# ANNAMACHARYA INSTITUTE OF TECHNOLOGY AND SCIENCES, KADAPA

|  |
| --- |
|  |
|  |  |



## Project Report on

**SALESFORCE - Garage Management system BY – POLA SRUTHI (22HM1A0447)**

[sruthip0222@gmail.com](mailto:sruthip0222@gmail.com)

## Table of Contents

1. **Project Overview**
2. **Project Objectives**
3. **Phase 1: Requirement Analysis & Planning**
   * Understanding Business Requirements
   * Project Scope & Objectives
   * Data & Security Model Design
4. **Phase 2: Salesforce Backend Development & Configurations**
   * Environment Setup
   * Custom Objects Created
   * Fields Configurations
   * Validations Rules
   * Automation
   * Apex trigger
5. **Phase 3: UI/UX Development & Customization**
   * Lightning App Creation
   * Test Setup
   * Page Layouts & Dynamic Forms
   * Reports
   * Dashboards
6. **Phase 4: Data Migration, Testing & Security**
   * Data Migration
   * Matching & Duplicate Rules
   * Field History Tracking
   * Security Tracking
   * Testing & Q&A
7. **Phase 5: Deployment, Documentation & Maintenance**
   * Deployment Strategy
   * System Maintenance
   * Documentation & Training
8. **Conclusion**
9. **Future Enhancements**

**Project Overview**

The **Garage Management System (GMS)** is a comprehensive, cloud-based CRM solution developed using the Salesforce platform, aimed at transforming and modernizing the traditional operations of automobile repair and service centers. It serves as a centralized system that automates and simplifies critical workflows such as customer registration, appointment scheduling, service management, billing, and customer feedback.

The system is designed to support garage managers, front-desk staff, and service personnel in managing the complete lifecycle of a customer's interaction with the garage — from the initial service inquiry and booking to the completion of the service and final payment. With Salesforce’s robust architecture, GMS leverages cloud scalability, built-in security, custom automation, and real-time data analysis to deliver a seamless user experience.

Salesforce’s point-and-click development tools, including Flows, Process Builders, and Apex Triggers, allow for powerful business logic implementation with minimal code. The platform’s advanced reporting and dashboard features help managers track key performance indicators (KPIs), analyze customer satisfaction, and monitor operational efficiency on a day-to-day basis.

**Key business challenges addressed by GMS include:**

* **Efficient appointment scheduling and service tracking:** Staff can schedule, manage, and monitor appointments and service records for each customer in real-time, reducing wait times and improving resource allocation.
* **Automated service cost calculations:** Based on selected services (e.g., maintenance, repairs, or part replacements), the system automatically calculates service costs through Apex logic, ensuring billing accuracy and transparency.
* **Real-time performance insights:** Custom dashboards and dynamic reports allow managers to visualize service trends, revenue generation, and customer feedback, empowering data-driven decision-making.
* **Enhanced customer engagement:** Automated email notifications and follow-ups, including personalized thank-you messages post-payment, help in building long-term customer relationships and encouraging return visits.

**Project Objectives**

The Garage Management System was developed with the aim of modernizing and automating the daily operations of an automobile service center. Below are the elaborated objectives that guided the development of this Salesforce-based solution:

* **Digitize Service Operations:** Eliminate manual paperwork by digitizing all customer, appointment, and service records using custom Salesforce objects. This allows for faster record keeping, easier access to historical data, and better tracking of each customer interaction.
* **Enable Real-Time Tracking:** Empower garage staff and managers to view and update customer appointments, service status, quality checks, and billing information in real-time from any device. This enhances coordination and operational transparency.
* **Minimize Errors Through Automation:** Reduce manual calculation errors and operational oversights by implementing automated Flows, Apex triggers, and validation rules. These automations ensure that service charges, billing, and data updates are accurate and consistent.
* **Enhance Data Security & Access Control:** Define precise role hierarchies and profiles to manage user permissions and ensure that sensitive information is only accessed by authorized personnel. Sharing rules, OWD settings, and field-level security further enforce safe data practices.
* **Provide Actionable Insights:** Use Salesforce Reports and Dashboards to convert raw service and billing data into actionable metrics. These help managers track performance trends, identify bottlenecks, and make strategic decisions.
* **Boost Customer Satisfaction:** Enhance customer engagement through timely service updates, automated thank-you emails, and feedback collection mechanisms. This creates a professional and responsive service experience.
* **Ensure Scalability and Maintainability:** Design the system with modularity and scalability in mind, allowing for the addition of new features such as AI-based recommendations, chatbot interfaces, or external integrations in future versions.
* **Support Role-Based Usability:** Tailor the user experience for different roles such as Managers, Salespersons, and Admins, ensuring that each user only sees relevant data and has access to the tools they need.

**Phase 1: Requirement Analysis & Planning**

**Understanding Business Requirements:**

* The garage requires a way to register customer details, schedule appointments, and track service types.
* Payments must be calculated automatically based on selected services.
* Managers need an overview of operations via reports and dashboards.
* Feedback and rating collection should be enabled for quality control.

**Project Scope and Objectives:**

* Create custom Salesforce objects to hold structured data for: Customer Details, Appointments, Service Records, Billing, and Feedback.
* Implement logic using Apex Triggers and Flows to automate payment calculations and update service statuses dynamically.
* Enable secure data access by defining roles (Manager, Salesperson, Admin) and assigning object-level and field-level permissions.
* Build relationships among objects to support a clear, hierarchical data model suitable for future reporting.

**Data & Security Model Design:**

An efficient and secure data model was essential to support reporting, automation, and access control. The object relationships were designed as follows:

* **Customer Details** (Parent) → **Appointments** (Child)
* **Appointments** (Parent) → **Service Records** (Child)
* **Service Records** (Parent) → **Billing & Feedback** (Child)

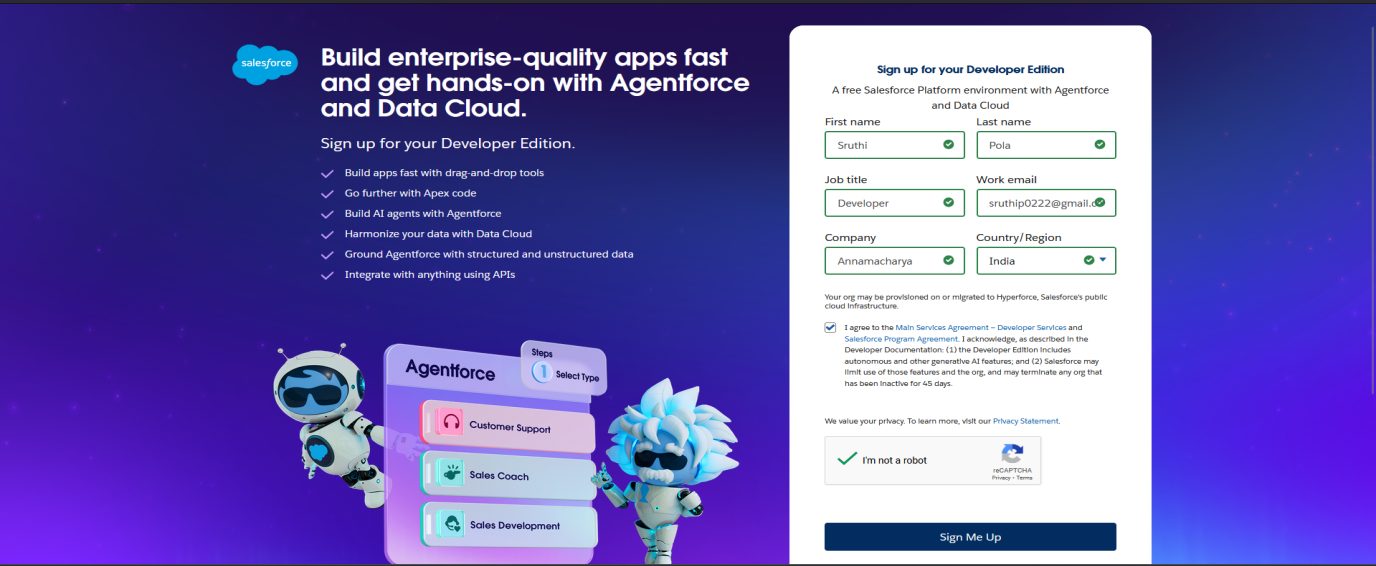
This lookup-based structure ensures data integrity while maintaining the flexibility to scale. Security was enforced using the following Salesforce mechanisms:

* Profile-based access control for defining object permissions
* Role Hierarchies to manage record visibility across different roles
* Organization-Wide Defaults (OWD) and Sharing Rules to control access to sensitive records, such as limiting access to service and billing records only to assigned staff

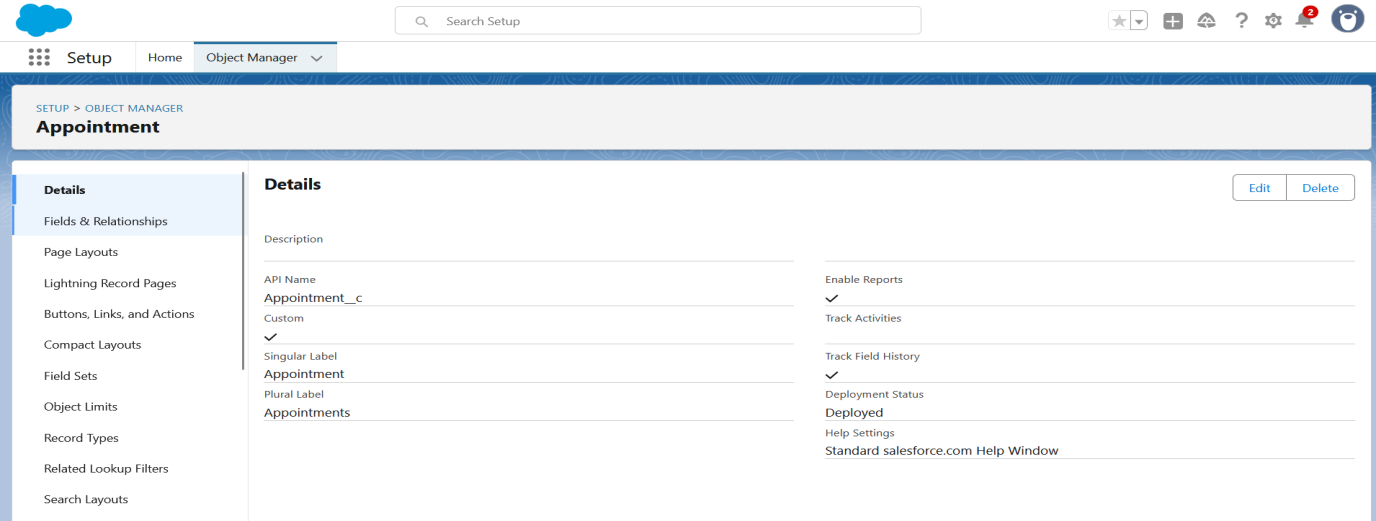
**Phase 2: Salesforce Backend Development & Configurations**

**Environment Setup:**

* Developer account created via [https://developer.salesforce.com/]
* Basic org setup and branding applied

**Fig: Signing Up for Developer Org**

**Custom Objects Created:**

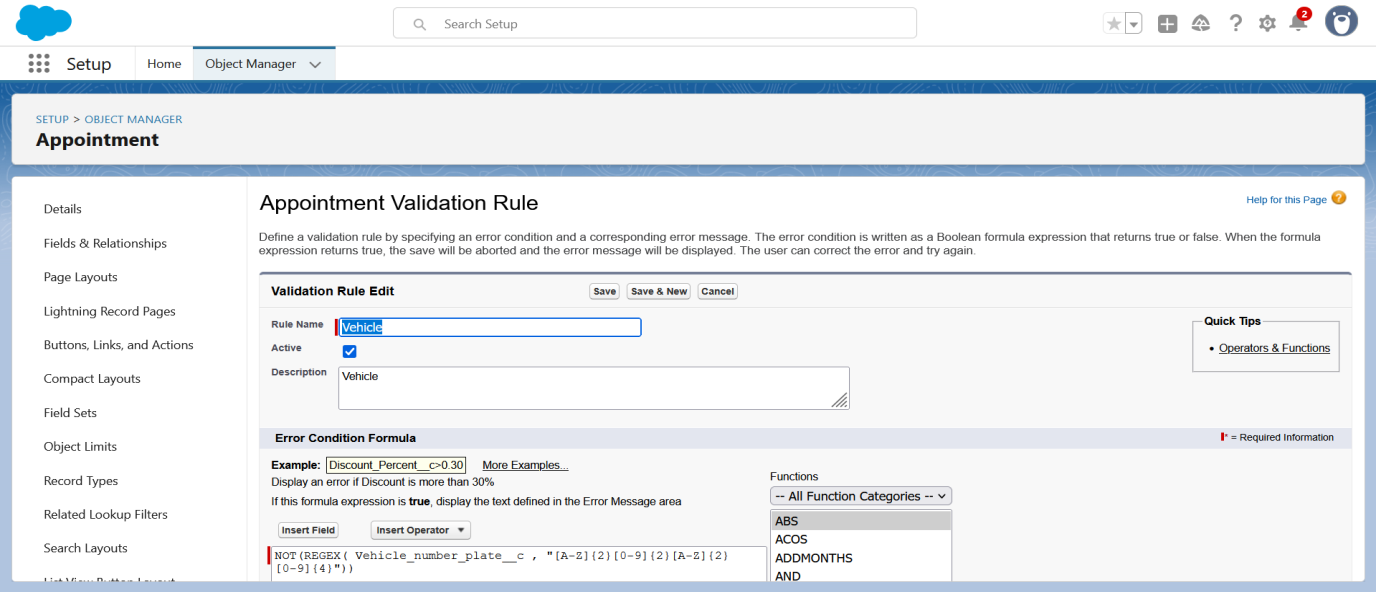
* Customer Details
* Appointment (Auto-number format: app-{000})
* Service Records (Auto-number format: ser-{000})
* Billing Details and Feedback (Auto-number format: bill-{000})

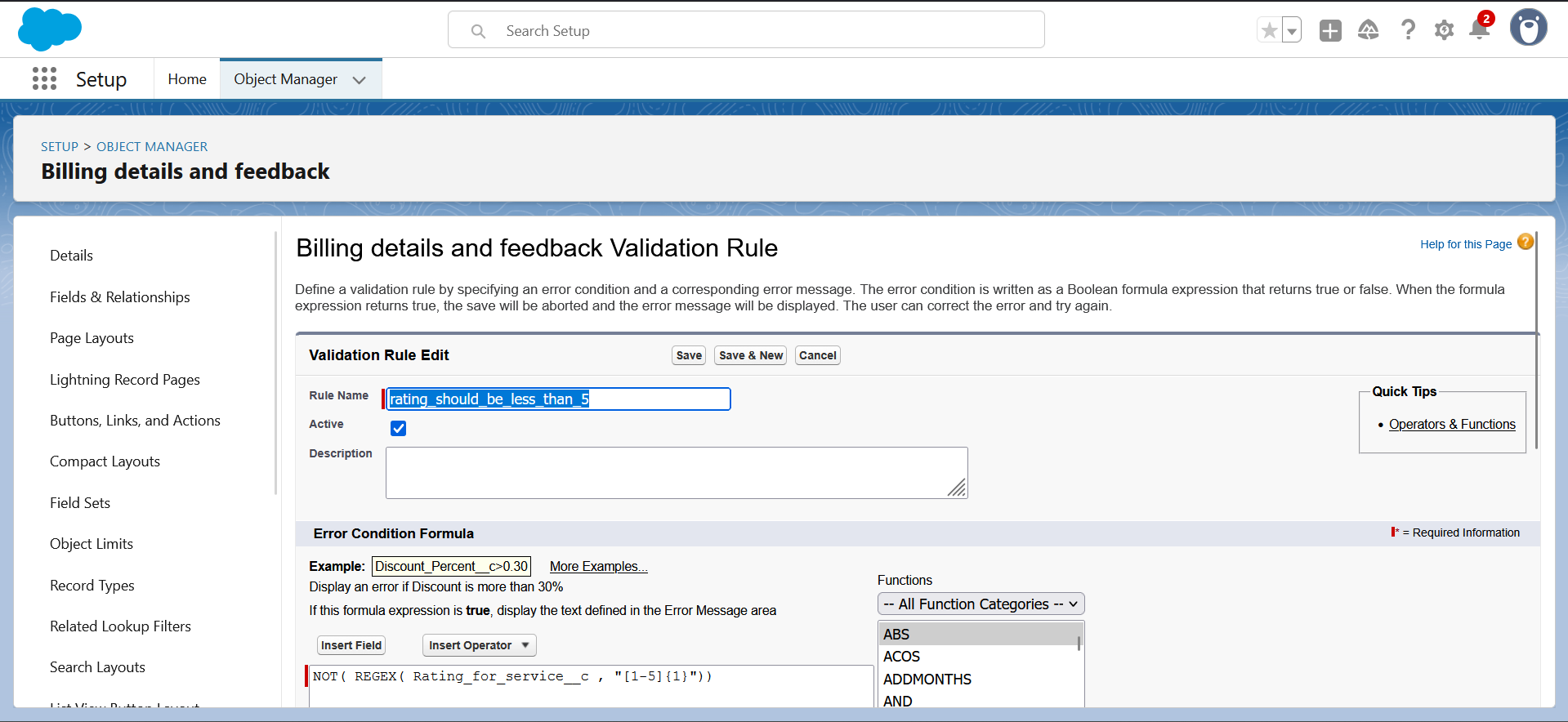
**Fig: Object Creation**

**Field Configurations:**

* Customer Details: Phone (Phone), Gmail (Email)
* Appointment: Date, Vehicle Plate (Text/Regex), Service Checkboxes (Maintenance, Repair, Replacement)
* Service Records: Quality Check Status (Checkbox), Service Status (Picklist)
* Billing: Payment Paid (Currency), Rating (Text, Required), Payment Status (Picklist)

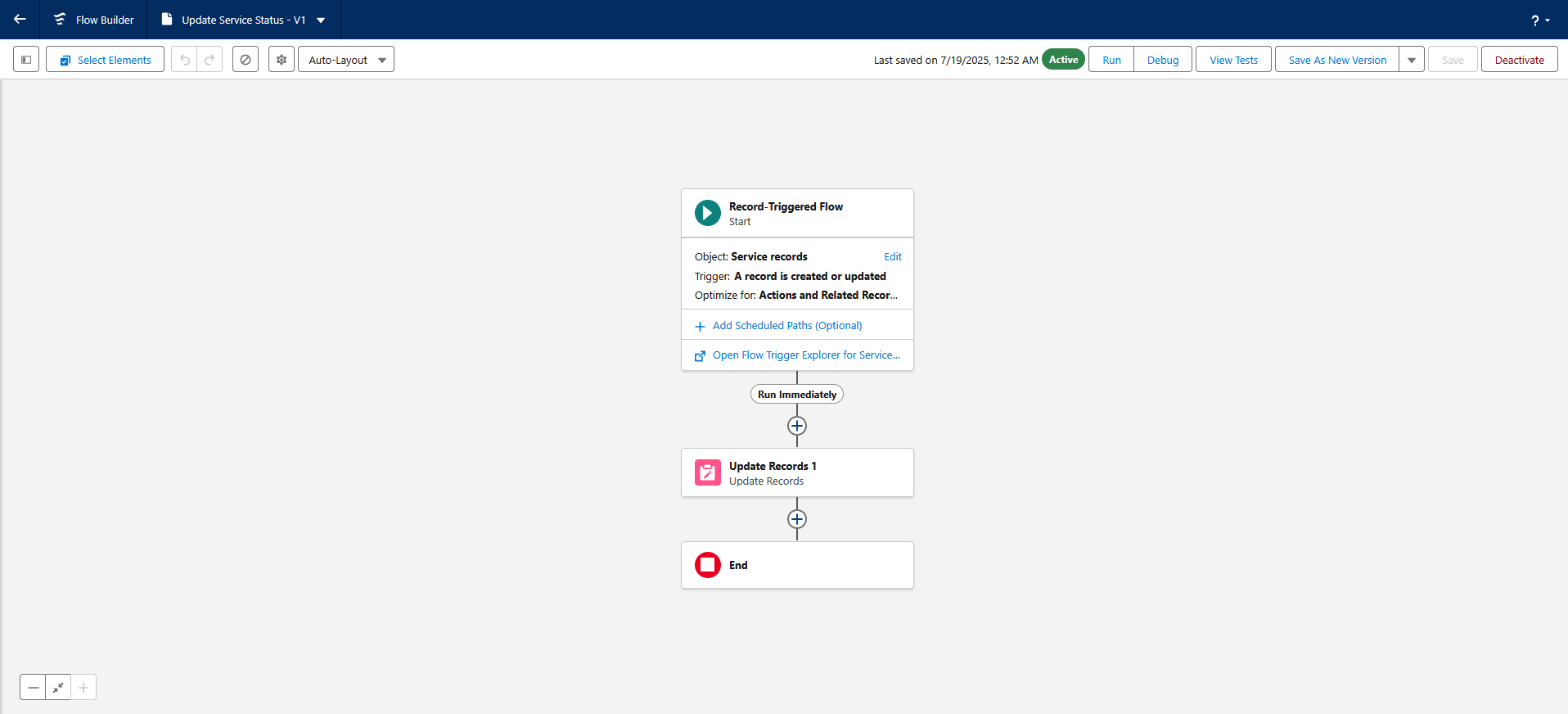
**Validation Rules:**

* Vehicle Plate Format (e.g., AP01AB1234)
* Rating (Only values 1–5)

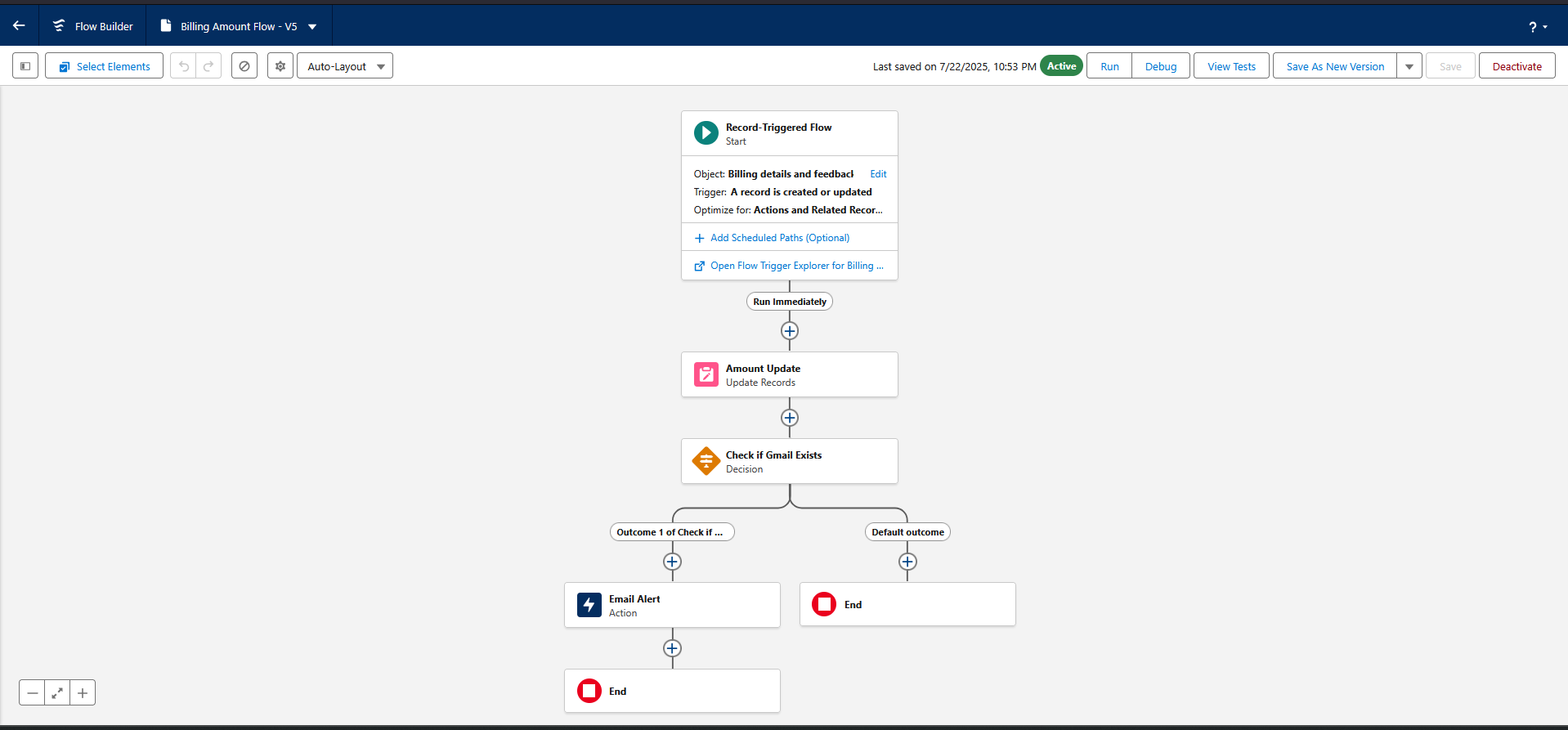
**Fig: Validation Rule for Vehicle Plate Format**

**Fig: Validation Rule for Rating**

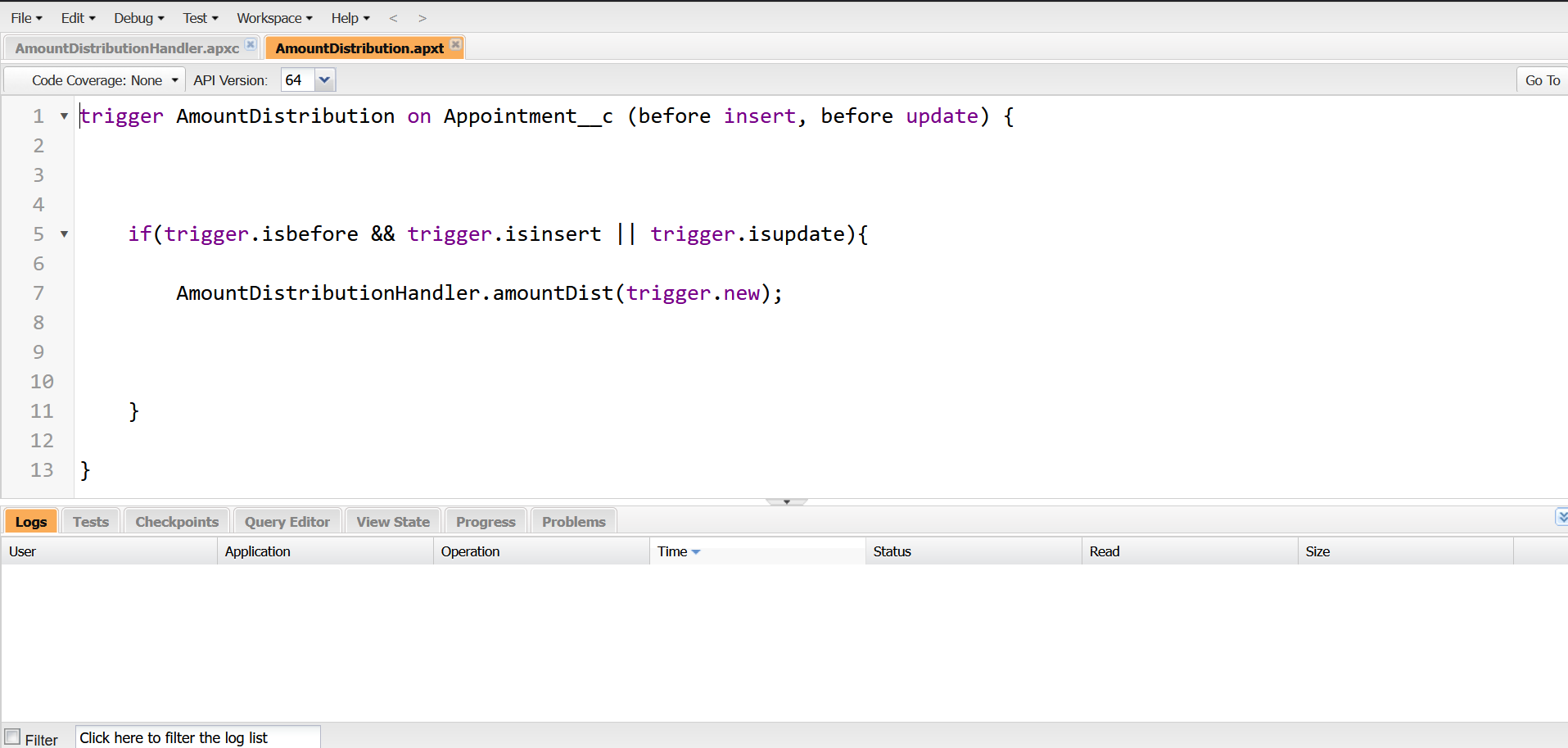
**Automation:**

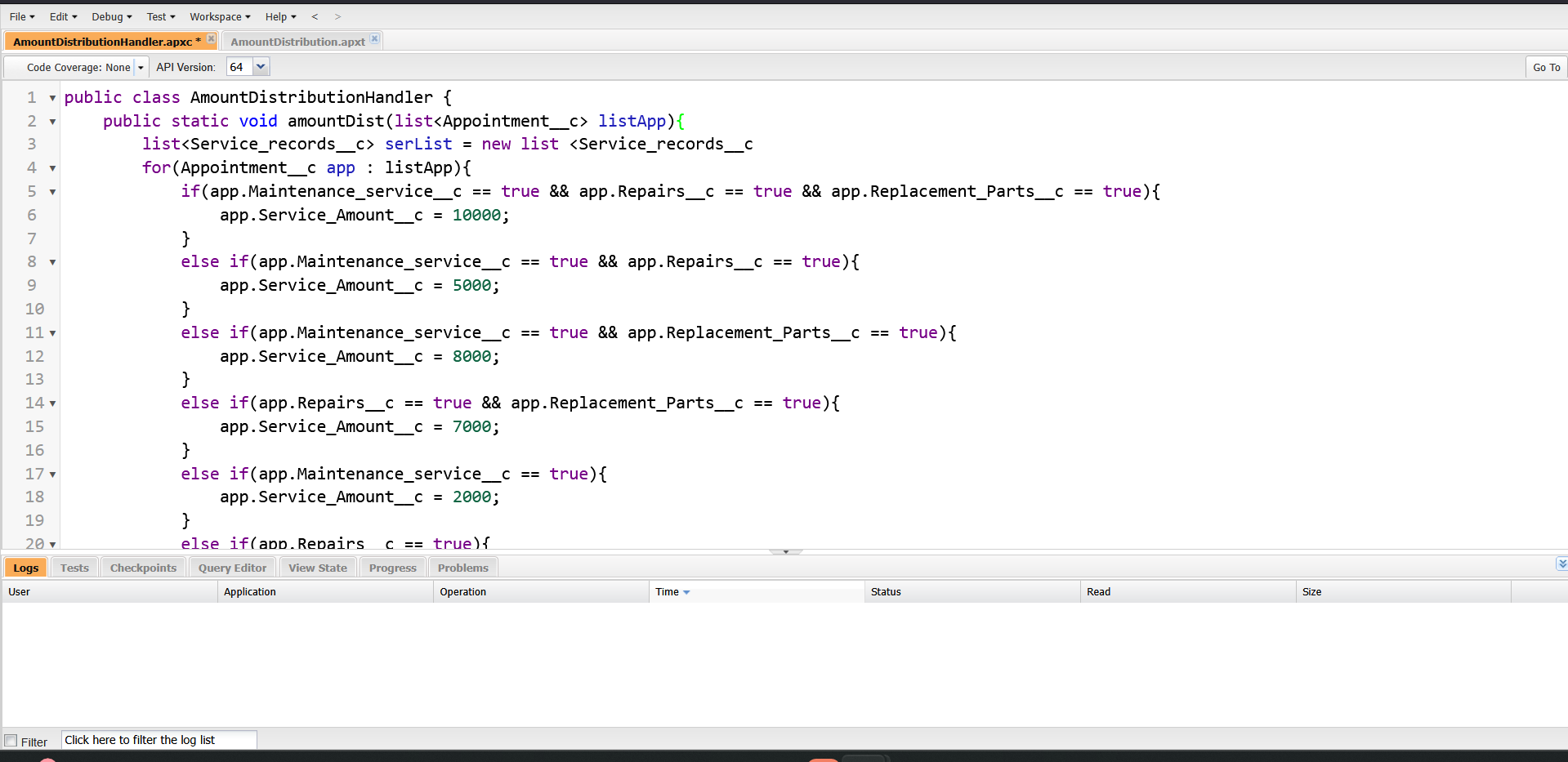
* **Flows:**
  + Update Payment after Status = Completed
  + Update Service Status after Quality Check is true

**Fig: Service Record Flow**

****

**Fig: Billing Amount Flow**

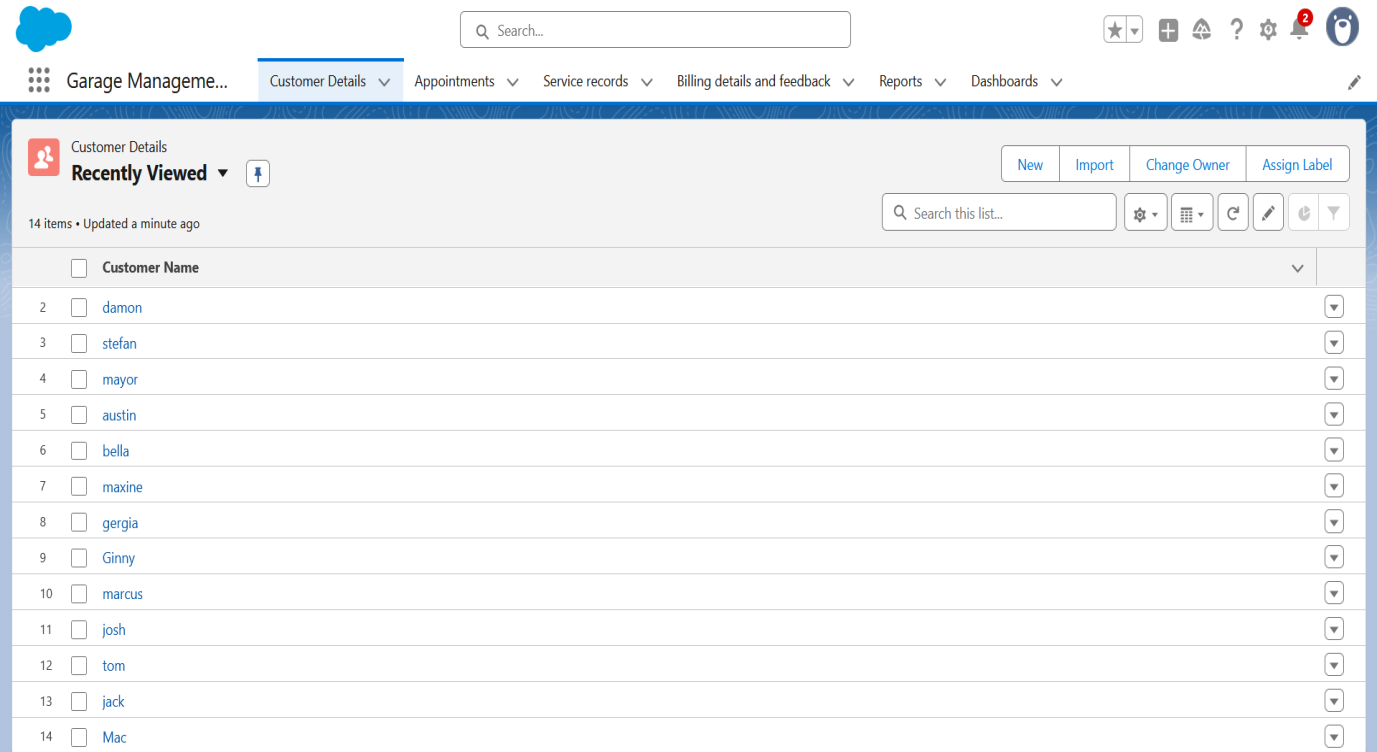
* **Apex Trigger:**
  + Trigger: AmountDistribution (before insert/update on Appointment)
  + Handler: AmountDistributionHandler class calculating service amount.

**Fig: AmountDistribution Trigger**

**Fig: AmountDistributionHandler Class**

**Phase 3: UI/UX Development & Customization**

**Lightning App Creation:**

* App Name: Garage Management Application
* Tabs: Customer Details, Appointments, Service Records, Billing, Reports, Dashboards.

**Fig: Garage Management Application**

**Tabs Setup:**

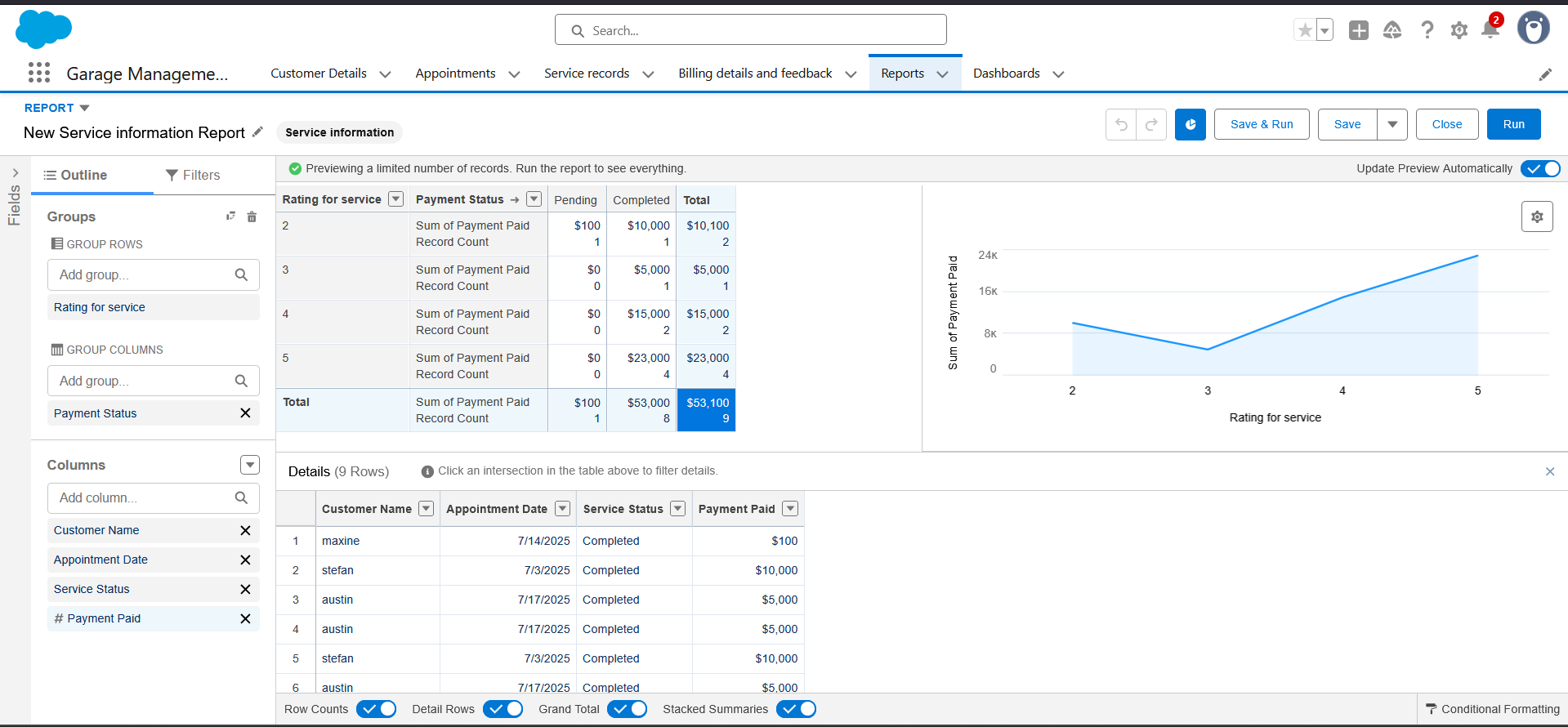
* Created custom tabs for all objects with default tab styles

**Page Layouts & Dynamic Forms:**

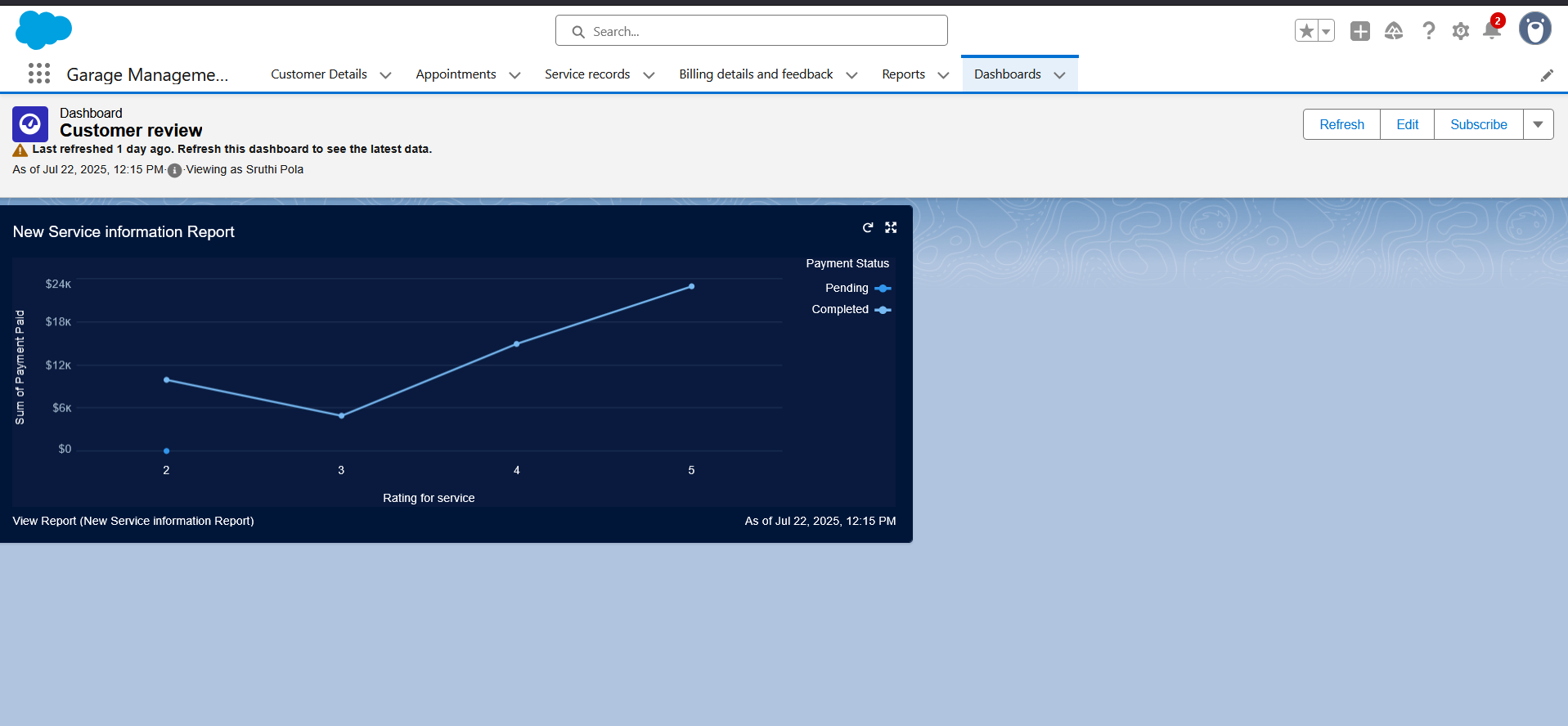
* Custom layouts for each object
* Required fields and checkboxes placed prominently
* Dynamic visibility used for service selections

**Reports:**

* Custom Report Type: Customer → Appointment → Service Record → Billing
* Fields: Appointment Date, Payment Paid, Service Status
* Grouped by: Rating of service, Payment Status

**Fig: Report**

**Dashboards:**

* Folder: Service Rating Dashboard
* Chart: Line chart displaying ratings and payments over time
* Subscription: Weekly report email every Monday

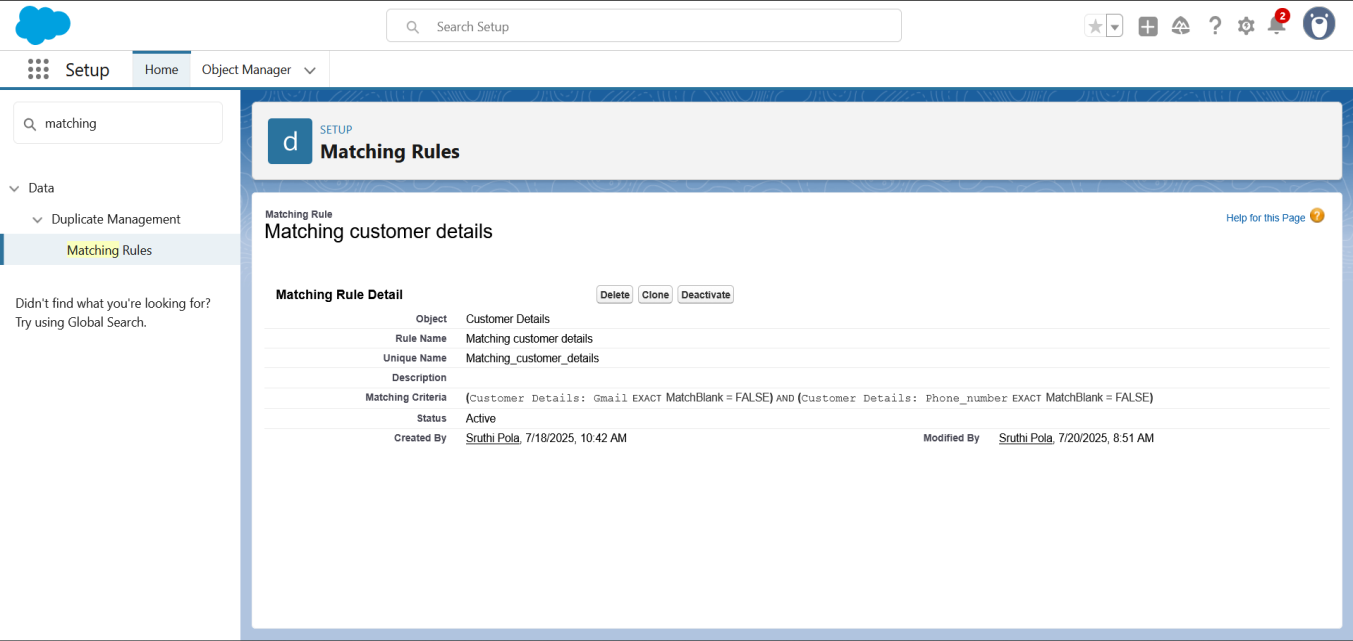
**Fig: Dashboards**

**Phase 4: Data Migration, Testing & Security**

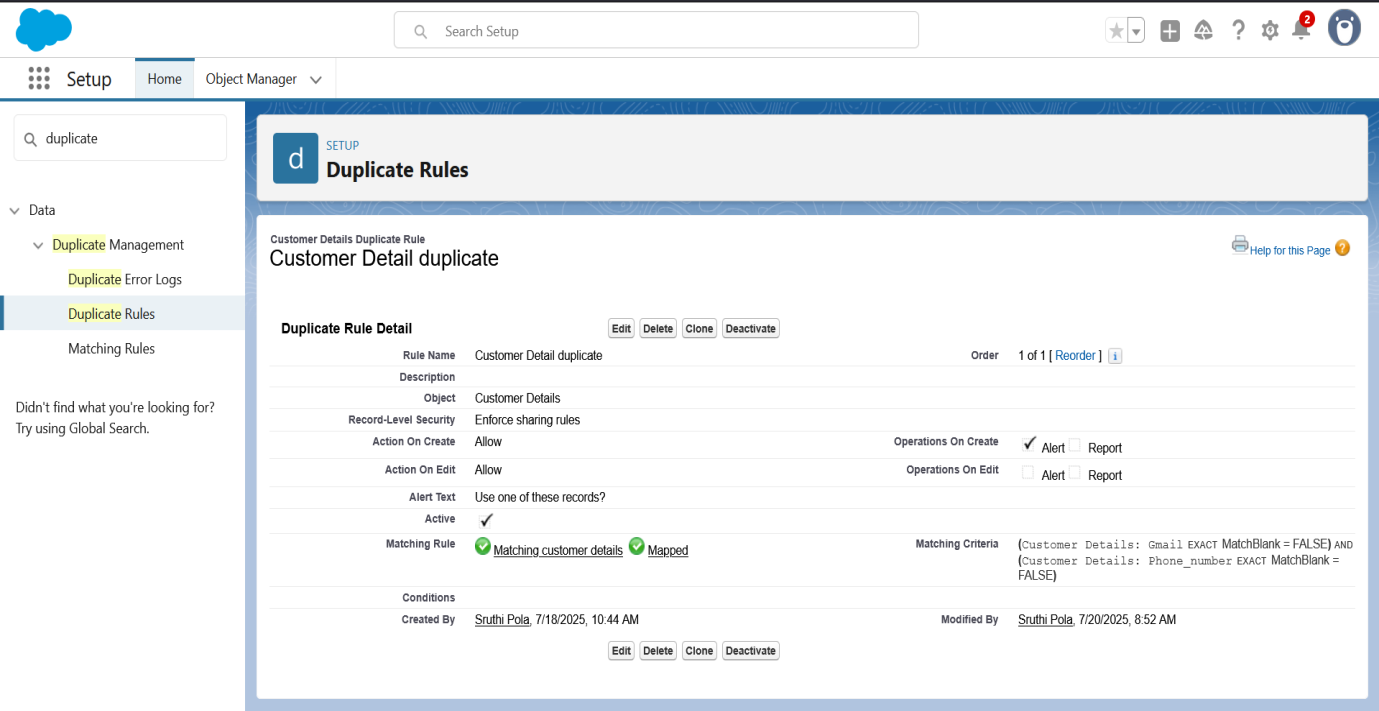
**Data Migration:**

* Used Salesforce UI for manual data entry (10+ records per object)

**Matching & Duplicate Rules:**

* Matching Rule: Gmail & Phone for Customer Details
* Duplicate Rule: Triggered on record creation

**Fig: Matching Customer Details**

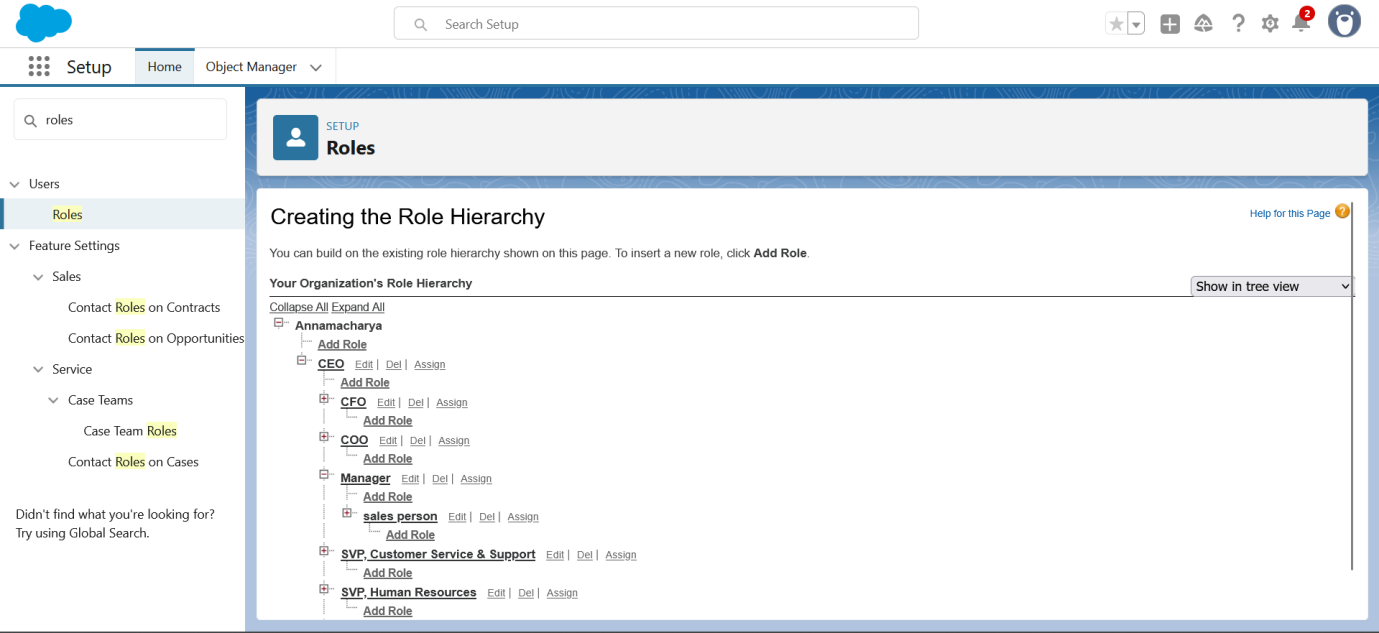
**Fig: Customer Detail Duplicate**

**Field History Tracking:**

* Enabled on all custom objects

**Security Settings:**

* OWD: Private for Service Records
* Sharing Rule: Salesperson → Manager (Read/Write)
* Public Group: Sales Team
* Profiles: Manager, Salesperson with appropriate permissions
* Roles: CEO > Manager > Salesperson (Role hierarchy)

**Fig: Roles**

**Testing & Q&A:**

* Manual tests conducted for each feature.
* Sample Inputs and Expected Outputs documented for:
  + Record creation
  + Validation errors
  + Flow updates
  + Trigger execution

**Phase 5: Deployment, Documentation & Maintenance**

**Deployment Strategy:**

Since the Garage Management System was developed within a Salesforce Developer Org, traditional deployment processes such as change sets or managed packages were not required during the initial build. However, for production or sandbox deployments in a real-world environment, the system is designed to support structured deployment using Salesforce Change Sets, Unmanaged Packages, or Salesforce DevOps Center tools. These options ensure safe migration of metadata components, including custom objects, flows, Apex classes, profiles, and validation rules, from one org to another. A well-documented deployment strategy ensures future scalability and maintainability of the solution in enterprise environments.

**System Maintenance:**

To ensure the application continues to function as expected, routine system monitoring and maintenance practices have been put in place. Debug logs are enabled in the org to help monitor flow executions, Apex errors, and data updates. These logs assist developers and administrators in troubleshooting issues quickly. Field history tracking is also enabled for critical objects to monitor changes made by different users. Scheduled reviews of flows and triggers are recommended to optimize performance over time.

**Documentation & Training:**

Comprehensive documentation was created throughout the project to support future users and administrators of the application. This includes:

* **Labeled Screenshots:** Each step of the configuration, including object creation, field setup, validation rules, flows, and dashboards, has been documented with screenshots for easy reference.
* **Object Relationship Diagram:** A clear visual representation of the relationships between objects (Customer → Appointment → Service Record → Billing & Feedback) helps users and developers understand data flow.
* **Testing Summary:** All major test cases for record creation, trigger functionality, flow automation, and validation rule enforcement have been compiled and summarized.
* **Admin Training Material:** A concise guide is included to support new Salesforce users or garage employees in navigating the app, performing updates, and accessing reports.

**Conclusion**

The Garage Management System (GMS) built using Salesforce CRM offers a robust, cloud-based solution to the complex needs of automobile service centers. Through the digital transformation of garage operations, the system effectively replaces traditional, error-prone manual processes with automated and structured workflows.

From the moment a customer’s details are captured, GMS ensures seamless handling of appointment bookings, service type selection, billing, and post-service follow-up. The use of validation rules minimizes incorrect data entry, while Apex triggers and flows automate the calculation of service charges and update statuses without manual effort.

By leveraging Salesforce’s built-in capabilities such as object customization, field-level security, and real-time analytics, this system empowers garage staff to focus more on customer service rather than administrative tasks. It also provides managers with valuable operational insights through custom reports and dashboards, helping in identifying trends, optimizing resources, and improving decision-making.

In addition, the system enhances customer satisfaction by sending personalized thank-you emails upon successful payment and collecting structured feedback to improve service quality. Security is maintained via well-structured role hierarchies and sharing rules, ensuring that sensitive information is accessible only to the right personnel.

Overall, this Garage Management System proves to be a scalable, secure, and maintainable solution that not only supports current business needs but also lays a strong foundation for future enhancements like AI integration, chatbot support, and mobile accessibility. It successfully showcases how Salesforce can be utilized to create an industry-specific CRM that improves efficiency, accuracy, and customer engagement. It simplifies appointment booking, cost computation, service tracking, and billing—while enhancing customer engagement through emails and feedback mechanisms. The system is secure, user-friendly, and scalable.

**Future Enhancements**

* Integration with WhatsApp for appointment reminders
* AI-based recommendation engine for service packages
* Voice-enabled service requests using Einstein Voice
* Mobile Lightning App for customers