

Garage Management system

The Garage Management System is a valuable tool for automotive repair facilities, helping them deliver top-notch service, increase operational efficiency, 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 workflows and higher customer satisfaction.

1. Appointment Scheduling:

- a. Simplifies the booking process for customers.
- b. Enables staff to manage daily schedules efficiently, reducing downtime and improving resource allocation.

2. Vehicle Management:

- a. Maintains detailed records of vehicles, including service history, repairs, and maintenance schedules.
- b. Tracks vehicle status during servicing for better communication with customers.

3. Customer Relationship Management (CRM):

- a. Stores customer details and preferences.
- b. Sends service reminders, follow-ups, and promotional offers to build loyalty.

4. Inventory and Spare Parts Management:

- a. Tracks spare parts stock levels, automates reorder processes, and prevents stockouts.
- b. Ensures that mechanics always have the necessary tools and parts on hand.

5. Billing and Invoicing:

- a. Generates professional invoices quickly and accurately.
- b. Supports multiple payment methods, discounts, and tax calculations.

6. Work Order Management:

- a. Creates detailed work orders with a list of tasks, estimated costs, and timelines.
- b. Helps staff prioritize jobs and ensure timely completion.

7. Reporting and Analytics:

- a. Provides insights into key performance indicators like revenue, job completion rates, and customer feedback.
- b. Helps identify trends and areas for improvement.

Salesforce

Introduction:

Are you new to Salesforce? Not sure exactly what it is, or how to use it? Don't know where you should start on your learning journey? If you've answered yes to any of these questions, then you're in the right place. This module is for you.

Welcome to Salesforce! Salesforce is game-changing technology, with a host of productivity-boosting features, that will help you sell smarter and faster. As you work toward your badge for this module, we'll take you through these features and answer the question, "What is Salesforce, anyway?".

What Is Salesforce?

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.

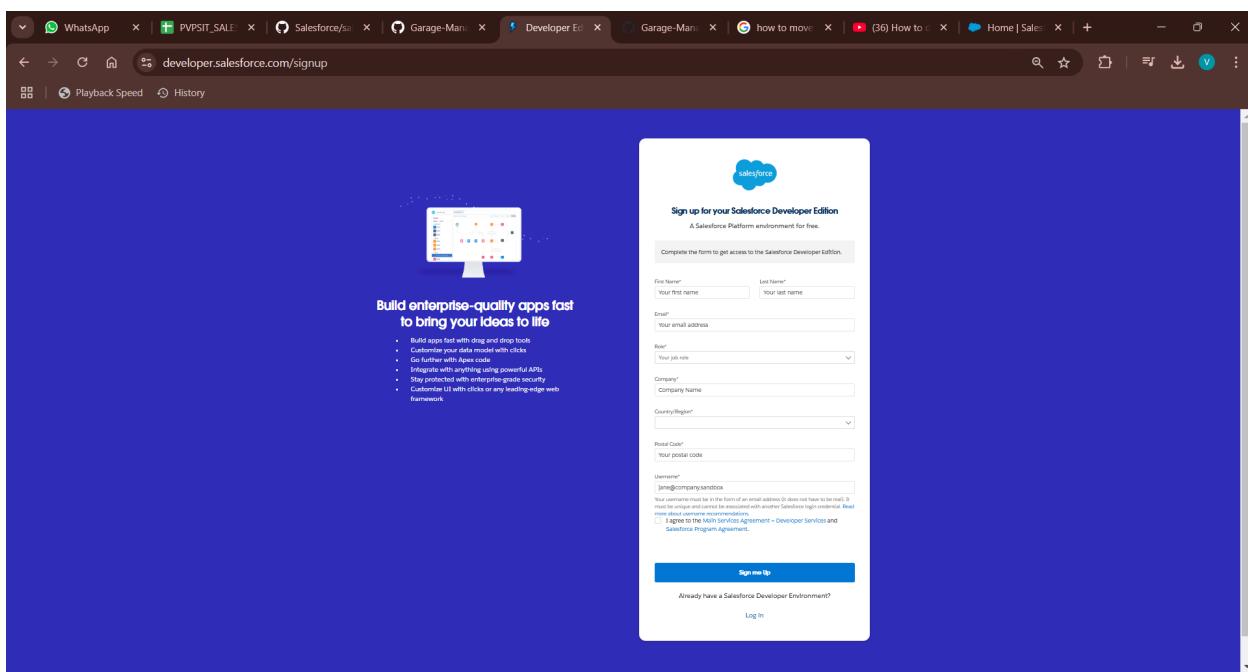
So what does that really mean? Well, before Salesforce, your contacts, emails, follow-up tasks, and prospective deals might have been organized something like this:

<https://youtu.be/r9EX3IGde5k>

Creating Developer Account:

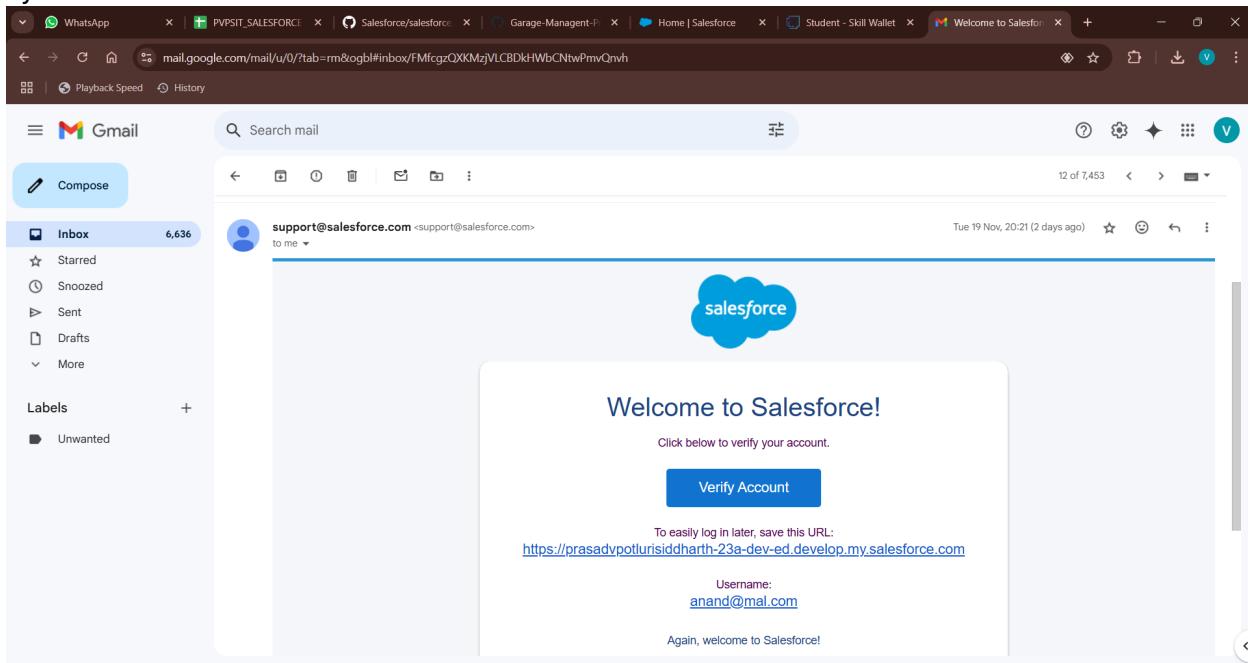
Creating a developer org in salesforce.

1. Go to <https://developer.salesforce.com/signup>
 2. On the sign up form, enter the following details :
 - a. First name & Last name
 - b. Email
 - c. Role : Developer
 - d. Company : College Name
 - e. County : India
 - f. Postal Code : pin code
 - g. Username :: should be a combination of your name and company This need not be an actual email id, you can give anything in the format : username@organization.com
- Click on sign me up after filling these



Account Activation

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



OBJECT

What Is an Object?

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

Salesforce objects are of two types:

1. **Standard Objects:** Standard objects are the kind of objects that are provided by salesforce.com such as users, contracts, reports, dashboards, etc.
2. **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 Details Object

To create an object:

1. From the setup page >> Click on Object Manager >> Click on Create >> Click on Custom Object.
1. Enter the label name >> CustomerDetails
2. Plural label name >> Customer Details
3. Enter RecordName Label and Format
 - a. Record Name >> CustomerName
 - b. DataType >> Text
2. Click on Allow reports and Track FieldHistory,
3. Allow search >> Save.

Create Appointment Object

To create an object:

1. From the setup page >> Click on Object Manager >> Click on Create >> Click on Custom Object.
1. Enter the label name >> Appointment
2. Plural label name >> Appointments
3. Enter Record Name Label and Format
 - a. Record Name >> Appointment Name
 - b. DataType >> Auto Number
 - c. Display Format >> app-{000}

- d. Starting number >> 1
- 2. Click on Allow reports and Track FieldHistory,
- 3. Allow search >> Save.

The screenshot shows the 'New Custom Object' setup page in Salesforce. The 'Custom Object Definition Edit' section contains the following information:

- Custom Object Information:**
 - Label: Appointment (Example: Account)
 - Plural Label: Appointments (Example: Accounts)
 - Start with lower case:
- Record Name Label and Format:**
 - Record Name: Appointment Name (Example: Account Name)
 - Date Type: Auto Number (Warning: If you plan to insert a high volume of records in this object, via the API for example, use the Text data type.)
 - Display Format: app-{000} (Example: A-0000) What Is This?
 - Starting Number: 1
- Optional Features:**
 - Allow Reports
 - Allow Activities
 - Allow Field History
 - Allow in Chatter Groups
 - Enable Licensing
- Object Classification:**

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

 - Allow Bulk API Access
 - Allow Streaming API Access
- Deployment Status:**
 - In Development
 - Deployed
- Search Status:**

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

 - Show Search
- Object Creation Options (Available only when custom object is first created):**
 - Add Notes and Attachments related list to default page layout
 - Launch New Custom Tab Wizard after saving this custom object

At the bottom right are 'Save', 'Save & New', and 'Cancel' buttons.

Create Service records Object

To create an object:

1. From the setup page >> Click on ObjectManager >> Click on Create >> Click on Custom Object.
1. Enter the label name >> Service records
2. Plural label name >> Service records
3. Enter Record Name Label and Format
 - a. Record Name >> Service recordsName
 - b. Data Type >> Auto Number
 - c. Display Format >> ser-{000}
 - d. Starting number >> 1
2. Click on Allow reports and Track FieldHistory,

3. Allowsearch >> Save.

The screenshot shows the 'Custom Object Definition Edit' page in the Salesforce setup. The object name is 'Service records'. The 'Record Name' field is set to 'Service recordsName' with a warning about high volume via API. The 'Display Format' is 'Auto Number' with a starting value of 1. Under 'Optional Features', 'Allow Reports' and 'Track Field History' are checked. The 'Deployment Status' is 'Deployed'. The 'Search Status' is 'Allow Search'. At the bottom, there are 'Save', 'Save & New', and 'Cancel' buttons.

Create Billing details and feedback Object

To createan object:

1. From the setup page >> Click on Object Manager >>Click on Create >> Click on Custom Object.
1. Enterthe label name >> Billingdetails and feedback
2. Plural label name >>Billing details and feedback
3. Enter RecordName Label and Format
 - a. Record Name >> Billingdetails and feedbackName
 - b. DataType >> Auto Number
 - c. Display Format >> bill-{000}
 - d. Starting number >> 1
2. Click on Allow reportsand Track FieldHistory,
3. Allow search>> Save.

New Custom Object

Permissions for this object are disabled for all profiles by default. You can enable object permissions in permission sets or by adding custom profiles. [Learn more](#) [Don't show this message again](#)

Custom Object Definition Edit

Custom Object Information

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

Label	<input type="text" value="Billing details and feedback"/> Example: Account
Plural Label	<input type="text" value="Billing details and feedback"/> Example: Accounts
Starts with record around	<input type="checkbox"/>

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

Object Name	<input type="text" value="Billing_details_and_feedback"/> Example: Account
-------------	--

Description

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

Content Name:

Enter Record Name Label and Format

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

Record Name	<input type="text" value="Billing details and feedback"/> Example: Account Name
-------------	---

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

Display Format: Example: A-(000) What Is This?

Starting Number:

Optional Features

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

Object Classification

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

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

Deployment Status

- In Development
- Deployed

Search Status

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

- Allow Search

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

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

Save **Save & New** **Cancel**

Tabs

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

Types of Tabs:

1. Custom Tabs

Custom object tabs are the user interface for custom applications that you build in salesforce.com. They look and behave like standard salesforce.com tabs such as accounts, contacts, and opportunities.

2. Web Tabs

Web Tabs are custom tabs that display web content or applications embedded in the salesforce.com window. Web tabs make it easier for your users to quickly access content and applications they frequently use without leaving the salesforce.com application.

3. Visualforce Tabs

Visualforce Tabs are custom tabs that display a Visualforce page. Visualforce tabs look and behave like standard salesforce.com tabs such as accounts, contacts, and opportunities.

4. Lightning Component Tabs

Lightning Component tabs allow you to add Lightning components to the navigation menu in Lightning Experience and the mobile app.

5. Lightning Page Tabs

Lightning Page Tabs let you add Lightning Pages to the mobile app navigation menu. Lightning Page tabs don't work like other custom tabs. Once created, they don't show up on the All Tabs page when you click the Plus icon that appears to the right of your current tabs. Lightning Page tabs also don't show up in the Available Tabs list when you customize the tabs for your apps.

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.

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 .

New Custom Object

Permissions for this object are disabled for all profiles by default. You can enable object permissions in permission sets or by editing custom profiles. [Set new rules](#) [Don't show this message again](#)

Custom Object Definition Edit

Custom Object Information

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

Label	Billing details and feedback	Example:	Account
Plural Label	Billing details and feedback	Example:	Accounts
Starts with vowel sound	<input type="checkbox"/>		

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

Object Name	Billing details and feedback	Example:	Account
-------------	------------------------------	----------	---------

Description

Content-Sensitive Help Settings

<input checked="" type="radio"/> Open the standard Salesforce.com Help & Training window
<input type="radio"/> Open a window using a Visualforce page

Content Name

Enter Record Name Label and Format

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

Record Name	Billing details and feedback	Example:	Account Name
Date Type	Auto Number	Warning:	If you plan to insert a high volume of records in this object, via the API for example, use the Text data type.
Display Format	bill-(000)	Example:	A-(0000) What Is This?
Starting Number	1		

Optional Features

<input checked="" type="checkbox"/> Allow Reports
<input type="checkbox"/> Allow Activities
<input checked="" type="checkbox"/> Track Field History
<input type="checkbox"/> Allow in Chatter Groups
<input type="checkbox"/> Enable Licensing

Object Classification

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

<input checked="" type="checkbox"/> Allow Sharing
<input checked="" type="checkbox"/> Allow Bulk API Access
<input checked="" type="checkbox"/> Allow Streaming API Access

Deployment status

<input type="radio"/> In Development
<input checked="" type="radio"/> Deployed

Search Status

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

<input checked="" type="checkbox"/> Allow Search
--

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

<input type="checkbox"/> Add Notes and Attachments related list to default page layout
<input type="checkbox"/> Launch New Custom Tab Wizard after saving this custom object

Buttons

[Save](#) [Save & New](#) [Cancel](#)

The Lightning App

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

Lightning apps let you brand your apps with a custom 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 efficiently by easily switching between apps.

Create a Lightning App

To create a lightningapp page:

1. Go to setup page >> search “app manager” in quick find >> select “app manager” >> click on New lightning App.
2. Fill the app name in app details as Garage Management Application >> Next >> (App option page) keep it as default >> Next >> (Utility Items) keep it as default >> Next.
3. To Add Navigation Items:
4. Select the items (Customer Details, Appointments, Service records, Billing details and feedback, Reports and Dashboards) from the searchbar and move it using the arrow button >> Next.
5. To Add User Profiles: Search profiles (System administrator) in the search bar >> click on the arrow button >> save & finish.

The screenshot shows the Salesforce Lightning Experience App Manager interface. The left sidebar navigation includes sections like SalesForce Mobile App, Data, Apps (with App Manager selected), Connected Apps, External Client Apps, Lightning Bolt, and Flow Category. The main content area displays a table titled "Lightning Experience App Manager" with 24 items. The table columns are: App Name, Developer Name, Description, Last Modified, App Type, and View. One row, "Garage Management Application" (Developer Name: Garage_Management_Application), is highlighted with a red border. The bottom status bar shows the date as 16-11-2024 and the time as 23:27.

App Name	Developer Name	Description	Last Modified	App Type	View
All Tabs	AllTabSet	Build CRM Analytics dashboards and apps	16/11/2024, 7:28 pm	Classic	
Analytics Studio	Insights	Build CRM Analytics dashboards and apps	16/11/2024, 7:28 pm	Classic	
App Launcher	AppLauncher	App Launcher tabs	16/11/2024, 7:28 pm	Classic	
Automation	FlowsApp	Automate business processes and repetitive tasks.	16/11/2024, 7:33 pm	Lightning	
Bolt Solutions	LightningBolt	Discover and manage business solutions designed for your industry.	16/11/2024, 7:32 pm	Lightning	
Business Rules Engine	ExpressionSetConsole	Create and maintain business rules that perform complex lookups and calculations.	16/11/2024, 7:28 pm	Lightning	
Community	Community	Salesforce CRM Communities	16/11/2024, 7:28 pm	Classic	
Content	Content	Salesforce CRM Content	16/11/2024, 7:28 pm	Classic	
Data Manager	DataManager	Use Data Manager to view limits, monitor usage, and manage recipes.	16/11/2024, 7:28 pm	Lightning	
Digital Experiences	SalesforceCMS	Manage content and media for all of your sites.	16/11/2024, 7:28 pm	Lightning	
Garage Management Application	Garage_Management_Application		16/11/2024, 9:27 pm	Lightning	
Lightning Usage App	LightningInstrumentation	View Adoption and Usage Metrics for Lightning Experience	16/11/2024, 7:28 pm	Lightning	
Marketing CRM Classic	Marketing	Track sales and marketing efforts with CRM objects.	16/11/2024, 7:28 pm	Classic	
Platform	Platform	The fundamental Lightning Platform	16/11/2024, 7:28 pm	Classic	

Fields

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

Types of Fields

1. Standard Fields
2. Custom Fields

Creation of fields for the Customer Details object

1. To create fields in an object:

- a. Go to setup >> click on Object Manager >> type object name (Customer Details) in search bar >> click on the object.
- b. Now click on "Fields & Relationships" >> New
- c. Select Data Type as a "Phone"
- d. Click on next.
- e. Fill the Above as following:
 - i. Field Label: Phone number
 - ii. Field Name : gets auto generated
 - iii. Click on Next >> Next >> Save and new.

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

To create another fields in an object:

- f. Go to setup >> click on Object Manager >> type object name (Customer Details) in search bar >> click on the object.
- g. Now click on "Fields & Relationships" >> New
- h. Select Data type as a "Email" and Click on Next
- i. Fill the Above as following:
- j. Field Label: Gmail
- k. Field Name : gets auto generated
- l. Click on Next >> Next >> Save and new.

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

- Header:** Welcome to Salesforce!, Student - Skill Wallet, Customer Details | Salesforce, (12) Garage Management, (13) 01 How to Create Cu..., Search Setup.
- Page:** SETUP > OBJECT MANAGER > Customer Details
- Left Sidebar:** Details, Fields & Relationships (selected), Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, Search Layouts, List View Button Layout, Restriction Rules, Scoping Rules.
- Main Content:** Fields & Relationships table with 6 items, sorted by Field Label.

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Created By	CreatedById	Lookup(User)		
Customer Name	Name	Text(80)		✓
Gmail	Gmail_c	Email		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		✓
Phone number	Phone_number_c	Phone		
- Bottom:** Weather icon (26°C, Mostly clear), Windows taskbar with various icons, system status (ENG IN), and date/time (16-11-2024, 21:33).

Creation of Lookup Fields

Creation of Lookup Field on Appointment Object:

1. Go to setup>> click on Object Manager>> type objectname(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 Activity4 Before cont

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Appointment Name	Name	Auto Number		✓
Created By	CreatedById	Lookup(User)		
Customer Details	Customer_Details__c	Lookup(Customer Details)		✓
Last Modified By	LastModifiedById	Lookup(User)		
Maintenance service	Maintenance_service__c	Checkbox		
Owner	OwnerId	Lookup(User,Group)		✓
Phone number	Phone_number__c	Phone		
Repairs	Repairs__c	Checkbox		
Replacement Parts	Replacement_Parts__c	Checkbox		

Creation of Lookup Field on Service recordsObject :

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 relatedobject “Appointment ” and click next.
5. Make it a required field so click on Required.
6. Scroll down for LookupFilter and click on Show filter settings.
7. Now add the filtercriteria.
8. Field : Appointment: Appointment Date >>Operator : less than >>select field >> Appointment: Created Date
9. Filter type should be Required.
10. Error Message : Value does not match the criteria.
11. Enable the filter by click on Active.
12. Next >> Next >>Save.

Creation of LookupField on Billingdetails and feedbackObject :

1. Go to setup>> click on Object Manager>> type objectname(Billing detailsand 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 relatedobject “ Servicerecords ” and clicknext.
5. Next >> Next >>Save & new.

Creation of CheckboxFields

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 AnotherCheckbox Field on Appointment Object:

1. Repeat the steps form 1 to 3.
2. Give the FieldLabel : Repairs
3. Field Nme : is auto populated
4. Default value : unchecked
5. Click on next >> next >> save.
6. Follow the same and createanother checkbox with given names
7. Give the Field Label: Replacement Parts

8. Field Nme : is auto populated

9. Default value : unchecked

10. Click on next >>next >> save.

The screenshot shows the Salesforce setup interface with the URL prasadypotlurisiddharth-22b-dev-ed.lightning.force.com/lightning/setup/ObjectManager/01IWU000001pI3F/FieldsAndRelationships/view. The page title is 'Appointment'. On the left, there's a sidebar with options like 'Page Layouts', 'Lightning Record Pages', 'Buttons, Links, and Actions', etc. The main content area is titled 'Fields & Relationships' and lists 9 items, sorted by Field Label. The table columns are 'FIELD LABEL', 'FIELD NAME', 'DATA TYPE', 'CONTROLLING FIELD', and 'INDEXED'. The data includes:

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Appointment Name	Name	Auto Number		✓
Created By	CreatedBy	Lookup(User)		
Customer Details	Customer_Details__c	Lookup(Customer Details)		✓
Last Modified By	LastModifiedBy	Lookup(User)		
Maintenance service	Maintenance_service__c	Checkbox		
Owner	OwnerId	Lookup(User,Group)		✓
Phone number	Phone_number__c	Phone		
Repairs	Repairs__c	Checkbox		
Replacement Parts	Replacement_Parts__c	Checkbox		

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 Nme : is auto populated

6. Default value : unchecked

7. Click on next >>next >> save

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

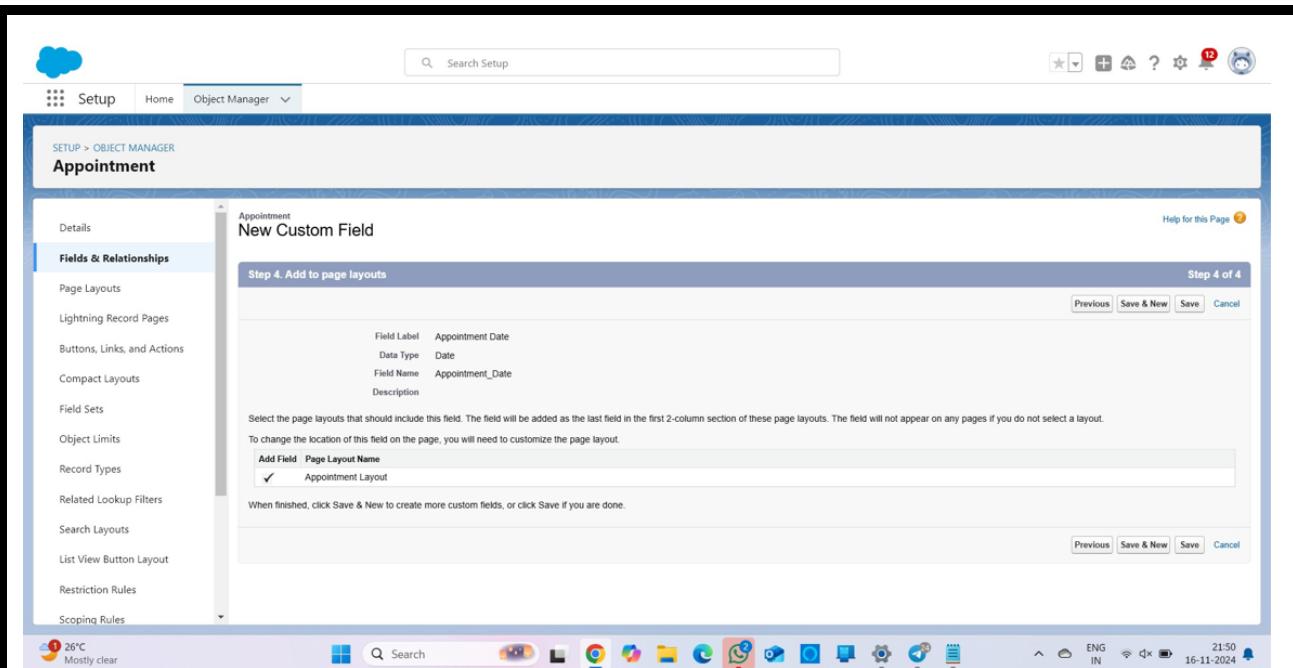
- Setup** tab selected.
- Object Manager** selected under the Service records object.
- Fields & Relationships** section selected.
- Fields & Relationships** table:

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Appointment	Appointment__c	Lookup(Appointment)		✓
Created By	CreatedById	Lookup(User)		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		✓
Phone number	Phone_number__c	Phone		
Quality Check Status	Quality_Check_Status__c	Checkbox		
Service recordsName	Name	Auto Number		✓
- Left sidebar categories: Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, Search Layouts, List View Button Layout, Restriction Rules, Scoping Rules.
- System bar at the bottom: Weather (26°C), Search, Notifications, System status (ENG IN), Date (16-11-2024), Time (21:48).

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 clickNext.
4. Give the Field Label : Appointment Date
5. Field Nme : is auto populated
6. Make it as a Required field by click on the Required option.
7. Click on next >>next >> save.



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 Nme : 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.

The screenshot shows the Salesforce Setup interface. The top navigation bar includes 'Setup', 'Home', and 'Object Manager'. A search bar says 'Search Setup' and there are various icons on the right. The main title is 'SETUP > OBJECT MANAGER' followed by 'Appointment'. On the left, a sidebar lists options like 'Details', 'Fields & Relationships' (which is selected), 'Page Layouts', 'Lightning Record Pages', etc. The main content area is titled 'Appointment New Custom Field' and 'Step 4. Add to page layouts'. It shows a table with one row: 'Field Label' is 'Service Amount', 'Data Type' is 'Currency', and 'Field Name' is 'Service_Amount'. Below the table, it says 'Select the page layouts that should include this field.' and shows two checkboxes: 'Add Field' and 'Page Layout Name' (which is checked) and 'Appointment Layout' (which is also checked). At the bottom, there are buttons for 'Previous', 'Save & New', 'Save', and 'Cancel'.

Creation of CurrencyField on Billingdetails and feedbackObject :

1. Follow the same steps as mentioned above in Billingdetails and feedbackObject.
2. Change the label name as mentioned.
3. Give the Field Label : PaymentPaid
4. Field Nme : is auto populated

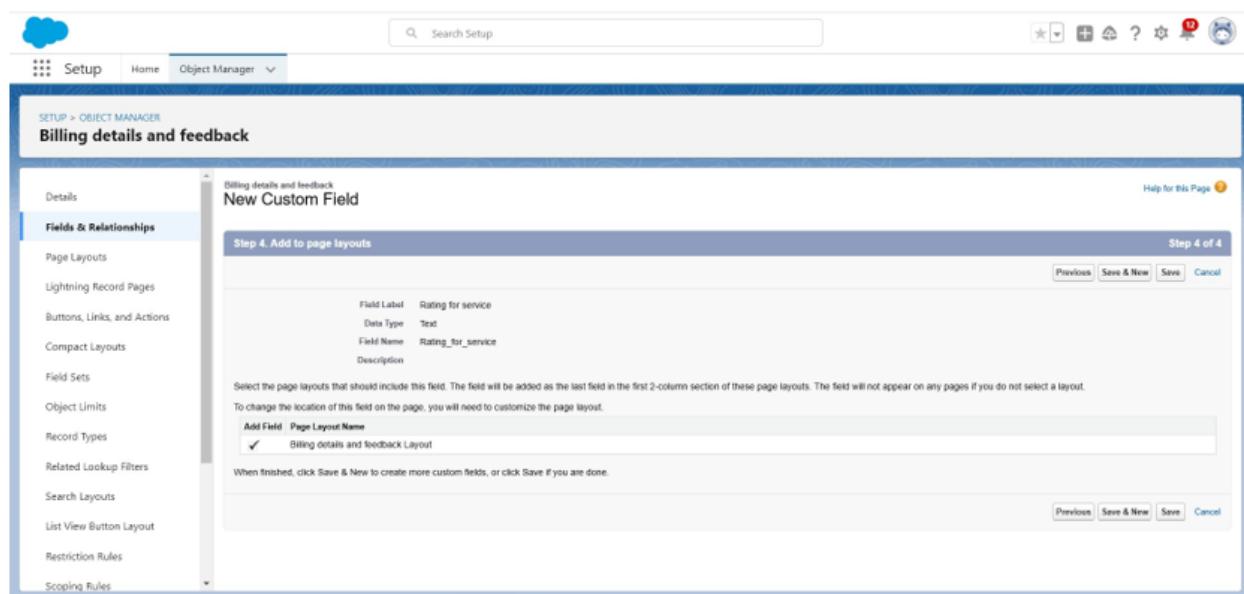
This screenshot is identical to the one above, showing the 'Appointment' object setup screen. The only difference is the object name in the title bar: 'SETUP > OBJECT MANAGER' followed by 'Billing details and feedback'. The rest of the interface, including the sidebar, field configuration, and step 4 details, remains the same.

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 FieldLabel : Vehiclenumber 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 detailsand feedback object:

1. Go to setup >> click on Object Manager >>type object name(Billing detailsand 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 and Unique.
8. Click on next >> next >> save



Creation of Picklist Fields

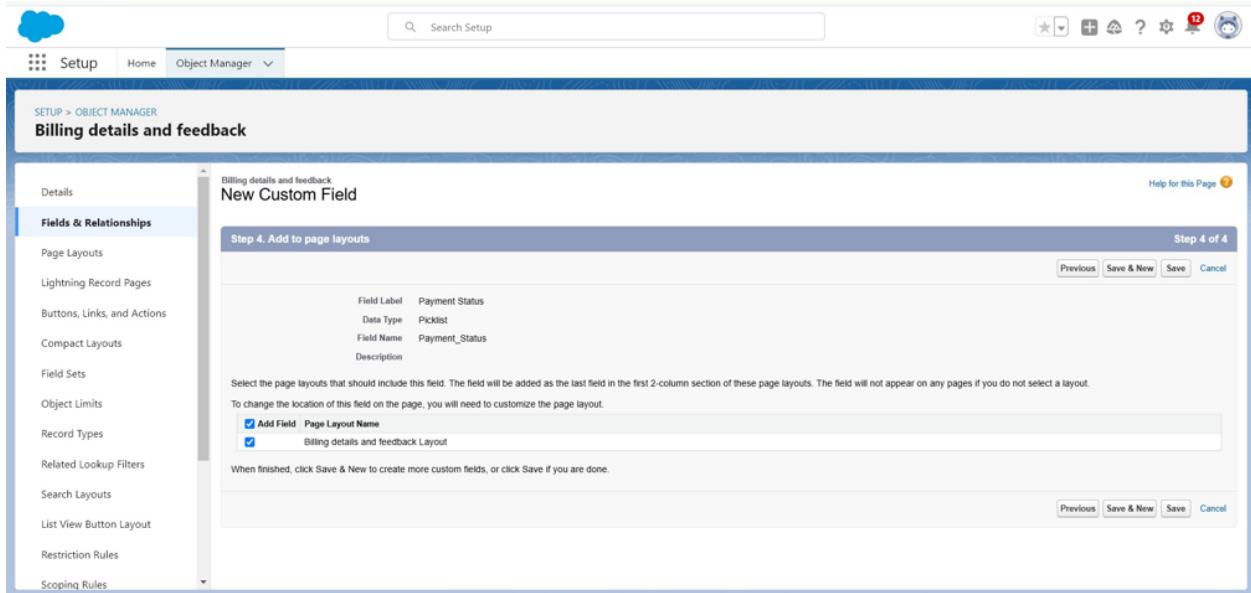
Creation of Picklist Fields in Service records object :

1. Go to setup >>click on ObjectManager >> type object name(Service records) in search bar >>click on the object.
2. Click on fields &relationship >> clickon 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 valuesare: Started, Completed.
6. Click Next.
7. Next >>Next >> Save.

Creation of Picklist Fields in Billing detailsand 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. Clickon fields &relationship >> clickon New.
3. Select Data type as "Picklist" and click Next.
4. Enter FieldLabel as "PaymentStatus", under valuesselect "Enter values,with each value separated by a new line" and enter values as shown below.

5. The values are: Pending,Completed.
6. Click Next.
7. Next >>Next >> Save.



Creating Formula Field in Service records Object

1. Go to setup >>click on ObjectManager >> type object name(Service records) in search bar >> click on the object.
2. Click on fields &relationship >> clickon New.
3. Select Data type as "Formula" and click Next.
4. Give Field Label and Field Name as "servicedate" 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 >> Sav

The screenshot shows the Salesforce Setup interface for the Object Manager. The top navigation bar includes the Setup icon, Home, and Object Manager. The search bar contains "Search Setup". The main content area displays the "Service records" object details under the "Fields & Relationships" tab. The sidebar on the left lists various setup categories: Details, Fields & Relationships (selected), Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, Search Layouts, List View Button Layout, Restriction Rules, and Scoping Rules.

Fields & Relationships
8 Items, Sorted by Field Label

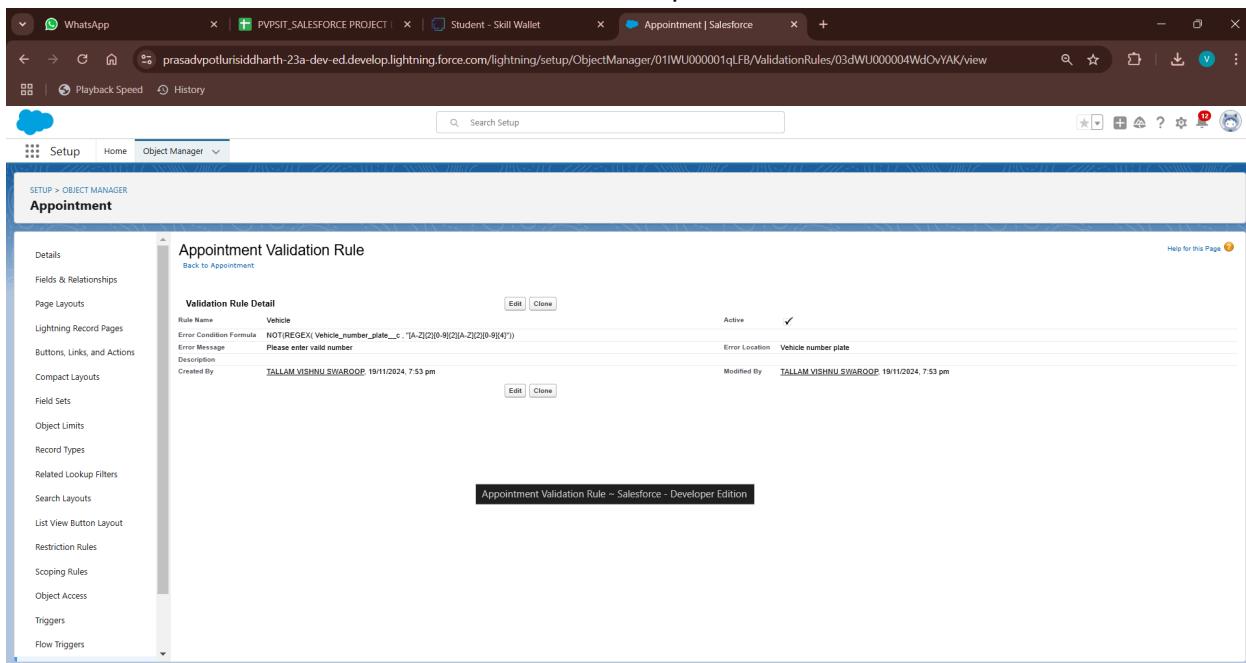
FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Appointment	Appointment_c	Lookup(Appointment)		✓
Created By	CreatedById	Lookup(User)		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		✓
Phone number	Phone_number_c	Phone		
Quality Check Status	Quality_Check_Status_c	Checkbox		
service date	service_date_c	Formula (Date)		
Service recordsName	Name	Auto Number		✓

Validation rule

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

To create a validation rule to an Appointment Object

1. Go to the setup page >> click on objectmanager >> 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 :-
 1. NOT(REGEX(Vehicle_number_plate__c , "[A-Z]{2}[0-9]{2}[A-Z]{2}[0-9]{4}"))
 6. Enter the Error Message as "Please enter valid number", select the Error location as Field and select the field as "Vehicle number plate", and click Save.



To create a validation rule to an Service recordsObject

1. Go to the setup page >> click on objectmanager >> From drop down click edit for Service records object.
2. Click on the validation rule >> click New.

3. Enter the Rule name as "service_status_note".
4. Insert the Error Condition Formulaas :-

`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.

The screenshot shows the Salesforce Setup interface for the Service records object. On the left, a sidebar lists various configuration options like Details, Fields & Relationships, Page Layouts, and Lightning Record Pages. The main content area is titled 'Service records Validation Rule' and displays the following details:

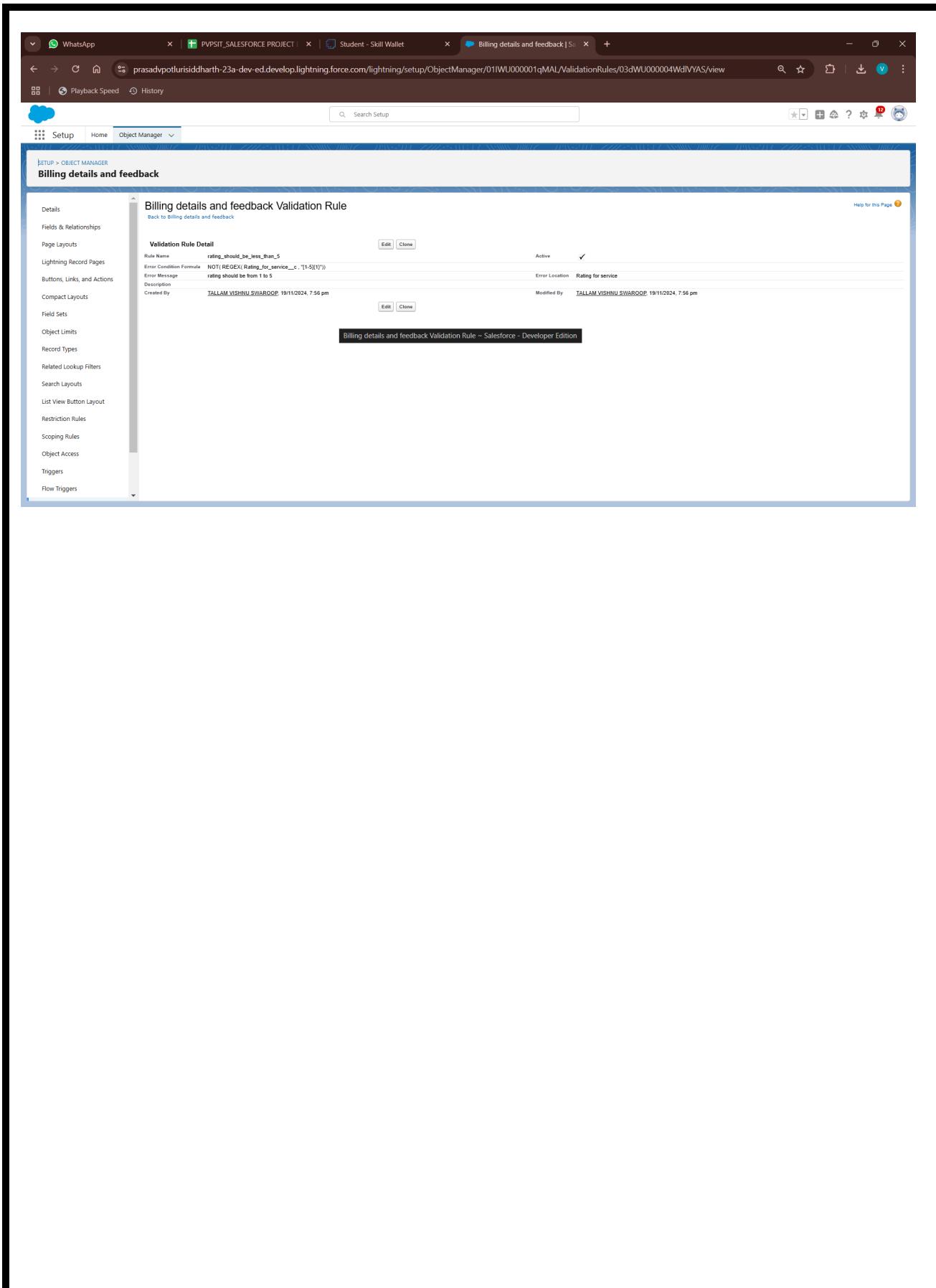
- Validation Rule Detail**
- Rule Name:** service_status_note
- Error Condition Formula:** NOT(ISPICKVAL(Service_Status__c , "Completed"))
- Error Message:** still it is pending
- Description:** (empty)
- Created By:** TALLAM VISHNU SWAROOP, 19/11/2024, 7:55 pm
- Active:** checked
- Error Location:** Service Status
- Modified By:** TALLAM VISHNU SWAROOP, 19/11/2024, 7:55 pm

To create a validation rule to an Billing detailsand feedback Object:

1. Go to the setup page >> clickon object manager>> From drop down clickedit 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 Formulaas :-

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

Enter the Error Messageas "rating shouldbe from 1 to 5", select the Error locationas Field and select the field as "Rating for Service", and click Save.



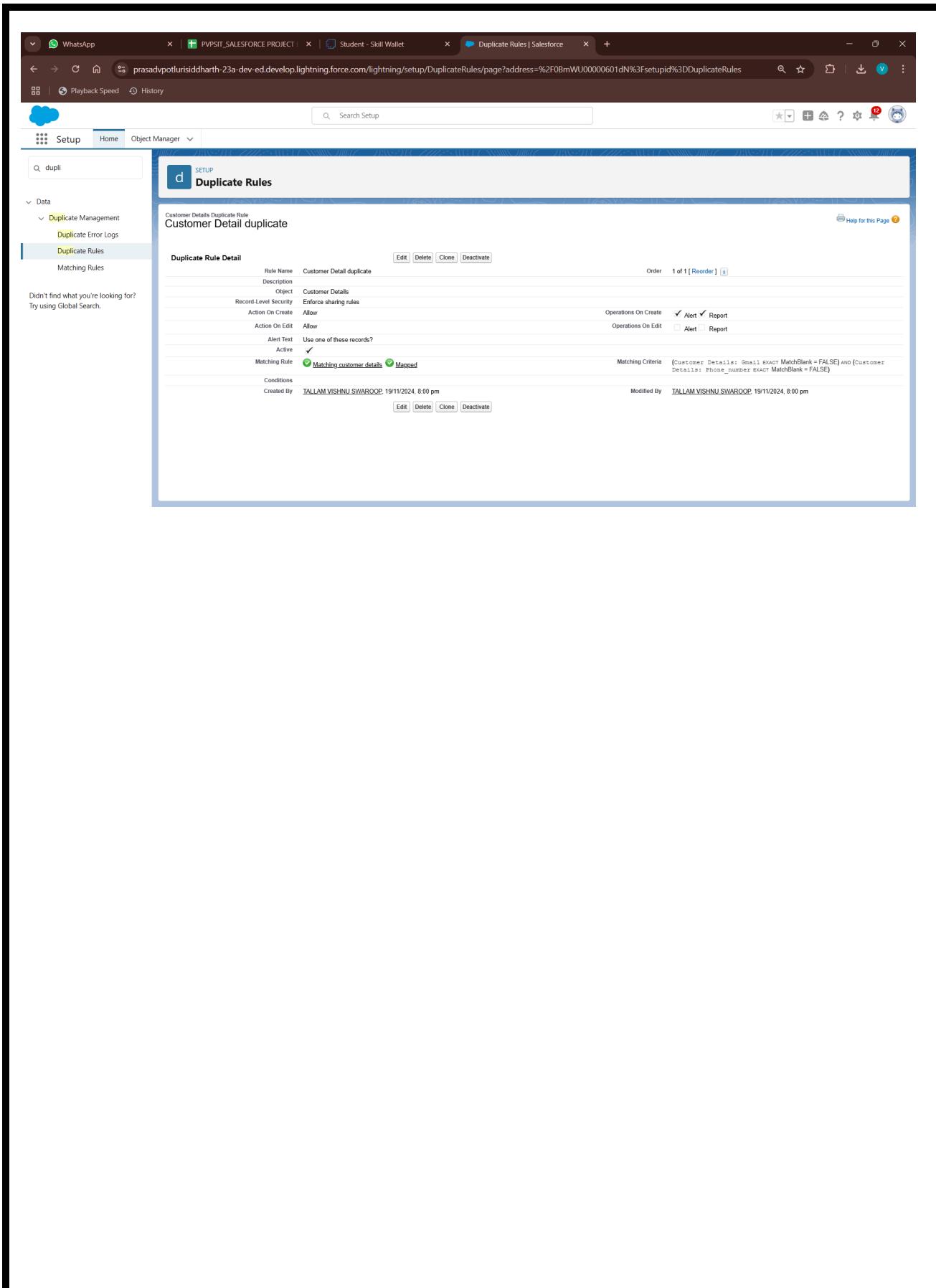
Duplicate rule

To create a matching rule to an Customer detailsObject:

1. Go to quick find box in setup and search for matching Rule.
 2. Click on matchingrule >> clickon New Rule.
 3. Select the objectas Customer detailsand click Next.
 4. Give the Rule name : Matchingcustomer details
 5. Unique name : is auto populated
 6. Define the matching criteria as
 7. Field MatchingMethod
a. Gmail Exact
b. Phone Number Exact
 8. Click save.
 9. After Saving Click on Activate.

To create a Duplicaterule to an Customer detailsObject:

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 Detailuplicate
 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.



Profiles

A profile is a group/collection of settings and permissions that define what a user can do in Salesforce. Profile controls Object permissions, Field permissions, User permissions, Tab settings, App settings, Apex class access, Visualforce page access, Page layouts, Record Types, Login hours & Login IP ranges. You can define profiles by the user's job function. For example System Administrator, Developer, Sales Representative.

Manager Profile

To create a new profile:

- a. Goto setup >> type profiles in quick find box >> click on profiles >> clone the desired profile (Standard User) >> enter profile name (Manager) >> Save.
- b. While still on the profile page, then click Edit.
- c. Select the Custom App settings as default for the Garagemanagement.
- d. 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.
- e. Changing the session times out after should be "8 hours of inactivity".
- f. Change the password policies as mentioned :
- g. User passwords expire in should be "never expires".
- h. Minimum password length should be "8", and click save.

The screenshot shows the Salesforce Profiles Manager page. At the top, there are tabs for Setup, Home, and Object Manager. The main title is "Profiles". Below the title, it says "Profile Manager" and "Users with this profile have the permissions and page layouts listed below. Administrators can change a user's profile by editing that user's personal information." A note at the bottom of this section states: "If your organization uses Record Types, use the Edit links in the Record Type Settings section below to make one or more record types available to users with this profile." There are several sections: "Enabled External Data Source Access" (No External Data Sources enabled), "Page Layouts" (Standard Object Layouts, Global, Email Application, Home Page Layout, Alternative Payment Method, Appointment Invitation, Asset, Asset Action, Asset Relationship), and "Custom Profile" (User License: Salesforce, Description: TALLAM VISHNU SWAROOP, Created By: TALLAM VISHNU SWAROOP, Modified By: TALLAM VISHNU SWAROOP). The URL in the browser bar is https://prasadvpotlurisiddharth-23a-dev-ed.develop.lightning.force.com/lightning/setup/EnhancedProfiles/page?address=%2F00eWU0000078p9N... .

Sales person Profile:

1. Goto setup >>type profiles in quick find box >>click on profiles>> clone the desiredprofile (Salesforce PlatformUser) >> enter profile name (sales person) >> Save.
2. While still on the profilepage, 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 detailsand feedback , service records and customer details objects as mentioned in the below diagram.
5. And click save.

WhatsApp | PVPSPIT SALESFORCE PROJECT | Student - Skill Wallet | Profiles | Salesforce

prasadvpotlurisiddharth-23a-dev-ed.lightning.force.com/lightning/setup/EnhancedProfiles/page?address=%2F00eWU0000078oqv

Setup Home Object Manager

Search Setup

Profile sales person

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

Profile IP Pages | Enabled Apex Class Access | Enabled Visualforce Page Access | Enabled External Data Source Access | Enabled Named Credential Access | Enabled External Credential Principal Access | Enabled Custom Metadata Type Access | Enabled Flow Access | Enabled Service Presence Status Access | Enabled Custom Permissions

Profile Detail

Name: sales person
User License: Salesforce Platform
Description:
Created By: TALLAM VISHNU SWAROOP 19/11/2024, 8:06 pm
Modified By: TALLAM VISHNU SWAROOP 19/11/2024, 9:51 pm

Page Layouts

Standard Object Layouts	
Global	Global Layout [View Assignment]
Email Application	Not Assigned [View Assignment]
Home Page Layout	Home Page Layout [View Assignment]
Account	Account Layout [View Assignment]
Alternative Payment Method	Alternative Payment Method Layout [View Assignment]
Appointment Invitation	Appointment Invitation Layout [View Assignment]
Asset	Asset Layout [View Assignment]
Asset Relationship	Asset Relationship Layout [View Assignment]
Assigned Resource	Assigned Resource Layout [View Assignment]
Associated Location	Associated Location Layout [View Assignment]

	Fulfillment Order Item Tax
Fulfillment Order Product	Fulfillment Order Item Tax Layout [View Assignment]
Idea	Fulfillment Order Product Layout [View Assignment]
Individual	Idea Layout [View Assignment]
Invoice	Individual Layout [View Assignment]
Invoice Line	Invoice Layout [View Assignment]
Lead	Invoice Line Layout [View Assignment]
Location	Lead Layout [View Assignment]
Location Group	Location Layout [View Assignment]
Location Group Assignment	Location Group Layout [View Assignment]

Role & Role Hierarchy

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

Creating Manager Role:

- a. Go to quick find >> Search for Roles >> click on set up roles.
- b. Click on Expand All and click on add role under whom this role works.
- c. Give Label as "Manager" and Role name gets auto populated. Then click on Save.

The screenshot shows the Salesforce Setup interface for creating a new role. The URL in the browser is prasadvpotlurisiddharth-23a-dev-ed.lightning.force.com/lightning/setup/Roles/page?address=%2F00EWU000006LCW%3Fsetupid%3DRoles. The page title is "SETUP Roles".

Role Detail

Label	Manager	Role Name as displayed on reports	Role Name	Manager
This role reports to	CEO	TALLAM VISHNU SWAROOP	TALLAM VISHNU SWAROOP	
Modified By	TALLAM VISHNU SWAROOP	19/11/2024, 8:11 pm		
Opportunity Access	Users in this role can edit all opportunities associated with accounts that they own, regardless of who owns the opportunities			
Case Access	Users in this role can edit all cases associated with accounts that they own, regardless of who owns the cases			

Users in Manager Role

Action	Full Name	Alias	Username	Active
Edit	Niklaus Mikaelsson	niklaus	niklaus@mikael.com	<input checked="" type="checkbox"/>

Creating another roles:

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

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

- Page Header:** WhatsApp, prasadvpotlurisiddharth-23a-dev-ed.lightning.force.com/lightning/setup/Roles/page?address=%2F00EWU000006LCh3%3Fsetupid%3DRoles
- Left Navigation Bar:** Setup, Home, Object Manager, Roles (selected), Feature Settings, Sales, Service, Case Teams, Case Team Roles, Contact Roles on Cases.
- Role Detail View:**
 - Role Name:** sales person
 - Label:** sales person
 - This role reports to:** Manager
 - Modified By:** TALLAM VISHNU SWAROOP 19/11/2024, 8:12 pm
 - Opportunity Access:** Users in this role can edit all opportunities associated with accounts that they own, regardless of who owns the opportunities
 - Case Access:** Users in this role can edit all cases associated with accounts that they own, regardless of who owns the cases
- Users in sales person Role:** A table listing users assigned to the role:

Action	Full Name	Alias	Username	Active
Edit	anand malhotra	anand	anand@gmail.com	✓
Edit	gajay kumar	akumar	gajaykum.com	✓
Edit	mahesh babu	mbabu	mb@mbmsd.com	✓

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 access to the company's records. Every user in Salesforce has a user account. The user account identifies the user, and the user account settings determine what features and records the user can access.

Create User:

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

The screenshot shows the Salesforce Setup interface for creating a new user. The user is named 'Niklaus Mikaelson' with the alias 'niklaus'. The email is 'niklausearop01@gmail.com' (verified). The username is 'niklaus@mikael'. The nickname is 'nick'. The role is set to 'Manager'. The profile is 'Salesforce Manager'. The user is active. The user detail section includes fields for Company, Department, Division, Address, Time Zone (GMT+05:30 India Standard Time (Asia/Kolkata)), Locale (English (India)), Language (English), Delegated Approver (Manager), and Receive Approval Request Emails (Only if I am an approver). The federation ID is listed as 'Federation ID'. Under 'Feature Settings', there are sections for Data.com (Prospector Users), Service (Embedded Service: Messaging for In-App and Web User Verification), and User Interface (Action Link Templates, Actions & Recommendations, App Menu, Custom Labels, Density Settings). The right side of the screen displays a list of available profiles and other user-related settings.

creating another users:

1. Repeat the steps and create another user using
 - a. Role : sales person
 - b. User licence : Salesforce Platform
 - c. Profile : sales person

Note : create at least 3 users with these permissions.

The screenshot shows the Salesforce Setup interface under the 'Users' section. The left sidebar includes categories like User Management Settings, Feature Settings, Service, and User Interface. The main content area displays a table of users with columns for Action, Full Name, Alias, Username, Role, Active, and Profile. The table lists several users, each with their respective details and assigned roles such as salesperson, Chatter Free User, salesperson, Manager, and System Administrator.

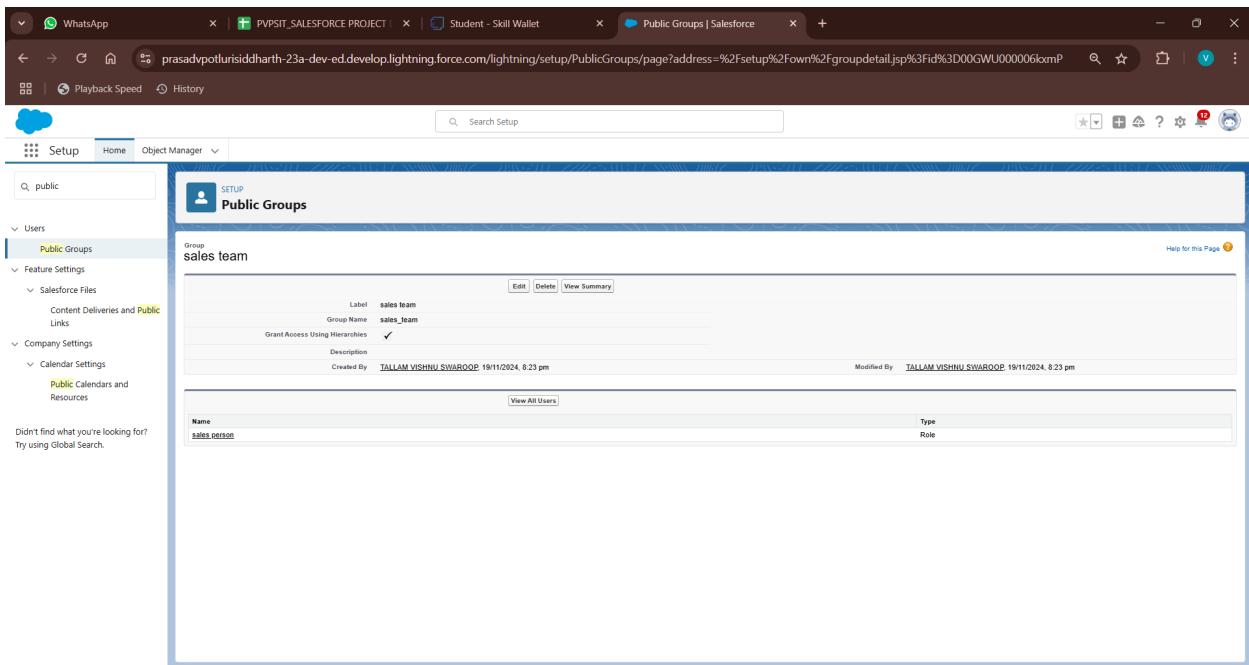
Action	Full Name	Alias	Username	Role	Active	Profile
<input type="checkbox"/>	babu_mahesh	mbabu	mb@mhmed.com	salesperson	<input checked="" type="checkbox"/>	sales person
<input type="checkbox"/>	Chatter_Freed	Chatter	chatty@0duv00000pbh2x1mbphbz5se@chatter.salesforce.com		<input checked="" type="checkbox"/>	Chatter Free User
<input type="checkbox"/>	Kumar_alex	akumar	alex@km.com	salesperson	<input checked="" type="checkbox"/>	sales person
<input type="checkbox"/>	mahota_anand	anand	anand@gmail.com	salesperson	<input checked="" type="checkbox"/>	sales person
<input type="checkbox"/>	Mikelson_Niklaus	niklaus	niklaus@mail.com	Manager	<input checked="" type="checkbox"/>	Manager
<input type="checkbox"/>	User_Interation	intero	interation@0duv00000pcpb2x3.com		<input checked="" type="checkbox"/>	Analytics Cloud Integration User
<input type="checkbox"/>	User_Security	sec	insightssecurity@0duv00000pcpb2x3.com		<input checked="" type="checkbox"/>	Analytics Cloud Security User
<input type="checkbox"/>	VISHNU SWAROOP TALLAM	TViSH	vishnu@payall.com		<input checked="" type="checkbox"/>	System Administrator

Public groups

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

Creating New Public Group:

1. Go to setup >>type users in quick find box >>select public groups>> click New.
2. Give the Label as “sales team”.
3. Group name is autopopulated.
4. Search for Roles.
5. In Available Members select Sales person and click on add it will be moved to selected member.
6. Click on save.



Sharing Setting

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

Organization-Wide Default (OWD) Settings:

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

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

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

Role Hierarchy:

Salesforce uses a role hierarchy to determine record access.

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

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

Profiles and Permission Sets:

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

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

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 ServiceRecords Object to private as shown in fig.
3. Click on save and refresh.
4. Scroll down a bit, Click new on ServiceRecords sharing Rules.
5. Give the Label name as "Sharingsetting"
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.

The screenshot shows the Salesforce Sharing Settings page. The URL is <https://prasadypoturisiddharth-23a-dev-ed.lightning.force.com/lightning/setup/SecuritySharing/home>. The page title is "Sharing Settings". A search bar at the top right contains "Search Setup". The left sidebar has a "Sharing" section with "Sharing Settings" selected. The main content area displays the "Organization-Wide Defaults" table. The table has columns for "Object", "Default Internal Access", "Default External Access", and "Grant Access Using Hierarchies". The "Default Internal Access" column includes options like "Public Read/Write", "Controlled by Parent", and "Private". The "Default External Access" column includes "Private", "Controlled by Parent", and "Controlled by Campaign". The "Grant Access Using Hierarchies" column contains checkmarks. The table lists various objects such as Lead, Account and Contact, Contact, Order, Asset, Opportunity, Case, Campaign, Campaign Member, User, Activity, Calendar, Price Book, Product, Individual, and Voice Call.

Object	Default Internal Access	Default External Access	Grant Access Using Hierarchies
Lead	Public Read/Write	Private	✓
Account and Contact	Public Read/Write	Private	✓
Contact	Controlled by Parent	Controlled by Parent	✓
Order	Controlled by Parent	Controlled by Parent	✓
Asset	Controlled by Parent	Controlled by Parent	✓
Opportunity	Public Read/Write	Private	✓
Case	Public Read/Write/Transfer	Private	✓
Campaign	Public Full Access	Private	✓
Campaign Member	Controlled by Campaign	Controlled by Campaign	✓
User	Public Read Only	Private	✓
Activity	Private	Private	✓
Calendar	Hide Details and Add Events	Hide Details and Add Events	✓
Price Book	Use	Use	✓
Product	Public Read/Write	Public Read/Write	✓
Individual	Public Read/Write	Private	✓
Voice Call	Private	Private	✓

Flows

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.
3. Select the Objectas "Billing detailsand feedback" in the Drop down list.
4. Select the Trigger Flow when: "A record is Created or Updated".
5. Select the Optimizethe 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". Give the Label Name : Amount Update
7. Api name : is auto populated
8. Set a filtercondition : All Conditions are met(AND)
9. Field : Payment_Status_c
10. Operator : Equals
11. Value : Completed
12. And Set FieldValues for the Billing detailsand feedback Record
13. Field : Payment_Paid_c
14. Value : {\$Record.Service_records_r.Appointment_r.Service_Amount_c}
15. Click On Done.Before creating another Element.Create a New Resource form Toolbox form top left.
16. Click on the New Resource, And select Variable.
17. Select the resourcetype as text template.
18. Enter the API name as " alert".
19. Change the view as Rich Text ? View to PlainText.
20. In body field paste the syntax that given below.

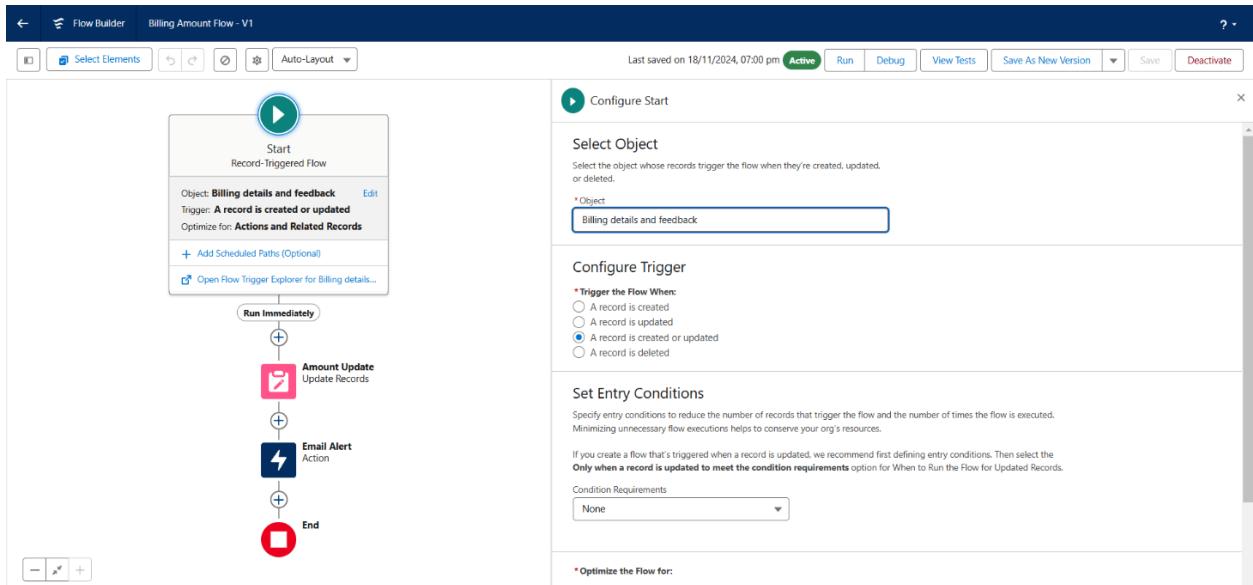
Dear {\$Record.Service_records_r.Appointment_r.Customer_Name_r.Name},

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 .

1. Click done.
2. Now Click on Add Element, select Action.
3. Their action bar will be opened in that search for “ send email ” and click on it.
4. Give the labelname as “ Email Alert”
5. API name will be auto populated.
6. Enable the body in set inputvalues for the selected action.
7. Select the text template that created , Body : {!alert}
8. Include recipient addresslist select the email form the record.
9. RecipientAddressList:
{!\$Record.Service_records__r.Appointment__r.Customer_Name__r.Gmail__c}
10. Include subject as “ Thank You for Your Payment - Garage Management”.
11. Click done.
12. Click on save. Give the Flow label , Flow Api name will be autopopulated.
13. And click save, and click on activate.



Apex Trigger

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

A trigger is Apex code that executes before or after the following types of operations:

- a. insert
- b. update
- c. delete
- d. merge
- e. upsert
- f. undelete

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

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

There are primarily two types of Apex Triggers:

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

```
AmountDistribution.apxt * | AmountDistributionHandler.apxc *
Code Coverage: None | API Version: 58 | Go X
1 * public class AmountDistributionHandler {
2
3 *     public static void amountDist(list<Appointment__c> listApp){
4         list<Service_records__c> serList = new list <Service_records__c>();
5
6     for(Appointment__c app : listApp){
7         if(app.Maintenance_service__c == true && app.Repairs__c == true && app.Replacement_Parts__c == true){
8             app.Service_Amount__c = 10000;
9         }
10        else if(app.Maintenance_service__c == true && app.Repairs__c == true){
11            app.Service_Amount__c = 5000;
12        }
13        else if(app.Maintenance_service__c == true && app.Replacement_Parts__c == true){
14            app.Service_Amount__c = 8000;
15        }
16        else if(app.Repairs__c == true && app.Replacement_Parts__c == true){
17            app.Service_Amount__c = 7000;
18        }
19        else if(app.Maintenance_service__c == true){
20            app.Service_Amount__c = 2000;
21        }
22        else if(app.Repairs__c == true){
23            app.Service_Amount__c = 3000;
24        }
25        else if(app.Replacement_Parts__c == true){
26            app.Service_Amount__c = 5000;
27        }
28    }
29 }
30 }
31 }
```

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

Code:

```
public class AmountDistributionHandler {

    public static void amountDist(list<Appointment__c> listApp){

        list<Service_records__c> serList = new list <Service_records__c>();

        for(Appointment__c app : listApp){

            if(app.Maintenance_service__c == true && app.Repairs__c == true &&
app.Replacement_Parts__c == true){

                app.Service_Amount__c = 10000;
```

```
}

else if(app.Maintenance_service_c == true && app.Repairs_c == true){

    app.Service_Amount_c = 5000;

}

else if(app.Maintenance_service_c == true && app.Replacement_Parts_c == true){

    app.Service_Amount_c = 8000;

}

else if(app.Repairs_c == true && app.Replacement_Parts_c == true){

    app.Service_Amount_c = 7000;

}

else if(app.Maintenance_service_c == true){ app.Service_Amount_c = 2000;

}

else if(app.Repairs_c == true){

    app.Service_Amount_c = 3000;

}

else if(app.Replacement_Parts_c == true){

    app.Service_Amount_c = 5000;

}

}

}

}
```

Apex handler

UseCase : This use case works for Amount Distribution for each Service the customer selected for their 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".

Trigger Handler :

How to create a new trigger:

1. While still in the trailhead account, navigate to the gear icon in the top right corner.
2. Click on developer console and you will be navigated to a new console window.
3. Click on File menu in the tool bar, and click on new? Trigger.
4. Enter the trigger name and the object to be triggered.
5. Name : AmountDistribution
6. sObject : Appointment_c

Syntax For creating trigger :

The syntax for creating trigger is :

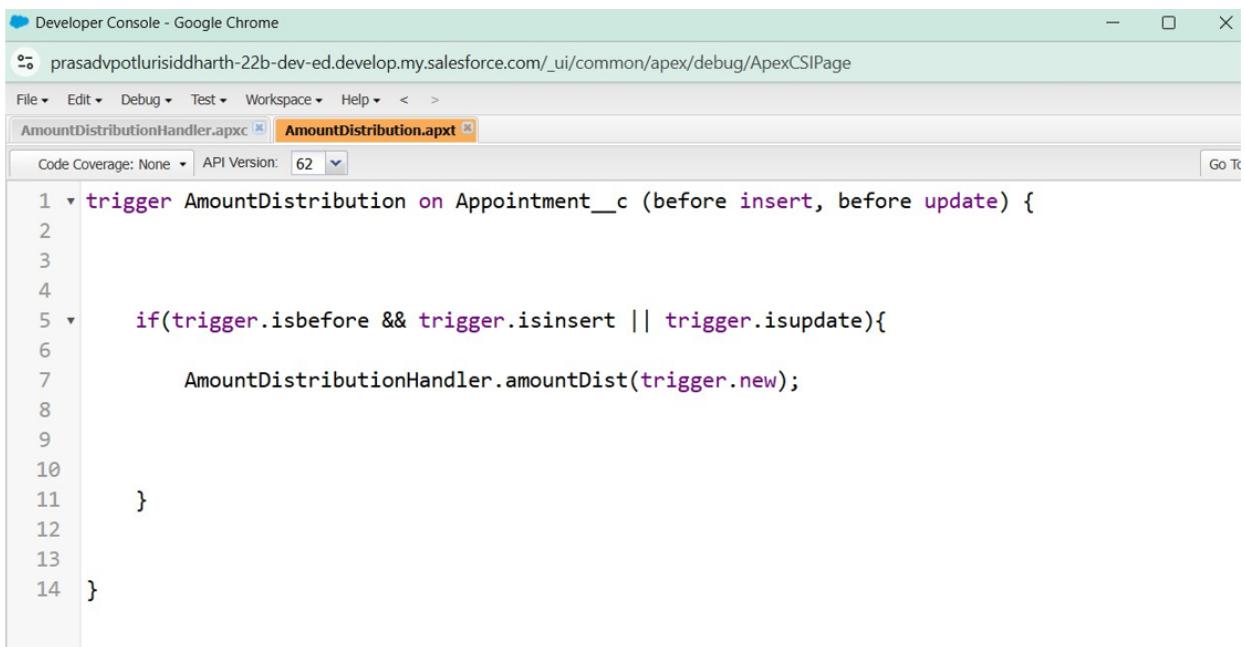
Trigger [triggername] on [objectname](Before/After event)

```
{  
}
```

In this project, trigger is called whenever the particular record sum exceeds 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);  
}  
}
```



The screenshot shows the Salesforce Developer Console interface. The title bar reads "Developer Console - Google Chrome". Below it, the URL is "prasadvpotlurisiddharth-22b-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage". The menu bar includes "File", "Edit", "Debug", "Test", "Workspace", "Help", and navigation icons. A tab bar at the top has two tabs: "AmountDistributionHandler.apxc" and "AmountDistribution.apxt", with "AmountDistribution.apxt" being the active tab. Below the tabs, there are dropdowns for "Code Coverage: None" and "API Version: 62". A "Go To" button is also present. The main content area displays the Apex trigger code:

```
1 trigger AmountDistribution on Appointment__c (before insert, before update) {  
2  
3  
4  
5     if(trigger.isbefore && trigger.isinsert || trigger.isupdate){  
6         AmountDistributionHandler.amountDist(trigger.new);  
7  
8  
9  
10    }  
11  
12  
13}  
14 }
```

Reports

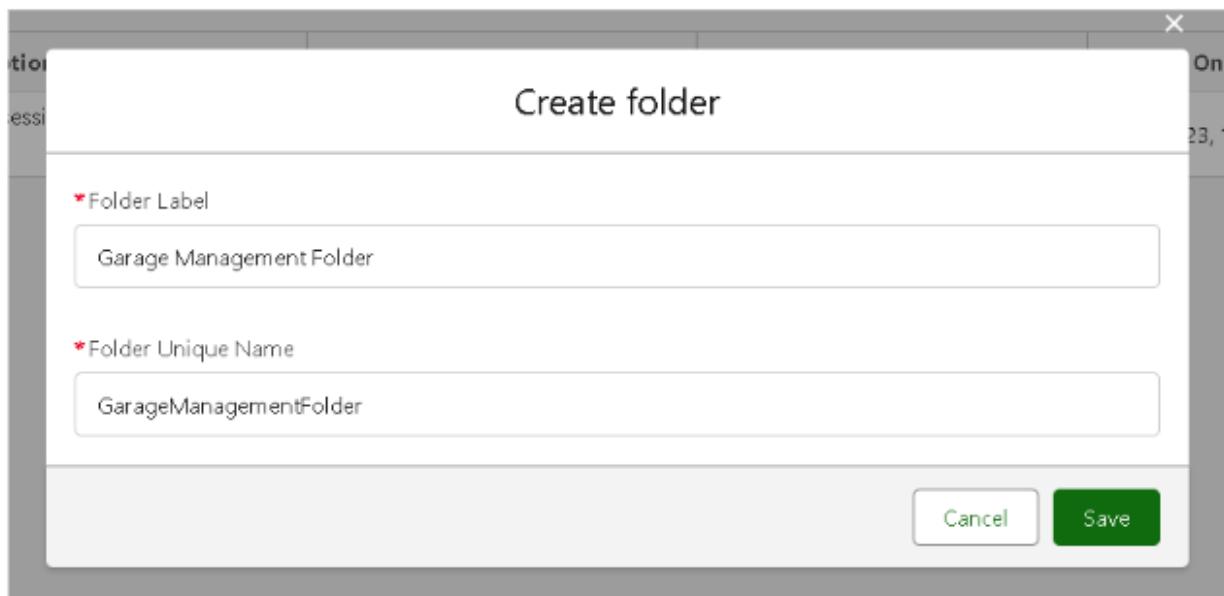
Reports give you access to your Salesforce data. You can examine your Salesforce data in almost infinite combinations, display it in easy-to-understand formats, and share the resulting insights with others. Before building, reading, and sharing reports, review these reporting basics.

Types of Reports in Salesforce

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

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 Folderlabel as "GarageManagement Folder", Folderunique name will be auto populated.
4. Click save.



Sharing a report folder:

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

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 autopopulated.
6. Keep the Description as same.
7. Select Store in Category as "other Reports"
8. Select the deployment status as "Deployed", click on Next.
9. now, Click on Related object box.
10. Click on Select Object, choose Appointment Object as shown in fig
11. Again Click to relate another object.
12. And select the related object as "service records".
13. Repeat the process and select the related object as "Billing details and feedback".
14. And click on save

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

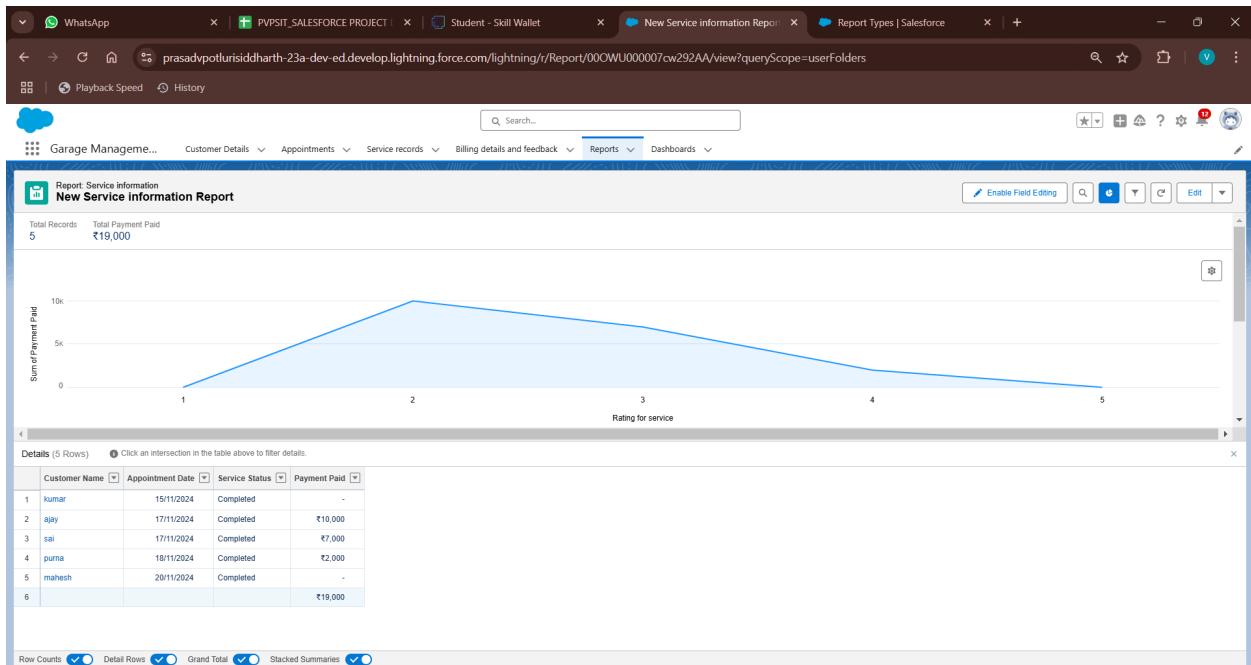
- Page Header:** WhatsApp, PVPST_PROJECT, Student - Skill Wallet, Recent | Reports | Salesforce, Report Types | Salesforce.
- Left Sidebar:** Feature Settings, Analytics, Reports & Dashboards (selected), Report Types, Reporting Snapshots, Reports and Dashboards, Settings, Security, Guest User Sharing Rule Access Report.
- Search Bar:** Search Setup.
- Main Content:**
 - Report Type Definition:** Custom Report Type, Service information. Fields: Report Type Label (Service information), Report Type Name (Service_information), Description (description), Created By (TALLAM VISHNU SWAROOP 19/11/2024, 10:46 pm).
 - Object Relationships:** Shows relationships between Customer Details (A), Appointments (B), Service records (C), and Billing details and feedback (D). A Venn diagram illustrates the overlaps between these objects.
 - Fields Available for Reports:** Shows fields from various objects like Service records, Appointments, and Customer Details, with a list of selected fields (11, 13, 10, 16).

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.
 - a. Customer name
 - b. Appointment Date
 - c. Service Status
 - d. Payment paid
 - e. Remove the unnecessary fields.
 - f. Select the fields that mentioned below in GROUP ROWS section.
 - i. Rating for Service

- g. Select the fields that mentioned below in GROUP ROWS section.
- i. Payment Status
- h. Click on Add Chart , Select the Line Chart.
- i. Click on save, Give the reportName : New Service information Report
- j. Report unique Name is auto populated.
- k. Select the folder the created and Click on save.

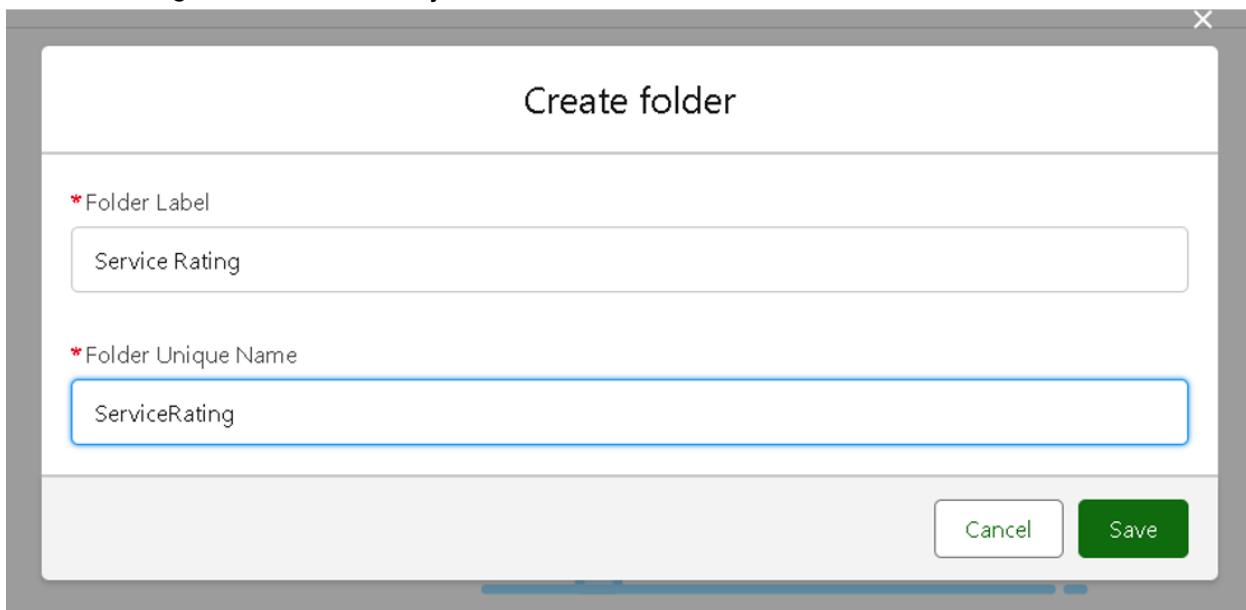


Dashboards

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 identify trends, sort out quantities, and measure the impact of their activities. Before building, reading, and sharing dashboards, review these dashboard basics.

Create Dashboard Folder:

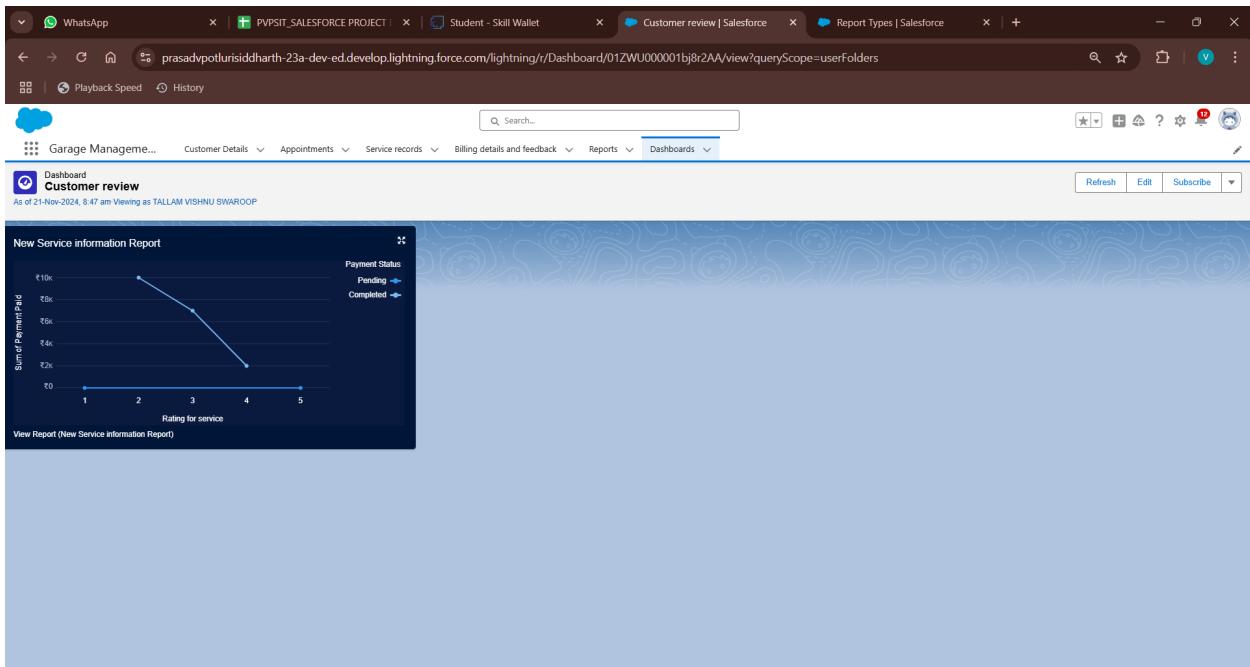
1. Click on the app launcher and search for dashboard.
2. Click on dashboard tab.
3. Click new folder, give the folder label as "Service Rating dashboard".
4. Folder unique name will be auto populated.
5. Click save.
6. Follow the same steps, from milestone 15, and activity2, and provide the sharing settings for the folder that just created.



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.



Subscription:

1. After that Click on Subscribeon top right.
2. Set the Frequency as “weekly” .
3. Set a day as monday.
4. And Click on save.

Edit Subscription

Schedule dashboard refreshes and subscribe to receive results.

Settings

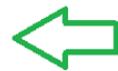
Frequency

Daily Weekly Monthly



Days

Sun Mon Tue Wed Thu Fri Sat



Time

3:00 pm

Recipients

Receive new results by email when dashboard is refreshed. i

Send email to

Me

[Edit Recipients](#)

[Cancel](#)

[Save](#)