

Garage Management System

GARAGE MANAGEMENT SYSTEM



SUBMITTED BY: SHAIK MOHAMMED GAFFAR ALI

EMAIL ID : shaikgaffar1519@gmail.com

COURSE: B.TECH (AI&ML)

ROLL NO: 219Y1A3951

COLLEGE: K.S.R.M COLLEGE OF ENGINEERING

Profile - Student | Student - Skill Wallet | New Service informa... | SHAIK MOHAMMED | Welcome to Salesfor... | Customer Review1 | New Tab

ksrmcollegeofengineering-df-dev-ed.develop.lightning.force.com/lightning/r/User/005WU0000090qe9YAA/view

YouTube | Study - Physics Wall... | All Bookmarks

Garage Management... Customer Details Appointments Service records Billing details and feedback Reports Dashboards * SHAIK MOHAMMED GAFFAR ALI

Search...

SHAIK MOHAMMED GAFFAR ALI

Share your awesomeness with the world.
(Or at least with your colleagues on Chatter.)

TRAILHEAD Learn new skills on Trailhead.

Connect with fellow Trailblazer

Details

Name: SHAIK MOHAMMED GAFFAR ALI Manager:

Title:

Company Name: K.S.R.M COLLEGE OF ENGINEERING

Email: shaikgaffar1519@gmail.com Phone:

Address: IN Mobile:

About Me
Share your awesomeness with the world. (Or at least with your colleagues on Chatter.)

Groups (0)

Files (1)
CAR-GARAGE-REPAIR-MANAGEMENT-SOLUTIONS
19-Dec-2024 • 60KB • jpg

Followers (0)

24°C Mostly cloudy

Search

08:16 24-12-2024

Project Overview

This project is focused on developing a robust **Garage Management System (GMS)** to address the challenges associated with managing garage operations and vehicle servicing efficiently. The goal is to deliver a comprehensive solution by leveraging **Salesforce's cloud-based platform**, renowned for its customer relationship management (CRM) capabilities and powerful customization features.

Through this project, we aim to enhance **operational efficiency, customer satisfaction, and data accuracy** while supporting the long-term objectives of seamless service delivery and operational scalability for garage businesses.

The Garage Management System serves as a vital tool for automotive repair facilities, enabling them to streamline processes such as **appointment scheduling, service tracking, inventory management, and customer communication**. With its intuitive interface and tailored functionalities, GMS empowers garage owners and staff to deliver superior service while ensuring a seamless and satisfying experience for customers.

This project utilizes Salesforce's platform to integrate standard CRM features with customized functionalities specific to garage management. The system enables garage owners to:

- **Centralize customer data**, including contact details, vehicle information, service preferences, and interaction history.
- Provide **personalized services** by leveraging historical customer and vehicle data.
- Facilitate the creation and management of **work orders** based on customer requests and staff availability.
- **Monitor service progress** and inventory in real time to ensure timely service delivery.
- Generate actionable insights with **real-time analytics and reporting tools**.

The Garage Management System addresses common challenges faced by traditional management methods, such as **fragmented data handling, manual record-keeping, and inefficient workflows**. By automating repetitive tasks, centralizing data, and providing advanced reporting, the solution reduces errors, improves productivity, and enhances decision-making capabilities.

This project is adaptable and scalable, catering to the needs of both small workshops and large multi-branch service centres, making it a versatile solution in today's competitive automotive service market.

Objectives

Business Goals

Enhance Operational Efficiency:

The Garage Management System aims to automate key tasks such as appointment scheduling, service tracking, and inventory management. By leveraging Salesforce's platform, the system will streamline workflows, reduce service turnaround time, and minimize errors caused by manual processes.

Improve Customer Experience:

One of the primary goals is to enhance customer satisfaction by providing a seamless experience through transparent service updates, personalized communication, and easy access to service history. With Salesforce CRM, customers can enjoy timely notifications and a user-friendly interface for appointment booking and updates.

Optimize Resource Utilization:

The system aims to optimize resource allocation, including technician scheduling, spare parts management, and tool availability. By offering real-time insights into resource usage, the solution will help minimize downtime and improve operational productivity.

Increase Revenue Opportunities:

By automating billing, identifying upselling opportunities through customer data, and improving service accuracy, the system is designed to boost revenue. It integrates advanced tracking of services and parts, allowing garages to offer targeted recommendations and promotions to customers.

Data-Driven Decision Making:

Through Salesforce's analytics and reporting capabilities, garage managers will gain actionable insights into customer behavior, service performance, and inventory trends. This will enable strategic planning and more informed decision-making to improve overall business efficiency.

Specific Outcomes

Streamlined Service Scheduling:

- Implementation of a smart scheduling tool to manage and organize appointments effectively.
- Reduction of scheduling conflicts and last-minute cancellations.

Enhanced Customer Relationship Management (CRM):

- Centralized database for managing customer interactions, service preferences, and feedback.
- Improved personalized communication based on historical data.

Inventory and Spare Parts Management:

- Automated tracking of inventory levels with timely restocking alerts.
- Seamless integration with suppliers to manage orders and stock updates.

Real-Time Reporting and Insights:

- Creation of interactive dashboards to monitor garage operations and technician performance.
- Generation of detailed reports to support business analysis and forecasting.

Workflow Automation:

- Automation of repetitive tasks such as service reminders, invoicing, and follow-ups.
- Smart technician assignments based on skills and availability.

Secure Payment Integration:

- Integration with payment gateways to offer secure and flexible payment options.
- Support for multiple payment methods, including digital wallets and credit card payments.

User-Friendly Design:

- Development of an intuitive and mobile-friendly interface for both staff and customers.
- Enhanced accessibility for real-time updates and seamless navigation.

Cost Efficiency and Profit Maximization:

- Reduction in operational costs by automating manual tasks.
- Improved profitability through better inventory control and accurate invoicing.

Salesforce Key Features and Concepts Utilized

The Garage Management System extensively utilizes Salesforce's robust platform capabilities to ensure efficient, scalable, and user-friendly functionality. Below are the key Salesforce features and concepts employed.

Objects

What are Salesforce Objects?

Salesforce objects are the core database structures used to store data within the platform. They were critical in modeling the Garage Management System's operational framework.

Types of Objects Utilized:

1. Standard Objects:

- **Accounts:** Used to represent the garage itself or corporate clients.
- **Contacts:** Used to store customer details, including vehicle owners.
- **Cases:** Mapped to service requests or issues raised by customers.
- **Opportunities:** Represented potential upselling opportunities like additional services or parts.

2. Custom Objects:

- **Vehicles:** To store vehicle details such as make, model, registration number, and service history.
- **Service Orders:** Used to track individual garage services, including labor, spare parts, and costs.
- **Technician Assignments:** Managed technician workloads and their assigned tasks.

Tabs

Tabs acted as navigational elements to access objects and functionalities quickly.

Types of Tabs:

1. Custom Tabs:

- Enabled users to view and interact with custom objects like Vehicles and Service Orders.

2. Web Tabs:

- Integrated third-party applications like supplier inventory systems or customer feedback portals.

3. Lightning Component Tabs:

- Added custom Lightning Components, such as a "Technician Dashboard," for real-time monitoring of ongoing tasks.

The Lightning App

Custom Lightning App:

A bespoke Lightning App was developed to provide a unified interface for garage operations:

- **User-Centric Design:** Incorporated a visually appealing design with custom branding, including garage-specific logos and color themes.
- **Navigation Simplicity:** Streamlined navigation with objects like Vehicles, Service Orders, and Inventory accessible via a customizable navigation bar.
- **Utility Bar Integration:** Added tools like a service cost calculator and appointment reminders to enhance user productivity.

Fields

Fields were pivotal for capturing detailed and accurate data across objects.

Standard Fields:

- Predefined fields such as **Created By**, **Last Modified By**, and **Record Owner** were used to maintain system logs and accountability.

Custom Fields:

- Developed fields tailored to garage operations, including:
 - **Vehicle Type:** Captured the vehicle's classification, such as sedan, SUV, or motorcycle.
 - **Service Status:** Tracked the progression of service orders, e.g., Pending, In Progress, or Completed.
 - **Estimated Service Duration:** Enabled accurate scheduling and technician planning.

Validation Rules

Validation rules were implemented to ensure the accuracy and consistency of data entry. Examples include:

- Preventing the creation of duplicate customer or vehicle records.
- Enforcing mandatory fields like "Service Type" and "Vehicle Registration Number."
- Verifying that the service appointment date is not set in the past.

Profiles and Role Hierarchy

Profiles:

- Configured custom profiles for garage roles such as:

- **Service Technicians:** Limited access to service orders assigned to them.
- **Service Managers:** Full access to all records, including dashboards and reports.
- **Front Desk Operators:** Access to appointment scheduling and customer information.

Role Hierarchy:

- Designed a hierarchy to align data visibility with organizational structure. For instance:
 - Managers had access to all subordinate data, while technicians could only view their assignments.

Sharing Settings

Organization-Wide Defaults (OWD):

- Configured private defaults for sensitive objects like Vehicles and Service Orders.

Sharing Rules:

- Automatically extended access to service orders for teams working on joint tasks.

Manual Sharing:

- Enabled ad-hoc sharing of specific customer records with technicians.

Flows

Flows were designed to automate and optimize business processes:

- **Service Reminder Flow:** Automated email or SMS notifications to customers reminding them of upcoming appointments.
- **Post-Service Follow-Up Flow:** Triggered customer feedback requests upon service completion.
- **Inventory Update Flow:** Adjusted stock levels automatically when spare parts were consumed during service.

Apex Triggers

Apex triggers enabled custom functionality:

- **Before Triggers:**
 - Validated service data, such as ensuring accurate cost estimates before saving.

- **After Triggers:**
 - Automatically updated related records, like reducing spare part inventory upon service order completion.

Reports and Dashboards

Reports:

- **Operational Efficiency Reports:** Measured service completion times and technician productivity.
- **Inventory Reports:** Monitored spare part usage and stock levels.
- **Customer Satisfaction Reports:** Analyzed feedback trends to improve services.

Dashboards:

- Created real-time dashboards displaying key metrics like:
 - Technician performance.
 - Revenue from completed services.
 - Current service workload.

Data Security and Compliance

- Implemented Salesforce Shield to encrypt sensitive data, such as payment details and customer PII.
- Ensured GDPR compliance by limiting data retention for inactive customers and anonymizing sensitive data where required.

Integration with External Systems

- **Payment Gateway Integration:** Facilitated seamless payment processing directly within Salesforce.
- **Supplier API Integration:** Real-time synchronization with suppliers for efficient stock replenishment.

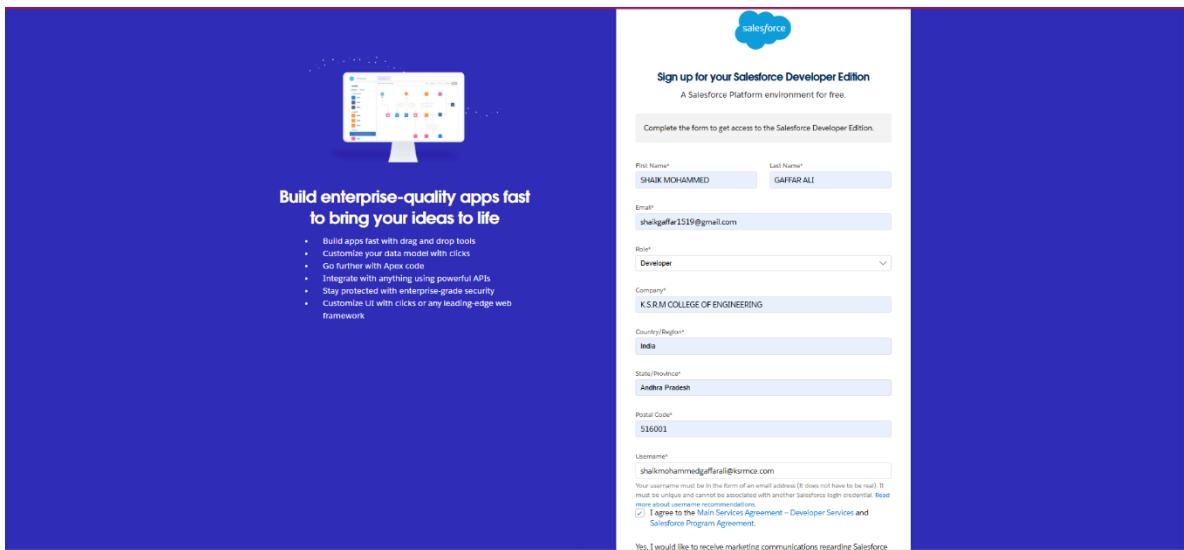
Detailed Steps to Solution Design

The solution design for the Garage Management System (GMS) was meticulously planned and executed to ensure that the platform effectively meets the business requirements. Below is a step-by-step guide that outlines the phases of design and implementation using Salesforce.

Step 1: Setting Up the Salesforce Environment

Create a Developer Account:

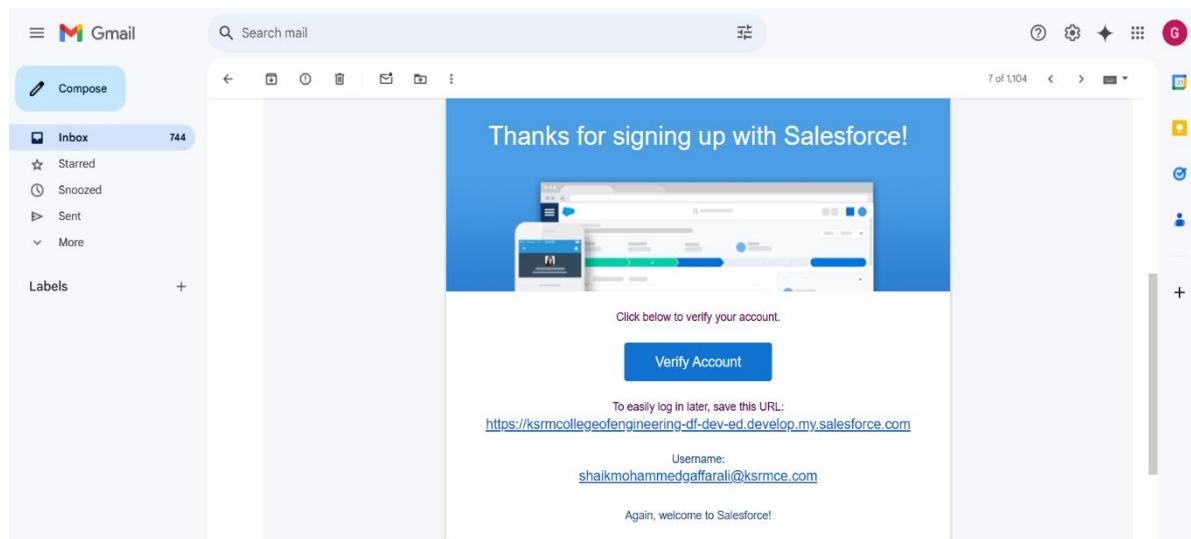
1. Navigate to the Salesforce Developer Signup page at <https://developer.salesforce.com/signup>.



2. Fill in the required details:
 - **First Name & Last Name:** Enter your full name.
 - **Email Address:** Provide a valid email address.
 - **Role:** Choose "Developer."
 - **Company:** Use your college name or organization.
 - **Country:** Select your country (e.g., India).
 - **Postal Code:** Enter your area's postal code.
 - **Username:** Use a combination of your name and company (e.g., username@organization.com).
3. Click **Sign Me Up** to complete the registration.

Activate Your Account:

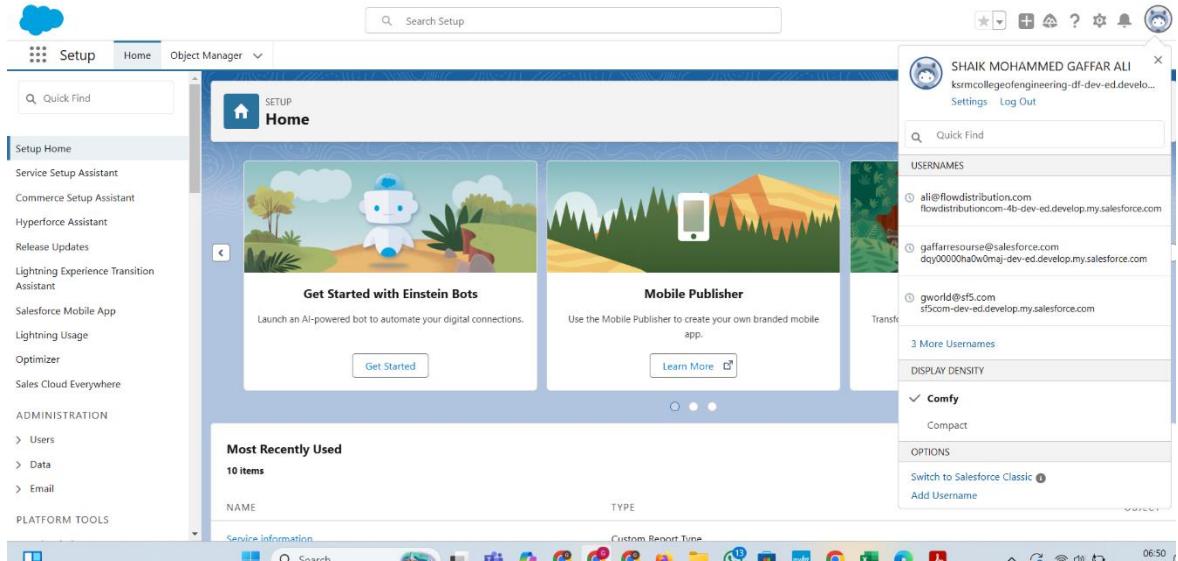
1. Check your email inbox for an account verification email.



2. Click the **Verify Account** link in the email.
3. Set a password, answer the security question, and click **Save Password**.

The screenshot shows the 'Change Your Password' page. At the top, it says 'Change Your Password'. Below that, it asks to enter a new password for lead@sb.oom. It specifies that the password must include at least 8 characters, 1 letter, and 1 number. A red box highlights the 'New Password' field, which contains '.....' and is labeled 'Good'. Another red box highlights the 'Confirm New Password' field, which also contains '.....' and is labeled 'Match'. Below these, there's a 'Security Question' section with a dropdown menu showing 'In what city were you born?'. Under 'Answer', there's a text input field containing 'asdfghjkl'. A large red box highlights the 'Change Password' button at the bottom.

4. You will be redirected to your Salesforce Setup page.

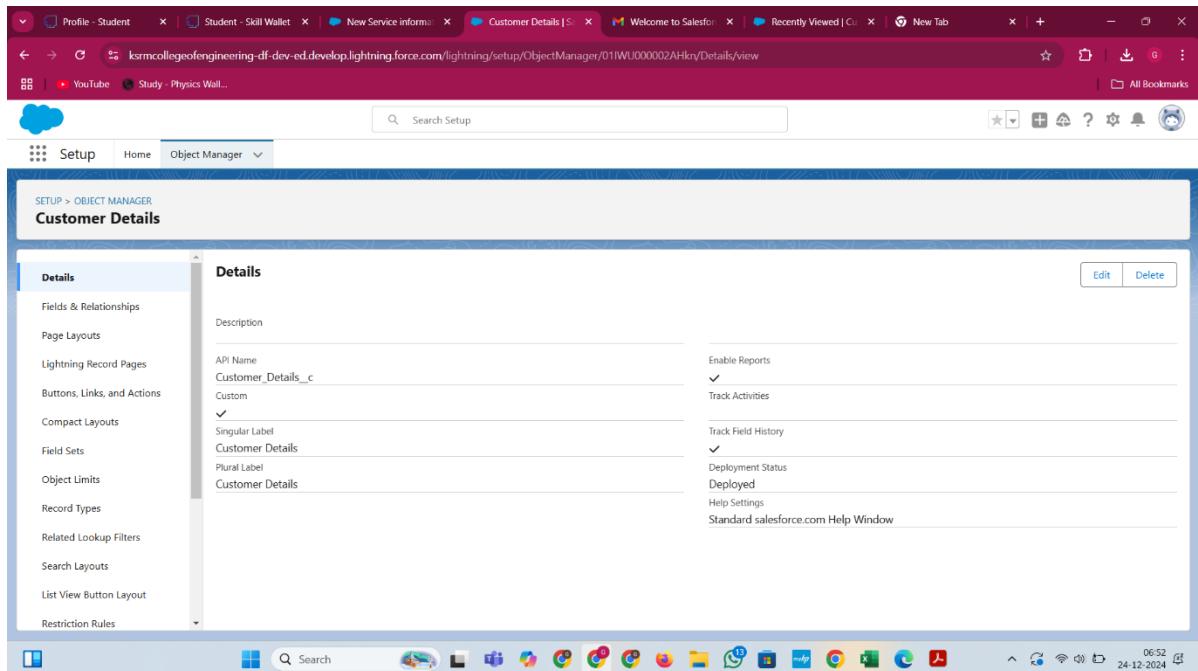


Step 2: Creating Essential Objects

Build Custom Objects to Support Business Operations:

1. Customer Information Object:

- Go to **Setup > Object Manager > Create > Custom Object**.
- Set the **Label Name** to "Customer Information" and the **Plural Name** to "Customer Information Records."
- Define the **Record Name** as "Customer Name" with a **Data Type** of "Text."
- Enable **Track Field History** and **Allow Reports**.
- Save the object.



2. Appointment Object:

- From the **Object Manager**, create a new custom object.

- Label: "Vehicle Details"
- Plural Name: "Vehicles"
- Record Name: "Vehicle ID" with **Data Type** set to "Auto Number" (e.g., format veh-{000}, starting at 1).
- **Enable Track Field History and Allow Search.**
- Save the object.

SETUP > OBJECT MANAGER
Appointment

Details	
Fields & Relationships	Description
Page Layouts	API Name
Lightning Record Pages	Appointment__c
Buttons, Links, and Actions	Custom
Compact Layouts	✓ Singular Label Appointment
Field Sets	Plural Label Appointments
Object Limits	
Record Types	
Related Lookup Filters	
Search Layouts	
List View Button Layout	
Restriction Rules	

Enable Reports
✓
Track Activities
✓
Track Field History
✓
Deployment Status
Deployed
Help Settings
Standard salesforce.com Help Window

3. Service Order Object:

- Add a custom object labeled "Service Order."
- Plural Name: "Service Orders"
- Record Name: "Order ID" with **Auto Number** format (e.g., svc-{000}, starting at 1).
- **Enable Allow Reports and Track Field History.**
- Save the object.

SETUP > OBJECT MANAGER
Service records

Details	
Fields & Relationships	Description
Page Layouts	API Name
Lightning Record Pages	Service_records__c
Buttons, Links, and Actions	Custom
Compact Layouts	✓ Singular Label Service records
Field Sets	Plural Label Service records
Object Limits	
Record Types	
Related Lookup Filters	
Search Layouts	
List View Button Layout	
Restriction Rules	

Enable Reports
✓
Track Activities
✓
Track Field History
✓
Deployment Status
Deployed
Help Settings
Standard salesforce.com Help Window

4. Billing and Feedback Object:

- Create a new object called "Billing and Feedback."
- Plural Name: "Billing Records"
- Record Name: "Feedback ID" with **Auto Number** format (e.g., bill-{000}, starting at 1).
- Enable **Track Field History** and **Allow Search**.
- Save the object.

The screenshot shows the Salesforce Object Manager interface. At the top, there's a search bar with the query 'Billing details and feed'. Below it, a table lists the object with the following details:

Label	API Name	Type	Description	Last Modified	Deployed
Billing details and feedback	Billing_details_and_feedback_c	Custom Object		19/12/2024	

The screenshot shows the 'Details' tab of the 'Billing details and feedback' object in the Object Manager. The left sidebar lists various configuration tabs like Details, Fields & Relationships, Page Layouts, etc. The main pane displays the following details:

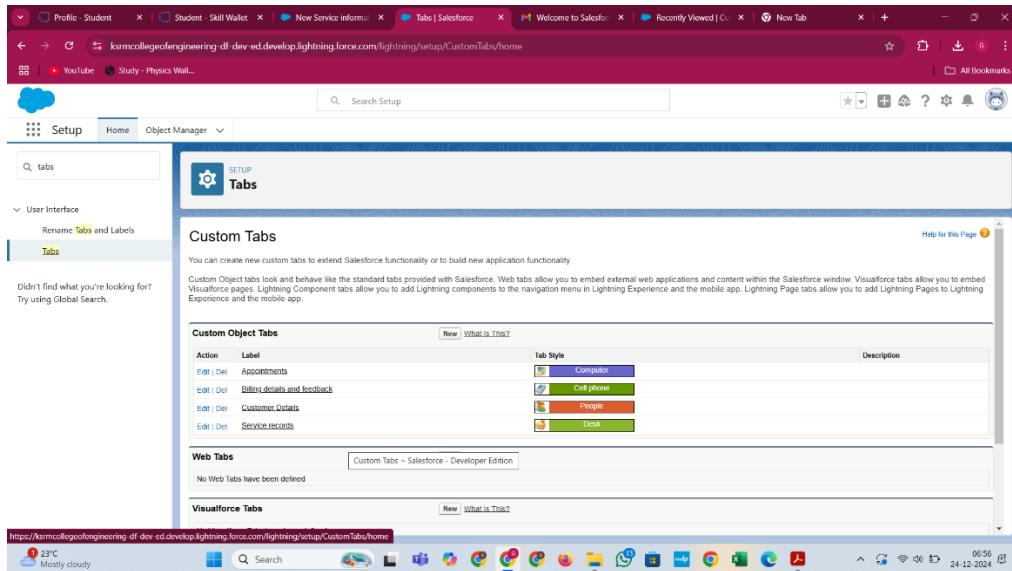
Details	Fields & Relationships	Page Layouts	Lightning Record Pages	Buttons, Links, and Actions	Compact Layouts	Field Sets	Object Limits	Record Types	Related Lookup Filters	Search Layouts	List View Button Layout	Restriction Rules
Description												
API Name	Billing_details_and_feedback_c	Custom										
Singular Label	Billing details and feedback											
Plural Label	Billing details and feedback											

Step 3: Customizing Tabs

Add Custom Object Tabs for Easy Navigation:

1. In **Setup**, search for **Tabs** in the Quick Find box.
2. Select **New** under Custom Object Tabs.
3. For each object (e.g., "Customer Information"):
 - Select the object name.
 - Choose a tab style from the options provided.

- Add the tab to appropriate user profiles by keeping the default settings.
 - For Custom Apps, uncheck “Include Tab” to manage its availability later.
 - Ensure the option to **Append the Tab to User’s Existing Customizations** is selected.
 - Click **Save**.
4. Repeat the process for all other objects (e.g., "Vehicles," "Service Orders," "Billing and Feedback").



Step 4: Establishing Relationships and Fields

Define Relationships Between Objects:

1. **Lookup Relationships:**
 - Create a Lookup relationship between **Service Orders** and **Vehicles** to link specific service details to a vehicle.
2. **Master-Detail Relationships:**
 - Establish a Master-Detail relationship between **Billing Records** and **Service Orders** to ensure billing entries are tightly linked to completed orders.

Add Custom Fields for Operational Needs:

- Add fields like **Appointment Date**, **Technician Assigned**, and **Service Status** in relevant objects to capture all necessary information.
- Use field-level security to restrict access based on roles.

Automating Business Processes:

Leverage Salesforce Automation Features:

1. **Workflow Rules:**
 - Automate email reminders for scheduled appointments or pending payments.
2. **Flows:**
 - Create a flow to update the status of a service order after it is completed.

Validation Rules:

- Ensure mandatory fields like "Vehicle Registration Number" are filled before saving records.
- Restrict users from creating duplicate entries based on the customer's contact number or email.

Testing and Deployment

Test the Solution:

- Test each custom object, tab, and automation in a Salesforce Sandbox environment.
- Validate relationships and ensure data integrity between objects.

Deploy to Production:

- Migrate configurations and customizations using Salesforce Change Sets.
- Conduct final testing in the production environment to verify functionality.

Step 5: Creating a Lightning App

Design a Lightning App to Enhance User Interface:

1. Navigate to **Setup** and search for **App Manager** in the Quick Find box.
2. Click on **App Manager** and select **New Lightning App**.
3. Fill in the app details:
 - **App Name:** "Garage Management Application"
 - Click **Next**.
4. In the **App Options** page, leave the default settings and click **Next**.
5. For **Utility Items**, leave the default settings and click **Next**.
6. Select the relevant items (e.g., "Customer Details," "Appointments," "Service Records," "Billing Details and Feedback," "Reports and Dashboards") from the search bar, then use the arrow button to move them to the selection list. Click **Next**.
7. **Assign User Profiles:**
 - Search for and select **System Administrator** from the profiles search bar.
 - Click the arrow button to add the profile, then click **Save & Finish**.

24 items • Sorted by App Name • Filtered by All appmenuitems - TabSet Type, App Type

App Name ↑	Developer Name	Description	Last Modified ...	Ap...	Vi...
1 All Tabs	AllTabSet		19/12/2024, 9:53 pm	Classic	▼
2 Analytics Studio	Insights	Build CRM Analytics dashboards and apps	19/12/2024, 9:53 pm	Classic	▼
3 App Launcher	AppLauncher	App Launcher tabs	19/12/2024, 9:53 pm	Classic	▼
4 Automation	FlowsApp	Automate business processes and repetitive tasks.	19/12/2024, 9:58 pm	Lightning	▼
5 Bolt Solutions	LightningBolt	Discover and manage business solutions designed for your industry.	19/12/2024, 9:57 pm	Lightning	▼
6 Business Rules Engine	ExpressionSetConsole	Create and maintain business rules that perform complex lookups and ca...	19/12/2024, 9:53 pm	Lightning	▼
7 Community	Community	Salesforce CRM Communities	19/12/2024, 9:53 pm	Classic	▼
8 Content	Content	Salesforce CRM Content	19/12/2024, 9:53 pm	Classic	▼
9 Data Manager	DataManager	Use Data Manager to view limits, monitor usage, and manage recipes.	19/12/2024, 9:53 pm	Lightning	▼
10 Digital Experiences	SalesforceCMS	Manage content and media for all of your sites.	19/12/2024, 9:53 pm	Lightning	▼
11 Garage Management Application	Garage_Management_Application		19/12/2024, 10:53 pm	Lightning	▼
12 Lightning Usage App	LightningInstrumentation	View Adoption and Usage Metrics for Lightning Experience	19/12/2024, 9:53 pm	Lightning	▼
		Track sales and marketing efforts with CRM objects.	19/12/2024, 9:53 pm	Classic	▼

24 items • Sorted by App Name • Filtered by All appmenuitems - TabSet Type, App Type

App Name ↑	Developer Name	Description	Last Modified ...	Ap...	Vi...
12 Lightning Usage App	LightningInstrumentation	View Adoption and Usage Metrics for Lightning Experience	19/12/2024, 9:53 pm	Lightning	▼
13 Marketing CRM Classic	Marketing	Track sales and marketing efforts with CRM objects.	19/12/2024, 9:53 pm	Classic	▼
14 Platform	Platform	The fundamental Lightning Platform	19/12/2024, 9:53 pm	Classic	▼
15 Queue Management	QueueManagement	Create and manage queues for your business.	19/12/2024, 9:53 pm	Lightning	▼
16 Sales	Sales	The world's most popular sales force automation (SFA) solution	19/12/2024, 9:53 pm	Classic	▼
17 Sales	LightningSales	Manage your sales process with accounts, leads, opportunities, and more	19/12/2024, 9:53 pm	Lightning	▼
18 Sales Console	LightningSalesConsole	(Lightning Experience) Lets sales reps work with multiple records on one ...	19/12/2024, 9:53 pm	Lightning	▼
19 Salesforce Chatter	Chatter	The Salesforce Chatter social network, including profiles and feeds	19/12/2024, 9:53 pm	Classic	▼
20 Salesforce Scheduler Setup	LightningScheduler	Set up personalized appointment scheduling.	19/12/2024, 9:56 pm	Lightning	▼
21 Service	Service	Manage customer service with accounts, contacts, cases, and more	19/12/2024, 9:53 pm	Classic	▼
22 Service Console	LightningService	(Lightning Experience) Lets support agents work with multiple records ac...	19/12/2024, 9:53 pm	Lightning	▼
23 Site.com	Sites	Build pixel-perfect, data-rich websites using the drag-and-drop Site.com ...	19/12/2024, 9:53 pm	Classic	▼
24 Subscription Management	RevenueCloudConsole	Get started automating your revenue processes	19/12/2024, 9:53 pm	Lightning	▼

Step 6: Creating Fields for Customer Details Object

Define Custom Fields for Enhanced Data Capture:

- From **Setup**, go to **Object Manager** and search for the **Customer Details** object.
- Click on the object name and select **Fields & Relationships**, then click **New**.
- For the field type, choose **Phone** and click **Next**.
- Fill out the following details:
 - **Field Label:** "Phone Number"
 - **Field Name:** This will auto-generate.
- Click **Next**, then **Save & New** to create additional fields.

The screenshot shows the Salesforce Object Manager interface. At the top, there are several tabs including 'Profile - Student', 'Student - Skill Wallet', 'New Service inform...', 'Object Manager | Sa...', 'Welcome to Salesfor...', 'Recently Viewed | Cu...', 'New Tab', and 'YouTube'. Below the tabs, the URL is 'ksrmcollegeofengineering df-dev-ed.lightning.force.com/lightning/setup/ObjectManager/home'. The main content area has a header 'SETUP Object Manager' with a search bar 'Search Setup' and buttons for 'Billing details and feed', 'Schema Builder', and 'Create'. A table lists one item: 'Billing details and feedback' (API Name: Billing_details_and_feedback__c, Type: Custom Object, Last Modified: 19/12/2024). The bottom status bar shows 'Very humid Now' and the date '24-12-2024 06:58'.

Add Email Field to Customer Details:

1. Repeat the previous steps, but this time select **Email** as the field type.
2. Enter the following details:
 - **Field Label:** "Gmail"
 - **Field Name:** This will auto-generate.
3. Click **Next**, then **Save & New** to continue adding fields.

Create Lookup Field in Appointment Object:

1. In **Object Manager**, select the **Appointment** object.
2. Click on **Fields & Relationships** and select **New**.
3. Choose the **Lookup Relationship** field type.
4. Set the relevant relationship between **Appointment** and **Customer**.

The screenshot shows the 'Fields & Relationships' page for the 'Billing details and feedback' object. The left sidebar lists various setup items like Page Layouts, Lightning Record Pages, etc. The main table displays eight fields:

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)		

 The bottom status bar shows 'Very humid Now' and the date '24-12-2024 06:58'.

Add Checkbox, Date, Currency, Text, Picklist, and Formula Fields:

- Create a **Checkbox** field on the **Appointment** object for confirming service status.
- Add a **Date** field on the **Appointment** object for scheduling purposes.
- Create a **Currency** field on the **Appointment** object to track service cost.
- Add **Text** fields to the **Billing Details and Feedback** object to capture customer reviews.
- Create a **Picklist** field in the **Service Records** object to categorize service types.
- Define a **Formula** field in the **Service Records** object to calculate the total service cost based on variables.

Step 7: Creating Validation Rules

Establish Validation Rules to Ensure Data Integrity:

1. For the **Appointment** object:
 - In **Setup**, go to **Object Manager**, search for **Appointment** and click **Edit**.
 - In the **Validation Rules** section, click **New**.

The screenshot shows the Salesforce Setup interface. The top navigation bar includes tabs for Profile - Student, Student - Skill Wallet, New Service Informa..., Appointment | Sales..., Welcome to Salefor..., Recently Viewed | C..., and New Tab. Below the navigation bar, there's a toolbar with icons for Home, Object Manager, and other setup functions. The main content area is titled 'SETUP > OBJECT MANAGER' and 'Appointment'. On the left, a sidebar lists various object configuration options like 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, and List View Button Layout. The right pane is titled 'Validation Rules' and shows a table with one item: 'Vehicle' under 'RULE NAME', 'Vehicle number plate' under 'ERROR LOCATION', 'Please enter valid number' under 'ERROR MESSAGE', and 'ACTIVE' status with 'SHAIK MOHAMMED GAFFAR ALI, 20/12/2024, 12:55 am' under 'MODIFIED BY'. A 'New' button is visible at the top right of the table. The bottom of the screen shows the Windows taskbar with various application icons and the system clock indicating 06:59 on 24-12-2024.

2. For the **Billing Details and Feedback** object:
 - In **Setup**, go to **Object Manager**, search for **Billing Details and Feedback**, and click **Edit**.
 - In the **Validation Rules** section, click **New**.
 - Set validation criteria such as ensuring **Billing Amount** is greater than zero.

- Click **Save** to finalize the rule.

Billing details and feedback

Validation Rules

Rule Name	Error Location	Error Message	Active	Modified By
rating_should_be_less_than_5	Rating for service	rating should be from 1 to 5	✓	SHAIK MOHAMMED GAFFAR ALI, 20/12/2024, 12:58 am

Step 8: Creating Matching and Duplicate Rules

Set Up Matching Rules to Prevent Duplicate Customer Records:

1. In **Setup**, search for **Matching Rules** in the Quick Find box.
2. Click **Matching Rules**, then select **New Rule**.

All Matching Rules

Action	Rule Name	Object	Status	Description	Last Modified Date	Last Modified By
Det Deactivate	Matching customer details	Customer Details	Active	Matching rule for account records. More Info	20/12/2024	SGAFE
Deactivate	Standard Account Matching Rule	Account	Active	Matching rule for contact records. More Info	19/12/2024	SGAFE
Deactivate	Standard Contact Matching Rule	Contact	Active	Matching rule for lead records. More Info	19/12/2024	SGAFE
Deactivate	Standard Lead Matching Rule	Lead	Active	Matching rule for lead records. More Info	19/12/2024	SGAFE

3. Define the matching criteria for customer details, such as matching **Phone Number** and **Email Address** to identify duplicates.
4. Click **Save** to create the rule.

Create Duplicate Rules for Data Quality Assurance:

1. In **Setup**, search for **Duplicate Rules** in the Quick Find box.
2. Click **Duplicate Rules**, then click **New Rule**.
3. Select the **Customer Details** object.

The screenshot shows the Salesforce Duplicate Rules page. The left sidebar has a search bar and navigation links for Data (Duplicate Management, Duplicate Error Logs, Duplicate Rules), Setup, and Matching Rules. The main area title is "All Duplicate Rules". It includes a "What Are Duplicate Rules?" section and a "View: All Duplicate Rules" dropdown. Below is a table listing four duplicate rules:

Rule Name	Description	Object	Matching Rule	Active	Last Modified By	Last Modified Date
Customer Detail duplicate		Customer Details	Matching customer details	✓	SGAFF	20/12/2024
Standard Account Duplicate Rule	Identify accounts that duplicate other accounts.	Account	Standard Account Matching Rule	✓	SGAFF	19/12/2024
Standard Contact Duplicate Rule	Identify contacts that duplicate other contacts and leads.	Contact	Standard Lead Matching Rule Standard Contact Matching Rule	✓	SGAFF	19/12/2024
Standard Lead Duplicate Rule	Identify leads that duplicate other leads and contacts.	Lead	Standard Lead Matching Rule Standard Contact Matching Rule	✓	SGAFF	19/12/2024

The status bar at the bottom shows the URL <https://ksrmcollegeofengineering-df-dev-ed.lightning.force.com/lightning/setup/DuplicateRules/home>, the date 24-12-2024, and the time 07:03.

4. Define the conditions under which duplicates are flagged, such as allowing duplicates but warning users when a record with matching details already exists.
5. Click **Save** to finalize the rule.

Step 9: Creating Manager and Salesperson Profiles

Create Customized Profiles for Different User Roles:

1. Navigate to **Setup** and type **Profiles** in the Quick Find box.
2. Click on **Profiles**, then select an existing profile (e.g., **Standard User**) to clone.
3. Give the new profile a name, such as **Manager**, and click **Save** to create the profile.

Create a Profile for Salesperson:

1. Again, go to **Setup** and type **Profiles** in the Quick Find box.
2. Click on **Profiles**, then clone the **Salesforce Platform User** profile.
3. Name the profile **Salesperson** and click **Save**.

The screenshot shows the Salesforce Setup interface with the 'Profiles' page open. The sidebar on the left has 'Users' expanded, with 'Profiles' selected. The main area displays a table of profiles:

Action	Profile Name	User License	Custom
<input type="checkbox"/>	Analytics Cloud Integration User	Analytics Cloud Integration User	<input type="checkbox"/>
<input type="checkbox"/>	Analytics Cloud Security User	Analytics Cloud Integration User	<input type="checkbox"/>
<input type="checkbox"/>	Authenticated Website	Authenticated Website	<input type="checkbox"/>
<input type="checkbox"/>	Authenticated Website	Authenticated Website	<input type="checkbox"/>
<input type="checkbox"/>	B2B Recreenging Portal Buyer Profile	External Apps Login	<input checked="" type="checkbox"/>
<input type="checkbox"/>	Chatter External User	Chatter External	<input type="checkbox"/>
<input type="checkbox"/>	Chatter Free User	Chatter Free	<input type="checkbox"/>
<input type="checkbox"/>	Chatter Moderator User	Chatter Free	<input type="checkbox"/>
<input type="checkbox"/>	Contract Manager	Salesforce	<input type="checkbox"/>
<input type="checkbox"/>	Cross Org Data Proxy User	XOrg Proxy User	<input type="checkbox"/>
<input type="checkbox"/>	Custom_Marketing_Profile	Salesforce	<input checked="" type="checkbox"/>
<input type="checkbox"/>	Custom_Sales_Profile	Salesforce	<input checked="" type="checkbox"/>
<input type="checkbox"/>	Custom_Support_Profile	Salesforce	<input checked="" type="checkbox"/>

The screenshot shows the Salesforce Setup interface with the 'Profiles' page open. The sidebar on the left has 'Users' expanded, with 'Profiles' selected. The main area displays a table of profiles:

Action	Profile Name	User License	Custom
<input type="checkbox"/>	sales.person	Salesforce Platform	<input checked="" type="checkbox"/>
<input type="checkbox"/>	Salesforce API Only System Integrations	Salesforce Integration	<input checked="" type="checkbox"/>
<input type="checkbox"/>	Silver_Partner_User	Silver Partner	<input type="checkbox"/>
<input type="checkbox"/>	Solution Manager	Salesforce	<input type="checkbox"/>
<input type="checkbox"/>	Standard Platform User	Salesforce Platform	<input type="checkbox"/>
<input type="checkbox"/>	Standard User	Salesforce	<input type="checkbox"/>
<input type="checkbox"/>	System Administrator	Salesforce	<input type="checkbox"/>

Step 10: Defining Roles and Role Hierarchy

Create a Manager Role and Define Role Hierarchy:

1. In **Setup**, search for **Roles** in the Quick Find box.
2. Click on **Set Up Roles**, then click **Expand All** to view the role hierarchy.
3. Under the appropriate parent role (e.g., CEO), click **Add Role**.
4. Name the new role **Manager** (the role name will auto-populate) and click **Save**.

Create Additional Roles Under the Manager Role:

1. In **Setup**, search for **Roles** again in the Quick Find box.
2. Click on **Set Up Roles**, then find the **CEO** role.
3. Click the plus icon next to the CEO role and add roles under **Manager** to define a clear hierarchy of subordinates.

Understanding Roles

Set up your Role Hierarchy to control how your organization reports on and accesses data.

Sample Role Hierarchy

View other sample Role Hierarchies: [Territory-based Sample](#)

Executive Staff

- CEO** President, CFO, VP, Sales
- View & edit data, roll up forecasts, & generate reports for all users directly below them. * Can't access data of other Executive Staff

Western Sales Director

- Western Sales Director** Director of W. Sales
- View & edit data, roll up forecasts, & generate reports for all users directly below them. * Can't access data of users above or at same level

Eastern Sales Director

- Eastern Sales Director** Director of E. Sales
- View & edit data, roll up forecasts, & generate reports for all users directly below them. * Can't access data of users above or at same level

International Sales Rep

- Sales Director** Director of Int'l Sales
- View & edit data, roll up forecasts, & generate reports only for own data & for users directly below them. * Can't access data of users above or at same level

Sample role hierarchy based on territory

Western Sales Rep

- Western Sales Rep** CA Sales Rep, OR Sales Rep
- View & edit data, roll up forecasts, & generate reports for all users directly below them. * Can't access data of users above or at same level

Eastern Sales Rep

- Eastern Sales Rep** NY Sales Rep, MA Sales Rep
- View & edit data, roll up forecasts, & generate reports for all users directly below them. * Can't access data of users above or at same level

International Sales Rep

- Asian Sales Rep**, **European Sales Rep**
- View & edit data, roll up forecasts, & generate reports only for own data & for users directly below them. * Can't access data of users above or at same level

Set Up Roles

Don't show this page again

Creating the Role Hierarchy

You can build on the existing role hierarchy shown on this page. To insert a new role, click **Add Role**.

Your Organization's Role Hierarchy

[Collapse All](#) [Expand All](#) [Show in tree view](#)

- K.S.R.M COLLEGE OF ENGINEERING**
 - CEO** Edit | Del | Assign
 - Add Role**
 - CFO** Edit | Del | Assign
 - Add Role**
 - COO** Edit | Del | Assign
 - Add Role**
 - Manager** Edit | Del | Assign
 - Add Role**
 - sales person** Edit | Del | Assign
 - Add Role**
 - SVP_Customer_Service & Support** Edit | Del | Assign
 - Add Role**
 - Customer_Support_International** Edit | Del | Assign
 - Add Role**
 - Customer_Support_North_America** Edit | Del | Assign
 - Add Role**
 - Installation & Repair Services** Edit | Del | Assign
 - Add Role**

Step 11: Creating Users

Add Users to Salesforce Platform:

1. Go to **Setup**, type **Users** in the Quick Find box, and click **Users**.
2. Click **New User** to create a new user.
3. Complete the fields for the user, including:

- **Role:** Assign the appropriate role (e.g., **Salesperson**).
- **User License:** Select **Salesforce Platform**.
- **Profile:** Choose the relevant profile (e.g., **Salesperson**).

4. Click **Save** to create the user.

Repeat the process for additional users, making sure to assign each user the appropriate roles, licenses, and profiles, ensuring at least three users are created under the **Salesperson** profile.

Step 12: Creating New Public Groups

Set Up Public Groups to Manage User Access:

1. Go to **Setup** and type **Users** in the Quick Find box.
2. Click on **Public Groups**, then click **New** to create a new group.
3. Name the group **Sales Team** and define the group's settings.
4. Click **Save** to finalize the creation of the public group.

Step 13: Configuring Sharing Settings

Adjust Sharing Settings for Objects:

1. Navigate to **Setup**, and type **Sharing Settings** in the Quick Find box.
2. Select **Sharing Settings**, then click **Edit**.
3. In the **Organization-Wide Default (OWD)** settings, locate the **Service Records Object** and set its sharing model to **Private**.
4. Click **Save** to apply the changes.

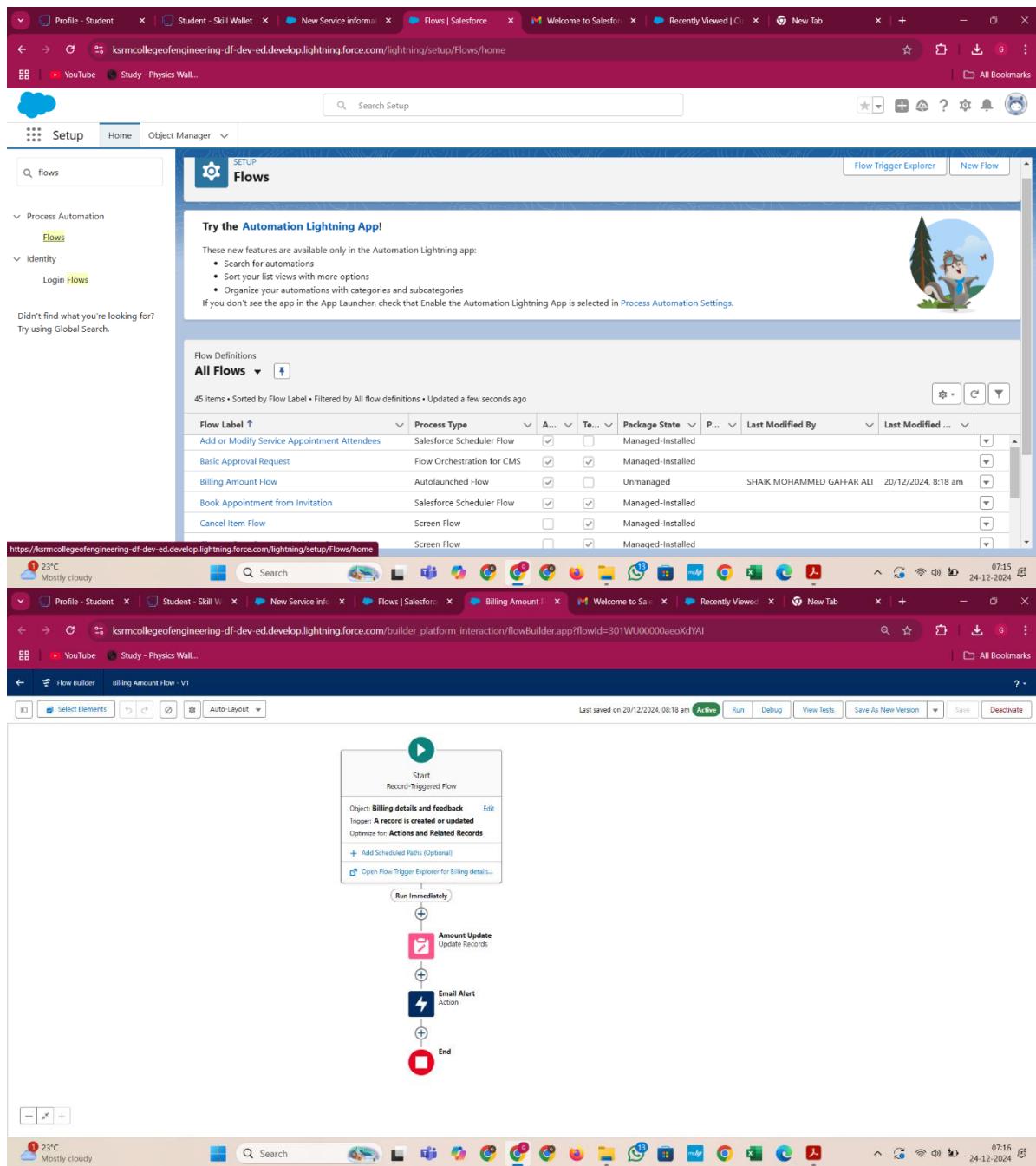
The screenshot shows the 'Sharing Settings' page in the Salesforce Setup. The left sidebar has 'Sharing' selected under 'Security'. The main area displays the 'Sharing Settings' page with the heading 'Sharing Settings'. It includes a note about organization-wide default sharing settings and a link to 'Background Jobs'. A dropdown menu 'Manage sharing settings for:' is set to 'All Objects'. The 'Default Sharing Settings' section contains a table titled 'Organization-Wide Defaults'. The table has columns for 'Object', 'Default Internal Access', 'Default External Access', and 'Grant Access Using Hierarchies'. The table lists various objects like Lead, Account, Contact, Order, Asset, Opportunity, Case, Campaign, Campaign Member, and User, with their respective sharing settings. The 'User' row is currently selected. The bottom right corner of the page shows the date and time as '24-12-2024 07:14'.

Object	Default Internal Access	Default External Access	Grant Access Using Hierarchies
Lead	Public Read/Write/Transfer	Private	✓
Account and Contract	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	✓

Step 14: Creating a Flow

Create a Record-Triggered Flow:

1. In **Setup**, type **Flow** in the Quick Find box, and select **Flows**.
2. Click on **New Flow**, then choose **Record-Triggered Flow**.
3. Click **Create** to start the flow design process.
4. Define the trigger conditions and flow logic as per the requirements of the project.



Step 15: Apex Handler for Amount Distribution

Use Case: This Apex handler is designed to manage the distribution of amounts based on the services selected for the vehicle.

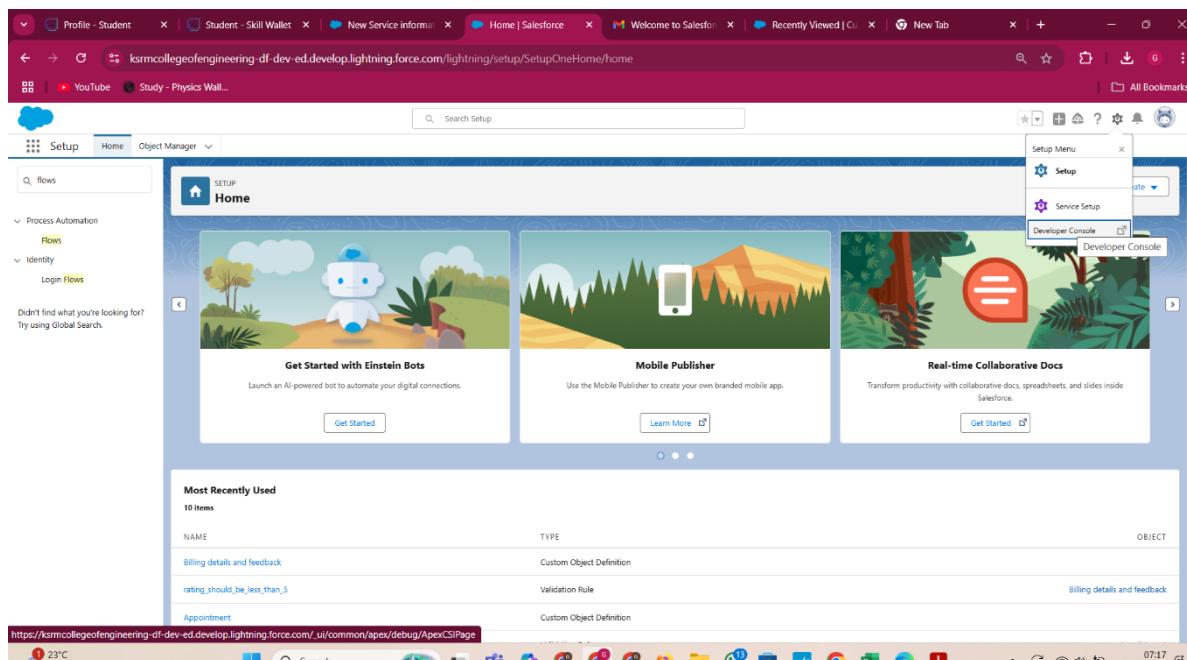
Creating the Apex Class:

1. Log in to your **Trailhead account** and click on the gear icon in the top-right corner.
2. Select **Developer Console** to open the console window.
3. In the toolbar, click on **File**, then select **New** and click **Apex Class**.
4. Name the class **AmountDistributionHandler** and write the required logic for handling the amount distribution.

5. Click **Save** once the code is written.
6. After saving, click **Run All** to execute the class.

Creating the Trigger for Amount Distribution:

1. Open the **Developer Console** by clicking the gear icon again in your **Trailhead account**.
2. In the console, click **File**, then **New** and select **Trigger**.
3. Name the trigger **AmountDistribution** and select the **Appointment** object as the target for this trigger.
4. Write the necessary trigger logic for amount distribution based on the selected services.
5. Click **Save**, then **Run All** to execute the trigger.



Developer Console - Google Chrome
ksrmcollegeofengineering-df-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage

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

Logs	Tests	Checkpoints	Query Editor	View State	Progress	Problems									
Status	Test Run				Enqueued Time	Duration	Failures	Total	Overall Code Coverage						
								<table border="1"> <thead> <tr> <th>Class</th> <th>Percent</th> <th>Lines</th> </tr> </thead> <tbody> <tr> <td>Overall</td> <td>0%</td> <td></td> </tr> </tbody> </table>		Class	Percent	Lines	Overall	0%	
Class	Percent	Lines													
Overall	0%														

Developer Console - Google Chrome
krmcollegeofengineering-df-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage

File Edit Debug Test Workspace Help < >

AmountDistributionHandler.apxc AmountDistribution.apxt

Code Coverage: None API Version: 62 Go To

```
1 public class AmountDistributionHandler {  
2  
3  
4  
5 public static void amountDist(list<Appointment__c> listApp){  
6  
7     list<Service_records__c> serList = new list <Service_records__c>();  
8  
9  
10    for(Appointment__c app : listApp){  
11  
12        if(app.Maintenance_service__c == true && app.Repairs__c == true && app.Replacement_Parts__c == true){  
13            app.Service_Amount__c = 10000;  
14  
15        }  
16  
17        else if(app.Maintenance_service__c == true && app.Repairs__c == true){  
18  
19    }
```

Logs Tests Checkpoints Query Editor View State Progress Problems

Status	Test Run	Enqueued Time	Duration	Failures	Total
Overall Code Coverage					
Class	Percent	Lines			
Overall	0%				
AmountDistribution	0%	0/2			
AmountDistributionHandler	0%	0/17			

23°C Mostly cloudy Search 07:19 24-12-2024

Developer Console - Google Chrome
krmcollegeofengineering-df-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage

File Edit Debug Test Workspace Help < >

AmountDistributionHandler.apxc AmountDistribution.apxt

Code Coverage: None API Version: 62 Go

```
20  
21        app.Service_Amount__c = 5000;  
22  
23    }  
24  
25    else if(app.Maintenance_service__c == true && app.Replacement_Parts__c == true){  
26  
27        app.Service_Amount__c = 8000;  
28  
29    }  
30  
31    else if(app.Repairs__c == true && app.Replacement_Parts__c == true){  
32  
33        app.Service_Amount__c = 7000;  
34  
35    }  
36  
37    else if(app.Maintenance_service__c == true){  
38  
39        app.Service_Amount__c = 2000;
```

Logs Tests Checkpoints Query Editor View State Progress Problems

Status	Test Run	Enqueued Time	Duration	Failures	Total
Overall Code Coverage					
Class	Percent	Lines			
Overall	0%				
AmountDistribution	0%	0/2			
AmountDistributionHandler	0%	0/17			

Developer Console - Google Chrome
krmcollegeofengineering-df-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage

File Edit Debug Test Workspace Help < >

AmountDistributionHandler.apxc AmountDistribution.apxt

Code Coverage: None API Version: 62

```
41  
42  
43    else if(app.Repairs__c == true){  
44  
45        app.Service_Amount__c = 3000;  
46  
47    }  
48  
49    else if(app.Replacement_Parts__c == true){  
50  
51        app.Service_Amount__c = 5000;  
52  
53    }  
54  
55  
56}  
57  
58}
```

Logs Tests Checkpoints Query Editor View State Progress Problems

Status	Test Run	Enqueued Time	Duration	Failures	Total
Overall Code Coverage					
Class	Percent	Lines			
Overall	0%				
AmountDistribution	0%	0/2			
AmountDistributionHandler	0%	0/17			

Step 16: Creating and Managing Reports

Creating a Report Folder:

1. From the **App Launcher**, search for and select **Reports**.
2. Click on the **Reports** tab, then click **New Folder** to create a new folder.
3. Name the folder appropriately (e.g., **Garage Management Reports**) and click **Save**.

Sharing a Report Folder:

1. Go to the **Reports** tab and select the newly created folder (e.g., **Garage Management Reports**).
2. Click on the drop-down menu next to the folder name and select **Share**.
3. In the **Share With** field, choose **Roles**, then search for the **Manager** role.
4. Set the access level to **View**, then click **Share** and **Done**.

Creating a Custom Report Type:

1. In **Setup**, search for **Report Types** in the Quick Find box and select **Report Types**.
2. Click **New Custom Report Type**, then select **Customer Details** as the primary object.
3. Set the **Report Type Label** as **Service Information** and let the **Report Type Name** auto-populate.
4. Enter a description and select **Other Reports** as the category.
5. Set the **Deployment Status** to **Deployed**, then click **Next**.
6. In the **Related Object** section, click **Select Object** and choose the **Appointment** object.
7. Click **Save** to create the custom report type.

Creating a New Report:

1. From the **Reports** tab, click **New Report** to create a new report.
2. Select the **Service Information** report type created earlier, then design the report as per your requirements.
3. Click **Save & Run** to finalize the report.

The screenshot shows the Salesforce Reports interface. The top navigation bar includes links for YouTube, Study - Physics Wall.., Garage Management, Customer Details, Appointments, Service records, Billing details and feedback, Reports, and Dashboards. The Reports tab is selected. On the left, there are three vertical navigation panels: 'Reports' (Recent, Created by Me, Private Reports, Public Reports, All Reports), 'Folders' (All Folders, Created by Me, Shared with Me), and 'Favorites' (All Favorites). The main content area displays a table of reports and folders. The table has columns for Name, Created By, Created On, Last Modified By, and Last Modified Date. The data includes:

	Name	Created By	Created On	Last Modified By	Last Modified Date
REPORTS	Einstein Bot Reports	Automated Process	19/12/2024, 9:53 pm	Automated Process	19/12/2024, 9:53 pm
Recent	Einstein Bot Reports Spring '23	Automated Process	19/12/2024, 9:53 pm	Automated Process	19/12/2024, 9:53 pm
Created by Me	Einstein Bot Reports Summer '23	Automated Process	19/12/2024, 9:53 pm	Automated Process	19/12/2024, 9:53 pm
Private Reports	Einstein Bot Reports Summer '22	Automated Process	19/12/2024, 9:53 pm	Automated Process	19/12/2024, 9:53 pm
Public Reports	Einstein Bot Reports Winter '23	Automated Process	19/12/2024, 9:53 pm	Automated Process	19/12/2024, 9:53 pm
All Reports	Enablement Dashboard Reports Spring '24	Automated Process	19/12/2024, 9:53 pm	Automated Process	19/12/2024, 9:53 pm
FOLDERS	Enablement Dashboard Reports Summer '24	Automated Process	19/12/2024, 9:53 pm	Automated Process	19/12/2024, 9:53 pm
All Folders	Garage Management Folder	SHAIK MOHAMMED GAFFAR ALI	20/12/2024, 12:42 pm	SHAIK MOHAMMED GAFFAR ALI	20/12/2024, 12:42 pm

The bottom of the screen shows the Windows taskbar with various icons and the system tray indicating it's 24°C and mostly cloudy.

Screenshot of the Salesforce Lightning interface showing the 'Reports' section. The left sidebar lists categories like 'Recent', 'Created by Me', 'Private Reports', etc. The main area displays a table of reports with columns for Name, Created By, Created On, Last Modified By, and Last Modified Date. A context menu is open over a report named 'Garage Management Folder'.

Name	Created By	Created On	Last Modified By	Last Modified Date
Einstein Bot Reports	Automated Process	19/12/2024, 9:53 pm	Automated Process	19/12/2024, 9:53 pm
Einstein Bot Reports Spring '23	Automated Process	19/12/2024, 9:53 pm	Automated Process	19/12/2024, 9:53 pm
Einstein Bot Reports Summer '23	Automated Process	19/12/2024, 9:53 pm	Automated Process	19/12/2024, 9:53 pm
Einstein Bot Reports Summer '22	Automated Process	19/12/2024, 9:53 pm	Automated Process	19/12/2024, 9:53 pm
Einstein Bot Reports Winter '23	Automated Process	19/12/2024, 9:53 pm	Automated Process	19/12/2024, 9:53 pm
Enablement Dashboard Reports Spring '24	Automated Process	19/12/2024, 9:53 pm	Automated Process	19/12/2024, 9:53 pm
Enablement Dashboard Reports Summer '24	Automated Process	19/12/2024, 9:53 pm	Automated Process	19/12/2024, 9:53 pm
Garage Management Folder	SHAIK MOHAMMED GAFFAR ALI	20/12/2024, 10:40 pm	SHAIK MOHAMMED GAFFAR ALI	20/12/2024, 12:40 pm

Screenshot of the Salesforce Setup interface showing the 'Report Types' page. The left sidebar includes 'Analytics' and 'Report Types'. The main area displays a diagram illustrating how objects relate to custom report types. A note states: 'Note that the visibility of custom report types in the report wizard is controlled by users' access to the objects in the report type.'

Screenshot of the Salesforce Lightning interface showing the 'New Service information Report' page. The top navigation bar includes 'YouTube' and 'Study - Physics Wall..'. The main area features a chart titled 'PERFORMANCE' showing the sum of payment paid for different service ratings (0, 1, 4, 5). Below the chart is a table of service details:

Customer Name	Appointment Date	Service Status	Payment Paid
1 Mac	13/12/2024	Completed	₹5,000
2 Abdullah	19/12/2024	-	₹2,000
3 Abdullah	19/12/2024	Completed	₹2,000
4 Ibn Abdullah	10/12/2024	Completed	₹10,000
5			₹19,000

Step 17: Creating and Managing Dashboards

Creating a New Dashboard Folder:

1. Open the **App Launcher** and search for **Dashboards**.
2. Click on the **Dashboards** tab, then click **New Folder**.
3. Name the folder **Service Rating Dashboard** and click **Save**.

Name	Created By	Created On	Last Modified By	Last Modified Date
Enablement Dashboard Spring '24	Automated Process	19/12/2024, 9:53 pm	Automated Process	19/12/2024, 9:53 pm
Enablement Dashboard Summer '24	Automated Process	19/12/2024, 9:53 pm	Automated Process	19/12/2024, 9:53 pm
Service Rating dashboard	SHAIK MOHAMMED GAFFAR ALI	20/12/2024, 10:08 pm	SHAIK MOHAMMED GAFFAR ALI	20/12/2024, 10:08 pm

Creating a New Dashboard:

1. Click on the **Dashboards** tab, then click **New Dashboard**.
2. Select the **Service Rating Dashboard** folder, name the dashboard accordingly, and define the components to include (e.g., graphs, charts).
3. Click **Save** to create and save the new dashboard.

New Service information Report

Sum of Payment Paid

Rating for service

Payment Status: Completed

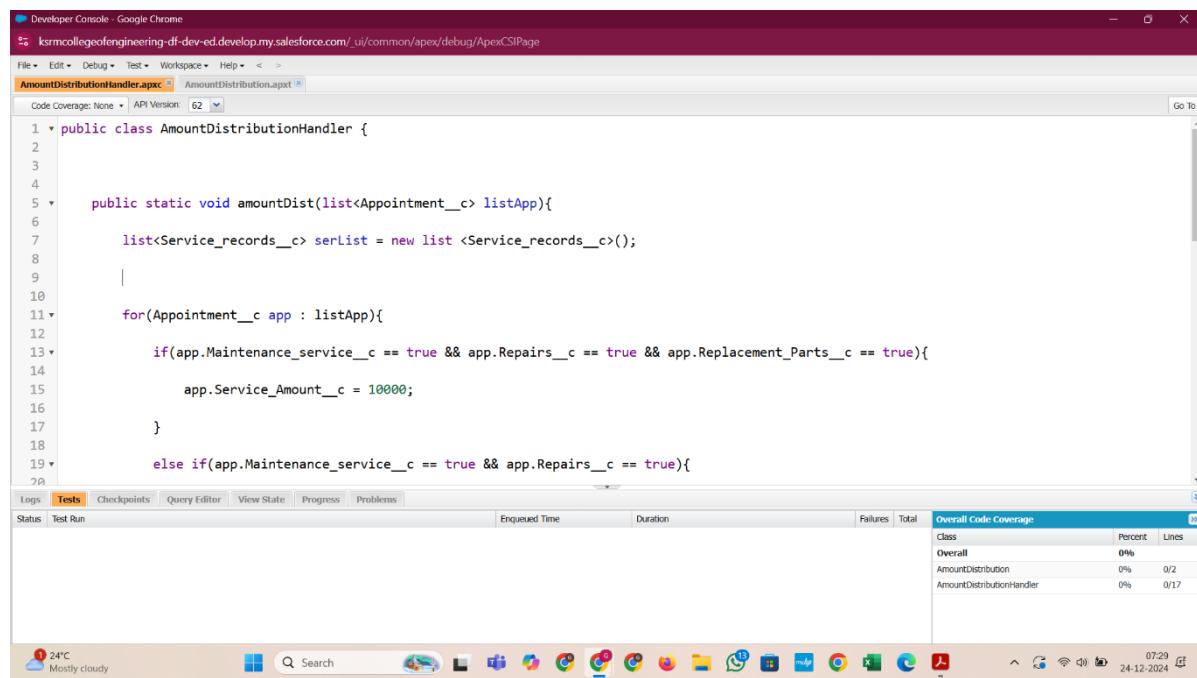
Testing and Validation

The testing and validation phase is vital in ensuring that all components of the Garage Management System (GMS) on Salesforce work as intended. This phase involves both **Unit Testing** for Apex classes and triggers and **User Interface (UI) Testing** for end-user functionality. Below is a detailed explanation of the approach to testing, covering each critical area to verify the solution's integrity.

Step 1: Unit Testing for Apex Classes (Apex Handler)

Use Case: This test is designed to validate the amount distribution logic for each service selected by the customer for their vehicle.

1. **Log into the Trailhead account** and click on the gear icon in the upper-right corner.
2. Select **Developer Console** from the dropdown menu to open a new console window.
3. From the **File** menu, select **New > Apex Class**.
4. Name the class **AmountDistributionHandler**, then write the logic for the class to handle amount distribution based on selected services.
5. Click **Save** to save the class.
6. To run the test, click **Run All** to execute the unit test and ensure the correct functionality.



The screenshot shows the Salesforce Developer Console interface. The top navigation bar includes File, Edit, Debug, Test, Workspace, Help, and a dropdown for API Version (set to 62). The main area displays the code for the `AmountDistributionHandler` class. The code defines a static void method `amountDist` that takes a list of `Appointment__c` objects. It initializes a list of `Service_records__c` objects and iterates through the appointment list. For each appointment, it checks if three specific service types are true: Maintenance_service__c, Repairs__c, and Replacement_Parts__c. If all three are true, it sets the Service_Amount__c to 10000. Otherwise, if only Maintenance_service__c and Repairs__c are true, it sets the Service_Amount__c to 5000. The code ends with a closing brace for the else if block. Below the code editor is a toolbar with tabs for Logs, Tests, Checkpoints, Query Editor, View State, Progress, and Problems. The Tests tab is selected, showing a table for the current test run. The table has columns for Status, Enqueued Time, Duration, Failures, Total, and Overall Code Coverage. The coverage table shows data for three classes: Overall (0%), AmountDistribution (0% / 0/2), and AmountDistributionHandler (0% / 0/17). The bottom of the screen shows the Windows taskbar with various pinned icons and the system tray.

```
1 public class AmountDistributionHandler {  
2  
3  
4  
5     public static void amountDist(list<Appointment__c> listApp){  
6  
7         list<Service_records__c> serList = new list<Service_records__c>();  
8  
9     }  
10  
11    for(Appointment__c app : listApp){  
12  
13        if(app.Maintenance_service__c == true && app.Repairs__c == true && app.Replacement_Parts__c == true){  
14  
15            app.Service_Amount__c = 10000;  
16  
17        }  
18  
19        else if(app.Maintenance_service__c == true && app.Repairs__c == true){  
20            app.Service_Amount__c = 5000;  
21        }  
22    }  
23}
```

Status	Test Run	Enqueued Time	Duration	Failures	Total	Overall Code Coverage
Logs	Logs					Class
Tests	Tests					Overall
Checkpoints	Checkpoints					0%
Query Editor	Query Editor					Percent
View State	View State					Lines
Progress	Progress					0/2
Problems	Problems					0/17

```
Developer Console - Google Chrome
krmcollageofengineering-df-dev-ed-develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage
File Edit Debug Test Workspace Help < >
AmountDistributionHandler.apxc AmountDistribution.apxt
Code Coverage: None API Version: 62 Go To
20
21 app.Service_Amount__c = 5000;
22
23 }
24
25 else if(app.Maintenance_service__c == true && app.Replacement_Parts__c == true){
26
27 app.Service_Amount__c = 8000;
28
29 }
30
31 else if(app.Repairs__c == true && app.Replacement_Parts__c == true){
32
33 app.Service_Amount__c = 7000;
34
35 }
36
37 else if(app.Maintenance_service__c == true){
38
39 app.Service_Amount__c = 2000;
Logs Tests Checkpoints Query Editor View State Progress Problems
Status Test Run Enqueued Time Duration Failures Total Overall Code Coverage
Class Percent Lines
Overall 0% 0/2
AmountDistribution 0% 0/2
AmountDistributionHandler 0% 0/17
24°C Mostly cloudy Search 07:29 24-12-2024
```

```
Developer Console - Google Chrome
krmcollageofengineering-df-dev-ed-develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage
File Edit Debug Test Workspace Help < >
AmountDistributionHandler.apxc AmountDistribution.apxt
Code Coverage: None API Version: 62 Go To
42
43 else if(app.Repairs__c == true){
44
45 app.Service_Amount__c = 3000;
46
47 }
48
49 else if(app.Replacement_Parts__c == true){
50
51 app.Service_Amount__c = 5000;
52
53 }
54
55
56 }
57
58 }
59 }
60
61 }
Logs Tests Checkpoints Query Editor View State Progress Problems
Status Test Run Enqueued Time Duration Failures Total Overall Code Coverage
Class Percent Lines
Overall 0% 0/2
AmountDistribution 0% 0/2
AmountDistributionHandler 0% 0/17
24°C Mostly cloudy Search 07:30 24-12-2024
```

Step 2: Unit Testing for Triggers

Use Case: This trigger will handle the distribution of amounts when an appointment record is created or updated.

1. Open the **Developer Console** by clicking the gear icon and selecting **Developer Console**.
2. In the console, go to **File > New > Trigger**.
3. Name the trigger **AmountDistribution** and select the **Appointment** object to be triggered.
4. Write the trigger code to apply the logic related to amount distribution.

- Click **Save**, then select **Run All** to execute the trigger and validate that the trigger logic works as expected.

The screenshot shows the Salesforce Developer Console in Google Chrome. The URL is <https://ksrmcollegeengineering-df-dev-ed-develop.my.salesforce.com/ui/common/apex/debug/ApexCSIPage>. The tab title is "AmountDistributionHandler.apxc". The code coverage is set to "None" and the API version is "62". The trigger code is:

```

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

```

The "Tests" tab is selected, showing a single "Test Run" entry. The "Overall Code Coverage" table is as follows:

Class	Percent	Lines
Overall	0%	
AmountDistribution	0%	0/2
AmountDistributionHandler	0%	0/17

The status bar at the bottom shows "24°C Mostly cloudy" and the date "24-12-2024 07:30".

Step 3: UI Testing for Customer Details Object

To ensure the **Customer Details** object is correctly capturing user inputs:

- Open the **App Launcher** located on the left side of the screen.
- Type and search for **Garage Management System** and click on it.
- Click on the **Customer Details** tab.
- Click **New** to create a new customer record.
- Fill in the required fields (e.g., Name, Contact Number) and click **Save** to verify if the record is successfully created.

The screenshot shows the Salesforce Lightning interface in a web browser. The URL is https://ksrmcollegeengineering-df-dev-ed-develop.lightning.force.com/lightning/o/Customer_Details__c/list?filterName=_Recent. The top navigation bar includes tabs for "Profile - Student", "Student - Skill Wallet", "New Service informa...", "Customer Review | S...", "Welcome to Salesforce", "Recently Viewed | Cu...", "New Tab", and "All Bookmarks". The sidebar on the left shows "Garage Management..." and "Customer Details". The main content area displays a list of recently viewed customer records:

Customer Name
Mac
Chan Basha
Abdullah
ibn abdullah

The right sidebar shows the user profile "SHAIK MOHAMMED GAFFAR ALI" from "ksrmcollegeengineering-df-dev-ed.develo...", "Settings", and "Log Out". It also includes sections for "Quick Find", "Usernames", "Display Density" (set to "Compact"), and "Options" (with links to "Switch to Salesforce Classic" and "Add Username").

New Customer Details

Information

*Customer Name: Shahid bhai

Owner: SHAIK MOHAMMED GAFFAR ALI

Phone number: 7755331166

Email: shahidma55@gmail.com

Customer Details: Shahid bhai

Related Details

Customer Name: Shahid bhai

Owner: SHAIK MOHAMMED GAFFAR ALI

Phone number: 7755331166

Email: shahidma55@gmail.com

Created By: SHAIK MOHAMMED GAFFAR ALI, 24/12/2024, 7:33 am

Last Modified By: SHAIK MOHAMMED GAFFAR ALI, 24/12/2024, 7:33 am

Customer Details: Chan Basha

Related Details

Customer Name: Chan Basha

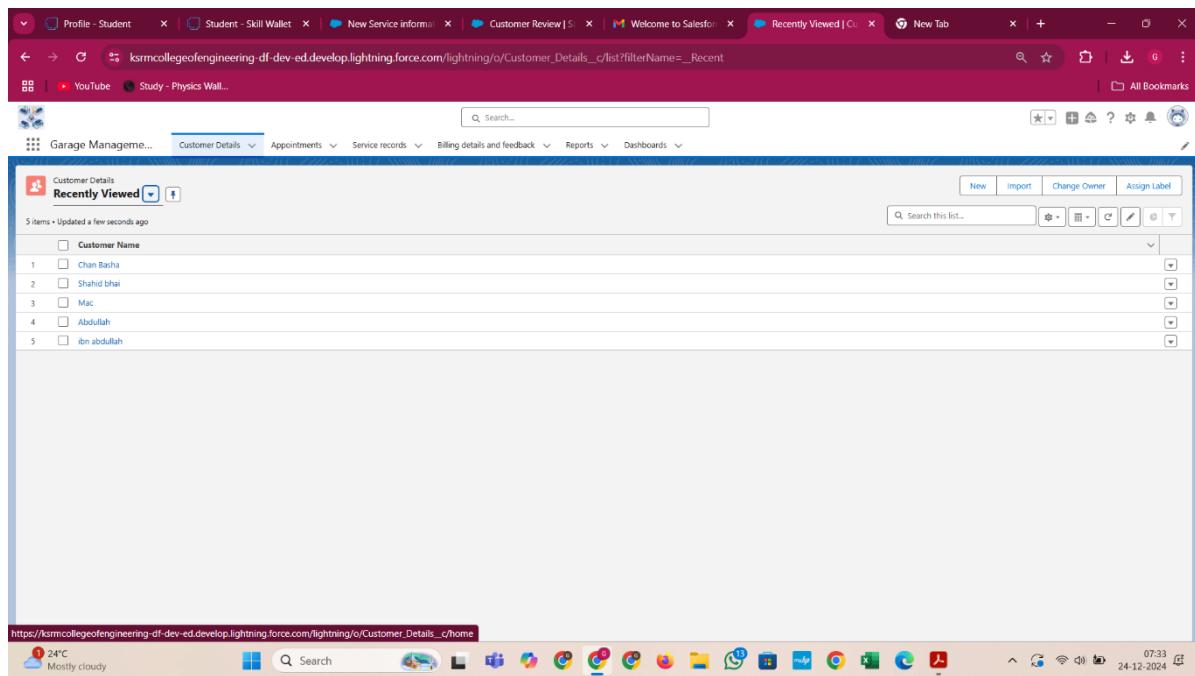
Owner: SHAIK MOHAMMED GAFFAR ALI

Phone number: 123987456

Email: abdullah@gmail.com

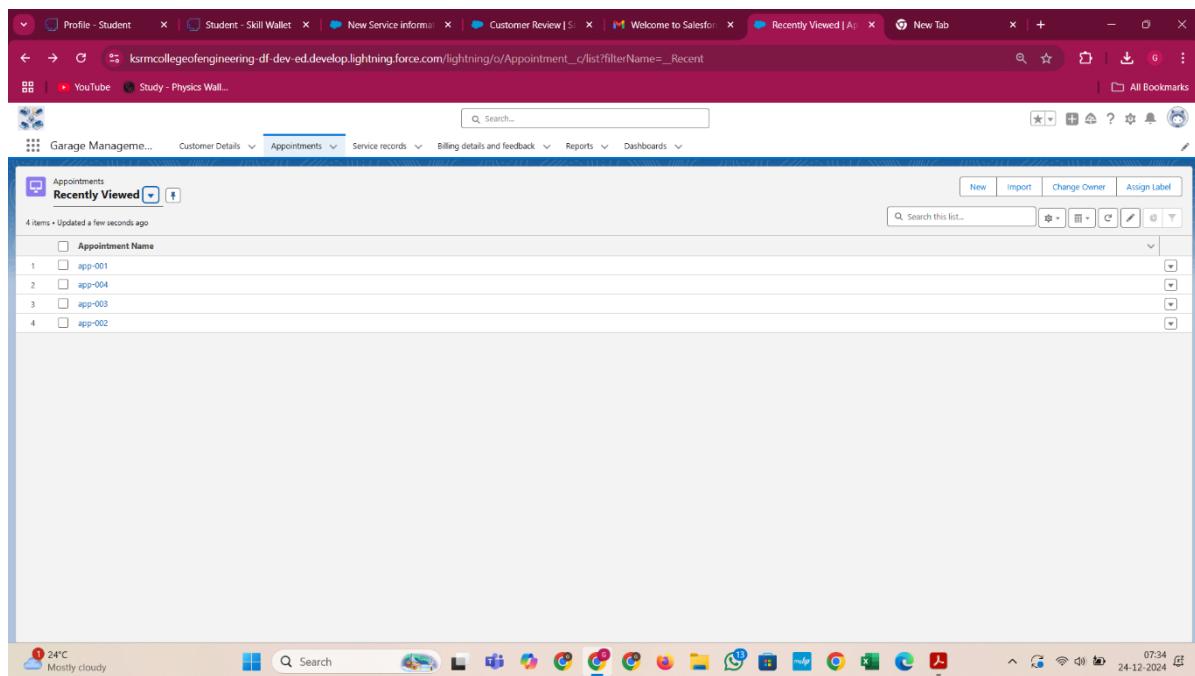
Created By: SHAIK MOHAMMED GAFFAR ALI, 20/12/2024, 7:03 am

Last Modified By: SHAIK MOHAMMED GAFFAR ALI, 20/12/2024, 7:03 am



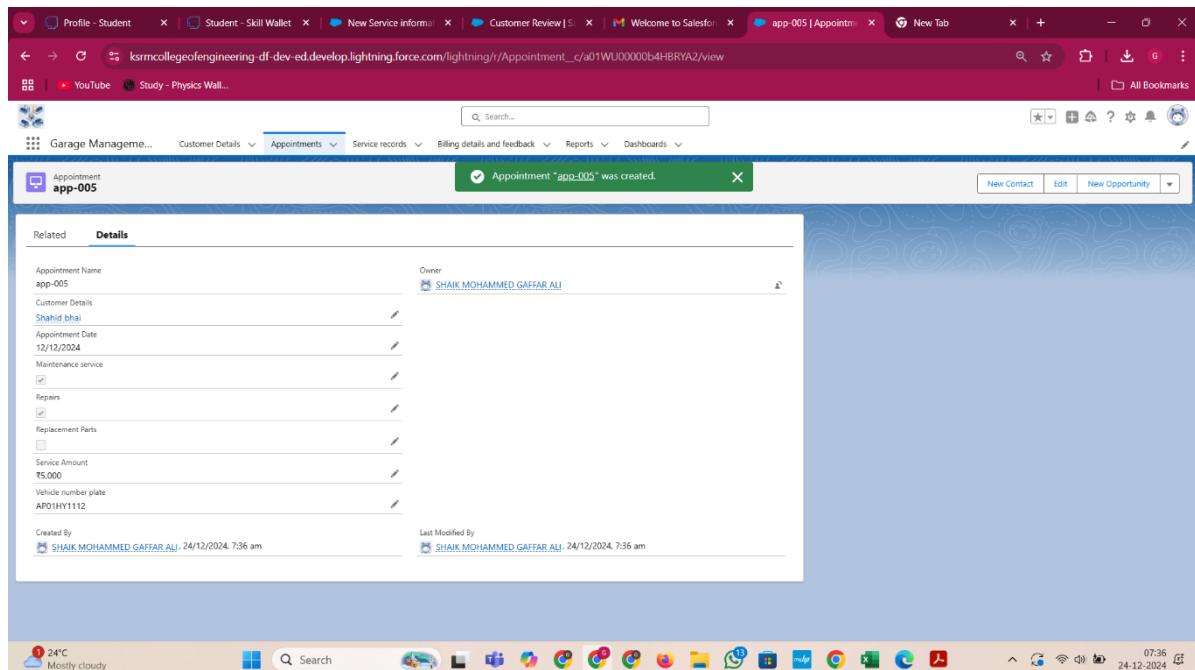
Step 4: UI Testing for Appointment Object

1. Click on the **Appointments** tab.
2. Enter the customer details from the **Customer Details** object, and make sure to input the **Appointment Date** as a past date to test the validation.
3. Test the **Vehicle Number Plate** validation by entering incorrect details and checking for error messages.
4. Select services and click **Save** to validate if the **Service Amount** is calculated correctly.



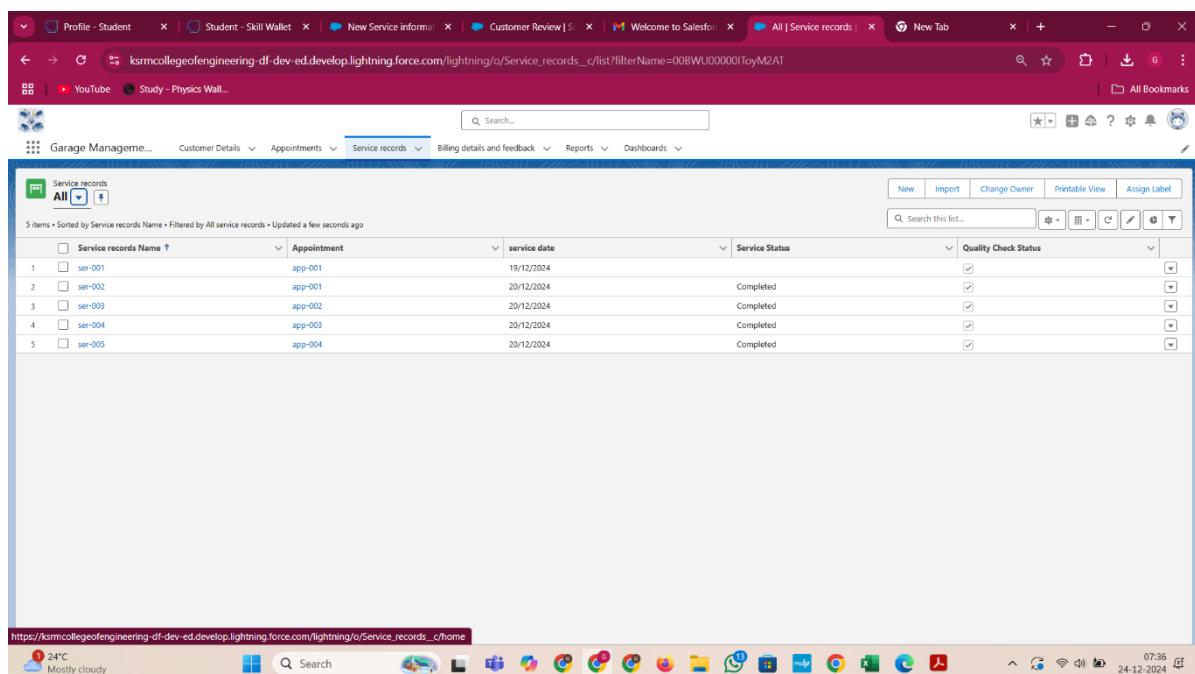
The screenshot shows a Salesforce Lightning App interface titled "Garage Management". The main title bar includes tabs like "Profile - Student", "Student - Skill Wallet", "New Service Informa...", "Customer Review | S...", "Welcome to Salesfor...", "app-002 | Appointm...", and "New Tab". The URL in the address bar is "krmcollegeofengineering-df-dev-ed.lightning.force.com/lightning/r/Appointment_c/a01WU00000ag131YAA/view". The page displays an appointment record for "Appointment app-002" owned by "SHAIK MOHAMMED GAFFAR ALI". The "Details" tab is selected, showing fields such as Appointment Name (app-002), Customer Details (Chan Basha), Appointment Date (15/12/2024), Maintenance service (checkbox checked), Repairs (checkbox checked), Replacement Parts (checkbox checked), Service Amount (\$0.00), and Vehicle number plate (AP01HY9956). The status bar at the bottom shows the date as 24-12-2024 and the time as 07:34.

The screenshot shows a Salesforce Lightning App interface titled "Garage Management". The main title bar includes tabs like "Profile - Student", "Student - Skill Wallet", "New Service Informa...", "Customer Review | S...", "Welcome to Salesfor...", "New Appointment |", and "New Tab". The URL in the address bar is "krmcollegeofengineering-df-dev-ed.lightning.force.com/lightning/o/Appointment_c/new?count=2&nooverride=1&useRecordTypeCheck=1&navigation.location=LIST_VIE...". A modal dialog box titled "New Appointment" is open, prompting for appointment details. The "Information" section requires fields include "Appointment Name" (shahid bhai), "Owner" (SHAIK MOHAMMED GAFFAR ALI), "Customer Details" (shahid bhai), "Appointment Date" (12/12/2024), "Maintenance service" (checkbox checked), "Repairs" (checkbox checked), "Replacement Parts" (checkbox checked), "Service Amount" (empty input field), and "Vehicle number plate" (AP01HY1112). The status bar at the bottom shows the date as 24-12-2024 and the time as 07:36.



Step 5: UI Testing for Service Records Object

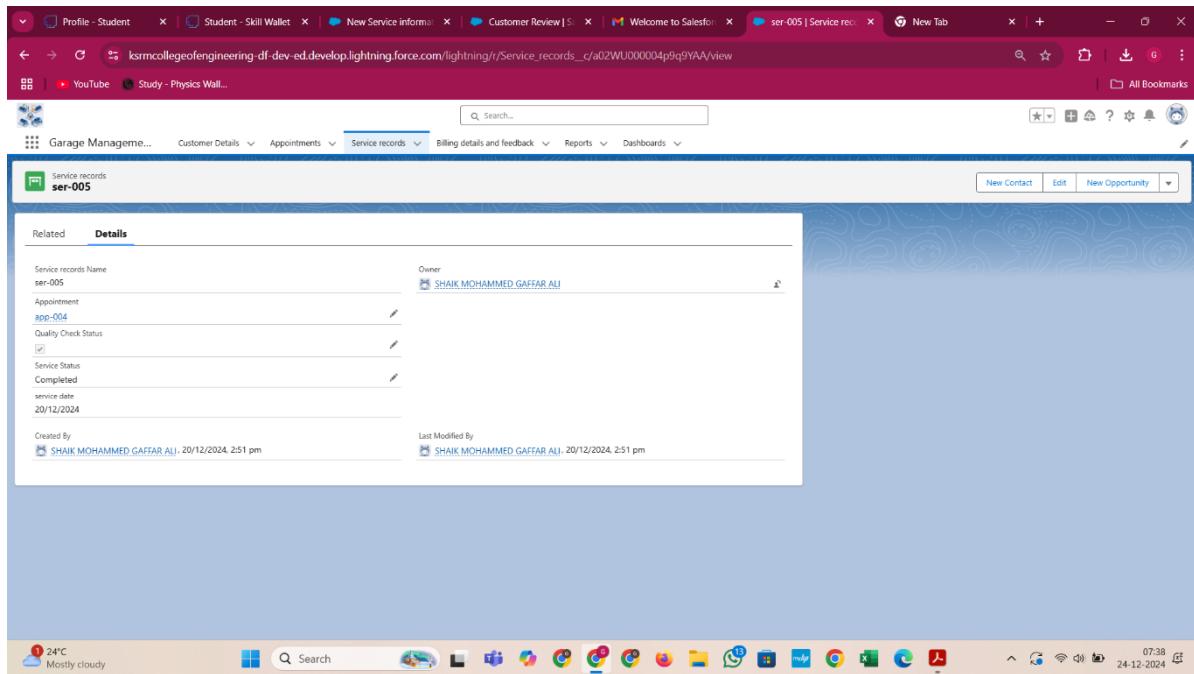
1. Navigate to the **Service Records** tab.
2. Create a new **Service Record** by entering the relevant **Appointment** details, which should be selected by default.
3. Click **Save**, then open the record to check if the **Quality Check Status** can be updated to **True**.
4. After saving the record, verify if the **Service Status** changes to **Completed** automatically.



This screenshot shows the Details tab of a Service Record in Salesforce. The record is named 'ser-003' and is owned by 'SHAIK MOHAMMED GAFFAR ALI'. It contains fields for Service records Name (ser-003), Appointment (app-002), Quality Check Status (in progress), Service Status (Completed), and service date (20/12/2024). The record was created by 'SHAIK MOHAMMED GAFFAR ALI' on 20/12/2024 at 12:58 pm and last modified by the same user at the same time. The browser interface includes a search bar, a ribbon menu with tabs like Garage Management, Customer Details, Appointments, Service records, Billing details and feedback, Reports, and Dashboards, and a toolbar with various icons.

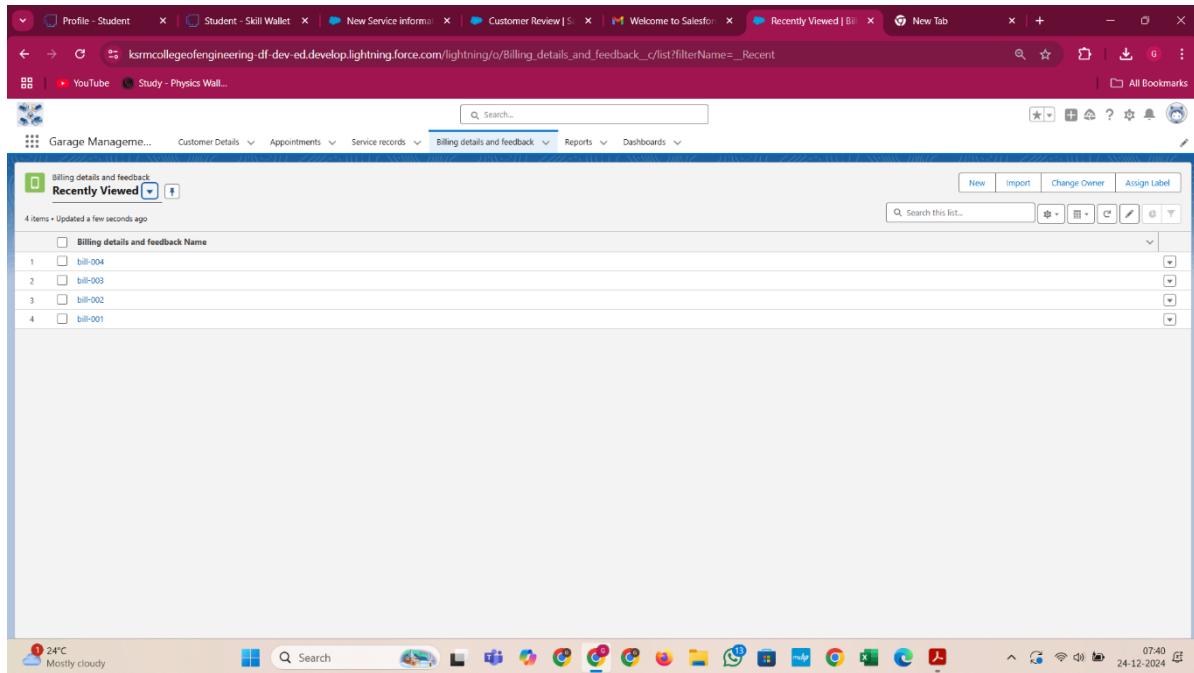
This screenshot shows the Details tab of a Service Record in Salesforce. The record is named 'ser-001' and is owned by 'SHAIK MOHAMMED GAFFAR ALI'. It contains fields for Service records Name (ser-001), Appointment (app-001), Quality Check Status (in progress), Service Status (Completed), and service date (19/12/2024). The record was created by 'SHAIK MOHAMMED GAFFAR ALI' on 20/12/2024 at 1:07 am and last modified by the same user at the same time. A tooltip for the Owner field shows additional information: Company Name (KSJUM COLLEGE OF ENGINE...) and Active (checkbox checked). The browser interface is identical to the first screenshot.

This screenshot shows the Details tab of a Service Record in Salesforce. The record is named 'ser-001' and is owned by 'SHAIK MOHAMMED GAFFAR ALI'. It contains fields for Service records Name (ser-001), Appointment (app-001), Quality Check Status (in progress), Service Status (Completed), and service date (19/12/2024). The record was created by 'SHAIK MOHAMMED GAFFAR ALI' on 20/12/2024 at 1:07 am and last modified by the same user at the same time. The browser interface is identical to the previous screenshots.



Step 6: UI Testing for Billing Details and Feedback Object

1. Open the **Billing Details and Feedback** tab.
2. Enter the customer's billing details, including the service ratings.
3. Click **Save**, then **New**, to ensure the system can handle multiple records.



New Billing details and feedback

Information

Billing details and feedback Name: bill-005

Service records: ser-005

Payment Paid: ₹5.00

Rating for service: 2

Payment Status: Completed

Owner: SHAIK MOHAMMED GAFFAR ALI

Billing details and feedback "bill-005" was created.

Billing details and feedback Name: bill-005

Service records: ser-005

Payment Paid: ₹5.00

Rating for service: 2

Payment Status: Completed

Owner: SHAIK MOHAMMED GAFFAR ALI

Recently Viewed

Billing details and feedback Name

- 1 bill-005
- 2 bill-004
- 3 bill-003
- 4 bill-002
- 5 bill-001

The screenshot shows a Salesforce Lightning interface for a 'Billing details and feedback' record named 'bill-001'. The record details are as follows:

- Billing details and feedback Name: bill-001
- Service records: ser-001
- Payment Paid: ₹2,000
- Rating for service: 4
- Payment Status: Completed

The record was created by SHAIK MOHAMMED GAFFAR ALI on 20/12/2024, 12:40 pm, and last modified by him on 20/12/2024, 12:41 pm.

Step 7: UI Testing for Reports

1. Click on the **Reports** tab.
2. Click **New Report** and select **Service Information** from the **Other Reports** category.
3. Drag and drop the fields **Customer Name**, **Appointment Date**, **Service Status**, and **Payment Paid** into the **Columns** section.
4. Add the **Rating for Service** and **Payment Status** to the **Rows** section.
5. Remove any unnecessary fields and add a **Line Chart** visualization.
6. Click **Save** and name the report **New Service Information Report**.
7. Select the appropriate folder and click **Save**.

The screenshot shows the Salesforce Reports page. The 'Recent' section displays a single report entry:

REPORTS	Report Name	Description	FOLDER	CREATED BY	CREATED ON	SUBSCRIBED
Recent	New Service information Report		Garage Management Folder	SHAIK MOHAMMED GAFFAR ALI	20/12/2024, 12:55 pm	

Profile - Student | Student - Skill Wallet | New Service informa... | Customer Review | S... | Welcome to Salesfor... | New Service informa... | New Tab

ksrmcollegeofengineering-df-dev-ed.develop.lightning.force.com/lightning//Report/00OWU000008W1z2AK/view?queryScope=userFolders

YouTube | Study - Physics Wall...

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

Report: Service Information
New Service information Report

Total Records: 5 Total Payment Paid: ₹24,000

PERFORMANCE

Details (5 Rows) Click an intersection in the table above to filter details.

Customer Name	Appointment Date	Service Status	Payment Paid
1 Mac	13/11/2024	Completed	₹5,000
2 Mac	13/11/2024	Completed	₹5,000
3 Abdullah	19/12/2024	-	₹2,000
4 Abdullah	19/12/2024	Completed	₹2,000
5 Ibn Abdullah	10/12/2024	Completed	₹10,000

Row Counts Detail Rows Grand Total Stacked Summaries

24°C Mostly cloudy 07:42 24-12-2024

Profile - Student | Student - Skill Wallet | New Service informa... | Customer Review | S... | Welcome to Salesfor... | Report Builder | Sale... | New Tab

ksrmcollegeofengineering-df-dev-ed.develop.lightning.force.com/lightning//Report/00OWU000008W1z2AK/edit?queryScope=userFolders

YouTube | Study - Physics Wall...

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

REPORT New Service information Report / Service information

Previewing a limited number of records. Run the report to see everything.

Rating for service ▾ Payment Status ▾ Completed Total

Groups	Rating for service	Payment Status	Completed	Total
GROUP ROWS	1	Sum of Payment Paid	₹5,000	₹5,000
		Record Count	1	1
GROUP COLUMNS	2	Sum of Payment Paid	₹5,000	₹5,000
		Record Count	1	1
Payment Status	4	Sum of Payment Paid	₹4,000	₹4,000
		Record Count	2	2
	5	Sum of Payment Paid	₹10,000	₹10,000
		Record Count	1	1
	Total	Sum of Payment Paid	₹24,000	₹24,000
		Record Count	5	5

PERFORMANCE

Details (5 Rows) Click an intersection in the table above to filter details.

Customer Name	Appointment Date	Service Status	Payment Paid
1 Mac	13/11/2024	Completed	₹5,000
2 Mac	13/11/2024	Completed	₹5,000
3 Abdullah	19/12/2024	-	₹2,000
4 Abdullah	19/12/2024	Completed	₹2,000
5 Ibn Abdullah	10/12/2024	Completed	₹10,000

Row Counts Detail Rows Grand Total Stacked Summaries

24°C Mostly cloudy 07:43 24-12-2024

Screenshot of a web browser showing the "Create Report" dialog in Salesforce. The dialog is titled "Create Report" and has a search bar for "Select a Report Type" with "service" typed in. On the left, there's a sidebar with categories like "Recently Used", "All", "Accounts & Contacts", etc. On the right, the "Service information" report type is selected, showing its details: "Custom Report Type", "Start Report" button, "Fields (42)", "Description" (Service Information), "Created By You" (New Service Information Report), "Created By Others" (No Reports Yet), and "Objects Used in Report Type" (Customer Details, Appointment, Service records). The background shows the Garage Management app interface.

Screenshot of a web browser showing the "New Service information Report" page in Salesforce. The title is "Report: Service Information New Service information Report". It displays a table with columns: Rating for service (1 to 5), Payment Status (Completed), Sum of Payment Paid, and Record Count. The table data is as follows:

Rating for service	Payment Status	Completed	Total
1	Sum of Payment Paid Record Count	₹5,000 1	₹5,000 1
2	Sum of Payment Paid Record Count	₹5,000 1	₹5,000 1
4	Sum of Payment Paid Record Count	₹4,000 2	₹4,000 2
5	Sum of Payment Paid Record Count	₹10,000 1	₹10,000 1
Total	Sum of Payment Paid Record Count	₹24,000 5	₹24,000 5

Below the table is a "Details (5 Rows)" section showing a list of customer payments:

Customer Name	Appointment Date	Service Status	Payment Paid
1 Mac	13/11/2024	Completed	₹5,000
2 Mac	13/11/2024	Completed	₹5,000
3 Abdullah	19/12/2024	-	₹2,000
4 Abdullah	19/12/2024	Completed	₹2,000
5 ibn Abdullah	10/12/2024	Completed	₹10,000
6			₹24,000

The bottom of the screen shows the Windows taskbar with various icons and the date/time: 24-12-2024 07:50.

Report: Service Information
New Service information Report

Rating 4

Rating for service	Payment Status	Completed	Total
□ 1	Sum of Payment Paid Record Count	₹5,000 1	₹5,000 1
□ 2	Sum of Payment Paid Record Count	₹5,000 1	₹5,000 1
□ 4	Sum of Payment Paid Record Count	₹4,000 2	₹4,000 2
□ 5	Sum of Payment Paid Record Count	₹10,000 1	₹10,000 1
Total	Sum of Payment Paid Record Count	₹24,000 6	₹24,000 6

Details (5 Rows) Click an intersection in the table above to filter details.

Customer Name	Appointment Date	Service Status	Payment Paid
1 Mac	13/11/2024	Completed	₹5,000
2 Mac	13/11/2024	Completed	₹5,000
3 Abdullah	19/12/2024	-	₹2,000
4 Abdullah	19/12/2024	Completed	₹2,000
5 ibn abdullah	10/12/2024	Completed	₹10,000
6			₹24,000

Row Counts Detail Rows Grand Total Stacked Summaries

24°C Mostly cloudy 08:04 24-12-2024

Report: Service Information
New Service Information Report

Total Records 5 Total Payment Paid ₹24,000

PERFORMANCE

Rating for service	Sum of Payment
1	5,000
2	5,000
4	4,000
5	10,000

Details (5 Rows) Click an intersection in the table above to filter details.

Customer Name	Appointment Date	Service Status	Payment Paid
1 Mac	13/11/2024	Completed	₹5,000
2 Mac	13/11/2024	Completed	₹5,000
3 Abdullah	19/12/2024	-	₹2,000
4 Abdullah	19/12/2024	Completed	₹2,000
5 ibn abdullah	10/12/2024	Completed	₹10,000
6			₹24,000

Row Counts Detail Rows Grand Total Stacked Summaries

24°C Mostly cloudy 08:04 24-12-2024

Report: Service Information
New Service Information Report

Sum of Payment

Rating for service	Sum of Payment
1	5,000
2	5,000
4	4,000
5	10,000

Details (5 Rows) Click an intersection in the table above to filter details.

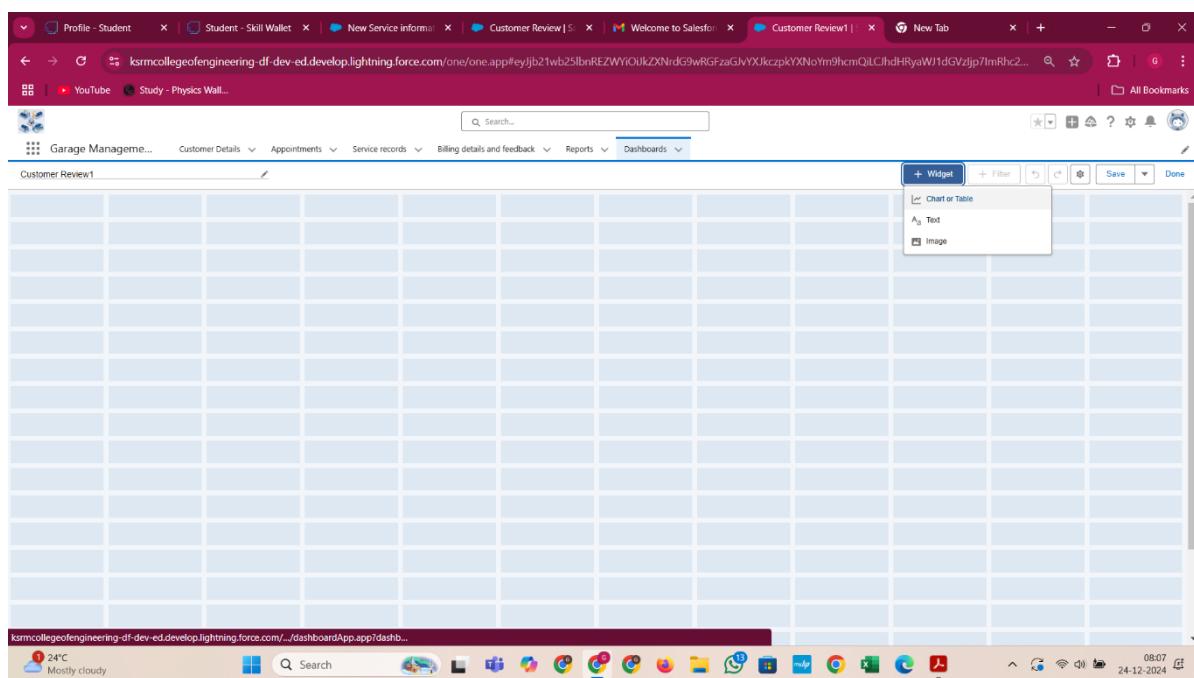
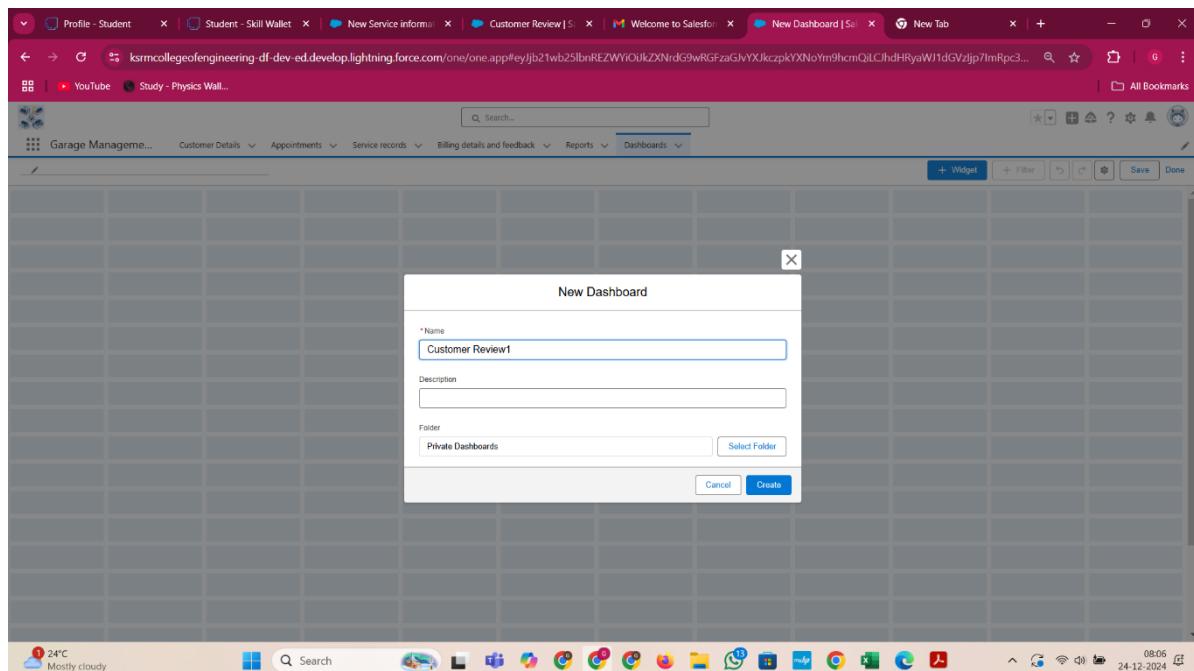
Customer Name	Appointment Date	Service Status	Payment Paid
1 Mac	13/11/2024	Completed	₹5,000
2 Mac	13/11/2024	Completed	₹5,000
3 Abdullah	19/12/2024	-	₹2,000
4 Abdullah	19/12/2024	Completed	₹2,000
5 ibn abdullah	10/12/2024	Completed	₹10,000
6			₹24,000

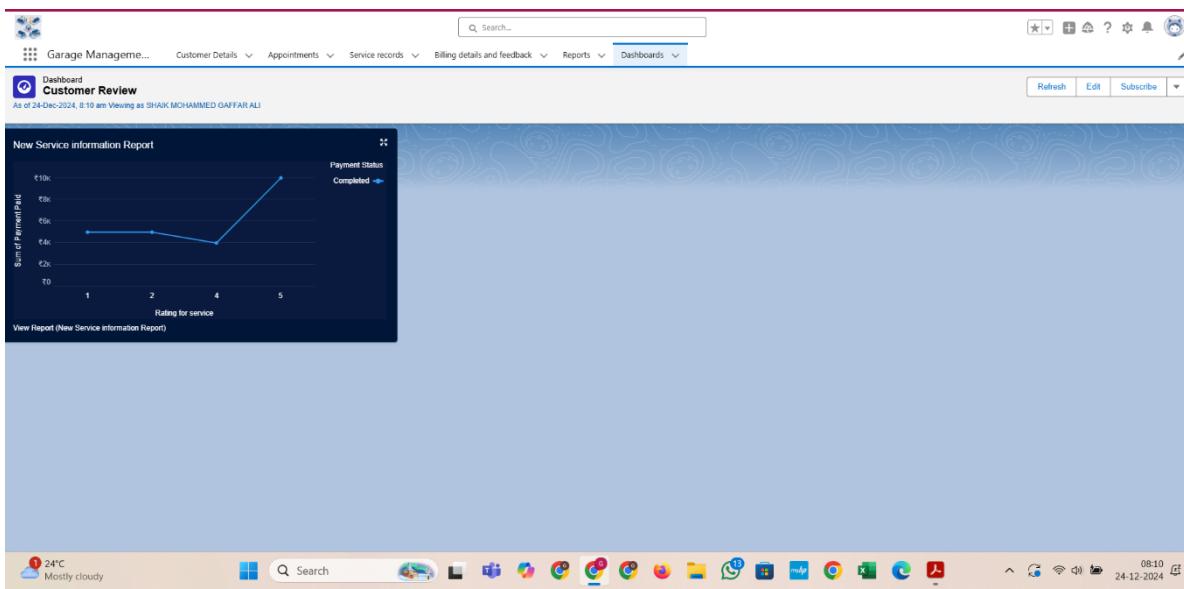
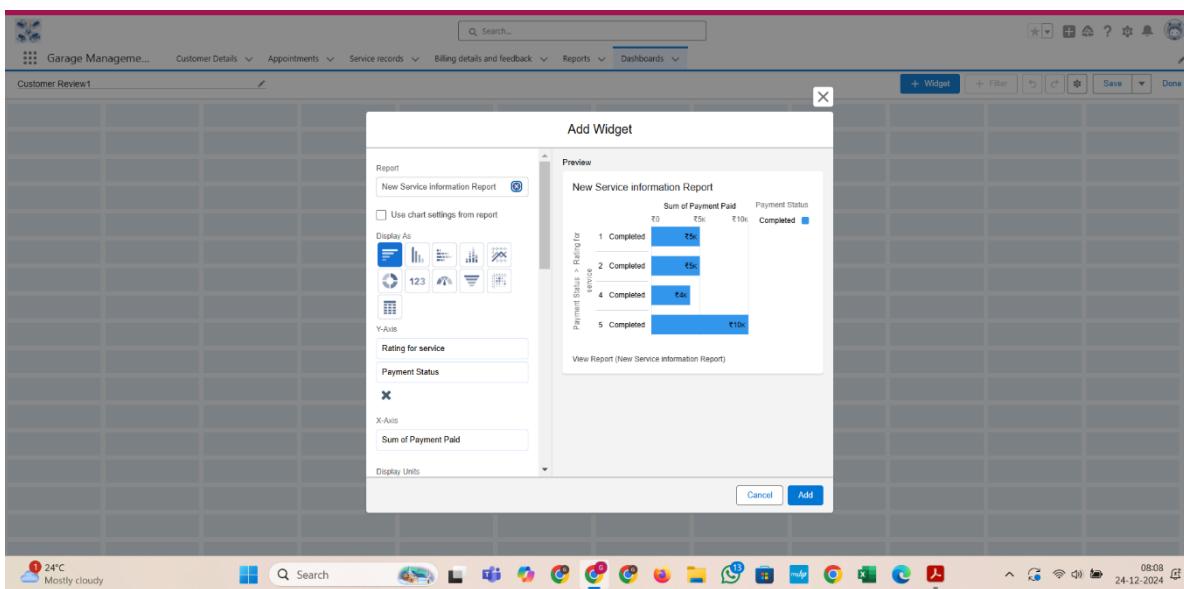
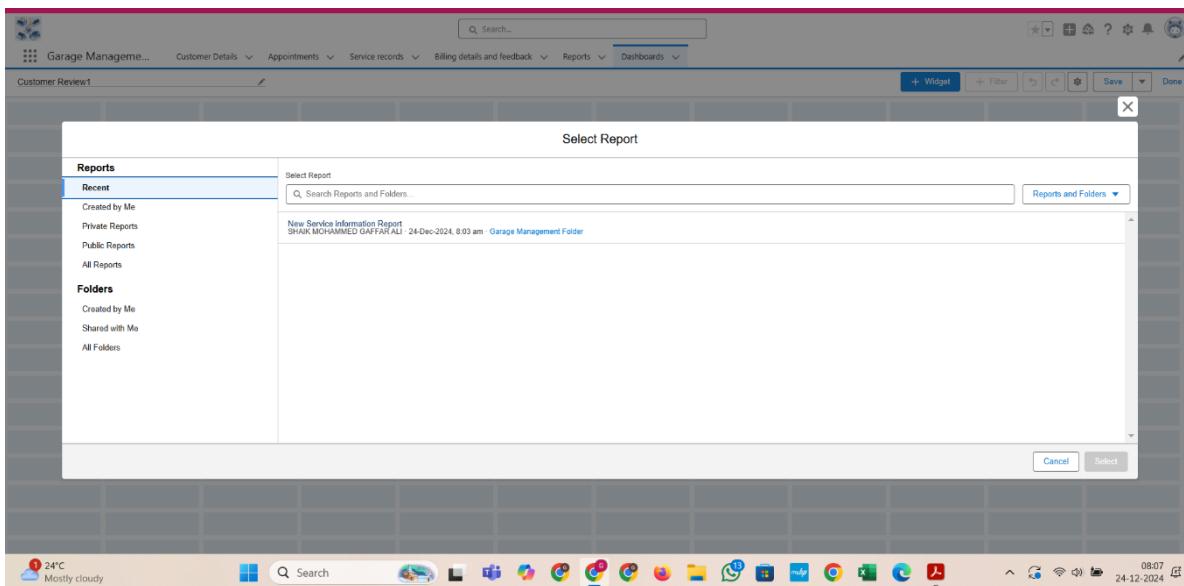
Row Counts Detail Rows Grand Total Stacked Summaries

24°C Mostly cloudy 08:05 24-12-2024

Step 8: UI Testing for Dashboards

1. Go to the **Dashboards** tab.
2. Click **New Dashboard**, give it a name (e.g., **Service Rating Dashboard**), and select the folder where it will be stored.
3. Click **Create** and add a **Component**.
4. Select the **Service Information Report** and choose the **Line Chart** visualization.
5. Customize the theme and click **Add**.
6. Click **Save**, then click **Done** to finalize the dashboard.
7. To set up email subscriptions, click on **Subscribe** at the top-right of the dashboard.
8. Set the frequency to **Weekly**, choose **Monday** as the preferred day, and click **Save**.





Screenshot of a web browser showing the "Edit Subscription" dialog box over a Salesforce dashboard.

The dashboard displays a line chart titled "New Service information Report" showing the "Sum of Payment Paid" versus "Rating for service". The chart shows data points at (1, ~4.5k), (2, ~4.2k), (4, ~3.8k), and (5, ~5.5k).

The "Edit Subscription" dialog box contains the following settings:

- Frequency:** Daily (selected)
- Days:** Sun, Mon, Tue, Wed, Thu, Fri, Sat
- Time:** 3:00 pm
- Recipients:** Me (checkbox checked)
- Send email to:** Me
- Buttons:** Unsubscribe, Cancel, Save

The status bar at the bottom shows: 24°C Mostly cloudy, Search, and system icons.

Screenshot of a web browser showing a confirmation message after saving a subscription.

The dashboard displays a line chart titled "New Service information Report" showing the "Sum of Payment Paid" versus "Rating for service". The chart shows data points at (1, ~4.5k), (2, ~4.2k), (4, ~3.8k), and (5, ~5.5k).

A green notification bar at the top right says: "You started a dashboard subscription." with a checkmark icon.

The status bar at the bottom shows: 24°C Mostly cloudy, Search, and system icons.

Key Scenarios Addressed by Salesforce in the Implementation of the Project

Salesforce provides a robust platform to address several business scenarios effectively in the context of a Garage Management System (GMS). These scenarios help streamline operations, improve customer satisfaction, automate processes, and offer meaningful insights. Below are the key scenarios that Salesforce handles during the implementation of this project:

1. Customer Information Management:

Salesforce simplifies the collection, storage, and management of customer details, ensuring that the information is easily accessible for service management. It handles:

- **Customer Name**
- **Contact Information (Phone Number, Email, etc.)**
- **Vehicle Details**
- **Preferred Service Details**
- **Service History**

With Salesforce, customer details can be stored in custom objects like **Customer Details**, offering a 360-degree view of the customer, making it easy to follow up, track service history, and offer personalized services.

2. Appointment Scheduling and Management:

Scheduling and managing customer appointments is automated through Salesforce, allowing customers to book their services and ensuring seamless management on the backend. Key features include:

- **Service Appointment Creation**
- **Real-Time Availability Tracking**
- **Automated Notifications** (e.g., Appointment Confirmation, Reminders)
- **Service Types** (e.g., Routine Maintenance, Repairs, Replacement Parts)
- **Appointment Confirmation and Rescheduling**

Salesforce provides tools such as **Calendar Integration** and **Automated Email Notifications**, ensuring a smooth appointment process for both customers and service staff.

3. Service Records Tracking:

Managing service records in Salesforce ensures that the entire service process is recorded and tracked efficiently. This scenario involves:

- **Service Record Creation**
- **Vehicle Details at Check-In**

- **Service Assignment to Technicians**
- **Real-Time Tracking of Service Progress**
- **Status Updates** (e.g., "Completed," "In Progress," "Pending")

By leveraging **Custom Forms** and **Automated Task Assignments**, Salesforce enables efficient service management and helps technicians and managers monitor service statuses seamlessly.

4. Billing and Payment Management:

Salesforce plays a crucial role in tracking payments, issuing invoices, and managing customer feedback. It includes:

- **Payment Processing Integration** with various gateways
- **Tracking Payment Status** (e.g., Pending, Paid, Overdue)
- **Issuing Invoices and Receipts**
- **Collecting Service Feedback and Ratings**
- **Customer Communication for Outstanding Payments**

With the help of **Custom Billing Objects** and integration capabilities, Salesforce streamlines the billing process, sending automatic payment reminders and collecting feedback post-service.

5. Analytics and Reporting:

Salesforce provides powerful reporting features that allow businesses to analyze data, track performance, and make informed decisions. Key aspects include:

- **Tabular Reports** for simple, row-based data
- **Summary Reports** for grouped data analysis
- **Matrix Reports** for cross-tabulation of data
- **Joined Reports** to combine data from different sources

Through Salesforce's **Report Builder** and the ability to apply complex filters and grouping, businesses can create reports that provide actionable insights into various aspects of the Garage Management System, including customer engagement, service performance, and revenue analysis.

6. Dashboard Visualizations:

Salesforce Dashboards allow real-time visualization of business data, providing insights into operational efficiency and trends. They include:

- **Real-Time Business Insights**
- **Trend Identification** (e.g., Service demand, Payment trends)
- **Impact Measurement** (e.g., Revenue Generation, Customer Satisfaction)
- **Interactive Dashboards** with Drill-Down Capability.

Conclusion

The implementation of the **Garage Management System (GMS)** using **Salesforce** has significantly transformed how the garage manages its operations, offering a more streamlined and efficient solution for handling customer data, service appointments, billing, and reporting. By leveraging Salesforce's powerful features, the system has automated and simplified key processes, making the garage more responsive to customer needs and improving overall operational efficiency.

Through **centralized customer data management**, Salesforce ensures that customer details are easily stored, retrieved, and updated, enabling a more personalized service. The system provides a **seamless appointment scheduling process**, integrating real-time tracking of service appointments, mechanic availability, and service requirements. This has resulted in fewer scheduling conflicts and a better customer experience with automated reminders and updates.

Service tracking is now more transparent, with real-time updates shared with customers regarding the status of their vehicle. This feature builds trust and improves customer satisfaction by keeping them informed throughout the service process. Additionally, the **automated billing and payment** system has minimized errors and optimized payment tracking, ensuring a smooth transaction experience for customers.

The system's **reporting and analytics capabilities** have empowered decision-makers to analyze performance, track service metrics, and uncover insights that drive business improvements. The use of **customizable reports** and **dynamic dashboards** ensures that key business data is presented in an easily digestible format, aiding quick and informed decision-making.

In summary, the **Salesforce-powered GMS** has optimized operational workflows, enhanced customer engagement, and delivered significant value by automating routine tasks, improving data management, and offering real-time visibility into garage activities. This implementation has not only improved day-to-day operations but also paved the way for future growth by providing a scalable and adaptable platform that can evolve with changing business needs and customer demands. The successful deployment of this system has set the foundation for ongoing business improvements and long-term success for the garage.

Key Achievements Include:

1. Efficient Customer Data Management

The system provides a comprehensive view of customer information, allowing for easy access and tracking of customer interactions, service history, and feedback.

2. Automated Appointment Scheduling and Management

With automated reminders, availability tracking, and rescheduling options, the scheduling process has been significantly optimized, reducing errors and improving service delivery.

3. Real-Time Service Tracking

The system offers real-time updates on service progress, from check-in to completion, ensuring that customers are informed and technicians are efficiently managing their tasks.

4. Seamless Billing and Payment Management

Integration with payment gateways and automated invoicing has streamlined the billing process, while tracking payment status and feedback collection has improved customer satisfaction and operational transparency.

5. Data-Driven Insights through Analytics and Reporting

Customizable reports and dashboards provide valuable insights into service performance, customer engagement, and financial metrics, enabling data-driven decision-making.

6. Improved Dashboard Visualizations

The use of real-time dashboards allows managers to monitor key performance indicators (KPIs), track trends, and assess the impact of various business activities.

Overall, the Garage Management System implemented using Salesforce has brought significant improvements to operational efficiency, customer satisfaction, and data management. With Salesforce's flexible and scalable platform, the garage can easily adapt to future changes, ensuring continued growth and success in managing its operations.

STUDENT DETAILS

STUDENT NAME: SHAIK MOHAMMED GAFFAR ALI

EMAIL: shaikgaffar1519@gmail.com

MOBILE NUMBER: 7731085768

COURSE: B. TECH

BRANCH: ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

STUDENT ID: 219Y1A3951

YEAR: 4TH YEAR

COLLEGE: K.S.R.M. COLLEGE OF ENGINEERING, KADAPA.