**GARAGE MANAGEMENT SYSTEM**

Project Document

**Prepared by**

**Name**  : ANAND D

**Mail id**  : anand.duraisamy110@gmail.com

**College Name** : Mahendra Engineering College(Autonomous)

Namakkal

INDEX

Garage Management system

Milestone 1:Salesforce

 Creating Developer Account

 Account Activation

Milestone 2: Object

 Create Customer DetailsObject

 Create Appointment Object

 Create Service records Object

 Create Billing details and feedback Object

Milestone 3: Tabs

 Creating a Custom Tab

 Creating Remaining Tabs

Milestone 4: The Lightning App

 Create a Lightning App

Milestone 5: Fields

 Creation of fields for the Customer Details object

 Creation of Lookup Fields

 Creation of Checkbox Fields

 Creation of date Fields

 Creation of Currency Fields

 Creation of Text Fields

 Creation of Picklist Fields

 Creating Formula Field in Service records Object

Milestone 6: Validation rule

 To create a validation rule to an Appointment Object

 To create a validation rule to an Service records Object

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

Milestone 7: Duplicate rule

 To create a matching rule to an Customer details Object

 To create a Duplicate rule to an Customer details Object

Milestone 8: Profiles

 Manager Profile

Sales person Profile

Milestone 9: Role & Role Hierarchy

 Creating Manager Role

 Creating another roles

Milestone 10: Users

 Create User

Creating another users

Milestone 11: Public groups

 Creating New Public Group

Milestone 12: Sharing Setting

 Creating Sharing settings

Milestone 13: Flows

 Create a Flow

Milestone 14: Apex Trigger

 Apex handler

Milestone 15: Reports

 Create a report folder

 Sharing a report folder

 Create Report Type

Create Report

Milestone 16: Dashboards

 Create Dashboard Folder

 Create Dashboard

Conclusion :

Garage Managment System

A garage management system (GMS) is a comprehensive software solution designed to revolutionize the way auto repair shops and garages operate by integrating various aspects of their daily activities into a single, streamlined platform. This advanced system facilitates a range of functions critical to efficient garage management, including appointment scheduling, work order tracking, inventory control, customer relationship management (CRM), billing and invoicing, and comprehensive reporting and analytics.

At the core of a GMS is its **appointment scheduling** functionality, which allows customers to easily book, reschedule, or cancel service appointments either online or through the system. This feature not only enhances customer convenience but also significantly reduces administrative workload by automating appointment confirmations and reminders. Automated notifications help minimize no-shows and ensure that both customers and staff are kept informed of upcoming appointments, contributing to a smoother operational flow and more effective time management.

The **work order management**  component is essential for maintaining operational efficiency. It allows for the creation, tracking, and management of detailed work orders that outline the specifics of each service request, including job descriptions, labor estimates, and required parts. This ensures that tasks are assigned accurately, progress is monitored in real time, and any issues are addressed promptly. By providing clear visibility into the status of each job, work order management helps streamline workflows, reduce delays, and maintain high standards of service quality.

**Inventory management**  within a GMS is another crucial feature that supports effective operations. This component tracks the availability of parts and supplies, providing real-time insights into stock levels. It automatically alerts managers when inventory is low and can trigger reorder processes to prevent stockouts. This proactive approach to inventory management helps ensure that the garage is always equipped with the necessary parts to complete repairs efficiently, reducing downtime and minimizing disruptions to service delivery.

The **customer relationship management (CRM)**  capabilities of a GMS enhance the way garages interact with their clients. By storing detailed customer information, including contact details, service history, and preferences, the system enables personalized service and targeted communication. This helps build stronger customer relationships, improve satisfaction, and foster loyalty. CRM features also support

marketing efforts by facilitating the creation of targeted promotions and follow-up communications based on customer data.

In terms of financial management, a GMS includes **billing and invoicing** functionalities that streamline the payment process. It generates accurate invoices, tracks payments, and manages accounts receivable. This automation reduces the risk of billing errors and simplifies financial management by providing clear and organized records of transactions. Integrated financial reporting tools also offer insights into revenue, expenses, and overall financial performance, helping garage owners make informed decisions and manage cash flow effectively.

**Reporting and analytics** are integral to a GMS, offering valuable insights into various aspects of garage operations. The system generates detailed reports on key performance indicators such as service efficiency, customer satisfaction, and revenue trends. These analytics help identify operational strengths and areas for improvement, enabling data-driven decision-making that supports business growth and optimization.

Furthermore, many modern GMS platforms provide **mobile access** and **customer portals**, enhancing overall convenience and accessibility. Mobile access allows staff to manage operations from anywhere, improving flexibility and responsiveness. Customer portals, on the other hand, enable clients to track the status of their service requests, view their service history, and make payments online, thereby enhancing their overall experience and engagement with the garage.

In summary, a garage management system is a powerful tool that integrates various operational functions into a single, cohesive platform. By enhancing appointment scheduling, work order management, inventory control, customer relationship management, billing, and reporting, a GMS significantly improves the efficiency and effectiveness of auto repair shops and garages. It not only streamlines daily operations but also provides valuable insights and tools to drive business growth, improve customer satisfaction, and ensure long-term success.

Salesforce

**Introduction:**

Salesforce is a prominent cloud-based customer relationship management (CRM) platform designed to help businesses streamline and optimize their customer interactions. Founded in 1999 by Marc Benioff, Salesforce has evolved into a comprehensive suite of tools that support sales, customer service, marketing, commerce, and analytics. Its core products include Sales Cloud, Service Cloud, Marketing Cloud, and Commerce Cloud, among others. By providing a unified platform that centralizes customer data and automates various processes, Salesforce enables organizations to improve customer satisfaction, drive sales growth, and enhance overall operational efficiency. Its extensive ecosystem, including the AppExchange marketplace, allows for further customization and integration with other business applications, making it a versatile solution for companies of all sizes.

**What Is Salesforce?**

Salesforce is a leading cloud-based customer relationship management (CRM) platform that helps businesses manage their sales, customer service, marketing, and other critical functions. It offers a variety of tools and services, including Sales Cloud for managing sales operations, Service Cloud for customer support, Marketing Cloud for digital marketing automation, and Commerce Cloud for e-commerce solutions. Additionally, Salesforce provides a robust analytics platform and a marketplace called AppExchange for third-party integrations and applications. By centralizing customer information and automating various processes, Salesforce enables companies to enhance customer relationships, streamline workflows, and improve overall productivity.

something like this:

<https://youtu.be/r9EX3lGde5k>

**1. Introduction:**

This document outlines the process of designing and implementing a Garage Management System using Salesforce. The goal is to streamline garage operations and enhance customer satisfaction by managing customer information, appointments, service records, billing details, and feedback efficiently.

**1. Create a Salesforce Developer Account**

**Step-by-Step Process:**

**1. Sign Up:**

- Visit [Salesforce Developer Signup]

Go to <https://developer.salesforce.com/signup>

- Fill out the sign-up form with the following details:

- First Name and Last Name

- Email Address

- Role: Select “Developer”

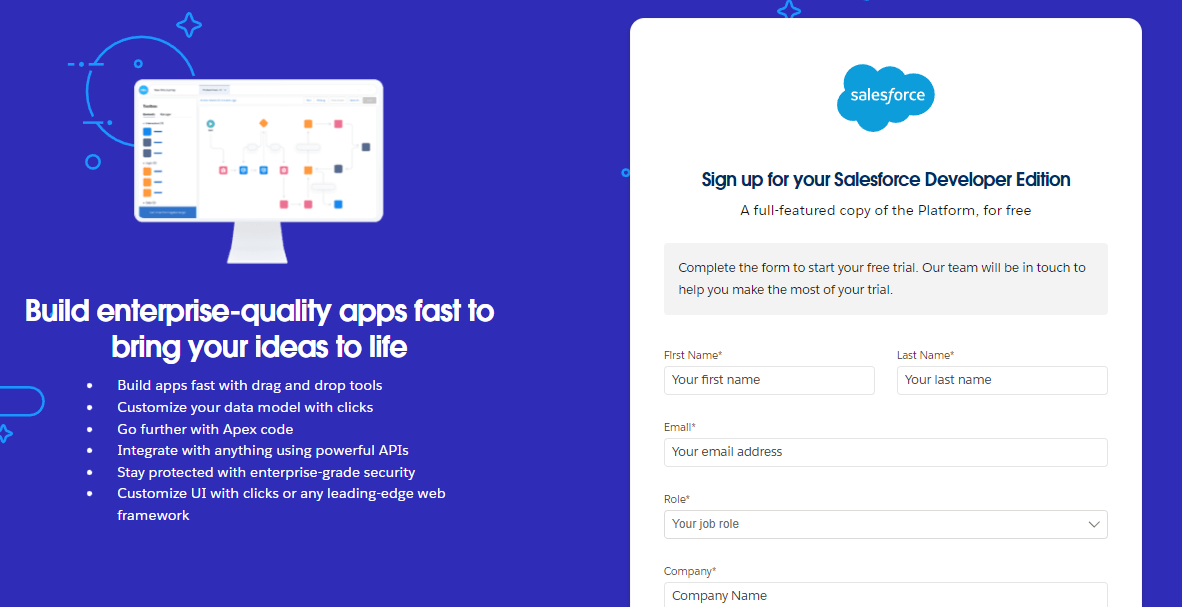
- Company: Enter your college or organization name

- Country: Select your country (e.g., India)

- Postal Code: Enter your postal code

- Username: Create a unique username in the format : username@organization.com

- Click Sign Me Up to submit the form.



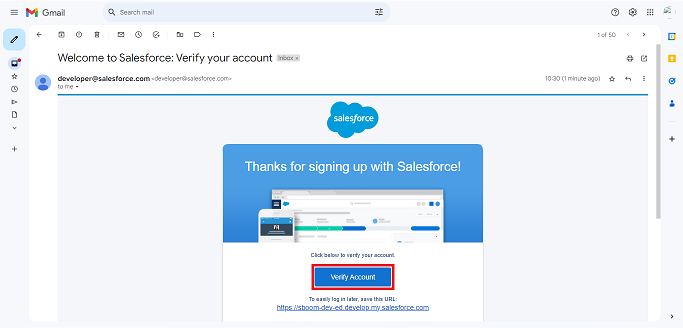
**2. Account Activation:**

- Check your email inbox for an activation email from Salesforce.

- Click on the Verify Account link.

- Set a password and answer the security question.

- After changing your password, you will be redirected to the Salesforce setup page.



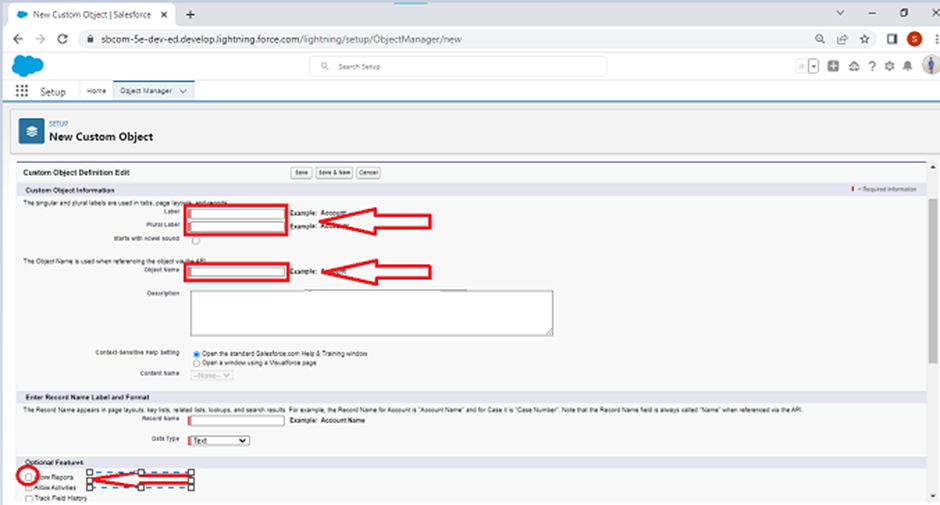
**2. Create Custom Objects**

Custom objects in Salesforce act like database tables where you can store specific information related to your business. Here’s how to create them:

Steps to Create Custom Objects:

**1. Navigate to Object Manager:**

- From the Salesforce Setup menu (click on the gear icon), select Object Manager



**2. Create an Object:**

- Click Create and choose Custom Object.

- Fill in the object details for each type:

- Customer Details

- Appointment

- Service Records

- Billing Details and Feedback

- For each object, set the following:

- Label: Name of the object (e.g., Customer Details)

- Plural Label: Plural form (e.g., Customer Details)

- Record Name: Define the primary field for records (e.g., Customer Name for Customer Details)

- Data Type: Choose appropriate types (e.g., Text, Auto Number)

- Enable options like Allow Reports, Track Field History, and \*\*Allow Search.

- Click Save to create the object.

**3. Create Custom Tabs**

Tabs provide a user interface for accessing custom objects and records.

Steps to Create Custom Tabs:



**1. Navigate to Tabs:**

- Go to Setup → Tabs → New (under Custom Object Tabs).

**2. Create a New Tab:**

- Select the object (e.g., Customer Details) from the list.

- Choose a tab style and icon to represent the object.

- Click Next, configure visibility settings for profiles, and click Save.

**3. Repeat for Remaining Objects:**

- Create tabs for Appointments, Service Records, and Billing Details and Feedback following similar steps.

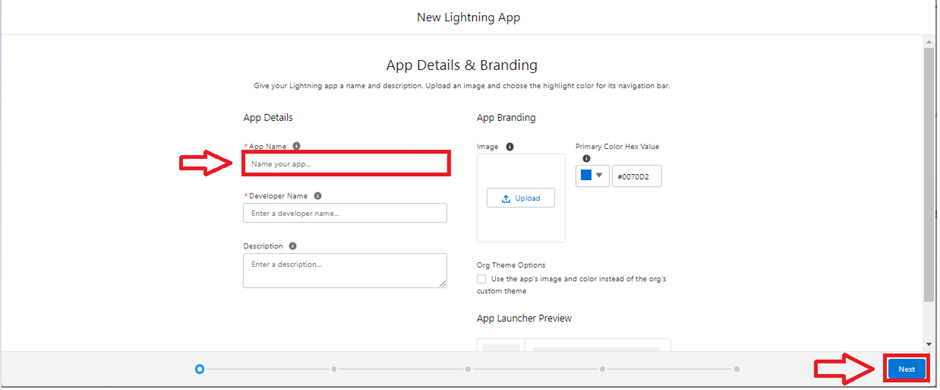
**4. Create a Lightning App**

A Lightning App bundles objects, tabs, and other items together for user access.

Steps to Create a Lightning App:

**1. Navigate to App Manager:**

- Go to Setup → App Manager → New Lightning App



**2. Configure the App:**

- App Details: Name your app (e.g., Garage Management Application).

- App Options: Choose default settings or customize as needed.

- Navigation Items: Add tabs for custom objects (Customer Details, Appointments, etc.).

-Utility Items: Configure utility bar settings if required.

- User Profiles: Assign the app to user profiles (e.g., System Administrator).

**3. Save and Finish:**

- Review the settings and click Save & Finish to complete the app creation.

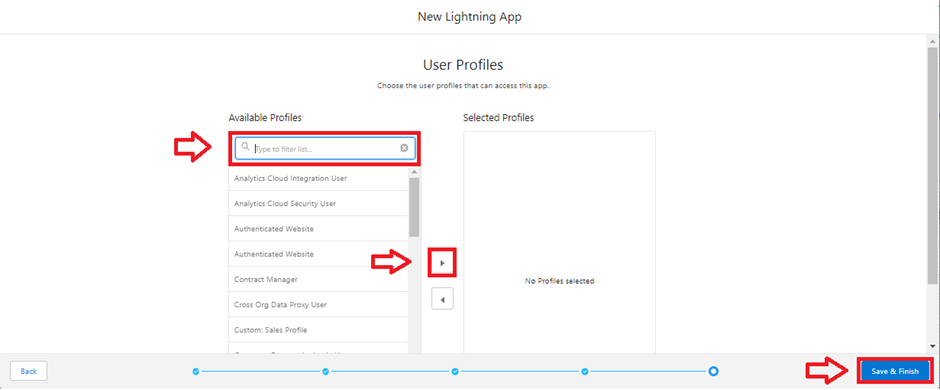
**4. Define and Create Custom Fields**

Fields represent the data points stored in each object. Custom fields are tailored to your specific needs.

Steps to Create Custom Fields:

**1. Navigate to Fields & Relationships:**

- Go to Setup → Object Manager → Select the Object (e.g., Customer Details) → Fields & Relationships → New.



**2. Configure the App:**

- App Details: Name your app (e.g., Garage Management Application).

- App Options: Choose default settings or customize as needed.

- Navigation Items: Add tabs for custom objects (Customer Details, Appointments, etc.).

- Utility Items: Configure utility bar settings if required.

- User Profiles: Assign the app to user profiles (e.g., System Administrator).

**3. Save and Finish**:

- Review the settings and click Save & Finish to complete the app creation.

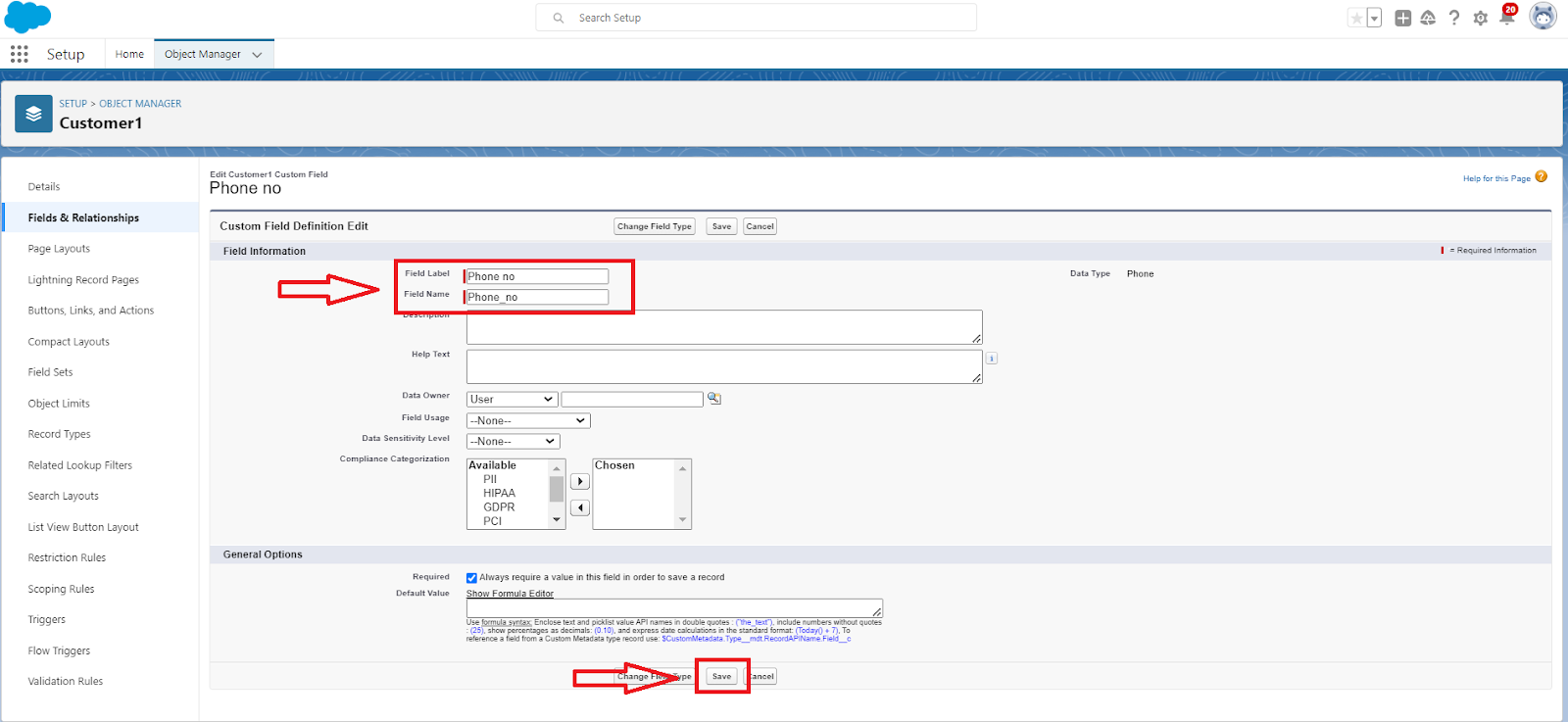
**5. Define and Create Custom Fields**

Fields represent the data points stored in each object. Custom fields are tailored to your specific needs.

Steps to Create Custom Fields:

**1. Navigate to Fields & Relationships:**

- Go to Setup → Object Manager → Select the Object (e.g., Customer Details) → Fields & Relationships → New.



**2. Create Field Types:**

- Text: For fields like Phone Number or Vehicle Number Plate.

- Date: For fields like Appointment Date.

- Currency: For fields like Service Amount.

- Checkbox: For fields like Maintenance Service, Repairs.

- Picklist: For fields like Service Status or Payment Status.

- Formula: For calculated fields like Service Date (use the formula function to calculate dates based on other fields).

**3. Configure Field Properties:**

- Set field label, data type, and properties such as required, unique, or read-only as needed.

- Click Next, configure field-level security and page layouts, and click Save.

**6. Create Lookup Relationships**

Lookup relationships link records from different objects, allowing for data integration. Steps to Create Lookup Relationships:

**1. Navigate to Fields & Relationships:**

- Go to Setup → Object Manager → Select the Object (e.g., Appointment) → Fields & Relationships→ New.

**2. Create a Lookup Relationship:**

- Select Lookup Relationship as the field type.

- Choose the related object (e.g., Customer Details for Appointments).

- Configure the relationship properties and filters.

- Click Save to establish the relationship.

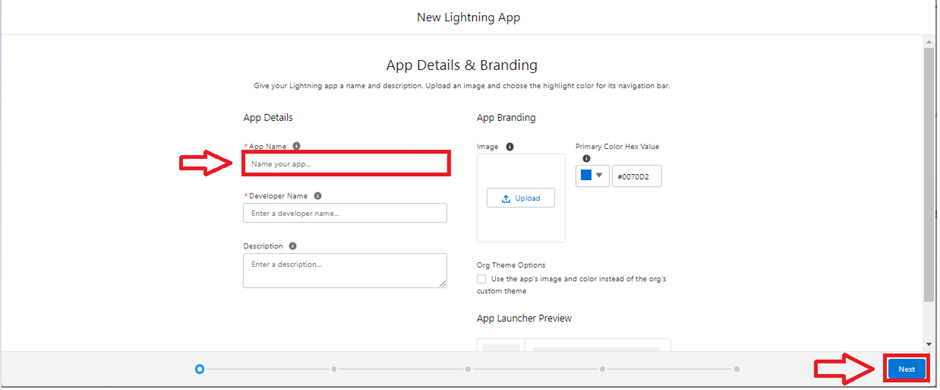
**3. Repeat for Related Objects:**

- Create lookup relationships for other objects (e.g., linking Service Records to Appointments).

**7. Define Validation Rules**

Validation rules ensure data integrity by enforcing certain criteria.

Steps to Create Validation Rules:



**1. Navigate to Validation Rules:**

- Go to Setup → Object Manager → Select the Object (e.g.,

Appointment) → Validation Rules → New.

**2. Define Rule Criteria:**

- Enter the rule name, description, and criteria for validation.

- Use formulas to define the conditions that must be met for the data to be valid.

-Enter error messages to display when validation fails.

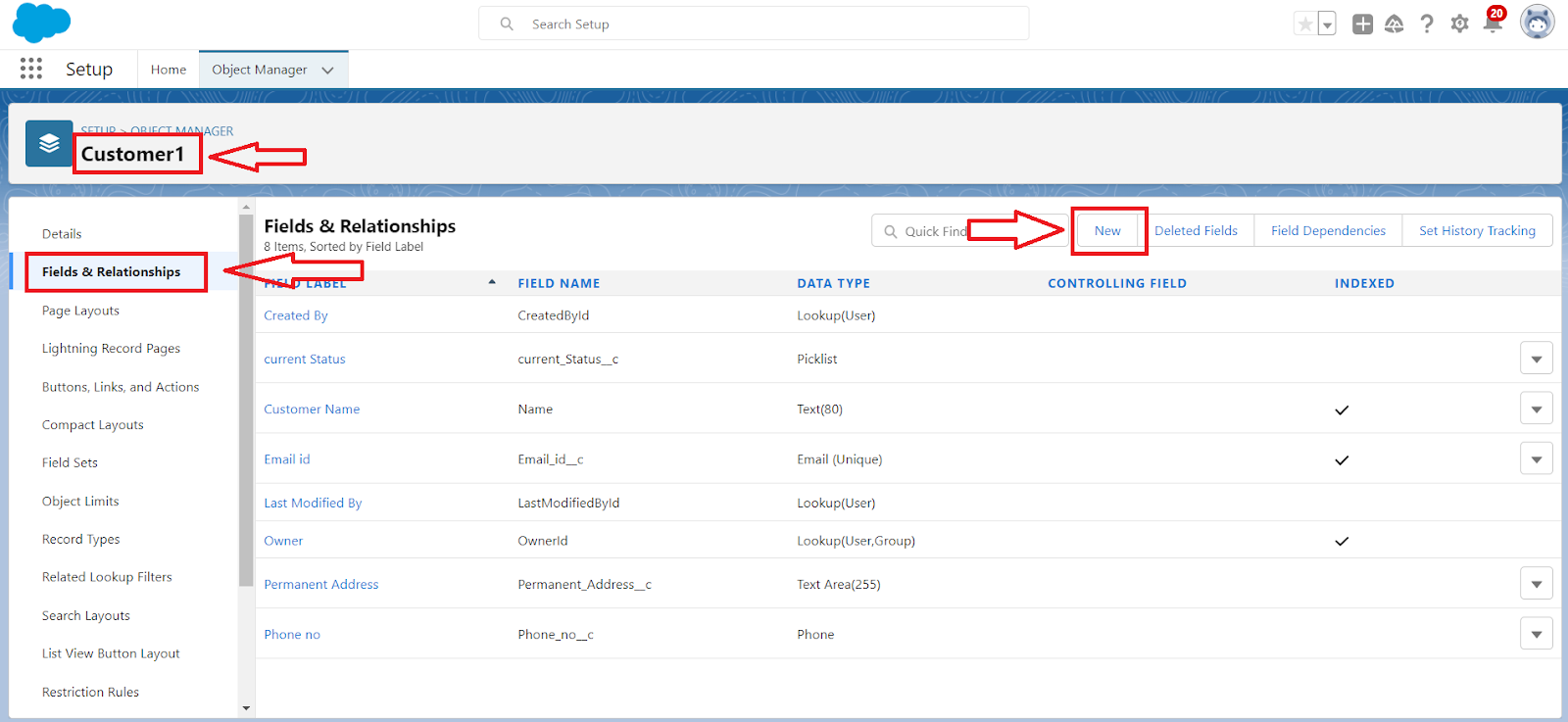
**3. Save and Test Rules:**

- Click Save to activate the rule.

- Test the rule by entering data into the system to ensure it works as intended.

**8. Configure Reports and Dashboards**

Reports and dashboards provide insights and visual representations of data.



Steps to Configure Reports:

**1. Navigate to Reports:**

- Go to Setup → Reports & Dashboards → Reports → New Report.

**2. Create a Report:**

- Select the report type and configure filters and columns.

- Save and run the report to view data.

**3. Create a Dashboard:**

- Go to Setup → Reports & Dashboards → Dashboards → New Dashboard.

- Add components such as charts and graphs to visualize report data.

- Configure dashboard settings and share with users as needed.

**9. Set Up Workflow Rules and Automation**

Automation helps streamline repetitive tasks and processes.

Steps to Set Up Workflow Rules:

**1. Navigate to Workflow Rules:**

- Go to Setup → Workflow Rules → New Rule.

**2. Define Rule Criteria:**

- Choose the object and set criteria for triggering the workflow.

- Define actions such as email alerts, field updates, or task creation.

**3. Activate Workflow Rule:**

- Click Save and Activate to enable the rule.

Steps to Use Process Builder:

**1. Navigate to Process Builder**:

- Go toSetup → Process Builder → New Process.

**2. Define Process**:

- Set up process criteria and actions using a visual interface.

- Save and activate the process.

**10. Customize Page Layouts**

Page layouts determine how fields and related information are displayed on record pages.

Steps to Customize Page Layouts:

**1. Navigate to Page Layouts:**

- Go to Setup → Object Manager → Select the Object (e.g., Appointment) → Page Layouts.

**2. Edit Layout:**

- Drag and drop fields and sections to arrange the layout.

- Configure visibility settings and field placements.

- Click Save to apply the layout changes.

**11. Create and Configure Email Template**

Email templates help standardize communication for notifications and updates.

Steps to Create Email Templates:

**1. Navigate to Email Templates:**

- Go to Setup → Email → Classic Email Templates.

**2. Create Template:**

- Choose New Template, select type (e.g., HTML), and enter template details.

- Include merge fields to personalize emails.

- Save the template and test by sending a sample email.

**12. Ensure Security and Access Control**

Managing user access is crucial for protecting sensitive data and ensuring proper permissions.

Steps to Configure Security and Access Control:

**1. Navigate to Profiles:**

- Go to Setup → Profiles.

- Edit profiles to set permissions for objects and fields.

**2. Use Permission Sets:**

- Go to Setup → Permission Sets.

- Click New Permission Set to create a new set of permissions.

- Define the permissions needed (object access, field-level security, etc.).

- Assign the permission set to users or groups as necessary.

- Click Save to apply the permissions.

**3. Set Up Role Hierarchy:**

- Go to Setup → Roles → Role Hierarchy.

- Define roles that reflect your organization’s structure.

- Set up roles to control access and data visibility based on user roles.

- Assign users to roles to manage their access level accordingly.

**13. Develop Custom Lightning Components (Optional)**

Custom Lightning Components can enhance the user interface and add specific functionalities.

Steps to Develop Custom Lightning Components:

**1. Set Up Salesforce DX:**

- Install Salesforce CLI and configure Salesforce DX for development. - Create a new project using the Salesforce CLI.

**2**. **Create Lightning Component:**

**-** In your Salesforce DX project, use the command sfdx force:lightning:component:create to create a new component.

- Define the component’s functionality using HTML, JavaScript, and Apex (ifneeded).

**3. Deploy and Test**:

- Deploy the component to your Salesforce environment using Salesforce CLI.

- Test the component in your Salesforce org to ensure it meets requirements.

**14. Testing and Deployment**

Before going live, thoroughly test your system and deploy it to production.

Steps for Testing:

**1. Unit Testing**:

- Test individual components, objects, and fields to ensure they work as expected. - Use Salesforce’s built-in testing tools or create test cases to validate functionality.

**2. User Acceptance Testing (UAT):**

- Have end-users test the system in a sandbox environment.

- Collect feedback and make necessary adjustments based on user experiences.

**3. System Testing:**

- Test the entire system, including integrations, automation, and user interfaces.

- Ensure that all components work together seamlessly.

Steps for Deployment:

**1. Prepare for Deployment**:

- Review deployment plans and ensure all components are ready for production.

- Use change sets or Salesforce DX for deployment.

**2. Deploy to Production:**

- Deploy the system using change sets or Salesforce CLI.

- Monitor the deployment process and address any issues that arise.

**3. Post-Deployment**:

- Verify that the system is functioning correctly in the production environment.

- Provide training and support to users as needed.

**15. Continuous Improvement and Maintenance**

After deployment, ongoing maintenance and improvements are essential to keep the system effective.

Steps for Continuous Improvement:

**1. Monitor System Performance:**

- Regularly check system performance and user feedback.

- Address any performance issues or bugs promptly.

**2. Update and Enhance Features**:

- Periodically review and update features based on evolving business needs.

- Implement new functionalities and enhancements as required.

**3. Provide User Support:**

- Offer training and support to users to ensure they can effectively use the system.

- Maintain a knowledge base or help resources for user assistance.

**4. Review Security:**

- Regularly review and update security settings and access controls.

- Ensure compliance with data protection regulations and organizational policies.

This document provides a detailed approach to creating a Garage

Management System in Salesforce. By following these steps, you can

effectively manage garage operations and improve overall efficiency

and customer satisfaction.

**Conclusion:**

The Garage Management System project has been a comprehensive and enriching experience, demonstrating the power of Salesforce in streamlining business processes and enhancing customer satisfaction. Through this project, we successfully designed and implemented a customized Salesforce solution to manage: - Customer information and appointments - Service records and billing details - Feedback and ratings Utilizing Salesforce's robust features, including custom objects, fields, validation rules, duplicate rules, profiles, and flows, we created a seamless and efficient system for garage owners to manage their business. The Garage Management System showcases our ability to apply Salesforce skills to real-world business challenges, driving process automation, data analysis, and customer relationship management. This project has not only honed our technical skills but also fostered collaboration, creativity, and problem-solving. As we conclude this project, we are confident that the Garage Management System will serve as a valuable asset for garage owners, empowering them to deliver exceptional customer experiences, drive business growth, and stay ahead in the competitive market. Thank you to our instructors, mentors, and team members for their guidance, support, and dedication throughout this project.

THANK YOU