

GARAGE MANAGEMENT SYSTEM

College Name : KPR COLLEGE OF ARTS SCIENCE AND RESEARCH

College Code : bruaz

TEAM ID : NM2025TMID21567

TEAM MEMBERS :

Team Leader Name: Pon Nisandhini S

Email : 23bda043@kprcas.ac.in

Team Member 1 : Preethi S

Email: 23bda048@kprcas.ac.in

Team Member 2 : Ranjith

Email: 23bda050@kprcas.ac.in

Team Member 3 : Rithika B

Email: 23bda051@kprcas.ac.in

Title: Garage Management System

Project Overview: The Garage Management System (GMS) is a comprehensive solution designed to optimize operations in automotive repair facilities. Built on Salesforce CRM, the system manages customer details, service requests, vehicle history, and billing processes through custom objects, flows, and automation. It streamlines workflows for mechanics, service advisors, and administrators, ensuring timely updates and transparent communication. With real-time dashboards and reports, garages can monitor performance, enhance customer satisfaction, and improve resource allocation.

Objectives:

- **Increase Service Efficiency:** To guarantee faster turnaround times, simplify repair operations, job scheduling, and service requests.
- **Preserve Transparency:** To increase responsibility and confidence, keep thorough records of clients, cars, and transactions.
- **Improve Team Coordination:** Automate reminders, task distribution, and mechanic and service advisor contact.
- **Optimise Resource Utilisation:** To save money and prevent shortages, keep an eye on labour, tools, and spare parts allocation.
- **Turn on Real-Time Tracking:** Provide employees and clients with real-time updates on the status of auto repairs, billing developments, and service completion.
- **Improve Customer Relationships:** Provide a seamless, dependable, and expert experience that promotes repeat business and loyalty.

Student Outcomes:

- **Practical Experience in Garage Process Automation:** Students acquire hands-on skills in creating Salesforce objects, automating workflows, and managing vehicle service records.
- **Understanding of End-to-End Project Implementation:** Students learn the complete lifecycle of building a CRM-based solution, from requirement gathering to testing and deployment.
- **Enhanced Problem-Solving and Analytical Skills:** Students strengthen their ability to analyze operational challenges in garages and design efficient technical solutions.
- **Improved Teamwork and Communication:** Students gain experience collaborating in groups, dividing responsibilities, and coordinating effectively across different project stages.
- **Industry-Relevant Learning:** Students are exposed to real-world applications of Salesforce CRM in the automotive service industry, boosting their career readiness.

System Requirements:

Hardware Requirements:

- Computer with min/sum 4 GB RAM, Dual-core processor
- Stable internet connection

Software Requirements:

- Salesforce Developer Edition Org
- Modern Web Browser (e.g., Google Chrome, Firefox)

Phase 1: Requirement Analysis & Planning

Garage Management System

Phase 2: Salesforce Development – Backend & Configurations

Milestone 1: Salesforce developer account creation

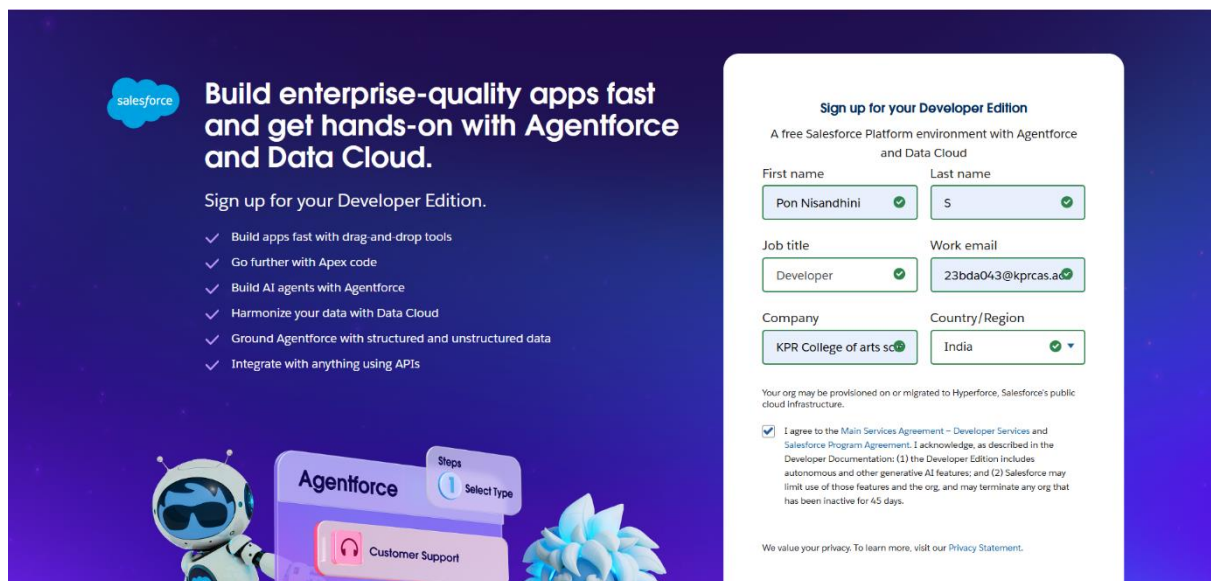
Activity 1: Creating Developer Account

Creating a developer org in salesforce.

1. Go to <https://developer.salesforce.com/signup>
2. On the sign up form, enter the following details:
 - First name & Last name:
 - Email:
 - Role: Developer
 - Company: Your college name
 - County: India
 - Postal Code : pin code
 - 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.



Build enterprise-quality apps fast and get hands-on with Agentforce and Data Cloud.

Sign up for your Developer Edition.

- ✓ Build apps fast with drag-and-drop tools
- ✓ Go further with Apex code
- ✓ Build AI agents with Agentforce
- ✓ Harmonize your data with Data Cloud
- ✓ Ground Agentforce with structured and unstructured data
- ✓ Integrate with anything using APIs

Sign up for your Developer Edition
A free Salesforce Platform environment with Agentforce and Data Cloud

First name: Pon Nisandhini ✓ Last name: S ✓

Job title: Developer ✓ Work email: 23bda043@kprcas.ac ✓

Company: KPR College of arts sc ✓ Country/Region: India ✓

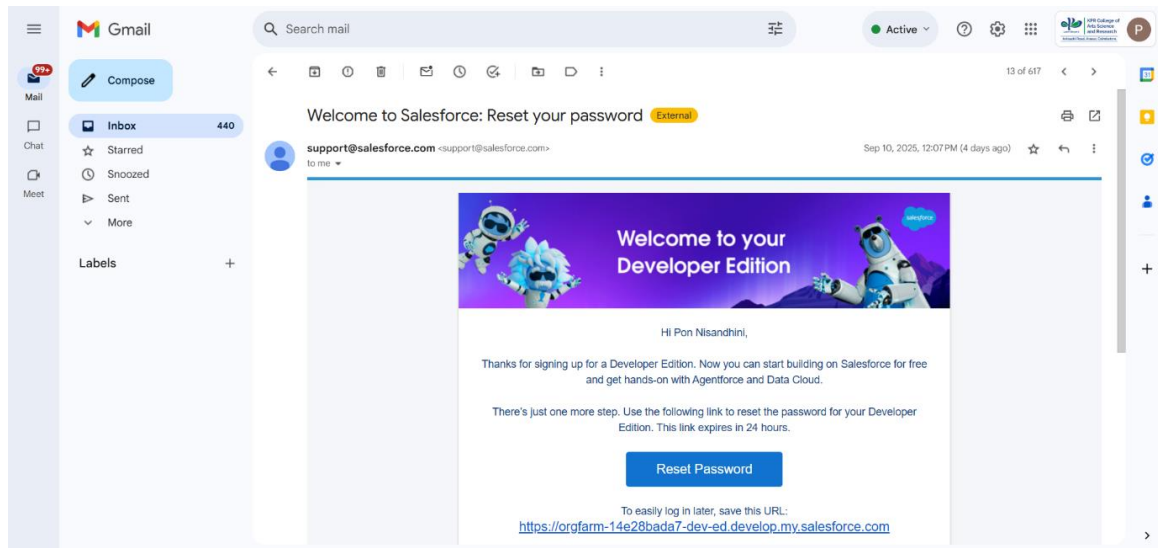
Your org may be provisioned on or migrated to Hyperforce, Salesforce's public cloud infrastructure.

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

We value your privacy. To learn more, visit our Privacy Statement.

Activity 2: 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 5-10 minutes.

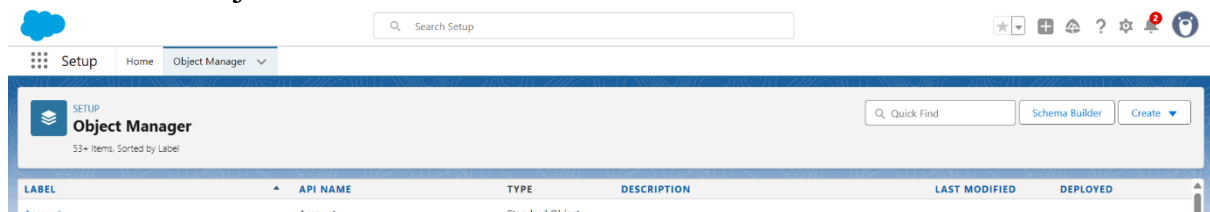


2. Click on Verify Account
3. Give a password and answer a security question and click on Change Password.
4. Then you will redirect to your salesforce setup page.

Milestone 2: OBJECT

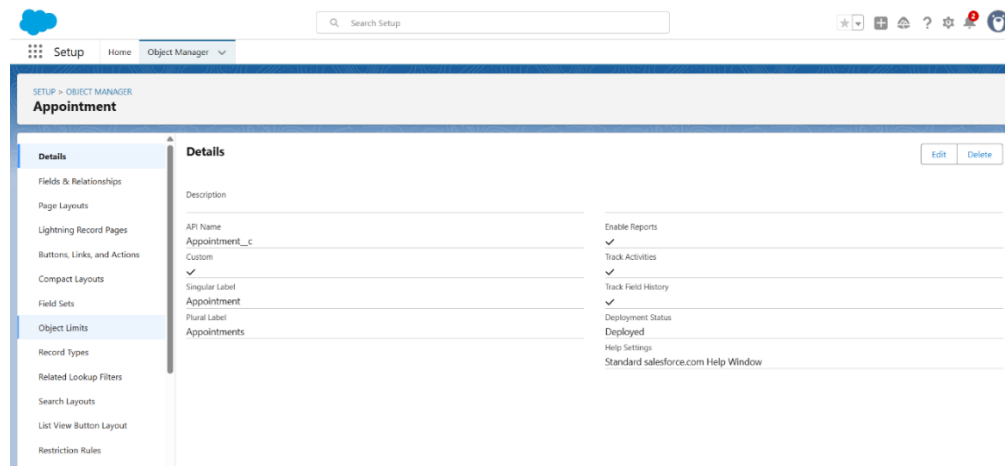
Activity 1: Create Customer Details Object

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
 - Record Name >> Customer Name
 - Data Type >> Text

5. Click on Allow reports and Track Field History,
6. Allow search >> Save.



Activity 2: Create Appointment 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 >> Appointment
3. Plural label name >> Appointments
4. Enter Record Name Label and Format
 - Record Name >> Appointment Name
 - Data Type >> Auto Number
 - Display Format >> app-{000}
 - Starting number >> 1
5. Click on Allow reports and Track Field History,
6. Allow search >> Save.

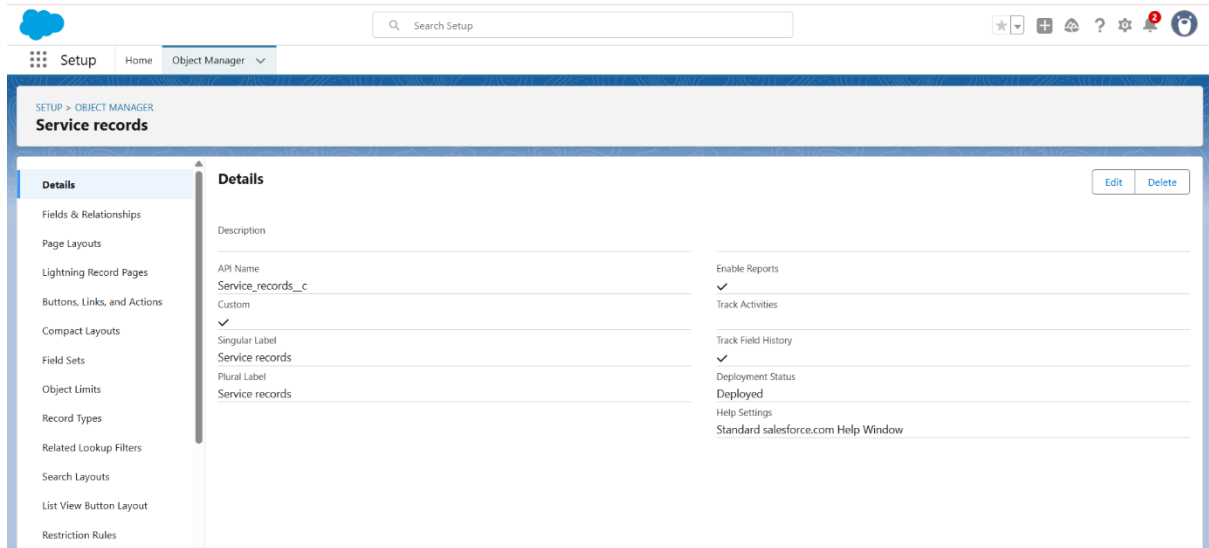
Activity 3: Create Service records Object

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 >> Service records
3. Plural label name >> Service records
4. Enter Record Name Label and Format

- Record Name >>Service records Name
- Data Type >> Auto Number
- Display Format >> ser- {000}
- Starting number >> 1

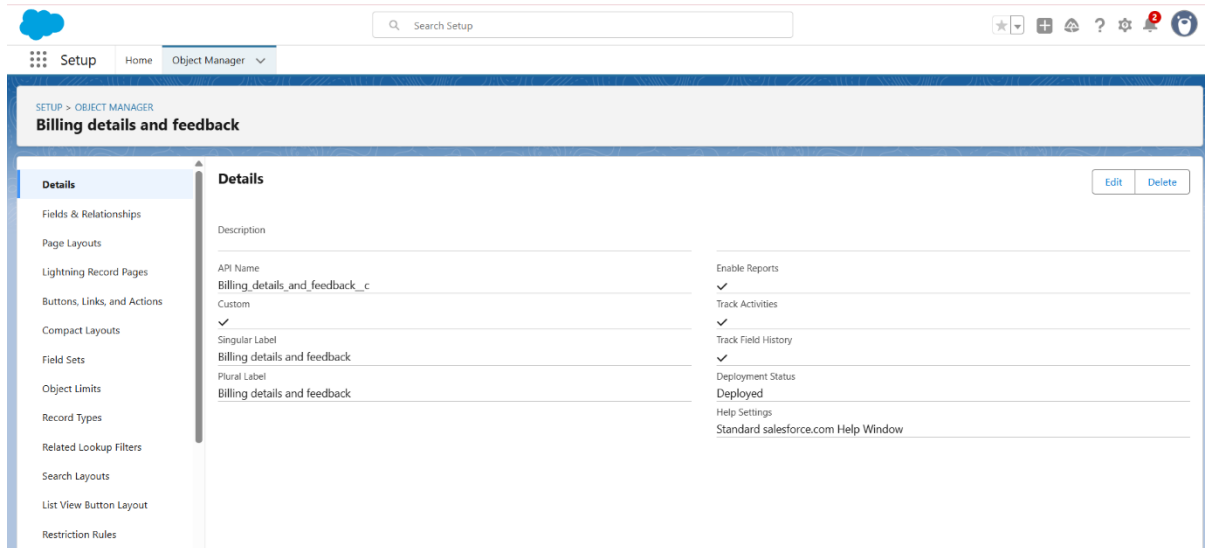
5. Click on Allow reports and Track Field History, Allow search >> Save.



Activity 4: Create Billing details and feedback Object

To create an object:

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



Milestone 3: TABS :

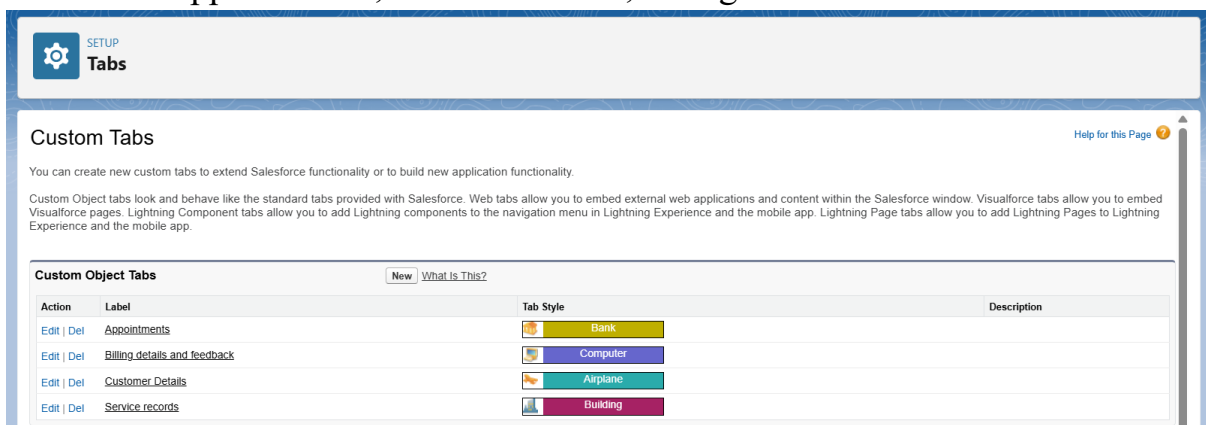
Activity 1: Creating a Custom Tab

To create a Tab:(Customer Details)

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

Activity 2: Creating Remaining Tabs

1. Now create the Tabs for the remaining Objects, they are “Appointments, Service records,Billing details and feedback”.



2. Follow the same steps as mentioned in Activity -1.

Phase 3: UI/UX Development & Customization:

Milestone 4: THE LIGHTNING APP

Activity 1: Create 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.

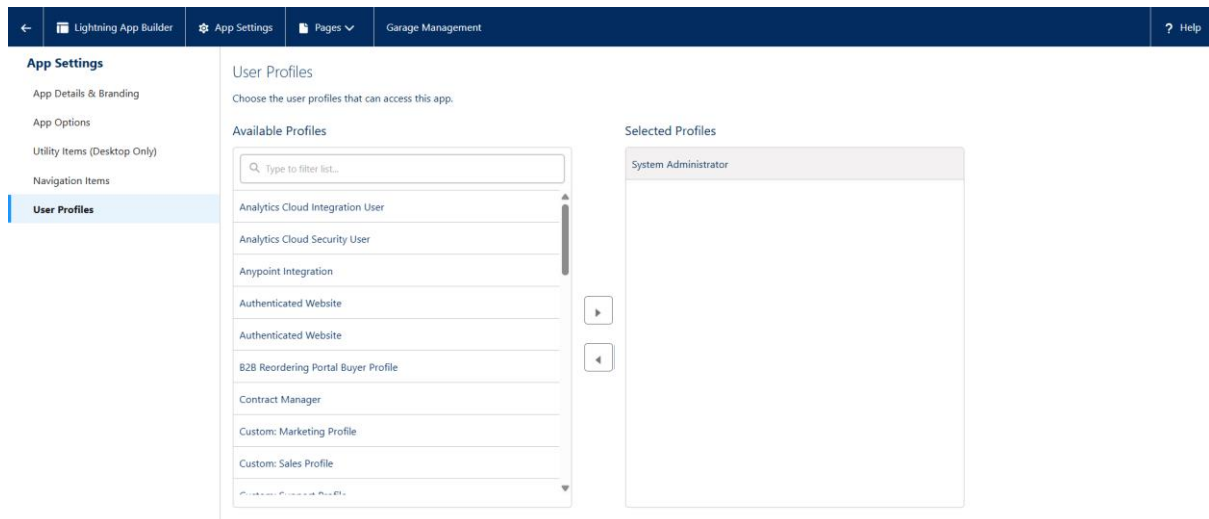
The screenshot shows the 'App Details & Branding' configuration page in the Lightning App Builder. The left sidebar lists 'App Settings' with sub-items: 'App Details & Branding' (selected), 'App Options', 'Utility Items (Desktop Only)', 'Navigation Items', and 'User Profiles'. The main content area is divided into 'App Details' and 'App Branding'. Under 'App Details', there are fields for 'App Name' (Garage Management), 'Developer Name' (Garage_Management_Application), and 'Description' (placeholder: Enter a description...). Under 'App Branding', there is an 'Image' upload button, a 'Primary Color Hex Value' field (set to #0070D2), and 'Org Theme Options' with a checkbox 'Use the app's image and color instead of the org's custom theme'. At the bottom, an 'App Launcher Preview' shows a blue square icon with 'GM' and the text 'Garage Management'.

3. To Add Navigation Items:

The screenshot shows the 'Navigation Items' configuration page in the Lightning App Builder. The left sidebar is the same as the previous screenshot, with 'Navigation Items' selected. The main content area is titled 'Navigation Items' and includes a sub-header: 'Choose the items to include in the app, and arrange the order in which they appear. Users can personalize the navigation to add or move items, but users can't remove or rename the items that you add. Some navigation items are available only for phone or only for desktop. These items are dropped from the navigation bar when the app is viewed in a format that the item doesn't support.' Below this, there are two panels: 'Available Items' and 'Selected Items'. The 'Available Items' panel has a search bar and a list of items including Accounts, Activation Targets, Activations, All Sites, Alternative Payment Methods, Analytics, App Launcher, Appointment Categories, Appointment Invitations, and Approval Requests. The 'Selected Items' panel shows a list of items: Customer Details, Appointments, Service records, Billing details and feedback, Reports, and Dashboards. Arrows between the panels allow for moving items back and forth.

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.

Milestone 5: FIELDS:

Activity 1: Creation of Relationship Fields in Object:

Creation of 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.
2. Now 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: Follow the above steps for the remaining field for the same object.

2. To create another fields in an object:

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

2. Now click on “Fields & Relationships” >> New
3. Select Data type as a “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.

Activity 2: Creation of Lookup Fields

Creation of Lookup Field on Appointment Object:

1. Go to setup >> click on Object Manager >> type object name (Appointment in the search bar >> click on the object.
2. Now click on “Fields & Relationships” >> New
3. Select “Look-up relationship” as data type and click Next.
4. Select the related object “Customer 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 to setup >> click on Object Manager >> type object name (Service records) in search bar >> click on the object.
2. Now 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.
6. Scroll down for Lookup Filter and click on Show filter settings.
7. Now add the filter criteria.
8. Field: Appointment: Appointment Date >> Operator: less than >> select field >> Appointment: Created Date
9. Filter type should be Required.
10. 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 to setup >> click on Object Manager >> type object name (Billing details and feedback) in search bar >> click on the object.
2. Now 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.

Activity 3: Creation of Checkbox Fields

Creation of Checkbox Field on Appointment Object:

1. Go to setup >> click on Object Manager >> type object name (Appointment) in search bar >> click on the object.
2. Now 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 auto populated
6. Default value: unchecked
7. Click on next >> next >> save.

Creation of Another Checkbox Field on Appointment Object:

1. Repeat the steps from 1 to 3.
2. Give the Field Label: Repairs
3. Field Name: is auto populated
4. Default value: unchecked
5. Click on next >> next >> save.
6. Follow the same and create another checkbox with given names
7. Give the Field Label: Replacement Parts
8. Field Name: is auto populated
9. Default value: unchecked
10. Click on next >> next >> save.

Creation of Checkbox Field on Service records Object:

1. Go to setup >> click on Object Manager >> type object name (Service records) in search bar >> click on the object.
2. Now 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 Name: is auto populated
6. Default value: unchecked
7. Click on next >> next >> save

Activity 4: Creation of Date Fields

Creation of Date Field on Appointment Object:

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

Activity 5: Creation of Currency Fields

Creation of Currency Field on Appointment Object:

1. Go to setup >> click on Object Manager >> type object name (Appointment) in the search bar >> click on the object.
2. Now 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 auto populated
6. Click on next
7. Give read only for all the profiles in field level security for profile.
8. Click on next > > save.

Creation of Currency Field on Billing details and feedback Object:

1. Follow 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 auto populated

Activity 6: Creation of Text Fields

1. Go to setup >> click on Object Manager >> type object name (Appointment) in the search bar >> click on the object.
2. Now 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 auto populated
6. Length: 10
7. Make field as Required and Unique.
8. Click on next >> next >> save.

Creation of Text Fields in Billing details and feedback object:

1. Go to setup >> click on Object Manager >> type object name (Billing details and feedback) in search bar >> click on the object.
2. Now 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 auto populated
6. Length : 1
7. Make field as Required.
8. Click on next >> next >> save

Activity 7: Creation of Picklist Fields

Creation of Picklist Fields in Service records object :

1. Go to 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.
6. Click Next.
7. Next >> Next >> Save.

Creation of Picklist Fields in Billing details and feedback object :

1. Go to 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.

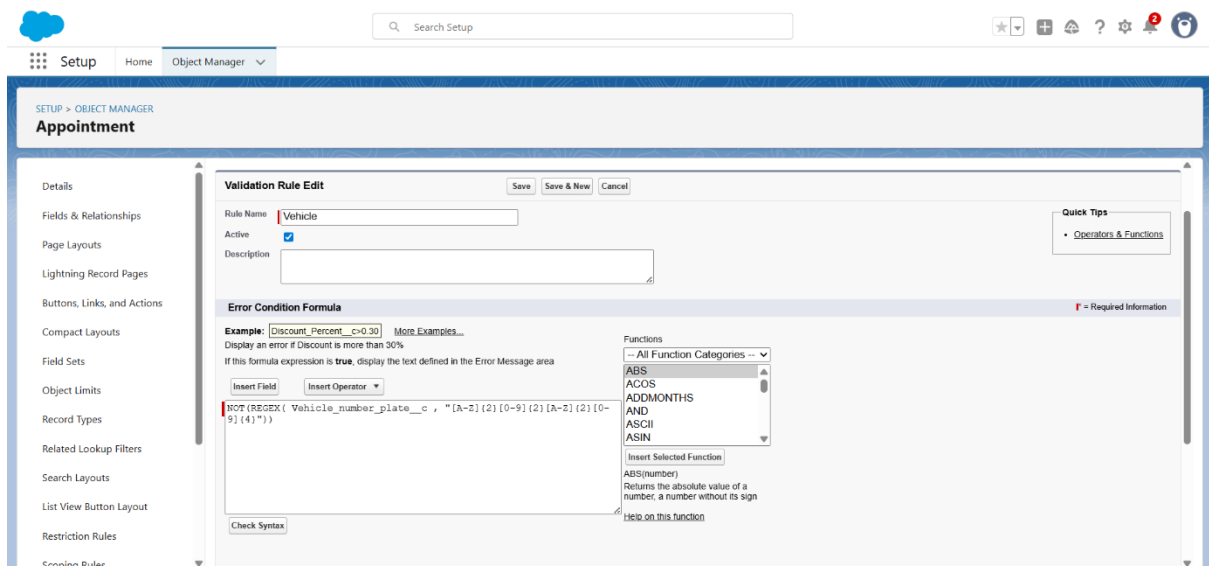
Activity 8: Creating Formula Field in Service records Object

1. Go to 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.
5. Insert field formula should be : CreatedDate
6. Click “Check Syntax” .
7. Click next >> next >> Save.

Milestone 6: VALIDATION RULE

Activity 1: To create a validation rule to an Appointment Object

1. Go to the setup page >> click on object manager >> From drop down click edit for Appointment object.
2. Click on the validation rule >> click New.
3. Enter the Rule name as “ Vehicle ”.
4. Insert the Error Condition Formula as : -
`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 vaild number ”, select the Error location as Field and select the field as “Vehicle number plate”, and click Save.



Activity 2: To create a validation rule to an Billing details and feedback Object

1. Go to the setup page >> click on object manager >> From drop down click edit for Billing details and feedback object.
2. Click on the validation rule >> click New.
3. Enter the Rule name as “ rating_should_be_less_than_5 ”.
4. Insert the Error Condition Formula as : -

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

5. 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.

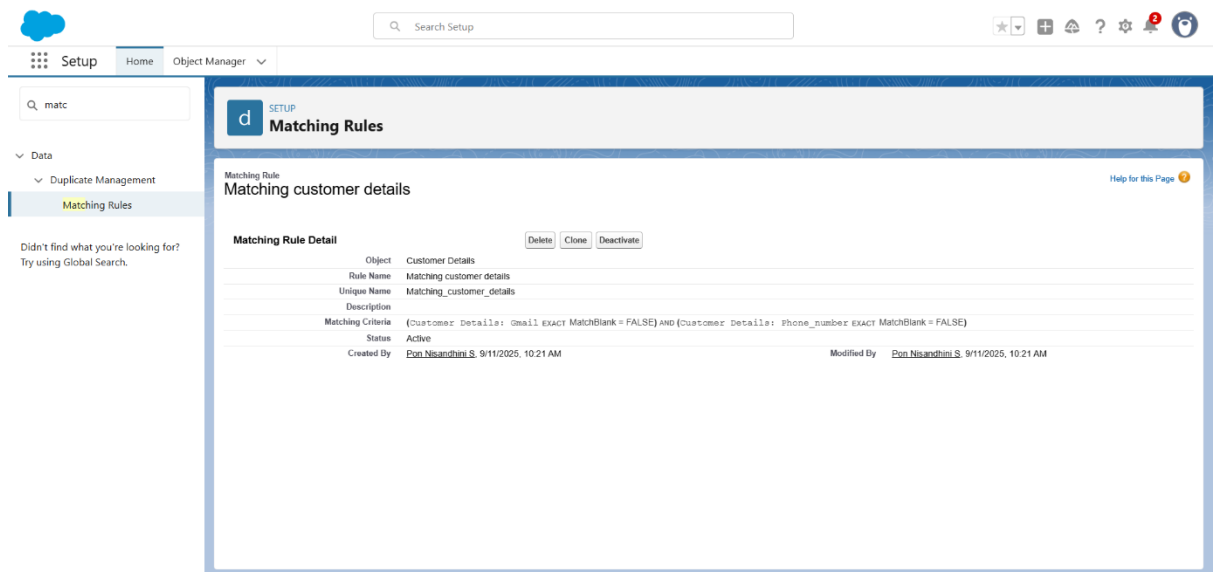
The screenshot shows the Salesforce Setup page with the 'Object Manager' tab selected. The left sidebar lists various setup options, and the main content area displays the 'Validation Rule Edit' form for the 'Billing details and feedback' object. The form includes fields for 'Rule Name' (set to 'rating_should_be_less_than_5'), 'Active' (checked), and 'Description'. The 'Error Condition Formula' section shows a formula: `NOT(REGEX(Rating_for_service__c , "[1-5]{1}"))`. A 'Functions' list on the right includes ABS, ACOS, ADDMONTHS, AND, ASCII, and ASIN. A 'Quick Tips' section on the right mentions 'Operators & Functions'.

Milestone 7 : DUPLICATE RULE

Activity 1: To create a matching rule to an Customer details Object

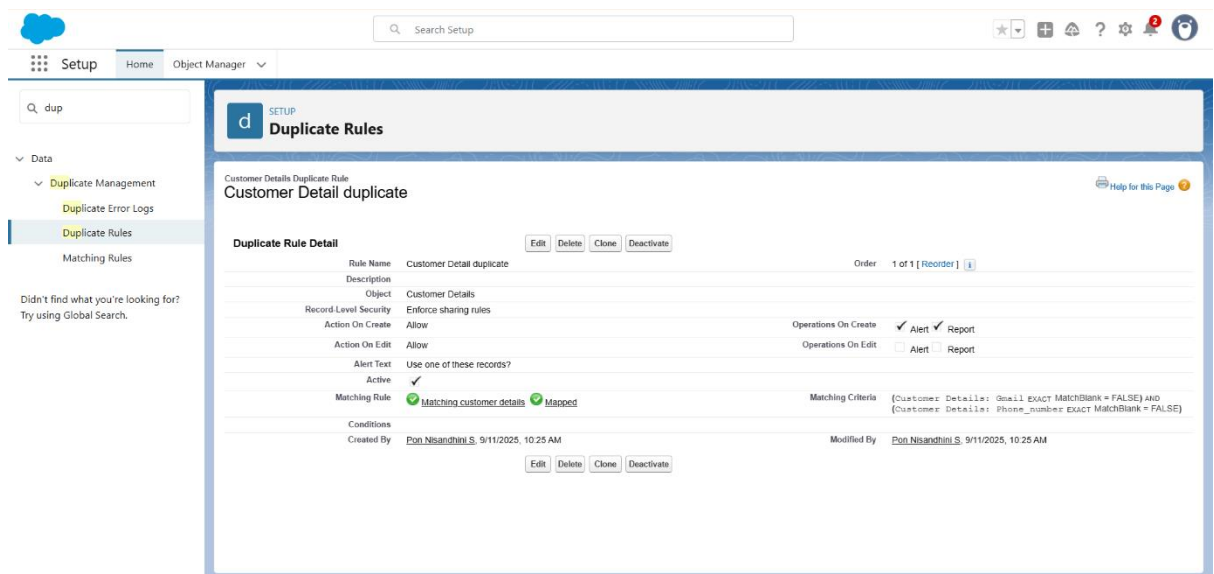
1. Go to quick find box in setup and search for matching Rule.
2. Click on matching rule >> click on New Rule.
3. Select the object as Customer details and click Next.
4. Give the Rule name : Matching customer details
5. Unique name : is auto populated
6. Define the matching criteria as

Field	Matching Method
• 1. Gmail	Exact
• 2. Phone Number	Exact
7. Click save. After Saving Click on Activate.



Activity 2: To create a Duplicate rule to an Customer details Object

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



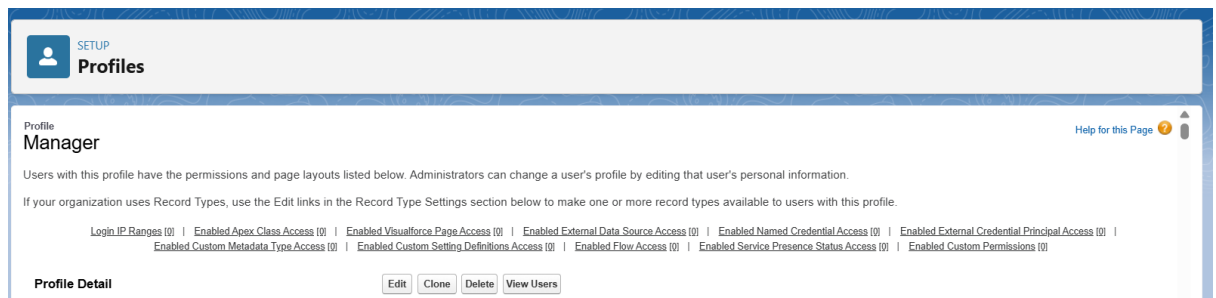
Phase 4 : Data Migration, Testing & Security

Milestone 8: PROFILES

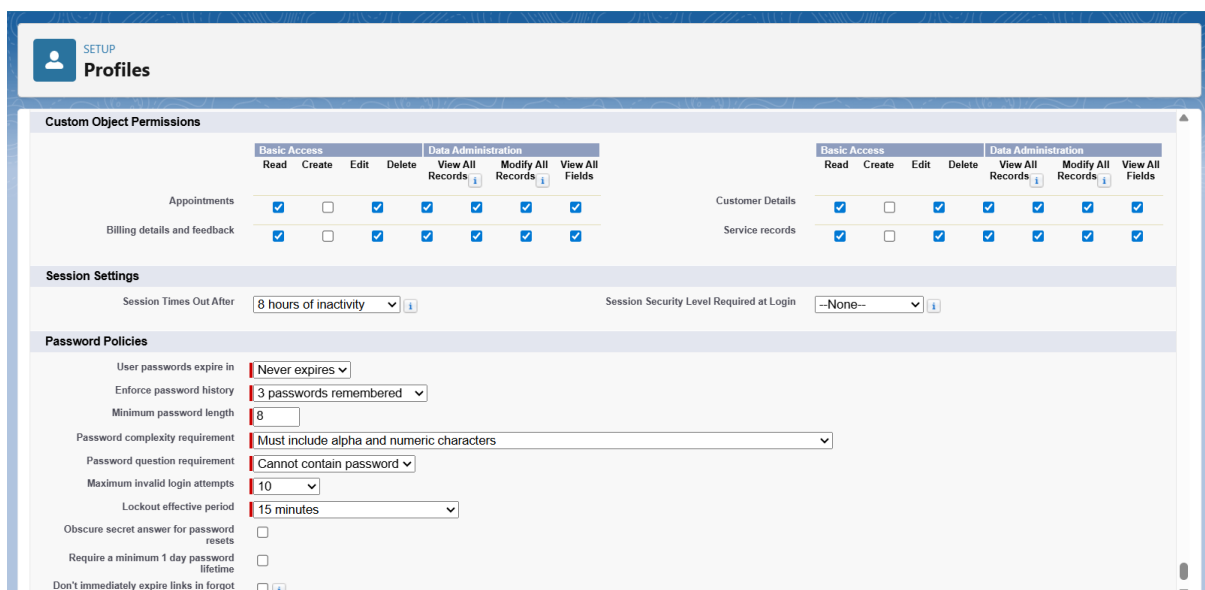
Activity 1: Manager Profile

To create a new profile:

1. Go to setup >> type profiles in quick find box >> click on profiles >> clone the desired profile (Standard User) >> enter profile name (Manager) >> Save.
2. While still on the profile page, then click Edit.



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



Activity 2: Salesperson Profile

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

The screenshot shows the 'Profiles' page in Salesforce Setup. The 'Custom Object Permissions' section is expanded, showing permissions for 'Appointments', 'Billing details and feedback', 'Customer Details', and 'Service records'. The 'Session Settings' section shows 'Session Times Out After' set to '2 hours of inactivity' and 'Session Security Level Required at Login' set to '--None--'. The 'Password Policies' section shows various settings like 'User passwords expire in' (90 days), 'Enforce password history' (3 passwords remembered), 'Minimum password length' (8), 'Password complexity requirement' (Must include alpha and numeric characters), 'Password question requirement' (Cannot contain password), 'Maximum invalid login attempts' (10), and 'Lockout effective period' (15 minutes).

Milestone 9: CREATING MANAGER ROLE

Activity 1: Creating Manager Role:

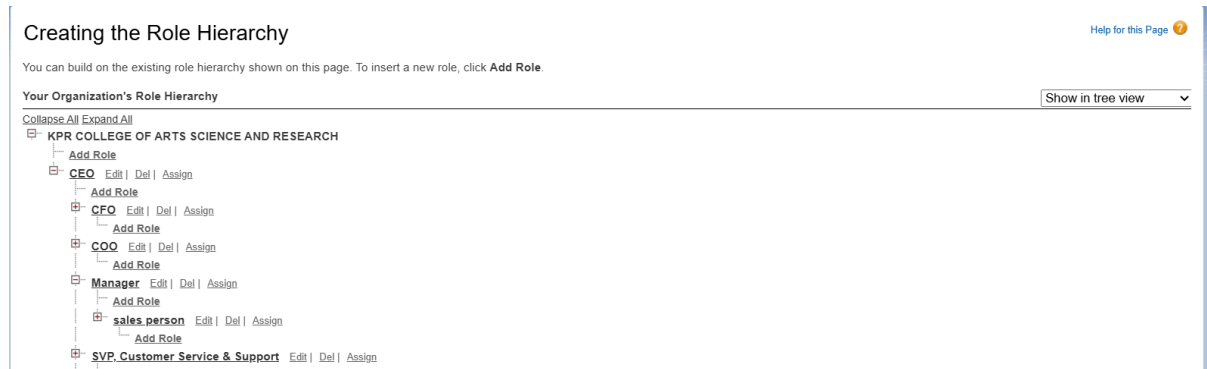
1. Go to quick find >> Search for Roles >> click on set up roles.
2. Click on Expand All and click on add role under whom this role works.
3. Give Label as “Manager” and Role name gets auto populated. Then click on Save.

The screenshot shows the 'Roles' page in Salesforce Setup. The 'New Role' form is displayed, with the 'Label' field set to 'Manager' and the 'Role Name' field set to 'Manager'. The 'This role reports to' field is set to 'CEO'. The 'Role Name as displayed on reports' field is empty. The 'Save' button is highlighted.

Activity 2: Creating Another Roles

Creating another two 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.

Milestone 10: USERS

Activity 1: Create User

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

General Information ! Required Information

First Name <input type="text" value="Niklaus"/>	Role <input type="text" value="Manager"/>
Last Name <input type="text" value="Mikaelson"/>	User License <input type="text" value="Salesforce"/>
Alias <input type="text" value="nmika"/>	Profile <input type="text" value="Manager"/>
Email <input type="text" value="niklaus@gmail.com"/>	Active <input checked="" type="checkbox"/>
Username <input type="text" value="niklaus@nmika.com"/>	Marketing User <input type="checkbox"/>
Nickname <input type="text" value="nik"/>	Offline User <input type="checkbox"/>
Title <input type="text"/>	Knowledge User <input type="checkbox"/>
Company <input type="text"/>	Flow User <input type="checkbox"/>
Department <input type="text"/>	Service Cloud User <input type="checkbox"/>
Division <input type="text"/>	Site.com Contributor User <input type="checkbox"/>
	Site.com Publisher User <input type="checkbox"/>
	WDC User <input type="checkbox"/>
	Data.com User Type <input type="text" value="--None--"/>
	Data.com Monthly Addition Limit <input type="text" value="300"/>
	Accessibility Mode (Classic Only) <input type="checkbox"/>

Activity 2: Creating another users

1. Repeat the steps and create another user using

- Role: sales person
- User licence: Salesforce Platform
- Profile: sales person

Note : create atleast 3 users with these permissions.

All Users Help for this Page

On this page you can create, view, and manage users.

To get more licenses, use the Your Account app. [Let's Go](#)

View: Edit | [Create New View](#)

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

Action	Full Name	Alias	Username	Role	Active	Profile
<input type="checkbox"/> Edit	Chatter Expert	Chatter	chatty.00d9k00000awkrpuaz.zugvrtfyp1@chatter.salesforce.com		✓	Chatter Free User
<input type="checkbox"/> Edit	EPIC_OrgFarm	OEPIK	epic.7a1046338ea0@orgfarm.salesforce.com		✓	System Administrator
<input type="checkbox"/> Edit	force_sales	sforc	sales@force.stcom	sales_person	✓	sales_person
<input type="checkbox"/> Edit	Mikaelson_Niklaus	nmika	niklaus@nmika.com	Manager	✓	Manager
<input type="checkbox"/> Edit	Mikaelson_Niklaus	nmika	sales@nmikal.com	sales_person	✓	sales_person
<input type="checkbox"/> Edit	S_Pon Nisandhini	23p	23bda043804@agentforce.com		✓	System Administrator
<input type="checkbox"/> Edit	User_Integration	integ	integration@00d9k00000awkrpuaz.com		✓	Analytics Cloud Integration User
<input type="checkbox"/> Edit	User_Security	sec	insightssecurity@00d9k00000awkrpuaz.com		✓	Analytics Cloud Security User

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](#)

Milestone 11: PUBLIC GROUPS

Activity 1: Creating New Public Group

1. Go to setup >> type users in quick find box >> select public groups >> click New.

Q public groups

Users

Public Groups

Public Groups

A public group is a set of users. It can contain individual users, other groups, the users in a particular role or territory, or the users in a role or territory plus all of the users below that role or territory in the hierarchy.

View: Edit | [Create New View](#)

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](#)

Label	Group Name	Created By	Created Date
No records to display.			

2. Give the Label as “sales team”.
3. Group name is autopopulated.
4. Search for Roles.
5. In Available Members select Sales person and click on add it will be moved to selected member.
6. Click on save.

Milestone 12: SHARING SETTING

Activity 1: Creating Sharing settings

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

SETUP Sharing Settings

Note: “Roles and subordinates” includes all users in a role, and the roles below that role. You can use sharing rules only to grant wider access to data, not to restrict access.

Step 1: Rule Name

Label:

Rule Name:

Description:

Step 2: Select your rule type

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

Step 3: Select which records to be shared

Service records: owned by members of

Step 4: Select the users to share with

Share with:

Step 5: Select the level of access for the users

Access Level:

Milestone 13: FLOWS

Activity 1: Create a Flow

1. Go to setup >> type Flow in quick find box >> Click on the Flow and Select the New Flow.
2. Select the Record-triggered flow and Click on Create.

New Automation

Get Started with Automations
Select a category, flow type, use search, or let Einstein build an automation for you.

Search automations...

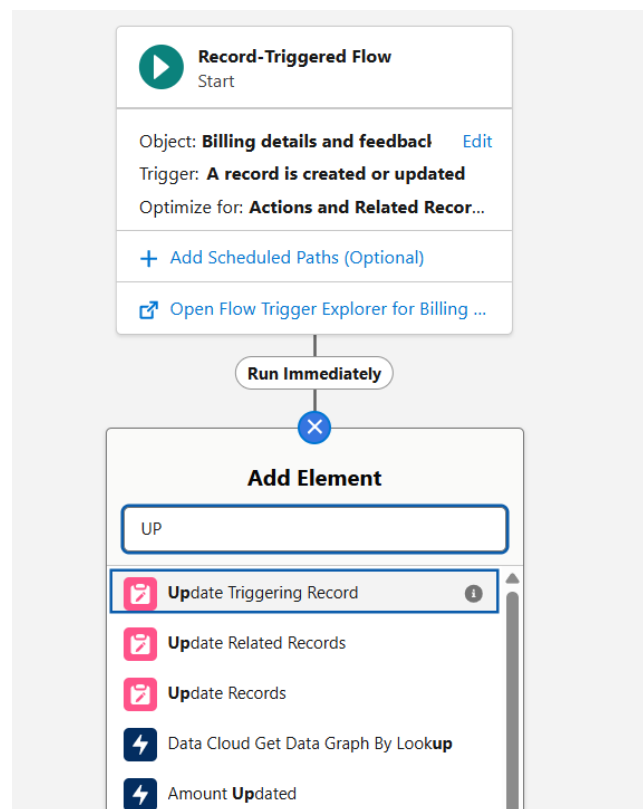
Categories

- Triggered**
Automations launched by records and events. This type of automation runs without user interaction.
[View All](#)
- Scheduled**
Time-based automations that launch at a specific time or frequency. This type of automation runs without user interaction.
[View All](#)
- Screen**
Interface-driven automations that guide users through business processes. This type of automation collects or displays...
[View All](#)
- Autolaunched**
Automations that automatically launch when invoked by APIs, templates, processes, conditions, or something else...
[View All](#)

Frequently Used

- Record-Triggered Flow**
Launches when a record is created, updated, or deleted. This autolaunched flow runs in the background.
[View All](#)
- Screen Flow**
Guides users through a business process that's launched from Lightning pages, Experience Cloud sites, quick actions, an...
[View All](#)
- Schedule-Triggered Flow**
Launches at a specified time and frequency for each record in a batch. This autolaunched flow runs in the...
[View All](#)
- Autolaunched Flow (No Trigger)**
Launches when invoked by Apex, processes, REST API, and more. This autolaunched flow runs in the...
[View All](#)

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)

Update Records

* Label

Amount Update

* API Name

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

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.

Set Filter Conditions

Condition Requirements to Update Record

All Conditions Are Met (AND)

Field

Payment Status

Operator

Equals

Value

Completed

+ Add Condition

Set Field Values for the Billing details and feedback Record

Field

Payment Paid

Value

...rvice records > Appointment > Service Amount

+ Add Field

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.

Dear

{!\$Record.Service_records__r.Appointment__r.Customer_Name__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 .

* API Name ⓘ

alert

Description

* Body ⓘ

Insert a resource... 🔍 View as Plain Text ▼

Dear {!\$Record.Service_records__r.Appointment__r.Customer_Details__r.Name}

Cancel Done

23. Click done.

24. Now Click on Add Element , select Action.

25. Their action bar will be opened in that search for “ send email ” and click on it.

26. Give the label name as “ Email Alert”

27. API name will be auto populated.

28. Enable the body in set input values for the selected action.

29. Select the text template that created , Body : {!alert}

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

RecipientAddressList:

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

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

32. Click done.

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

34. And click save, and click on activate.

Record-Triggered Flow
Start

Save as

A New Version A New Flow

*Flow Label
Billing Amount Flow

*Flow API Name
Billing_Amount_Flow

Description

Show Advanced

Cancel Save

Activity 2: Create another Flow

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

14. Given the Flow label as **Update Service Status**, Flow Api name will be auto populated.

15. And click save, and click on **activate**.

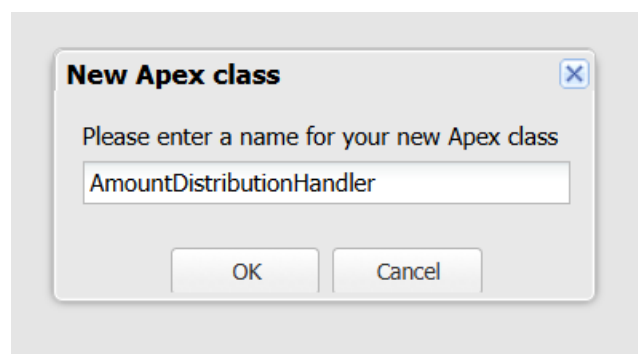
The screenshot displays the 'Set Filter Conditions' section of a Salesforce Flow. Under 'Condition Requirements to Update Record', a dropdown menu is set to 'All Conditions Are Met (AND)'. Below this, a table-like structure shows a single condition: the field 'Quality_Check_Status__c' is set to 'Equals' the value 'True'. To the right of the value field is a trash icon. Below the condition table is a '+ Add Condition' button. The second section, 'Set Field Values for the Service record Record', shows a table with the field 'Service_Status__c' and the value 'Completed'. Similar to the first section, there is a trash icon to the right of the value field and a '+ Add Field' button below the table.

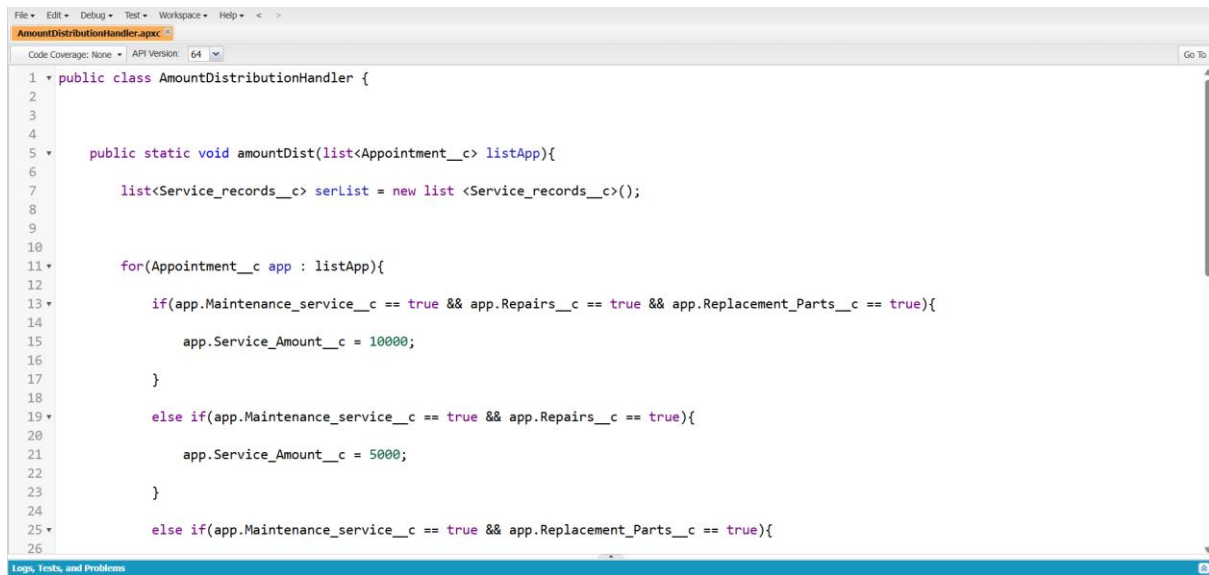
Milestone 14: APEX TRIGGER

Activity 1: Apex handler

Use Case : 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”.



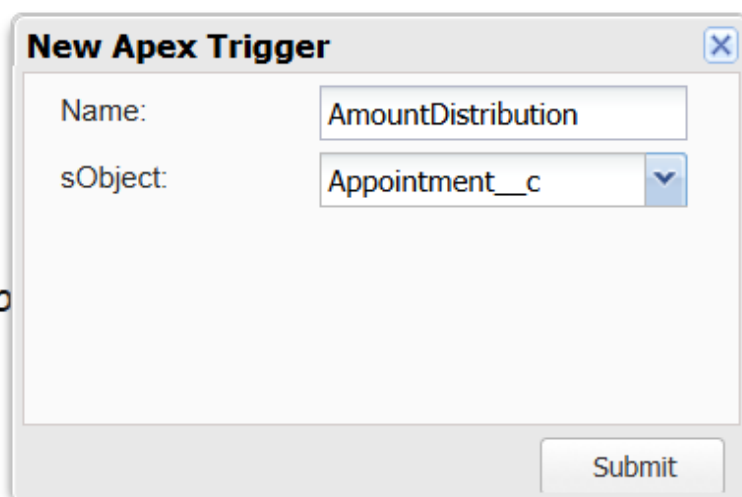
A screenshot of an IDE window showing the 'AmountDistributionHandler.apex' file. The code defines a public class 'AmountDistributionHandler' with a static method 'amountDist' that takes a list of 'Appointment__c' objects. Inside the method, a list of 'Service_records__c' is created. A loop iterates over each 'Appointment__c' object. If 'Maintenance_service__c' is true and 'Repairs__c' is true, 'Service_Amount__c' is set to 10000. If 'Maintenance_service__c' is true and 'Repairs__c' is false, 'Service_Amount__c' is set to 5000. If 'Maintenance_service__c' is false and 'Repairs__c' is true, 'Service_Amount__c' is set to 5000. The code is as follows:

```
1 public class AmountDistributionHandler {
2
3
4
5 public static void amountDist(list<Appointment__c> listApp){
6
7     list<Service_records__c> serList = new list<Service_records__c>();
8
9
10
11 for(Appointment__c app : listApp){
12
13     if(app.Maintenance_service__c == true && app.Repairs__c == true && app.Replacement_Parts__c == true){
14
15         app.Service_Amount__c = 10000;
16
17     }
18
19     else if(app.Maintenance_service__c == true && app.Repairs__c == false){
20
21         app.Service_Amount__c = 5000;
22
23     }
24
25     else if(app.Maintenance_service__c == false && app.Replacement_Parts__c == true){
26
```

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.
 - Name : AmountDistribution
 - sObject : Appointment__c

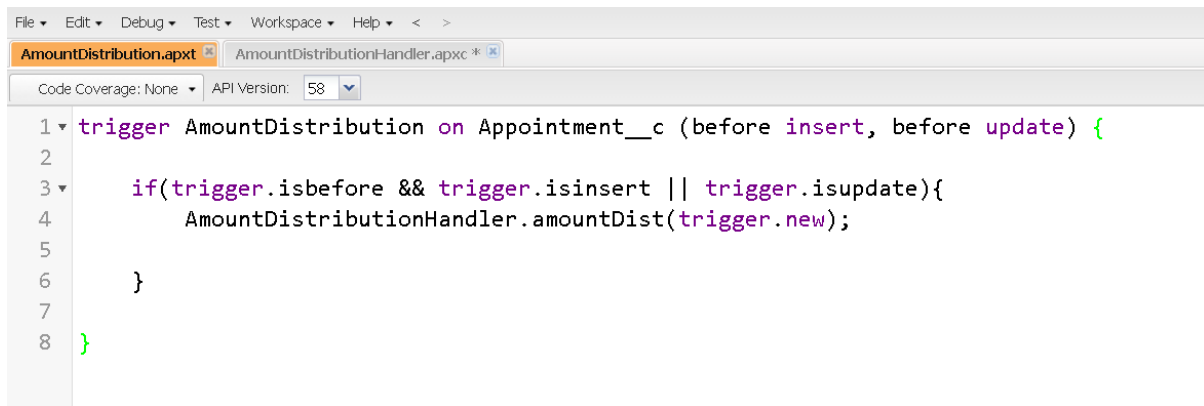
A screenshot of the 'New Apex Trigger' dialog box. It has a title bar with a close button. Inside, there are two labels: 'Name:' and 'sObject:'. The 'Name:' label is followed by a text input field containing 'AmountDistribution'. The 'sObject:' label is followed by a dropdown menu showing 'Appointment__c'. At the bottom right, there is a 'Submit' button.

New Apex Trigger

Name: AmountDistribution

sObject: Appointment__c

Submit

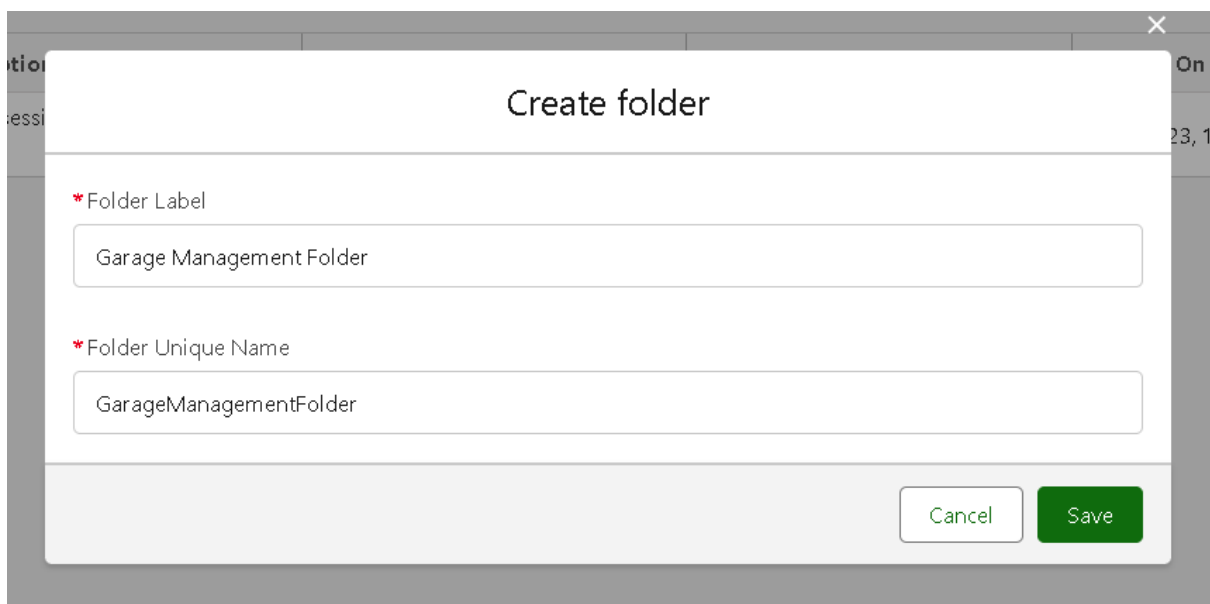


```
File Edit Debug Test Workspace Help < >
AmountDistribution.apxt AmountDistributionHandler.apxc *
Code Coverage: None API Version: 58
1 trigger AmountDistribution on Appointment__c (before insert, before update) {
2
3     if(trigger.isbefore && trigger.isinsert || trigger.isupdate){
4         AmountDistributionHandler.amountDist(trigger.new);
5     }
6 }
7
8 }
```

Milestone 15: REPORTS

Activity 1: Create 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.



Create folder

* Folder Label

Garage Management Folder

* Folder Unique Name

GarageManagementFolder

Cancel Save

Activity 2: 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.

Activity 3: Create Report Type

1. Go to setup >> type users in quick find box >> select Report Type >> click on Continue.
2. Click on new custom report type.
3. Select the Primary object as “ Customer details” .
4. Give the Report type Label as “ Service information ”
5. Report type Name is auto populated.Keep the Description as same.
6. Select Store in Category as “ other Reports ”
7. Select the deployment status as “ Depolyed ”, click on Next.
8. Now , Click on Related object box.
9. Click on Select Object, choose Appointment Object.
- 10.Again Click to relate another object.
- 11.And select the related object as “service records”.
- 12.Repeat the process and select the related object as “Billing details and feedback”.
- 13.And click on save.

Select related objects to define which records are included in reports using this report type.

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.

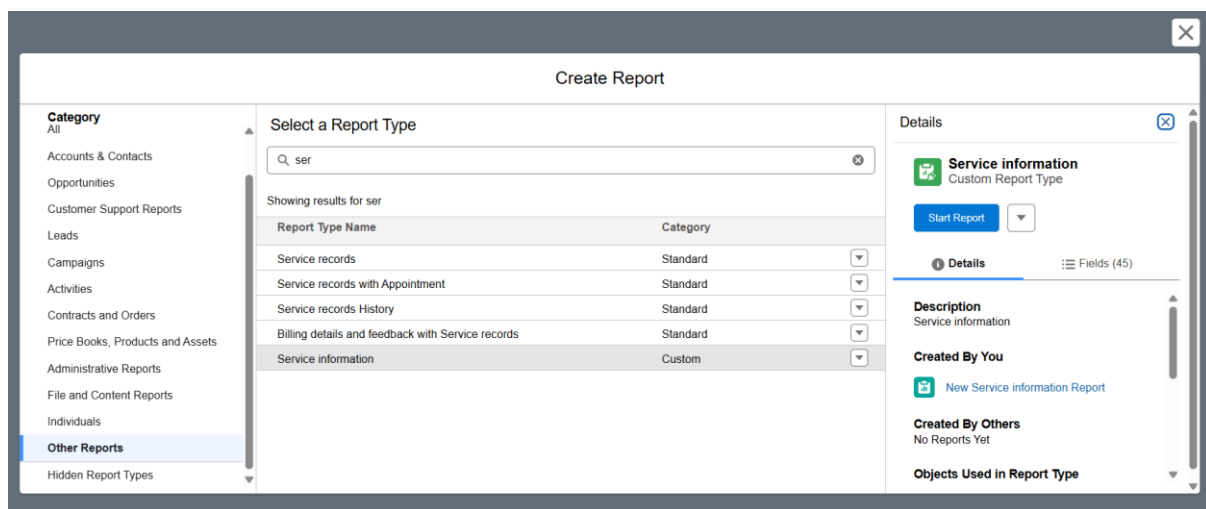
Cancel Save

Activity 4: Create Report

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

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

1. Go to the app >> click on the reports tab
2. Click New Report.
3. Select the Category as other reports, search for Service Information, select that report, click on it. And click on start report.



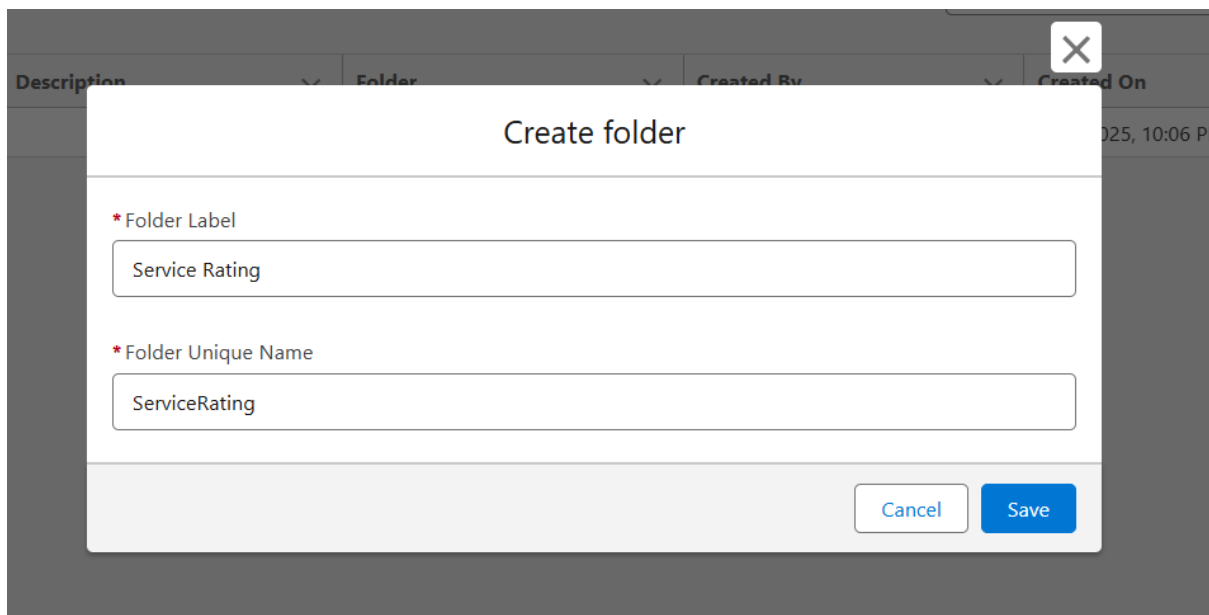
4. Their outline pane is opened already, select the fields that mentioned below in column section.
 - Customer name
 - Appointment Date
 - Service Status
 - Payment paid
5. Remove the unnecessary fields.
6. Select the fields that mentioned below in GROUP ROWS section.
 - Rating for Service
7. Select the fields that mentioned below in GROUP ROWS section.
 - Payment Status
8. Click on Add Chart, Select the Line Chart.
9. Click on save, Give the report Name: New Service information Report

10. Report unique Name is auto populated.
11. Select the folder the created and click on save.

Milestone 16: DASHBOARDS

Activity1: Create Dashboard Folder

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

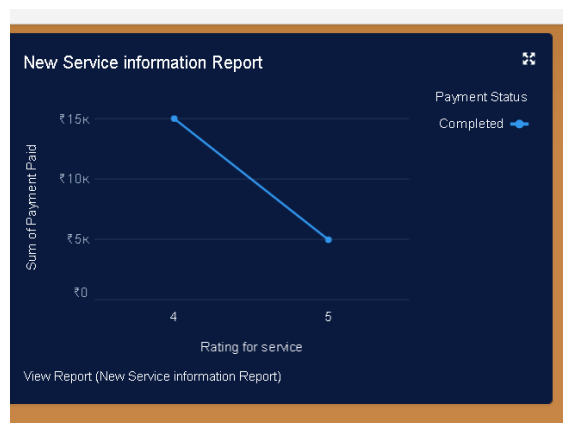


The screenshot shows a 'Create folder' dialog box overlaid on a dashboard interface. The dialog box has a title bar with a close button (X). It contains two input fields: 'Folder Label' with the text 'Service Rating' and 'Folder Unique Name' with the text 'ServiceRating'. Both fields are preceded by a red asterisk indicating they are required. At the bottom right of the dialog box are two buttons: 'Cancel' and 'Save'.

Activity 2: Create Dashboard

1. Go to the app >> click on the Dashboards tabs.
2. Give a Name and select the folder that created, and click on create.
3. Select add component.
4. Select a Report and click on select.
5. Select the Line Chart. Change the theme.
6. Click Add then click on Save and then click on Done.

7. Preview is shown below.



8. After that Click on Subscribe on top right.

9. Set the Frequency as “weekly”.

10. Set a day as Monday.

11. And click on save.

Edit Subscription

Schedule dashboard refreshes and subscribe to receive results.

Settings

Frequency

Days

Time

Recipients

⚠ Recipients see the same report data as the person running the report.

☒ Receive new results by email when dashboard is refreshed. ⓘ

Send email to

Me




Milestone 17: USER ADOPTION

Activity 1: Creating 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 “ **Customer details** tab”.
4. Click on new and fill the details as shown below figs, and click save.

* = Required Information

*Customer Name	Owner
<input type="text" value="Mac"/>	 Pon Nisandhini S
Phone number	
<input type="text" value="5678765567"/>	
Gmail	
<input type="text" value="mac@gmail.com"/>	
Created By	Last Modified By
 Pon Nisandhini S, 9/12/2025, 8:57 PM	 Pon Nisandhini S, 9/12/2025, 8:57 PM

CancelSave & NewSave

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.

Related	Details
Appointment Name	app-005
Customer Details	Mac
Appointment Date	9/3/2025, 12:00 PM
Service Amount	\$1,000
Payment Paid	
Vehicle number plate	TN01 TS 40
Customer Name	
Created By	Pon Nisandhini S , 9/12/2025, 9:47 PM
Owner	Pon Nisandhini S
Last Modified By	Pon Nisandhini S , 9/14/2025, 9:00 AM

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

Related	Details
Service records Name	ser-001
Appointment	app-005
Quality Check Status	<input checked="" type="checkbox"/>
Service Status	Completed
service date	9/13/2025
Created By	Pon Nisandhini S , 9/12/2025, 9:48 PM
Owner	Pon Nisandhini S
Last Modified By	Pon Nisandhini S , 9/12/2025, 9:48 PM

CONCLUSION:

The project effectively created a transparent and efficient method for overseeing garage operations by utilising the Salesforce platform. The initiative improved

customer satisfaction and operational efficiency by processing service requests, vehicle history, spare parts, and client interactions efficiently.

The successfully completed project "**Garage Management System using Salesforce**" shows how Salesforce CRM can be used practically in the auto repair sector.

- **Project accomplishments:**

- Made garage administration easier by effectively managing client billing, car records, and service requests.
- Made it possible for customers, service advisors, and mechanics to coordinate and communicate easily.
- To cut down on human labour, automation was implemented using custom objects, fields, Flows, and Apex triggers.
- Enhanced transparency through controlled data access, dashboards, and interactive reporting.
- Enhanced user experience thanks to a customised Lightning App, unique Home Page, and user-friendly UI.

- **Learning Outcomes for Students:**

- Acquired hands-on experience with CRM technologies designed for garage operations, automation, and Salesforce customisation.
- Improved analytical and problem-solving abilities through practical automobile service situations.
- Active involvement in requirement collecting, development, and testing led to the development of collaboration and communication skills.
- Gained knowledge of end-to-end project lifecycle management and industry-standard tools and techniques.

- **Learning outcomes for students:**

- **AI-Powered Predictive Maintenance:** Reduce breakdowns by integrating AI models to forecast vehicle issues based on usage patterns and service history.

- **IoT Integration:** Link sensors in cars with IoT capabilities to transmit real-time diagnostic data straight to the system.
- **Customer Mobile App:** Create a specific app for scheduling services, monitoring the status of repairs, getting alerts, and completing payments.
- **Advanced Analytics:** Use machine learning and data analytics to forecast inventories, optimise performance, and gain insights into customer behaviour.
- **Integration of Payment Gateways:** Facilitate safe and easy online payments for services and replacement parts.