NAAN MUDHALVAN- SALESFORCE REPORT

A CRM Application To Manage The Services

Offered by an Institution

Department of Artificial Intelligence And Data Science

Project created by

Arunkumar V - 421622243003

Keerthi K - 421622243026

Yashwanth Kumar A - 421622243057

Pugazhendhi N - 421622243041

Hariharan S -421622243020

PROJECT ABSTRACT

The Salesforce-based Garage Management System (GMS) is a sophisticated solution tailored for automotive repair facilities to elevate service quality, streamline operations, and cultivate enduring customer relationships. Leveraging Salesforce's powerful CRM capabilities, GMS offers an intuitive interface and comprehensive features, allowing garages to thrive in a competitive marketplace. The system automates critical processes such as appointment scheduling, inventory management, billing, and customer communications. By integrating these functionalities within Salesforce, GMS ensures a seamless and satisfying experience for both customers and staff. This advanced solution empowers automotive repair businesses to deliver top-notch service, optimize workflows, and maintain a competitive edge.

INTRODUCTION

Running an automotive repair shop involves juggling many tasks, from scheduling appointments to managing inventory and billing customers. The Salesforce-based Garage Management System (GMS) is designed to make these tasks easier and more efficient.

This system uses Salesforce's powerful tools to help garages deliver better service and keep operations running smoothly. With GMS, repair shops can automate important tasks like booking appointments, tracking inventory, sending bills, and communicating with customers.

The user-friendly design of GMS ensures that both staff and customers have a pleasant experience. By using this system, garages can stay ahead of the competition, provide exent service, and manage their business more effectively. GMS not only simplifies daily operations but also provides valuable insights to help make better business decisions.

INDEX PAGE

TASK 1

1.1 Creating the Developer Account

1)To create a developer account in Salesforce, follow these steps:

2)Go to Salesforce Developer Signup.

- Fill in the signup form with the following details:
- First name & Last name
- Email
- Role: Developer
- Company: College Name
- Country: India

- Postal Code: Pin code
- Username: A combination of your name and company (format: username@organization.com, not an actual email id)

3)Click Sign me up.

1.2 Activating the Account

- Check the inbox of the email you used for signup.
- Click on the verification link to activate your account (the email may take 5-10 minutes to arrive).
- Click on **Verify Account**.
- Set a password and answer a security question.
- Click on **Change Password**.
- You will be redirected to your Salesforce setup page.

TASK 2

Creating the Custom Objects:

This report outlines the steps to create various custom objects in Salesforce, specifically for a Garage Management System. The objects include Customer Details, Appointments, Service Records, and Billing Details and Feedback. These objects will help streamline operations and improve data management within the system.

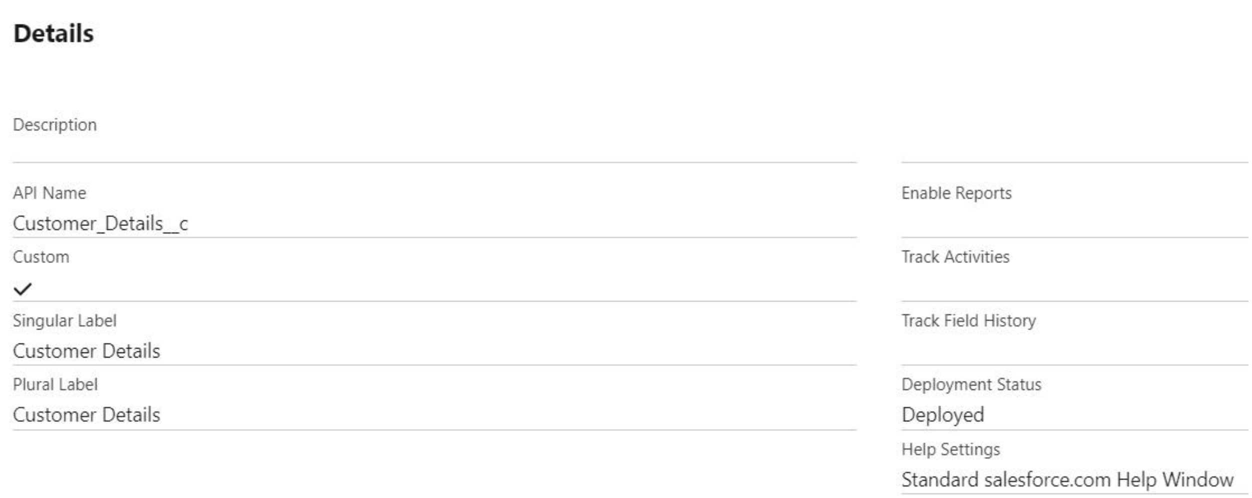
2.1 Customer Details Object

- Navigation Path: From the setup page, click on Object Manager || Create || Custom Object.

- Details:

  Label Name: Customer Details
  Plural Label Name: Customer Details
  Record Name: Customer Name
  Data Type: Text
  Options: Allow reports, Track Field History, Allow search

- Save: Click Save to create the object.



2.2 Appointment Object

- Navigation Path: From the setup page, click on Object Manager || Create || Custom Object.

- Details:

  Label Name: Appointment
  Plural Label Name: Appointments

Record Name: Appointment Name

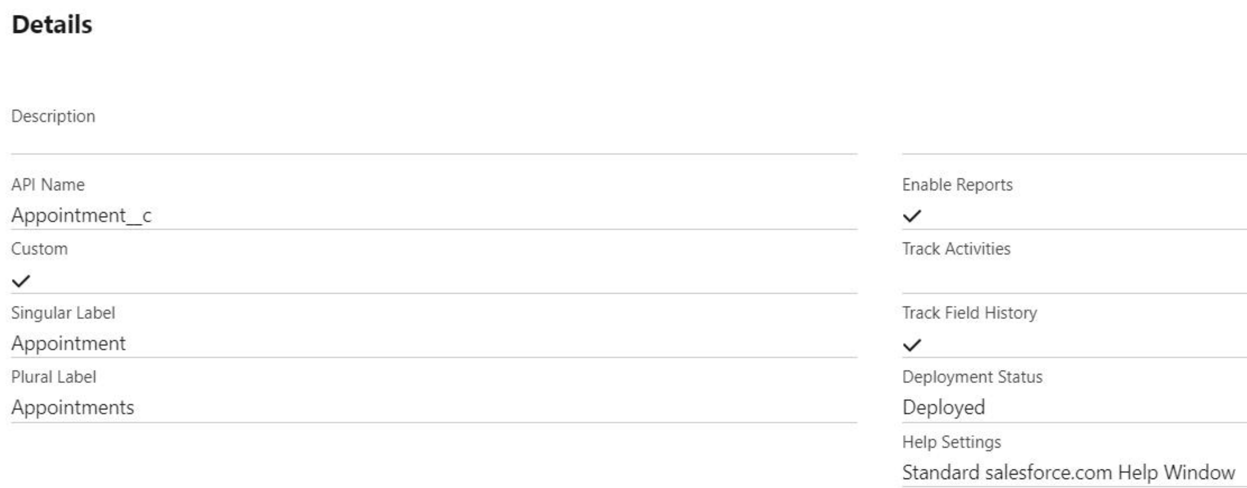
Data Type: Auto Number

Display Format: app - {000}

Starting Number: 1

Options: Allow reports, Track Field History, Allow search

- Save: Click Save to create the object.



2.3 Service Records Object

- Navigation Path: From the setup page, click on Object Manager || Create || Custom Object.
- Details:

Label Name: Service Records

Plural Label Name: Service Records

Record Name: Service Records Name

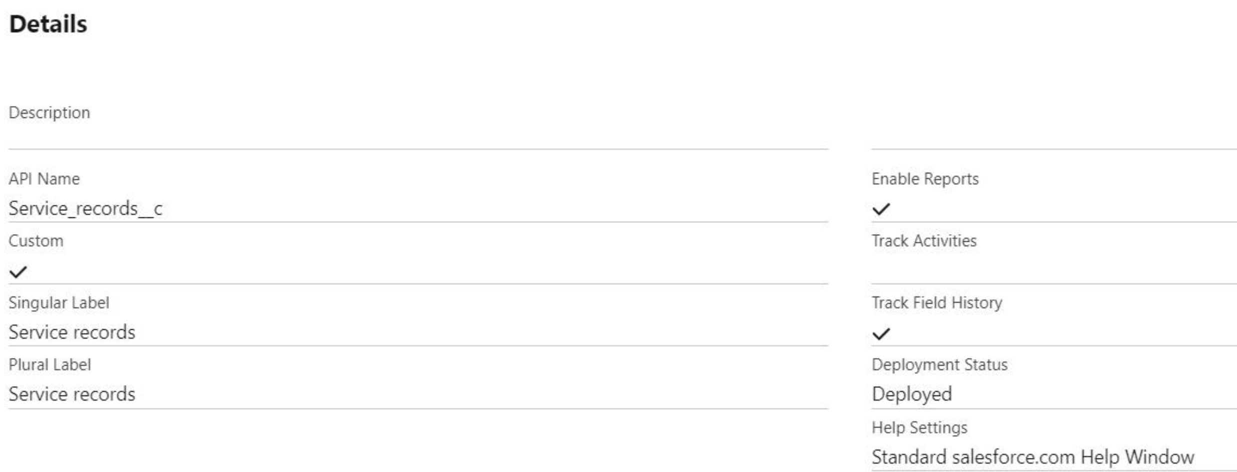
Data Type: Auto Number

Display Format: ser - {000}

Starting Number: 1

Options: Allow reports, Track Field History, Allow search

- Save: Click Save to create the object.

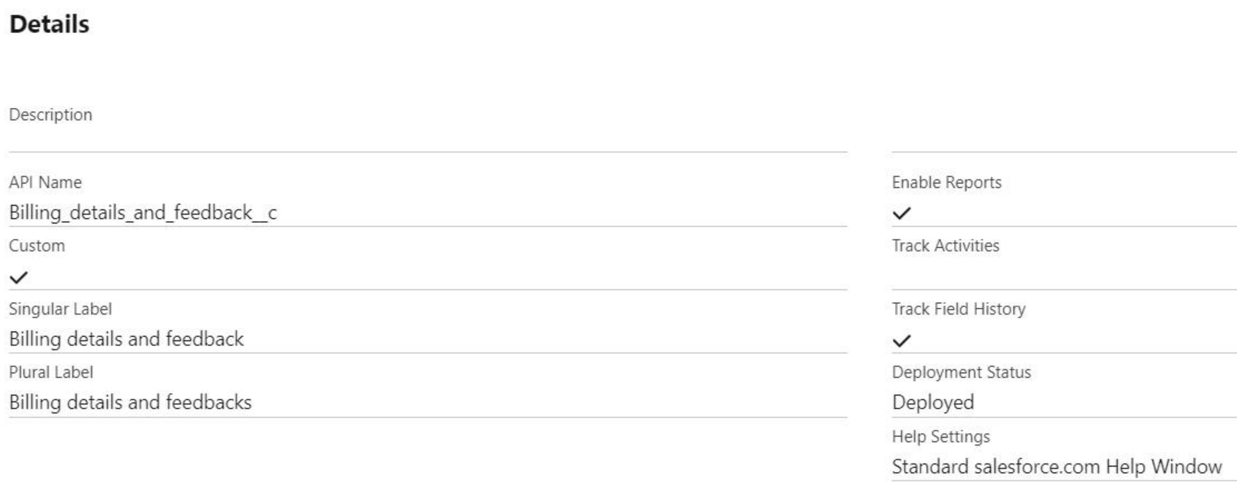


2.4 Billing Details and Feedback Object

- Navigation Path: From the setup page, click on Object Manager || Create || Custom Object.
- Details:

  Label Name: Billing Details and Feedback
  Plural Label Name: Billing Details and Feedback
  Record Name: Billing Details and Feedback Name
  Data Type: Auto Number
  Display Format: bill - {000}
  Starting Number: 1
  Options: Allow reports, Track Field History, Allow search

- Save: Click Save to create the object.



TASK 3

3.1 Creating a Custom Tab

- Creating a Custom Tab for Customer Details
- Navigation Path:

  Go to the setup page.
  Type ‘Tabs‘ in the Quick Find bar.
  Click on ‘Tabs‘.
  Click on ‘New‘ under the Custom Object Tabs.

- Details:

  Select Object: Customer Details
  Select Tab Style: Choose a style that represents Customer Details.

- Click Next.
- Add to Profiles Page: Keep the default settings.
- Click Next.
- Add to Custom App: Uncheck ‘Include Tab‘.
- Ensure ‘Append tab to users’ existing personal customizations‘ is checked.

- Click Save.

3.2 **Creating Remaining Tabs (Appointments, Service Records, Billing Details and Feedback)**

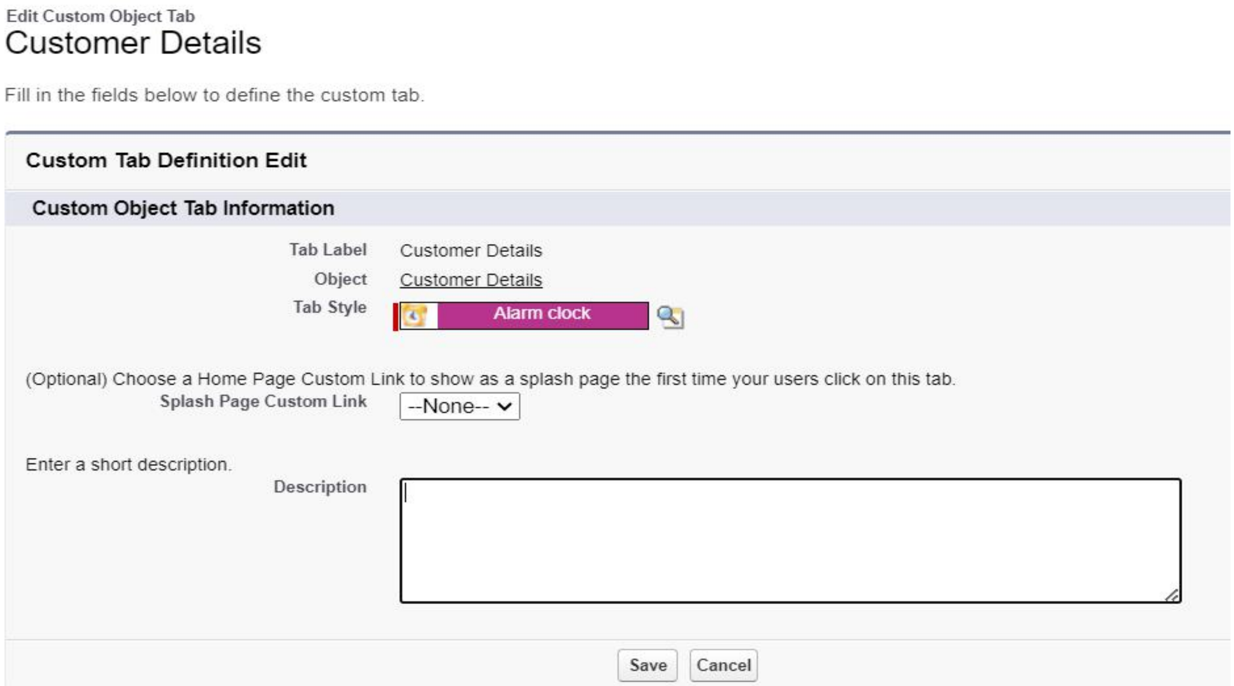
- Repeat the above steps for each remaining object: Appointments, Service Records, and Billing Details and Feedback.
- Navigation Path:

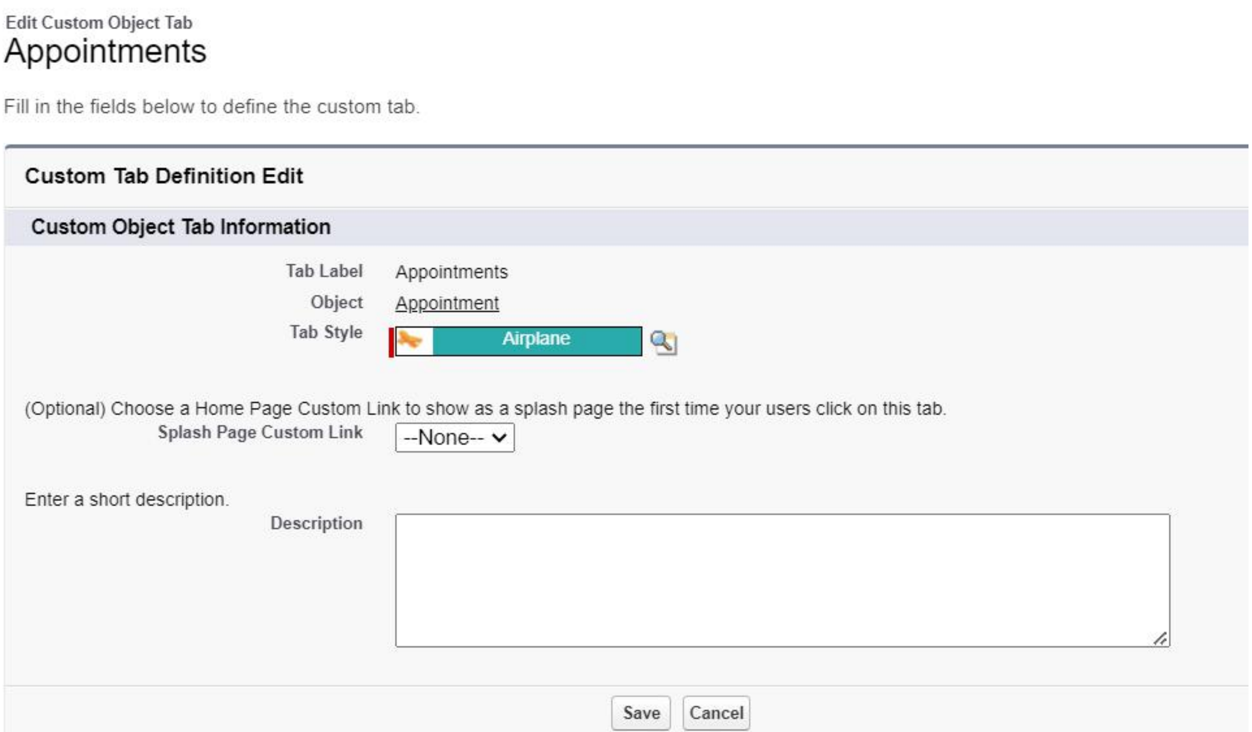
  Go to the setup page.
  Type ‘Tabs‘ in the Quick Find bar.
  Click on ‘Tabs‘.
  Click on ‘New‘ under the Custom Object Tabs.

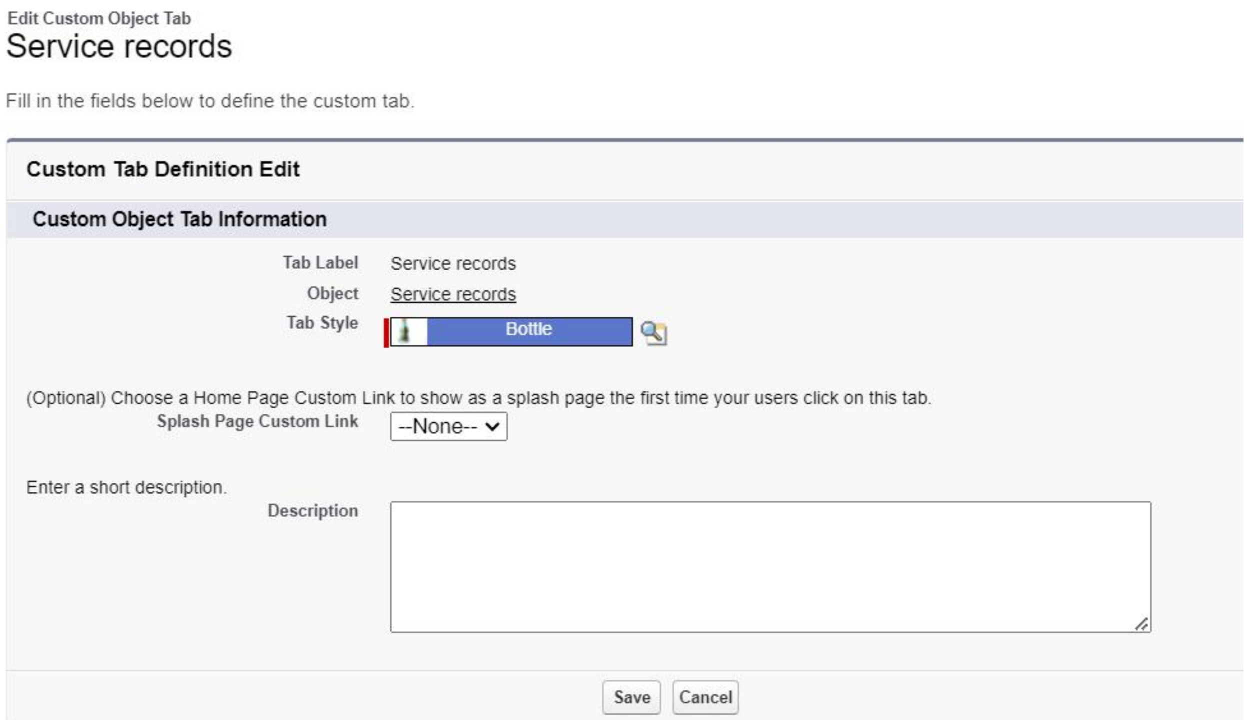
- Details for Each Object:

  Select Object: Choose the respective object (Appointments, Service Records, Billing Details and Feedback).
  Select Tab Style: Choose a suitable style for each object.

- Click Next.
- Add to Profiles Page: Keep the default settings.
- Click Next.
- Add to Custom App: Uncheck ‘Include Tab‘.
- Ensure ‘Append tab to users’ existing personal customizations‘ is checked.
- Click Save.


Edit Custom Object Tab
Customer Details
Fill in the fields below to define the custom tab.
Custom Tab Definition Edit
Custom Object Tab Information
Tab Label
Customer Details
Object
Customer Details
Tab Style
Alarm clock
(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
Save
Cancel


Edit Custom Object Tab
Appointments
Fill in the fields below to define the custom tab.
Custom Tab Definition Edit
Custom Object Tab Information
Tab Label
Appointments
Object
Appointment
Tab Style
Airplane
(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
Save
Cancel



Edit Custom Object Tab

Service records

Fill in the fields below to define the custom tab.

Custom Tab Definition Edit

Custom Object Tab Information

Tab Label	Service records
Object	Service records
Tab Style	Bottle

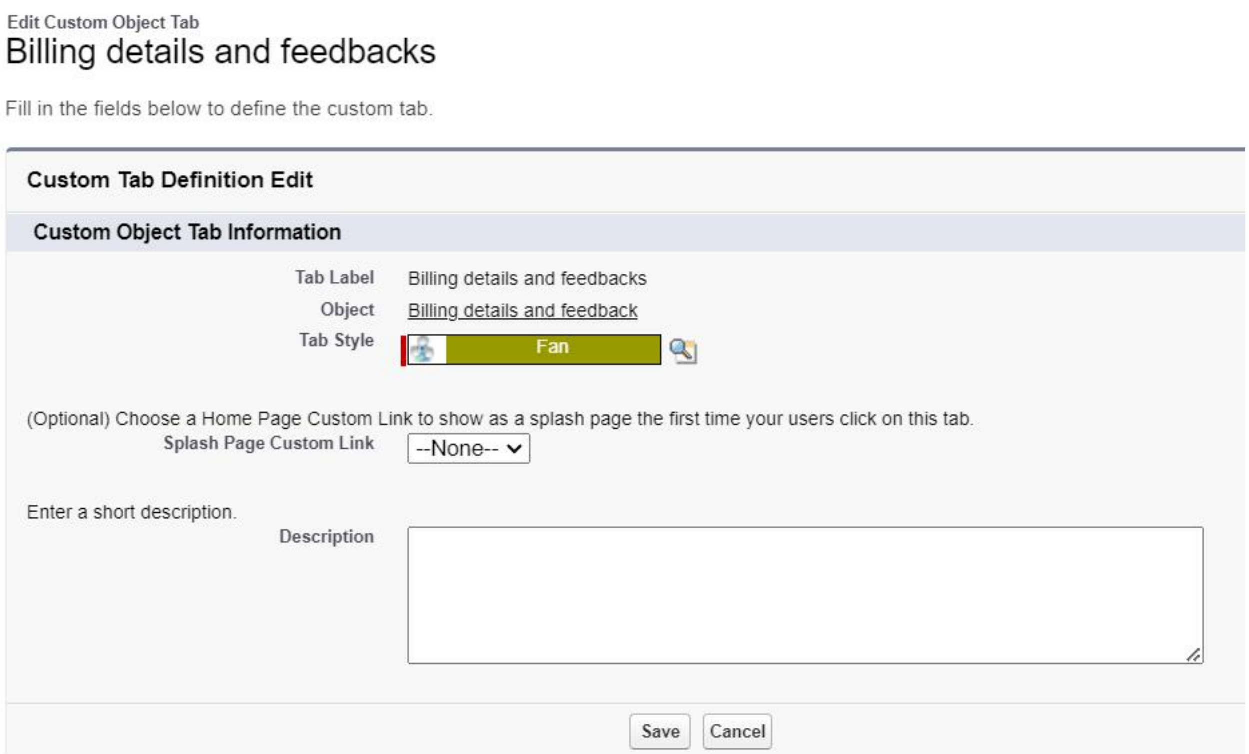
(Optional) Choose a Home Page Custom Link to show as a splash page the first time your users click on this tab.

Splash Page Custom Link	--None--

Enter a short description.

Description	

Save Cancel



Edit Custom Object Tab

Billing details and feedbacks

Fill in the fields below to define the custom tab.

Custom Tab Definition Edit

Custom Object Tab Information

Tab Label	Billing details and feedbacks
Object	Billing details and feedback
Tab Style	Fan

(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	

Save Cancel

Task 4

Building Lighting App

Creating a Lightning App in Salesforce for managing garage operations allows for streamlined processes and efficient management of various business aspects. This guide outlines the steps to create the Garage Management Application, including adding essential navigation items and user profiles.

4.1 Creating a Lightning App

Steps to Create a Lightning App

- Accessing App Manager:

  Begin by navigating to the setup page. In the Quick Find bar, search for 'App Manager' and select it. Click on 'New Lightning App' to start the creation process.

- App Details:

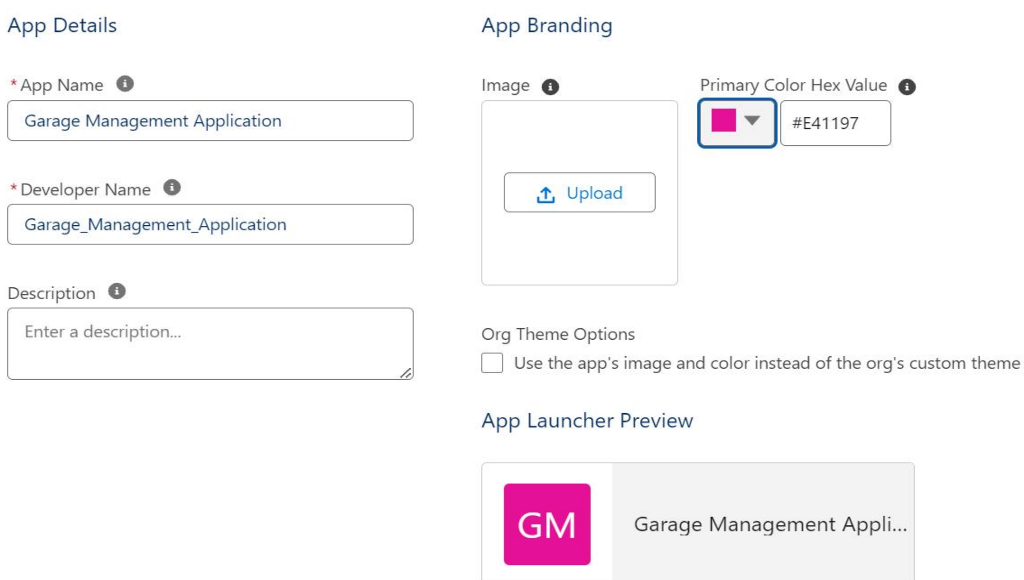
  Enter the name 'Garage Management Application' in the app details. Proceed to the next steps, keeping all settings on the App Options and Utility Items pages as default.

- Adding Navigation Items:

  To enhance navigation within the app, select essential items such as Customer Details, Appointments, Service Records, Billing Details and Feedback, Reports, and Dashboards from the search bar. Move these items using the arrow button to include them in the app's navigation menu.

- Assigning User Profiles:

To ensure appropriate access and functionality, search for the 'System Administrator' profile in the search bar. Add this profile by clicking the ar button. Complete the setup by saving and finishing the configuration.



Task 5

Creating Fields

5.1 Customer Details Object Fields

- Phone Field:

  Created a phone field labeled 'Phone Number' to store customer contact numbers.

- Email Field:

  Added an email field labeled 'Gmail' to capture customer email addresses.

5.2 Lookup Fields

- Appointment Object:

Established a lookup relationship field to link appointments to customer details, ensuring appointments are associated with the correct customer.

- Service Records Object:

  Created a lookup relationship field to link service records to appointments. Added a filter to ensure the appointment date is less than the created date, making it a required field with error validation.

- Billing Details and Feedback Object:

  Added a lookup relationship field to connect billing details and feedback with service records.

5.3 Checkbox Fields

- Appointment Object:

  Added multiple checkbox fields to indicate different services, including ‘Maintenance Service,‘ ‘Repairs,‘ and ‘Replacement Parts,‘ all with default values set to unchecked.

- Service Records Object:

  Created a checkbox field labeled ‘Quality Check Status‘ to track the quality check status of the service records.

5.4 Date Fields

- Appointment Object:

  Added a date field labeled ‘Appointment Date‘ and marked it as required to ensure each appointment has a specified date.

5.5 Currency Fields

- Appointment Object:

Created a currency field labeled ‘Service Amount‘ to record the cost of services provided, with read-only access for all profiles.

- Billing Details and Feedback Object:

  Added a currency field labeled ‘Payment Paid‘ to track payments made by customers.

5.6 Text Fields

- Appointment Object:

  Created a text field labeled ‘Vehicle Number Plate‘ with a length of 10 characters, marked as required and unique for vehicle identification.

- Billing Details and Feedback Object:

  Added a text field labeled ‘Rating for Service‘ with a length of 1 character, required and unique to capture customer ratings.

5.7 Picklist Fields

- Service Records Object:

  Created a picklist field labeled ‘Service Status‘ with values ‘Started‘ and ‘Completed‘ to track the progress of services.

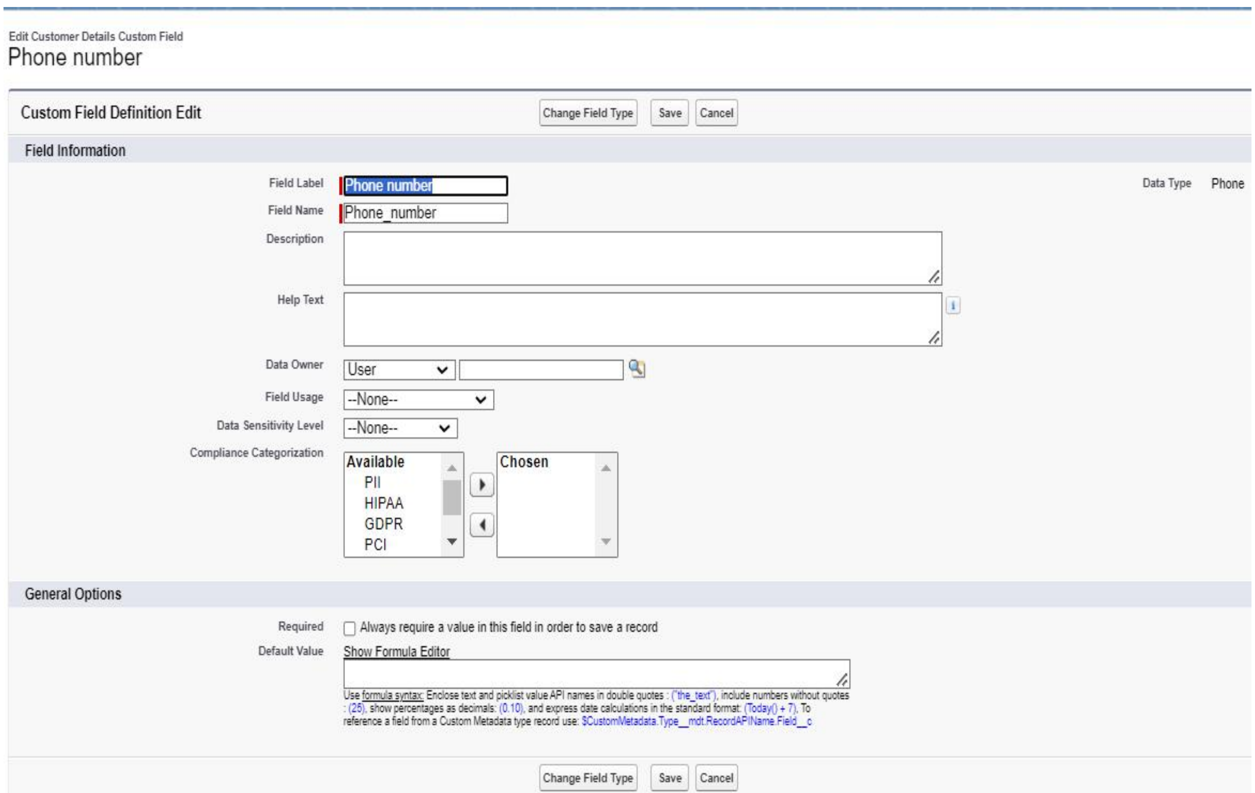
- Billing Details and Feedback Object:

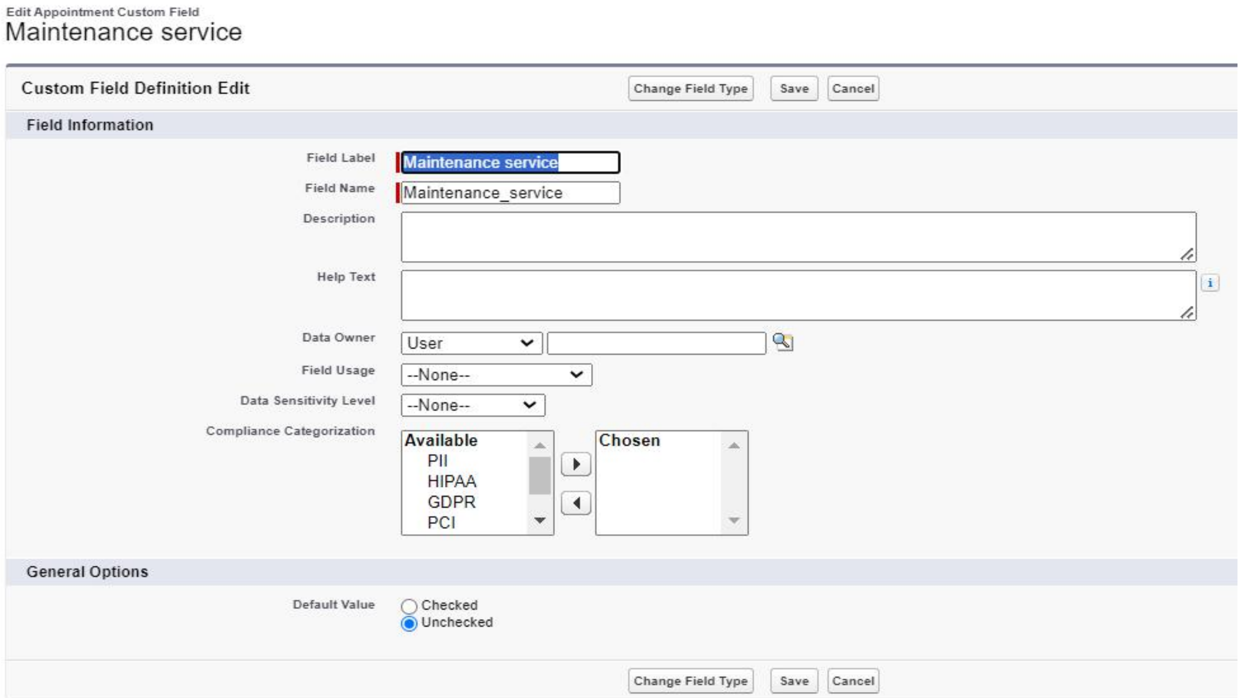
  Added a picklist field labeled ‘Payment Status‘ with values ‘Pending‘ and ‘Completed‘ to monitor payment statuses.

5.8 Formula Fields

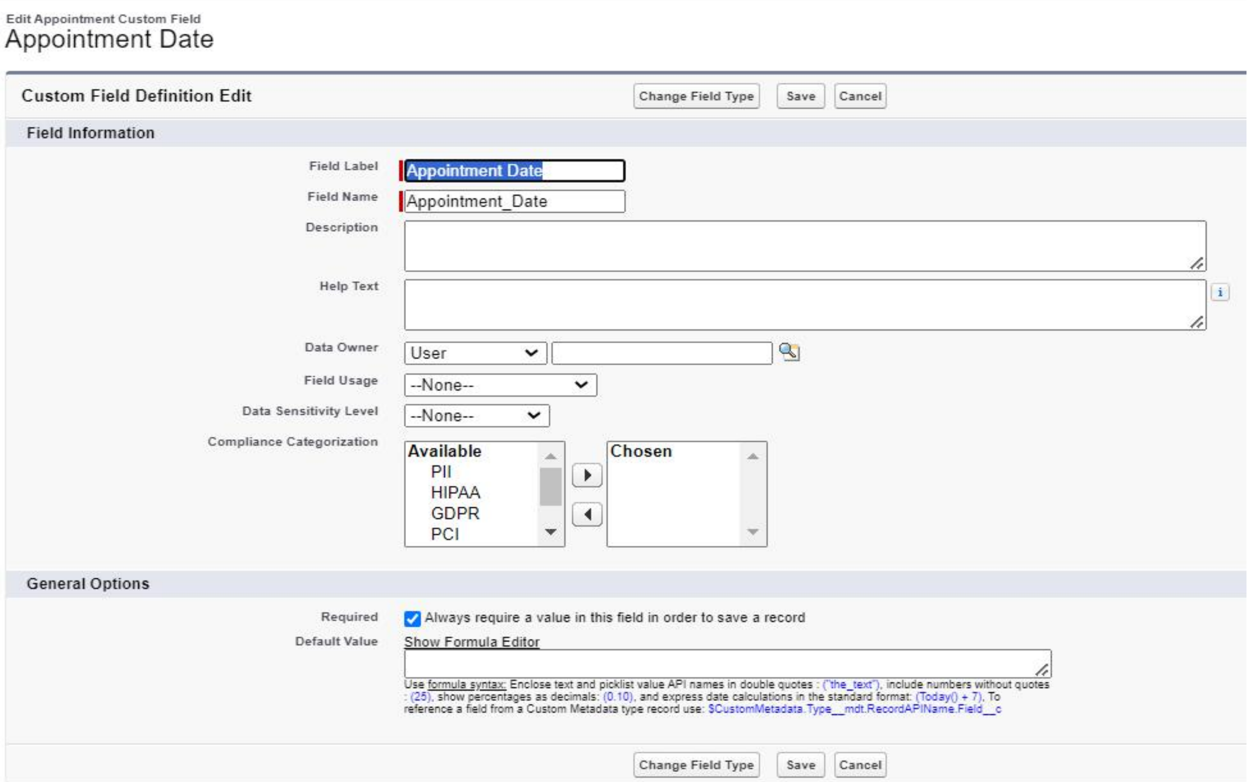
- Service Records Object:

Created a formula field labeled 'Service Date' to automatically populate with the created date of the record.

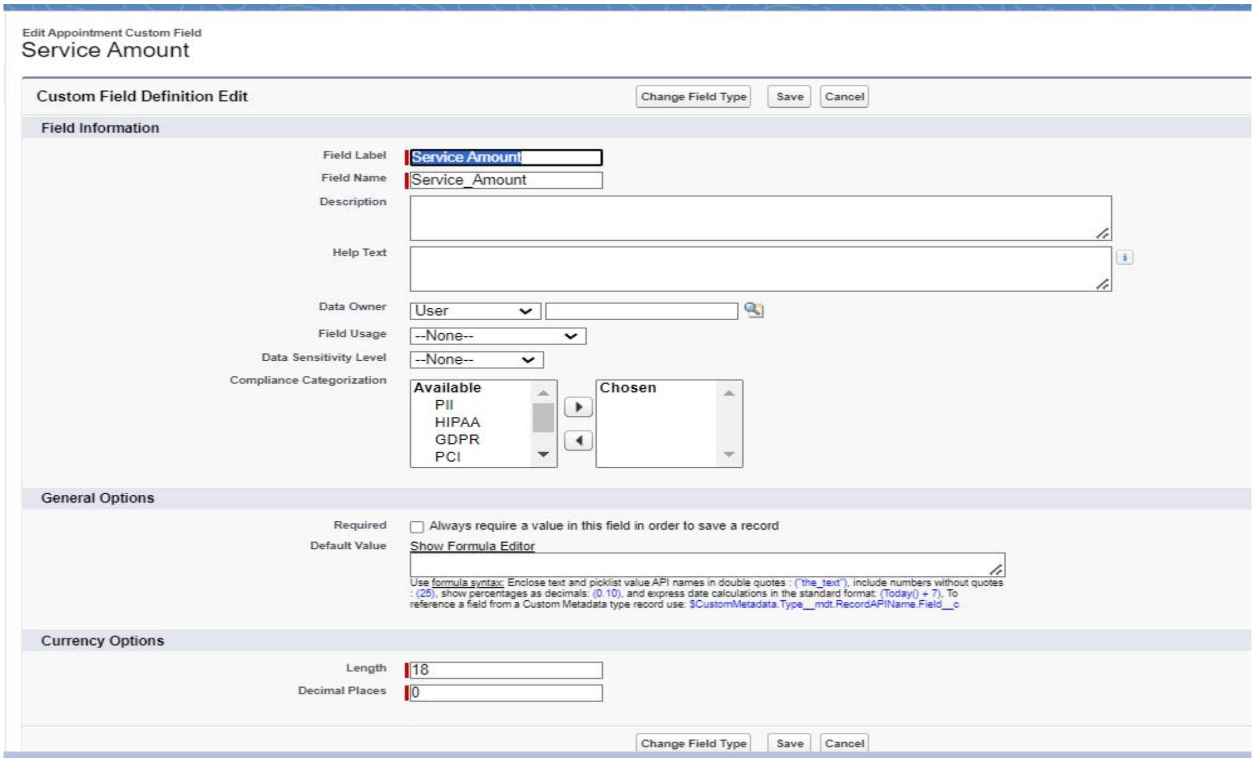


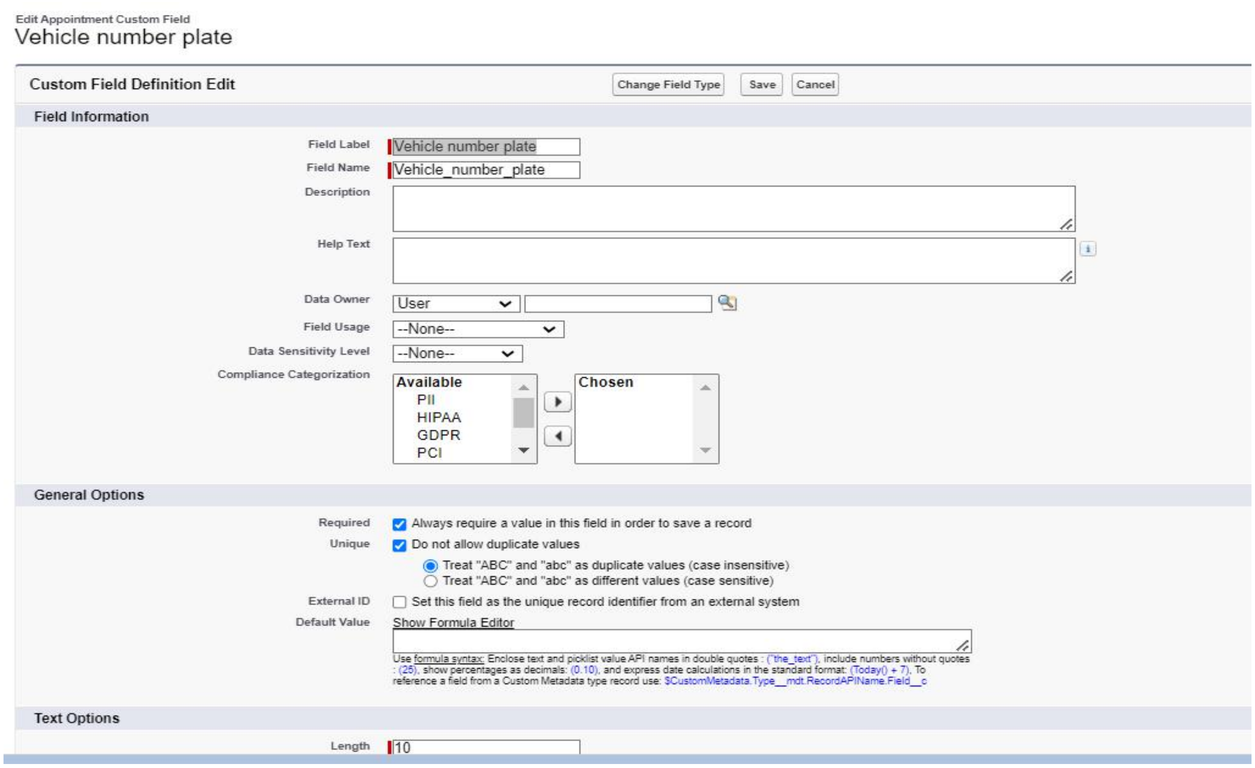


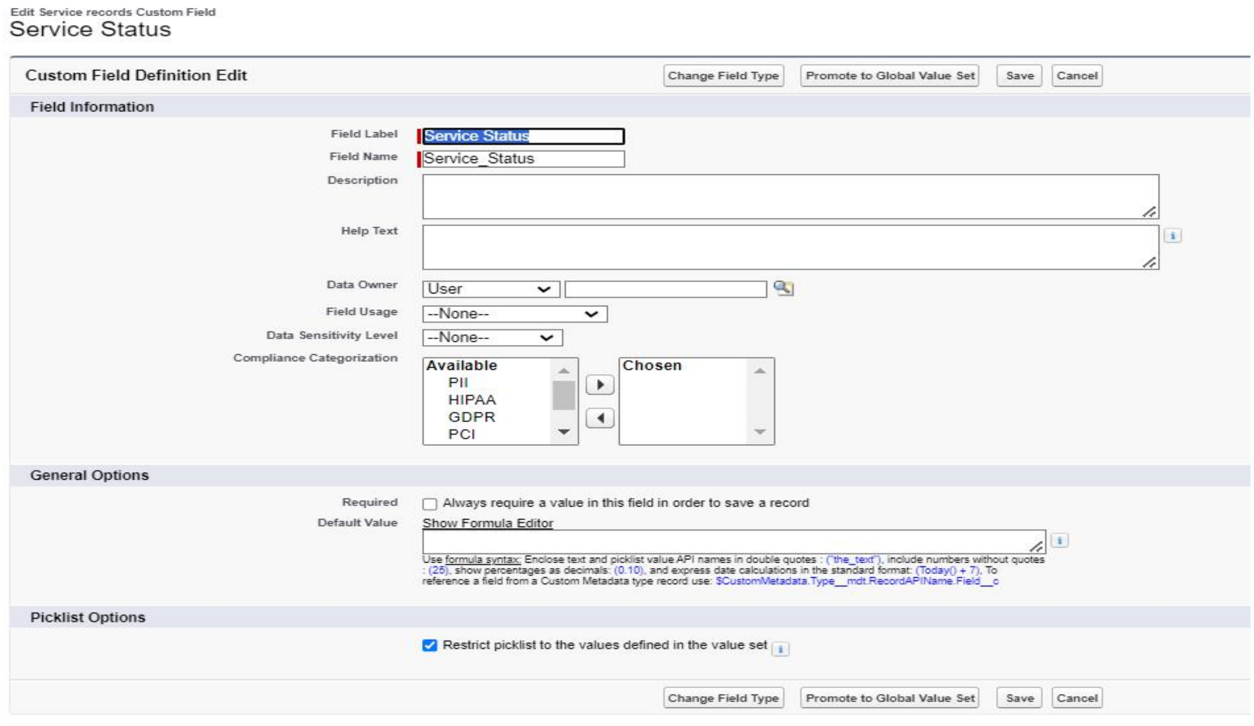
Edit Appointment Custom Field
Maintenance service
Custom Field Definition Edit
Change Field Type
Save
Cancel
Field Information
Field Label
Maintenance service
Field Name
Maintenance_service
Description
Help Text
Data Owner
User
Field Usage
--None--
Data Sensitivity Level
--None--
Compliance Categorization
Available
PII
HIPAA
GDPR
PCI
Chosen
General Options
Default Value
Checked
Unchecked
Change Field Type
Save
Cancel

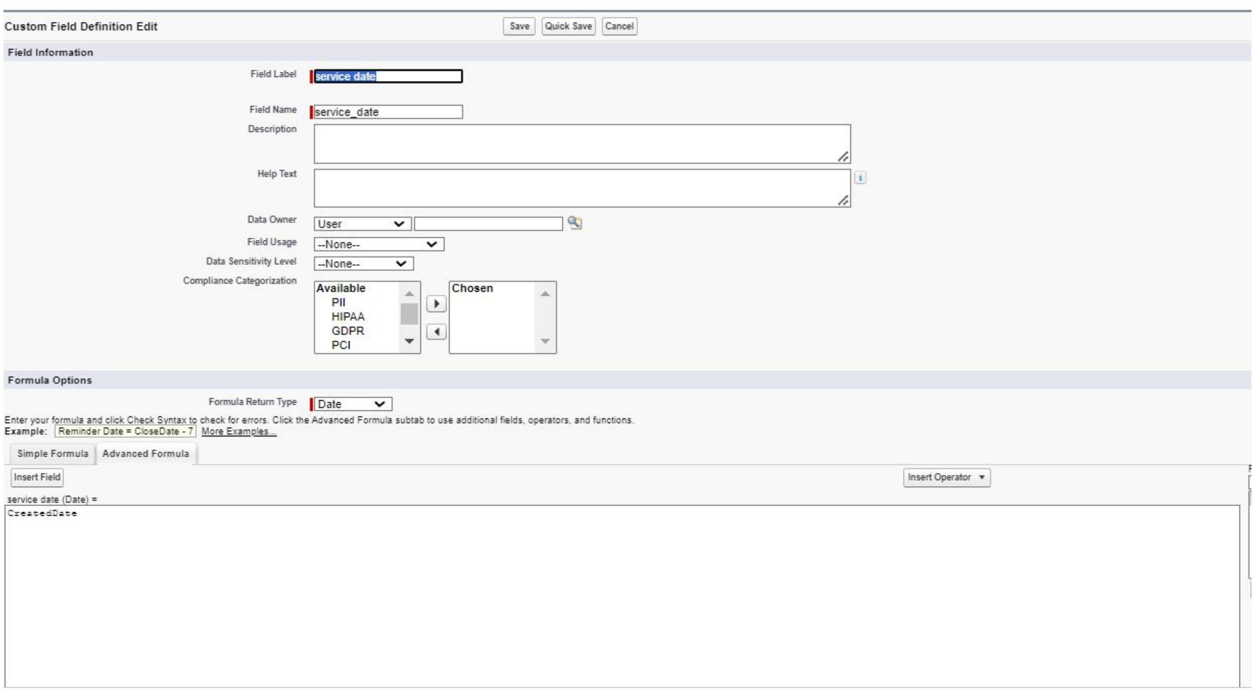


Edit Appointment Custom Field
Appointment Date
Custom Field Definition Edit
Change Field Type
Save
Cancel
Field Information
Field Label
Appointment Date
Field Name
Appointment_Date
Description
Help Text
Data Owner
User
Field Usage
--None--
Data Sensitivity Level
--None--
Compliance Categorization
Available
PII
HIPAA
GDPR
PCI
Chosen
General Options
Required
Always require a value in this field in order to save a record
Default Value
Show Formula Editor
Use formula syntax: Enclose text and picklist value API names in double quotes : ("the_text"), include numbers without quotes : (25), show percentages as decimals: (0.10), and express date calculations in the standard format: (Today() + 7). To reference a field from a Custom Metadata type record use: $CustomMetadata.Type__mdt.RecordAPIName.Field__c
Change Field Type
Save
Cancel


Edit Appointment Custom Field
Service Amount
Custom Field Definition Edit
Change Field Type
Save
Cancel
Field Information
Field Label
Service Amount
Field Name
Service_Amount
Description
Help Text
Data Owner
User
Field Usage
--None--
Data Sensitivity Level
--None--
Compliance Categorization
Available
PII
HIPAA
GDPR
PCI
Chosen
General Options
Required
Always require a value in this field in order to save a record
Default Value
Show Formula Editor
Currency Options
Length
18
Decimal Places
0
Change Field Type
Save
Cancel


Edit Appointment Custom Field
Vehicle number plate
Custom Field Definition Edit
Change Field Type
Save
Cancel
Field Information
Field Label
Vehicle number plate
Field Name
Vehicle_number_plate
Description
Help Text
Data Owner
User
Field Usage
--None--
Data Sensitivity Level
--None--
Compliance Categorization
Available
PII
HIPAA
GDPR
PCI
Chosen
General Options
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 duplicate values (case insensitive)
Treat "ABC" and "abc" as different values (case sensitive)
External ID
Set this field as the unique record identifier from an external system
Default Value
Show Formula Editor
Text Options
Length
10


Edit Service records Custom Field
Service Status
Custom Field Definition Edit
Change Field Type
Promote to Global Value Set
Save
Cancel
Field Information
Field Label
Service Status
Field Name
Service_Status
Description
Help Text
Data Owner
User
Field Usage
--None--
Data Sensitivity Level
--None--
Compliance Categorization
Available
PII
HIPAA
GDPR
PCI
Chosen
General Options
Required
Always require a value in this field in order to save a record
Default Value
Show Formula Editor
Use formula syntax: Enclose text and picklist value API names in double quotes : ("the_text"), include numbers without quotes : (25), show percentages as decimals: (0.10), and express date calculations in the standard format: (Today() + 7). To reference a field from a Custom Metadata type record use: $CustomMetadata.Type__mdt.RecordAPIName.Field__c
Picklist Options
Restrict picklist to the values defined in the value set
Change Field Type
Promote to Global Value Set
Save
Cancel


Custom Field Definition Edit
Save
Quick Save
Cancel
Field Information
Field Label
service date
Field Name
service_date
Description
Help Text
Data Owner
User
Field Usage
--None--
Data Sensitivity Level
--None--
Compliance Categorization
Available
PII
HIPAA
GDPR
PCI
Chosen
Formula Options
Formula Return Type
Date
Enter your formula and click Check Syntax to check for errors. Click the Advanced Formula subtab to use additional fields, operators, and functions.
Example: Reminder Date = CloseDate - 7 More Examples...
Simple Formula
Advanced Formula
Insert Field
Insert Operator
service date (Date) =
CreatedDate

Task 6

Validation Rules

6.1 Validation Rule for Appointment Object

- Rule Name: Vehicle

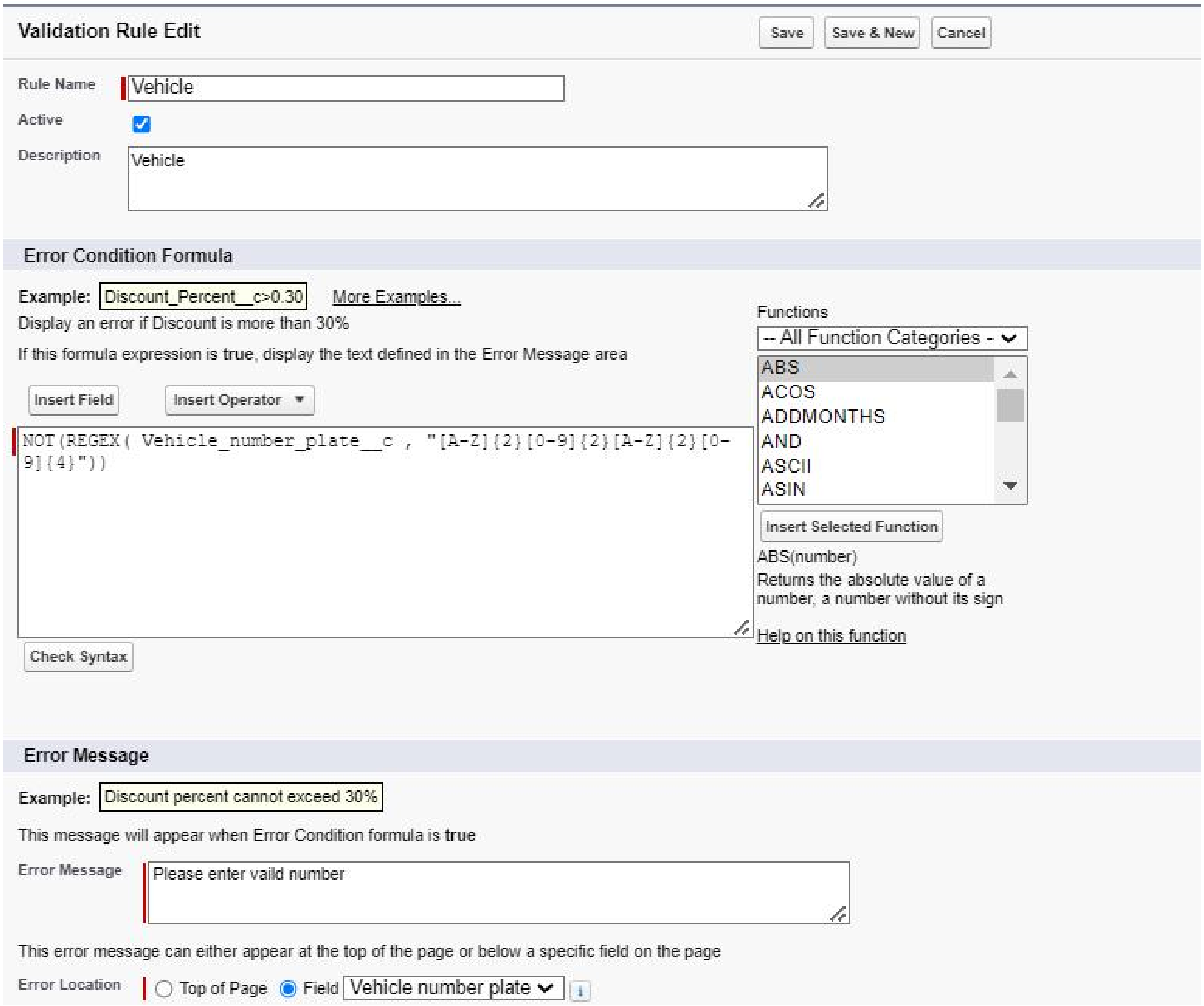
- Purpose: Ensure that the vehicle number plate follows a specific format.

- Error Condition Formula:

  NOT(REGEX(Vehicle_number_plate__c , '[A-Z]{2}[0-9]{2}[A-Z]{2}[0-9]{4}'))

- Error Message: 'Please enter a valid number'

- Error Location: Field - Vehicle number plate

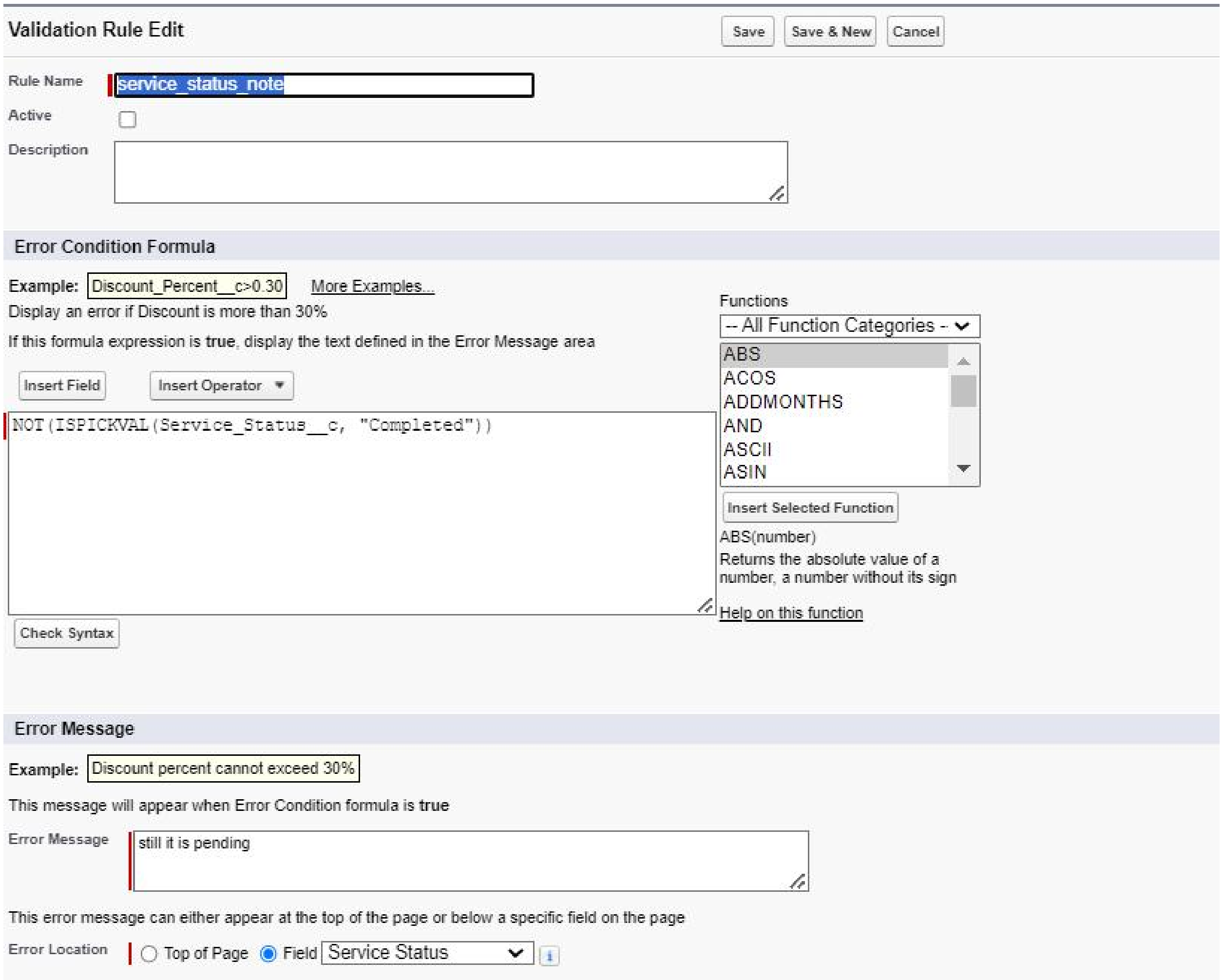


6.2 Validation Rule for Service Records Object

- Rule Name: service_status_note
- Purpose: Ensure that the service status is marked as 'Completed' before the record can be saved.
- Error Condition Formula:

NOT(ISPICKVAL(Service_Status__c, 'Completed'))

- Error Message: ‘Still it is pending‘
- Error Location: Field - Service status



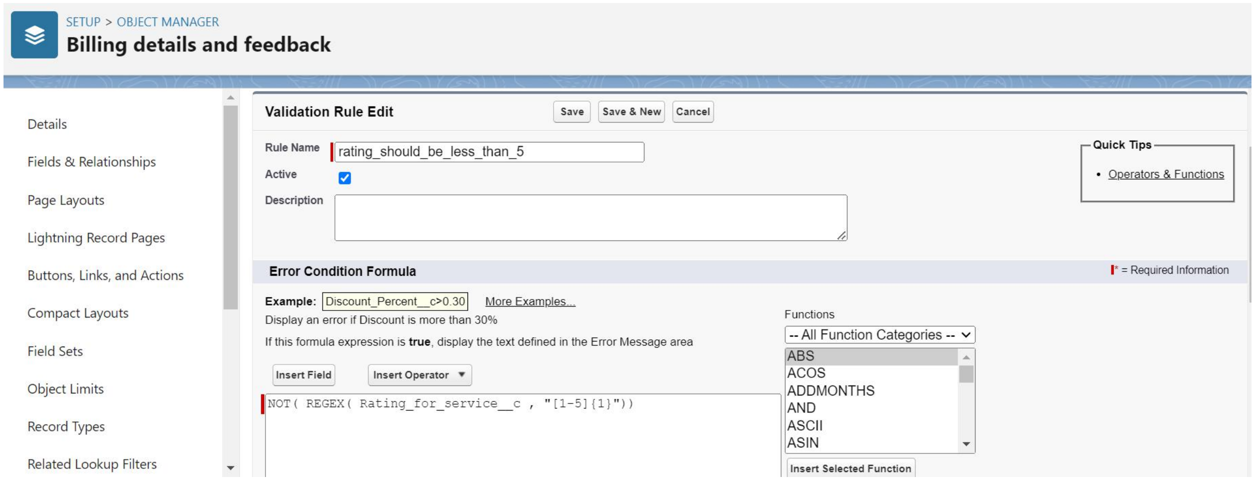
6.3 Validation Rule for Billing Details and Feedback Object

- Rule Name: rating_should_be_less_than_5
- Purpose: Ensure that the rating for service is between 1 and 5.
- Error Condition Formula:

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

- Error Message: ‘Rating should be from 1 to 5‘

- Error Location: Field - Rating for Service



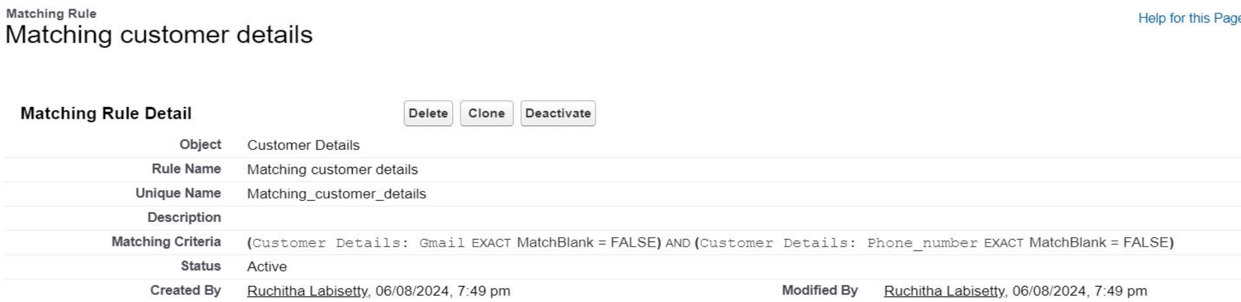
Task 7

Duplicate Rules

7.1 Creating a Matching Rule for Customer Details Object

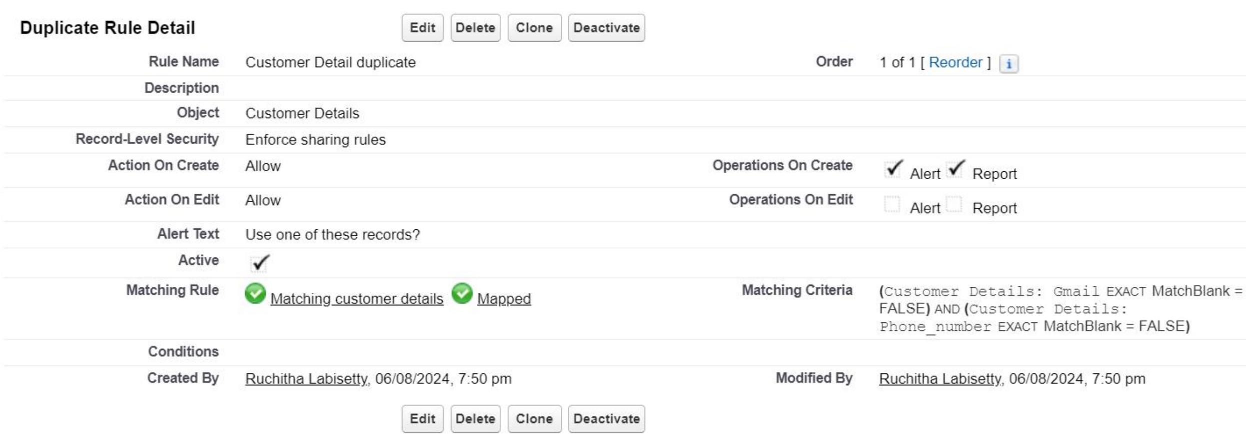
1. **Rule Name**: Matching Customer Details

    - **Object**: Customer Details
    - **Matching Criteria**:
        - **Gmail**: Exact match
        - **Phone Number**: Exact match
    - **Activation**: Once the rule is saved, it is activated to start matching customer details based on the specified criteria.



7.2 Creating a Duplicate Rule for Customer Details Object

1. **Rule Name**: Customer Detail Duplicate
   - **Object**: Customer Details
   - **Matching Rule**: Utilizes the previously created 'Matching Customer Details' rule.
   - **Activation**: After saving the duplicate rule, it is activated to begin identifying duplicate customer details based on the matching criteria.



Task 8

Profiles

8.1 Creating the Manager Profile

1. **Profile Cloning**:

    - Clone the Standard User profile to create a new profile named 'Manager.'

2. **Custom App Settings**:

    - Set the Garage Management Application as the default custom app for the Manager profile.

3. **Custom Object Permissions**:

    - Grant access permissions for the following objects:

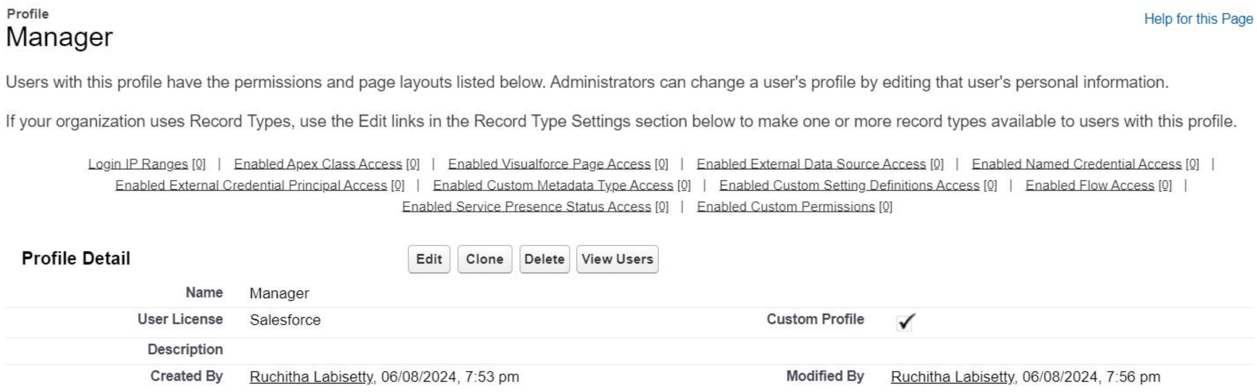
        - Appointments
        - Billing Details and Feedback
        - Service Records
        - Customer Details

4. **Session Timeout**:

    - Configure the session timeout to occur after 8 hours of inactivity.

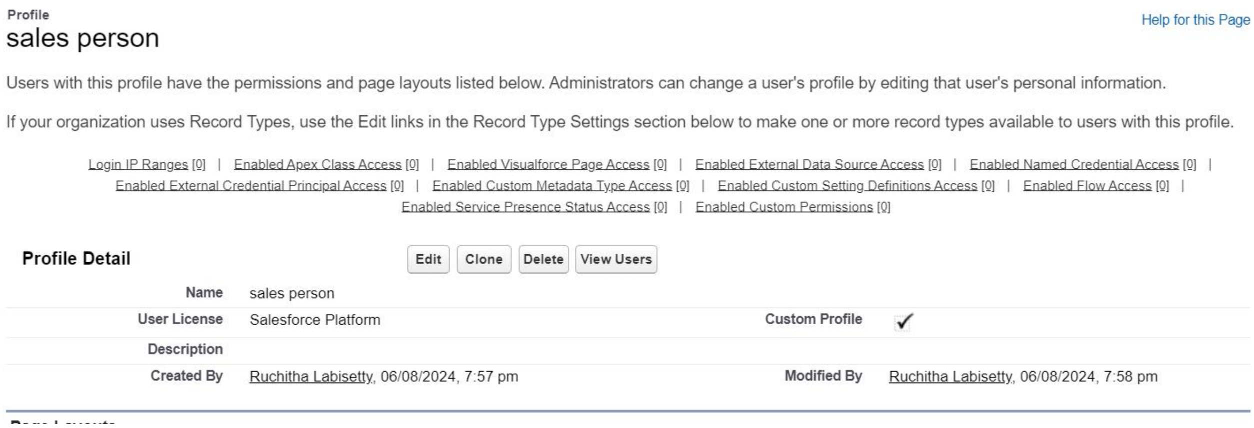
5. **Password Policies**:

    - Set user passwords to never expire.
    - Establish a minimum password length of 8 characters.



8.2 Creating the Sales Person Profile

1. **Profile Cloning**:
   - Clone the Salesforce Platform User profile to create a new profile named 'Sales Person.'
2. **Custom App Settings**:
   - Set the Garage Management Application as the default custom app for the Sales Person profile.
3. **Custom Object Permissions**:
   - Grant access permissions for the following objects:
     - Appointments
     - Billing Details and Feedback
     - Service Records
     - Customer Details

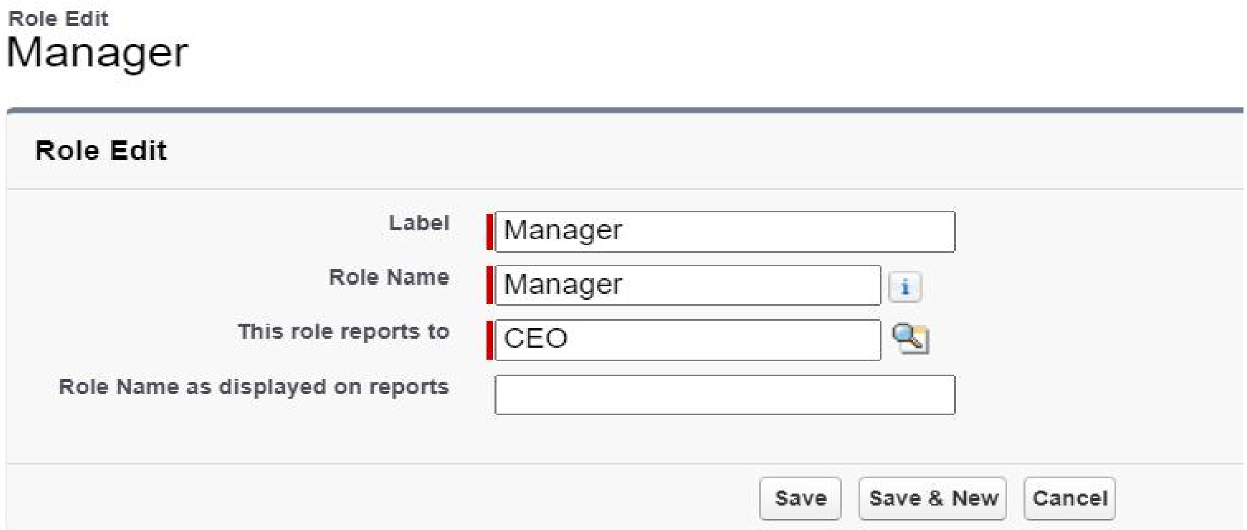


Task 9

Role And Role Hierarch

9.1 Creating the Manager Role

1. **Access Roles Setup**:
    - Navigate to the roles setup section in Salesforce.
2. **Expand Role Hierarchy**:
    - Expand all roles and add a new role under the appropriate superior role.
3. **Define Role Details**:
    - Label the new role as ‘Manager‘ and save the configuration.



9.2 Creating Sales Person Role

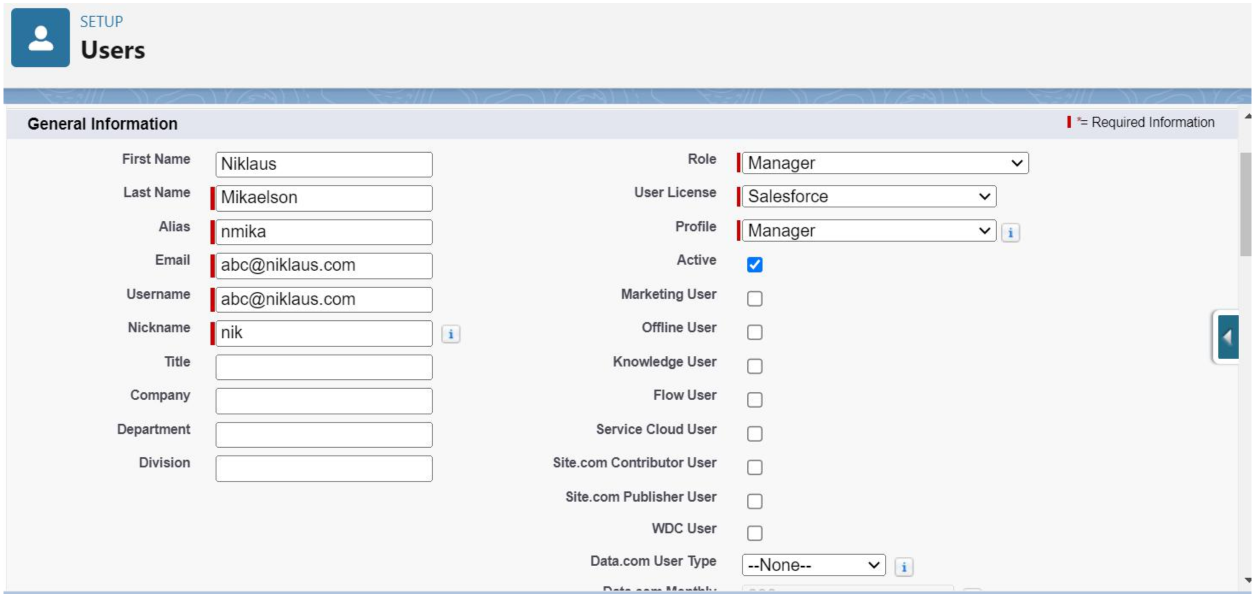
1. **Access Roles Setup**:
   - Navigate to the roles setup section in Salesforce.
2. **Add Role Under Manager**:
   - Expand the CEO role and add a new role under the Manager role.
3. **Define Role Details**:
   - Label the new role as 'Sales Person' and save the configuration.

Task 10

Users

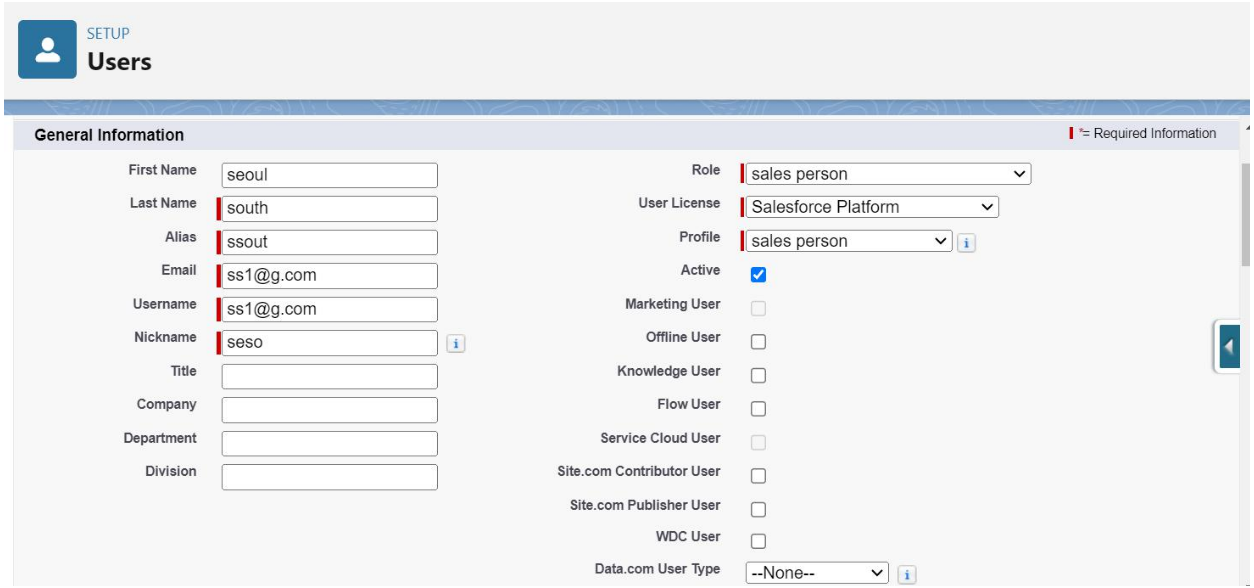
10.1 Creating a Manager User

1. **Access User Setup**:
   - Navigate to the users section in Salesforce setup.
2. **New User Details**:
   - Fill in the required fields with the user's details, such as first name, last name, alias, email, username, and nickname.
   - Assign the role of 'Manager.'
   - Set the user license to 'Salesforce.'
   - Assign the profile of 'Manager.'
3. **Save the Configuration**:
   - Save the user details to create the new Manager user.



10.2 **Creating Sales Person Users**

1. **Access User Setup**:
   - Navigate to the users section in Salesforce setup.
2. **New User Details**:
   - Fill in the required fields with the user's details.
   - Assign the role of 'Sales Person.'
   - Set the user license to 'Salesforce Platform.'
   - Assign the profile of 'Sales Person.'
3. **Save and Repeat**:
   - Save the user details and repeat the process to create at least three users with the Sales Person role and profile.

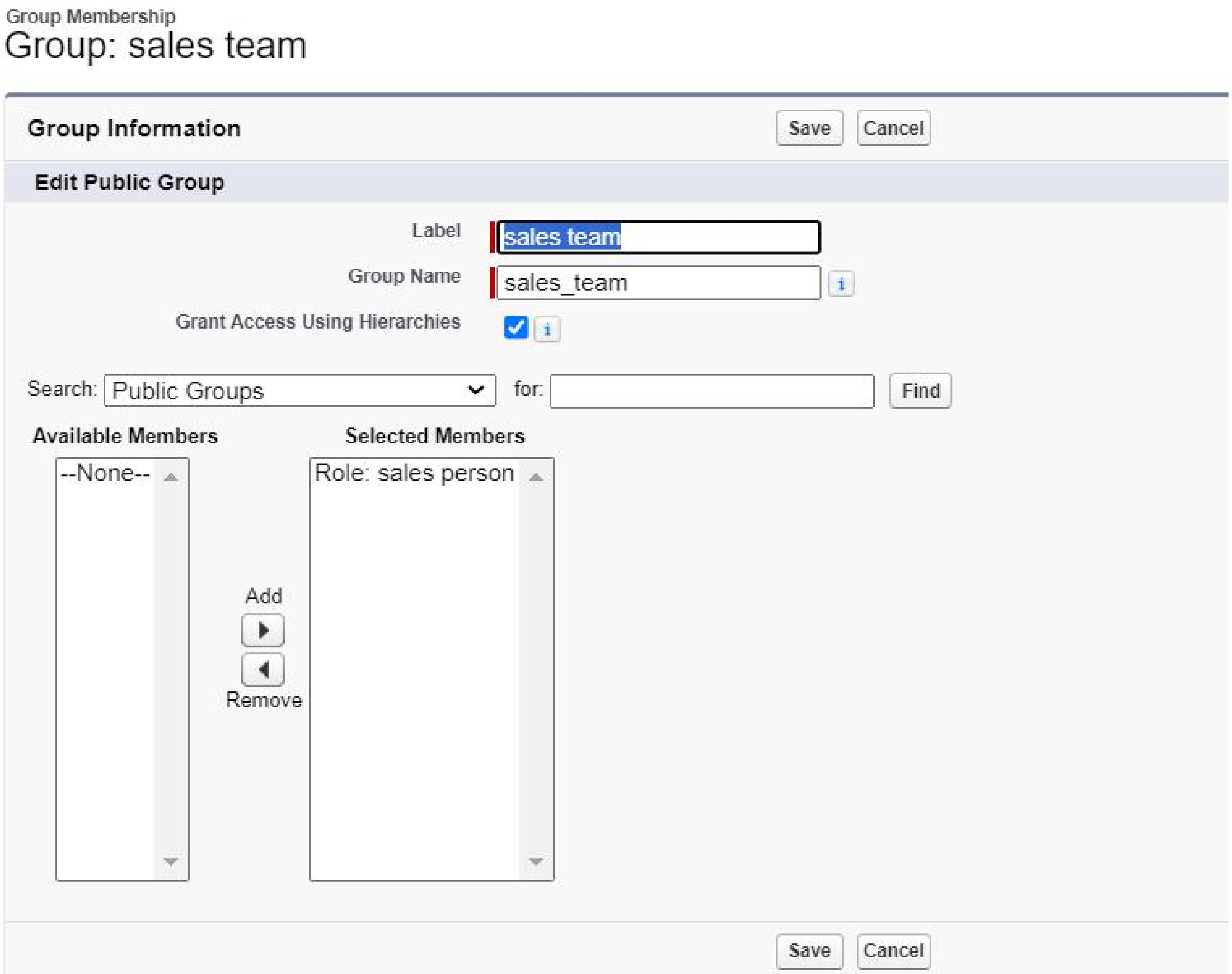


Task 11

Public Groups

11.1 Creating the Sales Team Public Group

1. **Access Public Groups Setup**:
   - Navigate to the public groups section in Salesforce setup.
2. **Define Group Details**:
   - Provide a label for the new group, such as 'Sales Team.' The group name is automatically generated based on the label.
3. **Add Members**:
   - Search for roles in the available member section.
   - Select the 'Sales Person' role and add it to the selected members list.
4. **Save the Group**:
   - Save the new public group configuration.



Task 12

Sharing Settings

12.1 Configuring Sharing Settings for Service Records

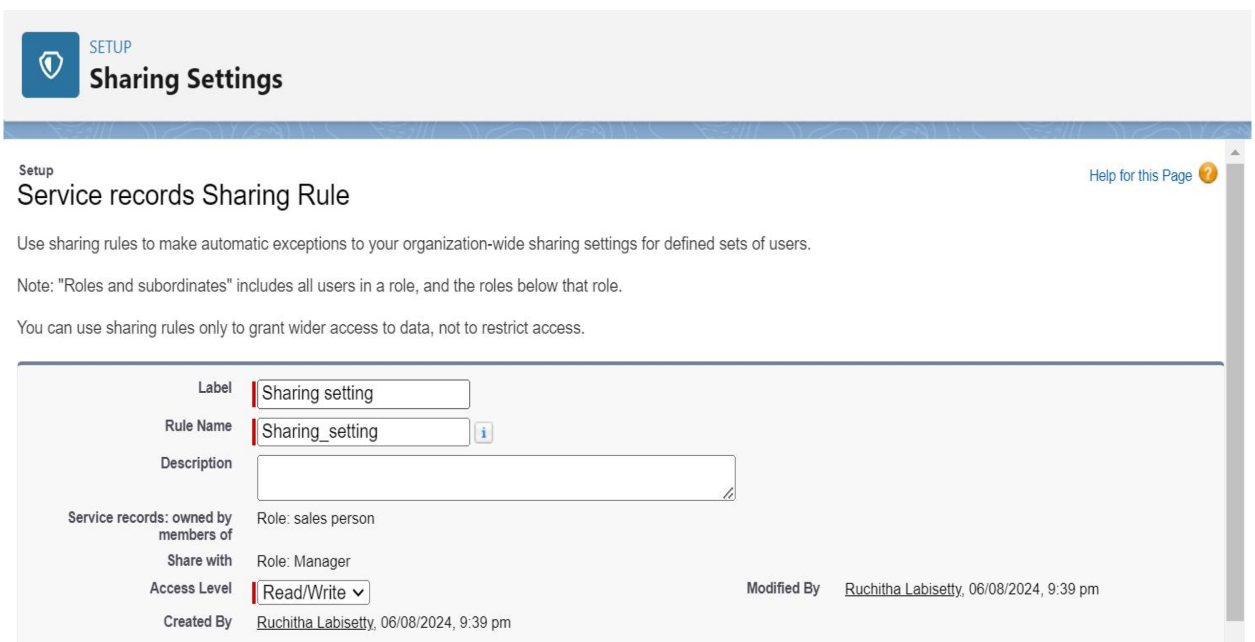
1. **Access Sharing Settings**:

    - Navigate to the Sharing Settings section in Salesforce setup.

2. **Set Object-Wide Default (OWD)**:

- Change the OWD setting for the Service Records object to ‘Private‘ to restrict access to records by default.

3. **Create Sharing Rules**:
   - Define a new sharing rule to grant access to specific roles.
4. **Define Sharing Rule Details**:
   - Label the sharing rule appropriately, such as ‘Sharing Setting.‘
   - Specify the members of the role to be shared, selecting ‘Sales Person.‘
   - Determine who will receive the access, selecting ‘Manager.‘
   - Set the access level to ‘Read/Write.‘
5. **Save and Refresh**:
   - Save the sharing rule and refresh the settings to apply changes.



Task 13

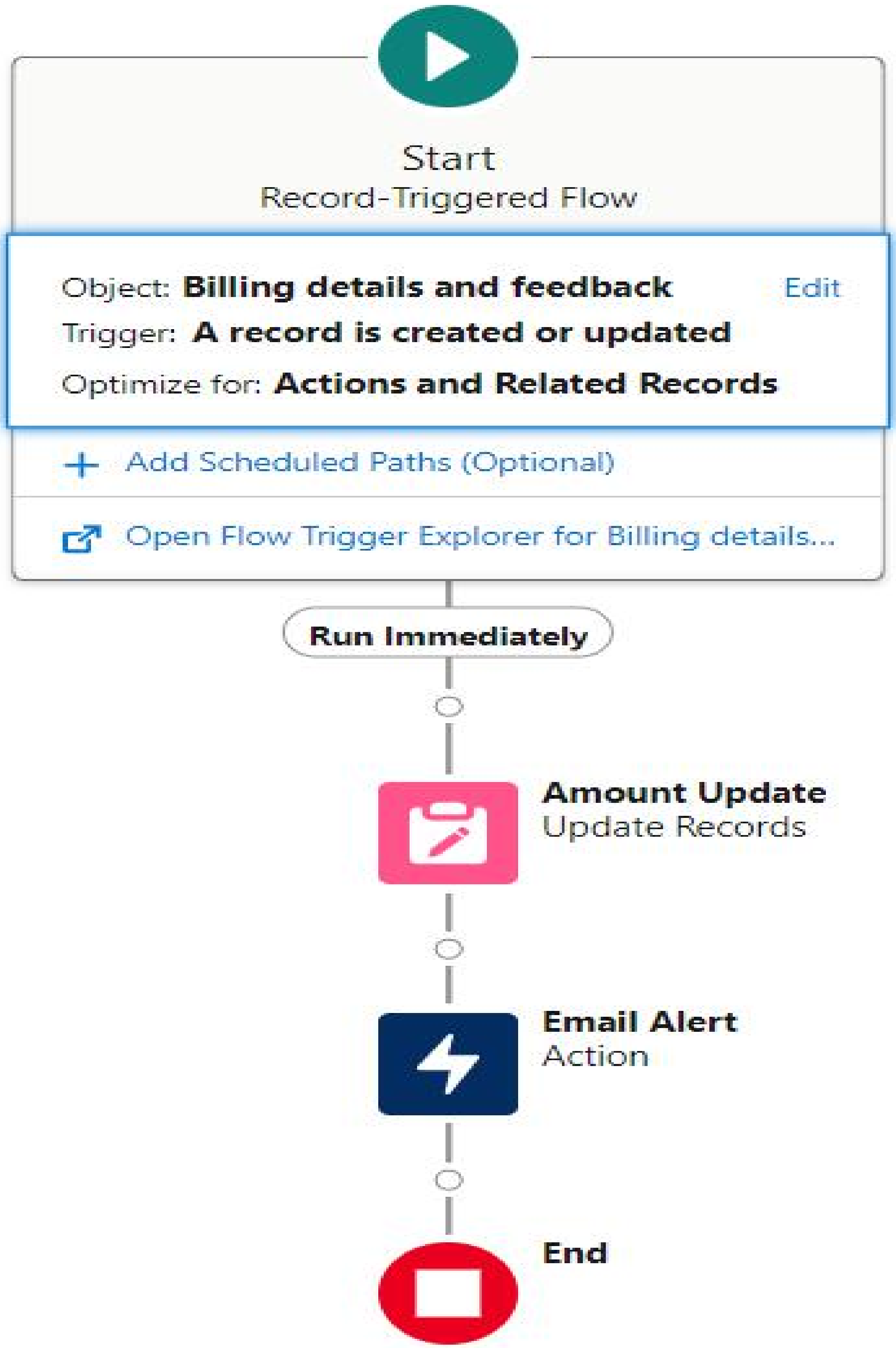
Flows

13.1 Creating the Flow

1. **Initiate Flow Creation**:
   - Access the Flow setup page and start a new Record-Triggered Flow.
2. **Configure Flow Trigger**:
   - Select ‘Billing Details and Feedback‘ as the object.
   - Set the flow to trigger when a record is created or updated.
   - Choose ‘Actions and Related Records‘ to optimize the flow.
3. **Add Update Records Element**:
   - Label the update element as “Amount Update.”
   - Configure it to update records where the field Payment_Status__c equals ‘Completed.‘
   - Set the field value for Payment_Paid__c to the service amount from the related Appointment record.
4. **Create and Configure Email Alert**:
   - Define a new resource as a text template to construct the email body.
   - Use rich text format for clarity and include placeholders for dynamic content.
   - The email body should express gratitude for the payment and include the payment amount.
5. **Add Email Action**:
   - Insert an ‘Action‘ element to send an email.

- Utilize the previously created text template for the email body.
- Set the recipient address dynamically from the related record.
- Define the subject of the email as ‘Thank You for Your Payment - Garage Management.‘

6. **Save and Activate the Flow**:
   - Save the flow with an appropriate label and auto-populated API name.
   - Activate the flow to make it operational.


Start
Record-Triggered Flow
Object: Billing details and feedback
Edit
Trigger: A record is created or updated
Optimize for: Actions and Related Records
Add Scheduled Paths (Optional)
Open Flow Trigger Explorer for Billing details...
Run Immediately
Amount Update
Update Records
Email Alert
Action
End

Task 14

Apex Triggers

14.1 Apex Handler: AmountDistributionHandler

Purpose: This Apex class calculates and updates the service amount based on selected services in the Appointment__c object.

Class Definition:

- **Class Name**: AmountDistributionHandler
- **Method**: amountDist
- **Parameters**: List<Appointment__c| – A list of appointment records to process.

Functionality:

- Iterates through the list of appointments.
- Updates the Service_Amount__c field based on the combination of services selected:
    - All three services: 10000
    - Maintenance and Repairs: 5000
    - Maintenance and Replacement Parts: 8000
    - Repairs and Replacement Parts: 7000
    - Maintenance only: 2000
    - Repairs only: 3000
    - Replacement Parts only: 5000

Trigger: AmountDistribution

Purpose: This trigger invokes the AmountDistributionHandler to update the Service_Amount__c field whenever an appointment record is inserted or updated.

Trigger Definition:

- **Trigger Name**: AmountDistribution
- **sObject**: Appointment__c
- **Events**: before insert, before update

Trigger Logic:

- Checks if the trigger event is before insert or before update.
- Calls the amountDist method from AmountDistributionHandler to perform the required updates on the Appointment__c records.

Code:

```
trigger AmountDistribution on Appointment__c (before insert, before update) {

   if (trigger.isbefore && (trigger.isinsert || trigger.isupdate)) {

      AmountDistributionHandler.amountDist(trigger.new);

   }

}
```

Task 15

Reports

15.1 **Creating a Report Folder**

1. **Objective**: Organize reports by creating a dedicated folder.
2. **Process**:
   - Access the Reports tab through the app launcher.

- Click on **New Folder** and name it **'Garage Management Folder'**. The folder's unique name is automatically generated.
- Save the new folder to store and manage related reports.

15.2 **Sharing the Report Folder**

1. **Objective**: Control access to the report folder.
2. **Process**:
   - Navigate to the Reports tab and locate the **'Garage Management Folder'**.
   - Click on the dropdown menu next to the folder name and select **Share**.
   - Choose **Roles** as the sharing option, search for the **'Manager'** role, and assign **View** access.
   - Confirm the sharing settings and finalize by clicking **Done**.

15.3 **Creating a Custom Report Type**

1. **Objective**: Define a report type to include multiple related objects.
2. **Process**:
   - Access **Setup** and search for **Report Types**.
   - Click on **New Custom Report Type**.
   - Set the **Primary Object** to **Customer Details**.
   - Name the report type **'Service Information'** and confirm the auto-populated name.
   - Categorize under **Other Reports** and set the deployment status to **Deployed**.
   - Click **Next** to define related objects:
     - **Appointment Object**
     - **Service Records**

- ○ **Billing Details and Feedback**

- ■ Save the new report type.

15.4 **Creating a Report**

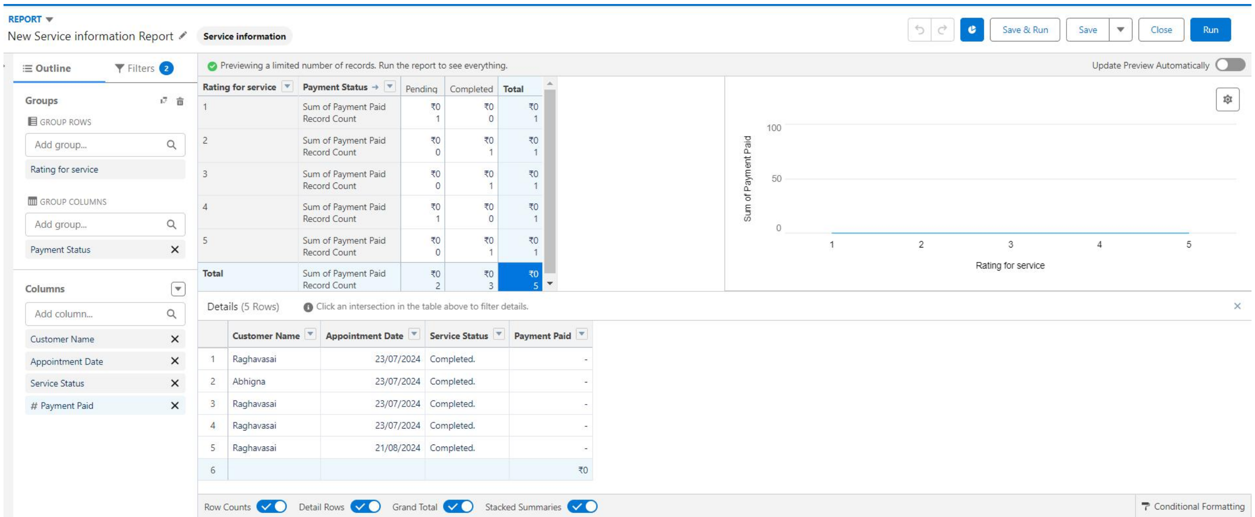
1. **Objective**: Generate a detailed report based on the created report type.

2. **Preparation**:

   - ■ Ensure to create at least 10 records for each object to ensure comprehensive data representation.

3. **Process**:

   - ■ Access the Reports tab and click on **New Report**.
   - ■ Select the category **Other Reports** and choose **Service Information** as the report type.
   - ■ Start the report builder and configure the fields:
     - ○ **Columns**: Customer Name, Appointment Date, Service Status, Payment Paid.
     - ○ **Group Rows**: Rating for Service, Payment Status.
   - ■ Add a Line Chart to visualize the data.
   - ■ Save the report with the name **'New Service Information Report'**, ensuring the unique name is auto-populated and the report is saved in the **'Garage Management Folder'**.



Task 16

Dashboards

16.1 Creating a Dashboard Folder

1. **Objective**: Organize dashboards by creating a specific folder.
2. **Process**:
   - Navigate to the **Dashboards** tab via the app launcher.
   - Click **New Folder** and name it **'Service Rating Dashboard'**. The unique name is auto-generated.
   - Save the folder to group and manage related dashboards efficiently.

Sharing the Dashboard Folder

1. **Objective**: Set access permissions for the dashboard folder.
2. **Process**:
   - Locate the **'Service Rating Dashboard'** folder in the Dashboards tab.
   - Click on the dropdown menu for the folder and select **Share**.

- Assign appropriate permissions based on user roles, ensuring that the relevant team members have access to view or edit the dashboards as needed.
- Confirm the sharing settings to finalize.

16.2 **Creating a Dashboard**

1. **Objective**: Build a dashboard to visualize data and insights.
2. **Process**:
    - Access the **Dashboards** tab and click **New Dashboard**.
    - Enter a name for the dashboard and select the **'Service Rating Dashboard'** folder created earlier.
    - Click **Create** to start building the dashboard.
    - Add a new component by selecting **Add Component**.
    - Choose a report to base the dashboard on, and select a **Line Chart** for visualization.
    - Customize the chart theme as desired.
    - Click **Add** to include the component in the dashboard, then click **Save** and **Done**.

Subscribing to the Dashboard

1. **Objective**: Automate the delivery of dashboard updates.
2. **Process**:
    - After saving the dashboard, click **Subscribe** at the top right.
    - Set the subscription frequency to **Weekly**.
    - Choose **Monday** as the day for the report to be sent.
    - Click **Save** to activate the subscription and ensure regular updates.


New Service information Report
Payment Status
Pending
Completed
₹8K
₹6K
₹4K
₹2K
₹0
Sum of Payment Paid
1
2
3
4
5
Rating for service
View Report (New Service information Report)