

# Garage Management System



**Project By,**

Kuchipudi Akhil

[21bq1a05B7@vvit.net](mailto:21bq1a05B7@vvit.net)

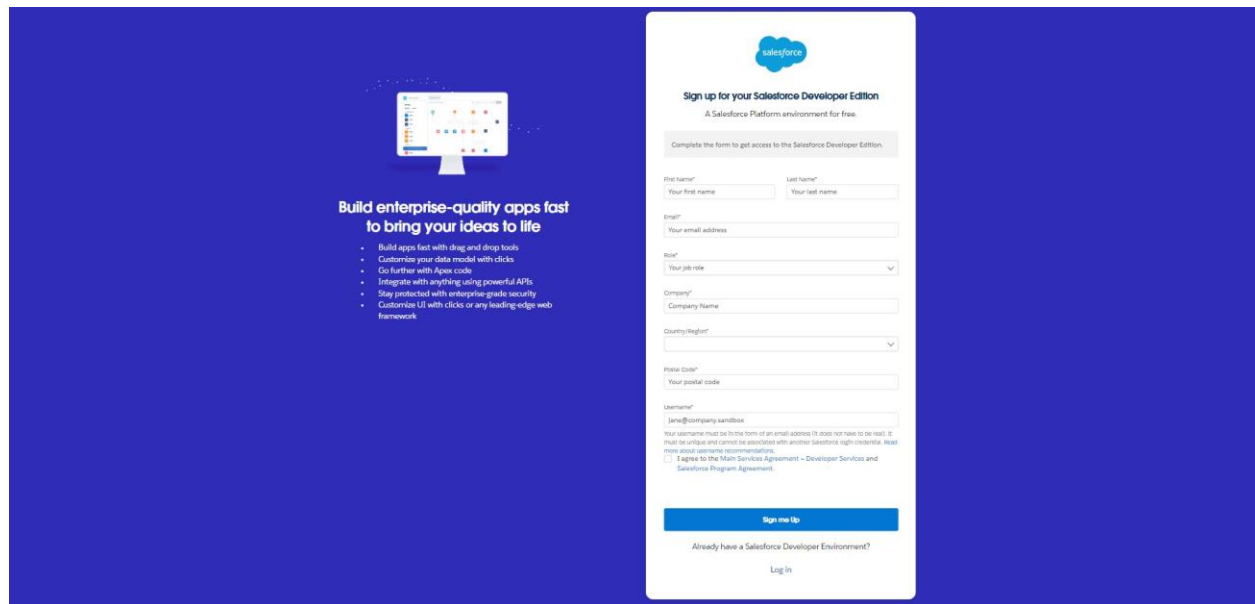
## About Garage Management System:

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.

## Garage Management System Setup Guide

### 1. Creating a Salesforce Developer Account:

1. Visit [Salesforce Developer Signup](#).



2. Fill out the signup form:

- First Name & Last Name
- Email
- Role: Developer
- Company: College Name
- Country: India
- Postal Code: Your Pin Code
- Username: (e.g., [username@organization.com](#))

## 2. Account Activation

1. Check your email inbox for a verification email from Salesforce.
2. Click Verify Account.
3. Set up your password and security question, then click Change Password.

## 3. Creating Salesforce Objects

### 3.1 Customer Details Object

1. Go to Object Manager > Create > Custom Object.

2. Configure the following:
  - Label Name: Customer Details
  - Plural Label Name: Customer Details
  - Record Name: Customer Name (Text)
3. Enable Allow Reports, Track Field History, and Allow Search.
4. Click Save.

## **3.2 Appointment Object**

1. Go to Object Manager > Create > Custom Object.
2. Configure the following:
  - Label Name: Appointment
  - Plural Label Name: Appointments
  - Record Name: Appointment Name (Auto Number)
  - Display Format: app-{000}, Starting Number: 1
3. Enable Allow Reports, Track Field History, and Allow Search.
4. Click Save.

## **3.3 Service Records Object**

1. Go to Object Manager > Create > Custom Object.
2. Configure the following:
  - Label Name: Service Records
  - Plural Label Name: Service Records
  - Record Name: Service Records Name (Auto Number)
  - Display Format: ser-{000}, Starting Number: 1
3. Enable Allow Reports, Track Field History, and Allow Search.
4. Click Save.

## **3.4 Billing Details and Feedback Object**

1. Go to Object Manager > Create > Custom Object.
2. Configure the following:
  - Label Name: Billing Details and Feedback
  - Plural Label Name: Billing Details and Feedback

- . Record Name: Billing Details and Feedback Name (Auto Number)
  - . Display Format: bill-{000}, Starting Number: 1
- 3. Enable Allow Reports, Track Field History, and Allow Search.
- 4. Click Save.

## 4. Creating Custom Tabs

1. Go to Setup > Tabs > New (under Custom Object Tabs).
2. Select the object (e.g., Customer Details), choose a tab style, and click Next.
3. Configure profile visibility as needed and click Next.
4. Click Save.

Repeat for remaining objects (Appointments, Service Records, Billing Details and Feedback).

## 5. Creating a Lightning App

1. Go to Setup > App Manager > New Lightning App.
2. Configure the app:
  - . App Name: Garage Management Application
  - . Leave other settings as default and click Next.
3. Add navigation items (Customer Details, Appointments, Service Records, Billing Details and Feedback, Reports, Dashboards) and click Next.
4. Assign the System Administrator profile to the app and click Save & Finish.

## 6. Creating Fields for Objects

### 6.1 Customer Details Object

1. Go to Setup > Object Manager > Customer Details > Fields & Relationships > New.
2. Create the following fields:
  - . Phone Number (Phone)
  - . Gmail (Email)

### 6.2 Appointment Object

1. Create a Lookup Relationship to Customer Details.
2. Create the following fields:
  - . Maintenance Service (Checkbox)
  - . Repairs (Checkbox)
  - . Replacement Parts (Checkbox)
  - . Appointment Date (Date)
  - . Service Amount (Currency)
  - . Vehicle Number Plate (Text)

### 6.3 Service Records Object

1. Create a Lookup Relationship to Appointment.
2. Create the following fields:
  - . Quality Check Status (Checkbox)
  - . Service Status (Picklist: Started, Completed)
  - . Service Date (Formula: Created Date)

### 6.4 Billing Details and Feedback Object

1. Create a Lookup Relationship to Service Records.
2. Create the following fields:
  - . Payment Paid (Currency)
  - . Rating for Service (Text)
  - . Payment Status (Picklist: Pending, Completed)

## 7. Creating Validation Rules

### 7.1 Appointment Object

- Rule Name: Vehicle
- Formula: `NOT(REGEX(Vehicle_number_plate__c, "[A-Z]{2}[0-9]{2}[A-Z]{2}[0-9]{4}"))`
- Error Message: Please enter a valid number.

### 7.2 Service Records Object

- Rule Name: service\_status\_note
- Formula: `NOT(ISPICKVAL(Service_Status__c, "Completed"))`
- Error Message: Still it is pending.

### 7.3 Billing Details and Feedback Object

- Rule Name: rating\_should\_be\_less\_than\_5
- Formula: `NOT(REGEX(Rating_for_service__c, "[1-5]{1}"))`
- Error Message: Rating should be from 1 to 5.

## 8. Creating a Flow

### 8.1 Create Flow for Billing Details and Feedback

1. Go to Setup > Flow > New Flow.
2. Select Record-triggered flow.
3. Configure the flow to trigger when a record is created or updated.
4. Add an Update Records element:
  - Label: Amount Update
  - Filter Condition: Payment Status = Completed
  - Field Values: Set Payment Paid to `Service_Amount__c`.
5. Add an Email Alert element:
  - Label: Email Alert
  - Recipient:
 

```
{!$Record.Service_records__r.Appointment__r.Customer_Name__r.Gmail__c}
```
  - Subject: Thank You for Your Payment - Garage Management
  - Body: Use a text template to include customer details and payment amount.
6. Save and activate the flow.

## 9. Apex Code for Amount Distribution

### 9.1 Create an Apex Handler Class

Code:

```

public class AmountDistributionHandler {
    public static void amountDist(list<Appointment_c> listApp) {
        for (Appointment_c app : listApp) {
            if (app.Maintenance_service_c && app.Repairs_c && app.Replacement_Parts_c) {
                app.Service_Amount_c = 10000;
            } else if (app.Maintenance_service_c && app.Repairs_c) {
                app.Service_Amount_c = 5000;
            } else if (app.Maintenance_service_c && app.Replacement_Parts_c) {
                app.Service_Amount_c = 8000;
            } else if (app.Repairs_c && app.Replacement_Parts_c) {
                app.Service_Amount_c = 7000;
            } else if (app.Maintenance_service_c) {
                app.Service_Amount_c = 2000;
            } else if (app.Repairs_c) {
                app.Service_Amount_c = 3000;
            } else if (app.Replacement_Parts_c) {
                app.Service_Amount_c = 5000;
            }
        }
    }
}

```

## 9.2 Create a Trigger for Appointment

Code:

```

trigger AmountDistribution on Appointment_c (before insert, before update) {
    if (trigger.isBefore && (trigger.isInsert || trigger.isUpdate)) {
        AmountDistributionHandler.amountDist(trigger.new);
    }
}

```

# 10. Creating Reports

## 10.1 Create a Report Folder

1. Go to App Launcher > Reports > New Folder.
2. Name the folder Garage Management Folder and click Save.

## **10.2 Create a Report Type**

1. Go to Setup > Report Types > New Custom Report Type.
2. Select Customer Details as the primary object.
3. Add Appointment, Service Records, and Billing Details and Feedback as related objects.

## **10.3 Create Reports**

1. Go to Reports > New Report.
2. Use the custom report type to create and customize your reports.

# **10. Creating Dashboards**

## **10.1 Create a 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. Share the Dashboard Folder with the Role "Manager".

## **10.2 Create Dashboard**

1. Go to the app >> click on the Dashboards tabs.
2. Give a Name and select the folder that created, and click on create.
3. Select add component.
4. Select a Report and click on select.
5. Select the Line Chart. Change the theme.
6. Click Add then click on Save and then click on Done.
7. Preview is shown below.
8. After that Click on Subscribe on top right.



9. Set the Frequency as “ weekly ”.
10. Set a day as monday.
11. And Click on save.
12. The Dashboard is created.

## Garage Management System Overview

### Customer Details

The Customer Details tab shows the list of Customers and their details.

Garage Manage... Customer Details Appointments Service records Billing details and feedback Reports Dashboards

Customer Details Recently Viewed 10 Items • Updated 3 minutes ago

New Import Change Owner Assign Label

Search this list...

<input type="checkbox"/>	Customer Name	
<input type="checkbox"/>	Jennifer	
<input type="checkbox"/>	Jaswanth	
<input type="checkbox"/>	Sasidhar	
<input type="checkbox"/>	Karthik	
<input type="checkbox"/>	Katharine Lang	
<input type="checkbox"/>	Olivia Morris	
<input type="checkbox"/>	Sumanth	
<input type="checkbox"/>	Mahesh	
<input type="checkbox"/>	Balu	
<input type="checkbox"/>	Gal Gadot	

#### Related Details

Customer Name	Owner
Jennifer	Arjun Gogulamudi
Phone number	
9988991001	
Gmail	
jennifer10@hotmail.com	
Customer Name	
app-010	
Created By	Last Modified By
Arjun Gogulamudi, 25/08/2024, 9:20 pm	Arjun Gogulamudi, 25/08/2024, 9:20 pm

### Appointments

Shows the list of appointments and the appointment details.

The screenshot displays the 'Appointments' section of a Garage Management System. The top navigation bar includes 'Garage Management...', 'Customer Details', 'Appointments' (selected), 'Service records', 'Billing details and feedback', 'Reports', and 'Dashboards'. A search bar is located at the top right. The 'Appointments' section has a sub-header 'Recently Viewed' with a dropdown arrow and a plus icon. Below this, it shows '11 items • Updated a few seconds ago'. A search bar for the list is also present. The list of appointments is as follows:

	<input type="checkbox"/> Appointment Name	
1	<input type="checkbox"/> app-001	▼
2	<input type="checkbox"/> app-010	▼
3	<input type="checkbox"/> app-009	▼
4	<input type="checkbox"/> app-008	▼
5	<input type="checkbox"/> app-007	▼
6	<input type="checkbox"/> app-006	▼
7	<input type="checkbox"/> app-005	▼
8	<input type="checkbox"/> app-004	▼
9	<input type="checkbox"/> app-003	▼
10	<input type="checkbox"/> app-002	▼
11	<input type="checkbox"/> app-012	▼

## Service Records

Shows the details of every Service record of the customer.

	Service records Name	
1	ser-012	
2	ser-008	
3	ser-011	
4	ser-010	
5	ser-009	
6	ser-007	
7	ser-006	
8	ser-005	
9	ser-004	
10	ser-003	

## Billing Details and Feedback

Contains the overall summary of service, amount paid by the customer, payment status and Rating for the service.

AVAILABLE: 0

- Profile
- Skill Bank
- Skill Wallet**
- Skill Card
- Job Board
- Subscription
- Transactions

- Fields
- Validation Rule
- Duplicate Rule
- Profiles
- Role & Role Hierarchy
- Users
- Public Groups
- Sharing Setting
- Flows
- Apex Trigger
- Apex Handler**
- Reports

UseCase : This use case works for Amount Distribution for each Service the customer selected for there Vehicle.

1. Login to the respective trailhead account and navigate to the gear icon in the top right corner.
2. Click on the Developer console. Now you will see a new console window.
3. In the toolbar, you can see FILE. Click on it and navigate to new and create New apex class.
4. Name the class as "AmountDistributionHandler ".

```

1 public class AmountDistributionHandler {
2
3     public static void amountDist(List<Appointment__c> listApp){
4         listService_records__c service = 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 = 18000;
9             }
10            else if(app.Maintenance_service__c == true && app.Repairs__c == true){
11                app.Service_Amount__c = 9000;
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 = 10000;
21            }
22        }
23    }
24 }

```

The screenshot displays the Salesforce interface for 'Billing details and feedback'. The top navigation bar includes 'Garage Manage...', 'Customer Details', 'Appointments', 'Service records', 'Billing details and feedback' (selected), 'Reports', and 'Dashboards'. A search bar is present in the top right.

The main section shows a list of 10 items under the heading 'Billing details and feedback Name'. The items are listed in a table with columns for a checkbox, a number, a name, and a dropdown menu.

	checkbox	Billing details and feedback Name	
1	<input type="checkbox"/>	bill-012	▼
2	<input type="checkbox"/>	bill-011	▼
3	<input type="checkbox"/>	bill-010	▼
4	<input type="checkbox"/>	bill-009	▼
5	<input type="checkbox"/>	bill-008	▼
6	<input type="checkbox"/>	bill-007	▼
7	<input type="checkbox"/>	bill-006	▼
8	<input type="checkbox"/>	bill-005	▼
9	<input type="checkbox"/>	bill-003	▼
10	<input type="checkbox"/>	bill-001	▼

Below the list, the detailed view for 'bill-010' is shown. It includes a 'Details' tab and a 'Related' section. The 'Details' section contains the following information:

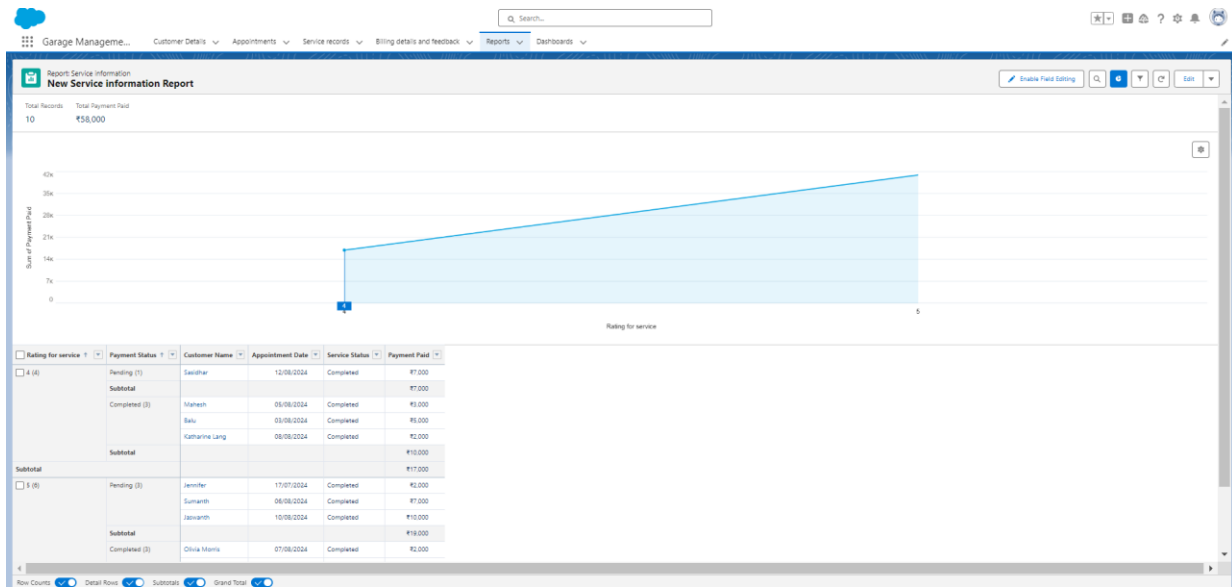
- Billing details and feedback Name:** bill-010
- Owner:** Arjun Gogulamudi
- Service records:** ser-010
- Payment Paid:** ₹7,000
- Rating for service:** 4
- Payment Status:** Pending
- Created By:** Arjun Gogulamudi, 25/08/2024, 10:08 pm
- Last Modified By:** Arjun Gogulamudi, 25/08/2024, 10:08 pm

## 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



## Dashboard

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.

