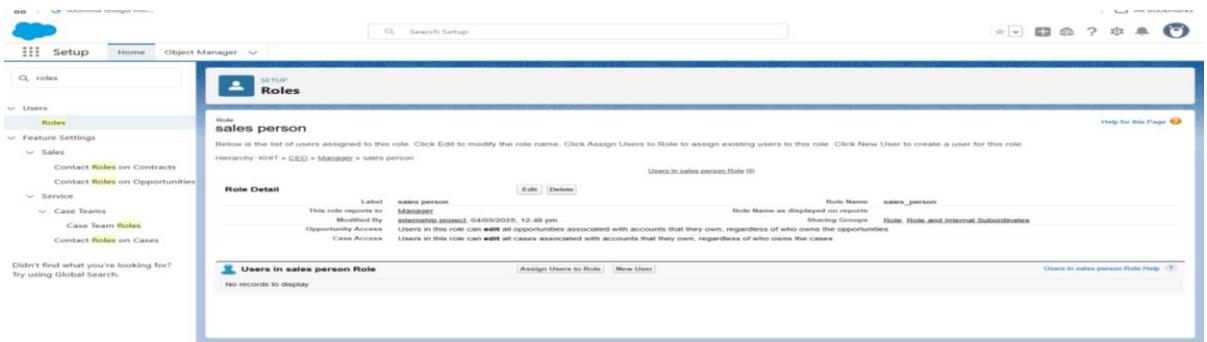
Buttons: Edit | Del | Assign

- Give Label as “Manager” and Role name gets auto populated. Then click on Save.



Creating another roles

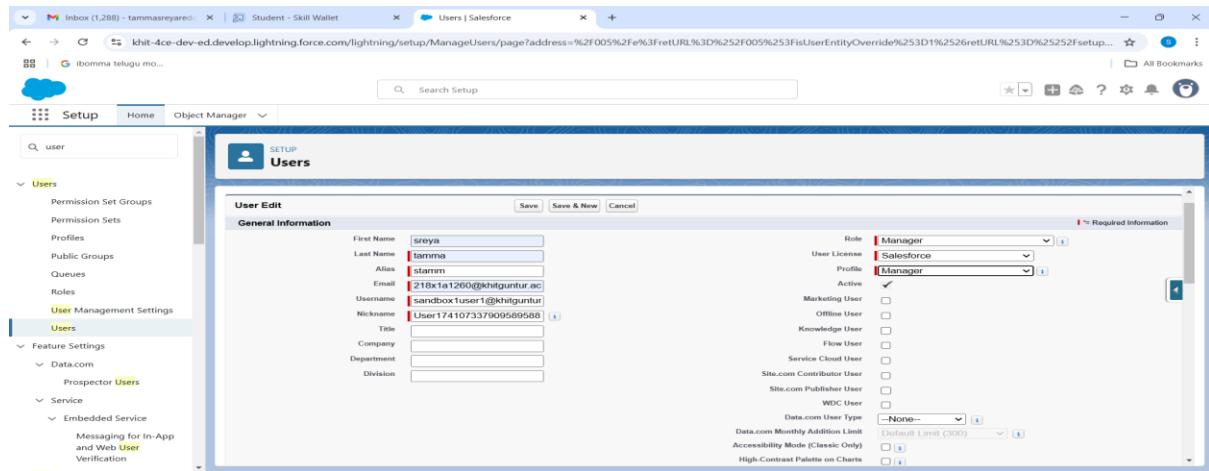
- Creating another two roles under manager
- Go to quick find >> Search for Roles >> click on set up roles.
- Click plus on CEO role, and click add role under manager.
- Give Label as “sales person” and Role name gets auto populated. Then click on Save.



Milestone 10 : Users

Create User

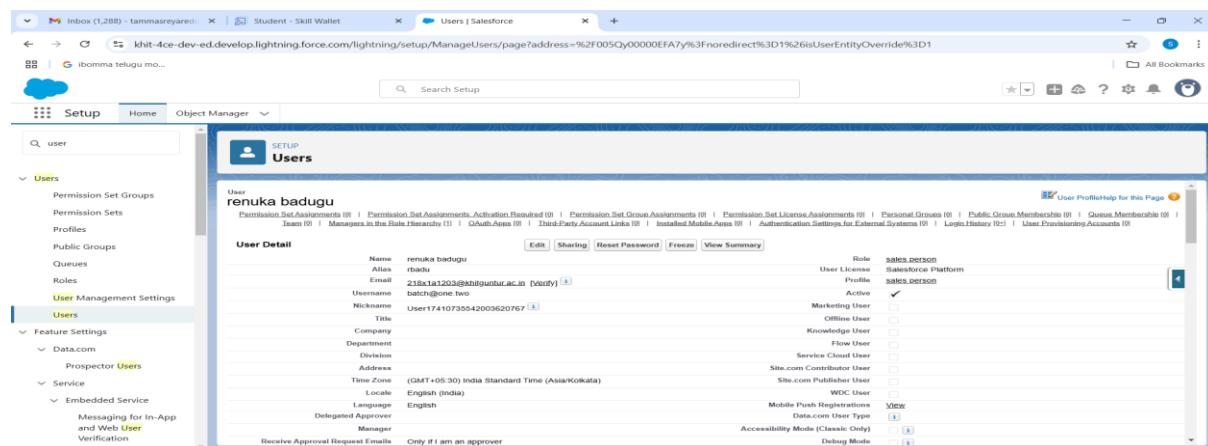
- Go to setup >> type users in quick find box >> select users >> click New user.
- Fill in the fields
 - First Name : Niklaus
 - Last Name : Mikaelson
 - Alias : Give a Alias Name
 - Email id : Give your Personal Email id
 - Username : Username should be in this form: text@text.text
 - Nick Name : Give a Nickname
 - Role : Manager
 - User licence : Salesforce
 - Profiles : Manager
- Save



creating another users

1. Repeat the steps and create another user using
 - a) Role : sales person
 - b) User licence : Salesforce Platform
 - c) Profile : sales person

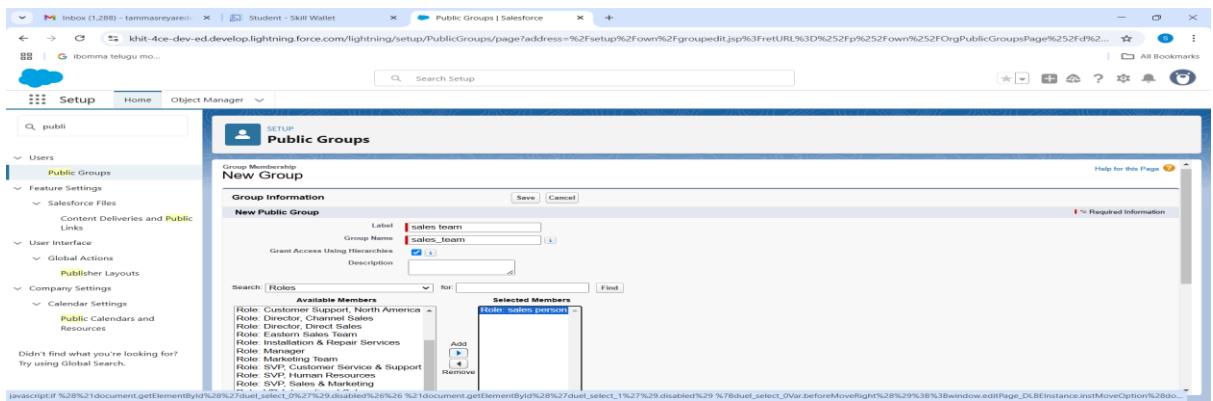
Note : create atleast 3 users with these permissions.



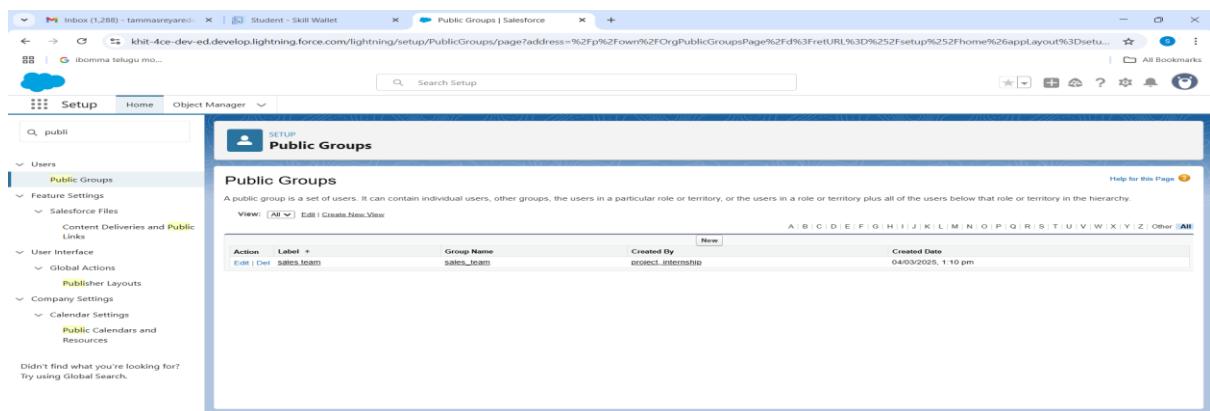
Milestone 11 : Public groups

Creating New Public Group

1. Go to setup >> type users in quick find box >> select public groups >> click New.
2. Give the Label as “sales team”.
3. Group name is autopopulated.



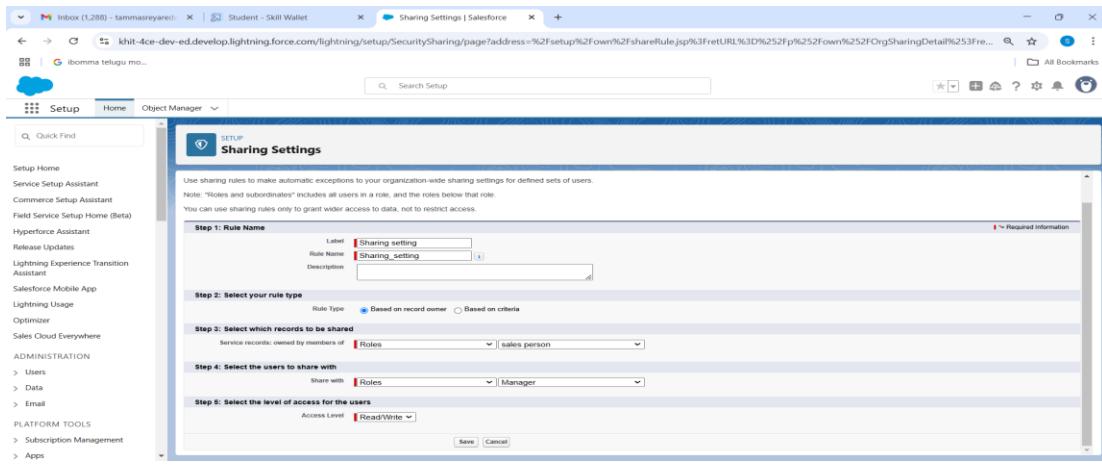
4. Search for Roles.
5. In Available Members select Sales person and click on add it will be moved to selected member.
6. Click on save.



Milestone 12 : Sharing Setting

Creating Sharing settings

1. Go to setup >> type users in quick find box >> select Sharing Settings >> click Edit.
2. Change the OWD setting of the Service records Object to private as shown in fig.
3. Click on save and refresh.
4. Scroll down a bit, Click new on Service records sharing Rules.
5. Give the Label name as “ Sharing setting”
6. Rule name is auto populated.
7. In step 3 : Select which records to be shared, members of “ Roles ” >> “ Sales person”
8. In step 4: share with, select “ Roles ” >> “ Manager ”
9. In step 5 : Change the access level to “ Read / write ”.
10. Click on save.

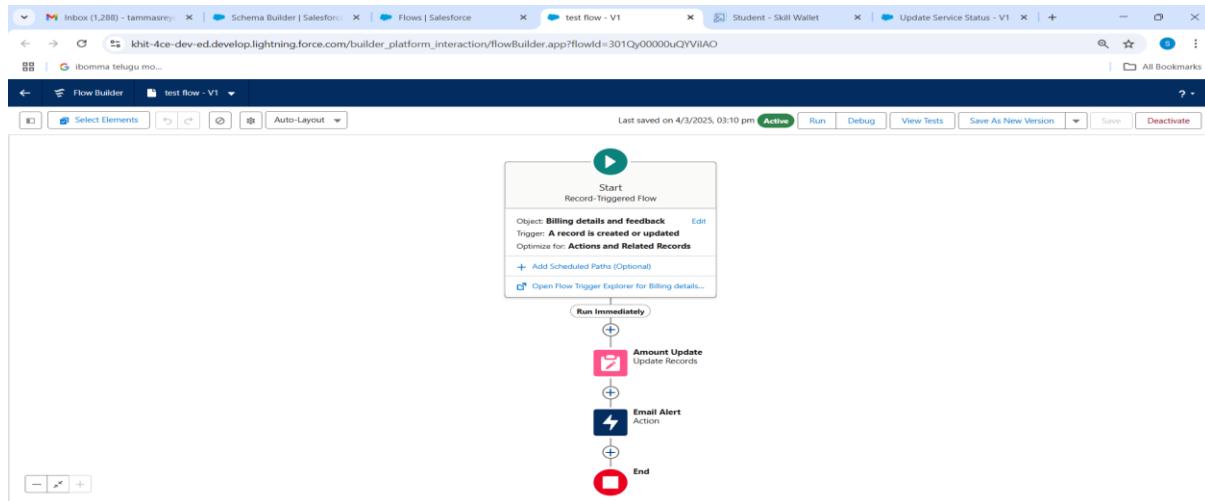


Milestone 13 : Flows

Create a Flow

1. Go to setup >> type Flow in quick find box >> Click on the Flow and Select the New Flow.
2. Select the Record-triggered flow and Click on Create.
3. Select the Object as “Billing details and feedback” in the Drop down list.
4. Select the Trigger Flow when: “A record is Created or Updated”.
5. Select the Optimize the flow for: “Actions and Related Records” and Click on Done.
6. Under the Record-triggered Flow Click on “+” Symbol and In the Drop down List select the “Update records Element”.
7. Give the Label Name : Amount Update
8. Api name : is auto populated
9. Give the Label Name : Amount Update
10. Api name : is auto populated
11. Click on the New Resource, And select Variable.
12. Select the resource type as text template.
13. Enter the API name as “ alert”.
14. Change the view as Rich Text ? View to Plain Text.
15. In body field paste the syntax that given below.
16. Dear {!\$Record.Service_records__r.Appointment__r.Customer_Name__r.Name},
17. Before creating another Element. Create a New Resource form Toolbox form top left.
18. Click on the New Resource, And select Variable
19. Enter the API name as “ alert”
20. Select the resource type as text template
21. Amount paid : {!\$Record.Payment_Paid__c}
22. Thank you for Coming .
23. Click done.
24. Now Click on Add Element , select Action.
25. Their action bar will be opened in that search for “ send email ” and click on it.
26. Give the label name as “ Email Alert”
27. API name will be auto populated.
28. Enable the body in set input values for the selected action.

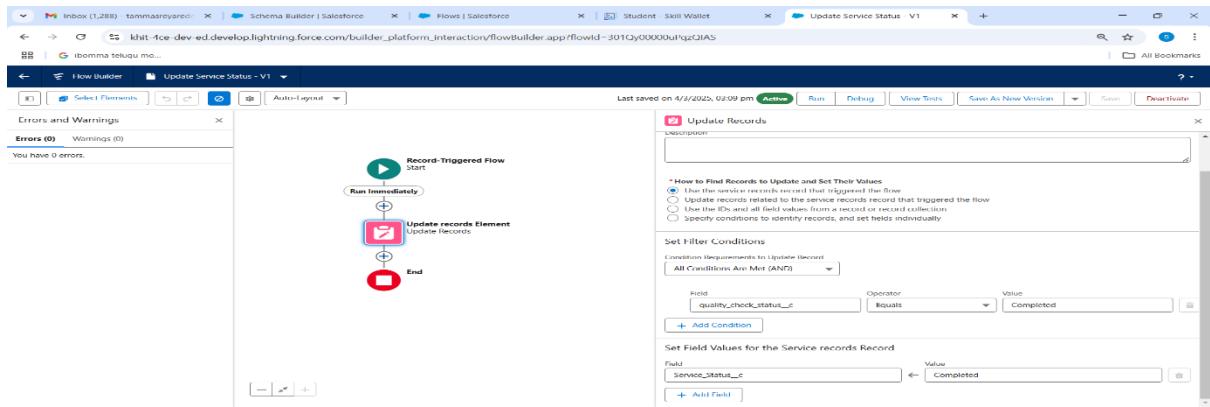
29. Select the text template that created , Body : { !alert }
30. Include recipient address list select the email form the record.
31. RecipientAddressList:
`{ !$Record.Service_records__r.Appointment__r.Customer_Name__r.Gmail__c }`
32. Include subject as “ Thank You for Your Payment - Garage Management”.
33. Click done.
34. Click on save. Give the Flow label , Flow Api name will be autopopulated.
35. And click save, and click on activate.



Create another Flow

1. Go to setup ? type Flow in quick find box ? Click on the Flow and Select the New Flow.
2. Select the Record-triggered flow and Click on Create.
3. Select the Object as “ Service records”in the Drop down list.
4. Select the Trigger Flow when: “A record is Created or Updated”.
5. Select the Optimise the flow for: “Actions and Related Records” and Click on Done.
6. Under the Record-triggered Flow Click on “+” Symbol and In the Drop down List select the “Update records Element”.
7. Set a filter condition : All Conditions are met(AND)
8. Field : Quality_Check_Status__c
9. Operator : Equals
10. Value : True
11. And Set Field Values for the Billing details and feedback Record
12. Field : Service_Status__c
13. Value : Completed
14. Click On Done.
15. Click on save

16. Given the Flow label as Update Service Status , Flow Api name will be auto populated.
17. And click save, and click on activate.



Milestone 14 : Apex Trigger Apex handler

UseCase : This use case works for Amount Distribution for each Service the customer selected for there Vehicle.

1. Login to the respective trailhead account and navigate to the gear icon in the top right corner.
2. Click on the Developer console. Now you will see a new console window.
3. In the toolbar, you can see FILE. Click on it and navigate to new and create New apex class.
4. Name the class as “AmountDistributionHandler ”.

```

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

```
Developer Console - Google Chrome
khit-4ce-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage
File Edit Debug Test Workspace Help < >
AmountDistributionHandler.apex
Code Coverage: None API Version: 63 Go To
14 }
15
16     else if(app.Maintenance_service__c == true && app.Repairs__c == true){
17
18         app.Service_Amount__c = 5000;
19
20     }
21
22     else if(app.Maintenance_service__c == true && app.Replacement_Parts__c == true){
23
24         app.Service_Amount__c = 8000;
25
26     }
27
28     else if(app.Repairs__c == true && app.Replacement_Parts__c == true){
29
30         app.Service_Amount__c = 7000;
31
32     }
33
34 }
```

```
Developer Console - Google Chrome
khit-4ce-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage
File Edit Debug Test Workspace Help < >
AmountDistributionHandler.apex
Code Coverage: None API Version: 63 Go To
31 }
32
33
34     else if(app.Maintenance_service__c == true){
35
36         app.Service_Amount__c = 2000;
37
38     }
39
40     else if(app.Repairs__c == true){
41
42         app.Service_Amount__c = 3000;
43
44     }
45
46     else if(app.Replacement_Parts__c == true){
47
48         app.Service_Amount__c = 5000;
49
50     }
51
52 }
```

```
Developer Console - Google Chrome
khit-4ce-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage
File Edit Debug Test Workspace Help < >
AmountDistributionHandler.apex
Code Coverage: None API Version: 63 Go To
38 }
39
40     else if(app.Repairs__c == true){
41
42         app.Service_Amount__c = 3000;
43
44     }
45
46     else if(app.Replacement_Parts__c == true){
47
48         app.Service_Amount__c = 5000;
49
50     }
51
52 }
53
54 }
55
56 }
57 }
```

Trigger Handler :

How to create a new trigger :

1. While still in the trailhead account, navigate to the gear icon in the top right corner.
2. Click on developer console and you will be navigated to a new console window.
3. Click on File menu in the tool bar, and click on new? Trigger.

4. Enter the trigger name and the object to be triggered.
5. Name : AmountDistribution
6. sObject : Appointment__c

Syntax For creating trigger :

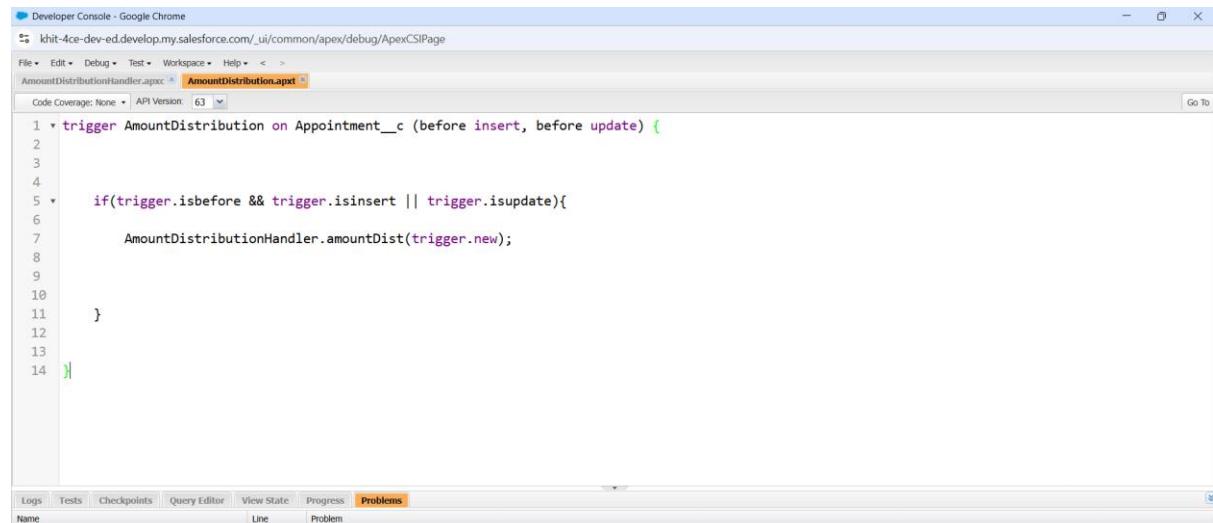
The syntax for creating trigger is :

Trigger [trigger name] on [object name](Before/After event)

```
{  
}
```

In this project , trigger is called whenever the particular records sum exceed the threshold i.e minimum business requirement value. Then the code in the trigger will get executed.

1. Handler for the Appointment Object



The screenshot shows the Salesforce Developer Console in Google Chrome. The URL is khit-4ce-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage. The tab bar shows 'AmountDistributionHandler.apxc' and 'AmountDistribution.apxc'. The code editor contains the following Apex code:

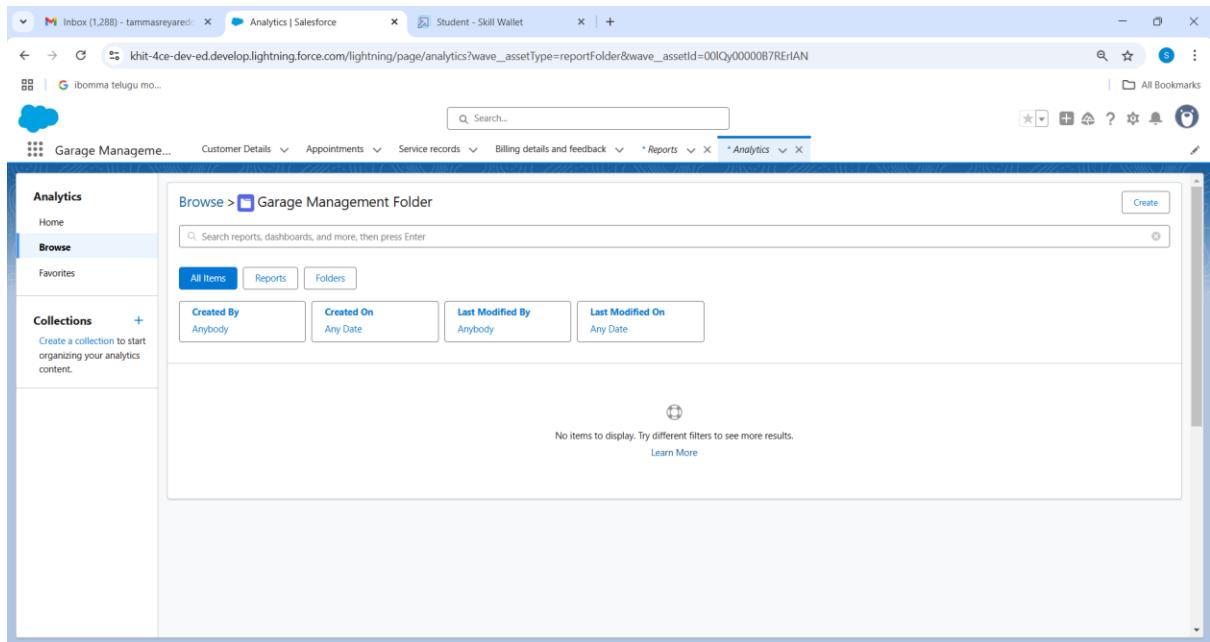
```
1 trigger AmountDistribution on Appointment__c (before insert, before update) {
2
3
4
5 if(trigger.isbefore && trigger.isinsert || trigger.isupdate){
6     AmountDistributionHandler.amountDist(trigger.new);
7
8
9
10 }
11
12
13
14 }
```

The code is written in Apex, defining a trigger named 'AmountDistribution' on the 'Appointment__c' object. It handles both 'before insert' and 'before update' events. Inside the trigger, it calls a static method 'amountDist' from the 'AmountDistributionHandler' class on the new records.

Milestone 15 : Reports

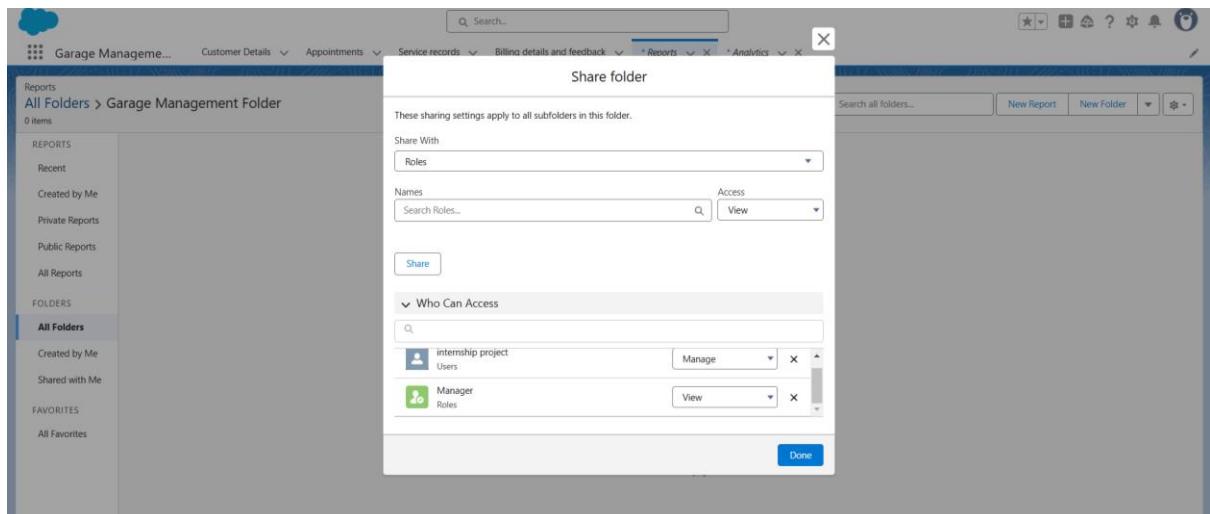
create a report folder :

2. Click on the app launcher and search for reports.
3. Click on the report tab, click on new folder.
4. Give the Folder label as “Garage Management Folder”, Folder unique name will be auto populated.
5. Click save.



Sharing a report folder :

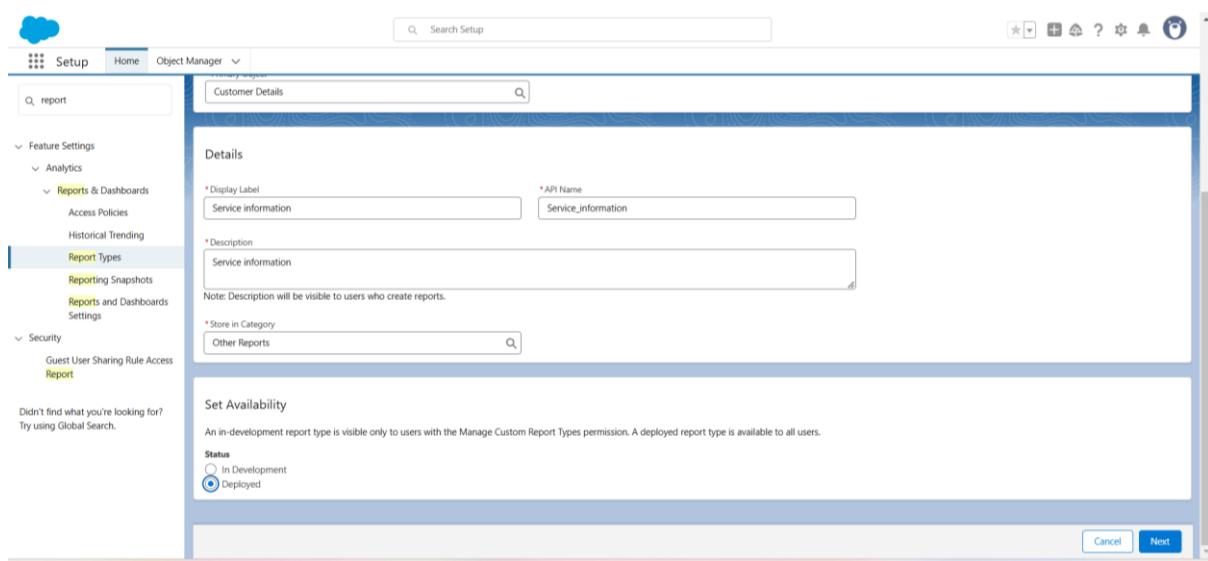
1. Go to the app >> click on the reports tab.
2. Click on the All folder , click on the Drop down arrow for Garage Management folder, and Click on share.
3. Select the share with as “roles”, in name field search for “manager”, give “view” as access for that role.
4. Then click share, and click on Done.



Create Report Type

1. Go to setup >> type users in quick find box >> select Report Type >> click on Continue.

2. Click on new custom report type.
3. Select the Primary object as “ Customer details” .
4. Give the Report type Label as “ Service information ”
5. Report type Name is autopopulated.
6. Keep the Description as same.
7. Select Store in Category as “ other Reports ”
8. Select the deployment status as “ Deployed ”, click on Next.
9. now , Click on Related object box.
10. Click on Select Object, choose Appointment Object as shown in fig.
11. Again Click to relate another object.
12. And select the related object as “ service records”.
13. Repeat the process and select the related object as “ Billing details and feedback”.
14. And click on save.



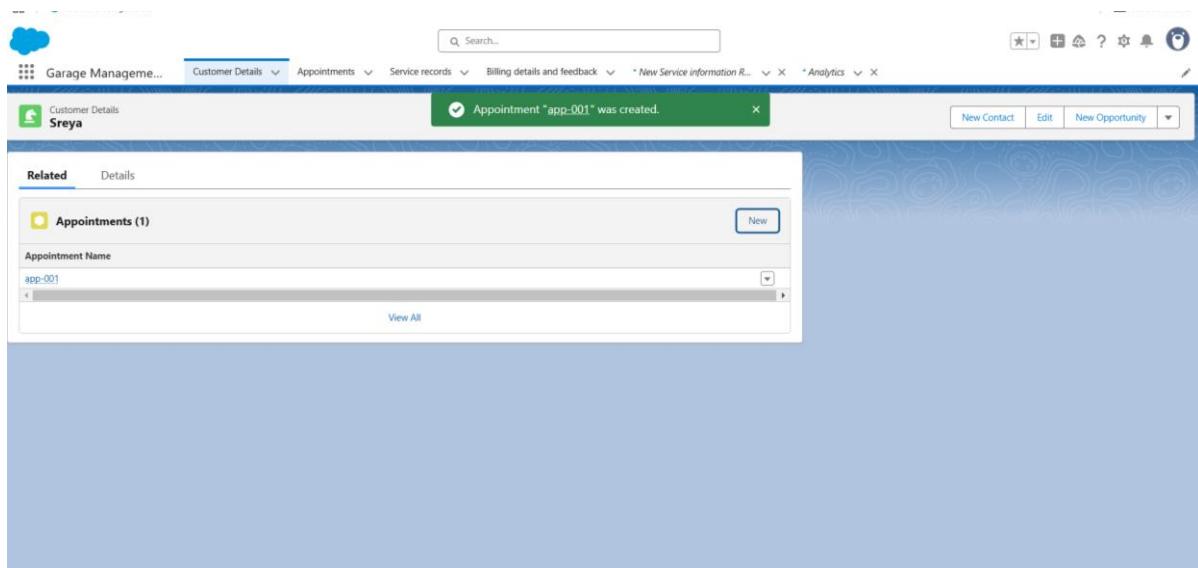
Create Report

Note : Before creating report, create latest “10” records in every object.

Try to fill every field in each record for better experience.

1. Go to the app >> click on the reports tab
2. Click New Report.
3. Select the Category as other reports, search for Service Information, select that report, click on it. And click on start report.
4. Their outline pane is opened already, select the fields that mentioned below in column section.
 - a. Customer name
 - b. Appointment Date

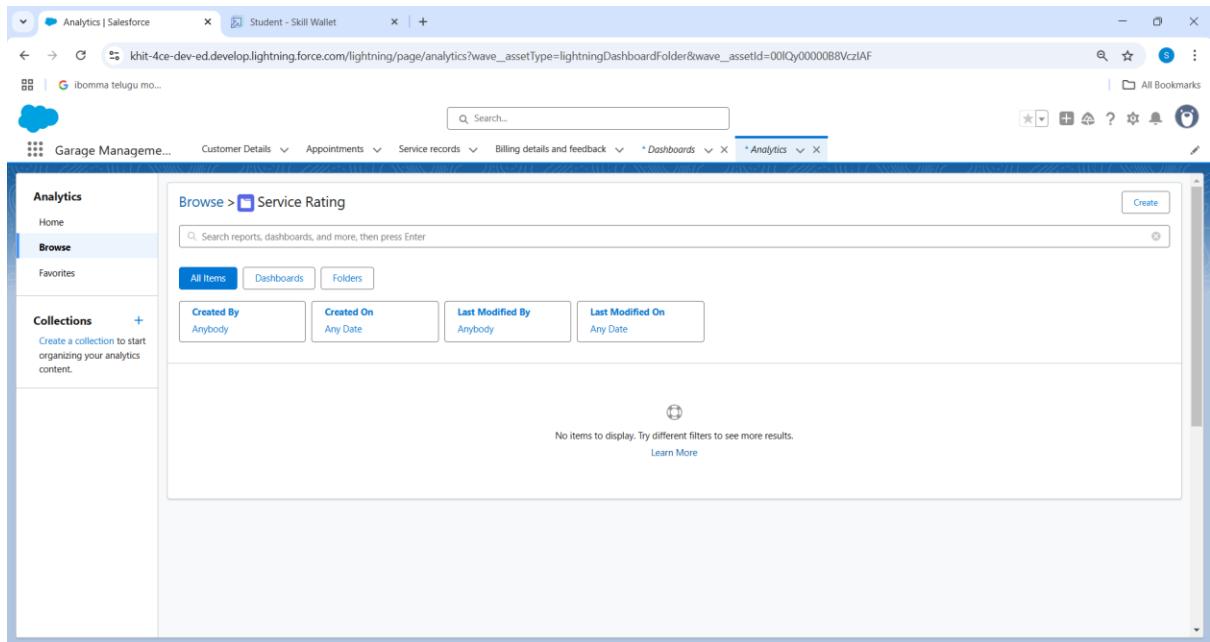
- c. Service Status
- d. Payment paid
- 5. Remove the unnecessary fields.
- 6. Select the fields that mentioned below in GROUP ROWS section.
 - a. Rating for Service
- 7. Select the fields that mentioned below in GROUP ROWS section.
 - a. Payment Status
- 8. Click on Add Chart , Select the Line Chart.
- 9. Click on save, Give the report Name : New Service information Report
- 10. Report unique Name is auto populated.
- 11. Select the folder the created and Click on save.



Milestone 16 : Dashboard

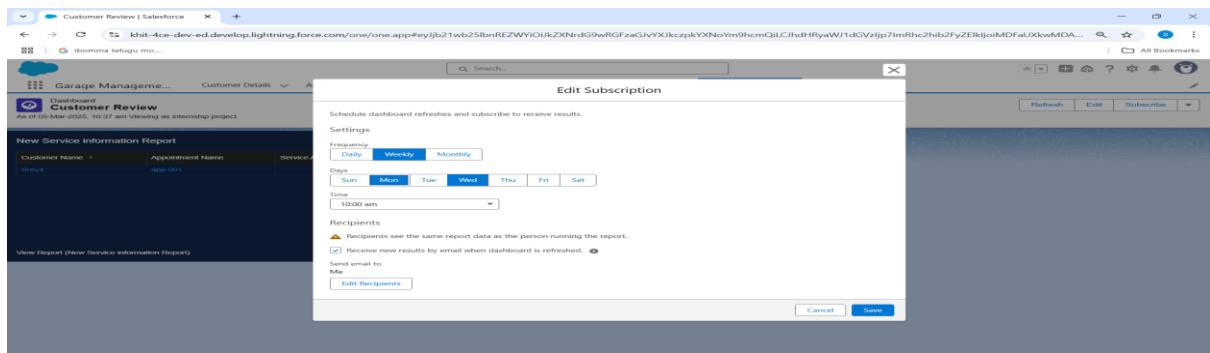
Create Dashboard Folder :

1. Click on the app launcher and search for dashboard.
2. Click on dashboard tab.
3. Click new folder, give the folder label as “ Service Rating dashboard”.
4. Folder unique name will be auto populated.
5. Click save.
6. Follow the same steps, from Reports Milestone and Activity 2, and provide the sharing settings for the folder that was just created.



Create Dashboard

1. Go to the app >> click on the Dashboards tabs.
2. Give a Name and select the folder that created, and click on create.
3. Select add component.
4. Select a Report and click on select.
5. Select the Line Chart. Change the theme.
6. Click Add then click on Save and then click on Done.
7. Preview is shown below.
8. After that Click on Subscribe on top right.
9. Set the Frequency as “ weekly ”.
10. Set a day as monday.
11. And Click on save.

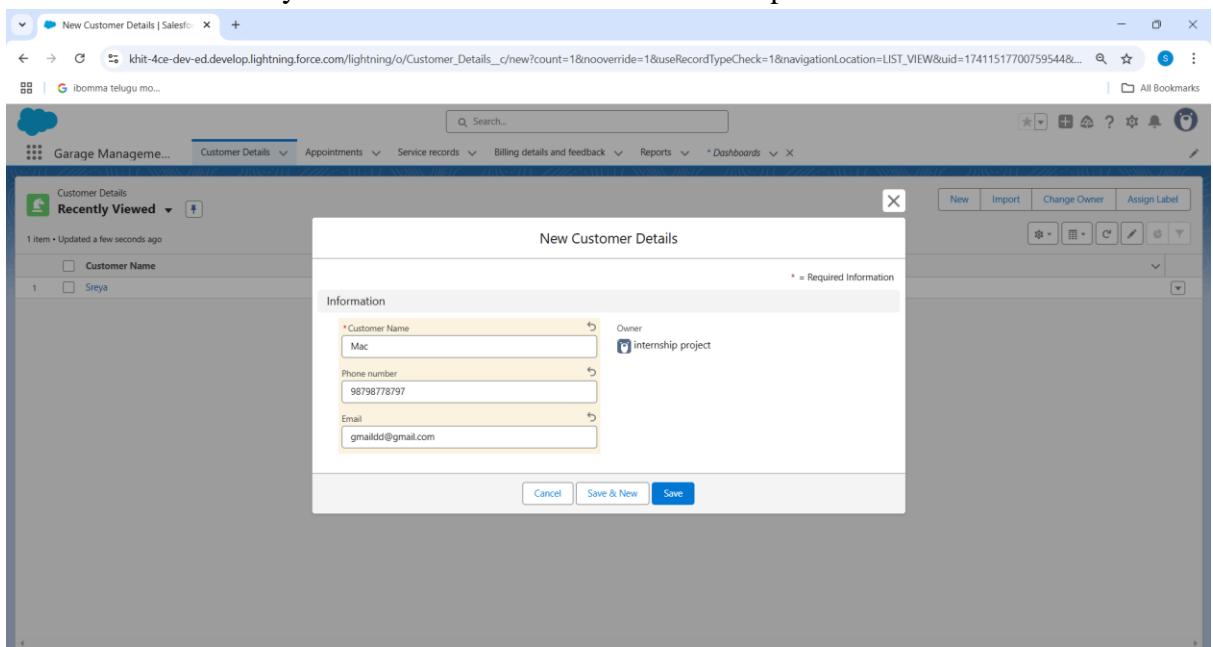


Milestone 17 : User Adoption

Creation Record:

1. To create a record in the follow objects follow these steps
2. Click on the app launcher located at the left side of the screen.

3. Search for “ **Garage Management** ” and click on it.
4. Click on the “ **Consumer details tab** ”.
5. Click on new and fill the details as shown below figs, and click save.
6. Now, Create the Appointment Record
7. Click on the “ **Appointment tab** ”.
8. Enter the customer details as created, while entering Appointment Date enter the date less than the created date.
9. Match the validation while entering the vehicle number plate.
10. Select the services you need.
11. Click on save to see the Service Amount.
12. Now, Create a service Record
13. Click on the “ **Service record tab** ”.
14. Enter the Appointment, and started is selected as default.
15. Click on save.
16. Open the record and click on Quality check status as true.
17. Click on save.
18. Now automatically Service status will be moved to completed



New Appointment | Salesforce

khit-4ce-dev-ed.lightning.force.com/lightning/o/Appointment__c/new?count=2&nooverride=1&useRecordTypeCheck=1&navigationLocation=LIST_VIEW&uid=174115186888342663&ba...

ibomma telugu mo...

Garage Management

Appointments Recently Viewed

1 item • Updated a few seconds ago

Appointment Name: app-001

Customer Details: Mac

Maintenance service: checked

Appointment Date: 07/03/2025

Service Amount: ₹70,000

*Vehicle number plate: T501TG5355

Repairs: checked

Replacement Parts: unchecked

Owner: internship project

Information

Save & New Save

All Bookmarks

A screenshot of the Salesforce Lightning interface. The top navigation bar shows the URL 'khit-4ce-dev-ed.lightning.force.com/lightning/o/Appointment__c/new?count=2&nooverride=1&useRecordTypeCheck=1&navigationLocation=LIST_VIEW&uid=174115186888342663&ba...'. Below the header, there's a sidebar titled 'Garage Management' with sections for 'Appointments' and 'Recently Viewed'. A message indicates '1 item • Updated a few seconds ago' with an entry for 'Appointment Name: app-001'. The main content area is titled 'Information' and contains fields for 'Customer Details' (set to 'Mac'), 'Maintenance service' (checkbox checked), 'Appointment Date' ('07/03/2025'), 'Service Amount' ('₹70,000'), 'Vehicle number plate' ('T501TG5355'), 'Repairs' (checkbox checked), and 'Replacement Parts' (checkbox unchecked). The 'Owner' field is populated with 'internship project'. At the bottom are buttons for 'Save & New' and 'Save'.

ser-001 | Service records | Sales...

Service records | Salesforce

Service records: ser-001 ~ Sales...

khit-4ce-dev-ed.lightning.force.com/lightning/r/Service_records__c/a02Qy00000BSOsllAX/view

ibomma telugu mo...

Garage Management

Customer Details Appointments Service records Billing details and feedback Reports Dashboards

Service records ser-001

New Contact Edit New Opportunity

Related Details

Service records Name: ser-001

Service Status: Completed

Service date: 05/03/2025

Appointment: app-004

Created By: internship project, 05/03/2025, 10:48 am

Last Modified By: internship project, 05/03/2025, 10:48 am

A screenshot of the Salesforce Lightning interface. The top navigation bar shows the URL 'khit-4ce-dev-ed.lightning.force.com/lightning/r/Service_records__c/a02Qy00000BSOsllAX/view'. Below the header, there's a sidebar titled 'Garage Management' with sections for 'Customer Details', 'Appointments', 'Service records', 'Billing details and feedback', 'Reports', and 'Dashboards'. The main content area shows a service record for 'ser-001'. It includes tabs for 'Related' and 'Details'. Under 'Details', it shows 'Service records Name: ser-001', 'Service Status: Completed', 'Service date: 05/03/2025', 'Appointment: app-004', and 'Created By: internship project, 05/03/2025, 10:48 am'. The 'Last Modified By' field also shows 'internship project, 05/03/2025, 10:48 am'. The 'Owner' field is listed as 'internship project'.

Conclusion : The Garage Management System (GMS) has proven to be an indispensable asset for automotive repair facilities, revolutionizing the way they operate and interact with customers. By providing a comprehensive suite of tools that streamline everything from customer and vehicle management to service tracking, inventory control, financial oversight, and staff coordination, GMS enables garages to optimize their workflows, reduce operational costs, and enhance service quality. The system's intuitive interface ensures that both staff and customers can navigate it with ease, fostering a seamless experience that builds trust and loyalty. Additionally, GMS's robust data analytics and reporting capabilities offer valuable insights, empowering garage owners to make informed decisions that drive growth and success in a highly competitive market.

The implementation of the Garage Management System is a strategic move towards a more efficient and customer-centric approach to automotive repair. It equips garages with the tools they need to deliver exceptional service, manage operations smoothly, and build strong, lasting relationships with their customers. In an industry that is constantly evolving, GMS is not just a beneficial addition but a necessary step for businesses aiming to stay ahead and thrive in the modern market landscape.