

GARAGE MANAGEMENT SYSTEM

Project Overview:

This project focuses on the Garage Management System (GMS), designed to tackle the challenge of enhancing service quality and operational efficiency in automotive repair facilities. By leveraging Salesforce technology, the goal is to provide a comprehensive solution that improves operational efficiency, user experience, and customer relationship management, ultimately supporting the long-term success of automotive repair businesses.

Objectives:

Business Goals:

- Enhance service quality in automotive repair facilities.
- Increase operational efficiency, reduce costs and maximize productivity.
- Build and maintain strong customer relationships to foster loyalty and repeat business.
- Position the garage as a leading service provider in a competitive market.

Specific Outcomes:

- **Improved Customer Experience:** Implement a user-friendly interface for easier scheduling and communication with customers.
- **Streamlined Operations:** Automate routine tasks, such as inventory management and appointment scheduling, save time and reduce human error.
- **Data Accuracy:** Utilize Salesforce technology to ensure accurate and real-time data tracking and reporting.
- **Enhanced CRM:** Develop a robust customer relationship management system to keep track of customer history, preferences, and feedback.
- **Comprehensive Reporting:** Provide detailed reports and analytics to monitor performance and make informed decisions.

Salesforce Key Features and Concepts Utilized

Fields and Objects

- **Fields:** Custom fields are specific data points relevant to the automotive repair process.
- **Objects:** Custom objects represent key entities such as vehicles, service appointments, and customer profiles.

Lightning App

- A user-friendly and customizable interface for managing garage operations efficiently.

Flows

- Automate processes such as appointment scheduling, invoicing, and follow-up communications with customers.

Triggers

- Implement real-time actions based on specific conditions, such as sending notifications when a service is completed.

Apex Triggers

- Advanced custom logic handles complex business processes and integrations.

Relationships

- Define relationships between objects to ensure seamless data connectivity, such as linking customers to their vehicles and service records.

Different Data Types

- Utilize various data types to accurately capture and store information, such as text, numbers, dates, and picklists.

Profiles and Users

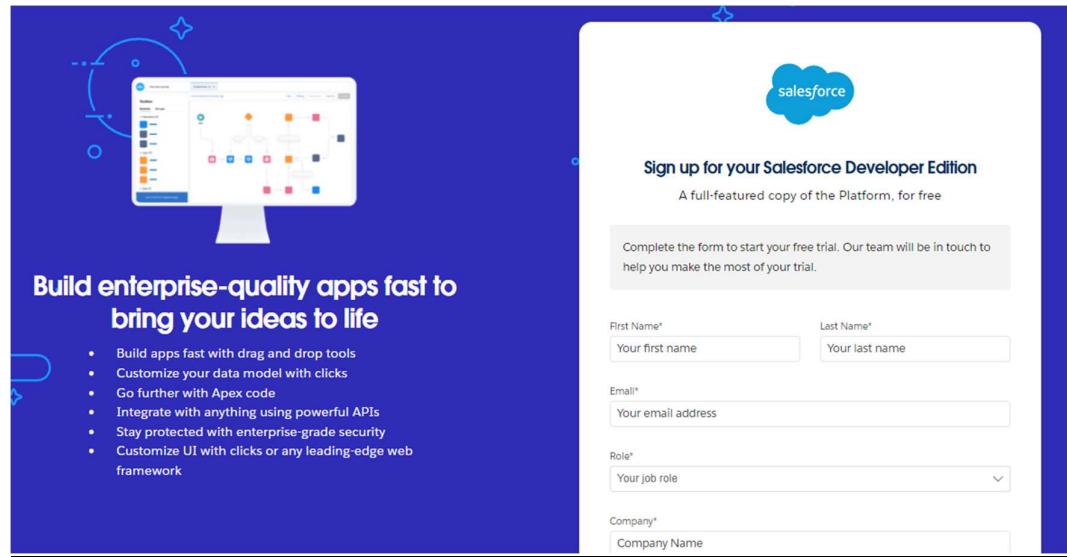
- Manage user access and permissions to ensure data security and proper functionality within the system.

Reports and Dashboards

- Create comprehensive reports and visual dashboards to monitor performance, track key metrics, and make informed decisions.

Detailed Steps Solution Design

The sign up form, enter the following details :



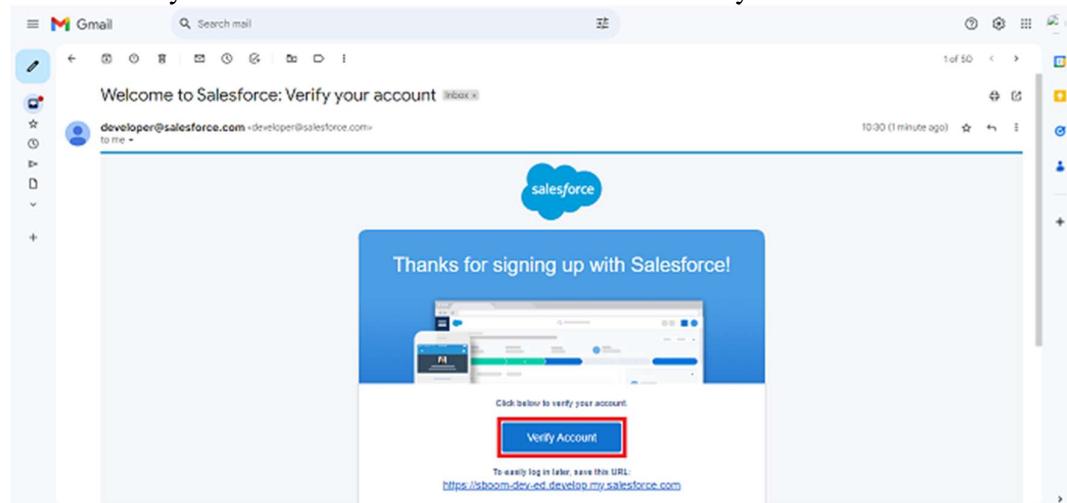
1. First name & Last name
2. Email
3. Role : Developer
4. Company : College Name
5. County : India
6. Postal Code : pin code
7. Username : should be a combination of your name and company

This need not be an actual email id, you can give anything in the format : `username@organization.com`

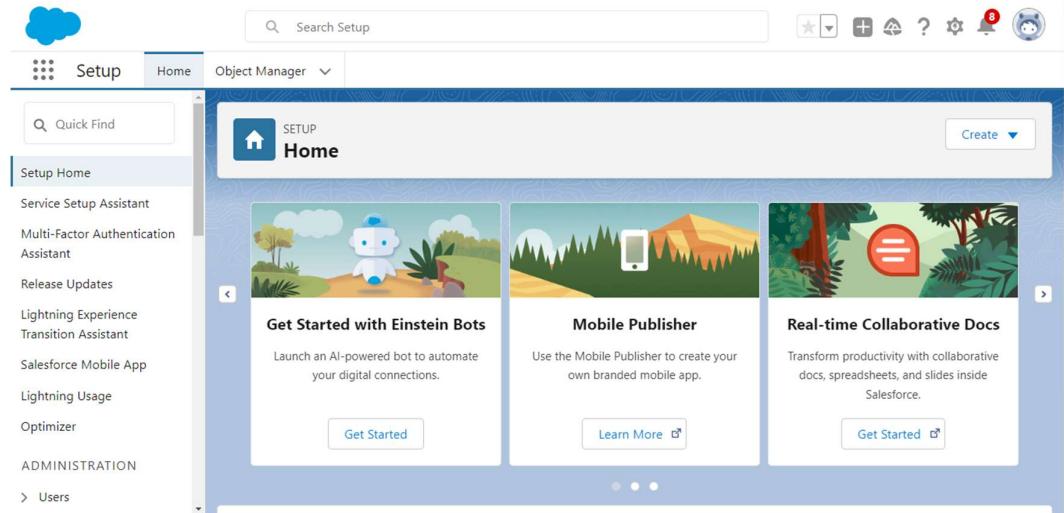
Click on sign me up after filling these.

Account Activation

1. Go the inbox of the email that you used while signing up. Click on the verify account 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.
4. Then you will redirect your salesforce setup page.



CREATED Cusmer Details Object:

CREATED an object:

1. From the setup page >> Click on Object Manager >> Click on CREATED >> Click on Cusm Object.
 - Enter the label name >> Cusmer Details
 - Plural label name >> Cusmer Details
 - Enter Record Name Label and Format
 - Record Name >> Cusmer Name
 - Data Type >> Text
2. Click on Allow reports and Track Field Hisry,
3. Allow search >> Save.

CREATED Appointment Object

CREATED an object:

1. From the setup page >> Click on Object Manager >> Click on CREATED >> Click on Cusm Object.
 - Enter the label name >> Appointment
 - Plural label name >> Appointments
 - Enter Record Name Label and Format
 - Record Name >> Appointment Name

- Data Type >> Au Number
 - Display Format >> app-{000}
 - Starting number >> 1
2. Click on Allow reports and Track Field Hisry,
 3. Allow search >> Save.

CREATED Service records Object

CREATED an object:

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

CREATED Billing details and feedback Object

CREATED an object:

1. From the setup page >> Click on Object Manager >> Click on CREATED >> Click on Cusm Object.
 1. Enter the label name >> Billing details and feedback
 2. Plural label name >> Billing details and feedback
 3. Enter Record Name Label and Format
 - Record Name >> Billing details and feedback Name
 - Data Type >> Au Number
 - Display Format >> bill-{000}
 - Starting number >> 1
2. Click on Allow reports and Track Field Hisry,
 3. Allow search >> Save.

Creating a Cusm Tab

CREATED a Tab:(Cusmer Details)

1. Go setup page >> type Tabs in Quick Find bar >> click on tabs >> New (under cusm object tab)

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

- Setup Home button (highlighted with a red box and arrow).
- Search bar (highlighted with a red box and arrow).
- User Interface section.
- Recent Test and Labels section.
- Custom Tabs section (highlighted with a red box and arrow).
- New button (highlighted with a red box and arrow).

The main content area displays a table of "Custom Object Tabs" with columns for Action, Label, Tab Style, and Description. One row is selected, showing details for a tab labeled "Customer Details".

Action	Label	Tab Style	Description
Edit	Customer	Desk	created to setup with student activity(junction object)
Edit	Reservations	Airplane	
Edit	Courses	Chess piece	
Edit	Students	Jewel	This tab is related to Hotel Reservation App
Edit	Books	Airplane	
Edit	Books	Phone	
Edit	Books	Camera	This tab is related to College Management System
Edit	Books	Desk	
Edit	Books	Computer	
Edit	Books	Desk	
Edit	Books	Highway Sign	This tab is related to Hotel Reservation App
Edit	Books	Highway Sign	This tab is related to Hotel Reservation App
Edit	Books	Airplane	
Edit	Books	Phone	This tab is related to College Management System

2. Select Object(Cusmer Details) >> Select the tab style >> Next (Add profiles page) keep it as default >> Next (Add Cusm App) uncheck the include tab .
3. Make sure that the Append tab users' existing personal cumsizations is checked.
4. Click save.

The screenshot shows the "New Custom Object Tab" configuration dialog, Step 1: Enter the Details:

Step 1. Enter the Details

Choose the custom object for this new custom tab. Fill in other details.

Select an existing custom object or [create a new custom object now](#).

Object: Customer Details

Tab Style: (highlighted with a red box and arrow)

(Optional) Choose a Home Page Custom Link to show as a splash page the first time your users click on this tab.

Splash Page Custom Link: None

Enter a short description.

Description: (empty text input field)

Next Cancel

Tab Style Selector

Create your own style

Hide styles which are used on other tabs

Airplane	Alarm clock	Apple	Balls
Bank[1]	Bell	Big top	Boat[1]
Books	Bottle	Box	Bridge
Building	Building Block	Caduceus	Camera
Can	Car	Castle	CD/DVD
Cell phone	Chalkboard	Chess piece	Chip
Circle	Compass	Computer	Credit card
CRT TV	Cup	Desk[1]	Diamond
Dice	Factory	Fan	Flag
Form	Gears	Globe	Guitar
Hammer	Hands	Handsaw	Headset
Heart[1]	Helicopter	Hexagon	Highway Sign
Hot Air Balloon	Insect	IP Phone	Jewel
Keys	Laptop	Leaf	Lightning

Save **Cancel**

Step 3. Add to Custom Apps

Choose the custom apps for which the new custom tab will be available. You may also examine or alter the visibility of tabs from the detail and edit pages of each Custom App.

Custom App	<input type="checkbox"/> Include Tab
Platform (standard_Platform)	<input type="checkbox"/>
Sales (standard_Sales)	<input type="checkbox"/>
Service (standard_Service)	<input type="checkbox"/>
Marketing (standard_Marketing)	<input type="checkbox"/>
Sample Console (standard_ServiceConsole)	<input type="checkbox"/>
High Volume Customer Portal User	<input type="checkbox"/>
Authenticated Website User	<input type="checkbox"/>
App Launcher (standard_AppLauncher)	<input type="checkbox"/>

Step 3. Add to Custom Apps

Choose the custom apps for which the new custom tab will be available. You may also examine or alter the visibility of tabs from the detail and edit pages of each Custom App.

Custom App	<input type="checkbox"/> Include Tab
Analytics Studio (standard_Insights)	<input type="checkbox"/>
Sales Console (standard_LightningSalesConsole)	<input type="checkbox"/>
Service Console (standard_LightningService)	<input type="checkbox"/>
Sales (standard_LightningSales)	<input type="checkbox"/>
Lightning Usage App (standard_LightningInstrumentation)	<input type="checkbox"/>
Digital Experiences (standard_SalesforceCMS)	<input type="checkbox"/>
Queue Management (standard_QueueManagement)	<input type="checkbox"/>
Bolt Solutions (standard_LightningBolt)	<input type="checkbox"/>
Data Manager (standard_DataManager)	<input type="checkbox"/>
Salesforce Scheduler Setup (standard_LightningScheduler)	<input type="checkbox"/>

Append tab to users' existing personal customizations

Previous **Save** **Cancel**

The Lightning App

An app is a collection of items that work together to serve a particular function. In Lightning Experience, Lightning apps give your users access to sets of objects, tabs, and other items all in one convenient bundle in the navigation bar.

Lightning apps let you brand your apps with a custom color and logo. You can even include a utility bar and Lightning page tabs in your Lightning app. Members of your org can work more efficiently by easily switching between apps.

CREATED a Lightning App

CREATED a lightning app page:

1. Go setup page >> search “app manager” in quick find >> select “app manager” >> click on New lightning App.

The screenshot shows the Salesforce App Manager interface. At the top, there is a search bar with the text "app manager" and a "Search Setup" button. Below the search bar, there are tabs for "Setup", "Home", and "Object Manager". A red box highlights the "app manager" search term. To the right of the search bar, there is a "New Lightning App" button, which is also highlighted with a red box. Below the search bar, there is a message about cloning apps in beta. A red arrow points to the "Clone Apps(Beta)" link. On the left side of the main content area, there is a sidebar with a "Cloud Apps(Beta)" section, which is also highlighted with a red box. The main content area displays a table of existing apps, with columns for "App Name", "Developer Name", "Description", "Last Modified", "App Type", and "Vl...". The table lists 13 items, including "All Tabs", "Analytics Studio", "App Launcher", "Bolt Solutions", "Chatter Desktop", "Chatter Mobile for BlackBerry", "College Management System", "Community", "Content", "Data Manager", "AI Test", "Alfredset", "Insights", "AppLauncherTab", "Lightningbolt", "ChatterDesktop", "ChatterMobile", "demodata", "Community", "Content", and "DataManager". The "App Type" column indicates whether the app is "Classic", "Lightning", or "Connected (Managed)". The "Last Modified" column shows dates ranging from April 12, 2022, to December 28, 2022.

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. 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. Add User Profiles:

Search profiles (System administrator) in the search bar >> click on the arrow button >> save & finish.

Fields

When we talk about Salesforce, Fields represent the data stored in the columns of a relational database. It can also hold any valuable information that you require for a specific object. Hence, the overall searching, deletion, and editing of the records become simpler and quicker.

Types of Fields

1. Standard Fields
2. Custom Fields

Standard Fields:

As the name suggests, the Standard Fields are the predefined fields in Salesforce that perform a standard task. The main point is that you can't simply delete a Standard Field until it is a non-required standard field. Otherwise, users have the option to delete them at any point from the application freely. Moreover, we have some fields that you will find common in every Salesforce application. They are,

- CREATED By
- Owner
- Last Modified
- Field Made During object Creation

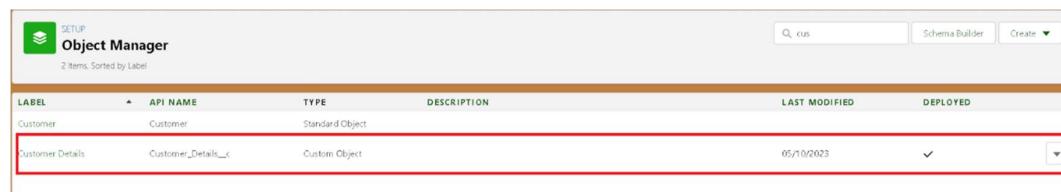
Custom Fields:

On the other side of the coin, Custom Fields are highly flexible, and users can change them according to requirements. Moreover, each organization or company can use them if necessary. It means you need not always include them in the records, unlike Standard fields. Hence, the final decision depends on the user, and he can add/remove Custom Fields of any given form.

Creation of fields for the Customer Details object

1. CREATED fields in an object:

1. Go to setup >> click on Object Manager >> type object name (Customer Details) in search bar >> click on the object.



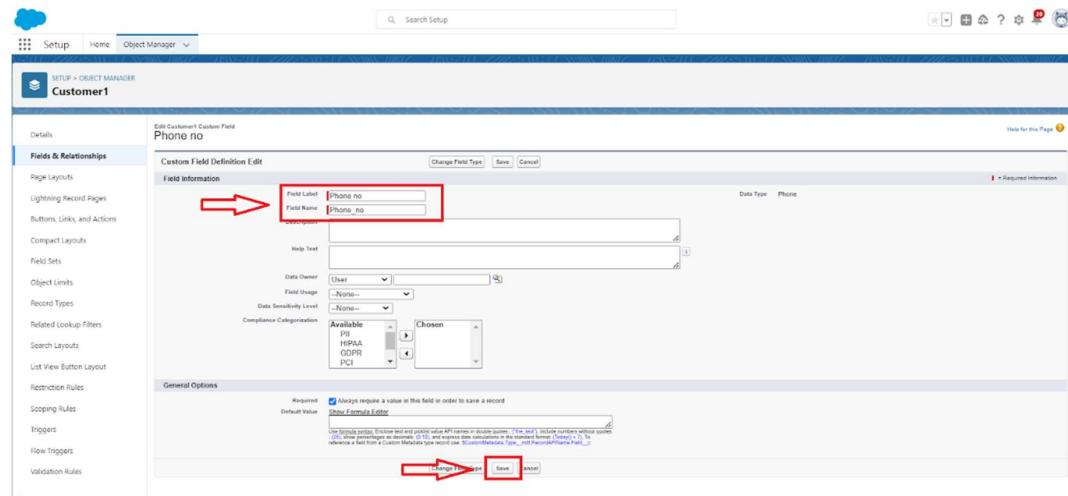
The screenshot shows the Salesforce Object Manager. At the top, there's a search bar with 'cus' and a 'Schema Builder' button. Below the header, a table lists objects: 'Customer' (Standard Object) and 'Customer_Details__c' (Custom Object). The 'Customer_Details__c' row is highlighted with a red border. The table has columns: LABEL, API NAME, TYPE, DESCRIPTION, LAST MODIFIED, and DEPLOYED.

LABEL	API NAME	TYPE	DESCRIPTION	LAST MODIFIED	DEPLOYED
Customer	Customer	Standard Object			
Customer Details	Customer_Details__c	Custom Object		05/10/2023	

2. click on "Fields & Relationships" >> New

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.

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

2. CREATED another fields in an object:

1. Go setup >> click on Object Manager >> type object name(Cusmer Details) in search bar >> click on the object.
2. click on “Fields & Relationships” >> New
3. Select Data type as a “Email” and Click on Next
4. Fill the Above as following:

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

Creation of Lookup Fields

Creation of Lookup Field on Appointment Object :

1. Go setup >> click on Object Manager >> type object name(Appointment) in the search bar >> click on the object.

Label	API Name	Type	Description	Last Modified	Deployed
Appointment	Appointment_c	Custom Object		24/08/2023	✓
Appointment Category	AppointmentCategory	Standard Object			
Appointment Invitation	AppointmentInvitation	Standard Object			
Appointment Invitee	AppointmentInvitee	Standard Object			

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

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Appointment Date	Appointment_Date_c	Date		
Appointment Name	Name	Auto Number		

3. Select “Look-up relationship” as data type and click Next.

Specify the type of information that the custom field will contain.

Data Type

- None Selected Select one of the data types below.
- Auto Number A system-generated sequence number that uses a display format you define. The number is automatically incremented for each new record.
- Formula A read-only field that derives its value from a formula expression you define. The formula field is updated when any of the source fields change.
- Roll-Up Summary A read-only field that displays the sum, minimum, or maximum value of a field in a related list or the record count of all records listed in a related list.
- Lookup Relationship Creates a relationship that links this object to another object. The relationship field allows users to click on a lookup icon to select a value from a popup list. The other object is the source of the values in the list.
- Master-Detail Relationship Creates a special type of parent-child relationship between this object (the child, or "detail") and another object (the parent, or "master") where:
 - The relationship field is required on all detail records.
 - The ownership and sharing of a detail record are determined by the master record.

4. Select the related object “ Cusmer Details ” and click next.

5. Next >> Next >> Save.

Note: Make sure you complete Activity 4 Before continuing.

Creation of Lookup Field on Service records Object :

1. Go setup >> click on Object Manager >> type object name(Service records) in search bar >> click on the object.
2. click on “Fields & Relationships” >> New
3. Select “Look-up relationship” as data type and click Next.
4. Select the related object “ Appointment ” and click next.
5. Make it a required field so click on Required.

Lookup Options

Related To: Appointment

Related List Label: Service records

Child Relationship Name: Service_records

Required: Always require a value in this field in order to save a record
 Clear the value of this field. You can't choose this option if you make this field required.
 Don't allow deletion of the lookup record that's part of a lookup relationship.

6. Scroll down for Lookup Filter and click on Show filter settings.
7. add the filter criteria.

8. Field : Appointment: Appointment Date >> Operar : less than >> select field >> Appointment: CREATEDd Date
9. Filter type should be Required.

Lookup Filter

Optional, create a filter to limit the records available to users in the lookup field. [Tell me more!](#)

[Hide Filter Settings](#)

Filter Criteria [Insert Suggested Criteria](#)

Field	Operator	Value / Field
Appointment: Appointment Date	less than	Field: Appointment: Created Date
All	Begin typing to search for a field...	-None-
	Value	

Filter Type **Required.** The user-entered value must match filter criteria.
If it doesn't, display this error message on save:
Value does not exist or does not match filter criteria. [Reset to default message](#)

Optional. The user can remove the filter or enter values that don't match criteria.

Lookup Window Text Add this informational message to the lookup window.

Active Enable this filter.

[Change Field Type](#) [Save](#) [Cancel](#)

10. Error Message : Value does not match the criteria.
11. Enable the filter by click on Active.
12. Next >> Next >> Save.

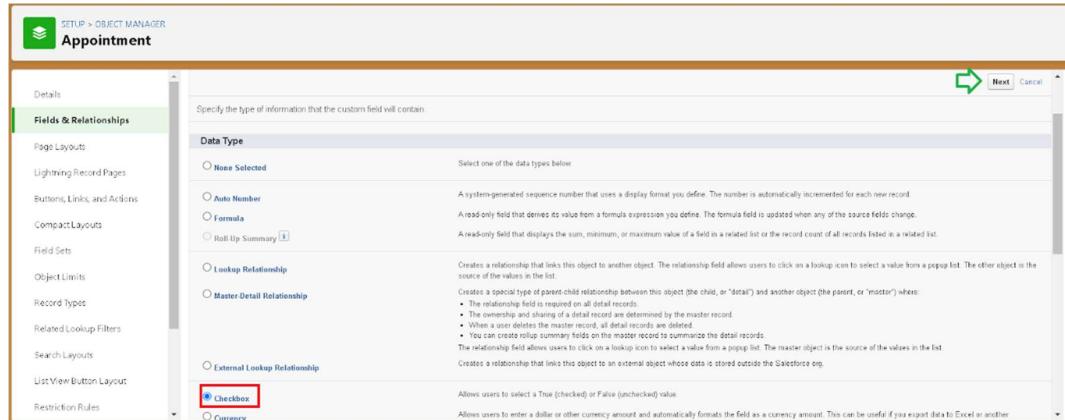
Creation of Lookup Field on Billing details and feedback Object :

1. Go setup >> click on Object Manager >> type object name(Billing details and feedback) in search bar >> click on the object.
2. click on “Fields & Relationships” >> New.
3. Select “Look-up relationship” as data type and click Next.
4. Select the related object “ Service records” and click next.
5. Next >> Next >> Save & new.

Creation of Checkbox Fields

Creation of Checkbox Field on Appointment Object :

1. Go setup >> click on Object Manager >> type object name(Appointment) in search bar >> click on the object.
2. click on “Fields & Relationships” >> New.
3. Select “Check box” as data type and click Next.



4. Give the Field Label : Maintenance service
5. Field Name : is au populated
6. Default value : unchecked

Appointment
New Custom Field

Step 2. Enter the details Step 2 of 4

Field Label	<input type="text" value="Maintenance service"/>
Default Value	<input checked="" type="radio"/> Checked <input checked="" type="radio"/> Unchecked
Field Name	<input type="text" value="Maintenance_service"/>
Description	<input type="text"/>
Help Text	<input type="text"/>

Auto add to custom report type Add this field to existing custom report types that contain this entry

Previous Next Cancel

7. Click on next >> next >> save.

Creation of Another Checkbox Field on Appointment Object :

1. Repeat the steps form 1 - 3.
2. Give the Field Label : Repairs
3. Field Nme : is au populated
4. Default value : unchecked
5. Click on next >> next >> save.
6. Following the same and CREATED another checkbox with given names
7. Give the Field Label : Replacement Parts
8. Field Nme : is au populated
9. Default value : unchecked
10. Click on next >> next >> save.

Creation of Checkbox Field on Service records Object :

1. Go setup >> click on Object Manager >> type object name(Service records) in search bar >> click on the object.
2. click on “Fields & Relationships” >> New.
3. Select “Check box” as data type and click Next.
4. Give the Field Label : Quality Check Status
5. Field Nme : is au populated
6. Default value : unchecked
7. Click on next >> next >> save

Creation of date Fields

Creation of Date Field on Appointment Object :

1. Go setup >> click on Object Manager >> type object name(Appointment) in the search bar >> click on the object.
2. click on “Fields & Relationships” >> New.
3. Select “Date” as data type and click Next.
4. Give the Field Label : Appointment Date
5. Field Nme : is au populated
6. Make it as a Required field by click on the Required option.
7. Click on next >> next >> save.

The screenshot shows the 'New Custom Field' wizard, Step 2: Enter the details. The 'Field Label' is set to 'Appointment Date'. The 'Field Name' is 'Appointment_Date'. The 'Required' checkbox is checked. A green arrow points to the 'Required' checkbox. Other fields like Description and Help Text are empty. Buttons for Previous, Next, and Cancel are visible at the top right.

Creation of Currency Fields

Creation of Currency Field on Appointment Object :

1. Go setup >> click on Object Manager >> type object name(Appointment) in the search bar >> click on the object.
2. click on “Fields & Relationships” >> New.

3. Select “Currency” as data type and click Next.
4. Give the Field Label : Service Amount
5. Field Name : is au populated

Step 2. Enter the details Step 2 of 4

Field Label: Service Amount

Please enter the length of the number and the number of decimal places. For example, a number with a length of 8 and 2 decimal places can accept values up to "12345678.90".

Length: 18	Number of digits to the left of the decimal point	Decimal Places: 0	Number of digits to the right of the decimal point
Field Name: Service_Amount	Description:		
Help Text:			

Required: Always require a value in this field in order to save a record
 Add this field to existing custom report types that contain this entity

6. Click on next
7. Give read only for all the profiles in field level security for profile.

Appointment
New Custom Field Step 3 of 4

Select the profiles to which you want to grant edit access to this field via field-level security. The field will be hidden from all profiles if you do not add it to field-level security.

Field-Level Security for Profile	Visible	Read Only
Analytics Cloud Integration User	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Analytics Cloud Security User	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Authenticated Website	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Authenticated Website	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Contract Manager	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Cross Org Data Proxy User	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

8. Click on next >> save.

Creation of Currency Field on Billing details and feedback Object :

1. Following the same steps as mentioned above in Billing details and feedback Object.
2. Change the label name as mentioned.
3. Give the Field Label : Payment Paid
4. Field Name : is au populated

Creation of Text Fields

1. Go setup >> click on Object Manager >> type object name(Appointment) in the search bar >> click on the object.
2. click on “Fields & Relationships” >> New.
3. Select “Text” as data type and click Next.
4. Give the Field Label : Vehicle number plate

5. Field Name : is au populated
6. Length : 10
7. Make field as Required and Unique.

Step 2. Enter the details

Field Label: Vehicle number plate

Length: 10

Field Name: Vehicle_number_plate

Description:

Help Text:

Required: Always require a value in this field in order to save a record

Unique: Do not allow duplicate values
 Treat "ABC" and "abc" as different values (case insensitive)
 Treat "ABC" and "abc" as duplicate values (case insensitive)

External ID: Set this field as the unique record identifier from an external system

Auto add to custom report type: Add this field to existing custom report types that contain this entity

8. Click on next >> next >> save.

Creation of Text Fields in Billing details and feedback object :

1. Go setup >> click on Object Manager >> type object name(Billing details and feedback) in search bar >> click on the object.
2. click on “Fields & Relationships” >> New.
3. Select “text” as data type and click Next.
4. Give the Field Label : Rating for service
5. Field Name : is au populated
6. Length : 1
7. Make field as Required.
8. Click on next >> next >> save

Creation of Picklist Fields

Creation of Picklist Fields in Service records object :

1. Go setup >> click on Object Manager >> type object name(Service records) in search bar >> click on the object.
2. Click on fields & relationship >> click on New.
3. Select Data type as “Picklist” and click Next.
4. Enter Field Label as “Service Status”, under values select “Enter values, with each value separated by a new line” and enter values as shown below.
5. The values are: Started, Completed.

New Custom Field

Step 2. Enter the details Step 2 of 4

Field Label i

Values Use global picklist value set
 Enter values, with each value separated by a new line

Started
Completed

Display values alphabetically, not in the order entered
 Use first value as default value i
 Restrict picklist to the values defined in the value set i

Field Name i

Description

6. Click Next.
7. Next >> Next >> Save.

Creation of Picklist Fields in Billing details and feedback object :

1. Go setup >> click on Object Manager >> type object name(Billing details and feedback) in search bar >> click on the object.
2. Click on fields & relationship >> click on New.
3. Select Data type as “Picklist” and click Next.
4. Enter Field Label as “Payment Status”, under values select “Enter values, with each value separated by a new line” and enter values as shown below.
5. The values are: Pending, Completed.
6. Click Next.
7. Next >> Next >> Save.

Creating Formula Field in Service records Object

1. Go setup >> click on Object Manager >> type object name(Service records) in search bar >> click on the object.
2. Click on fields & relationship >> click on New.
3. Select Data type as “Formula” and click Next.
4. Give Field Label and Field Name as “service date” and select formula return type as “Date” and click next.

Step 2. Choose output type Step 2 of 5

Field Label i

Field Name i U

Auto add to custom report type Add this field to existing custom report types that contain this entity i

Formula Return Type

None Selected Select one of the data types below.

Checkbox Calculate a boolean value
Example: `[TODAY() > CloseDate]`

Currency Calculate a dollar or other currency amount and automatically format the field as a currency amount.
Example: `Gross Margin = Amount - Cost_c`

Date Calculate a date, for example, by adding or subtracting days to other dates.
Example: `Reminder Date + CloseDate - 7`

Date/Time Calculate a datetime, for example, by adding a number of hours or days to another datetime.
Example: `Next = NOW() + 1`

5. Insert field formula should be : CREATEDdDate

The top screenshot shows the 'Insert Field' dialog. On the left, there's a tree view with 'Service records >' expanded, showing options like '\$Api >', '\$Label >', '\$Organization >', '\$Profile >', '\$System >', '\$User >', '\$UserRole >'. To its right is a list of fields: 'Appointment', 'Appointment >', 'Created By >', 'Created By ID', 'Created Date' (which is highlighted in blue), 'Last Activity Date', 'Last Modified By >', 'Last Modified By ID', and 'Last Modified Date'. On the far right, it says 'You have selected: CreatedDate', 'Type: Date/Time', and 'API Name: CreatedDate', with an 'Insert' button.

The bottom screenshot shows the 'Step 3. Enter formula' dialog. It has tabs for 'Simple Formula' and 'Advanced Formula', with 'Simple Formula' selected. The formula input field contains 'service dates (Date) = CreatedDate'. A green arrow points to the 'CreatedDate' part of the formula. To the right, there's a 'Functions' dropdown set to 'All Function Categories -->' containing 'ABS', 'ACOS', 'ADDMONTHS', 'AND', and 'ASCII'. A 'Quick Tips' box is open, showing 'Getting Started' and 'Operators & Functions'.

6. click "Check Syntax".
 7. Click next >> next >> Save.

Validation rule

Validation rules are applied when a user tries save a record and are used check if the data meets specified criteria. If the criteria are not met, the validation rule triggers an error message and prevents the user from saving the record until the issues are resolved.

CREATED a validation rule an Appointment Object

1. Go 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 : -

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.

CREATED a validation rule an Billing details and feedback Object

1. Go the setup page >> click on object manager >> From drop down click edit for Billing details and feedback object.

- Click on the validation rule >> click New.
- Enter the Rule name as “rating_should_be_less_than_5”.

The screenshot shows the 'Validation Rule Edit' interface. At the top, there are buttons for 'Save', 'Save & New', and 'Cancel'. Below these, the 'Rule Name' field contains the value 'rating_should_be_less_than_5', which is highlighted with a red box. The 'Active' checkbox is checked. A 'Description' field is present but empty. On the right side, there is a 'Quick Tips' section with a link to 'Operators & Functions'. At the bottom, there is an 'Error Condition Formula' section with a note 'R = Required Information'.

- Insert the Error Condition Formula as :-

NOT(REGEX(Rating_for_service__c , "[1-5]{1}"))

- Enter the Error Message as “rating should be from 1 to 5”, select the Error location as Field and select the field as “Rating for Service”, and click Save.

The screenshot shows the 'Validation Rule Edit' interface. The 'Error Condition Formula' field contains the value 'NOT(REGEX(Rating_for_service__c , "[1-5]{1}"))', which is highlighted with a red box. To the right, a tooltip for the 'REGEX' function is visible, showing its parameters: ACOS, ADDMONTHS, AND, ASCII, ASIN. Below the formula field is a 'Check Syntax' button. The 'Error Message' field contains the value 'rating should be from 1 to 5', which is also highlighted with a red box. The 'Example' field shows 'Discount percent cannot exceed 30%'. A note at the bottom states 'This error message can either appear at the top of the page or below a specific field on the page'.

Duplicate rule

CREATED a matching rule an Cusmer details Object

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

The screenshot shows the 'Matching Rules' screen in the Setup menu. The top navigation bar includes 'Setup', 'Home', and 'Object Manager'. A search bar on the left contains the text 'matching'. The main area is titled 'Matching Rules' and displays a table of 'All Matching Rules'. The table has columns for 'Action', 'Rule Name', 'Object', 'Status', 'Description', 'Last Modified Date', and 'Last Modified By'. A green arrow points to the 'Matching Rules' tab in the breadcrumb navigation. Another green arrow points to the 'Create New View' button at the bottom of the table header.

- Select the object as Cusmer details and click Next.

Matching Rule
New Matching Rule

Step 1: Select object Step 1 of 2

Select the object to which this matching rule applies.

Object: Customer Details

Next Cancel

4. Give the Rule name : Matching cusmer details
 5. Unique name : is au populated
 6. Define the matching criteria as
 7. Field Matching Method
- | | |
|-----------------|-------|
| 1. Gmail | Exact |
| 2. Phone Number | Exact |
8. Click save.
 9. After Saving Click on Activate.

Save Cancel

Rule Details

Object: Customer Details
Rule Name: matching Customer details
Unique Name: matching_Customer_det
Description:

Matching Criteria

Tell the rule which fields to compare and how.

Field: Gmail	Matching Method: Exact	Match Blank Fields: AND
Field: Phone Number	Matching Method: Exact	Match Blank Fields: AND
Field: -None-	Matching Method: Exact	Match Blank Fields: AND
Field: -None-	Matching Method: Exact	Match Blank Fields: AND
Field: -None-	Matching Method: Exact	Match Blank Fields: AND

Add Filter Logic... 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: project2, 25/09/2023, 10:15 am
Modified By: project2, 10/10/2023, 3:32 pm

Edit Delete Clone Activate

CREATED a Duplicate rule an Cusmer details Object

1. Go quick find box in setup and search for Duplicate rules.
2. Click on Duplicate rule >> click on New Rule >> select cusmer details object.

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 Customer Matching Rule	<input checked="" type="checkbox"/>	o2	10/10/2023
Standard Account Duplicate Rule	Identify contacts that duplicate other contacts and leads	Apartment	Standard Account Matching Rule	<input checked="" type="checkbox"/>	o2	24/09/2023
Standard Contact Duplicate Rule	Identify contacts that duplicate other contacts and leads	Billing details and feedback	Standard Contact Matching Rule	<input checked="" type="checkbox"/>	o2	24/09/2023
Standard Lead Duplicate Rule	Identify leads that duplicate other leads and contacts	Contact	Standard Lead Matching Rule	<input checked="" type="checkbox"/>	o2	24/09/2023
		Customer Details	Standard Contact Matching Rule	<input checked="" type="checkbox"/>	o2	24/09/2023
		Environment	Standard Lead Matching Rule	<input checked="" type="checkbox"/>	o2	24/09/2023
		Individual	Standard Contact Matching Rule	<input checked="" type="checkbox"/>	o2	24/09/2023
		Laptop	Standard Lead Matching Rule	<input checked="" type="checkbox"/>	o2	24/09/2023
		Lead	Standard Contact Matching Rule	<input checked="" type="checkbox"/>	o2	24/09/2023

3. Give the Rule name as : Cusmer Detail duplicate
4. Scroll a little in Matching rule section
5. Select the matching rule : Matching cusmer details
6. And Click on save.
7. After saving the Duplicate Rule, Click on Activate.

Help for this Page ?

Customer Detail duplicate

Duplicate Rule Edit

Save Save & New Cancel

Rule Details

Rule Name: Customer Detail duplicate *

Description:

Object: Customer Details

Record Level Security: Enforce sharing rules Bypass sharing rules

Actions

Action On Create: Allow Alert Report

Action On Edit: Allow Alert Report

Alert Text: Use one of these records?

Help for this Page ?

Matching Rules

Compare Customer Details With: Customer Details *

Matching Rule: matching Customer details *

Matching Criteria: (Customer Details: Email EXACT MatchBlank = FALSE) AND (Customer Details: Phone_Number EXACT MatchBlank = FALSE)

Field Mapping: Mapping Selected

Add Rule Remove Rule

Conditions

Field Operator Value AND

None None None AND

None None None AND

None None None AND

None None None AND

None None None

Add Filter Logic... *

Save Save & New Cancel

Profiles

A profile is a group/collection of settings and permissions that define what a user can do in salesforce. Profile controls “Object permissions, Field permissions, User permissions, Tab settings, App settings, Apex class access, Visualforce page access, Page layouts, Record Types, Login hours & Login IP ranges. You can define profiles

by the user's job function. For example System Administrar, Developer, Sales Representative.

Types of profiles in salesforce

1. Standard profiles:

By default salesforce provides below standard profiles.

- Contract Manager
- Read Only
- Marketing User
- Solutions Manager
- Standard User
- System Administrar.

We cannot deleted standard ones

Each of these standard ones includes a default set of permissions for all of the standard objects available on the platform.

2. Customer Profiles:

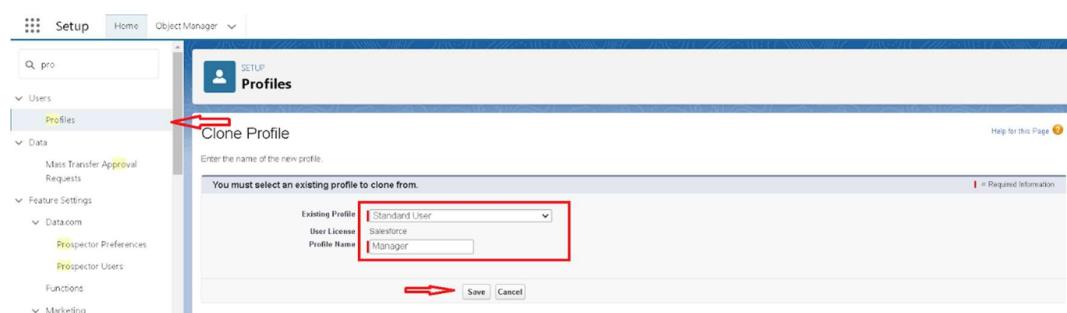
Customer ones defined by us.

They can be deleted if there are no users assigned with that particular one.

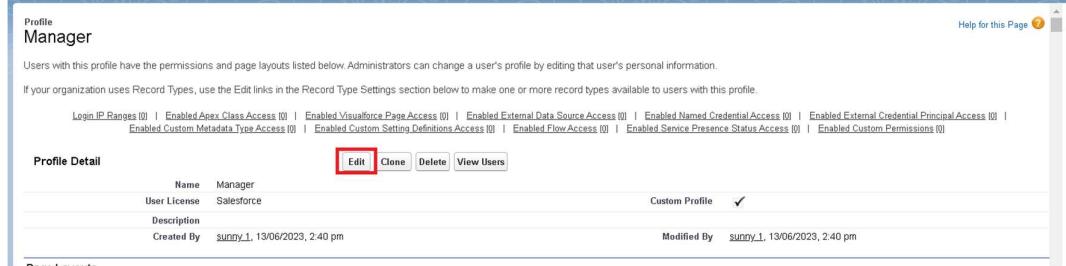
Manager Profile

CREATED a new profile:

1. Go 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.



- Select the Cusm App settings as default for the Garage management.



- Scroll down Cusm Object Permissions and Give access permissions for Appointments,Billing details and feedback , service records and cusmer details objects as mentioned in the below diagram.



- Changing the session times out after should be “ 8 hours of inactivity”.
- Change the password policies as mentioned :
- User passwords expire in should be “ never expires ”.
- Minimum password length should be “ 8 ”, and click save.

sales person Profile

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



5. And click save.

Role & Role Hierarchy

A role in Salesforce defines a user's visibility access at the record level. Roles may be used to specify the types of access that people in your Salesforce organization can have to data. Simply put, it describes what a user could see within the Salesforce organization.

Creating Manager Role

Creating Manager Role:

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

The screenshot shows the Salesforce Setup interface. The top navigation bar includes 'Setup', 'Home', and 'Object Manager'. A search bar with the text 'roles' is highlighted with a red box. Below the search bar, the 'Users' section is expanded, and the 'Roles' link is also highlighted with a red box. To the right, there is a 'SETUP Roles' section with a heading 'Understanding Roles' and a 'Sample Role Hierarchy' diagram. The diagram illustrates a hierarchy from 'Executive Staff' down to 'Western Sales Reps'. A legend explains the icons: a person icon for 'View & edit data, can't update records & generate reports', a person icon with a minus sign for 'Can't access data of other Executive Staff', and a person icon with a plus sign for 'View & edit data, can't update records for all users directly under this level'. At the bottom right of the page, there is a 'Set Up Roles' button and a 'Don't show this page again' checkbox.

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

The screenshot shows the 'Your Organization's Role Hierarchy' page. At the top, there is a 'Collapse All' and 'Expand All' link, with 'Expand All' highlighted with a red box. Below this, the 'Nick Enterprises' node is expanded, showing its child nodes: 'CFO', 'HR', 'Manager', 'On Site Emp', and 'Remote Emp'. Each of these child nodes has an 'Edit | Del | Assign' link and an 'Add Role' link, which is also highlighted with a red box. The 'Manager' node is currently selected.

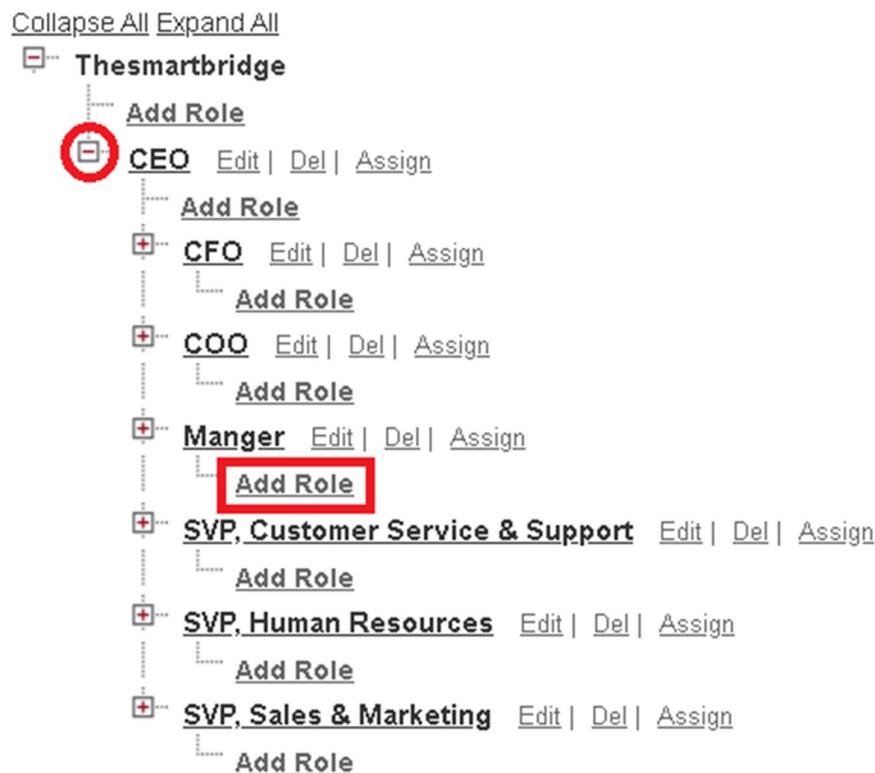
3. Give Label as "Manager" and Role name gets auto populated. Then click on Save.

The screenshot shows the 'Role Edit' dialog box. It contains fields for 'Label' (set to 'Manger'), 'Role Name' (auto-filled with 'Manger'), 'This role reports to' (set to 'CEO'), and 'Role Name as displayed on reports' (empty). At the bottom, there are 'Save', 'Save & New', and 'Cancel' buttons, with 'Save' highlighted with a red arrow.

Creating another roles

Creating another two roles under manager

1. Go 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 populated. Then click on Save.

Users

A user is anyone who logs in Salesforce. Users are employees at your company, such as sales reps, managers, and IT specialists, who need access the company's records. Every user in Salesforce has a user account. The user account identifies the user, and the user account settings determine what features and records the user can access.

creating another users

1. Repeat the steps and CREATED another user using
 1. Role : sales person
 2. User licence : Salesforce Platform

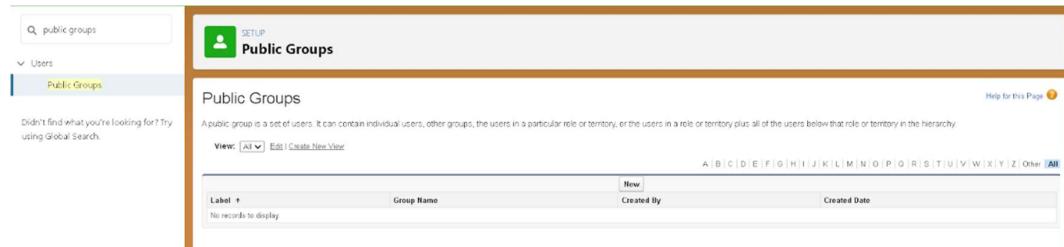
Profile : sales person

Public groups

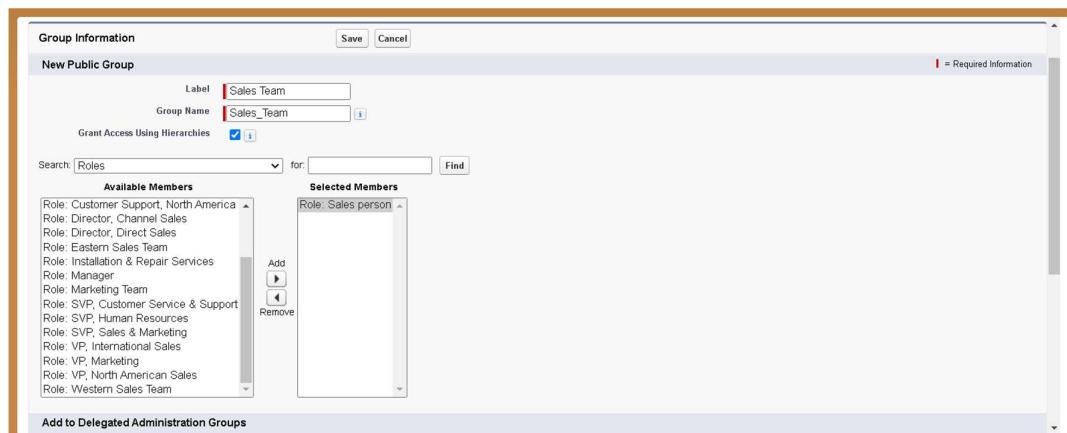
Public groups are a valuable tool for Salesforce administrators and developers to streamline user management, data access, and security settings. By creating and using public groups effectively, you can maintain a secure and organized Salesforce environment while ensuring that users have appropriate access to the resources they need.

Creating New Public Group

1. Go 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 members.
6. Click on save.



Sharing Setting

Salesforce allows you to configure sharing settings to control how records are accessed and shared within your organization. These settings are crucial for maintaining data security and privacy. Salesforce provides a variety of tools and mechanisms to define and enforce sharing rules, such as:

Organization-Wide Default (OWD) Settings:

These settings define the default level of access for all objects within your Salesforce org.

OWD settings include Private, Public Read-Only, Public Read/Write, and Controlled by Parent.

OWD settings can be configured for each standard and custom object.

Role Hierarchy:

Salesforce uses a role hierarchy to determine record access.

Users at higher levels in the hierarchy have greater access to records owned by or shared with users lower in the hierarchy.

The role hierarchy is often used in combination with OWD settings to grant different levels of access.

Profiles and Permission Sets:

Profiles and permission sets allow administrators to specify object-level and field-level permissions for users.

Profiles are typically used to grant general object and field access, while permission sets can be used to extend those permissions to specific users.

Sharing Rules:

Sharing rules are used to extend access to records for users who meet specific criteria.

They can be used to grant read-only or read-write access to records owned by other users.

Manual Sharing:

Administrators and record owners can manually share specific records with other users or groups.

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.

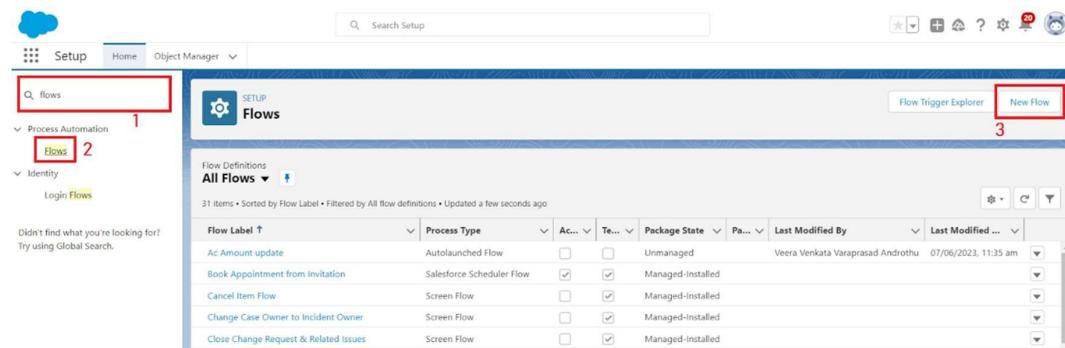
4. Scroll down a bit, Click new on Service records sharing Rules.
5. Give the Label name as “ Sharing setting”
6. Rule name is au populated.
7. In step 3 : Select which records be shared, members of “ Roles ” >> “ Sales person”
8. In step 4: share with, select “ Roles ” >> “ Manager ”
9. In step 5 : Change the access level “ Read / write ”.
10. Click on save.

Flows

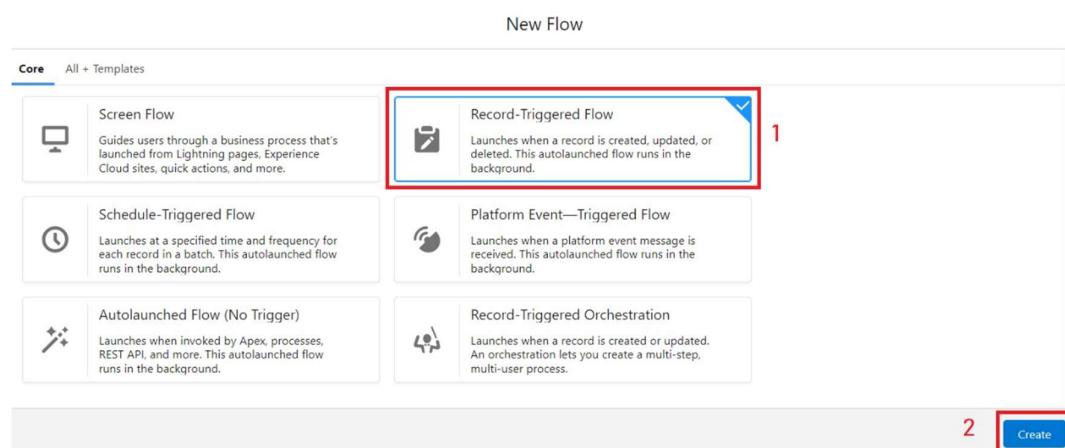
In Salesforce, a flow is a powerful tool that allows you to automate business processes, collect and update data, and guide users through a series of screens or steps. Flows are built using a visual interface and can be created without any coding knowledge.

CREATE a Flow

1. Go 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.

Configure Start

Select Object
Select the object whose records trigger the flow when they're created, updated, or deleted.

*Object

Configure Trigger
Trigger the Flow When:
 A record is created
 A record is updated
 A record is created or updated
 A record is deleted

Set Entry Conditions
Specify entry conditions to reduce the number of records that trigger the flow and the number of times the flow is executed. Minimizing unnecessary flow executions helps to conserve your org's resources.

If you create a flow that's triggered when a record is updated, we recommend first defining entry conditions. Then select the **Only when a record is updated to meet the condition requirements** option for When to Run the Flow for Updated Records.

Condition Requirements

Optimize the Flow for:

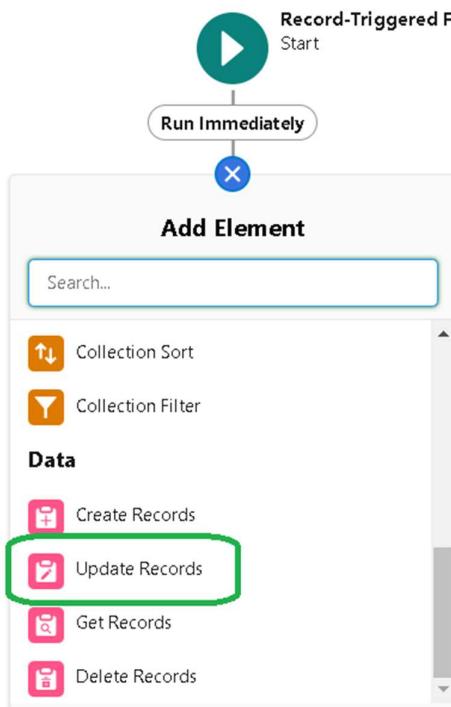
Fast Field Updates
Update fields on the record that triggers the flow to run. This high-performance flow runs *before* the record is saved to the database.

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.

Include a Run Asynchronously path to access an external system after the original transaction for the triggering record is successfully committed

Done 4

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 au populated

Edit Update Records

Update Salesforce records using values from the flow.

*Label	*API Name
Amount Update	Amount_Update

Description

***How to Find Records to Update and Set Their Values**

- Use the billing details and feedback record that triggered the flow
- Update records related to the billing details and feedback record that triggered the flow
- Use the IDs and all field values from a record or record collection
- Specify conditions to identify records, and set fields individually

Set Filter Conditions

Condition Requirements to Update Record

All Conditions Are Met (AND)

Cancel **Done**

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...

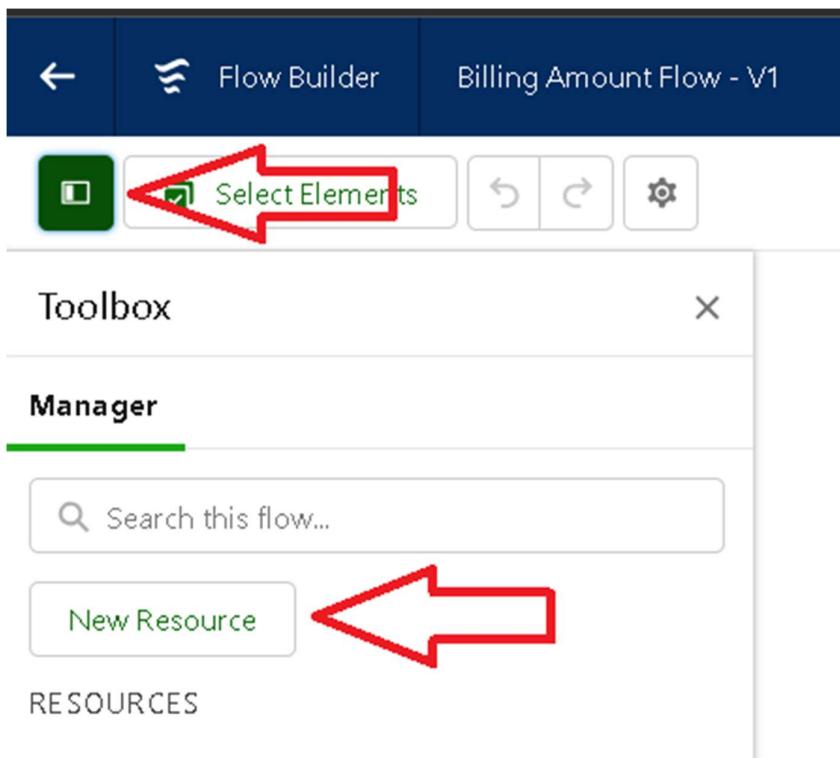
Add Field

Cancel **Done**

9. Set a filter condition : All Conditions are met(AND)
10. Field : Payment_Status__c
11. Operar : 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. CREATED a New Resource form toolbox form p 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 Plain Text.

22. In body field paste the syntax that given below.

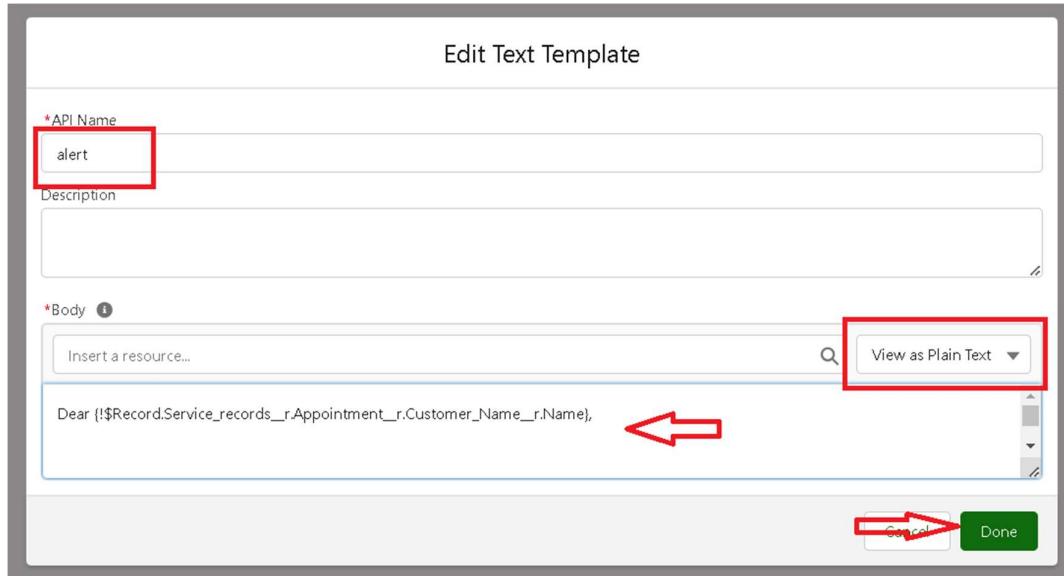
Dear {!\$Record.Service_records__r.Appointment__r.Cusmer_Name__r.Name},

I hope this message finds you well. I wanted take a moment 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 provide p-notch services you and all our valued customers.

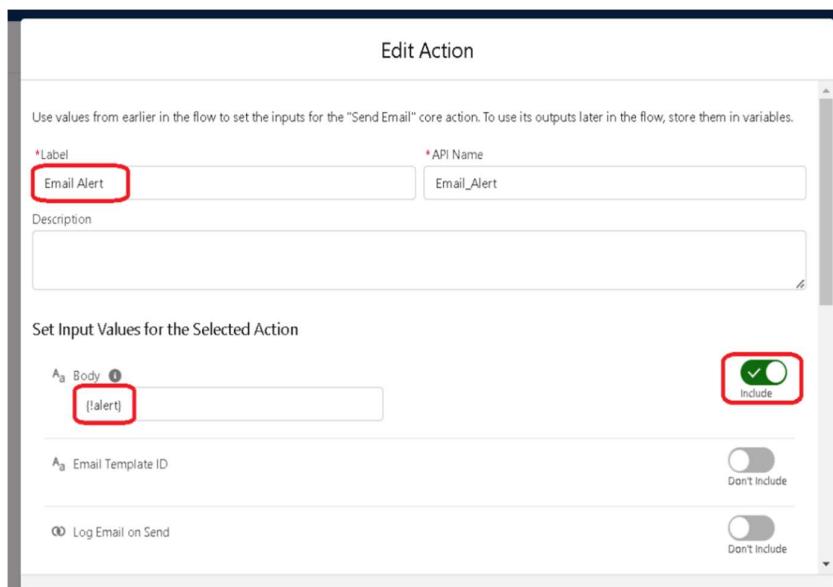
Amount paid : {!\$Record.Payment_Paid__c}

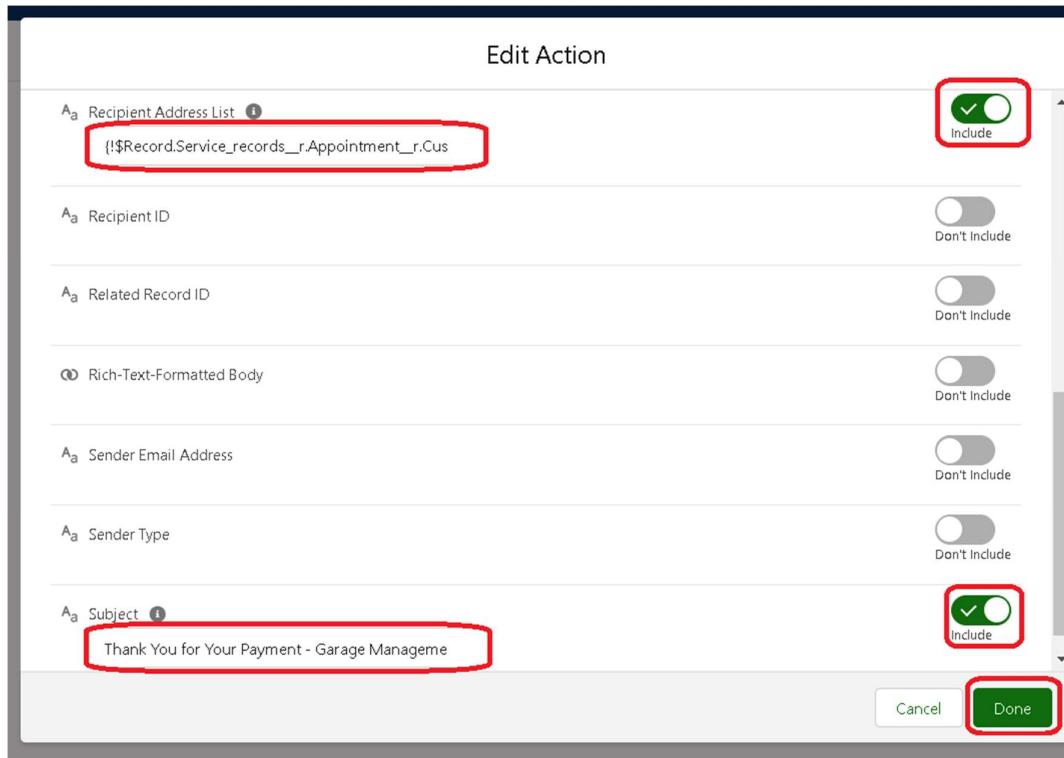
Thank you for Coming .

23. Click done.

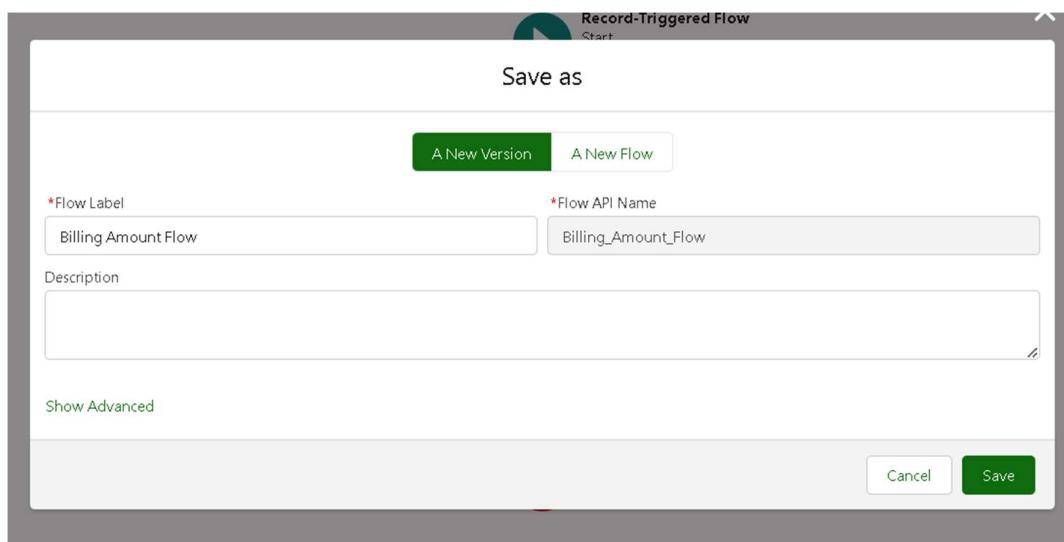


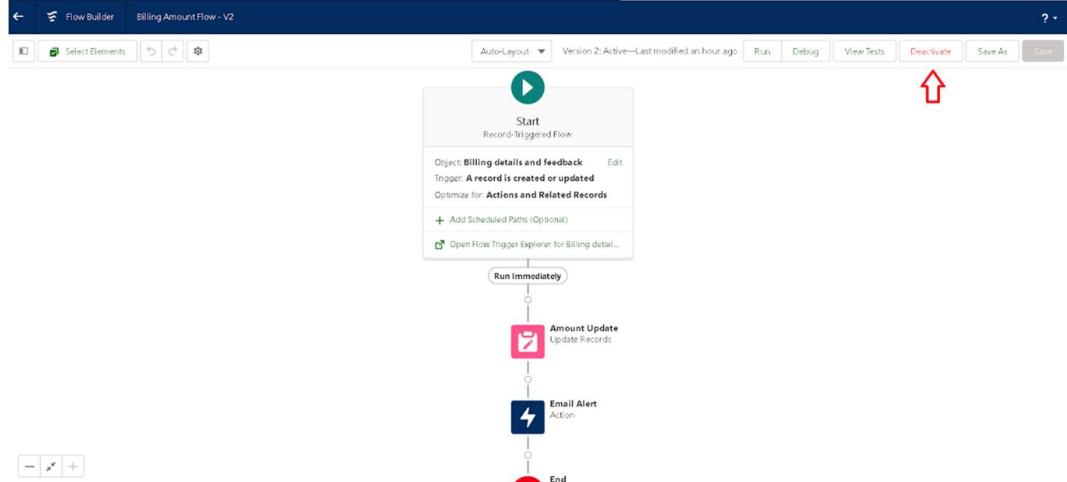
24. Click on Add Element , select Action.
25. Their action bar will be opened in that search for “ send email ” and click on it.
26. Give the label name as “ Email Alert”
27. API name will be au populated.
28. Enable the body in set input values for the selected action.
29. Select the text template that CREATEDd , Body : {!alert}
30. Include recipient address list select the email form the record.
31. RecipientAddressList:
 {!\$Record.Service_records__r.Appointment__r.Cusmer_Name__r.Gmail__c}
32. Include subject as “ Thank You for Your Payment - Garage Management”.
33. Click done.





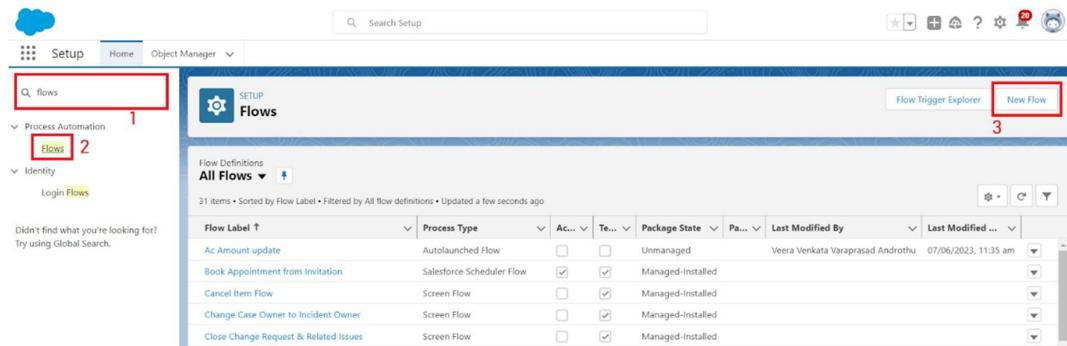
34. Click on save. Give the Flow label , Flow Api name will be autopopulated.
35. And click save, and click on activate.



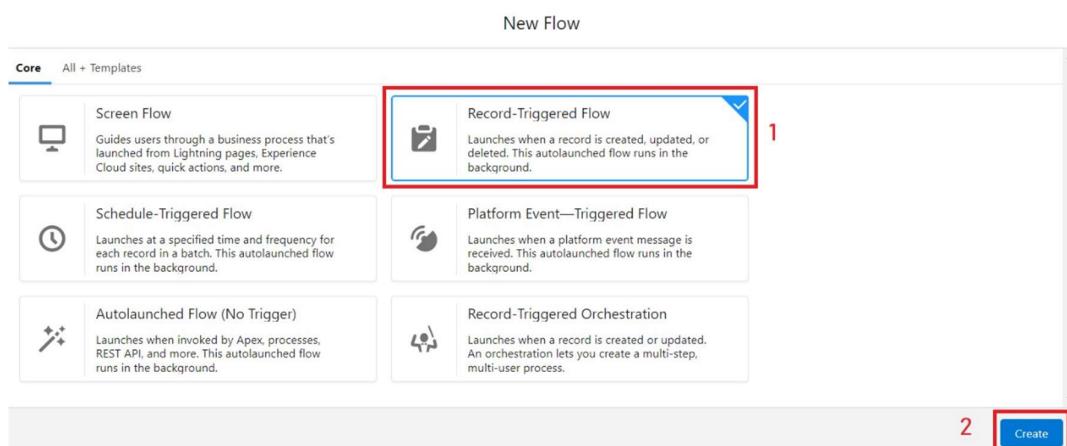


CREATED another Flow

1. Go 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 CREATED.



3. Select the Object as “Service records” in the Drop down list.
4. Select the Trigger Flow when: “A record is CREATEDd 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. Operar : Equals
10. Value : True
11. And Set Field Values for the Billing details and feedback Record
12. Field : Service_Status__c
13. Value : Completed

Set Filter Conditions

Condition Requirements to Update Record

All Conditions Are Met (AND) ▾

Field	Operator	Value
Quality_Check_Status__c	Equals	True X

+ Add Condition

Set Field Values for the Service record Record

Field	Value
Service_Status__c	Completed

+ Add Field

14. Click On Done.
15. Click on save
16. Given the Flow label as Update Service Status , Flow Api name will be au populated.
17. And click save, and click on activate.

Reports

Reports give you access to your Salesforce data. You can examine your Salesforce data in almost infinite combinations, display it in easy-to-understand formats, and share the resulting insights with others. Before building, reading, and sharing reports, review these reporting basics.

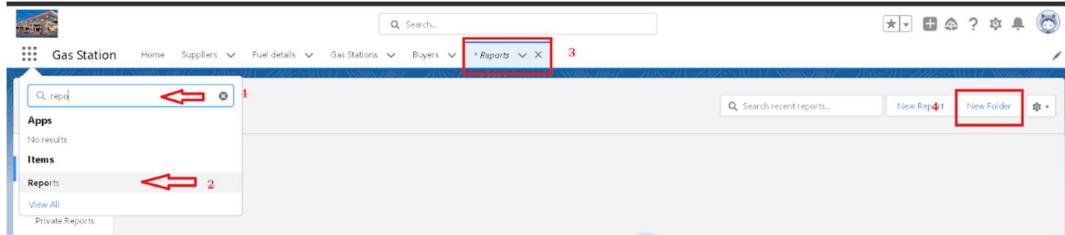
Types of Reports in Salesforce

1. Tabular
2. Summary
3. Matrix
4. Joined Reports

CREATED a report folder

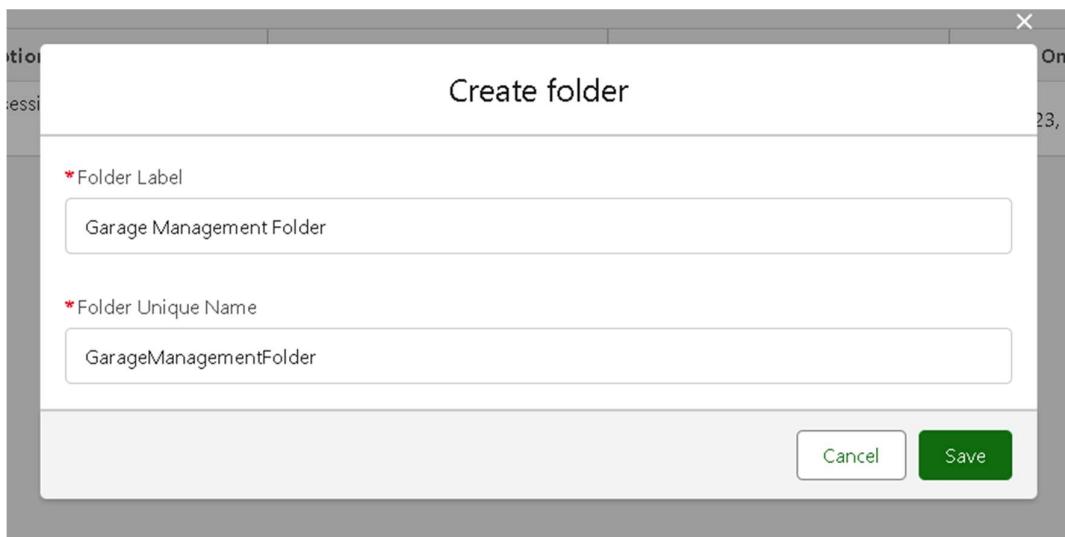
1. Click on the app launcher and search for reports.

2. Click on the report tab, click on new folder.



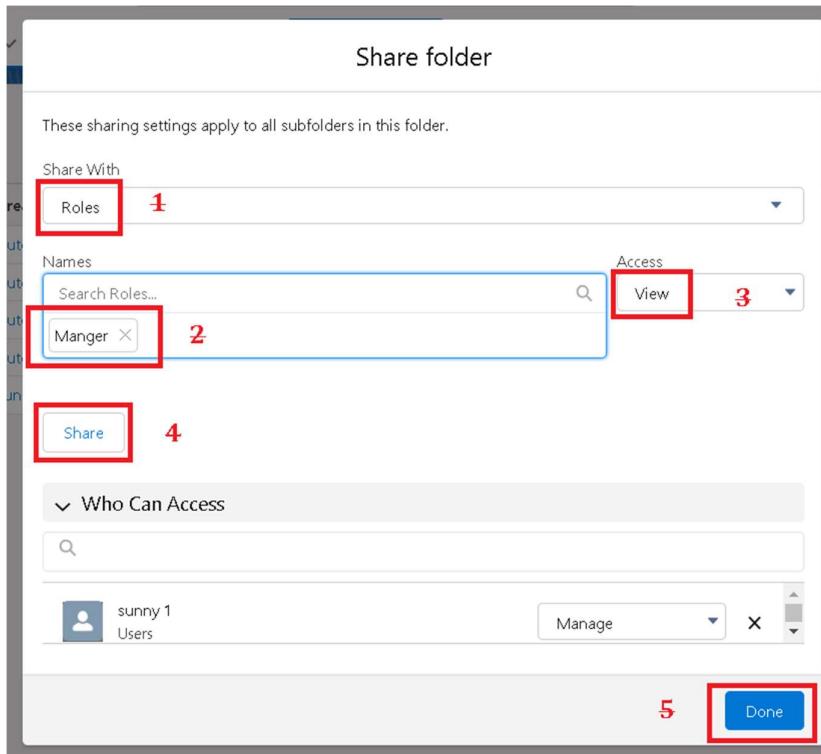
3. Give the Folder label as “Garage Management Folder”, Folder unique name will be auto populated.

4. Click save.



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.



CREATED Report Type

1. Go setup >> type users in quick find box >> select Report Type >> click on Continue.
2. Click on new cstm report type.

Action	Label	Description	Category	Deployed	Created By Alias	Created Date
Edit Del	Bot Metrics Daily Summer '23	Einstein Bot metrics aggregated by day	Other Reports	✓	autopilot	28/09/2023
Edit Del	Bot Metrics Hourly Summer '23	Einstein Bot metrics aggregated by hour	Other Reports	✓	autopilot	28/09/2023
Edit Del	Screen Flows	Find out which flows get executed and how long users take to complete each flow screen	Other Reports	✓	autopilot	24/09/2023
Edit Del	Session Metrics Summer '23	Einstein Bot session metrics	Other Reports	✓	autopilot	28/09/2023

3. Select the Primary object as “ Cusmer details” .
4. Give the Report type Label as “ Service information ”
5. Report type Name is autopopulated.
6. Keep the Description as same.
7. Select Sre in Category as “ other Reports ”
8. Select the deployment status as “ Deployed ”, click on Next.

Report Type Focus

Specify what type of records (rows) will be the focus of reports generated by this report type.

Example: If reporting on "Contacts with Opportunities with Partners," select "Contacts" as the primary object.

Identification

Primary Object: Customer Details

Report Type Label: Service information

Report Type Name: Service_information

Description: Service information

Store in Category: Other Reports

Deployment

A report type with deployed status is available for use in the report wizard. While in development, report types are visible only to authorized administrators and their delegates.

Deployment Status: Deployed

Next >

9. , Click on Related object box.

10. Click on Select Object, choose Appointment Object as shown in fig.

New Custom Report Type
Service information

Step 2. Define Report Records Set

This report type will generate reports about Customer Details. You may define which related records from other objects are returned in report results by choosing a relationship to another object.

A Customer Details
Primary Object

B Select Object

- Select Object -
- Activities
- Appointments
- Duplicate Record Items

Get one related 'B' record.
Get related 'B' records.

A B

Previous Save Cancel

Step 2. Define Report Records Set

This report type will generate reports about Customer Details. You may define which related records from other objects are returned in report results by choosing a relationship to another object.

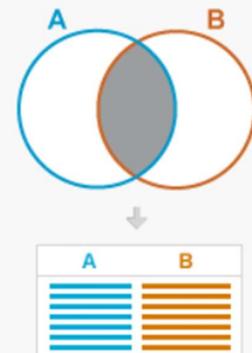
A Customer Details
Primary Object

B Appointments

A to B Relationship:

- Each "A" record must have at least one related "B" record.
- "A" records may or may not have related "B" records.

(Click to relate another object)



11. Again Click 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.

A Customer Details
Primary Object

B Appointments
A to B Relationship:
 Each "A" record must have at least one related "B" record.
 "A" records may or may not have related "B" records.

C Service records
B to C Relationship:
 Each "B" record must have at least one related "C" record.
 "B" records may or may not have related "C" records.

D Billing details and feedback
C to D Relationship:
 Each "C" record must have at least one related "D" record.
 "C" records may or may not have related "D" records.

Object Limit Reached
You can associate up to four objects to a custom report type.

Previous Save Cancel

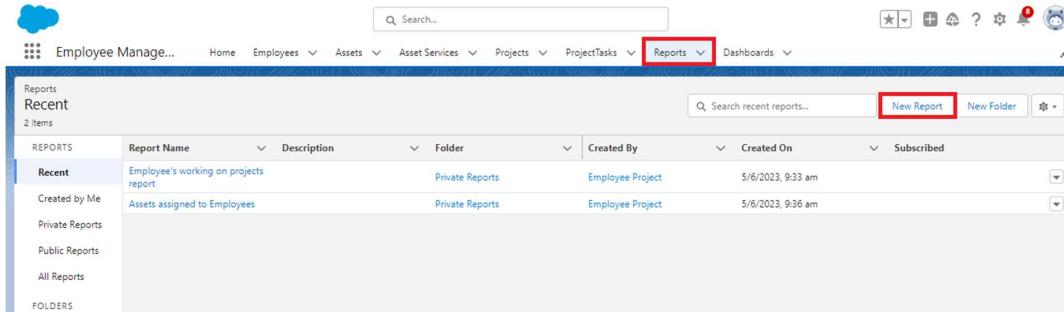
CREATED Report

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

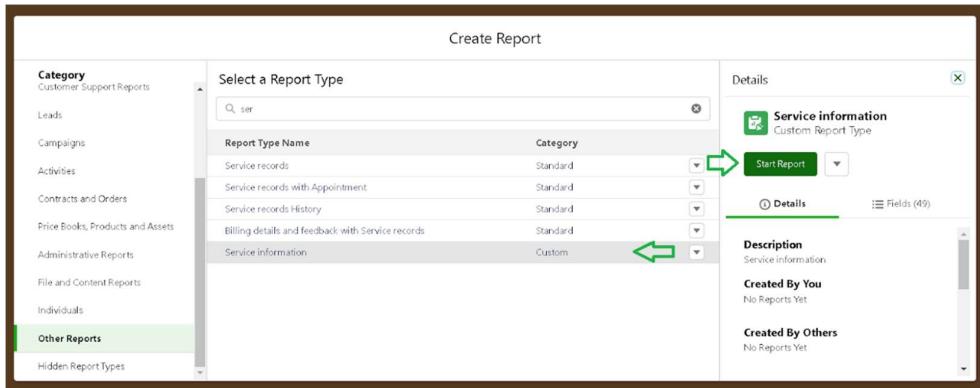
Try fill every field in each record for better experience.

1. Go 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.

1. Cusmer name

2. Appointment Date

3. Service Status

4. Payment paid

5. Remove the unnecessary fields.

6. Select the fields that mentioned below in GROUP ROWS section.

1. Rating for Service

7. Select the fields that mentioned below in GROUP ROWS section.

1. 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 au populated.

11. Select the folder the CREATEDd and Click on save.

The screenshot shows two windows side-by-side. On the left is the 'Service information' report interface. It features a table with columns: Rating for service, Payment Status, Completed, and Total. The table has three rows: a header row, a data row with values Sum of Payment Paid, Record Count, ₹15,000, and 4; another data row with values Sum of Payment Paid, Record Count, ₹5,000, and 2; and a total row with values Sum of Payment Paid, Record Count, ₹20,000, and 6. To the right of the table is a line chart titled 'Sum of Payments' vs 'Rating for service'. The chart shows a single blue line starting at approximately (4, 15) and ending at approximately (5, 10). On the right is the 'Save Report' dialog box. It contains fields for 'Report Name' (New Service information Report), 'Report Unique Name' (New_Service_information_Report_oVu), 'Report Description' (empty), and 'Folder' (Garage Management Folder). A green arrow points to the 'Report Name' field, and another green arrow points to the 'Folder' field. At the bottom right of the dialog box are 'Cancel' and 'Save' buttons.

Dashboards

Dashboards help you visually understand changing business conditions so you can make decisions based on the real-time data you've gathered with reports. Use dashboards help users identify trends, sort out quantities, and measure the impact of their activities. Before building, reading, and sharing dashboards, review these dashboard basics.

CREATED Dashboard

1. Go the app >> click on the Dashboards tabs.

2. Give a Name and select the folder that CREATEDd, and click on CREATED.

New Dashboard

* Name
Customer review

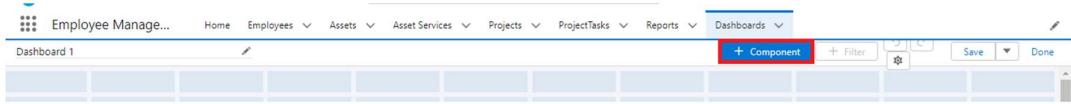
Description

Folder
Service Rating

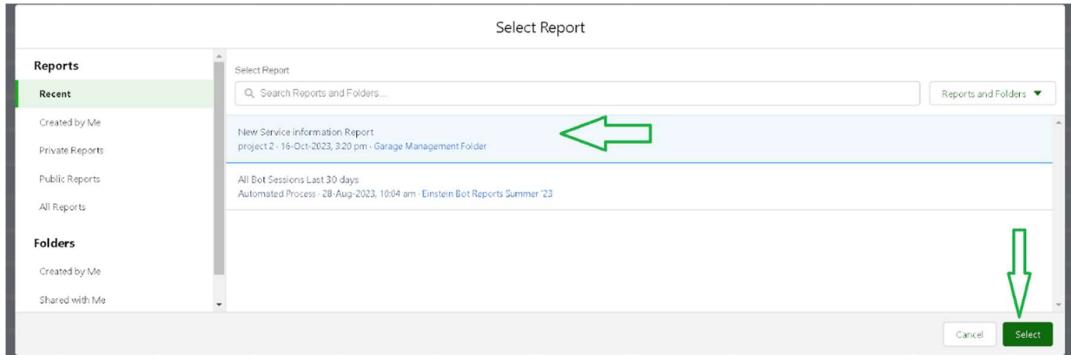
Select Folder

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.

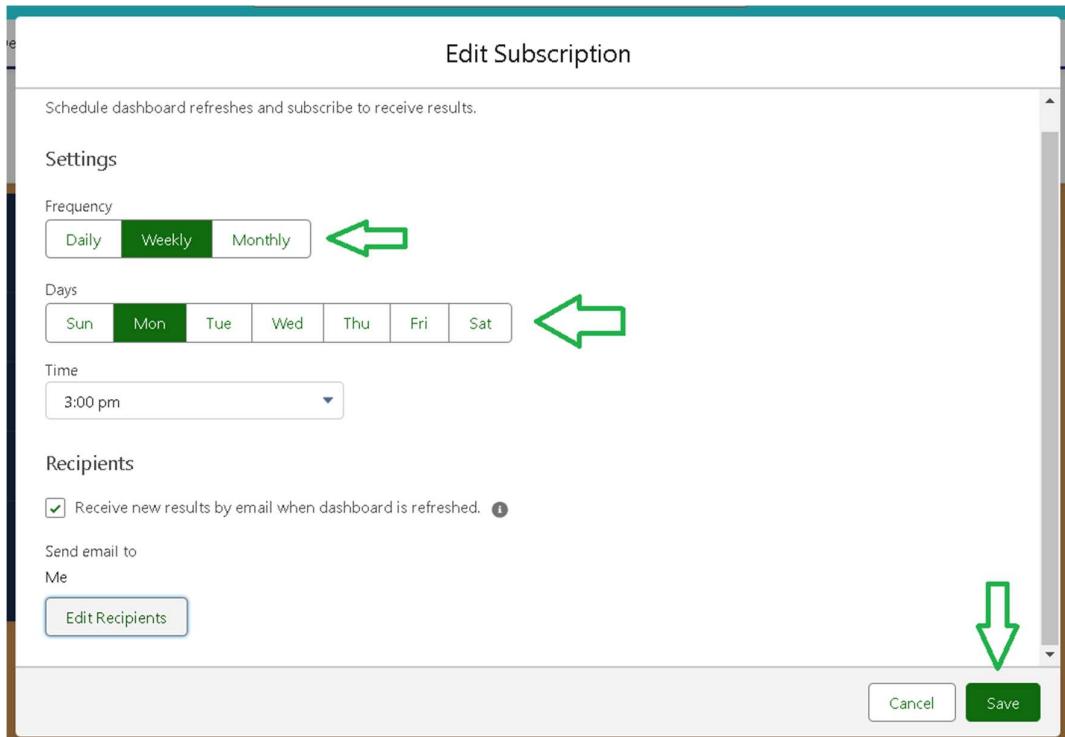
7. Preview is shown below.

8. After that Click on Subscribe on p right.

9. Set the Frequency as " weekly ".

10. Set a day as monday.

11. And Click on save.



User Adoption

creating records

CREATED a record in the following objects following 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.

CREATED the Appointment Record

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

CREATED a service Record

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

3. Click on save.
4. Open the record and click on Quality check status as true.
5. Click on save.
6. Automatically Service status will be moved completed.

Testing and Validation:

- Unit Testing (Apex Classes, Triggers):

Apex Trigger

Apex can be invoked by using triggers. Apex triggers enable you perform custom actions before or after changes Salesforce records, such as insertions, updates, or deletions. A trigger is Apex code that executes before or after the following types of operations:

- insert
- update
- delete
- merge
- upsert
- undelete

For example, you can have a trigger run before an object's records are inserted in the database, after records have been deleted, or even after a record is restored from the Recycle Bin.

You can define triggers for primary-level standard objects that support triggers, such as a Contact or an Account, some standard child objects, such as a CaseComment, and custom objects. To define a trigger, from the object management settings for the object whose triggers you want access, go Triggers.

There are primarily two types of Apex Triggers:

Before Trigger: This type of trigger in Salesforce is used either to update or validate the values of a record before they can be saved in the database. So, basically, the before trigger validates the record first and then saves it. Some criteria or code can be set to check data before it gets ready to be inserted in the database.

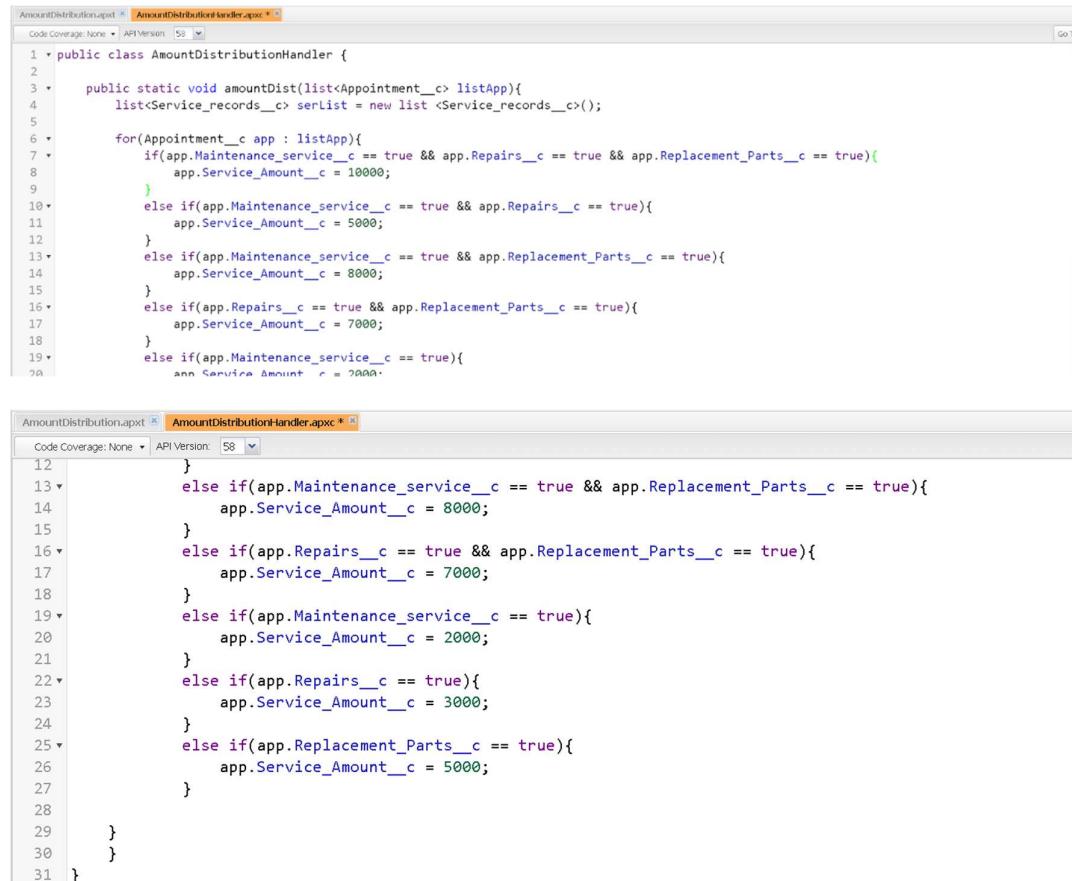
After Trigger: This type of trigger in Salesforce is used to access the field values set by the system and affect any change in the record. In other words, the after trigger makes changes to the value from the data inserted in some other record.

Apex handler

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

Login the respective trailhead account and navigate the gear icon in the p right corner.

1. Click on the Developer console. you will see a new console window.
2. In the olbar, you can see FILE. Click on it and navigate new and CREATED New apex class.
3. Name the class as “AmountDistributionHandler”.



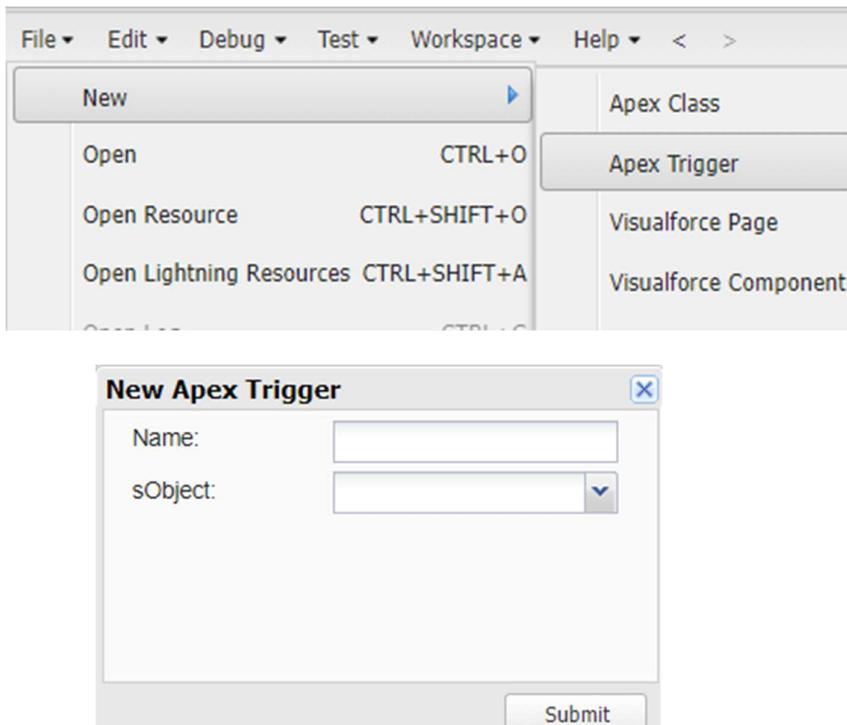
```
AmountDistribution.apxc | AmountDistributionHandler.apxc * [x]
Code Coverage: None | API Version: 58 | Go To
1 * public class AmountDistributionHandler {
2
3     public static void amountDist(List<Appointment__c> listApp){
4         List<Service__c> serList = new List<Service__c>();
5
6         for(Appointment__c 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.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 }
```

Trigger Handler :

How CREATED a new trigger :

1. While still in the trailhead account, navigate the gear icon in the p right corner.
2. Click on developer console and you will be navigated a new console window.
3. Click on File menu in the ol bar, and click on new? Trigger.
4. Enter the trigger name and the object be triggered.

5. Name : AmountDistribution
6. sObject : Appointment__c



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

```

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

```

The screenshot shows the code editor with the file 'AmountDistribution.apxc' open. The code defines a trigger named 'AmountDistribution' on the 'Appointment__c' object. The trigger is set to run before insert or update. Inside the trigger, there is a condition that checks if it's before and either an insert or update. If true, it calls the 'amountDist' method from the 'AmountDistributionHandler' class on the 'trigger.new' records.

Key Scenarios Addressed by Salesforce in the Implementation of Project:

1. Customer Management and Engagement

- Streamlined customer onboarding with custom Salesforce objects and workflows.
- Centralized storage of customer information, including contact details, vehicle history, and preferences.
- Enhanced customer relationship management with automated reminders for vehicle servicing, follow-ups, and personalized offers.

2. Service Scheduling and Appointment Management

- Implementation of flows to automate appointment scheduling and rescheduling.
- Real-time availability updates for technicians and resources.
- Integration of calendar views for easy visualization of service appointments.

3. Inventory and Parts Management

- Tracking of automotive parts and inventory levels using custom Salesforce fields and relationships.
- Automated alerts for low-stock items and reordering workflows to maintain stock levels.
- Reports and dashboards for inventory utilization trends and forecasting.

4. Operational Workflow Automation

- Use of Salesforce triggers to automate tasks such as invoice generation and status updates.
- Automated assignment of service requests to technicians based on skill set and availability.
- Optimization of repair workflows through approval processes and guided workflows.

5. Performance Tracking and Reporting

- Creation of dashboards and reports for monitoring key performance indicators (KPIs), such as service turnaround time, technician productivity, and customer satisfaction.
- Data-driven insights for decision-making, including trend analysis and predictive analytics for business growth.

6. Customer Feedback and Satisfaction

- Automated feedback collection after service completion using Salesforce forms and surveys.
- Analysis of feedback through dashboards to identify areas for improvement.
- Personalized follow-ups to address customer concerns and enhance satisfaction.

7. Billing and Payment Processing

- Seamless integration of billing workflows with custom objects and fields for tracking service charges and payments.
- Automated generation of invoices and receipts.
- Integration with payment gateways for a smooth and secure payment experience.

8. Compliance and Documentation

- Storage and management of compliance documents for the repair facility and customer vehicles.
- Automated reminders for vehicle inspections, insurance renewals, and other regulatory requirements.

9. Scalability and Customization

- Flexibility to adapt the system to varying business sizes and requirements.
- Easy addition of new service categories, locations, and users without disrupting existing processes.
- Scalable infrastructure to support future growth and additional functionalities.

Conclusion:

The implementation of the Garage Management System (GMS) using Salesforce has successfully addressed key challenges in the automotive repair industry, delivering measurable improvements in both operational efficiency and customer satisfaction. By leveraging Salesforce's robust capabilities, the project achieved:

- **Streamlined Workflows:** Automation of service scheduling, inventory tracking, and billing processes, reducing manual effort and operational delays.
- **Enhanced Customer Experience:** Improved engagement through personalized communication, automated reminders, and efficient service delivery.
- **Data-Driven Decision-Making:** Comprehensive dashboards and reports enabled real-time monitoring of performance and informed decision-making.
- **Scalability and Adaptability:** A flexible system design that can grow with business needs, ensuring long-term sustainability and success.
- **Regulatory Compliance:** Simplified document management and timely reminders for compliance requirements.