Atisha Shrivas

Garage Management system

1. Project Overview

Garage Management System is a software platform designed to optimize the operations of automotive service centers and garages. It facilitates the management of key business areas such as vehicle service records, customer details, inventory, and billing. The system supports scheduling appointments, tracking repairs, managing spare parts, and creating invoices. Common features include real-time status updates, maintenance reminders, and reporting tools for business analysis. Many systems also integrate with payment solutions and customer communication platforms. By streamlining processes, it minimizes manual errors, boosts efficiency, and enhances customer satisfaction in garage management.

2. Objectives

Business Goals

- **Elevate Customer Satisfaction:** Provide a seamless, personalized experience by simplifying service booking, real-time updates, and feedback collection.
- **Optimize Operational Efficiency:** Streamline workflows and minimize manual tasks through automation of key processes like invoicing, scheduling, and inventory management.
- **Drive Revenue Growth:** Maximize resource utilization and capitalize on upselling opportunities with actionable insights into customer needs and service trends.
- **Enhance Staff Productivity:** Equip employees with tools to efficiently manage tasks and access critical information, reducing downtime and errors.
- **Strengthen Market Competitiveness:** Empower garages with advanced, reliable tools to excel in a competitive automotive service landscape.

Specific Outcomes

- **Centralized Management System**: A unified platform to oversee bookings, service history, customer details, and inventory.
- **Customizable Dashboard**: Real-time visualization of revenue, service metrics, and customer interactions for informed decision-making.
- **Process Automation**: Automated follow-ups for services, overdue payments, and inventory replenishment to reduce manual workload.
- **Salesforce Integration**: Seamless synchronization with Salesforce CRM for enhanced customer relationship management and streamlined data handling.
- **User-Friendly Interface**: A simple, intuitive design that minimizes training requirements and ensures ease of use for staff and customers alike.
- **Advanced Reporting**: Comprehensive performance analytics, including metrics like customer retention, average service duration, and inventory turnover.
- **Mobile Accessibility**: Full functionality on mobile devices for efficient management on the go.
- **Robust Security and Compliance**: Strong data protection measures to safeguard customer information and ensure adherence to industry regulations.

3. Salesforce Key Features and Concepts Utilized

Reporting and Dashboards

In the Garage Management System, Reports and Dashboards provide insights into operational performance and key metrics:

Reports: Allow users to track revenue trends, monitor the number of service requests completed, or evaluate customer retention rates. For example, a report can display the total revenue generated by service type within a specific timeframe.

Dashboards: Offer a visual representation of reports using charts and graphs. For instance, a dashboard may include a pie chart showing the distribution of service requests by status (e.g., Completed, Pending) or a bar graph illustrating monthly revenue growth. These tools help managers make data-driven decisions and identify areas for improvement.

Cross-Object Formula Field

Cross-object formula fields allow the system to display or calculate data across related objects.

Example: A formula field on the Vehicle object can pull the Customer Name from the Customer object, making it easy for garage staff to view customer details directly from the vehicle record. This enhances user convenience and minimizes navigation between records.

Validation Rules

Validation rules ensure data integrity by enforcing specific conditions before saving records.

Example: A validation rule on the Service Request object could ensure that the "Service Date" must not be in the past. Similarly, another rule might check that the "Estimated Cost" is greater than zero to avoid incomplete or incorrect entries.

Permission Sets

Permission Sets provide additional access permissions to users without altering their base profiles.

Example: In the Garage Management System, a service technician might need temporary access to view inventory data for spare parts. By assigning a permission set, they can gain this access

without modifying their default profile permissions, ensuring flexibility and security.

These features work together to enhance the Garage Management System's functionality, ensuring accuracy, efficiency, and user convenience while maintaining data security and compliance.

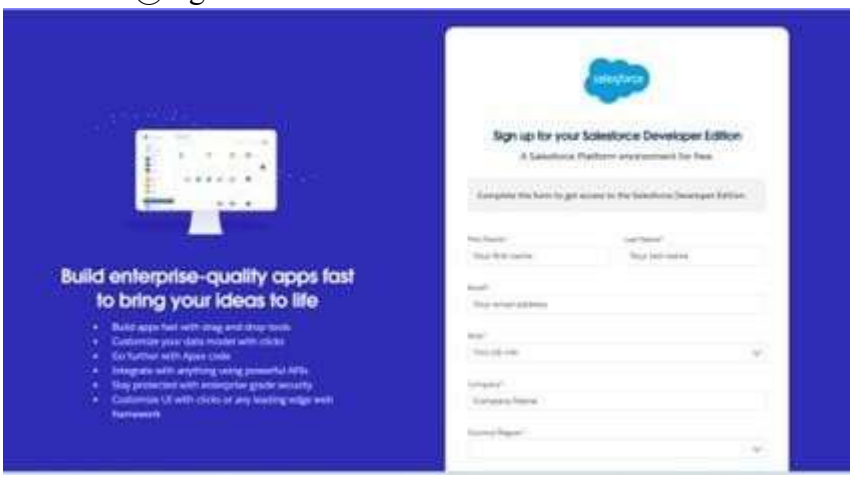
4. Detailed Steps to Solution Design

Creating Developer Account and Account activation.

Steps:

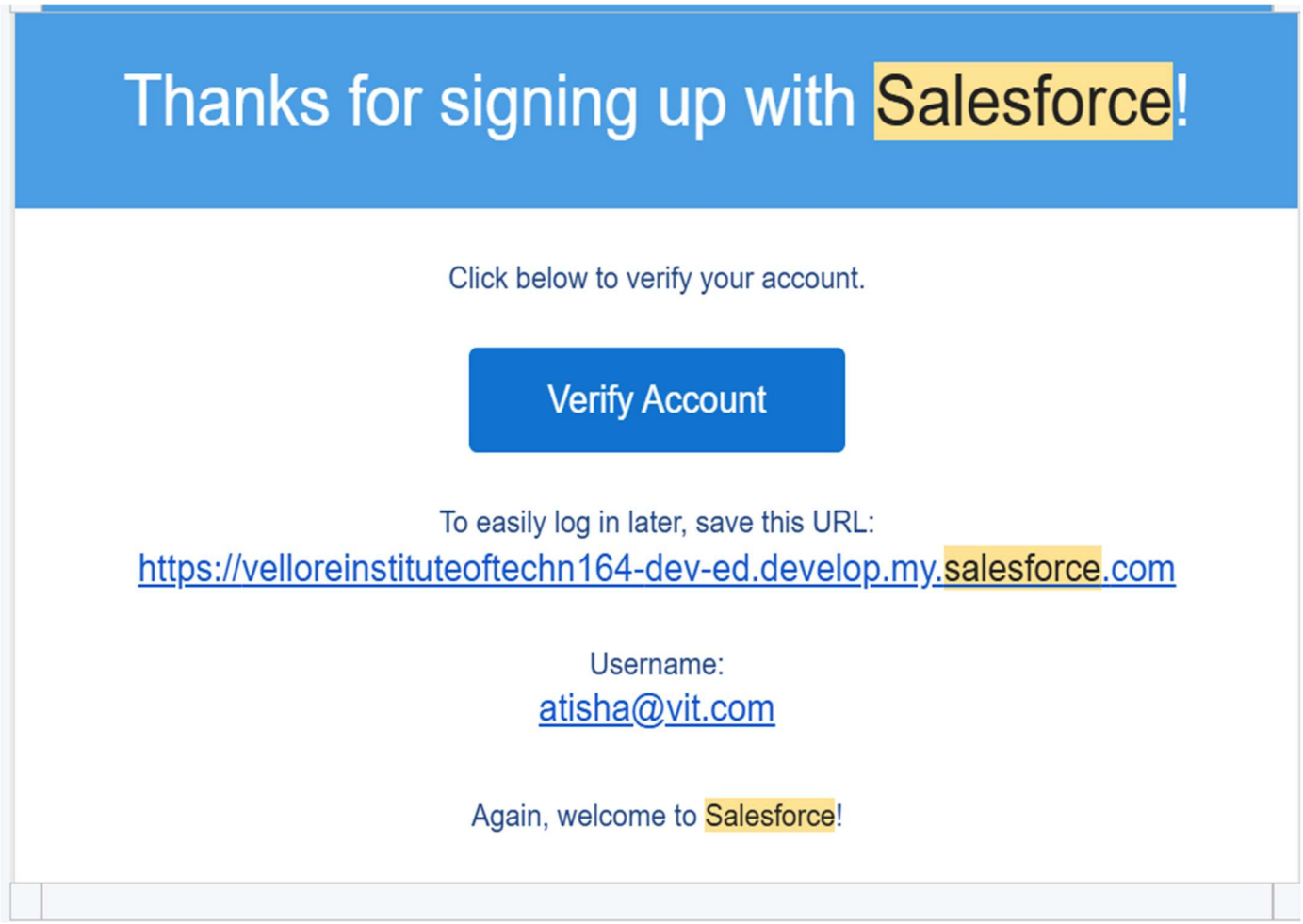
- On the sign up form, enter the following details
- Click on sign me up after filling these.
- First name & Last name
- Email
- Role : Developer
- Company : College Name
- Country : India
- Postal Code : pin code
- Username : should be a combination of your name and company
- This need not be an actual email id, you can give anything in the format.

  username@organization.com



ACTIVATION :

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



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.

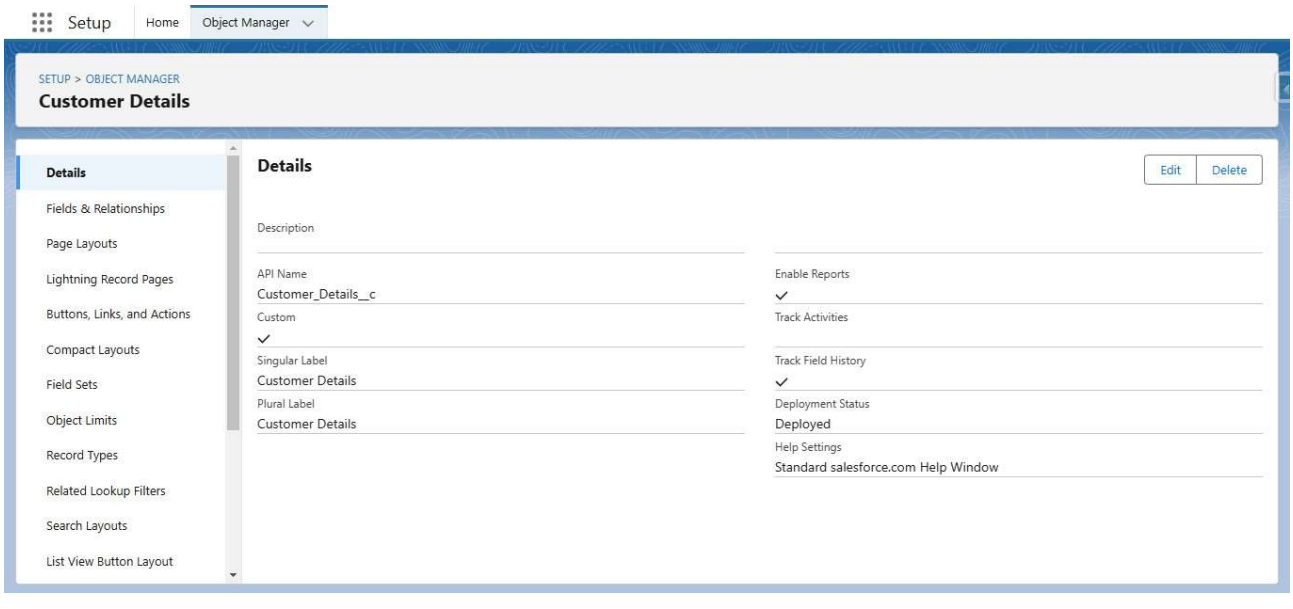
2. **Custom Objects**: Custom objects are those objects that are created by users. They supply information that is unique and essential to their organization, are the heart of any application, and provide a structure for data.

Steps:

Create Customer Details Object:

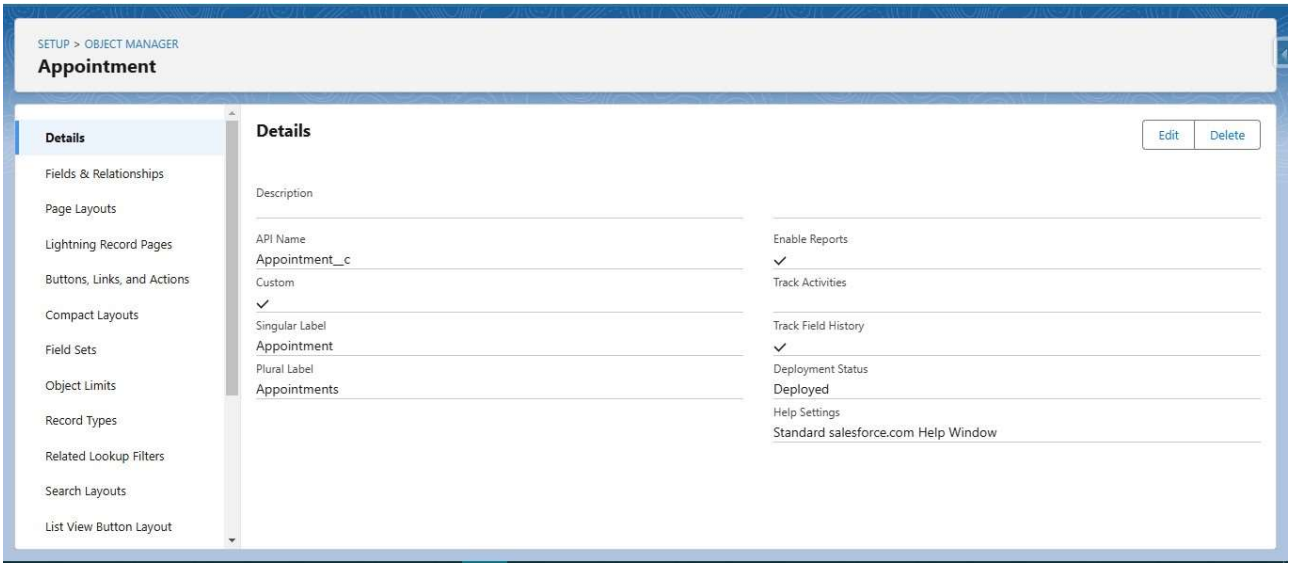
- From the setup page >> Click on Object Manager >> Click on Create >> Click on Custom Object.
- Enter the label name >> Customer Details
- Plural label name >> Customer Details
- Enter Record Name Label and Format
- Record Name >> Customer Name

- Data Type >> Text
- Click on Allow reports and Track Field History,
- Allow search >> Save.



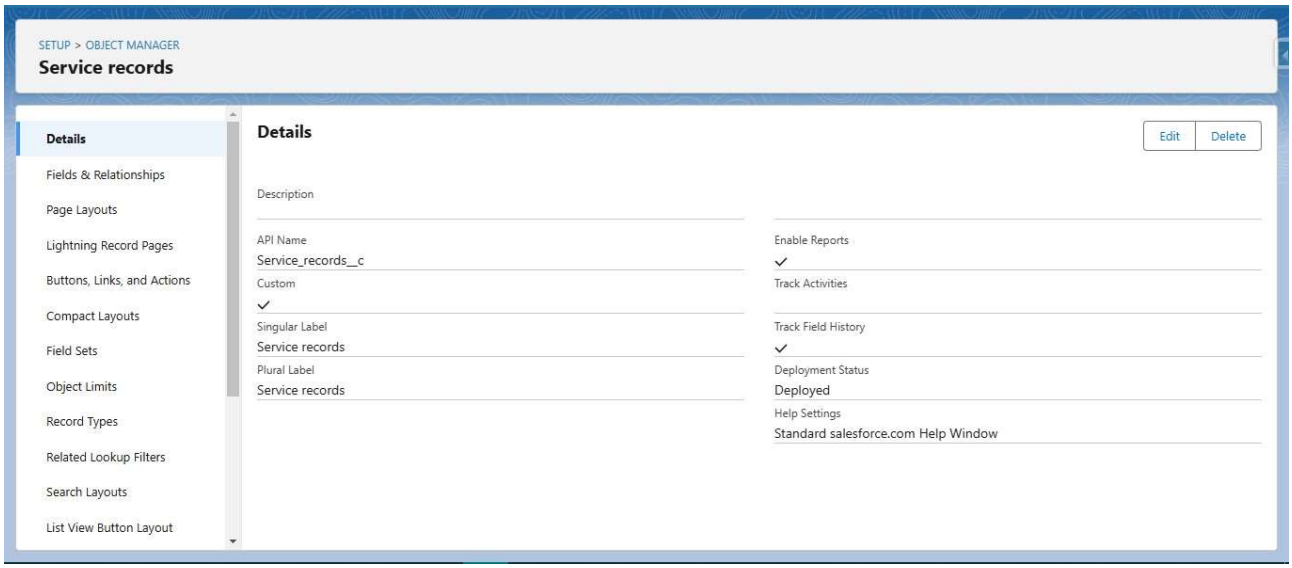
Create Appointment Object:

- 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.



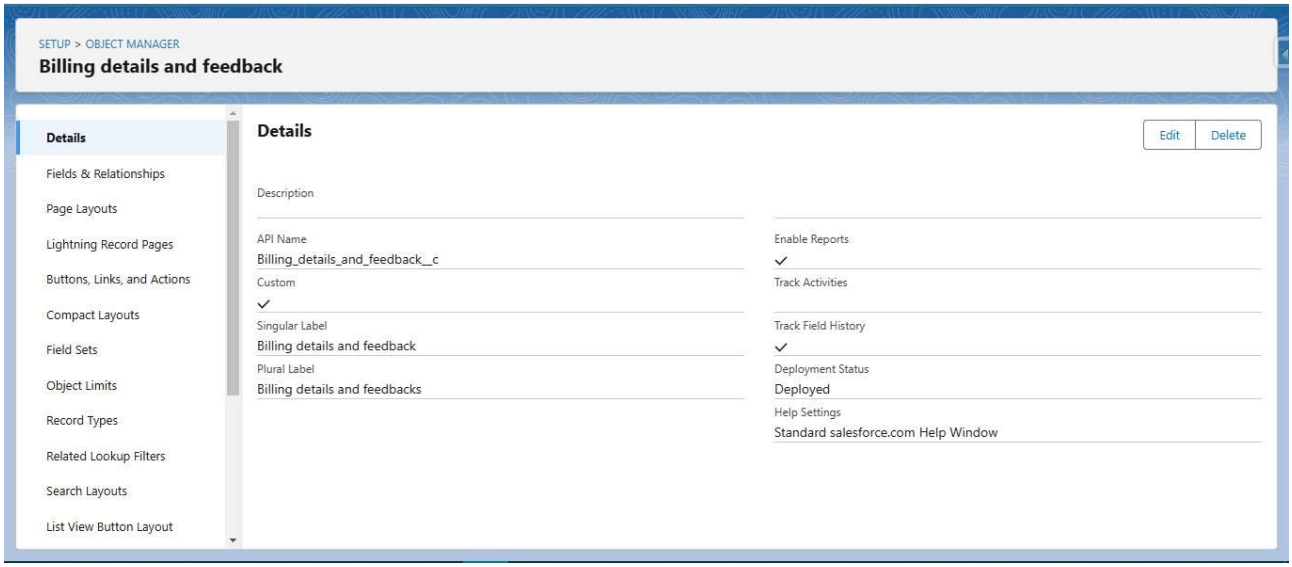
Create Service records Object:

- 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 recordsName
- 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:

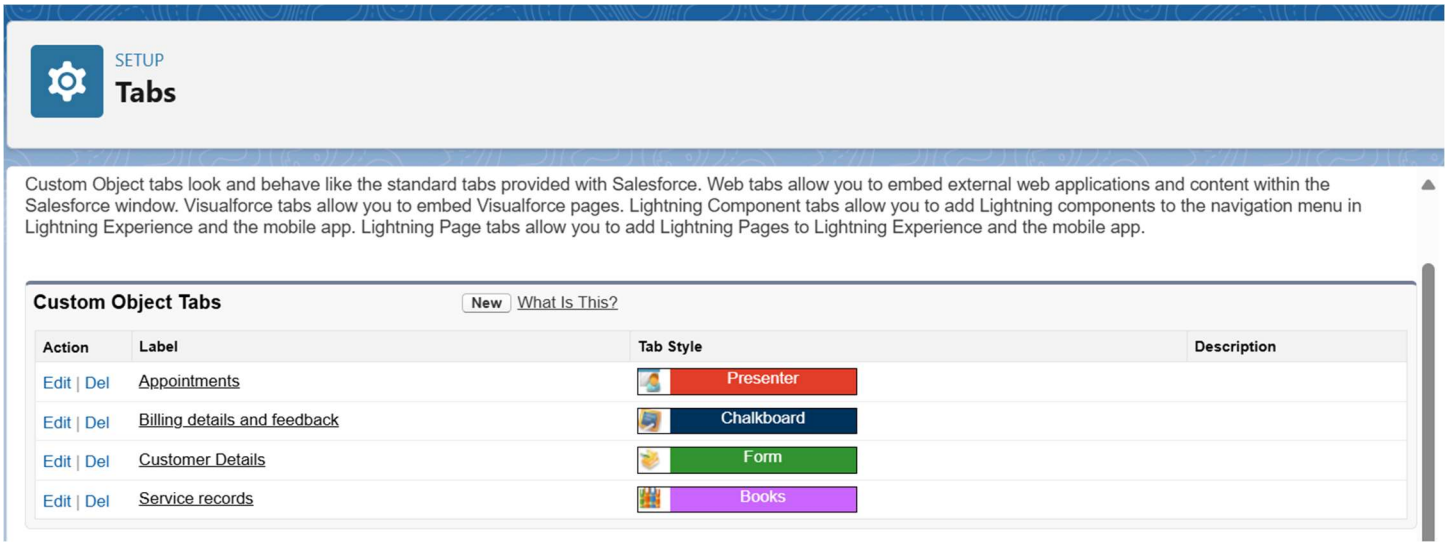
- To create an object:
- From the setup page >> Click on Object Manager >> Click on Create >> Click on Custom Object.
- Enter the label name >> Billing details and feedback
- Plural label name >> Billing details and feedback
- Enter Record Name Label and Format
- Record Name >> Billing details and feedbackName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



Tabs

Creating a Custom Tab:

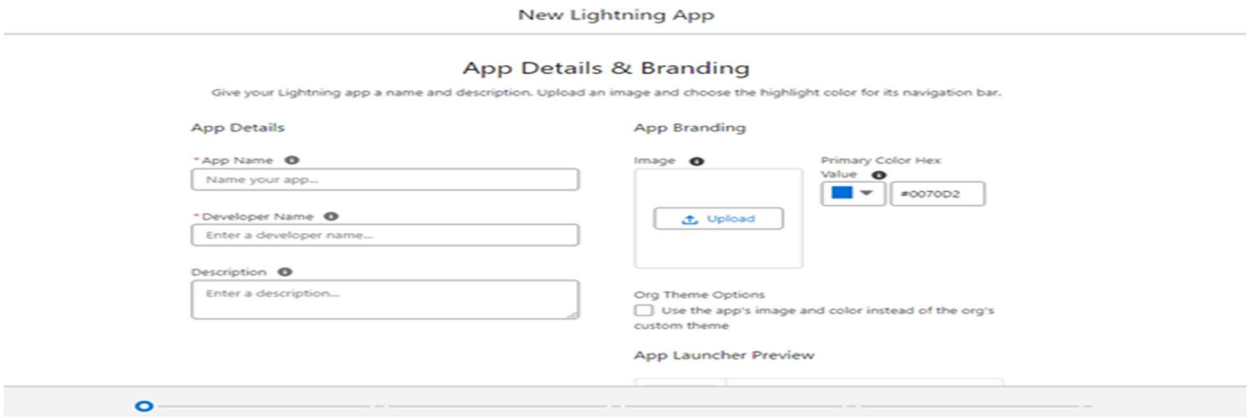
- Go to setup page >> type Tabs in Quick Find bar >> click on tabs >> New (under custom object tab)
- Select Object(Customer Details) >> Select the tab style >> Next (Add to profiles page) keep it as default >> Next (Add to Custom App) uncheck the include tab .
- Make sure that the Append tab to users'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



The Lightning App

Create a Lightning App

- Go to setup page >> search “app manager” in quick find >> select “app manager” >> click on New lightning App.
- Fill the app name in app details asGarage 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 Profiles:
- Search profiles (System administrator) in the search bar >> click on the arrow button >> save & finish.

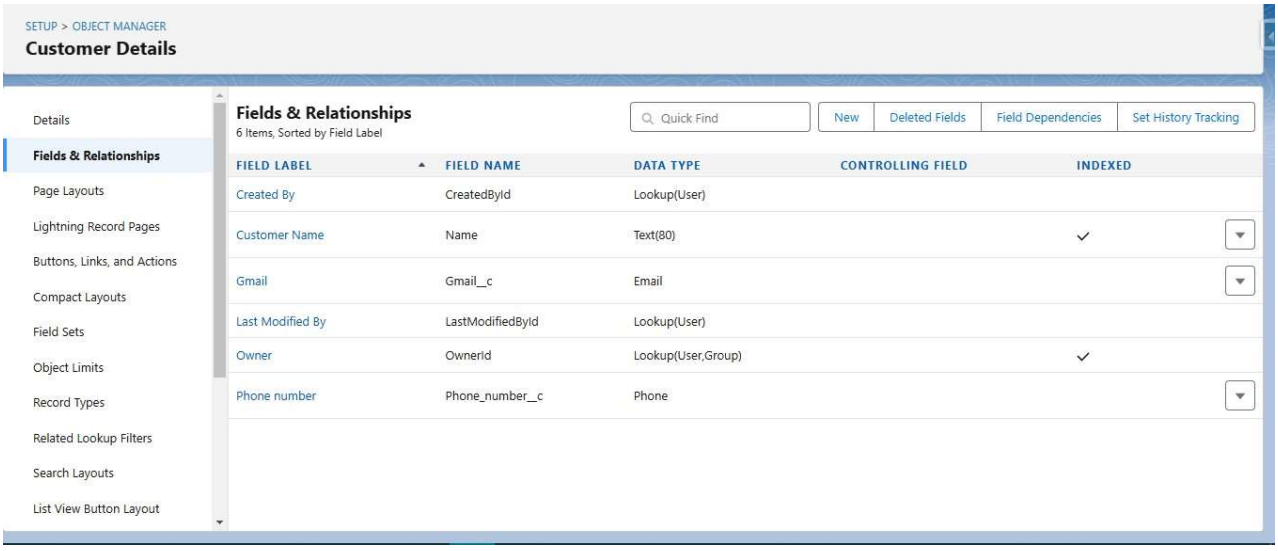


Fields

Creation of fields for the Customer Details 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 field for the same object.
- To create another fields 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

- 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.
- 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 onService recordsObject :

- 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 field so click on Required.
- Scroll down for Lookup Filter and click on Show filter settings.
- Now add the filter criteria.
- Field : Appointment: Appointment Date >> Operator : less than >> select field >> Appointment: Created Date
- Filter type should be Required.
- Error Message : Value does not match the criteria.
- Enable the filter by click on Active.
- Next >> Next >> Save.

Creation of Lookup Field onBilling details and feedbackObject :

- 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:

- 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 Another Checkbox 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 Checkbox Field onService recordsObject :

- 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 : Quality Check Status

- Field Nme : is auto populated
- Default value : unchecked
- Click on next >> next >> save.

Creation of date 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 "Date" as data type and click Next.
- Give the Field Label : Appointment Date
- Field Name : is auto populated
- Make it as a Required field by click on the Required option.
- Click on next >> next >> save.

Creation of Currency 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 "Currency" as data type and click Next.
- Give the Field Label : Service Amount
- Field Name : 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 Currency Field on Billing details and feedback Object :

- Follow the same steps as mentioned above in Billing details and feedback Object.

- Change the label name as mentioned.
- Give the Field Label : Payment Paid
- Field Name : is auto populated

Creation of Text Fields:

Creation of Text Fields in 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 “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.
- Click on next >> next >> save

Creation of Picklist Fields:

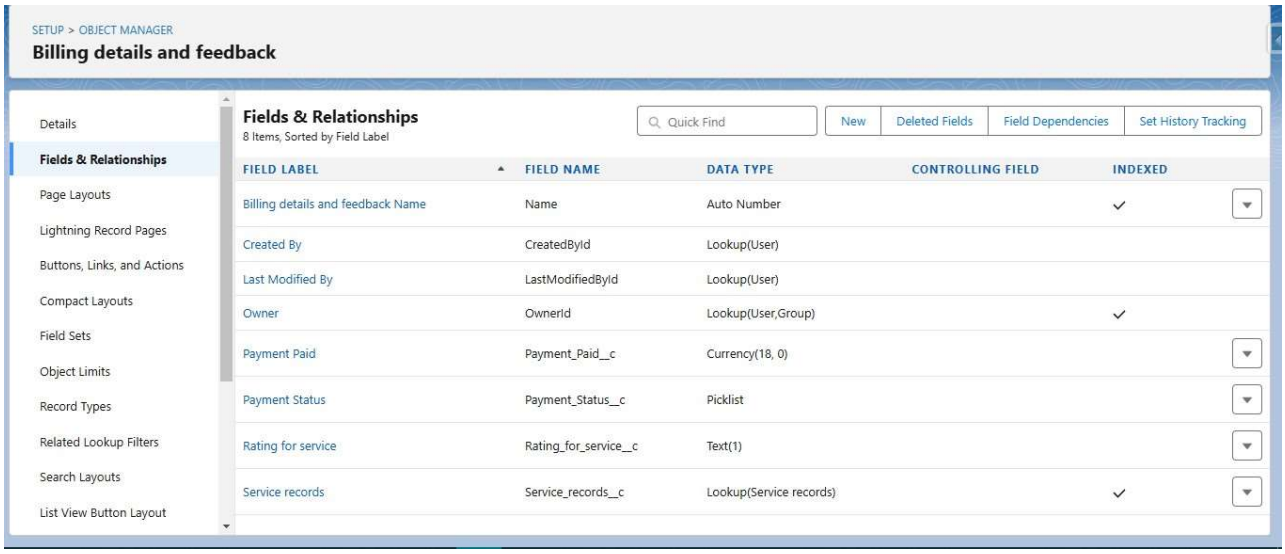
- Go to setup >> click on Object Manager >> type object name(Service records)in search bar >> click on the object.
- Click on fields & relationship >> click on New.
- Select Data type as “Picklist” and click Next.
- Enter Field Label as “Service Status”, under values select “Enter values, with each value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

Creation of Picklist Fields in 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.
- 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 Field in Service records Object:

- Go to setup >> click on Object Manager >> 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 “service date” and select formula return type as “Date” and click next.
- Insert field formula should be : CreatedDate
- click “Check Syntax” .
- Click next >> next >> Save.



SETUP > OBJECT MANAGER

Billing details and feedback

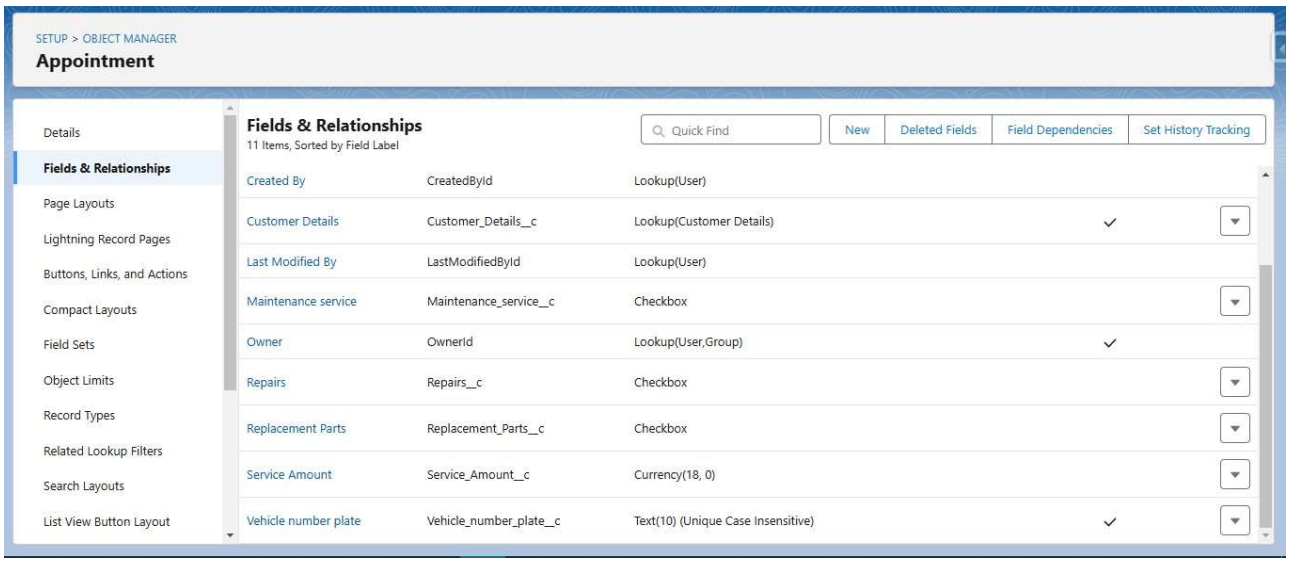
Details
Fields & Relationships
Page Layouts
Lightning Record Pages
Buttons, Links, and Actions
Compact Layouts
Field Sets
Object Limits
Record Types
Related Lookup Filters
Search Layouts
List View Button Layout

Fields & Relationships

8 Items, Sorted by Field Label

Quick Find | New | Deleted Fields | Field Dependencies | Set History Tracking

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Billing details and feedback Name	Name	Auto Number		✓
Created By	CreatedById	Lookup(User)		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		✓
Payment Paid	Payment_Paid__c	Currency(18, 0)		
Payment Status	Payment_Status__c	Picklist		
Rating for service	Rating_for_service__c	Text(1)		
Service records	Service_records__c	Lookup(Service records)		✓



SETUP > OBJECT MANAGER

Appointment

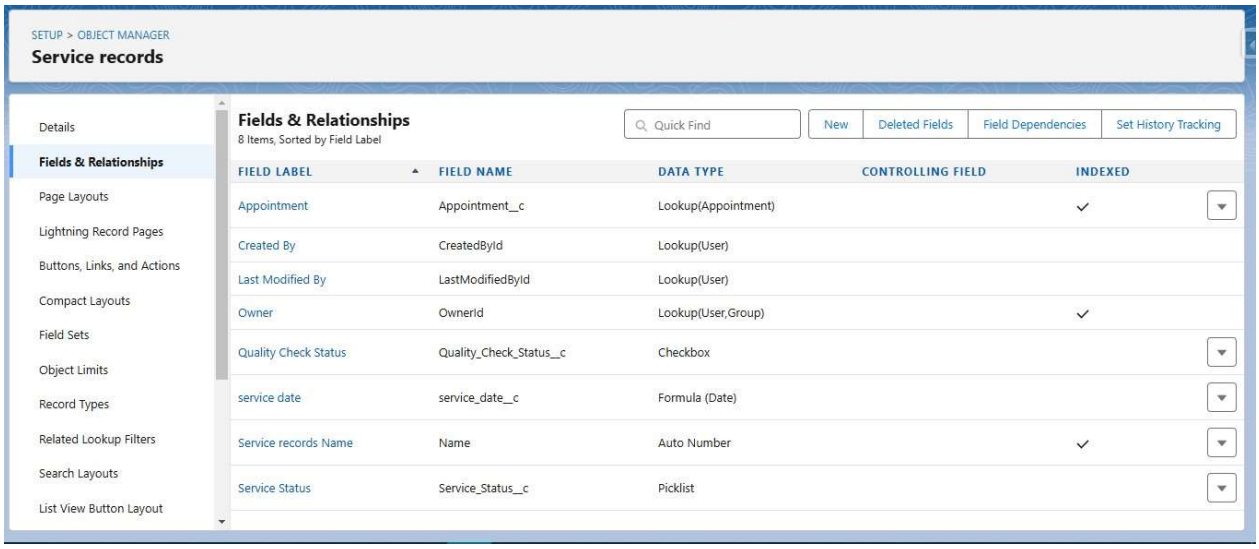
Details
Fields & Relationships
Page Layouts
Lightning Record Pages
Buttons, Links, and Actions
Compact Layouts
Field Sets
Object Limits
Record Types
Related Lookup Filters
Search Layouts
List View Button Layout

Fields & Relationships

11 Items, Sorted by Field Label

Quick Find | New | Deleted Fields | Field Dependencies | Set History Tracking

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)	✓
Repairs	Repairs__c	Checkbox	
Replacement Parts	Replacement_Parts__c	Checkbox	
Service Amount	Service_Amount__c	Currency(18, 0)	
Vehicle number plate	Vehicle_number_plate__c	Text(10) (Unique Case Insensitive)	✓



SETUP > OBJECT MANAGER

Service records

Details
Fields & Relationships
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

Fields & Relationships

8 Items, Sorted by Field Label

Quick Find | New | Deleted Fields | Field Dependencies | Set History Tracking

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)		✓
Quality Check Status	Quality_Check_Status__c	Checkbox		
service date	service_date__c	Formula (Date)		
Service records Name	Name	Auto Number		✓
Service Status	Service_Status__c	Picklist		

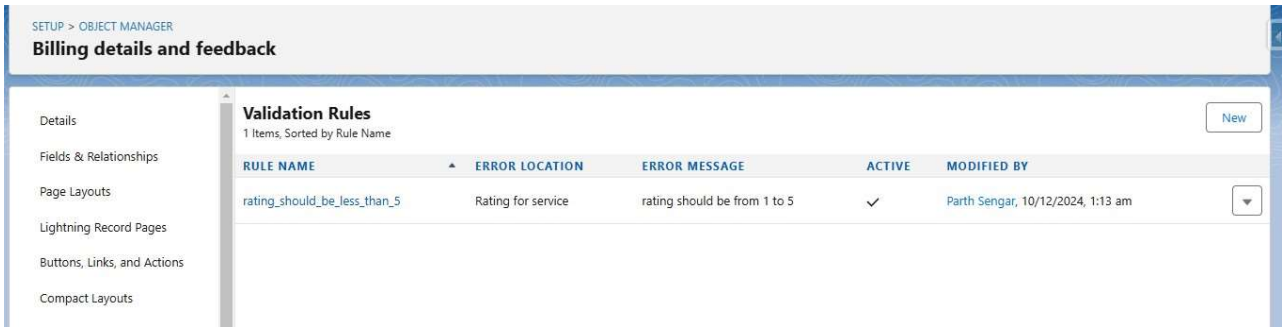
Validation rule

To create a validation rule to an Appointment Object:

- Go to the setup page >> click on object manager >> From drop down click edit for Appointmentobject.
- Click on the validation rule >> click New.
- Enter the Rule name as " Vehicle ".
- Insert the Error Condition Formula as : -
- NOT(REGEX(Vehicle_number_plate__c , "[A-Z]{2}[0-9]{2}[A-Z]{2}[0-9]{4}"))
- Enter the Error Message as "Please enter vaild number ", select the Error location as Field and select the field as "Vehicle number plate", and click Save.

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

- . Go to the setup page >> click on object manager >> From drop down click edit for Billing details and feedbackobject.
- Click on the validation rule >> click New.
- Enter the Rule name as " rating_should_be_less_than_5".
- Insert the Error Condition Formula as : -
- NOT(REGEX(Rating_for_service__c , "[1-5]{1}"))
- Enter the Error Message as "rating should be from 1 to 5", select the Error location as Field and select the field as "Rating for Service", and click Save



Profiles

Manager Profile

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

sales person Profile

- Go to setup >> type profiles in quick find box >> click on profiles >> clone the desired profile (Salesforce Platform User) >> enter profile name (sales person) >> Save.
- While still on the profile page, then click Edit.
- Select the Custom App settings as default for the GArage management.
- 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.

- And click save.

Action	Profile Name ↑	User License	Custom
Del | Clone	Manager	Salesforce	✓

Action	Profile Name ↑	User License	Custom
Del | Clone	sales person	Salesforce Platform	✓

Role & Role Hierarchy

Creating Manager Role

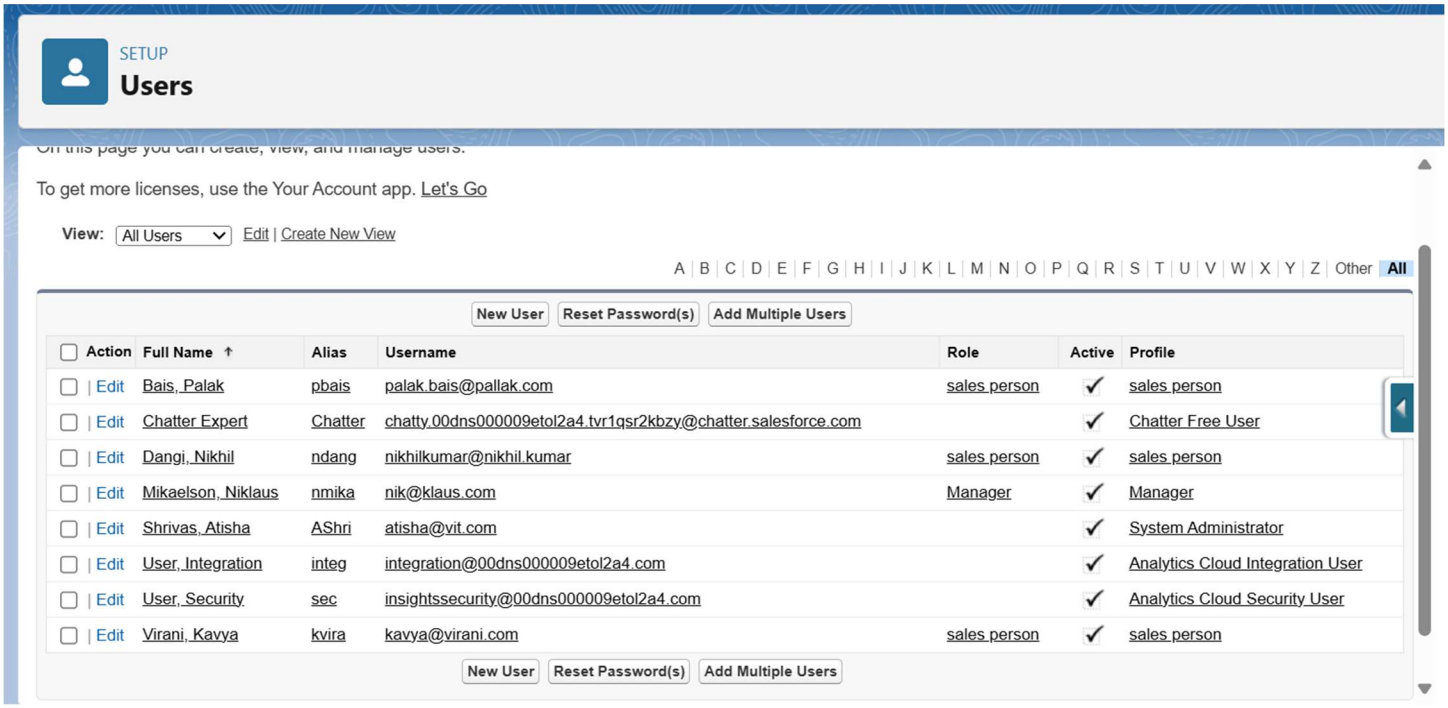
- Go to quick find >> Search for Roles >> click on set up roles.
- Click on Expand All and click on add role under whom this role works.
- Give Label as “Manager” and Role name gets auto populated. Then click on Save.

Users

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 Email id
- Username: Username should be in this form: text@text.text
- Nick Name: Give a Nickname
- Role: Manager
- User licence: Salesforce
- Profiles: Manager
- Save.



Sharing Setting

Creating Sharing settings

- Go to setup >> type users in quick find box >> select Sharing Settings >> click Edit.
- Change the OWD setting of the Service records Object to private as shown in fig.

- Click on save and refresh.
- Scroll down a bit, Click new on Service records 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: share with, select " Roles " >> " Manager "
- In step 5 : Change the access level to " Read / write ".
- Click on save.

Flows

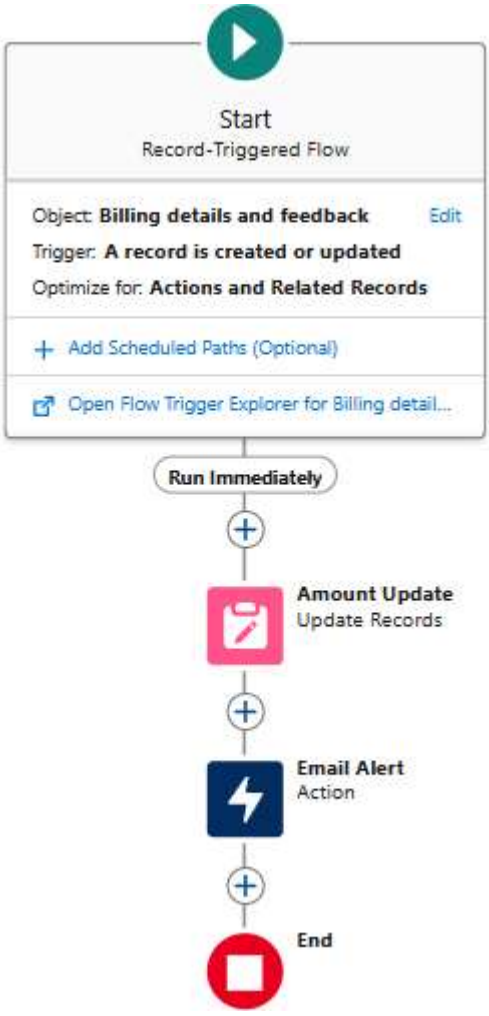
Create a Flow:

- 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 "+" Symbol and 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
- AndSet Field Values for the Billing details and feedback Record
- Field : Payment_Paid__c
- Value : {!$Record.Service_records__r.Appointment__r.Service_Amount__c}
- Click On Done.
- Before creating another Element. Create a New Resource form Toolbox form top left.
- Click on the New Resource, And select Variable.
- Select the resource type as text template.
- Enter the API name as " alert".
- Change the view as Rich Text ? View to Plain Text.
- In body field paste the syntax that given below.
- Dear {!$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 .
- Click done.
- Now Click on Add Element , select Action.
- Their action bar will be opened in that search for " 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.



Report

Create Report:

- Go to the app >> click on the reports tab
- Click New Report.
- Select the Category as other reports, search for Service Information, select that report, click on it. And click on start report.
- Their outline pane is opened alredy, select the fields that mentioned below 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 report Name : New Service information Report
- Report unique Name is auto populated.
- Select the folder the created and Click on save.

5. Testing and Validation

Creating an Apex Class (amount distribution):

- 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 console window.

- In the toolbar, you can see FILE. Click on 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 console and you will be navigated to 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 records sum exceed the threshold i.e minimum business requirement value. Then the code in the trigger will get executed.

- Handler for the Appointment Object

Code:

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

6. Key Scenarios Addressed by Salesforce in the Implementation Project

- **Sales Process Automation**: Salesforce can automate sales workflows, reducing manual tasks.
- **Customer Support**: Salesforce can provide tools to manage customer service cases and track resolutions.
- **Data Analytics and Reporting**: Salesforce can generate detailed reports for business insights.
- **Resource Management: :** Salesforce can help allocate resources efficiently based on business needs.

7. Conclusion

In this project, Salesforce streamlined operational processes by enabling automated data calculations, real-time reporting, and secure access control. Custom widgets provided visual insights into rice sales, production, and revenue, enhancing decision-making. Validation rules ensured data accuracy, while role-based access protected sensitive information. Rollup summaries and formulas reduced manual effort in calculations. Overall, Salesforce optimized business operations, contributing to improved productivity and planning.