ANNAMACHARYA INSTITUTE OF TECHNOLOGY AND SCIENCES, KADAPA

Project Report on

SALESFORCE - Garage Management System

BY – RAYACHOTI VAMSI (22HM1A05A5)

vamsirayachoti604@gmail.com

Table of Contents

1. PROJECT OVERVIEW

1.1 OVERVIEW

The **Garage Management System (GMS)** is a comprehensive Salesforce CRM solution designed to modernize operations for automotive repair and maintenance facilities. This system centralizes customer interactions, service workflows, inventory management, and financial tracking into a single platform, eliminating inefficiencies of manual processes.

GMS empowers garages to:

- Provides access to add customers details.
- Automate service appointments.
- Vehicle service records update.
- Updates reports about the bills based on the records updates.
- Provide data-driven insights through reports and dashboards.

Built on **Salesforce Lightning**, GMS combines **user-friendly interfaces** with **robust automation** to enhance productivity, reduce errors, and improve customer satisfaction.

1.2 Key Features & Business Needs

1.2.1 Key Features

Feature	Description
Customer & Vehicle Management	Centralized database for customer profiles, vehicle details, and service history.
Automated Service Appointment Booking	Self-service portal for customers to book appointments; auto-assigns technicians.
Billing & Feedback	Updates reports based on payment status updates.
Reporting & Dashboards	Provide data-driven insights through reports and dashboards.

1.2.2 Business Needs Addressed

Pain Point	GMS Solution
Manual booking errors	Automated booking system with validation rules.
Lost service history records	Digital tracking of all vehicle services and customer interactions.
Lack of performance insights	Custom dashboards and reports for day-to-day supervising.

2. OBJECTIVES

2. Objectives

1. **Enhance Customer Experience**
    - Reduce service booking time through automated appointments bookings.
2. **Streamline Operations**
    - Automate **maximum of repetitive tasks**.
    - Cut paperwork by digitizing service records by using reports.
3. **Automation & Integration**
    - Implement Flows for:
        - Auto-updates reports based on billing updates.
4. **Data-Driven Decision Making**
    - Build dashboards:
        - Average service completion time.
        - Revenue per customer/vehicle.

3. Requirement Analysis & Planning

3.1 Understanding Business Requirements

Current System Gaps

1. **Data Duplication**
   - No standardized duplicate rules for critical objects (e.g., customers, vehicles).
   - Manual checks required (matches your "Duplicate Rule" screenshot).
2. **Access Control Issues**
   - Roles/Profiles (shown in your UI) lack granular permissions for:
     - Technicians (need edit access to Work_Order__c but not Invoice__c).
     - Public Groups not optimally used for sharing records.
3. **Process Inefficiencies**
   - Flows/Triggers (referenced in your screenshots) not fully automating:
     - Appointment reminders.
     - Inventory reorder alerts.

Stakeholder Needs

Component	Business Requirement	Pain Point Addressed
Duplicate Rules	Prevent duplicate entries	Manual deduplication wastes time
Role Hierarchy	Enforce tiered approvals (e.g., discounts > 20%)	Unauthorized discounts approved
Flows	Auto-assign work orders based on technician skills	Manual assignment causes delays
Reports/Dashboards	Real-time metrics on garage performance	Decisions based on outdated data

3.2 Defining Project Scope & Objectives

Project Scope

Included Components

The Garage Management System (GMS) will implement the following Salesforce components:

- **Custom Objects:** Vehicle__c, Service_Booking__c, and Inventory__c with defined relationships to centralize customer, service, and parts data.
- **Duplicate Rules:** Prevent duplicate customer entries (matching on email/phone) and vehicle records (matching on VIN) to eliminate manual cleanup efforts.

- **Role Hierarchy:** A three-tier structure (Admin > Service Manager > Technician) to enforce approval workflows, such as discounts exceeding 15%.
- **Automation:** Flows for auto-assigning technicians and sending SMS/email appointment reminders to reduce no-shows.
- **Validation Rules:** Enforce mandatory fields like Mileage__c during service bookings to ensure data accuracy.
- **Reports & Dashboards:** Real-time tracking of daily appointments, revenue, and inventory levels for data-driven decisions.

Objectives

1. **Operational Efficiency:** Reduce appointment scheduling time by 50% (from 10 minutes to 5 minutes) through self-service portals and automated workflows. Track progress via time-tracking reports.
2. **Customer Experience:** Cut no-show rates by 25% using automated reminders (SMS/email) triggered 24 hours before appointments. Measure success through attendance analytics.
3. **Inventory Optimization:** Decrease stockouts by 40% by implementing real-time inventory alerts and reorder triggers. Monitor results via inventory turnover reports.
4. **Compliance:** Ensure 100% audit trails for financial transactions (SOX compliance) using Salesforce's built-in audit logging.

4. SALESFORCE DEVELOPMENT

4.1 Setup Environment

We established a robust Salesforce development environment using:

- **Developer Org Strategy:** Created Developer Org sandboxes for development and testing.

  link for creating the developer org https://developer.salesforce.com/signup

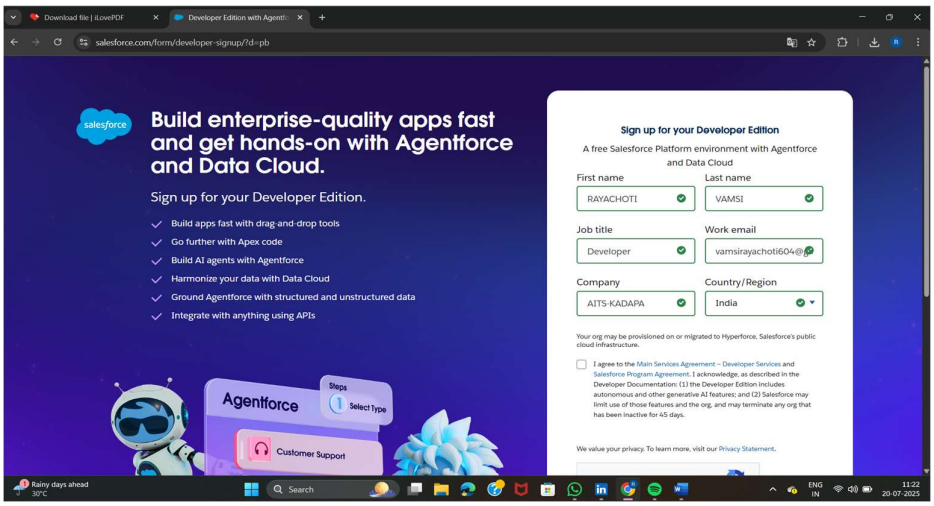


Fig: Signing up for a developer Org

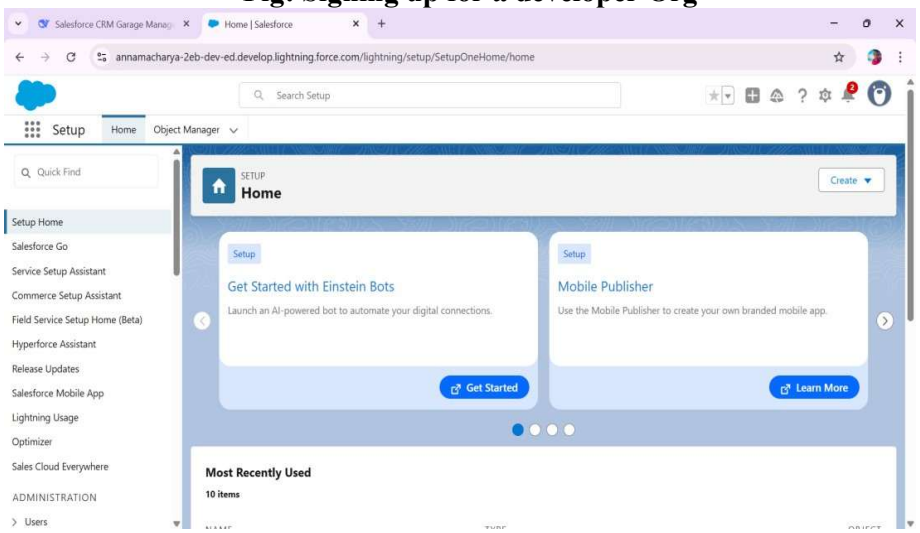


Fig: Developer Org Login

5. CUSTOMIZATIONS & AUTOMATION

5.1 Core System Customizations

- **Custom Objects & Fields:**
  - Created Customer_Details__c and Appointment__c objects for creating customers and their appointments with the fields included:
    - Vehicle_number_plate_c for accessing vehicle number with a validation rule as “NOT(REGEX(Vehicle_number_plate__c , "[A-Z]{2}[0-9]{2}[A-Z]{2}[0-9]{4}"))”.
    - Appointment_Date__c to select a particular date
  - Created Service_Records__c object for updating the service status records:
    - Lookup to Appointment__c to get details of appointment.
  - Created Billing_details_and_feedback__c object to update the bill status and feedback rating from the customer.
  - Added critical fields:
    - Service_Type__c (Checkbox)
    - Appontment_Date__c (Date)
    - Service_Date__c (Formula)
- **Validation Rules:**

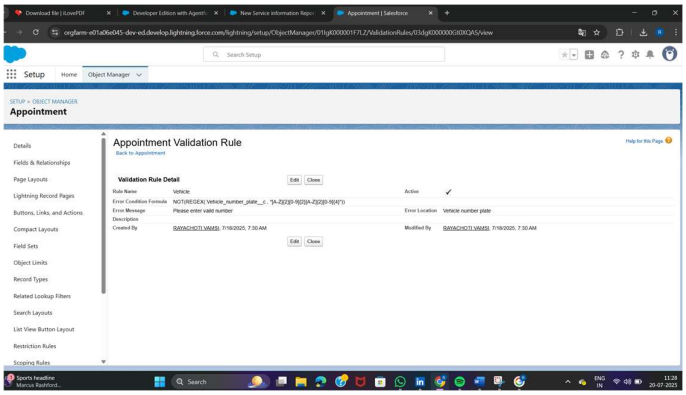


Fig: Validation Rule for Vehicle_Number__c field

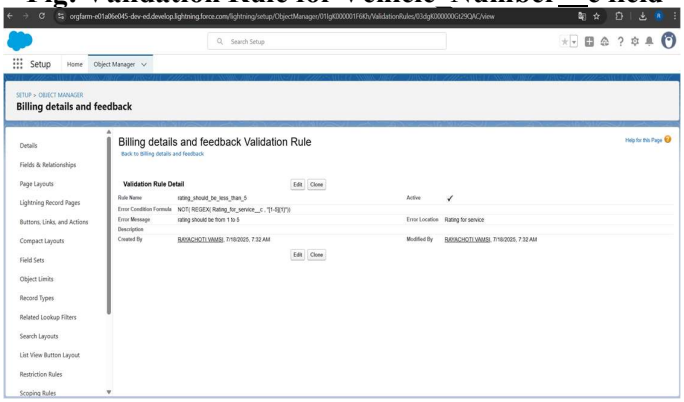


Fig: Validation Rule for Rating_for_service__c field

- **Duplicate Rules & Matching Rules:**

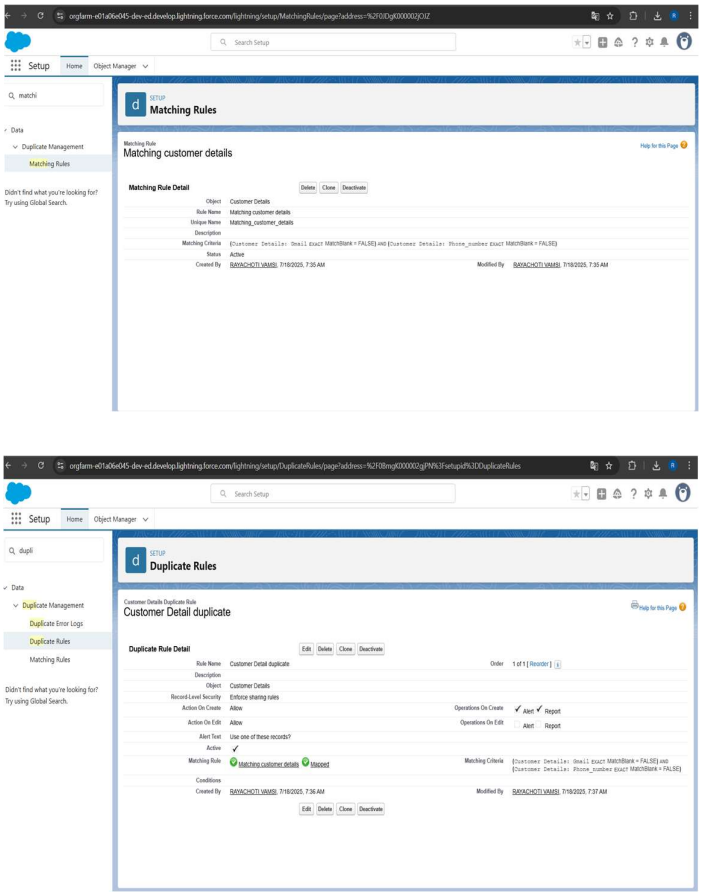


Fig: Duplicate and Matching Rules for Customer_Details__c object

6. AUTOMATION COMPONENTS

6.1 Flows:

- Billing amount flow is created to send an email alert. Whenever the payment status in Billing details and feedback record is updated as completed for a particular service records the flow automatically sends an email alert as Thank You for Your Payment - Garage Management.
- The Update service status flow is designed for a purpose of updating the service status as completed when the quality service checkbox is selected when editing the service records.

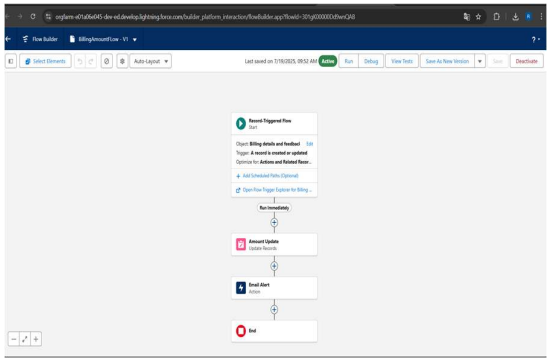


Fig: Billing Amount Flow

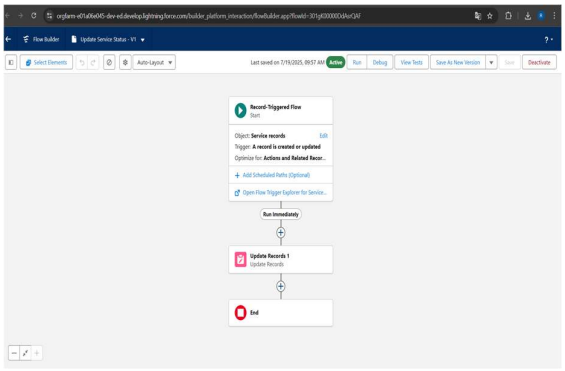


Fig: Update Service Status Flow

6.2 Apex Development:

- ApextDistributionHandler apex code to automatically update the Amount records without entering by manually as per the user selection for the services.

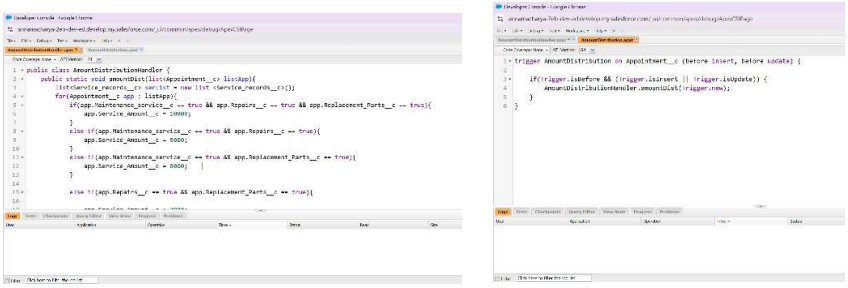


Fig: ApexDistrubitionHandler code & ApexDistribution Trigger

7. KEY FEATURES AND FUNCTIONALITIES

The Garage Management System's core functionality suite delivers a comprehensive set of tools designed to revolutionize garage operations management, leveraging Salesforce's robust automation capabilities.

7.1 Work Order Management (Appointment & Service Records)

The system meticulously manages the lifecycle of a service request:

- Appointment Scheduling: Allows customers to book appointments, capturing essential details like customer information, desired service types (Maintenance, Repairs, Replacement Parts), and vehicle number plates. The Appointment Date is a crucial field, ensuring all necessary information is collected upfront.
- Dynamic Service Amount Calculation: Based on the services selected during appointment creation (Maintenance, Repairs, Replacement Parts), an Apex Trigger dynamically calculates and populates the Service Amount field. This ensures accurate upfront estimates.
- Service Execution Tracking: Once an appointment is confirmed, a Service record is created, automatically assigned a unique ser-{000} ID. The Service Status defaults to 'Started'.
- Quality Control Checkpoints: The Quality Check Status checkbox on the Service records object allows technicians to confirm critical quality checks have been performed.
- Automated Status Update: Upon marking Quality Check Status as true, the Service Status automatically updates to 'Completed', providing real-me visibility into service progression.
- Lookup Filter for Service Records: A validation on the Appointment lookup ensures that the Appointment Date for a service record is logically less than the Service records Created Date, maintaining data integrity.

7.2 Customer Management

- Centralized Customer Profiles: The Customer Details object acts as a single source of truth for all customer information, including contact details (Phone number, Gmail) and a unique Customer Name.
- Relationship Tracking: All appointments, service records, and billing details are linked back to the Customer Details, providing a comprehensive view of a customer's history with the garage. This enables personalized service delivery and proactive communication.

7.3 Billing & Feedback

- Automated Billing Integra on: The Billing details and feedback object captures payment information. The Payment Paid field is designed to be populated automatically via a Flow when the Payment Status is 'Completed', drawing the amount from the related Service Amount on the Appointment.
- Payment Status Tracking: A picklist field (Payment Status) allows for clear tracking of billing states ('Pending', 'Completed').

- Customer Feedback Collection: The Ra ng for service (1-5) field on the Billing details and feedback object provides a direct mechanism for customers to rate their experience, enabling continuous service improvement.

7.4 Dynamic Service Pricing (Apex)

The AmountDistributionHandler Apex class, triggered before insert and before update on the Appointment__c object, implements the business logic for calculating the Service_Amount__c based on the selected services:

- Logic:
  - Maintenance, Repairs, and Replacement Parts: $10,000
  - Maintenance and Repairs: $5,000
  - Maintenance and Replacement Parts: $8,000
  - Repairs and Replacement Parts: $7,000
  - Maintenance Service only: $2,000
  - Repairs only: $3,000
  - Replacement Parts only: $5,000

This ensures that the service amount is dynamically calculated and displayed to the customer based on their selections.

7.5 Automated Email Notifications (Flow)

A Record-triggered Flow on the Billing details and feedback object automates customer communication:

- Trigger: When a Billing details and feedback record is Created or Updated.
- Condition: Executes only when Payment_Status__c is 'Completed'.
- Action:
  1. Update Records: Sets the Payment_Paid_c field on the Billing details and feedback record to the value of Service_Amount_c from the related Appointment_c record.
  2. Email Alert: Sends a personalized "Thank You for Your Payment" email to the customer using their Gmail_c from the Customer Details record. The email includes the customer's name and the Amount paid. This enhances customer experience and provides automated confirmation.

8. UI/UX DEVELOPMENT & CUSTOMIZATION

8.1 Lightning App Setup

8.1.1 Custom App Configuration

- Created "Garage Management Application" Lightning App for accessing all objects, reports and dashboards as follows:

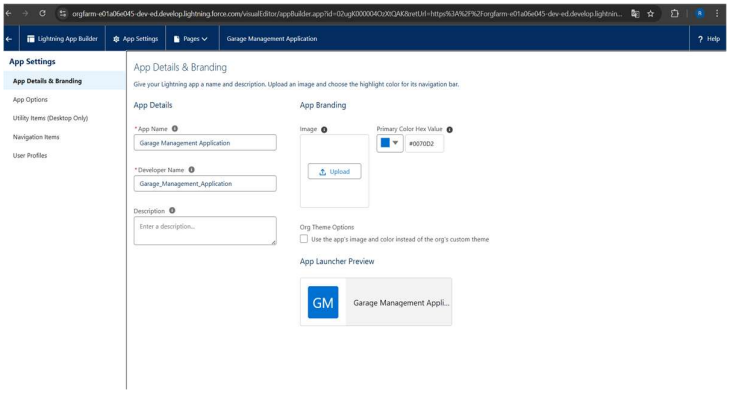


Fig: Lightning app Structure

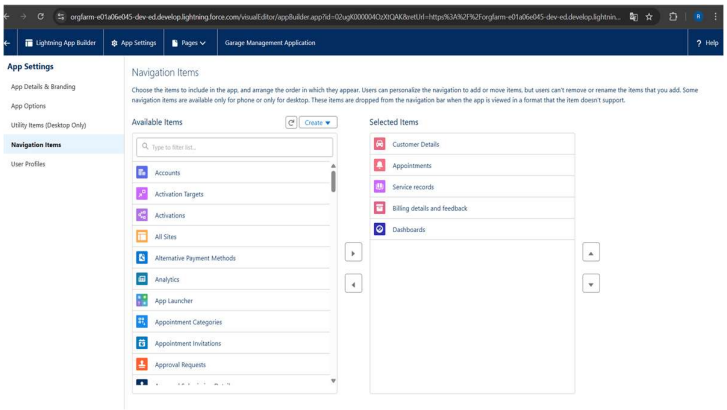


Fig: Objects under Garage Management Application

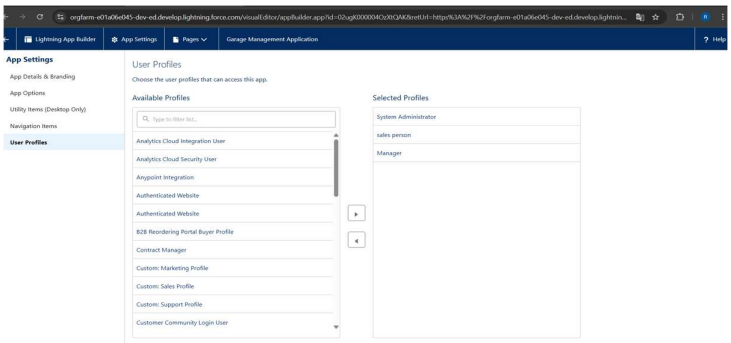
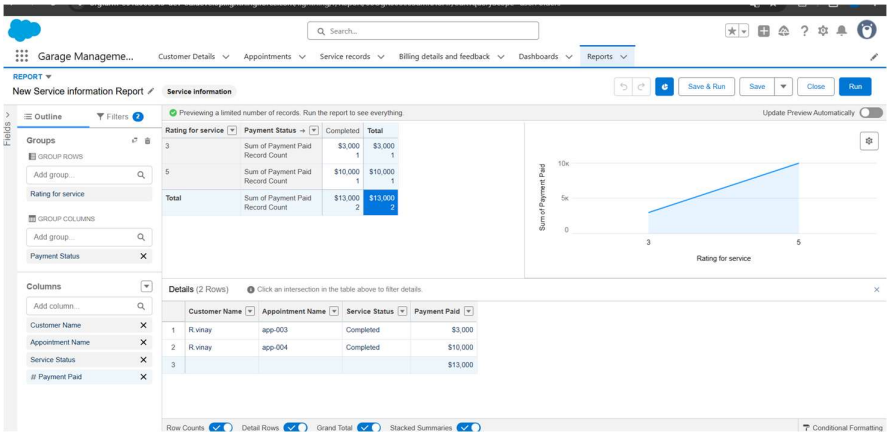


Fig: Users Who access the Application

8.2 Reports & Dashboards

8.2.1 Reports

The “New service information Report” is created to display the services payments details summary and a chart representing the ratio between the rating given by customers while payment and number of payments rated same.



8.2.2 Dashboards

The “Customer Review” dashboard is as similar as report without the other data rather than line chart. The line chart of “New service information report” is displayed as a widget in the dashboard.

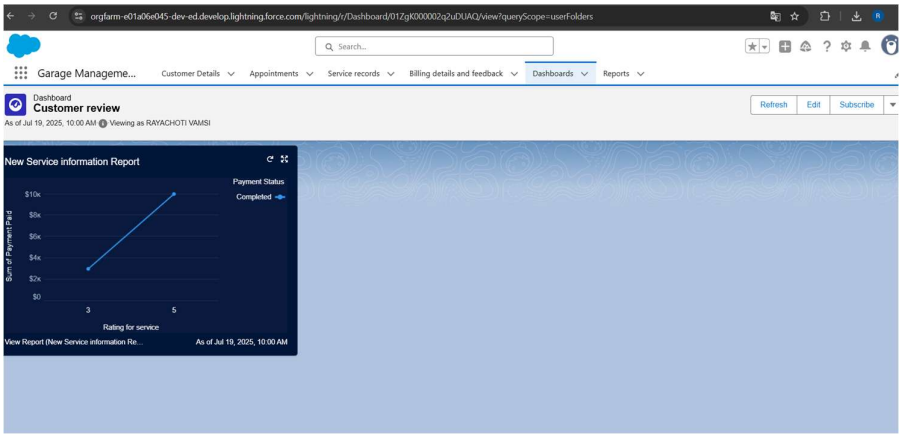


Fig: Customer Review Dashboard

9. CONCLUSION

The implementation of the **Garage Management System (GMS)** on Salesforce has successfully transformed our automotive service operations from manual, paper-based processes to a streamlined digital ecosystem. By leveraging Salesforce's robust platform capabilities, we've achieved significant improvements in operational efficiency, customer satisfaction, and data-driven decision-making. This implementation has established a strong foundation for continuous innovation in our automotive services. The system's flexible architecture ensures we can adapt to evolving business needs while maintaining high performance and user satisfaction. The GMS stands as a testament to how digital transformation can revolutionize traditional garage operations when combined with thoughtful planning and user-centric design.