GARAGE MANAGEMENT SYSTEM

College Name: Bharathidasan college of arts and science
College code:bru3d
Team ID:NM2025TMID24652

Team Members :

Team Leader Name : Priyadarshini.V

Email ID : priyadarshini231205@gmail.com

Team Member 1 : Subhiksha.P

Email ID : subhikshaperumalsamy@gmail.com

Team Member 2 : Maheswari.A

Email ID : maheswariskp2006@gmail.com

Team Member 3 : Poornima.D

Email ID : poornima150505@gmail.com

1.INTRODUCTION

1.1 Project Overview

The Garage management system is designed to improve accuracy, reduce manual efforts and provide better visibility into vehicle maintenance, inventory and service data for automotive garages

Garage Management System

The system is designed to improve efficiency, reduce manual effort, and provide better visibility intogarage operations, vehicle records, and service history.

Vehicle management		Service & Maintenance

Garage Management System

Reports & Analytics

User interface
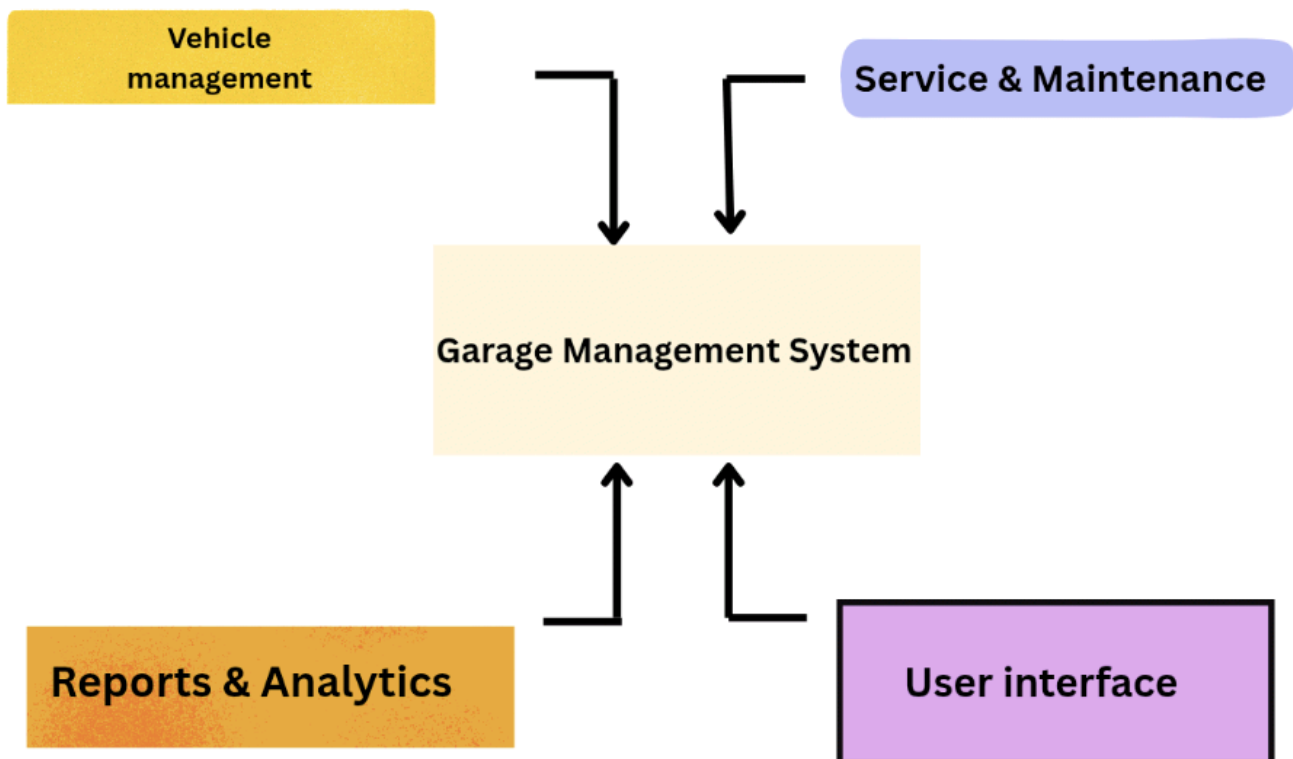
1.2 Purpose
The purpose of the Garage Management System is to provide an efficient and user-triendly olatform that streamlines vehicle management service schedulina. and repair trackina. It ensures accurate record-keeping of customer vehicles, facilitates seamless billing and service transactions, and enhances overall operational efficiency for garage businesses

▶ Salesforce

Creating Developer Account:

Creating a developer org in salesforce.

1. Go to https://developer.salesforce.com/signup
2. On the sign up form, enter the following details :

Account Activation

Go to the inbox of the email that you used while signing up. Click on the verify account to activate your account. The email may take 5-10mins.
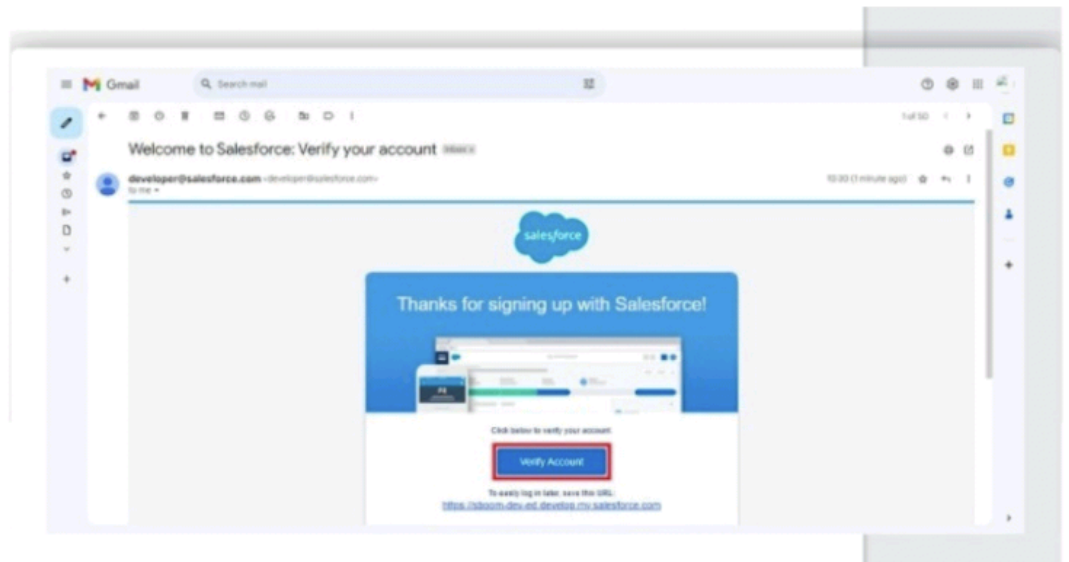
Click on Verify Account

Give a password and answer a security question and click on change password.

Then you will redirect to your salesforce setup page.
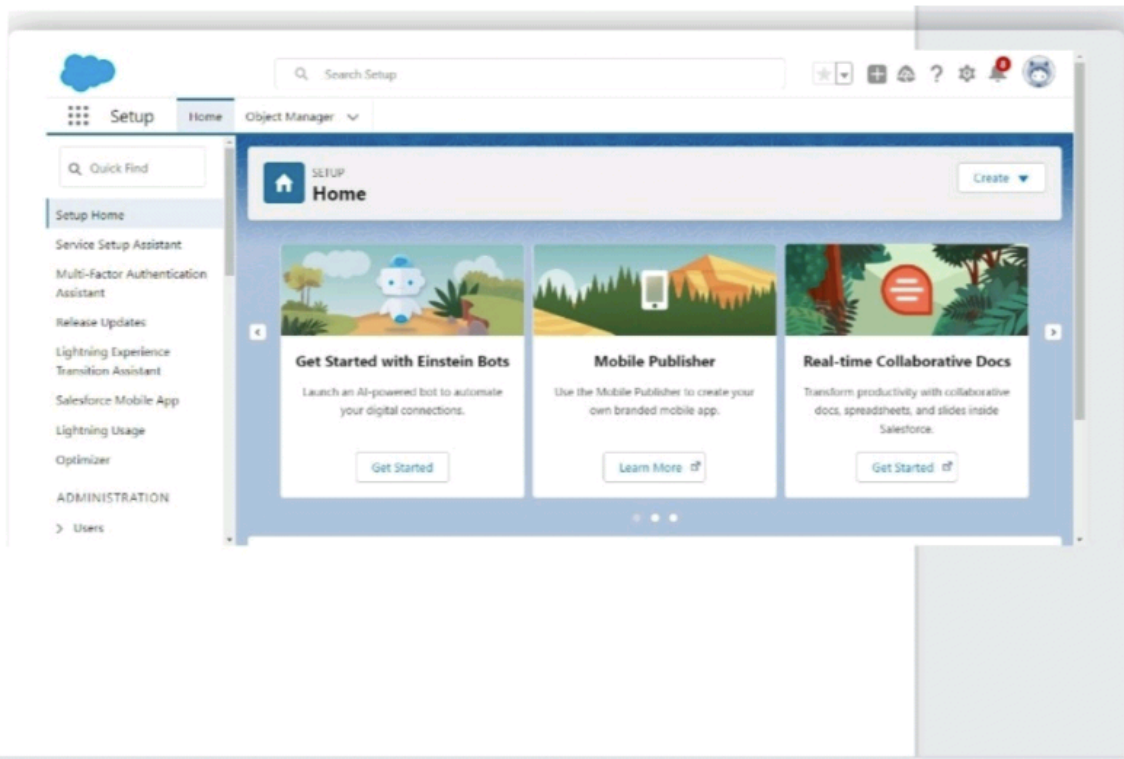
▶ Object

Create Customer DetailsObject

Create Appointment Object

Create Service records Object

Create Billing details and feedback Object

▶ Tabs

Creating a Custom Tab

To create a Tab:(Customer Details)

To create a Tab:(Appointments)

To create a Tab:(Service record)
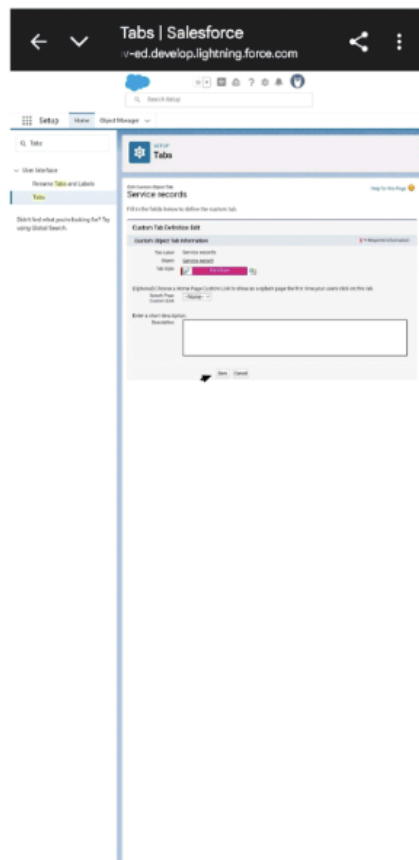
To create a Tab:(Billing details and feedback)
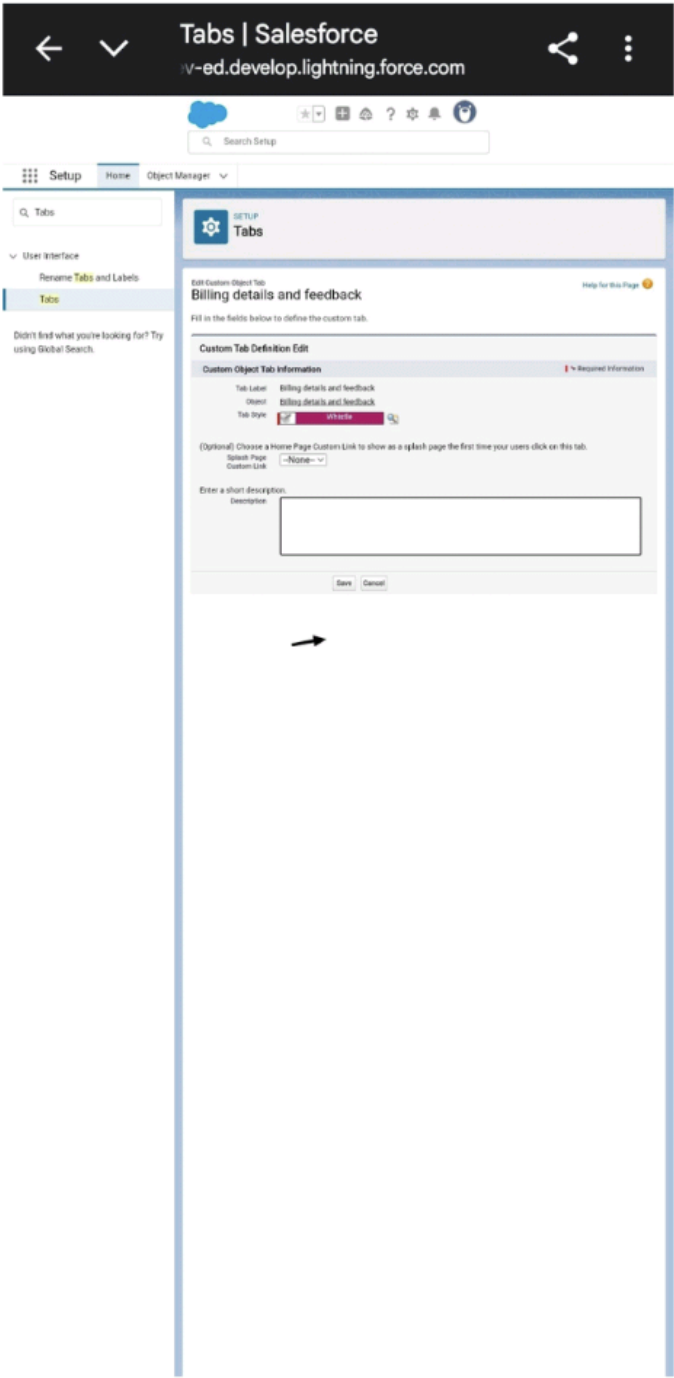
▶ The Lightning App

1. **Go to setup page >> search "app manager" in quick find >> select "app manager" >> click on New lightning App.**

Fill the app name in app details as Garage Management Application >> Next >> (App option page) keep it as default >> Next >> (Utility Items) keep it as default >> Next.

▶ Fields

Creation of fields for the Customer Details object

Go to setup >> click on Object Managr >> type object name(Customer Details) in search bar >> click on the object.
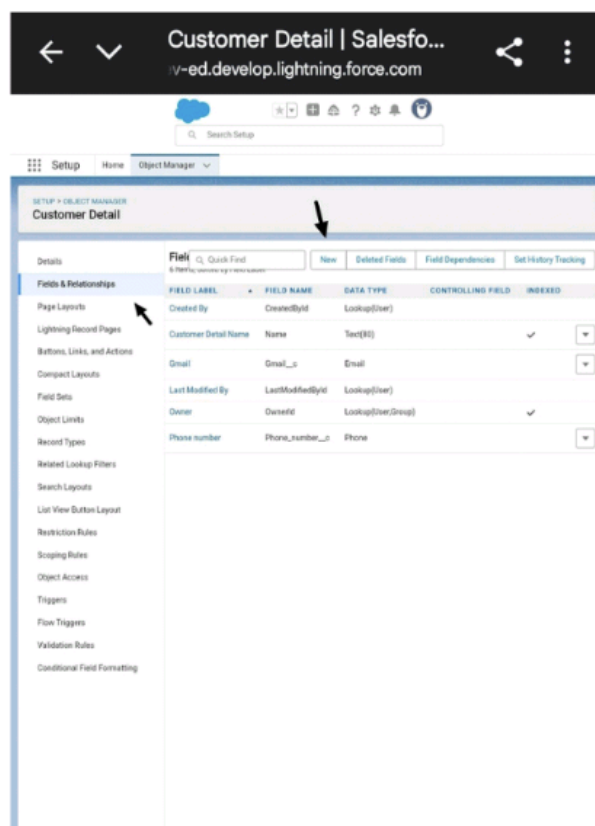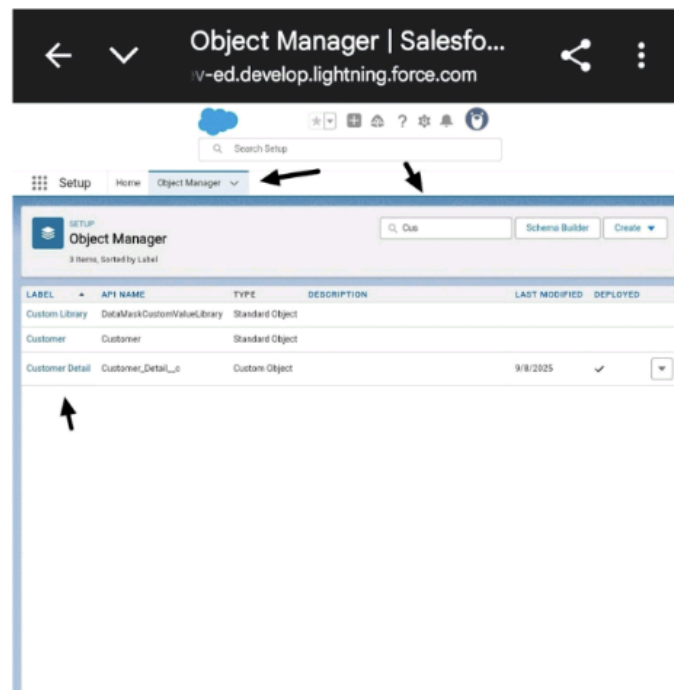
Creation of Lookup Fields

Creation of Lookup Field on Appointment Object :

Go to setup >> click on Object Manager >> type object name(Appointment) in the search bar >> click on the object
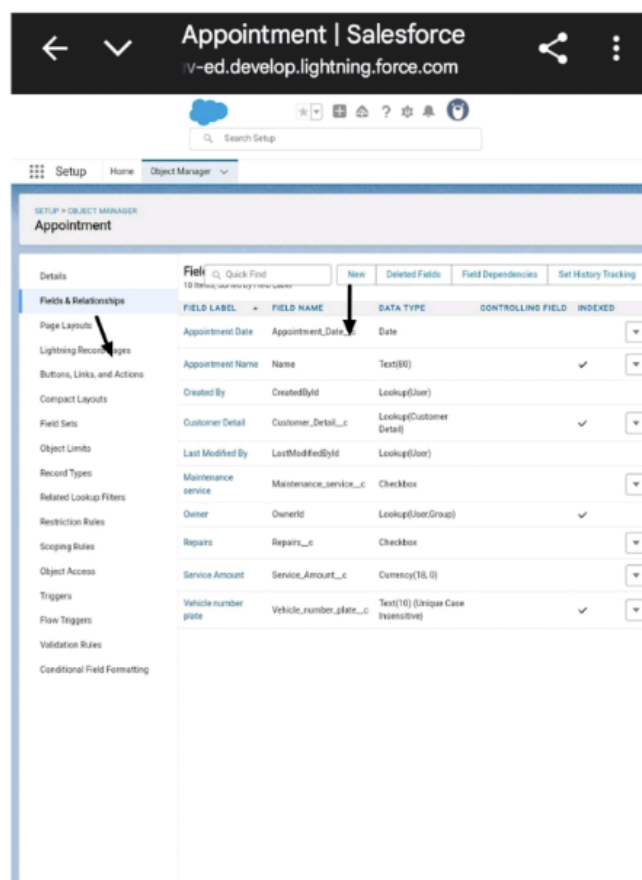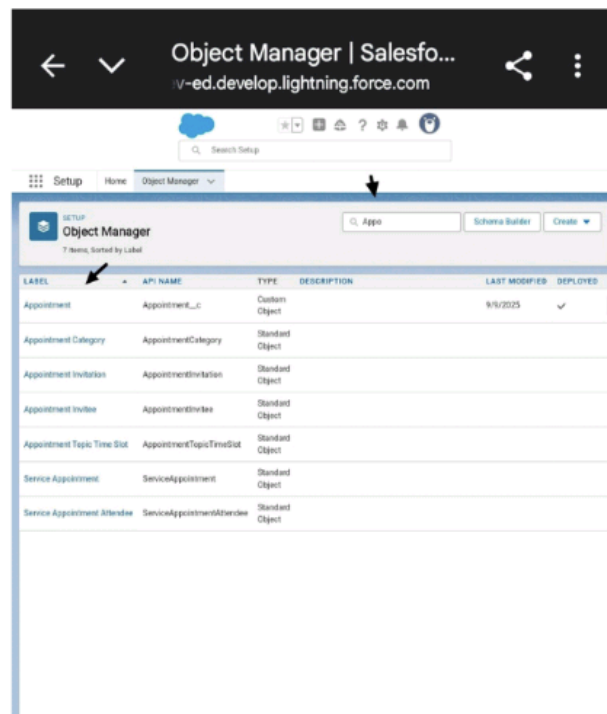
Creation of Checkbox Fields

Creation of Checkbox Field on Appointment Object :
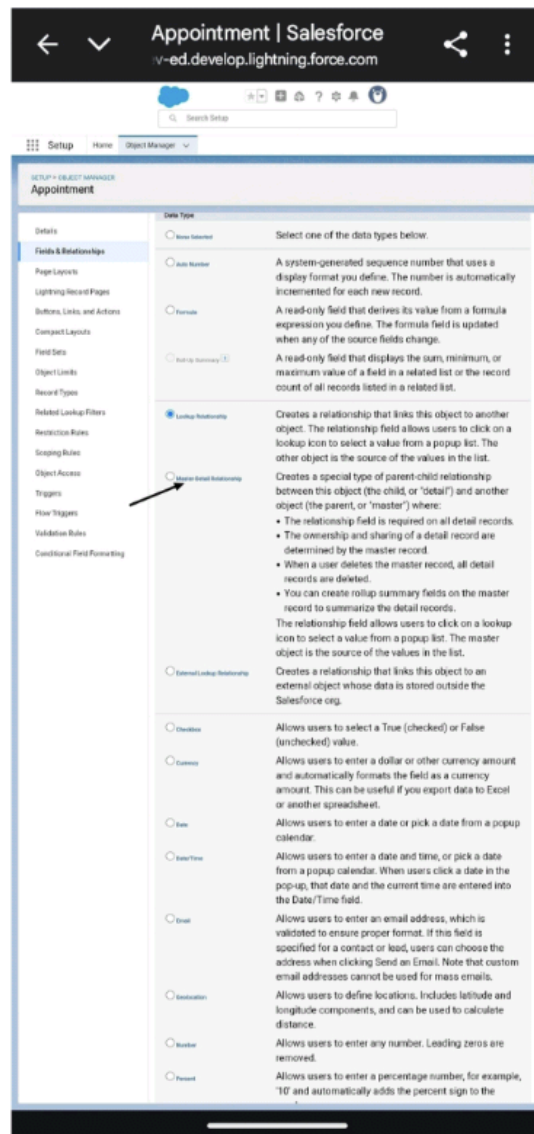
Give the Field Label : Maintenance service

Field Name : is auto populated

Default value : unchecked

Click on next >> next >> save.

Creation of Date Field on Appointment Object :

▶ Validation rule

To create a validation rule to an Appointment Object

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

Q Search Setup

⋮⋮⋮ Setup   Home   Object Manager ∨

SETUP > OBJECT MANAGER
Billing details and feedback

Details

Details   Edit Delete

Fields & Relationships

Page Layouts   Description

Lightning Record Pages

Buttons, Links, and Actions   API Name   Enable Reports
                                           Billing_details_and_feedback__c   ✓
Compact Layouts   Custom   Track Activities
                                           ✓   ✓
Field Sets   Singular Label   Track Field History
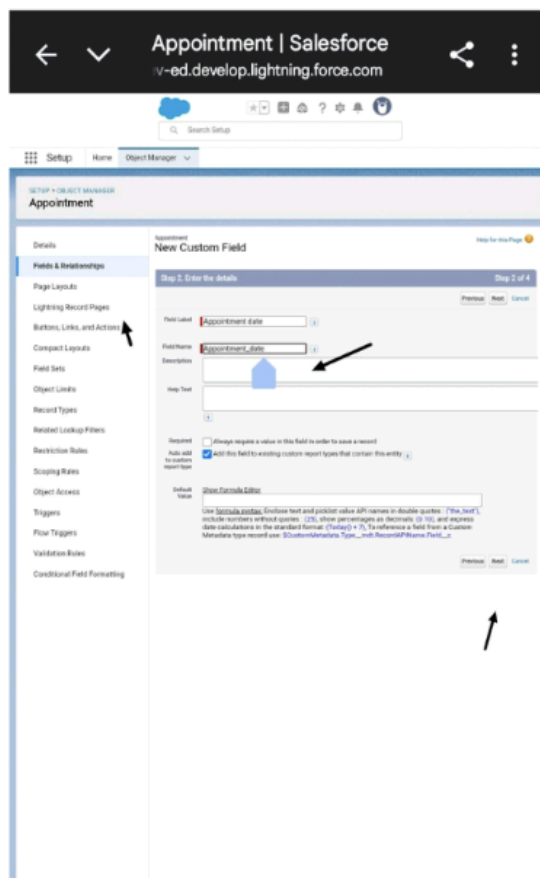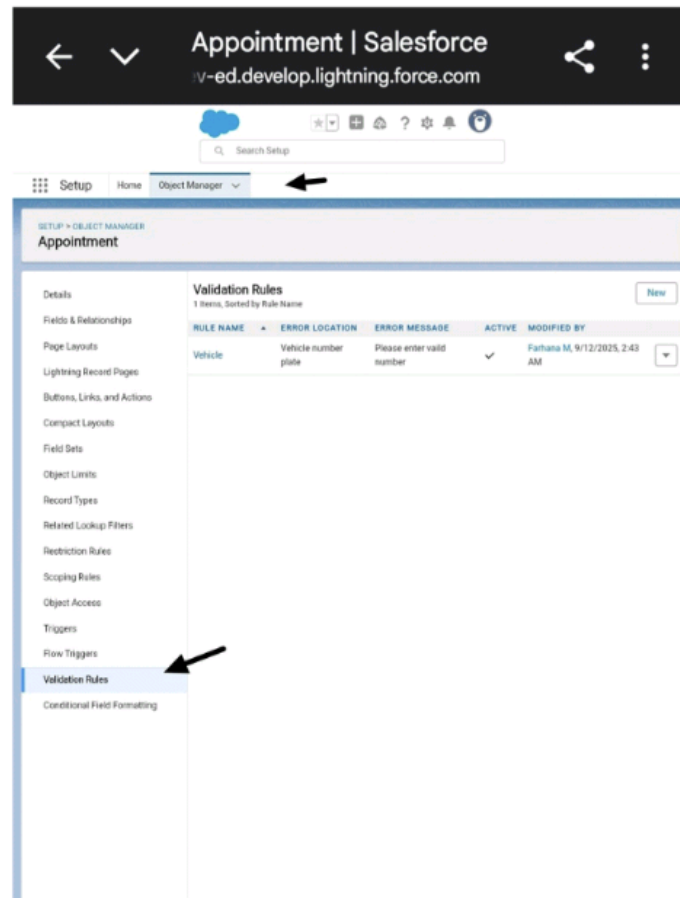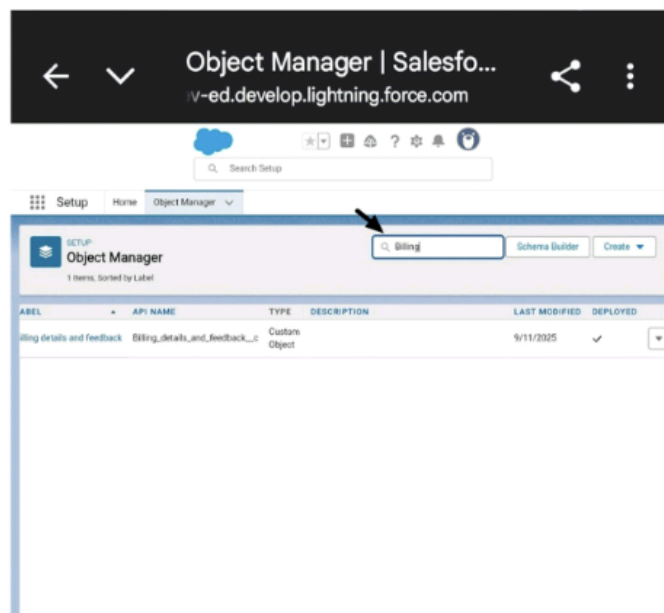   Billing details and feedback   ✓
Object Limits   Plural Label   Deployment Status
                                           Billing details and feedback   Deployed
Record Types   Help Settings
                                                                             Standard salesforce.com Help Window
Related Lookup Filters

Restriction Rules

Scoping Rules

Object Access

Triggers

Flow Triggers

Validation Rules

Conditional Field Formatting

Q Search Setup

⋮⋮⋮ Setup   Home   Object Manager ∨

SETUP > OBJECT MANAGER
Billing details and feedback

Details   **Validation Rules**   New
                                 1 items, Sorted by Rule Name
Fields & Relationships

Page Layouts   RULE NAME ▲   ERROR LOCATION   ERROR MESSAGE   ACTIVE   MODIFIED BY

Lightning Record Pages   rating_should_be_less_than_5   Rating for service   rating should be from   ✓   Farhana M,
                                                                                  1 to 5   9/12/2025, 2:50 AM   ▾
Buttons, Links, and Actions

Compact Layouts

Field Sets

Object Limits

Record Types

Related Lookup Filters

Restriction Rules

Scoping Rules

Object Access

Triggers

Flow Triggers

Validation Rules

Conditional Field Formatting

▶ Duplicate rule

Go to quick find box in setup and search for matching Rule.

Click on matching rule >> click on New Rule.

To create a Duplicate rule to an Customer details Object

ADVANTAGES & DISADVANTAGES

Advantages

Improved efficiency

Accurate record -keeping

Enhanced customer satisfaction

Date-driven decisions

Time-saving

Disadvantages

Initial Investment

Technical Issues

Data security risks

Training required

Dependence on technology

Conclusion
The Garage Management System is a comprehensive solution designed to streamline vehicle management, enhance service tracking, and improve operational efficiency for automotive garages. By providing a user-friendly interface and generating insightful reports, the system enables garages to make data-driven decisions, reduce manual errors, and ultimately improve customer satisfaction. With its robust features and benefits, the Garage Management System is an essential tool for modern automotive garages looking to optimize their operations and stay competitive in the industry.

APPENDIX

- Source Code: Provided in Apex Classesand Triggers

```apex
public class VehicleManagement  {

Public static void updateVehicleStatus (Id vehicleId,String status){

   Vehicle__c  vehicle = [SELECT Id, Status__c FROM Vehicle __c WHERE Id =  : vehicle Id];

   Vehicle.Status__c = status;

   update vehicle;

   }
   }
```

```apex
trigger VehicleTrigger on Vehicle__c (before insert, before update) {
            if (Trigger.isBefore && Trigger.isInsert) {
                  // Logic for before insert
            } else if (Trigger.isBefore && Trigger.isUpdate) {
                  // Logic for before update
                        }
                        }
```