UNNAMALAI INSTITUTE OF TECHNOLOGY

Suba Nagar, Kovilpatti - 628502

Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai

NACC Accredited, ISO 9001:2015 Certified Institution

Project Viva-Voce Examination

The Viva-Voce Examination of this Project work submitted by

M. A. PADMA KUMAR (953322104035)

D. SUDALAI RAJA (953322104056)

M. PIRAVIN KUMAR (953322104040)

B. MUTHUESWARAN (953322104029)

S.ARIKARAMURTHY (953322104005)

Was held on _____

at

UNNAMALAIINSTITUTEOFTECHNOLOGY

Internal Examiner **External Examiner**

ANNA UNIVERSITY: CHENNAI 600025

BONAFIDE CERTIFICATE

Certified that this project report **"GARAGE MANAGEMENT SYSTEM"** is the bonafide work of M.A.PADMAKUMAR (953322104035), D.SUDALAIRAJA (953322104056), M.PIRAVIN KUMAR (953322104040), B.MUTHU ESWARAN (953322104029), S.ARI KARA MURTHY (953322104005)" who carried out the project work under my supervision

SIGNATURE

Mrs.A.CHITRA, M.E.,

HEAD OF THE DEPARTMENT

Assistant Professor

Computer Science and Engineering

Unnamalai Institute of Technology

Kovilpatti-628 502

SIGNATURE

Mr.J.SETHURAM,M.E,

SUPERVISOR

Assistant Professor

Computer Science and Engineering

Unnamalai Institute of Technology

Kovilpatti-628 502

ACKNOWLEDGEMENT

We express our sincere gratitude to the Management, the Head of the Institution, **Dr.D.RAVINDHRAN** M.E.,PHD., and Head of the CSE Department **Mrs. A.CHITRA, M.E.,** Assistant Professor, for encouraging and providing all the necessary facilities to carry out this project.

Profound thanks to the Project Coordinator **Mr.J.SETHURAM, M.E,** Assistant Professor, Department of CSE for providing necessary environment and encouragement in completing this project work.

We express our heartful thanks to the guide Mr.J.SETHURAM, M.E, Assistant Professor, Department of CSE who motivated us, to carry out this project successfully.

We express our heartful thanks to all the staff members who involved directly or indirectly to make this project more purposeful

Garage Management System

Abstraction

The Garage Management System is a valuable tool for automotive repair facilities, helping them deliver top-notch service, increase operational eciency, and build lasting customer relationships. With its user-friendly interface and powerful features, GMS empowers garages to thrive in a competitive market while ensuring a seamless and satisfying experience for both customers and staff.The **Garage Management System (GMS)** is a comprehensive software solution designed to streamline and optimize the operations of automotive repair facilities, service centers, and garages. It provides an array of features tailored to meet the needs of mechanics, service advisors, and business owners, ensuring smoother workows and higher customer satisfaction.

The **Garage Management System** is a valuable tool for automotive repair facilities, helping them deliver top-notch service, increase operational eciency, and build lasting customer relationships. With its user-friendly interface and powerful features, GMS empowers garages to thrive in a competitive market while ensuring a seamless and satisfying experience for both customers and staff.The Garage Management System (GMS) is a comprehensive software solution designed to streamline and optimize the operations of automotive repair facilities, service centers, and garages. It provides an array of features tailored to meet the needs of mechanics, service advisors, and business owners, ensuring smoother workows and higher customer satisfaction.

- **Appointment Scheduling:**
  - Simplies the booking process for customers.
  - Enables staff to manage daily schedules eciently, reducing downtime and improving resource allocation.
- **Vehicle Management:**
  - Maintains detailed records of vehicles, including service history, repairs, and maintenance schedules.
  - Tracks vehicle status during servicing for better communication with customers.
- **Customer Relationship Management (CRM):**
  - Stores customer details and preferences.
  - Sends service reminders, follow-ups, and promotional offers to build loyalty.

- **Inventory and Spare Parts Management:**
  - Tracks spare parts stock levels, automates reorder processes, and prevents stockouts.
  - Ensures that mechanics always have the necessary tools and parts on hand.
- **Billing and Invoicing:**
  - Generates professional invoices quickly and accurately.
  - Supports multiple payment methods, discounts, and tax calculations.
- **Work Order Management**:
  - Creates detailed work orders with a list of tasks, estimated costs, and timelines.
  - Helps staff prioritize jobs and ensures timely completion.
- **Reporting and Analytics:**
  - Provides insights into key performance indicators like revenue, job completion rates, and customer feedback.
  - Helps identify trends and areas for improvement.

Salesforce Introduction:

Salesforce is your customer success platform, designed to help you sell, service, market, analyze, and connect with your customers.
Salesforce has everything you need to run your business from anywhere. Using standard products and features, you can manage relationships with prospects and customers, collaborate and engage with employees and partners, and store your data securely in the cloud.

Creating Developer Account:

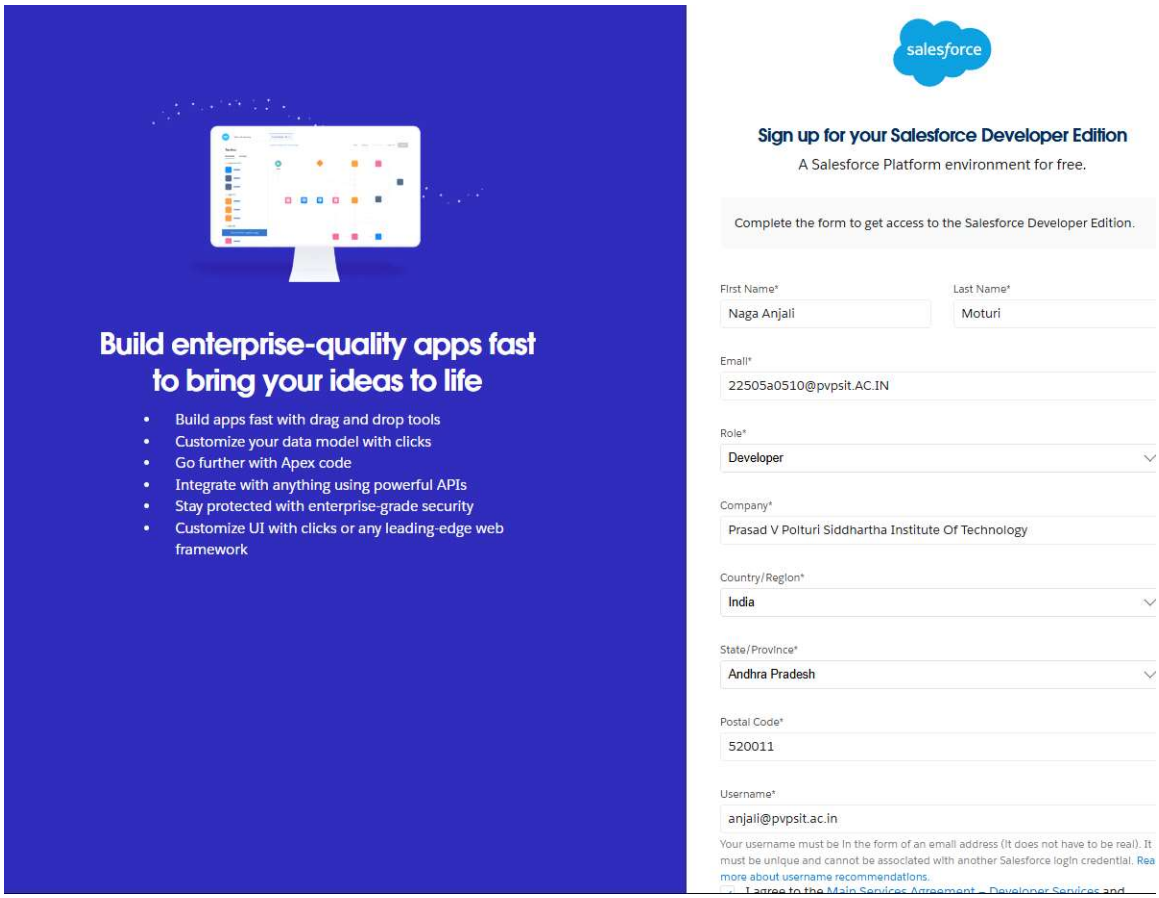
Creating a developer org in salesforce.

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 : College Name
- County : India
- Postal Code : pin code
- Username : should be a combination of your name and company

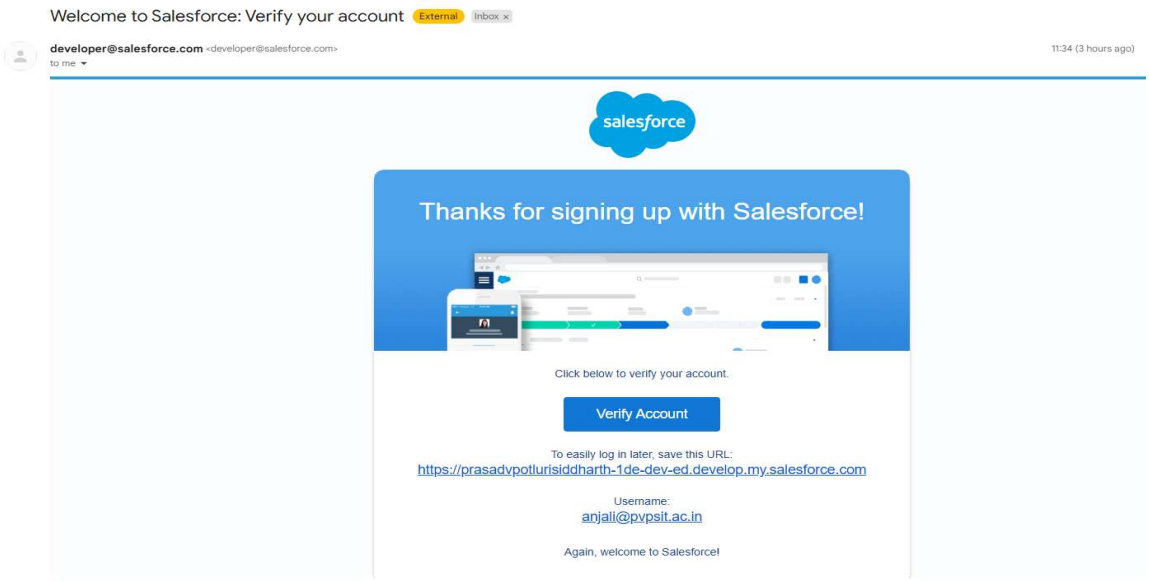
This need not be an actual email id, you can give anything in the format : username@organization.com

Click on sign me up after filling these.



Account Activation

Go to the inbox of the email that you used while signing up. Click on the verify account to activate your account.



OBJECT

Salesforce objects are database tables that permit you to store data that is specic to an organization. What are the types of Salesforce objects. Salesforce objects are of two types:

- **Standard Objects:** Standard objects are the kind of objects that are provided by salesforce.com such as users, contracts, reports, dashboards, etc.
- **Custom Objects:** Custom objects are those objects that are created by users. They supply information that is unique and essential to their organization. They are the heart of any application and provide a structure for sharing data.

Create Customer DetailsObject

To create an object: From the setup page >> Click on Object Manager >> Click on Create >> Click on Custom Object.

- Enter the label name >> Customer Details
- Plural label name >> Customer Details

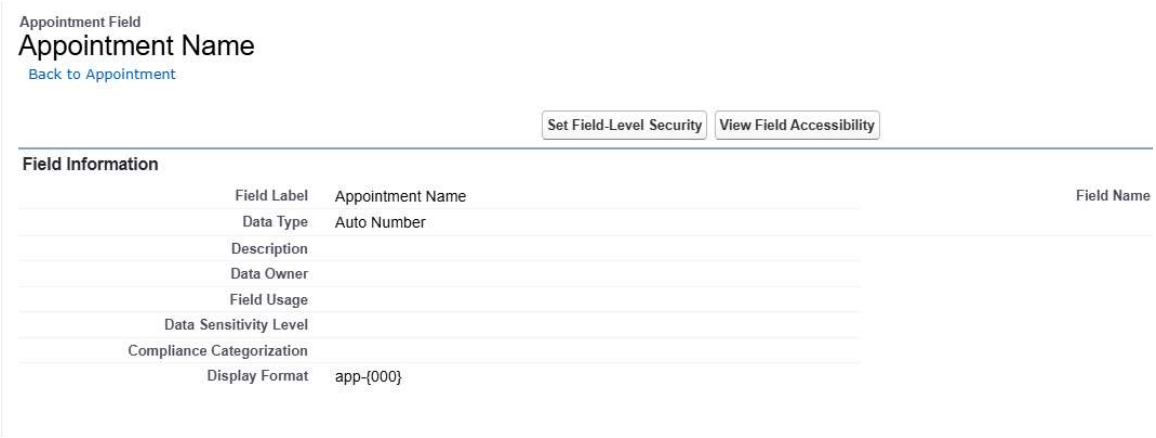
- Enter Record Name Label and Format
    - Record Name >> Customer Name
    - Data Type >> Text
- Click on Allow reports and Track Field History,
- Allow search >> Save.

Create Appointment Object

To create an object:

From the setup page >> Click on Object Manager >> Click on Create >> Click on Custom Object.

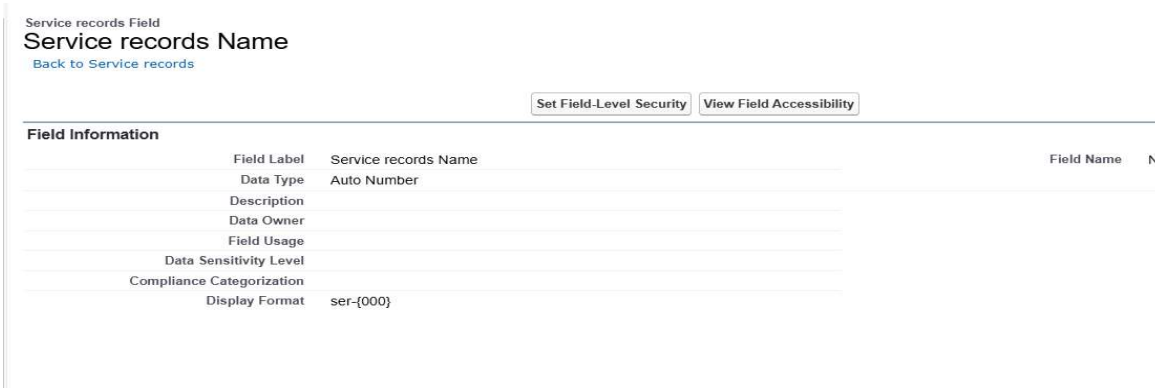
- Enter the label name >> Appointment
- Plural label name >> Appointments
- Enter Record Name Label and Format
    - Record Name >> Appointment Name
    - Data Type >> Auto Number
    - Display Format >> app-{000}
    - Starting number >> 1
- Click on Allow reports and Track Field History,
- Allow search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



Create Service records Object

To create an object: From the setup page >> Click on Object Manager >> Click on Create >> Click on Custom Object.

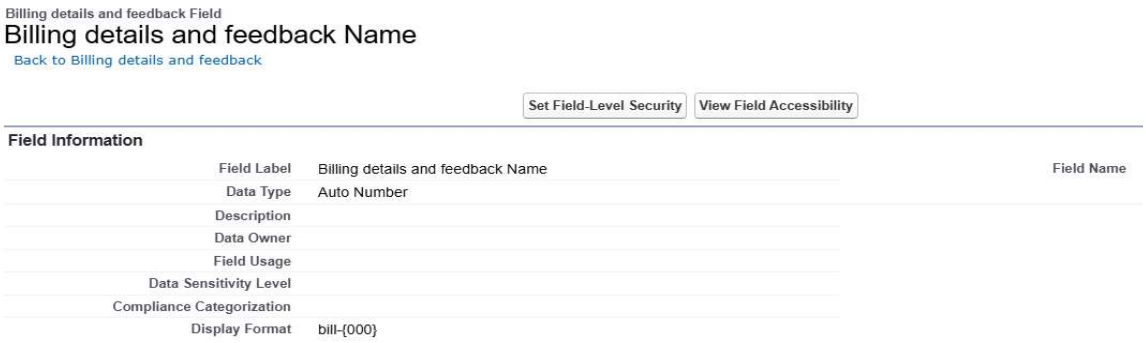
- Enter the label name >> Service records
- Plural label name >> Service records
- Enter Record Name Label and Format
    - Record Name >>Service records Name
    - Data Type >> Auto Number
    - Display Format >> ser-{000}
    - Starting number >> 1
- Click on Allow reports and Track Field History,
- Allow search >> Save.



Create Billing details and feedback Object

To create an object: From the setup page >> Click on Object Manager >> Click on Create >> Click on Custom Object.

- Enter the label name >> Billing details and feedback
- Plural label name >> Billing details and feedback
- Enter Record Name Label and Format
    - Record Name >> Billing details and feedback Name
    - Data Type >> Auto Number
    - Display Format >> bill-{000}
    - Starting number >> 1
- Click on Allow reports and Track Field History,
- Allow search >> Save.


Billing details and feedback Field
Billing details and feedback Name
Back to Billing details and feedback

Set Field-Level Security | View Field Accessibility

Field Information

Field Label	Billing details and feedback Name	Field Name
Data Type	Auto Number	
Description		
Data Owner		
Field Usage		
Data Sensitivity Level		
Compliance Categorization		
Display Format	bill-{000}	

Tabs

A tab is like a user interface that is used to build records for objects and to view the records in the objects.

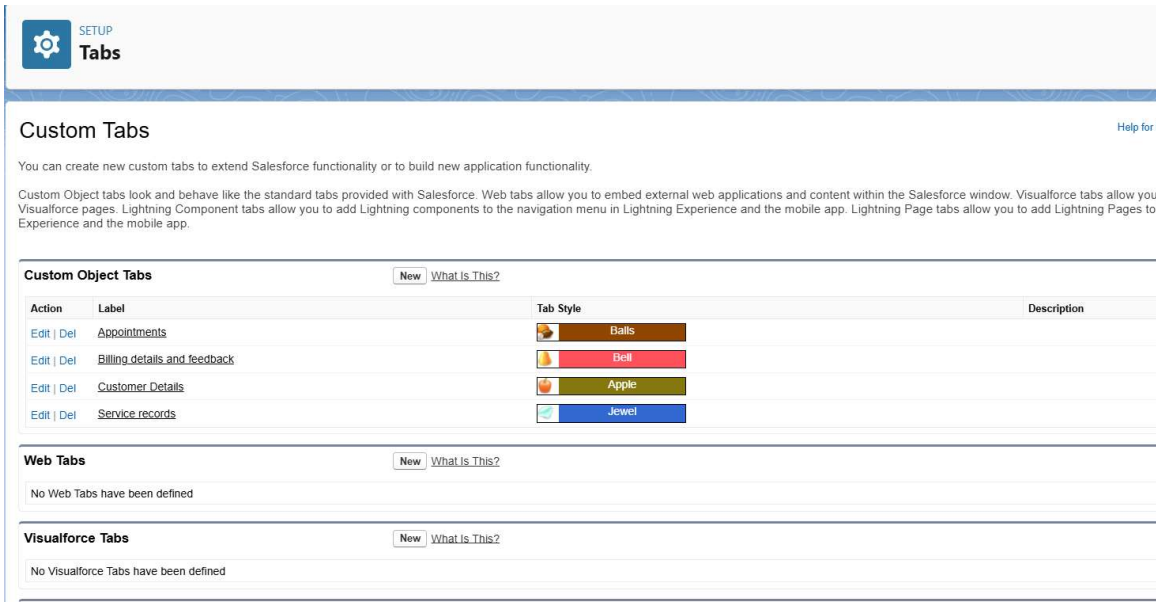
Creating a Custom Tab

To create a Tab:(Customer Details)

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

Creating Remaining Tabs

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



The Lightning App

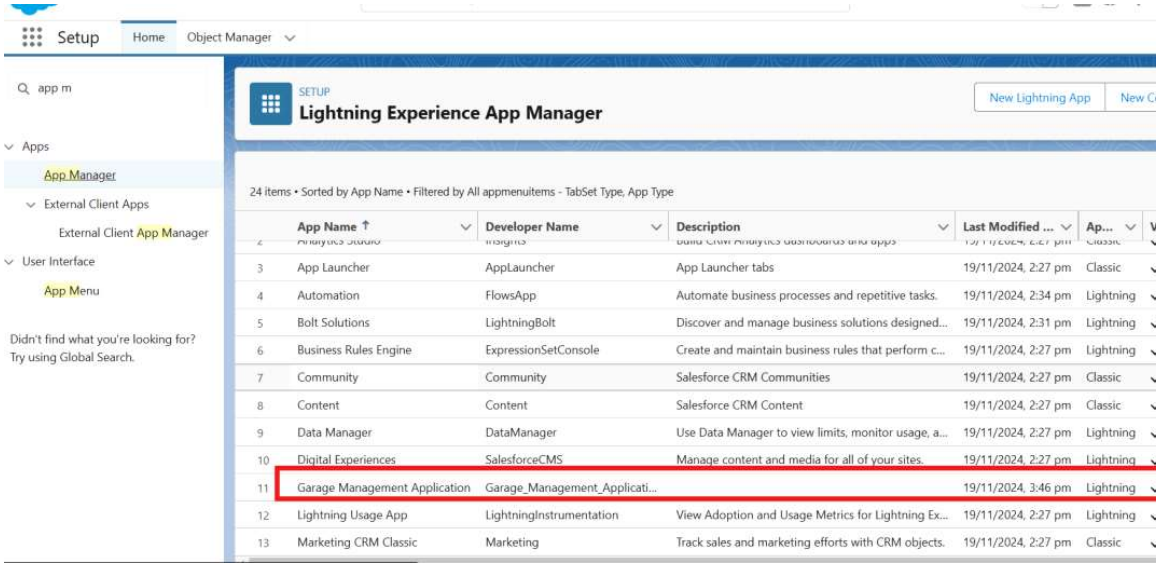
An app is a collection of items that work together to serve a particular function. In Lightning Experience, Lightning apps give your users access to sets of objects, tabs, and other items all in one convenient bundle in the navigation bar.
Lightning apps let you brand your apps with a custom colour and logo. You can even include a utility bar and Lightning page tabs in your Lightning app. Members of your org can work more eciently by easily switching between apps.

Create a Lightning App

To create a lightning app page:

- Go to setup page >> search “app manager” in quick nd >> select “app manager” >> click on New lightning App.
- 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.
- To Add Navigation Items:
- 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.

- To Add User Proles: Search proles (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 specic object. Hence, the overall searching, deletion, and editing of the records become simpler and quicker.

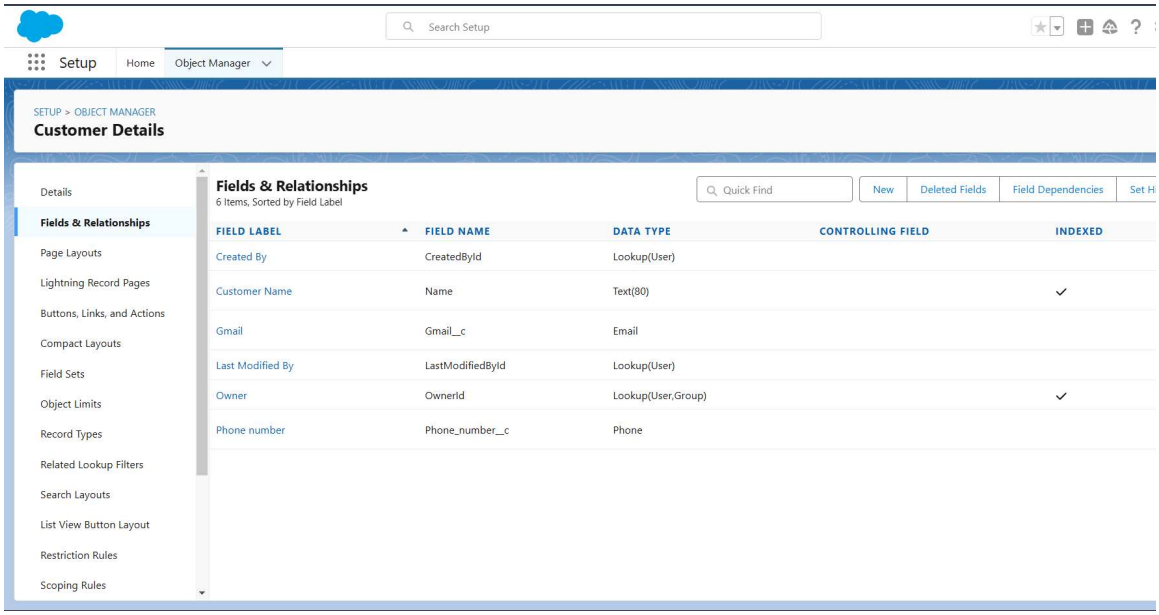
Creation of elds for the Customer Details object

- To create elds in an object:
  - Go to setup >> click on Object Manager >> type object name(Customer Details) in search bar >> click on the object.
  - Now click on “Fields & Relationships” >> New
  - Select Data Type as a “Phone”
  - Click on next.
  - Fill the Above as following:
    - Field Label: Phone number
    - Field Name : gets auto generated

- Click on Next >> Next >> Save and new.

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

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



Creation of Lookup Fields

Creation of Lookup Field on Appointment Object :

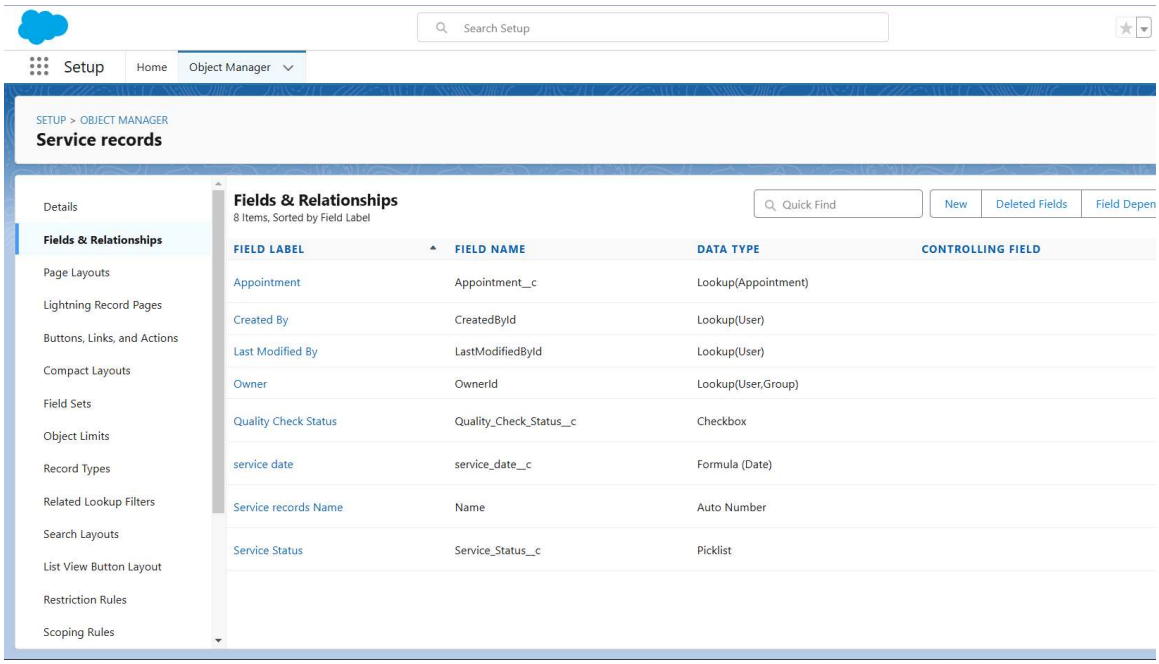
- Go to setup >> click on Object Manager >> type object name(Appointment) in the search bar >> click on the object.
- Now click on “Fields & Relationships” >> New
- Select “Look-up relationship” as data type and click Next. 4. Select the related object “ Customer Details” and click next.

- Next >> Next >> Save.

Note: Make sure you complete Activity 4 Before continuing.

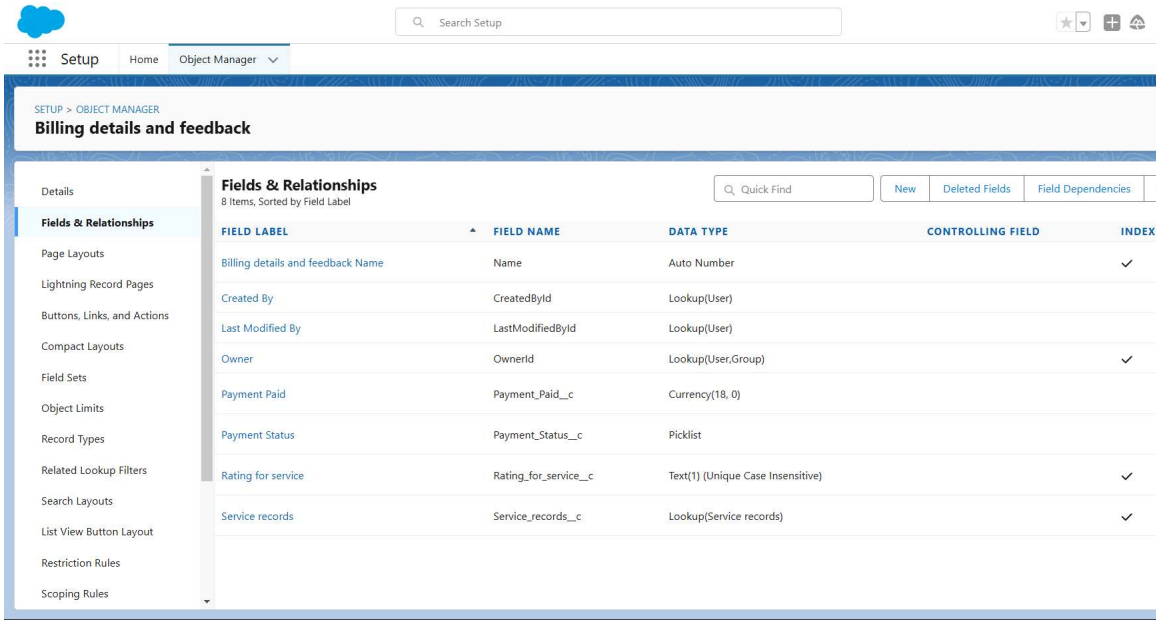
Creation of Lookup Field on Service records Object :

- Go to setup >> click on Object Manager >> type object name(Service records) in search bar >> click on the object.
- Now click on “Fields & Relationships” >> New
- Select “Look-up relationship” as data type and click Next.
- Select the related object “ Appointment ” and click next.
- Make it a required eld so click on Required.
- Scroll down for Lookup Filter and click on Show lter settings.
- Now add the lter criteria.
- Field : Appointment: Appointment Date >> Operator : less than >> select eld >> Appointment: Created Date
- Filter type should be Required.
- Error Message : Value does not match the criteria.
- Enable the lter by click on Active.
- Next >> Next >> Save.



Creation of Lookup Field on Billing details and feedback Object :

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



Creation of Checkbox Fields

Creation of CheckboxField on Appointment Object :

- Go to setup >>click on Object Manager >>type object name(Appointment) in search bar >> click on the object.
- Now click on “Fields & Relationships” >>New.
- Select “Check box” as data type and click Next.
- Give the Field Label: Maintenance service
- Field Name : is auto populated
- Default value : unchecked
- Click on next >>next >> save.

Creation of AnotherCheckbox Field on Appointment Object:

- Repeat the steps form 1 to 3.
- Give the Field Label : Repairs
- Field Nme : is auto populated
- Default value : unchecked

- Click on next >>next >> save.
- Follow the same and create another checkbox with given names
- Give the Field Label : Replacement Parts
- Field Nme : is auto populated
- Default value : unchecked
- Click on next >> next >> save.

Creation of CheckboxField on Servicerecords Object :

- Go to setup >>click on Object Manager >>type object name(Service records) in search bar >>click on the object.
- Now click on “Fields & Relationships” >>New.
- Select “Check box” as data type and click Next.
- Give the Field Label : QualityCheck Status
- Field Nme : is auto populated
- Default value : unchecked
- Click on next >>next >> save

Creation of date Fields

Creation of Date Field on Appointment Object:

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

Creation of CurrencyFields

Creation of CurrencyField on Appointment Object :

- Go to setup >>click on Object Manager >>type object name(Appointment) in the search bar >>click on the object.
- Now click on “Fields& Relationships” >>New.
- Select “Currency” as data type and click Next.
- Give the Field Label: Service Amount
- Field Nme : is auto populated
- Click on next
- Give read only for all the profiles in field level security for profile.
- Click on next > > save.

Creation of CurrencyField on Billingdetails and feedbackObject :

- Follow the same steps as mentioned above in Billing details and feedback Object.
- Change the label name as mentioned.
- Give the Field Label : PaymentPaid
- Field Nme : is auto populated

Creation of Text Fields

- Go to setup >>click on Object Manager >>type object name(Appointment) in the search bar >> click on the object.
- Now click on “Fields & Relationships” >>New.
- Select “Text” as data type and click Next.
- Give the Field Label : Vehicle number plate
- Field Name : is auto populated
- Length : 10
- Make field as Required and Unique.
- Click on next >>next >> save.

Creation of Text Fields in Billingdetails and feedbackobject :

- Go to setup >>click on Object Manager >>type object

name(Billing detailsand feedback) in search bar >> click on the object.

- Now click on “Fields & Relationships” >>New.
- Select “text” as data type and click Next.
- Give the Field Label: Rating for service
- Field Name : is auto populated
- Length : 1
- Make field as Required and Unique.
- Click on next >> next >> save

Creation of PicklistFields

Creation of PicklistFields in Servicerecords object :

- Go to setup >>click on ObjectManager >> type object name(Service records) in search bar >> click on the object.
- Click on fields & relationship >> click on New.
- Select Data type as “Picklist” and click Next.
- 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.
- The values are: Started, Completed.
- Click Next.
- Next >> Next >>Save.

Creation of Picklist Fieldsin Billing detailsand feedback object:

- Go to setup >>click on Object Manager >>type object name(Billing details and feedback) in search bar >> click on the object.
- Click on fields & relationship >> click on New.
- Select Data type as “Picklist” and click Next.
- 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.

- The values are: Pending, Completed.
- Click Next.
- Next >> Next >>Save.

Creating Formula Fieldin Service recordsObject

- Go to setup >>click on ObjectManager >> type object name(Service records) in search bar >> click on the object.
- Click on fields & relationship >> click on New.
- Select Data type as “Formula” and click Next.
- Give Field Label and Field Name as “servicedate” and select formula return type as “Date” and click next.
- Insert field formulashould be : CreatedDate
- click “Check Syntax” .
- Click next >> next >> Save.

Validation rule

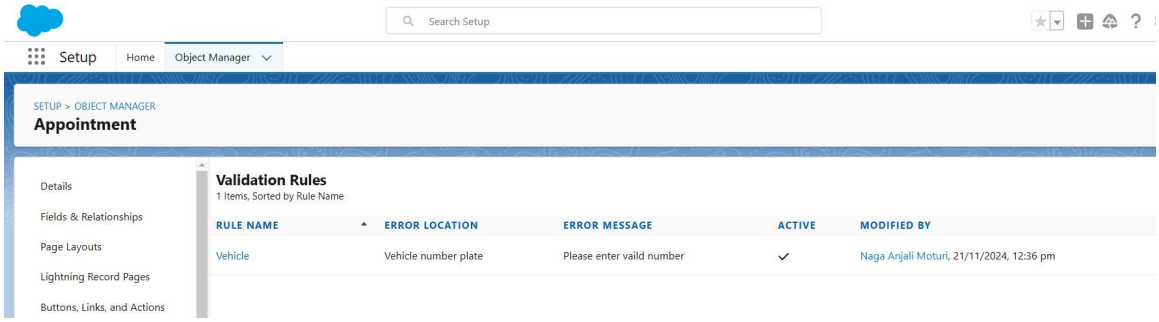
Validation rules are appliedwhen a user tries to save a record and are used to checkif the data meets specified criteria.If the criteria are not met, the validation rule triggers an error message and prevents the user from savingthe record untilthe issues are resolved.

To create a validation rule to an Appointment Object

- Go to the setuppage >> click on objectmanager >> From drop down click edit for Appointment object.
- Click on the validation rule >> clickNew.
- Enter the Rule name as “ Vehicle ”.
- Insert the ErrorCondition Formula as : - NOT(REGEX(Vehicle_number_plate__c , "[A-Z]{2}[0-9]{2}[A-Z]{2}[0-9]{4}"))

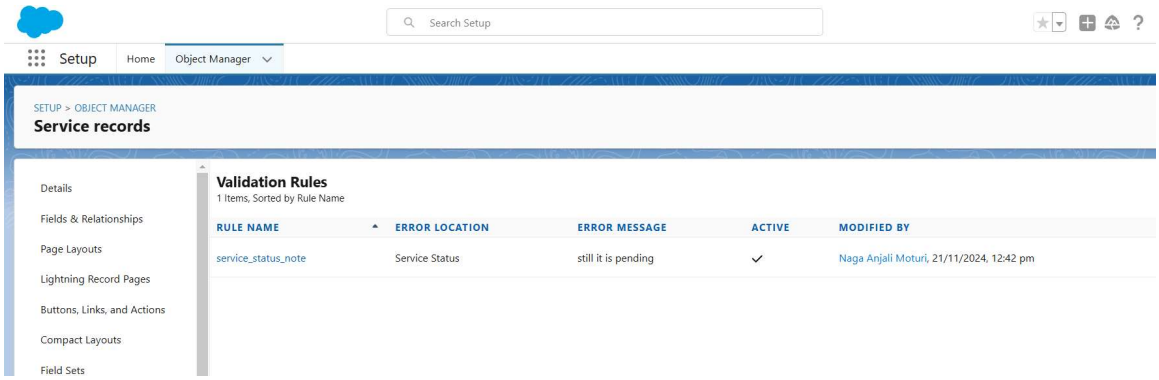
5 . Enter the Error Messageas “Please enter vaild number”, select the Error locationas

Field and selectthe field as “Vehicle number plate”, and click Save.



To create a validation rule to an Service recordsObject

- Go to the setuppage >> click on objectmanager >> From drop down click edit for Service records object.
- Click on the validation rule >> clickNew.
- Enter the Rule name as “ service_status_note ”.
- Insert the ErrorCondition Formula as : -
- NOT(ISPICKVAL(Service_Status__c , "Completed"))
- Enter the Error Messageas “still it is pending”,select the Error location as Field and select the field as “Service status”, and click Save.



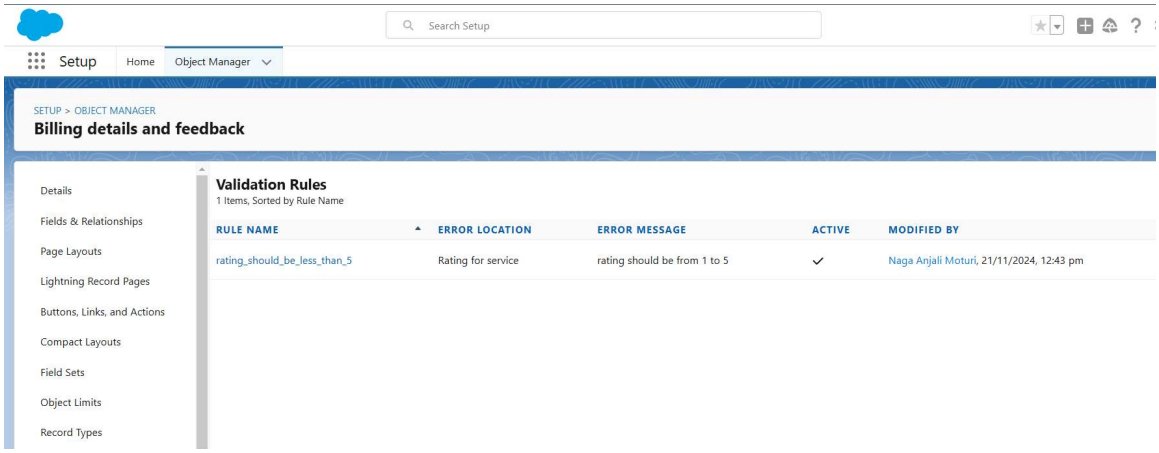
To create a validation rule to an Billing detailsand feedback Object

- Go to the setuppage >> clickon object manager>> From drop down

clickedit for Billing details and feedback object.

- Click on the validation rule >> click New.
- Enter the Rule name as “ rating_should_be_less_than_5”.
- Insert the ErrorCondition Formula as : - NOT(REGEX(Rating_for_service c , "[1-5]{1}"))

5.Enter the ErrorMessage as “ratingshould be from 1 to 5”, selectthe Error locationas Field and select the field as “Rating for Service”, and click Save.

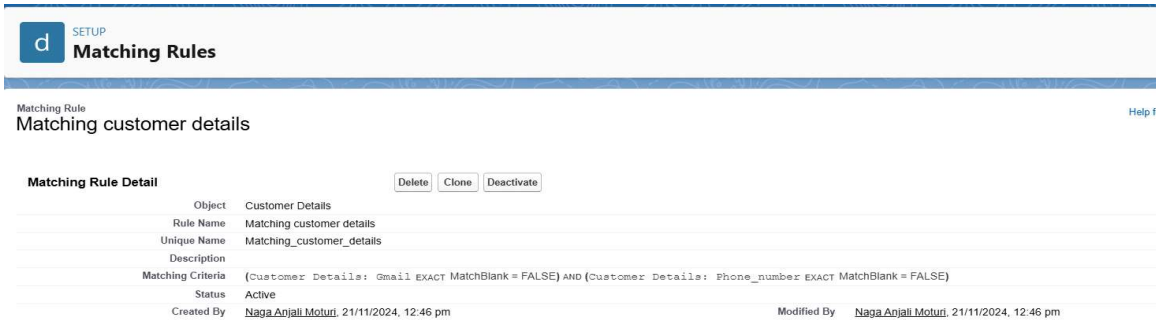


Duplicate rule

To create a matching rule to an Customer detailsObject

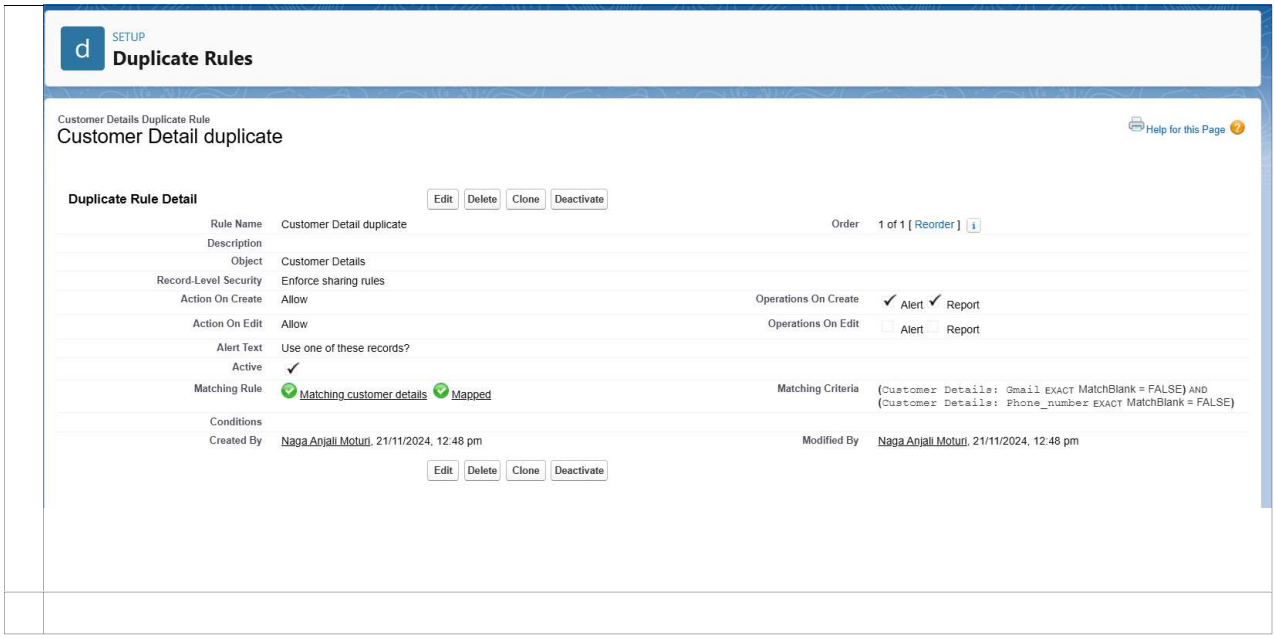
- Go to quick find box in setup and search for matchingRule.
- Click on matchingrule >> click on New Rule.
- Select the object as Customer detailsand click Next.
- Give the Rule name : Matching customer details
- Unique name : is auto populated
- Define the matching criteria as
    - Field : MatchingMethod
    - Gmail:Exact
    - Phone Number:Exact
- Click save.

- After Saving Click on Activate.



To create a Duplicate rule to an Customer detailsObject

- Go to quick find box in setup and search for Duplicate rules.
- Click on Duplicate rule >> click on New Rule >>select customer detailsobject.
- Give the Rule name as : CustomerDetail duplicate
- Scroll a little in Matching rule section
- Select the matching rule : Matching customer details
- And Click on save.
- After saving the Duplicate Rule, Click on Activate.



Role & Role Hierarchy

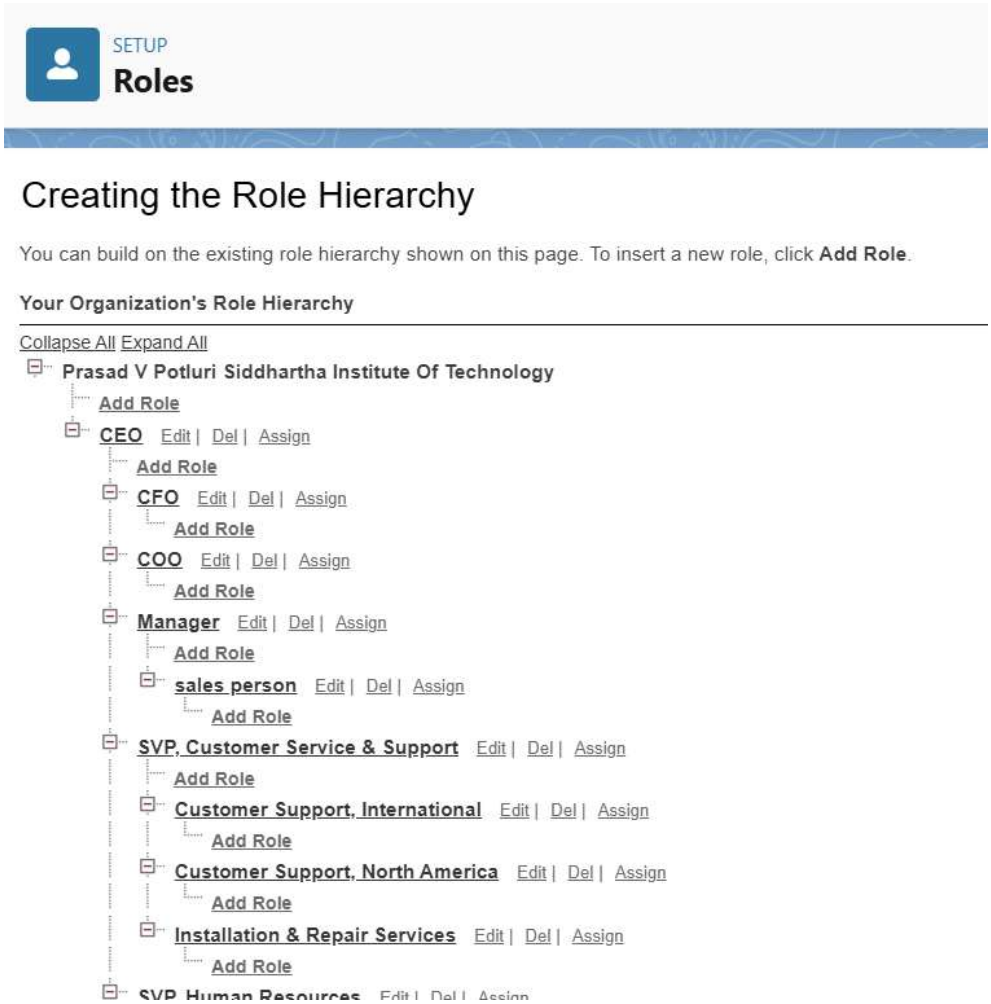
A role in Salesforce definesa user's visibility access at the record level.Roles may be used to specify the types of access that people in your Salesforce organization can have to data. Simply put, it describes what a user could see within the Salesforce organization.

Creating Manager Role

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

Creating another roles

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



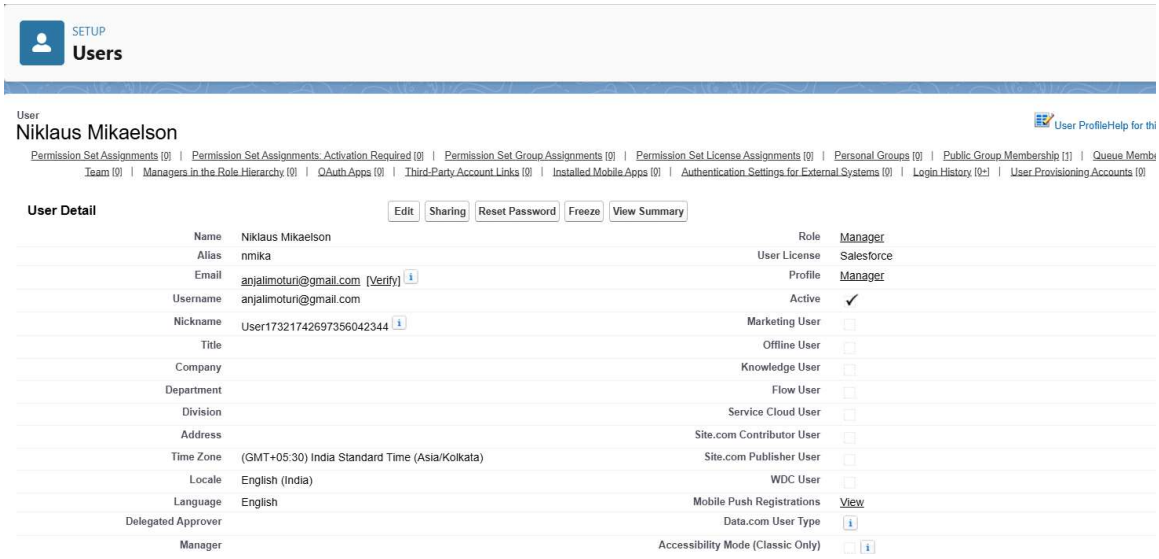
Users

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

Create User

- Go to setup >>type users in quick find box >>select users >>click New user.

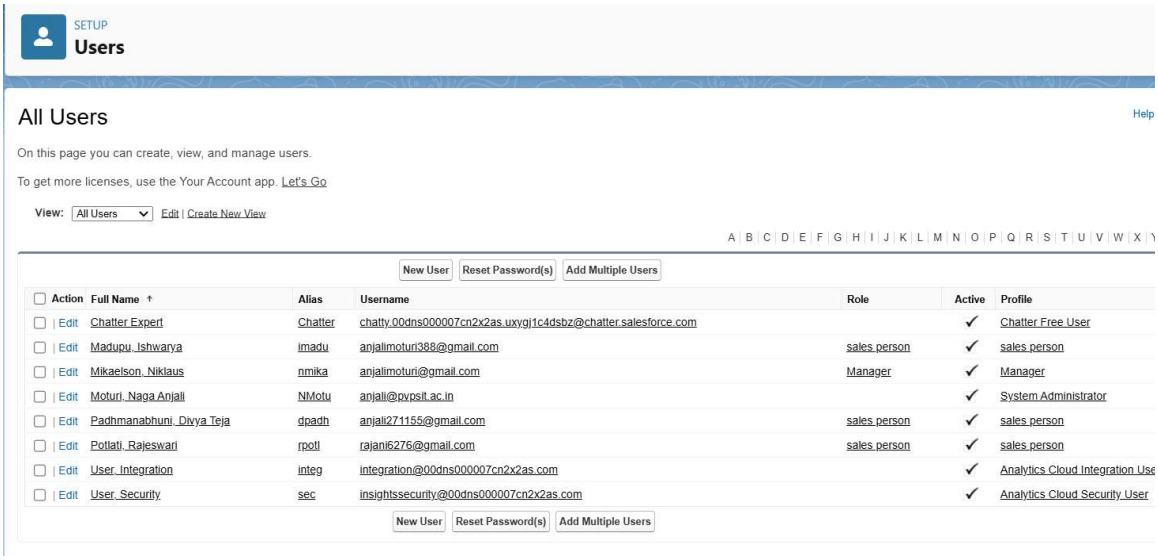
- Fill in the fields
    - First Name : Niklaus
    - Last Name : Mikaelson
    - Alias : Give a Alias Name
    - Email id : Give your Personal Emailid
    - Username : Username should be in this form: text@text.text
    - Nick Name : Give a Nickname
    - Role : Manager
    - User licence : Salesforce
    - Profiles : Manager
- Save.



creating another users

- Repeat the steps and create another user using
    - Role : sales person
    - User licence : Salesforce Platform
    - Profile :sales person

Note : create atleast3 users with these permissions.



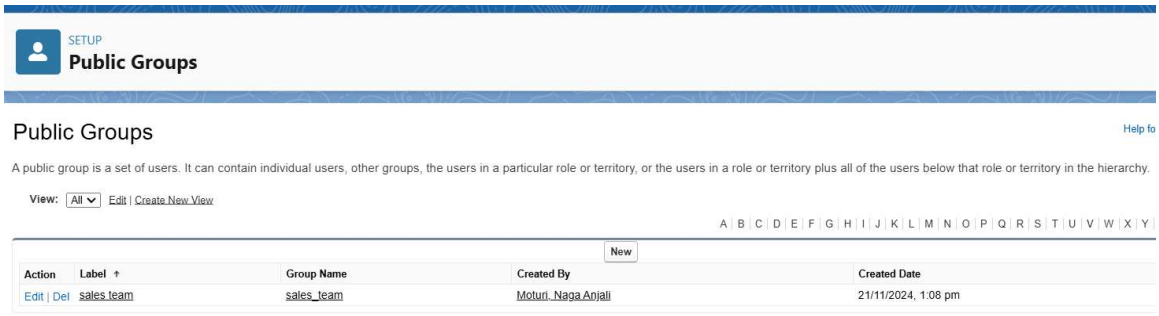
Public groups

Public groups are a valuabletool 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 ensuringthat users have appropriate access to the resources they need.

Creating New Public Group

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

- Search for Roles.
- In Available Members selectSales person and click on add it will be moved to selected member.
- Click on save.



Sharing Setting

Salesforce allows you to configure sharingsettings to controlhow 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 definethe default levelof access for all objectswithin 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.

Usersat higher levelsin the hierarchy have greateraccess to recordsowned by or shared with users lower in the hierarchy.

The role hierarchy is often used in combination with OWD settingsto 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 typicallyused to grant general object and field access, while permission sets can be used to extend those permissions to specific users.

Creating Sharing settings

- Go to setup >>type users in quick find box >>select Sharing Settings>> click Edit.
- Change the OWD settingof the Service records Objectto private as shown in fig.
- Click on save and refresh.
- Scroll down a bit, Click new on Servicerecords sharing Rules.
- Give the Label name as " Sharing setting"
- Rule name is auto populated.
- In step 3 : Select which records to be shared,members of " Roles " >> " Sales person"
- In step 4: sharewith, select " Roles " >> " Manager "
- In step 5 : Changethe access levelto " Read / write ".
- Click on save.

Flows

Create a Flow

- Go to setup >>type Flow in quick find box >>Click on the Flow and Select the New Flow.
- Select the Record-triggered flow and Click on Create.
- Select the Object as "Billing details and feedback"in the Drop down list.
- Select the Trigger Flow when: "A record is Created or Updated".
- Select the Optimize the flow for: "Actions and Related Records" and Click on Done.
- Under the Record-triggered Flow Click on "+" Symboland In the Drop down List select the "Update records Element".Give the Label Name : Amount Update
- Api name : is auto populated
- Set a filter condition : All Conditions are met(AND)
- Field : Payment_Status_c

- Operator : Equals
- Value : Completed
- And Set Field Values for the Billingdetails and feedbackRecord
- Field : Payment_Paid__c
- Value : {!$Record.Service_records_r.Appointment_r.Service_Amount_c}
- Click On Done.Before creating another Element.Create a New Resource form Toolbox form top left.
- Click on the New Resource,And select Variable.
- Select the resourcetype as text template.
- Enter the API name as " alert".
- Change the view as Rich Text ? View to Plain Text.
- In body fieldpaste the syntaxthat given below.

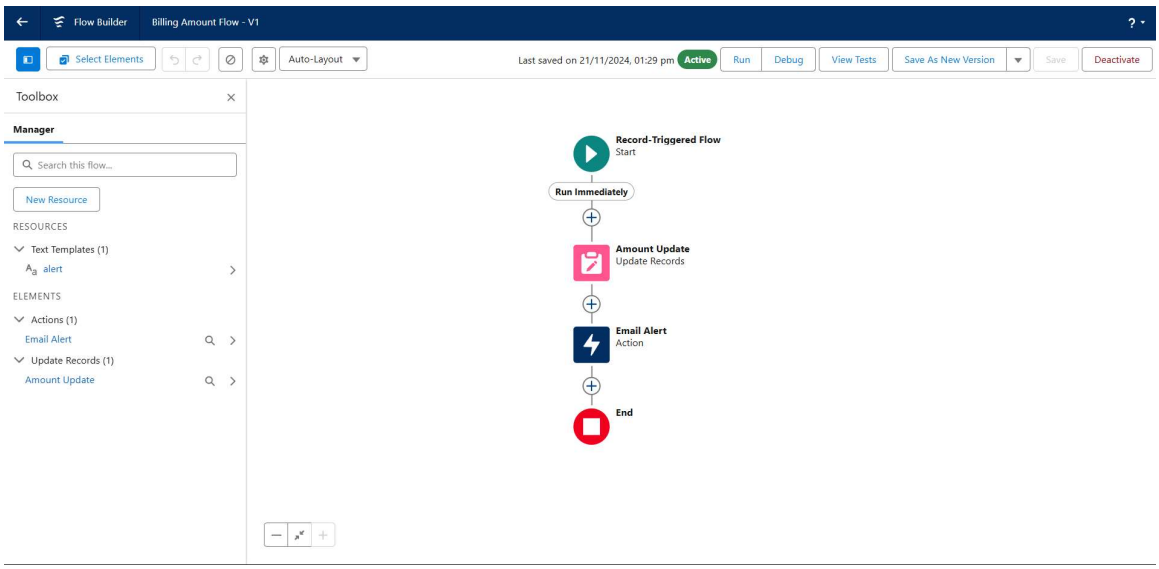
Dear {!$Record.Service_records_r.Appointment_r.Customer_Name_r.Name},

I hope this message findsyou well. I wanted to take a moment to express my sincere gratitude for your recent paymentfor the servicesprovided by our garage management team. Your prompt paymentis 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 .

- Click done.
- Now Click on Add Element, select Action.
- Their action bar will be opened in that searchfor " send email " and click on it.
- Give the label name as " Email Alert"
- API name will be auto populated.
- Enable the body in set input values for the selected action.
- Select the text template that created , Body : {!alert}
- Include recipient address list select the email form the record.

- RecipientAddressList: {!$Record.Service_records__r.Appointment__r.Customer_Name__r.Gmail c}
- Include subject as “ Thank You for Your Payment - Garage Management”.
- Click done.
- Click on save. Give the Flow label , Flow Api name will be autopopulated.
- And click save, and click on activate.



Apex Trigger

Apex can be invoked by using triggers. Apex triggers enableyou to perform custom actions before or after changes to Salesforce records, such as insertions, updates, or deletions.

A triggeris Apex code that executesbefore or after the following types of operations:

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

For example, you can have a triggerrun before an object's recordsare inserted into the database, after recordshave been deleted,or even aftera record is restored from the Recycle Bin.

You can definetriggers for top-levelstandard 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 objectmanagement settings for the objectwhose triggers you want to access, go to Triggers.

There are primarilytwo types of Apex Triggers:

Before Trigger: This type of trigger in Salesforce is used either to update or validate the values of a recordbefore they can be savedinto the database. So, basically, the before triggervalidates the record firstand then savesit. Some criteriaor code can be set to checkdata before it gets ready to be inserted into 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 Servicethe customer selected for there Vehicle.

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

Code:

```
public class AmountDistributionHandler {

  public static void amountDist(list<Appointment__c>
```

```
    listApp){ list<Service_records__c> serList = new list
    <Service_records__c>(); for(Appointment__c app :
    listApp){
      if(app.Maintenance_service__c == true && app.Repairs__c == true &&
app.Replacement_Parts_c == true){
        app.Service_Amount__c = 10000;
      }
      else if(app.Maintenance_service__c == true && app.Repairs__c == true)
        { app.Service_Amount_c = 5000;
      }
      else if(app.Maintenance_service_c == true && app.Replacement_Parts_c
        == true){ app.Service_Amount_c = 8000;
      }
      else if(app.Repairs__c == true && app.Replacement_Parts__c == true)
        { app.Service_Amount_c = 7000;
      }
      else  if(app.Maintenance_service___c  ==  true)
        { app.Service_Amount_c = 2000;
      }
      else if(app.Repairs__c ==
        true)
        {  app.Service_Amount
        c = 3000;
      }
      else  if(app.Replacement_Parts___c  ==  true)
        { app.Service_Amount_c = 5000;
      }
  }
  }
}
```

Trigger Handler :

How to create a new trigger :

- While still in the trailhead account, navigate to the gear icon in the top right

corner.

- Click on developer consoleand you will be navigatedto a new console window.
- Click on File menu in the tool bar, and click on new? Trigger.
- Enter the trigger name and the object to be triggered.
- Name : AmountDistribution
- sObject : Appointment__c

Syntax For creating trigger:

The syntax for creating trigger is :

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

{

}

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

Code:

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

  }

}
```

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 sharingreports, review these reporting basics.

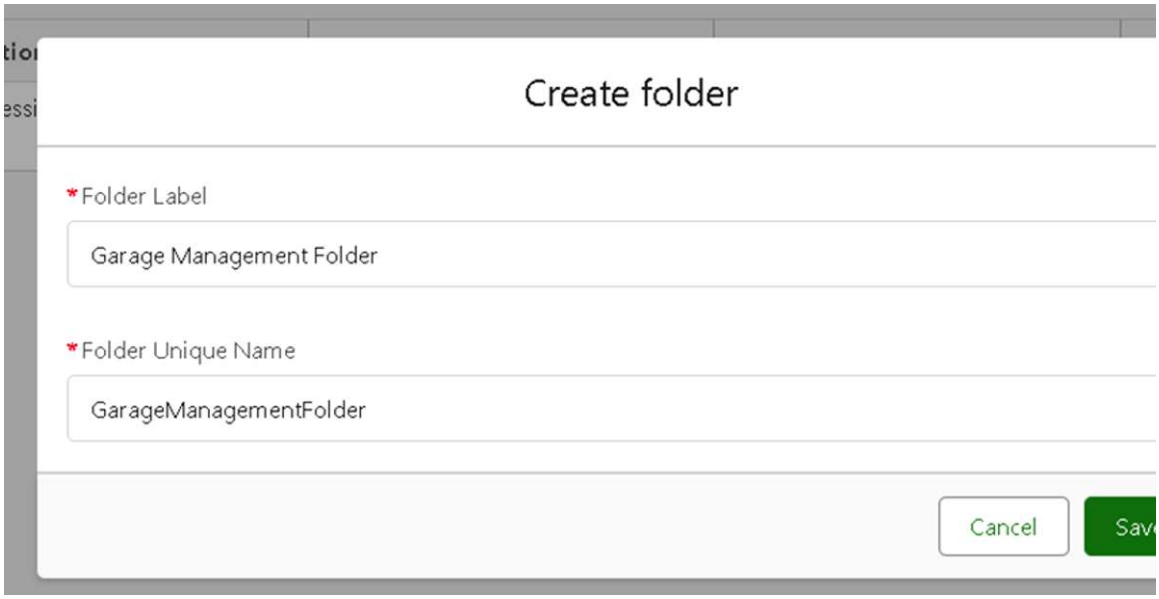
Types of Reports in Salesforce

- Tabular
- Summary

- Matrix
- Joined Reports

create a report folder

- Click on the app launcher and search for reports.
- Click on the report tab, click on new folder.
- Give the Folder labelas “Garage Management Folder”, Folder uniquename will be auto populated.
- Click save.



Sharing a report folder

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

Create Report Type

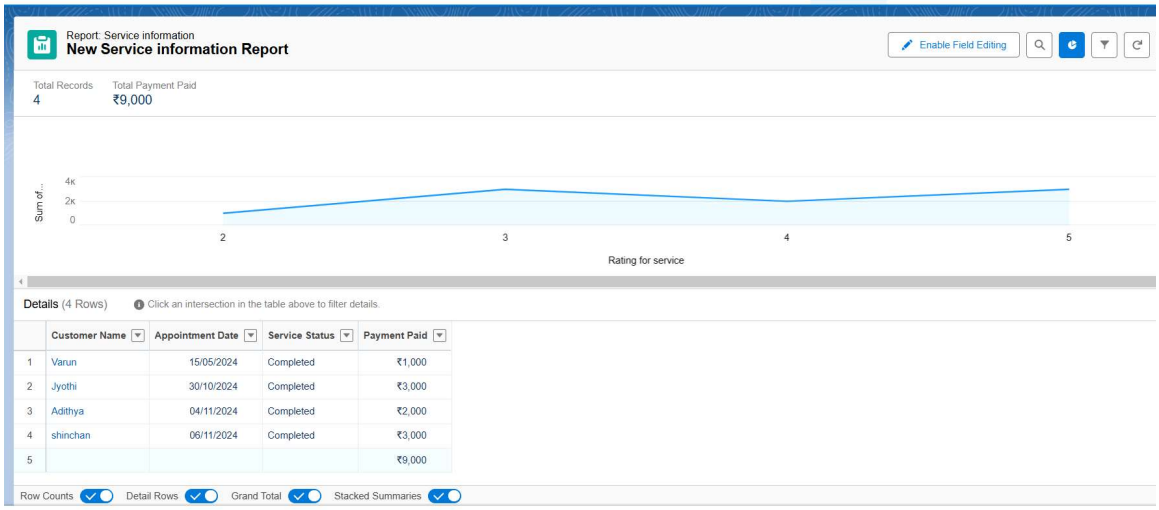
- Go to setup >> type users in quick find box >>select Report Type >> click on Continue.
- Click on new custom reporttype.
- Select the Primary object as “ Customer details” .
- Give the Reporttype Label as “ Serviceinformation ”
- Report type Name is autopopulated.
- Keep the Description as same.
- Select Store in Category as “ otherReports ”
- Select the deployment status as “ Depolyed ”, click on Next.
- now , Click on Relatedobject box.
- Click on SelectObject, choose Appointment Object as shownin fig
- Again Click to relate anotherobject.
- And select the related objectas “ service records”.
- Repeat the process and select the related object as “ Billing details and feedback”.
- And click on save.

Create Report

Note : Before creating report,create latest “10” records in every object. Try to fill every field in each record for better experience.

- Go to the app >>click on the reports tab
- Click New Report.
- Select the Category as other reports,search for ServiceInformation, select that report, click on it. And click on start report.
- Their outline pane is opened alredy,select the fields that mentionedbelow in column section.
    - Customer name
    - Appointment Date
    - Service Status

- Payment paid
- Remove the unnecessary fields.
- Select the fields that mentioned below in GROUP ROWS section.
    - Rating for Service
- Select the fields that mentioned below in GROUP ROWS section.
    - Payment Status
- Click on Add Chart , Select the Line Chart.
- Click on save, Give the reportName : New Service information Report
- Report unique Name is auto populated.
- Select the folder the createdand Click on save.



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 to help users identifytrends, sort out quantities, and measure the impact of their activities. Before building, reading, and sharing dashboards, review these dashboardbasics.

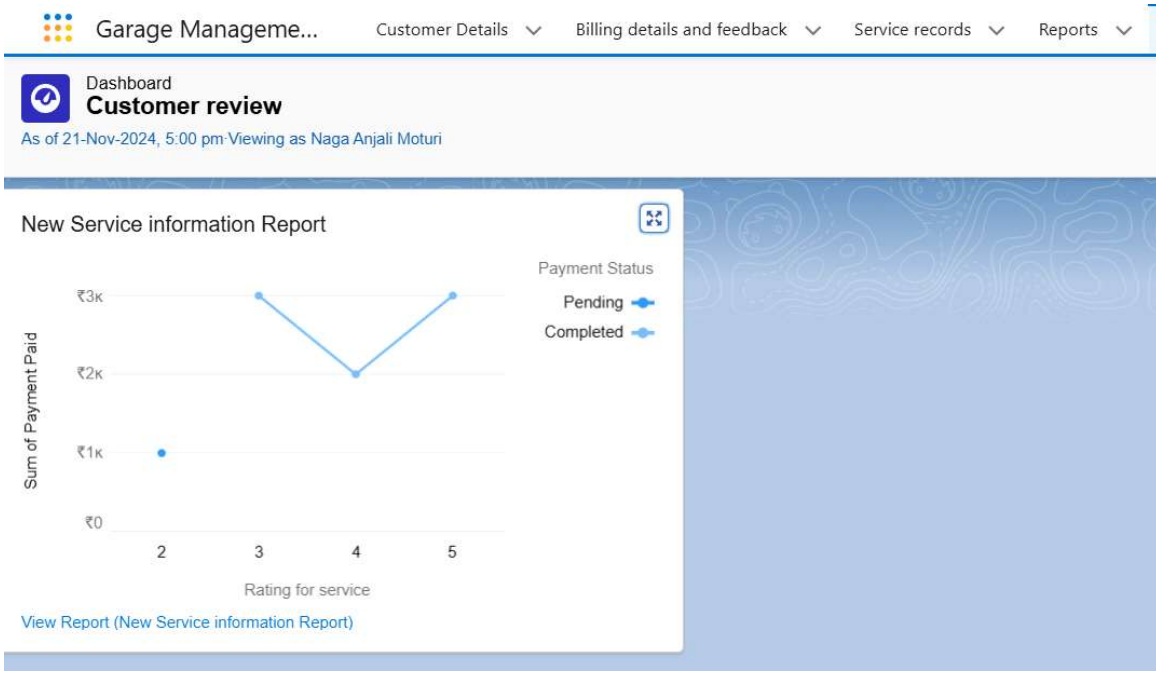
Create Dashboard Folder

- Click on the app launcher and search for dashboard.

- Click on dashboard tab.
- Click new folder,give the folderlabel as “ Service Ratingdashboard”.
- Folder unique name will be auto populated.
- Click save.
- Follow the same steps,form milestone 15, and activity2, and provide the sharing settings for the folder that just created.

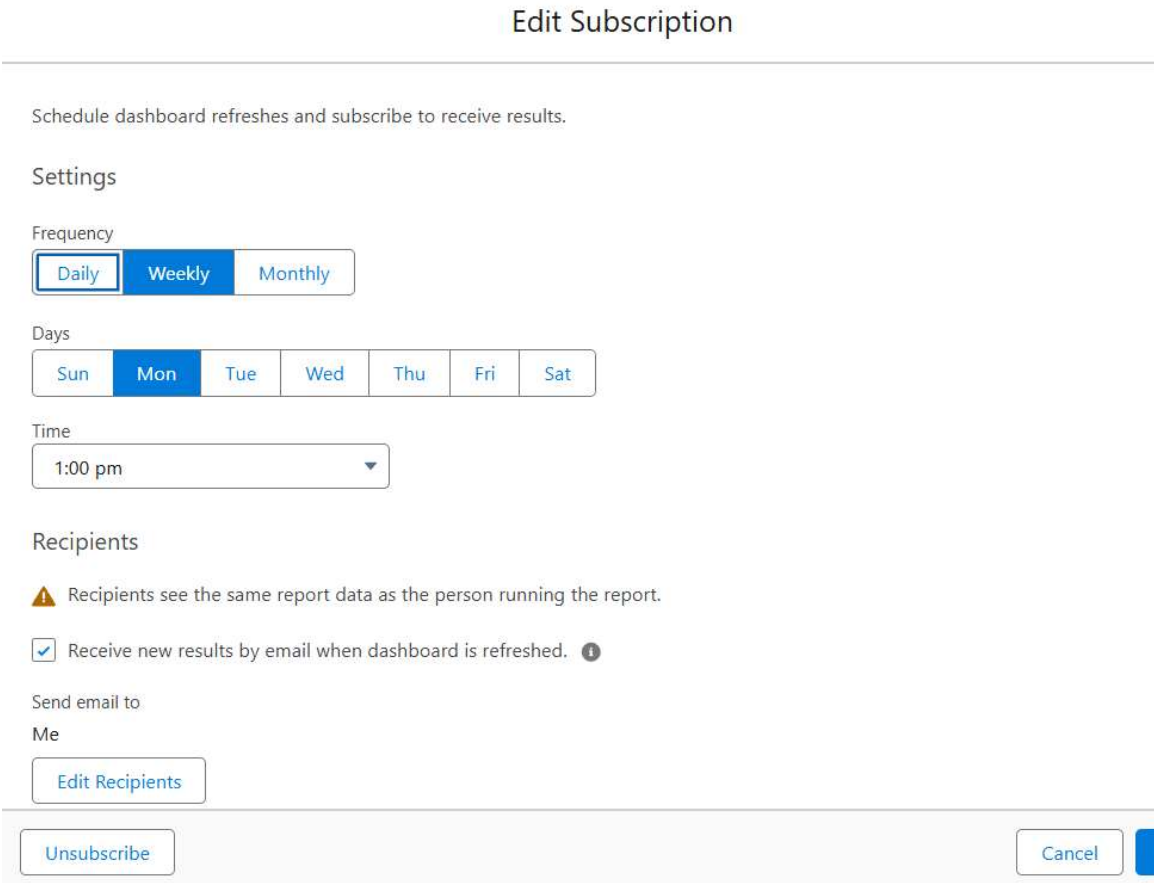
Create Dashboard

- Go to the app >>click on the Dashboards tabs.
- Give a Name and select the folder that created, and click on create.
- Select add component.
- Select a Reportand click on select.
- Select the Line Chart. Change the theme.
- Click Add then click on Save and then click on Done.
- Preview is shown below.



Subcription:

- After that Click on Subcribe on top right.
- Set the Frequency as “ weekly”.
- Set a day as monday.
- And Click on save.



Conclusion:

The Garage Management System (GMS) implemented on the Salesforce platform successfully achieves its goal of streamlining and automating daily garage operations. By leveraging Salesforce’s cloud-based CRM capabilities, the system provides an integrated solution for managing customers, vehicles, service bookings, inventory, billing, and reporting—all within a single, scalable environment.

Through this project, we demonstrated how Salesforce can be customized using objects, workflows, process builders, and automation tools such as Flow and Apex to meet the unique needs of a garage business. The system improves operational efficiency, data accuracy, and customer satisfaction by ensuring real-time tracking of service requests, automated communication, and transparent billing processes.