

**PROJECT TITLE:**

Garage Management System

**College Name:**

Sri Vasavi College Self Finance Wing, Erode

**College Code:**

bru17

**Team ID:** NM2025TMID23350

**TEAM MEMBERS:**

**Team Leader Name:** DINESH R

**Email:** dineshit144@gmail.com

**Team Member:** KAVIN M

**Email:** kavinms2006@gmail.com

**Team Member:** BOSE R

**Email:** karateraju56@gmail.com

**Team Member:** SOWMIYA P

**Email:** sowmiyap50@gmail.com

**Team Member:** NIVASKUMAR M

**Email:** nivash28122005@gmail.com

# **Garage Management System Project Documentation**

## **1. Project Overview**

The **Garage Management System (GMS)** is a Salesforce application designed to simplify and automate garage operations. It helps manage vehicles, customers, services, billing, and reporting on a single platform.

This system improves:

- Vehicle service tracking
- Customer management
- Billing and payments
- Workflow automation
- Report generation and analytics

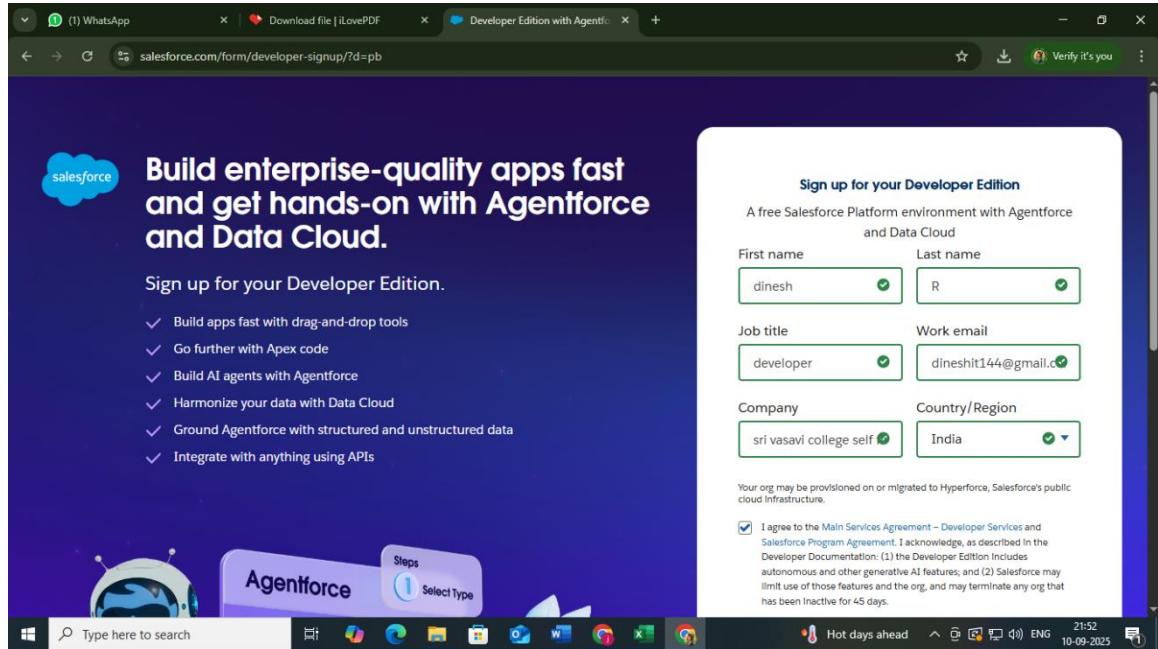
## **2. Objectives**

The main objectives of this project are:

1. To efficiently manage vehicle services, customers, and billing in one system.
2. To automate routine garage operations for improved productivity.
3. To ensure accurate tracking of services, payments, and customer records.
4. To provide real-time reports and dashboards for better decision-making.
5. To enhance customer satisfaction by delivering faster and organized services.

### 3. Creating a developer org in salesforce.

1. Go to <https://developer.salesforce.com/signup>
2. On the sign up form, enter the following details :



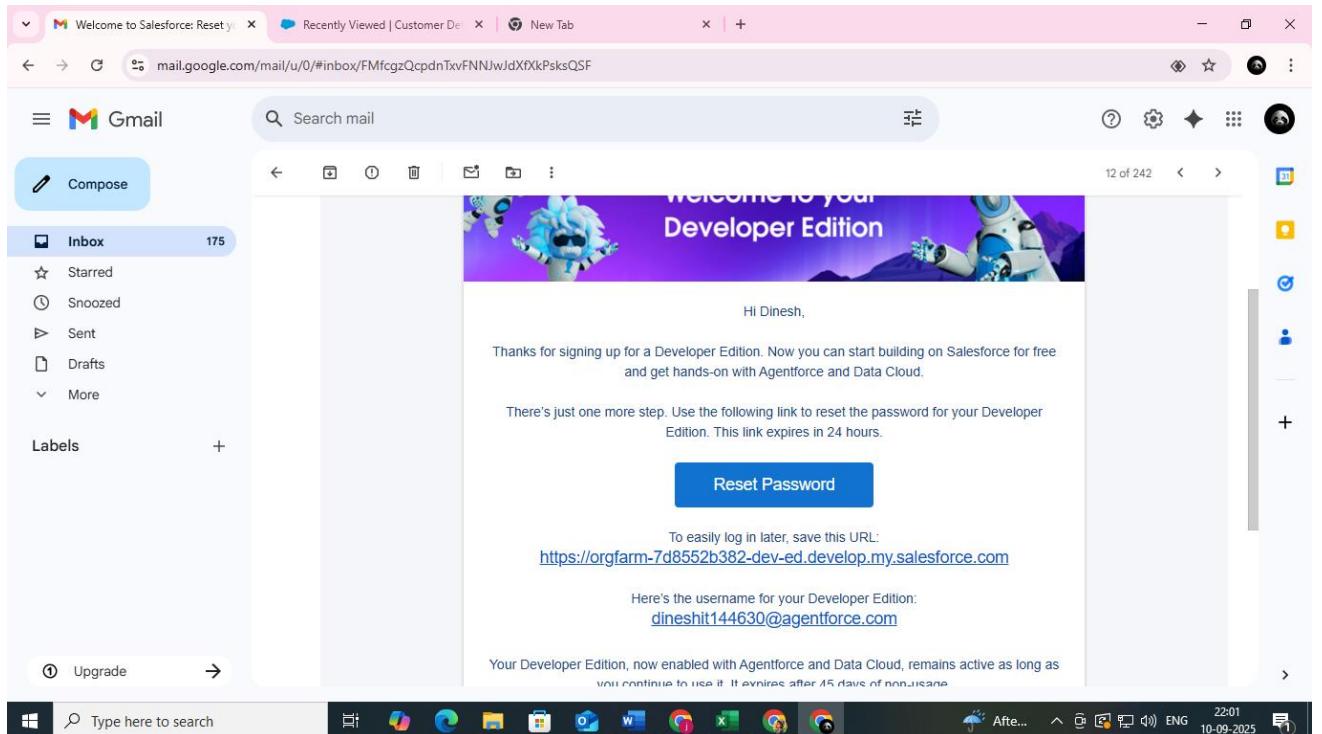
1. First name & Last name
2. Email
3. Role : Developer
4. Company : College Name
5. County : India
6. Postal Code : pin code

Click on sign me up after filling these

---

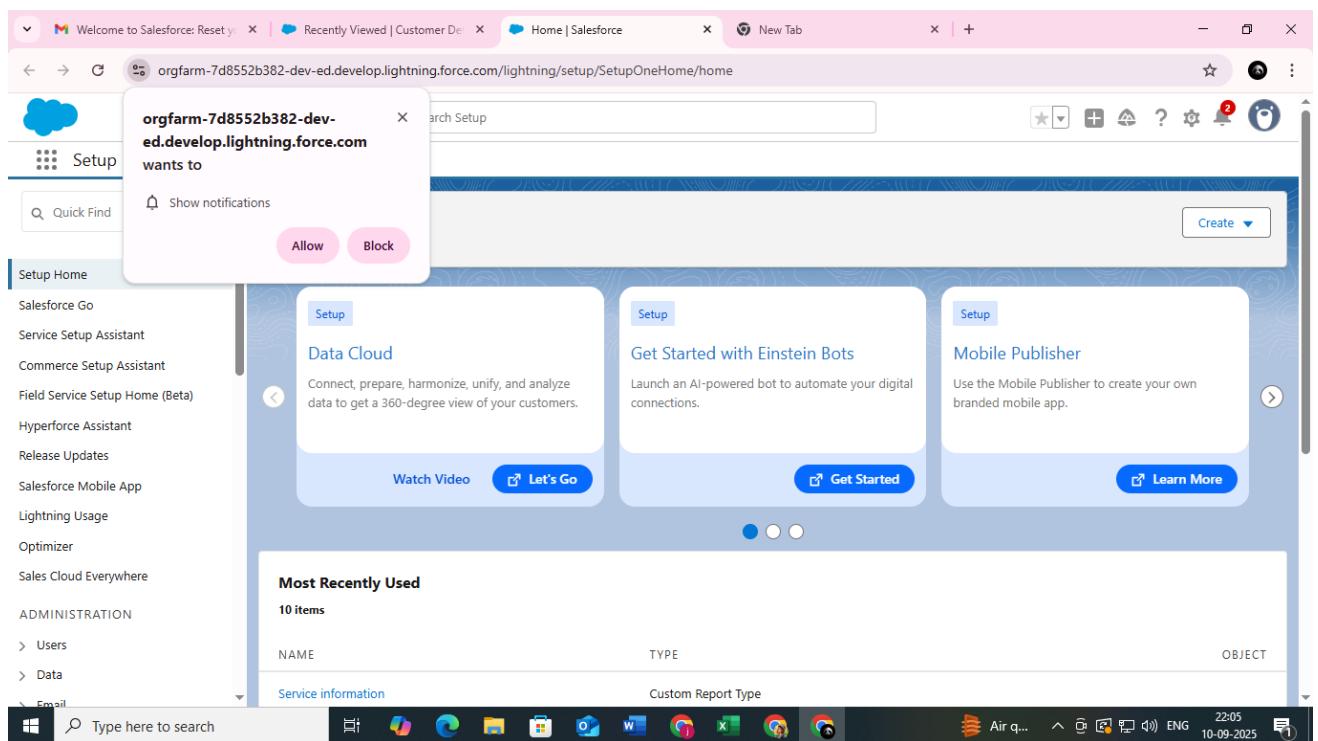
### 4. Account Activation

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. Give a password and answer a security question and click on change password

3. Then you will redirect to your salesforce setup page.



## 5. Created Customer Details Object

To create an object:

1. From the setup page >> Click on Object Manager >> Click on Create >> Click on Custom Object.
2. Enter the label name >> Customer Details
3. Plural label name >> Customer Details
4. Enter Record Name Label and Format
5. Record Name >> Customer Name
6. Data Type >> Text
7. Click on Allow reports and Track Field History,
8. Allow search >> Save.

LABEL	API NAME	TYPE	DESCRIPTION
Custom	Custom Object	Custom	Custom Object

**New Custom Object**

Custom Object Definition Edit

**Custom Object Information**

The singular and plural labels are used in lists, page layouts, and reports.

Label: **Customer Details** Example: Account  
 Plural Label: **Customer Details** Example: Accounts  
 Starts with vowel sound:

The Object Name is used when referencing the object via the API.

Object Name: **Customer\_Details** Example: Account

Description:

Content-Sensitive Help Setting:  Open the standard Salesforce.com Help & Training window  Open a window using a Visualforce page

Content Name: **Customer Details Name** Example: Account Name

**Enter Record Name Label and Format**

The Record Name appears in page layouts, key lists, related lists, lookups, and search results. For example, the Record Name for Account is "Account Name" and for Case it is "Case Number". Note that the Record Name field is always called "Name" when referenced via the API.

Record Name: **Customer Details Name** Example: Account Name

Data Type: **Text** Warning: If you plan to insert a high volume of records in this object, via the API for example, use the Text data type.

**Optional Features**

- Allow Reports
- Allow Activities
- Track Field History
- Allow in Chatter Groups
- Enable Licensing

**New Custom Object**

Content Name: **Customer Details Name** Example: Account Name

**Enter Record Name Label and Format**

The Record Name appears in page layouts, key lists, related lists, lookups, and search results. For example, the Record Name for Account is "Account Name" and for Case it is "Case Number". Note that the Record Name field is always called "Name" when referenced via the API.

Record Name: **Customer Details Name** Example: Account Name

Data Type: **Text** Warning: If you plan to insert a high volume of records in this object, via the API for example, use the Text data type.

**Optional Features**

- Allow Reports
- Allow Activities
- Track Field History
- Allow in Chatter Groups
- Enable Licensing

**Object Classification**

When these settings are enabled, this object is classified as an Enterprise Application object. When these settings are disabled, this object is classified as a Light Application object. Learn more.

- Allow Sharing
- Allow Bulk API Access
- Allow Streaming API Access

**Deployment Status**

- In Development
- Deployed

**Search Status**

When this setting is enabled, your users can find records of this object type when they search. Learn more.

- Allow Search

**Object Creation Options (Available only when custom object is first created)**

- Add Notes and Attachments related list to default page layout
- Launch New Custom Tab Wizard after saving this custom object

In the same way Created Appointment, Service record , Billing details and feedback

Label	API Name	Type	Description	Last Modified	Deployed
Appointment	Appointment_c	Custom Object		9/7/2025	✓
Customer Details	Customer_Detail_c	Custom Object		9/6/2025	✓
Billing details and feedback	Billing_details_and_c	Custom Object		9/6/2025	✓
Service records	Service_records_c	Custom Object		9/6/2025	✓

## 6. Created 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. A new custom tab for 'Customer Details' has been created, labeled 'Customer Details'. It is set to 'Building Block' style and is included in the 'Customer Details' profile. The 'Append' checkbox is checked, and the 'Include in Tabs' checkbox is unchecked. Other sections like Web Tabs, Visualforce Tabs, Lightning Component Tabs, and Lightning Page Tabs are shown but empty.

The screenshot shows the Salesforce Setup interface with the 'Tabs' tab selected. A modal window titled 'New Custom Object Tab' is open, showing Step 1: Enter the Details. The 'Object' dropdown is set to 'Customer Details'. The 'Tab Style' dropdown is set to 'Standard'. There is a note about optional splash page custom links. A text input field for 'Description' is empty. The bottom right of the modal has 'Next' and 'Cancel' buttons.

The screenshot shows the Salesforce Setup interface with the 'Tabs' tab selected. A modal window titled 'Tabs' displays a list of standard and custom tabs. The list includes: Authenticated Website User, App Launcher (standard\_AppLauncher), Community (standard\_Community), Site.com (standard\_Sites), Salesforce Chatter (standard\_Chatter), Content (standard\_Content), Analytics Studio (standard\_Insights), Sales Console (standard\_LightningSalesConsole), Service Console (standard\_LightningService), Sales (standard\_LightingSales), Lightning Usage App (standard\_LightningInstrumentation), Digital Experiences (standard\_SalesforceCMS), Queue Management (standard\_QuesManagement), Data Manager (standard\_DataManager), Subscription Management (standard\_RevenueCloudConsole), Sales Cloud Mobile (standard\_SalesCloudMobile), Bolt Solutions (standard\_LightingBolt), Data Cloud (standard\_Audience360), Approvals (standard\_Approvals), My Service Journey (standard\_MSJApp), Salesforce Scheduler Setup (standard\_LightningScheduler), Automation (standard\_FlowsApp), Garage Management Application (Garage\_Management\_Application), and a checked checkbox for 'Append tab to user's existing personal customizations'. The bottom right of the modal has 'Previous', 'Save', and 'Cancel' buttons.

Now created the Tabs for the remaining Appointment, **Service records**

, **Billing details and feedback**

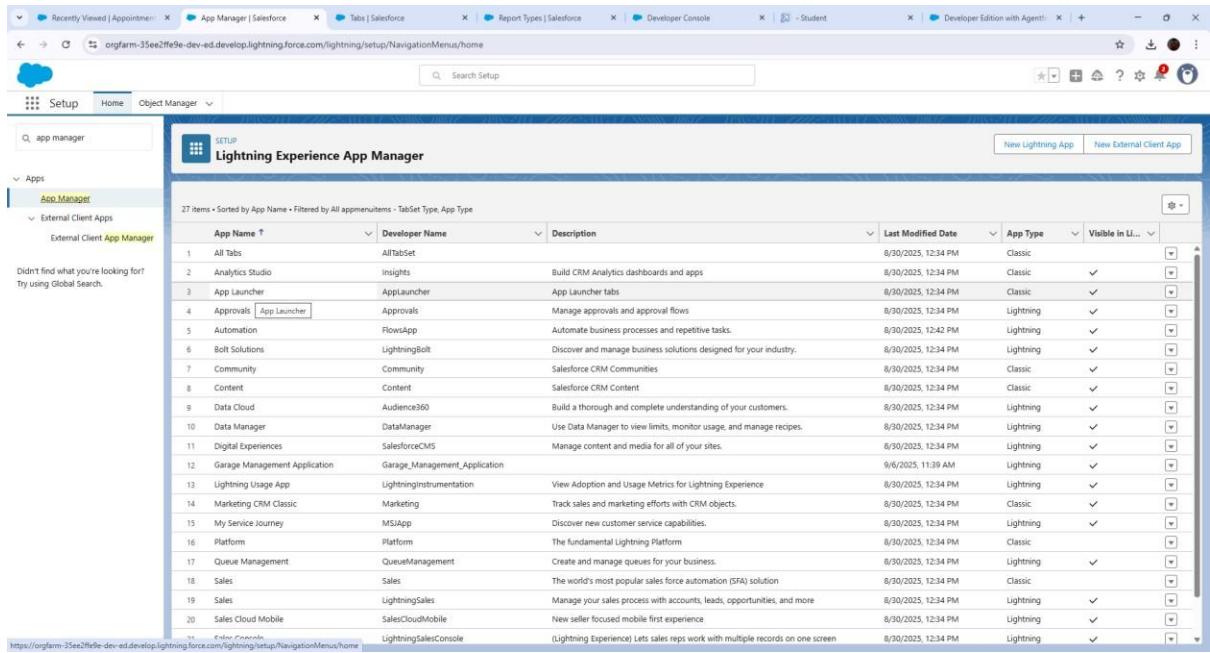
Objects

---

## 7. Created a 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:
6. Search profiles (System administrator) in the search bar >> click on the arrow button >> save & finish.



## 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 ⓘ  
Upload

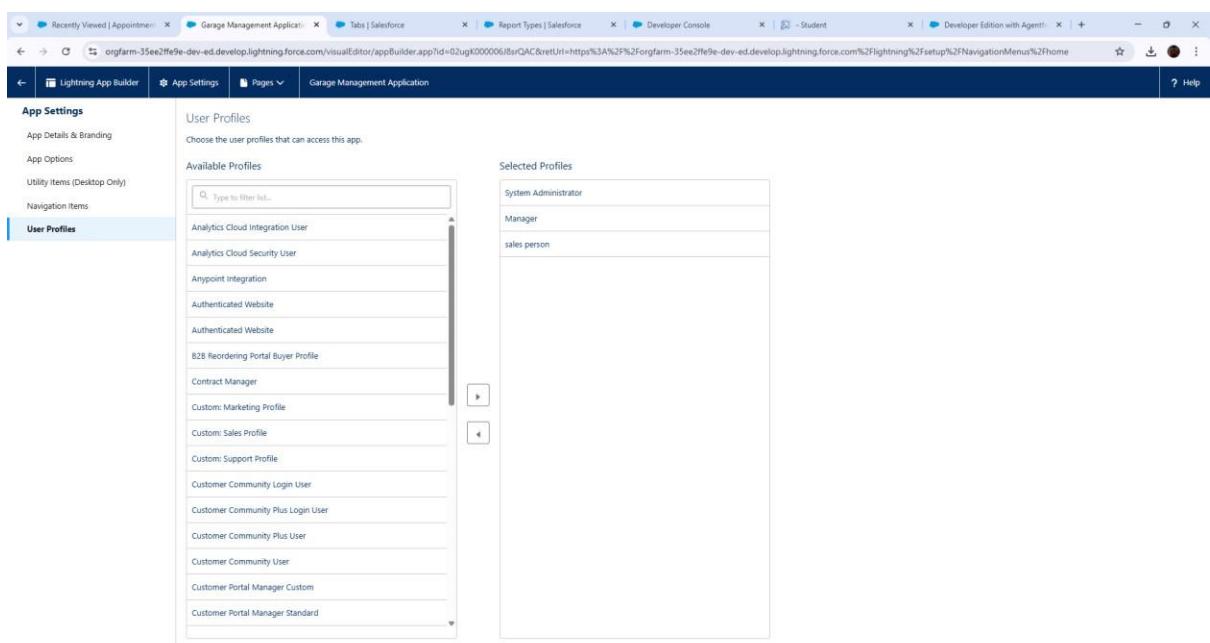
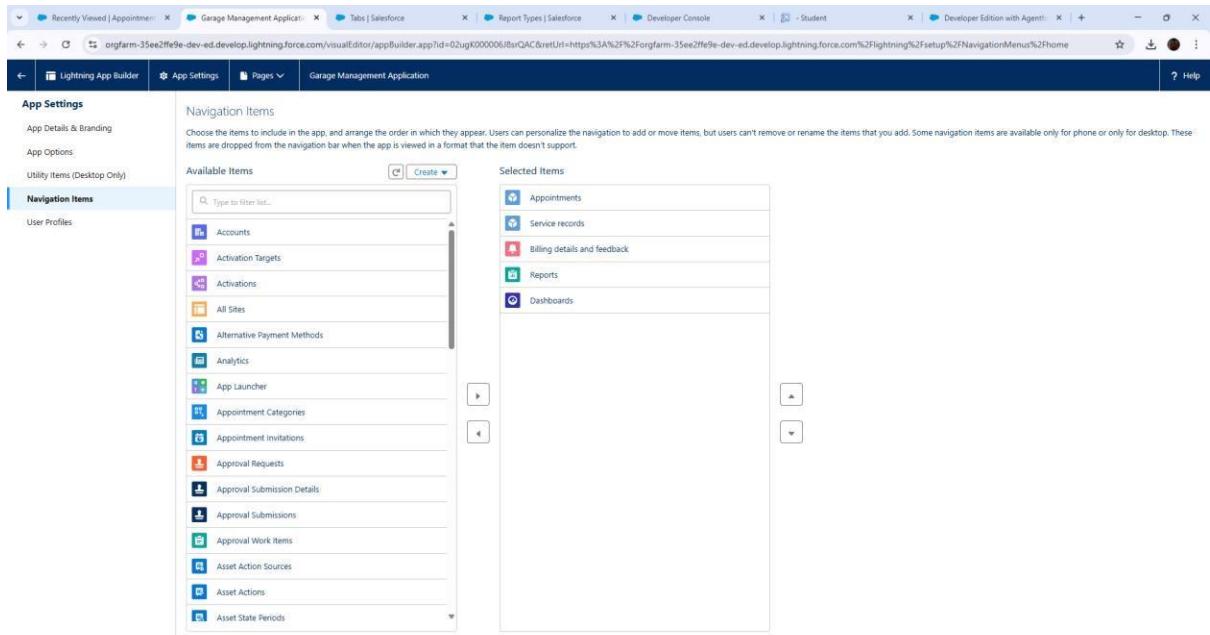
Primary Color Hex Value ⓘ  
#0070D2

### Org Theme Options

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

### App Launcher Preview





## 8. Created a fields for the Customer Details object

### 1. 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.

Object Manager					
Label	API Name	Type	Description	Last Modified	Deployed
Customer Details	Customer_Detail__c	Custom Object		9/6/2025	✓

2. Now click on “Fields & Relationships” >> New

Customer Details					
Fields & Relationships					
FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED	
Created By	CreatedById	Lookup(User)			
Currency	CurrencyIsoCode	Picklist			
Customer Detail 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			

3. Select Data Type as a “Phone”

4. Click on next.

5. Fill the Above as following:

- Field Label: Phone number
- Field Name : gets auto generated
- Click on Next >> Next >> Save and new.

The screenshot shows the Salesforce Setup interface under the 'Customer Details' object. On the left, a sidebar lists various setup categories like Details, Page Layouts, Lightning Record Pages, etc. The main pane is titled 'Fields & Relationships' and shows a list of field types. The 'Phone' field type is selected, highlighted with a blue border. A detailed description of the 'Phone' field type is provided on the right, including its use cases and validation rules.

The screenshot shows the 'Customer Details' object's Fields & Relationships section. A new custom field is being created. The 'Field Label' is set to 'Phone number' and the 'Field Name' is set to 'Phone\_number'. The 'Required' checkbox is unchecked. The 'Auto add to custom report type' checkbox is checked. The 'Default Value' is set to 'Show Formula Editor'. The right side of the screen shows the 'Step 2. Enter the details' configuration page with 'Step 2 of 4' indicated at the top right.

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

Object	Field Label	Data Type	Details / Notes
Customer Details	Phone Number	Phone	Auto-generated Field Name
	Gmail	Email	Auto-generated Field Name

<b>Object</b>	<b>Field Label</b>	<b>Data Type</b>	<b>Details / Notes</b>
	<i>Other fields as required</i>	<i>Various Types</i>	Follow same steps for creating additional fields
<b>Appointment</b>	Maintenance Service	Checkbox	Default Value: Unchecked
	Repairs	Checkbox	Default Value: Unchecked
	Replacement Parts	Checkbox	Default Value: Unchecked
	Appointment Date	Date	Required Field
	Service Amount Currency		Read-Only for all profiles
	Vehicle Number Plate	Text (Length: 10)	Required & Unique
<b>Service Records</b>	Customer (Lookup)	Lookup (Customer Details)	Links Appointment to Customer Details
	Quality Check Status	Checkbox	Default Value: Unchecked
	Service Status	Picklist	Values: Started, Completed
	Appointment (Lookup)	Lookup (Appointment)	Required Field with Filter: <i>Appointment Date &lt; Created Date</i>
	Service Date	Formula (Return Type: Date)	Formula: CreatedDate
<b>Billing Details &amp; Feedback</b>	Payment Paid	Currency	Auto-generated Field Name

<b>Object</b>	<b>Field Label</b>	<b>Data Type</b>	<b>Details / Notes</b>
	Rating for Service	Text (Length: 1)	Required Field
	Payment Status	Picklist	Values: Pending, Completed
	Service Record (Lookup)	Lookup (Service Records)	Links Billing to Service Records

### **created 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 : -  

$$\text{NOT(REGEX( Vehicle_number_plate__c , "[A-Z]{2}[0-9]{2}[A-Z]{2}[0-9]{4}"))}$$
5. 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.

SETUP > OBJECT MANAGER

## Appointment

Validation Rules	Rule Name	Error Location	Error Message	Active	Modified By
	Vehicle	Vehicle number plate	Please enter valid number	✓	Arjunan S, 9/9/2023, 11:36 AM

### Appointment Edit Validation Rule

Define a validation rule by specifying an error condition and a corresponding error message. The error condition is written as a Boolean formula expression that returns true or false. When the formula expression returns true, the save will be aborted and the error message will be displayed. The user can correct the error and try again.

**Validation Rule Edit**

Validation Rule	Save	Save & New	Cancel
Rule Name: <input type="text" value="Vehicle"/> Active: <input checked="" type="checkbox"/> Description: <input type="text"/>	<small>Help for this Page</small> <b>Quick Tips</b> <ul style="list-style-type: none"> <li>Operators &amp; Functions</li> </ul>		
<b>Error Condition Formula</b> <p>Example: <code>Discount_Percent_c &lt; 0.30</code> <a href="#">More Examples...</a></p> <p>Display an error if Discount is more than 30%</p> <p>If this formula expression is true, display the text defined in the Error Message area</p> <div style="border: 1px solid #ccc; padding: 5px;"> <input type="button" value="Insert Field"/> <input type="button" value="Insert Operator"/>   <code>REGEX( Vehicle_number_plate__c , "[A-Z]{2}[0-9]{2}[A-Z]{2}[0-9]{4}" )</code> </div> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <b>Functions</b>   <code>-- All Function Categories --</code> <ul style="list-style-type: none"> <li><code>ABS</code></li> <li><code>ACOS</code></li> <li><code>ADDMONTHS</code></li> <li><code>AND</code></li> <li><code>ASCII</code></li> <li><code>ASIN</code></li> </ul> </div>			
<small>Check Syntax</small>			
<b>Error Message</b> <p>Example: <code>Discount percent cannot exceed 30%</code></p> <p>This message will appear when Error Condition formula is true</p> <p>Error Message: <input type="text" value="Please enter valid number"/></p> <p>This error message can either appear at the top of the page or below a specific field on the page</p> <p>Error Location: <input type="radio"/> Top of Page <input checked="" type="radio"/> Field/ <input type="text" value="Vehicle number plate"/></p>			
<small>Save</small> <small>Save &amp; New</small> <small>Cancel</small>			

## 9. created a validation rule to an Billing details and feedback Object

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	✓	Ajjanan s, 9/8/2025, 1:16 AM

**Validation Rule Edit**

Save Save & New Cancel

**Rule Name:** rating\_should\_be\_less\_than\_5

**Active:**

**Description:**

**Error Condition Formula**

Example:  More Examples...  
Display an error if Discount is more than 30%

If this formula expression is true, display the text defined in the Error Message area

Insert Field Insert Operator NOT( REGEX( Rating\_for\_service\_\_c , "[1-5]{1}" ))

Functions

- All Function Categories --
- ABS
- ACOS
- ADDMONTHS
- AND
- ASCII
- ASIN

Insert Selected Function  
ABS(number)  
Returns the absolute value of a number, a number without its sign  
[Help on this function](#)

**Error Message**

Example:  Discount percent cannot exceed 30%  
This message will appear when Error Condition formula is true

Error Message: rating should be from 1 to 5

## 10.1 created 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
8. Gmail Exact
9. Phone Number Exact
10. Click save.
11. After Saving Click on Activate.

The screenshot shows the Salesforce Matching Rules page. The URL is <https://orgfarm-35ee2fffe-dev-ed.lightning.force.com/lightning/setup/MatchingRules/home>. The page title is "Matching Rules". It displays a table of existing matching rules:

Action	Rule Name	Object	Status	Description	Last Modified Date	Last Modified By
Del   Deactivate	Matching customer details	Customer Details	Active	Matching rule for account records <a href="#">More info</a>	9/9/2025	xyz
Deactivate	Standard Account Matching Rule	Account	Active	Matching rule for contact records <a href="#">More info</a>	8/9/2025	GEPIC
Deactivate	Standard Contact Matching Rule	Contact	Active	Matching rule for lead records <a href="#">More info</a>	8/9/2025	GEPIC
Deactivate	Standard Lead Matching Rule	Lead	Active	Matching rule for account records <a href="#">More info</a>	8/9/2025	GEPIC

The screenshot shows the "New Matching Rule" wizard, Step 1 of 2. The URL is <https://orgfarm-35ee2fffe-dev-ed.lightning.force.com/lightning/setup/MatchingRules/new>. The page title is "Matching Rules". It has a header "Step 1: Select object" and "Step 1 of 2". It includes a note "Select the object to which this matching rule applies." and a dropdown menu set to "Customer Details".

## Edit Rule Matching customer details

Step 2: Configure Matching Rule Step 2 of 2

**Rule Details**

Object: Customer Details  
 Rule Name: Matching customer details  
 Unique Name: Matching\_customer\_details  
 Description:

**Matching Criteria**

Tell the rule which fields to compare and how.

Field	Matching Method	Match Blank Fields	AND
Gmail	Exact	<input type="checkbox"/>	AND
Phone number	Exact	<input type="checkbox"/>	AND
-None--	Exact	<input type="checkbox"/>	AND
-None--	Exact	<input type="checkbox"/>	AND
-None--	Exact	<input type="checkbox"/>	AND

Add Filter Logic... Previous Save Cancel

Matching Rule Matching customer details Help for this Page

**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: Phone\_number EXACT MatchBlank = FALSE)  
 Status: Inactive  
 Created By: Arjunan.s 9/8/2025, 1:20 AM Modified By: Arjunan.s 9/9/2025, 1:04 PM

Edit Delete Clone Activate

## 10.2 Created 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.

Setup Home Object Manager

Dupli

All Duplicate Rules Help for this Page

What Are Duplicate Rules? [Expand]

View: All Duplicate Rules

Rule Name	Description	New Rule	Matching Rule	Active	Last Modified By	Last Modified Date
Customer Detail duplicate	Identify accounts that duplicate other accounts.	Account	Standard Account Matching Rule	<input checked="" type="checkbox"/>	GERIC	9/8/2025
Standard Account Duplicate Rule	Identify contacts that duplicate other contacts and leads.	Address	Standard Contact Matching Rule	<input checked="" type="checkbox"/>	GERIC	8/30/2025
Standard Contact Duplicate Rule	Identify leads that duplicate other leads and contacts.	Apellido	Standard Lead Matching Rule	<input checked="" type="checkbox"/>	GERIC	8/30/2025
Standard Lead Duplicate Rule		Billing details and feedback	Standard Contact Matching Rule	<input checked="" type="checkbox"/>	GERIC	8/30/2025
		Contact	Standard Lead Matching Rule	<input checked="" type="checkbox"/>	GERIC	8/30/2025
		Customer Details	Standard Contact Matching Rule	<input checked="" type="checkbox"/>	GERIC	8/30/2025
		Individual	Standard Lead Matching Rule	<input checked="" type="checkbox"/>	GERIC	8/30/2025
		Lead	Standard Contact Matching Rule	<input checked="" type="checkbox"/>	GERIC	8/30/2025
		Service records	Standard Lead Matching Rule	<input checked="" type="checkbox"/>	GERIC	8/30/2025

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

## 11. Created 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.

The screenshot shows the Salesforce Setup interface with the following details:

- Header:** Includes the Salesforce logo, a search bar labeled "Search Setup", and various navigation icons.
- Left Navigation Bar:** Shows a tree view of the setup categories:
  - Hyperforce Assistant
  - Users
    - Profiles (selected)
  - Data
    - Mass Transfer Approval Requests
    - Feature Settings
    - Approval Settings
    - Data.com
      - Prospector Preferences
      - Prospector Users
    - Functions
  - Marketing
    - Lead Processes
  - Sales
    - Products
      - Asset Settings
      - Product Schedules
      - Settings
    - Sales Processes
    - Salesforce Scheduler
  - Assignment Policies
- Current Page:** The "Profiles" page under the "Users" section. It displays a list of profiles with columns for Action, Profile Name, User License, and Custom status.
- Page Elements:** Includes a "Help for this Page" link and a "Print" icon.

Q prof

**SETUP**

## Profiles

Clone Profile

Enter the name of the new profile

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

You must select an existing profile to clone from.

Existing Profile	Standard User
User License	Salesforce
Profile Name	Manager

Required information

Save Cancel

The screenshot shows the Salesforce Setup interface. The top navigation bar includes 'Setup' (selected), 'Home', 'Object Manager', and a search bar 'Search Setup'. Below the navigation is a sidebar with 'Q prof' and 'Users' (selected). The main content area is titled 'Profiles' and contains a sub-section 'Profile Manager'. A note states: 'Users with this profile have the permissions and page layouts listed below. Administrators can change a user's profile by editing that user's personal information. If your organization uses Record Types, use the Edit links in the Record Type Settings section below to make one or more record types available to users with this profile.' Below this are several permission checkboxes: 'Login IP History', 'Enabled Apex Class Access', 'Enabled Visualforce Page Access', 'Enabled Email Send/Email Bounce Access', 'Enabled External Content Access', 'Enabled External-Cache/Principals Access', 'Enabled Custom Metadata Type Access', 'Enabled Custom Setup Definition Access', 'Enabled Flow Access', 'Enabled Service Processor Status Access', and 'Enabled Custom Permissions'. A 'Profile Detail' section shows the profile for 'Manager'. It lists 'Name: Manager', 'User License: Salesforce', 'Description: ', 'Created By: Arunam S.', 'Created Date: 9/8/2025, 1:15 PM', 'Modified By: Arunam S.', 'Modified Date: 9/8/2025, 1:15 PM', and a 'Custom Profile' checkbox which is checked. Below this is a 'Page Layouts' section showing standard object layouts for various objects like Global, Email Application, Home Page Layout, Account, Alternative Payment Method, and Appointment Invitations, each with their respective layout assignments. A 'Help for this Page' link is at the top right.

## 11.1 Created a sales person Profile

The first screenshot shows the 'Clone Profile' screen where a new profile named 'sales person' is being created based on the 'Salesforce Standard User' profile. The second screenshot shows the 'Profile Detail' screen for the newly created 'sales person' profile, which includes a list of page layouts and custom object permissions.

**Clone Profile Screen:**

Existing Profile	User License	Standard User
Salesforce	Salesforce	Sales person

**Profile Detail Screen:**

Name	sales person	Custom Profile
User License	Salesforce Platform	
Description		Modified By: Arman.s 9/8/2025, 2:41 AM

**Page Layouts:**

Standard Object Layouts	Global	Lead
Global Layout [View Assignment]		Lead Layout [View Assignment]
Email Application Not Assigned [View Assignment]		Location Layout [View Assignment]
Home Page Layout Home Page Default [View Assignment]		Location Group Layout [View Assignment]
Account Account Layout [View Assignment]		Location Group Assignment Location Group Assignment Layout [View Assignment]
Alternative Payment Method Alternative Payment Method Layout [View Assignment]		Object Milestone Object Milestone Layout [View Assignment]
Appointment Invitation Appointment Invitation Layout [View Assignment]		Operating Hours Operating Hours Layout [View Assignment]
Asset Asset Layout [View Assignment]		Order Order Layout [View Assignment]
Asset Relationship Asset Relationship Layout [View Assignment]		Order Product Order Product Layout [View Assignment]
Assigned Resource Assigned Resource Layout [View Assignment]		Payment Payment Layout [View Assignment]
Associated Location Associated Location Layout [View Assignment]		Payment Authorization Payment Authorization Layout [View Assignment]
Armen Operation Log Armen Operation Log Layout [View Assignment]		Payment Authorization Adjustment Payment Authorization Adjustment Layout [View Assignment]
Authorization Form Authorization Form Layout [View Assignment]		Payment Gateway Payment Gateway Layout [View Assignment]
Authorization Form Consent Authorization Form Consent Layout [View Assignment]		Payment Gateway Log Payment Gateway Log Layout [View Assignment]

**Custom Object Permissions:**

Object	Basic Access						Data Administration	
	Read	Create	Edit	Delete	View All	Modify All		
Appointments	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
Billing details and feedback	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
Customer Details	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
Environments	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
Laptops	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
Service records	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
SessionData	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		

## 12. Created Manager Role

Creating Manager Role:

1. Go to quick find >> Search for Roles >> click on set up roles.
2. Click on Expand All and click on add role under whom this role works.
3. Give Label as “Manager” and Role name gets auto populated. Then click on Save.

The screenshot shows the Salesforce Setup Roles page. On the left, there's a sidebar with navigation links like 'Setup', 'Home', 'Object Manager', 'Users', 'Roles', 'Sales', 'Service', 'Case Teams', and 'Feature Settings'. The main content area has a title 'Understanding Roles' and a sub-section 'Sample Role Hierarchy'. It displays a hierarchical tree diagram under the heading 'Territory-based Sample' with nodes like 'Executive Staff', 'CEO', 'President', 'CFO', 'VP, Sales', 'Western Sales Director', 'Eastern Sales Director', 'International Sales Director', 'Western Sales Rep', 'Eastern Sales Rep', and 'International Sales Rep'. Each node has a brief description of its permissions. At the bottom right of the main content area are buttons for 'Set Up Roles' and 'Don't show this page again'.

This screenshot shows the 'Your Organization's Role Hierarchy' section. It displays a detailed tree structure of roles across various departments and regions. The root node is 'sri vasavi college self finance wing', which branches into 'Add Role', 'CEO', 'CFO', 'COO', 'Manager', 'sales person', 'SVP, Customer Service & Support', 'Customer Support, International', 'Customer Support, North America', 'Installation & Repair Services', 'SVP, Human Resources', 'SVP, Sales & Marketing', 'VP, International Sales', 'VP, Marketing', 'Marketing Team', and 'VP, North American Sales'. Each node includes edit, delete, and assign buttons.

The screenshot shows the 'Role Edit' form. The 'Label' field is set to 'Manager', and the 'Role Name' field is also 'Manager'. In the 'This role reports to' dropdown, 'CEO' is selected. Below the form are buttons for 'Save', 'Save & New', and 'Cancel'.

## 12.1. Created another roles

Creating another roles under manager

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

## 13. Created User

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

### 13.1. Created 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 : created 3 users with these permissions.**

## 14. Created a 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 auto populated.
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.

## 15. 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.
6. Give the Label name as “ Sharing setting”
  7. Rule name is auto populated.
  8. In step 3 : Select which records to be shared, members of “ Roles ” >> “ Sales person”
  9. In step 4: share with, select “ Roles ” >> “ Manager ”
  10. In step 5 : Change the access level to “ Read / write ”.
  11. Click on save.

The screenshot shows the 'Sharing Settings' page in the Salesforce setup. At the top, there's a header with 'Sharing Settings' and a 'Save' button. Below the header is a table where each row represents an object and its sharing settings. The columns are 'Object' (list of objects like Service Appointment, Service Contract, etc.), 'Action' (dropdown menu), and 'Criteria' (dropdown menu). The 'Action' column for most objects shows 'Private' or 'Public Read/Write'. The 'Criteria' column for 'Service records' shows 'Private'. At the bottom of the table, there are several checkboxes: 'Standard Report Visibility' (checked), 'Manual User Record Sharing' (unchecked), 'Manager Groups' (unchecked), 'Secure guest user record access' (unchecked), and 'Require permission to view record names in lookup fields' (unchecked). There are also 'Save' and 'Cancel' buttons at the bottom.

The screenshot shows the 'Service records Sharing Rules' page in the Salesforce setup. At the top, there's a header with 'Service records Sharing Rules' and a 'Save' button. Below the header is a table where each row represents an object and its sharing rules. The columns are 'Action' (dropdown menu), 'Criteria' (dropdown menu), 'Shared With' (list of users or groups), and 'Access Level' (dropdown menu). The 'Action' column for most objects shows 'Private' or 'Public Read/Write'. The 'Criteria' column for 'Service records' shows 'Private'. The 'Shared With' column for 'Service records' lists 'Salesperson' and 'Manager'. The 'Access Level' column for 'Service records' shows 'Read / write'. There are also 'New' and 'Recalculate' buttons at the top left.

## 16. Created 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. Set a filter condition : All Conditions are met(AND)
10. Field : Payment\_Status\_\_c
11. Operator : Equals
12. Value : Completed
13. And Set Field Values for the Billing details and feedback Record
14. Field : Payment\_Paid\_\_c
15. Value :
 

```
{ !$Record.Service_records__r.Appointment__r.Service_Amount__c }
```
16. Click On Done.
17. Before creating another Element. Create a New Resource form Toolbox form top left.
18. Click on the New Resource, And select Variable.
19. Select the resource type as text template.
20. Enter the API name as “ alert”.
21. Change the view as Rich Text ? View to Plain Text.
22. In body field paste the syntax that given below.
23. Dear
 

```
{ !$Record.Service_records__r.Appointment__r.Customer_Name__r.Name },
```
24. I hope this message finds you well. I wanted to take a moment to express my sincere gratitude for your recent payment for the services provided by our garage management team. Your prompt payment is greatly appreciated, and it helps us continue to provide top-notch services to you and all our valued customers.
25. Amount paid : 

```
{ !$Record.Payment_Paid__c }
```
26. Thank you for Coming .
27. Click done.
28. Now Click on Add Element , select Action.
29. Their action bar will be opened in that search for “ send email ” and click on it.
30. Give the label name as “ Email Alert”
31. API name will be auto populated.
32. Enable the body in set input values for the selected action.
33. Select the text template that created , Body : 

```
{ !alert }
```

34. Include recipient address list select the email form the record.

35. RecipientAddressList:

```
{ !$Record.Service_records__r.Appointment__r.Customer_Name__r.Gmail__c }
```

36. Include subject as “ Thank You for Your Payment - Garage Management”.

37. Click done.

38. Click on save. Give the Flow label , Flow Api name will be autopopulated.

39. And click save, and click on activate.

The screenshot shows two overlapping pages from the Salesforce setup interface.

The top window is titled "Flows" and displays a list of flow definitions. The columns include Flow Label, Process Type, Active, Template, Package State, Package Name, Last Modified By, and Last Modified Date. The flows listed are:

Flow Label	Process Type	Active	Template	Package State	Package Name	Last Modified By	Last Modified Date
Add or Modify Service Appointment Attendees	Salesforce Scheduler Flow	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Managed-installed			
Approvals Workflow: Evaluate Approval Requests	Screen Flow	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Managed-installed			
Approvals Workflow: Process Approval Submission	Screen Flow	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Managed-installed			
Authentication Provider User Registration	Identity User Registration Flow	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Managed-installed			
Basic Approval Request	Flow Orchestration for CMS	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Managed-installed			

The bottom window is titled "New Automation" and is a wizard for creating a new automation. It has sections for "Get Started with Automations", "Categories" (Triggered, Scheduled, Screen, Autolaunched), "Frequently Used" (Record-Triggered Flow, Screen Flow, Schedule-Triggered Flow, Autolaunched Flow (No Trigger)), and "Select Object". The "Object" field is set to "Billing details and feedback".

## Optimize Flow

Optimize the Flow for:

### Fast Field Updates

Update fields on the record that triggers the flow to run. This high-performance flow runs **before the record is saved** to the database.

### Actions and Related Records

Update any record and perform actions, like send an email. This more flexible flow runs **after the record is saved** to the database.

Is this flow making an external callout or connecting to an external system?

An asynchronous path is required for flows that involve external systems.

Add Asynchronous Path



### Record-Triggered Flow

Start

Object: **Billing details and feedback** [Edit](#)

Trigger: **A record is created or updated**

Optimize for: **Actions and Related Recor...**

[+ Add Scheduled Paths \(Optional\)](#)

[↗ Open Flow Trigger Explorer for Billing ...](#)

[Run Immediately](#)



### Add Element

[update records](#)

**Update Records**

ⓘ



**Set Filter Conditions**

Condition Requirements to Update Record

All Conditions Are Met (AND)

Field	Operator	Value
Payment_Status__c	Equals	Completed

**+ Add Condition**

**Set Field Values for the Billing details and feedback Record**

Field	Value
Payment_Paid__c	\$Record > Service records > Appointment > Service A...

**+ Add Field**

**Cancel** **Done**

← Flow Builder → billing amount flo

Select Elements

**Toolbox**

**Manager**

Search this flow...

New Resource

## Edit Text Template

\* API Name ⓘ

alert

Description

\* Body ⓘ

Insert a resource...



View as Rich Text ▾

Dear {!\$Record.Service\_records\_\_r.Appointment\_\_r.Customer\_Details\_\_r.Name}

I hope this message finds you well. I wanted to take a moment to express my sincere gratitude for your recent payment for the services provided by our garage management team. Your prompt payment is greatly appreciated, and it helps us continue to provide top-notch services to you and all our valued customers.

Amount paid : {!\$Record.Payment\_Paid\_\_c}

Thank you for Coming .

Salesforce Sans ▾

12 ▾



Cancel

Done

## ⚡ Send Email



\* Label

Email Alert

\* API Name ⓘ

Email\_Alert

Description

Use values from earlier in the flow to set the inputs for the "Send Email" core action. To use its outputs later in the flow, store them in variables.

**Send Email**

**A\_a Subject** ⓘ

Thank You for Your Payment - Garage Manager 

**A\_a Body** ⓘ

  alert 

**Rich-Text-Formatted Body** ⓘ

Enter value or search resources... 

**Use Line Breaks** ⓘ

Enter value or search resources... 

**A\_a Attachment ID** ⓘ

 Search records... 

Record-Triggered Flow  
Start

Save as

A New Version  A New Flow

\*Flow Label

\*Flow API Name

Description

Show Advanced

Cancel  Save

## 1. **16.1Created another Flow**

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

## Set Filter Conditions

Condition Requirements to Update Record

All Conditions Are Met (AND) ▾

Field	Operator	Value
Quality_Check_Status__c	Equals	True <span style="color: red;">X</span>

[+ Add Condition](#)

## Set Field Values for the Service record Record

Field	Value
Service_Status__c	Completed <span style="color: red;">X</span>

[+ Add Field](#)

## 17. 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 ”.

```

AmountDistribution.apex * AmountDistributionHandler.apex *
Code Coverage: None API Version: 58 Go To
1 public class AmountDistributionHandler {
2
3     public static void amountDist(list<Appointment__c> listApp){
4         list<Service_records__c> serList = new list <Service_records__c>();
5
6         for(Appointment__c app : listApp){
7             if(app.Maintenance_service__c == true && app.Repairs__c == true && app.Replacement_Parts__c == true){
8                 app.Service_Amount__c = 10000;
9             }
10            else if(app.Maintenance_service__c == true && app.Repairs__c == true){
11                app.Service_Amount__c = 5000;
12            }
13            else if(app.Maintenance_service__c == true && app.Replacement_Parts__c == true){
14                app.Service_Amount__c = 8000;
15            }
16            else if(app.Repairs__c == true && app.Replacement_Parts__c == true){
17                app.Service_Amount__c = 7000;
18            }
19            else if(app.Maintenance_service__c == true){
20                app.Service_Amount__c = 2000;
21            }
22            else if(app.Maintenance_service__c == true && app.Replacement_Parts__c == true){
23                app.Service_Amount__c = 8000;
24            }
25            else if(app.Repairs__c == true && app.Replacement_Parts__c == true){
26                app.Service_Amount__c = 7000;
27            }
28        }
29    }
30 }
31

```

```

AmountDistribution.apex * AmountDistributionHandler.apex *
Code Coverage: None API Version: 58 Go To
12
13    else if(app.Maintenance_service__c == true && app.Replacement_Parts__c == true){
14        app.Service_Amount__c = 8000;
15    }
16    else if(app.Repairs__c == true && app.Replacement_Parts__c == true){
17        app.Service_Amount__c = 7000;
18    }
19    else if(app.Maintenance_service__c == true){
20        app.Service_Amount__c = 2000;
21    }
22    else if(app.Repairs__c == true){
23        app.Service_Amount__c = 3000;
24    }
25    else if(app.Replacement_Parts__c == true){
26        app.Service_Amount__c = 5000;
27    }
28
29 }
30 }
31

```

**Code:**

```

public class AmountDistributionHandler {
    public static void amountDist(list<Appointment__c> listApp){
        list<Service_records__c> serList = new list <Service_records__c>();

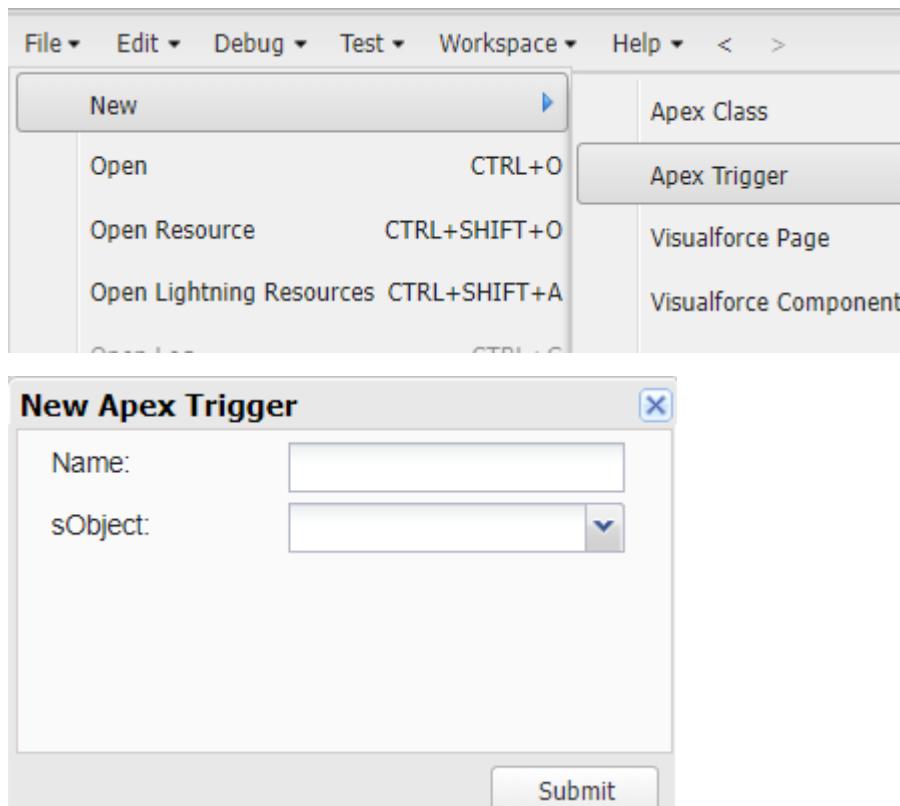
```

```
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){  
        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;  
    }  
}  
}
```

## 17.1. 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 :

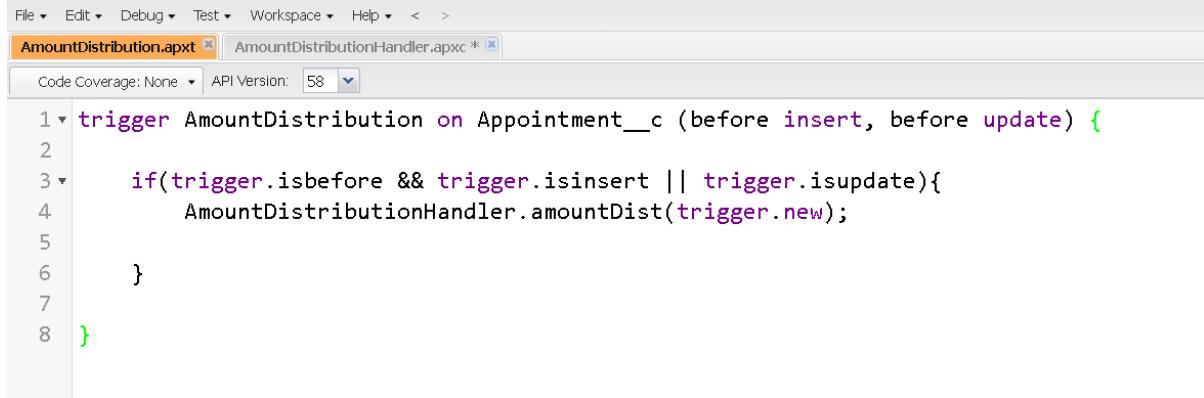
The syntax for creating trigger is :

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 IDE interface with the following details:

- File menu: File, Edit, Debug, Test, Workspace, Help.
- Toolbar: Standard icons for file operations.
- Tab bar: AmountDistribution.apxt (highlighted in orange) and AmountDistributionHandler.apxc \*.
- Status bar: Code Coverage: None, API Version: 58.
- Code editor:

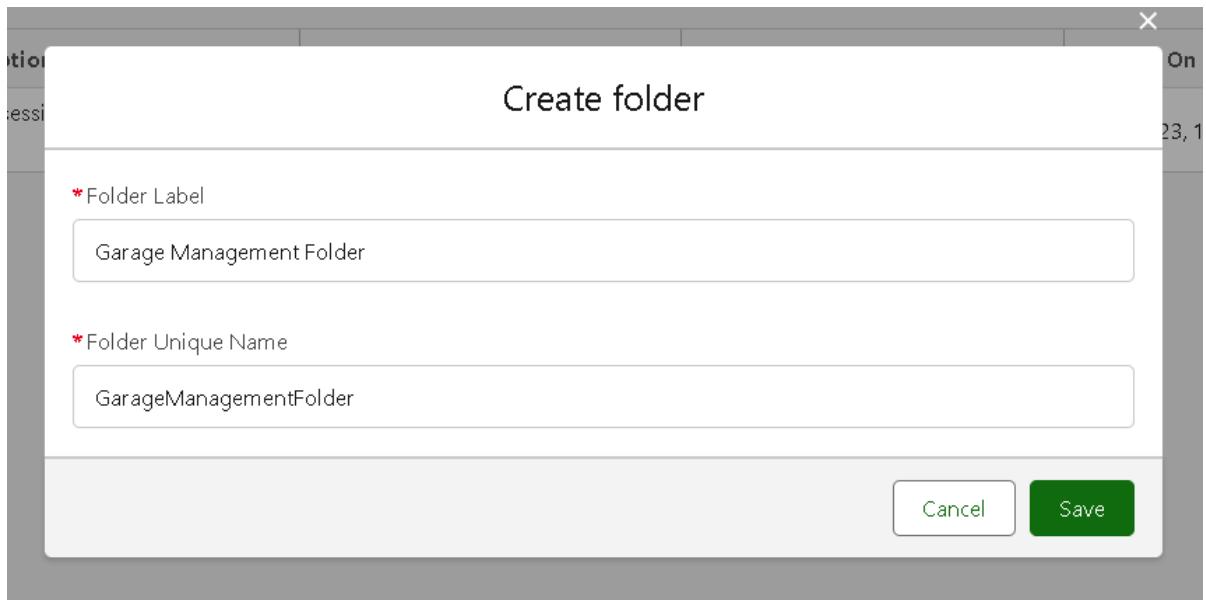
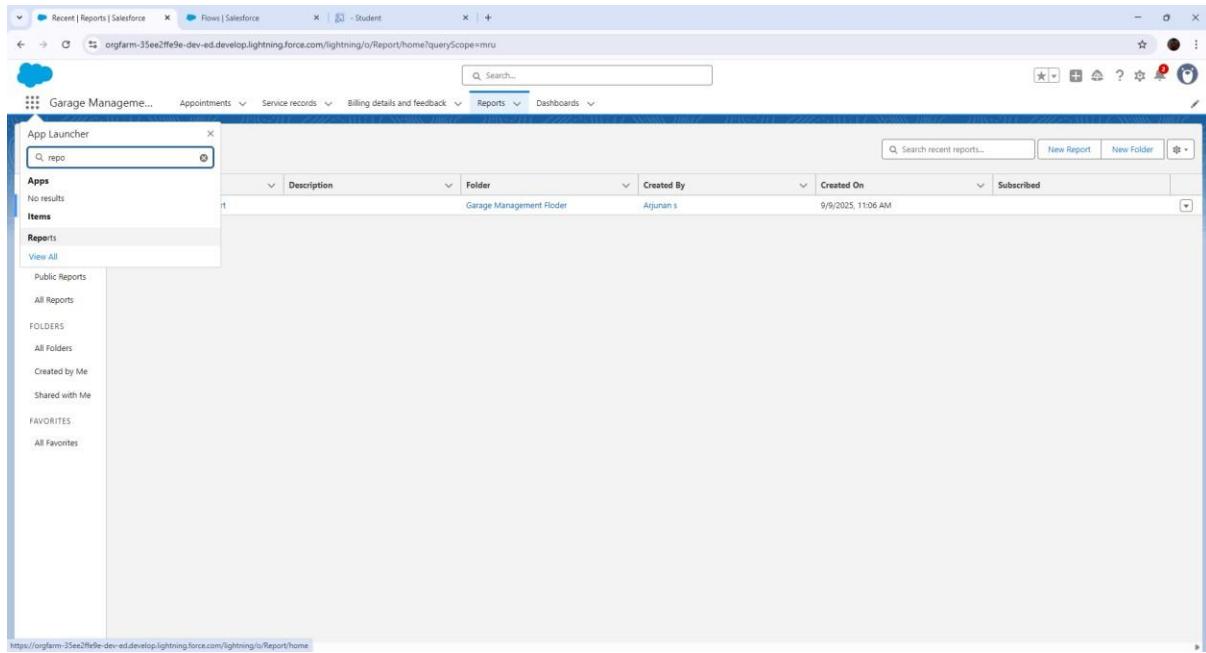
```
1 trigger AmountDistribution on Appointment__c (before insert, before update) {
2
3     if(trigger.isbefore && trigger.isinsert || trigger.isupdate){
4         AmountDistributionHandler.amountDist(trigger.new);
5
6     }
7
8 }
```

Code:

```
trigger AmountDistribution on Appointment__c (before insert, before update) {
    if(trigger.isbefore && trigger.isinsert || trigger.isupdate){
        AmountDistributionHandler.amountDist(trigger.new);
    }
}
```

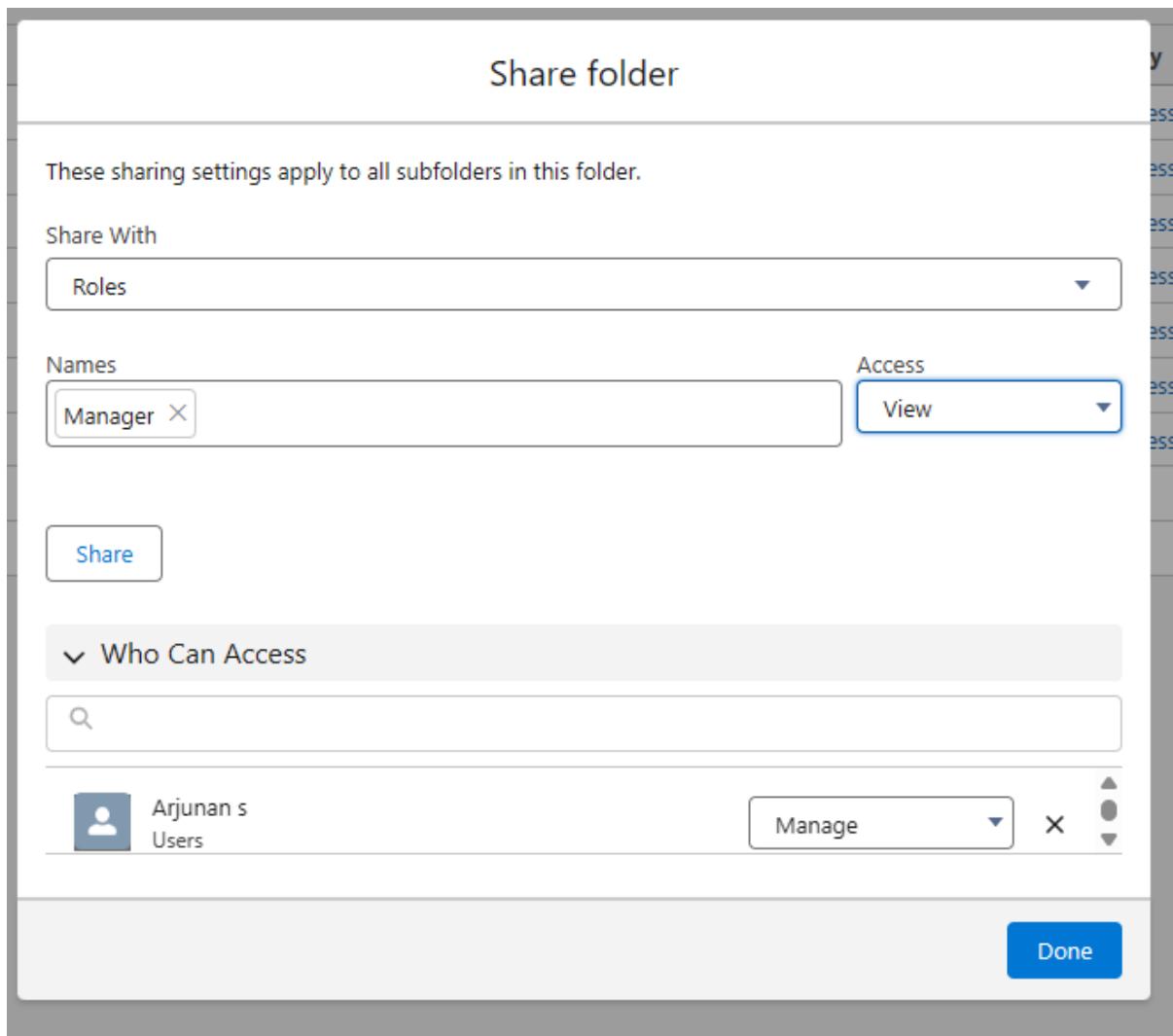
## 18. Created A Report Folder

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



## 18.1. Shared 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.



## 18.2. Created 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 “Depolyed ”, 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.

Label	Name	Description	Category	Created Date
Orchestration Run Logs Spring '24	flow_orchestration_log_oott_crt_two_four_eight	Find out which orchestration run logs were created and what happened in their associated orchestration runs.	Other Report	8/30/2025, 12:34 PM
Orchestration Runs Spring '24	flow_orchestration_run_oott_crt_two_four_eight	Find out which orchestration runs were created.	Other Report	8/30/2025, 12:34 PM
Orchestration Stage Runs Spring '24	flow_orchestration_stage_run_oott_crt_two_four_eight	Find out which orchestration stage runs were created and the current status of each run.	Other Report	8/30/2025, 12:34 PM
Orchestration Step Runs Spring '24	flow_orchestration_step_run_oott_crt_two_four_eight	Find out which orchestration step runs were created and the current status of each run.	Other Report	8/30/2025, 12:34 PM
Orchestration Work Items Spring '24	flow_orchestration_work_item_oott_crt_two_four_eight	Find out which orchestration work items were created, who's the associated assignee, and what's the current status of each item.	Other Report	8/30/2025, 12:34 PM
Program Definition Spring '24	Program_Definition_sfDCSESV0	Review your analytics with a program-like structure. See each program task, target day, results, and more directly in a report.	Other Report	8/30/2025, 12:34 PM
Program Definition Summer '24	Program_Definition_sfDCSESV1	Review your analytics with a program-like structure. See each program task, target day, results, and more directly in a report.	Other Report	8/30/2025, 12:34 PM
Program Item Progress Spring '24	Program_Task_Progress_sfDCSESV0	Report on tasks like exercises, milestones, and outcomes progress. Overall program progress isn't captured in this report.	Other Report	8/30/2025, 12:34 PM
Program Item Progress Summer '24	Program_Task_Progress_sfDCSESV1	Report on tasks like exercises, milestones, and outcomes progress. Overall program progress isn't captured in this report.	Other Report	8/30/2025, 12:34 PM
Program Progress Spring '24	Program_Progress_sfDCSESV0	Report on program progress. Specific progress on milestones and exercises aren't captured in this report.	Other Report	8/30/2025, 12:34 PM
Program Progress Summer '24	Program_Progress_sfDCSESV1	Report on program progress. Specific progress on milestones and exercises aren't captured in this report.	Other Report	8/30/2025, 12:34 PM
Screen Flows	screen_flows_prebuilt_crt	Find out which flows get executed and how long users take to complete each flow screen.	Other Report	8/30/2025, 12:34 PM
Service information	Service_information	Service information	Other Report	9/9/2025, 10:59 AM

**1 Define the Custom Report Type**

**Report Type Focus**

**Primary Object**: Customer Details

**Details**

\*Display Label: Service information   \*API Name: Service\_information

\*Description: Service information

Note: Description will be visible to users who create reports.

\*Store in Category: Other Reports

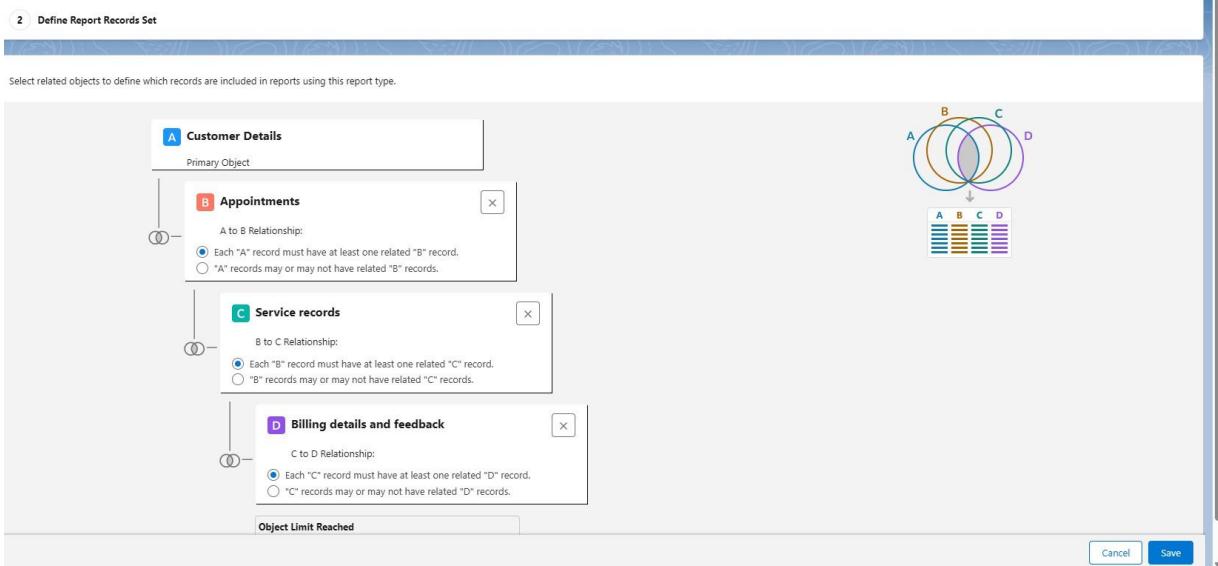
**Set Availability**

An in-development report type is visible only to users with the Manage Custom Report Types permission. A deployed report type is available to all users.

Status:

- In Development
- Deployed

**Save**



### 18.3. CreatedReport

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.
5. 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
6. Remove the unnecessary fields.
7. Select the fields that mentioned below in GROUP ROWS section.
  - a. Rating for Service
8. Select the fields that mentioned below in GROUP ROWS section.
  - a. Payment Status
9. Click on Add Chart , Select the Line Chart.

10. Click on save, Give the report Name : New Service information Report
11. Report unique Name is auto populated.
12. Select the folder the created and Click on save.

The screenshot shows the 'Reports' section of the Garage Management software. On the left, there's a sidebar with 'Recent' and 'Folders' sections. The main area displays a table with columns: Report Name, Description, Folder, Created By, Created On, and Subscribed. One row is selected, showing 'New Service information Report' under 'Report Name' and 'Garage Management Folder' under 'Folder'. The 'Created On' field shows '9/9/2025, 11:06 AM'.

The screenshot shows the 'Create Report' dialog. On the left, a sidebar lists categories like 'Recently Used', 'Accounts & Contacts', 'Opportunities', etc. The main area shows a table of report types. A specific row for 'Service information' is highlighted, showing its details: 'Custom Report Type', 'Start Report' button, 'Description' (Service information), 'Created By You' (New Service information Report), and 'Created By Others' (No Reports Yet). The 'Objects Used in Report Type' section lists 'Customer Details', 'Appointment', 'Service records', and 'Billing details and feedb...'. The right side of the dialog has tabs for 'Details', 'Fields (44)', and 'Fields'.

The screenshot shows the preview of the 'New Service information Report'. At the top, it says 'Previewing a limited number of records. Run the report to see everything.' Below is a table with columns: Rating for service, Payment Status, Completed, and Total. The table data is as follows:

Rating for service	Payment Status	Completed	Total
4	Sum of Payment Paid Record Count	₹15,000 4	₹15,000 4
5	Sum of Payment Paid Record Count	₹5,000 2	₹5,000 2
<b>Total</b>	Sum of Payment Paid Record Count	₹20,000 6	₹20,000 6

To the right of the table is a line chart titled 'Sum of Payment Paid' vs 'Rating for service'. The chart shows a downward trend from approximately 16K at rating 4 to 4K at rating 5. At the bottom, there are buttons for 'Save & Run', 'Save', 'Close', and 'Run'.

Save Report

* Report Name	New Service information Report
Report Unique Name	New_Service_Information_Report_Xo1
Report Description	
Folder	Garage Management folder <input type="button" value="Select Folder"/>
<input type="button" value="Cancel"/> <input type="button" value="Save"/>	

## 19. Created 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.

Create folder

* Folder Label	Service Rating
* Folder Unique Name	ServiceRating
<input type="button" value="Cancel"/> <input type="button" value="Save"/>	

6. Follow the same steps, from Reports Milestone and Activity 2, and provide the sharing settings for the folder that was just created.

### 19.1 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,

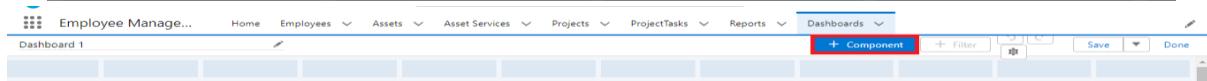
New Dashboard

\* Name  
Customer Review

Description

Folder  
Service Rating dashboard

3.

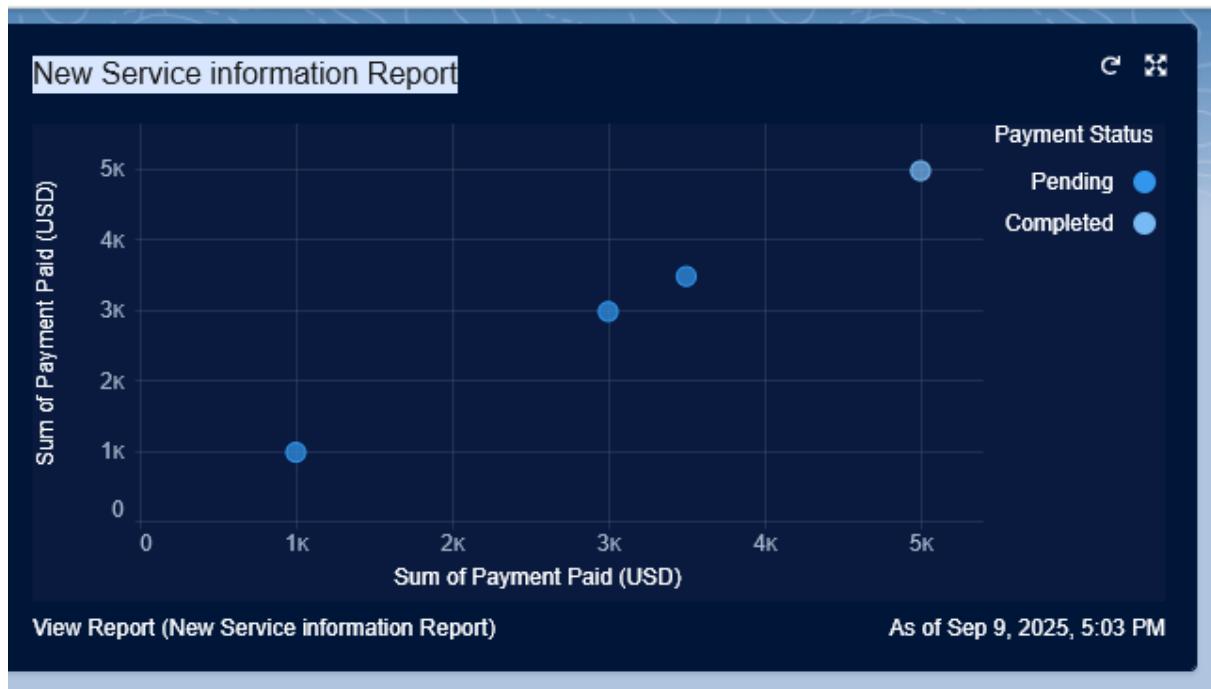


4. Select a Report and click on select.

Select Report

Reports	Selected Report
Recent	Q Search Reports and Folders... <input type="button" value="Reports and Folders"/>
Created by Me	
Private Reports	
Public Reports	
All Reports	
Folders	
Created by Me	
Shared with Me	
All Folders	

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.



## Edit Subscription

Schedule dashboard refreshes and subscribe to receive results.

Settings

Frequency

- 
- 
- 

Days

- 
- 
- 
- 
- 
- 
- 

Time

3:00 PM

Recipients

**!** Recipients see the same report data as the person running the report.

Receive new results by email when dashboard is refreshed. (i)

Send email to

Me

[Edit Recipients](#)

[Cancel](#) [Save](#)

## 20. User Adoption

### Created Records

To create a record in the follow objects follow these steps

1. Click on the app launcher located at the left side of the screen.
2. Search for “**Garage Management**” and click on it.
3. Click on the “**Consumer details tab**”.
4. Click on new and fill the details as shown below figs, and click save.

The screenshot shows the 'New Customer Detail' form. At the top right, there is a note: '\* = Required Information'. The form has a section titled 'Information' containing fields for 'Customer Name' (Mac), 'Phone number' (5678765567), and 'Gmail' (mac@gmail.com). To the right of these fields, under 'Owner', is a profile icon and the text 'Annapurna SmartBridge'. At the bottom of the form are three buttons: 'Cancel', 'Save & New' (highlighted in blue), and 'Save'.

Now, Create the Appointment Record

1. Click on the “**Appointment tab**”.
2. Enter the customer details as created, while entering Appointment Date enter the date less than the created date.
3. Match the validation while entering the vehicle number plate.
4. Select the services you need.
5. Click on save to see the Service Amount.

Garage Management... Customer Details Appointments Service records Billing details and feedback Reports Dashboards

**Appointment app-016**

Appointment Name: app-016 Owner: Annapurna SmartBridge

Customer Details: Mac

\* Appointment Date: 13/11/2024

Maintenance service:

Repairs:

Replacement Parts:

Service Amount:

\* Vehicle number plate: TS30EU0443

Created By: Annapurna SmartBridge, 18/11/2024, 3:28 pm Last Modified By: Annapurna SmartBridge, 18/11/2024, 3:28 pm

Cancel Save

## Now, Create a service Record

1. Click on the “Service record tab”.
2. Enter the Appointment, and started is selected as default.
3. Click on save.

New Service record

\* = Required Information

Information

Service Record Name: app-016 Owner: Annapurna SmartBridge

\* Appointment: app-016

Quality Check Status:

Service Status: Started

Cancel Save & New Save

4. Open the record and click on Quality check status as true.
5. Click on save.

Service Record Name  
ser-010

Owner  
 Annapurna SmartBridge

\* Appointment  
 app-016 X

Quality Check Status ↻

Service Status  
 Started

service date  
18/11/2024  
*This field is calculated upon save*

Created By  
 Annapurna SmartBridge, 18/11/2024, 4:32 pm

Cancel Save Last Modified By  
 Annapurna SmartBridge, 18/11/2024, 4:34 pm

6. Now automatically Service status will be moved to completed.

Related Details

Service Record Name  
ser-010

Owner  
 Annapurna SmartBridge Person

Appointment  
[app-016](#) Edit

Quality Check Status Edit

Service Status  
Completed Edit

service date  
18/11/2024

Created By  
 Annapurna SmartBridge, 18/11/2024, 4:32 pm

Last Modified By  
 Annapurna SmartBridge, 18/11/2024, 4:34 pm