APPLICATION TO MAKE THE GAS FILLING STATION EASY USING CRM

Abstract

The abstract serves as a high-level summary of the entire project. It introduces the core problem: the need for efficient management of customer interactions and operations at a gas filling station. The solution presented is the development of a custom application on the **Salesforce CRM** platform. The abstract highlights the key components of this solution, including the creation of custom objects to store essential business data (Buyers, Fuel Details, Gas Stations, and Suppliers). It also touches on the user-centric design through page layouts and security features like user roles, profiles, and permissions. The main takeaway is that the application simplifies daily tasks for employees and improves the customer experience.

Objective

The objectives section defines the project's goals with a clear vision to improve both customer service and store management. The core aims are:

- **Streamline Operations:** Consolidate all data—customer, fuel, supplier, and station details—into a single platform to reduce manual work.
- **Enhance Customer Experience:** Maintain accurate records of customers, their vehicles, and receipts for better service and transparency.
- **Role-Based Access:** Implement secure user roles (Manager, Executive, Salesperson) to control who can view or modify data.
- **Improve Efficiency:** Automate routine tasks, like managing buyer details and payment transactions, to minimize human error.
- **Ensure Data Security:** Protect sensitive data by enforcing strong policies and permissions.
- **Enable Scalability:** Build a platform that can easily support future additions, such as new reports or dashboards.

Technology Description

- **Salesforce:** The fundamental cloud-based CRM platform used to manage customer relationships, streamline operations, and store data securely.

- **Custom Objects:** These are user-defined database tables created to store information unique to the gas station business, such as Buyers, Fuel Details, Gas Stations, and Suppliers.
- **Tabs:** User interface elements that provide quick access to the custom objects and other applications.
- **Lightning App:** A collection of related items, including objects and tabs, that creates a unified, user-friendly interface for employees.
- **Page Layouts:** Tools used to organize the fields and sections on a record page, making data entry more structured and intuitive.
- **Profiles & Roles:** Profiles define what objects, fields, and features a user can access, while roles control record-level access based on the organizational hierarchy.
- **Users:** Individual accounts created for each person who needs to log in and use the system, with permissions and roles assigned accordingly.
- **Permission Sets:** A way to grant additional access rights to users without changing their profile, providing more granular control.

Project Implementation Steps

Step 1: Get Started with a Salesforce Developer Org
• Sign up for a Developer Org with your details using
https://developer.salesforce.com/signup
• Verify your email, set a password, and access the setup page.

Activation through the salesforce developer mail
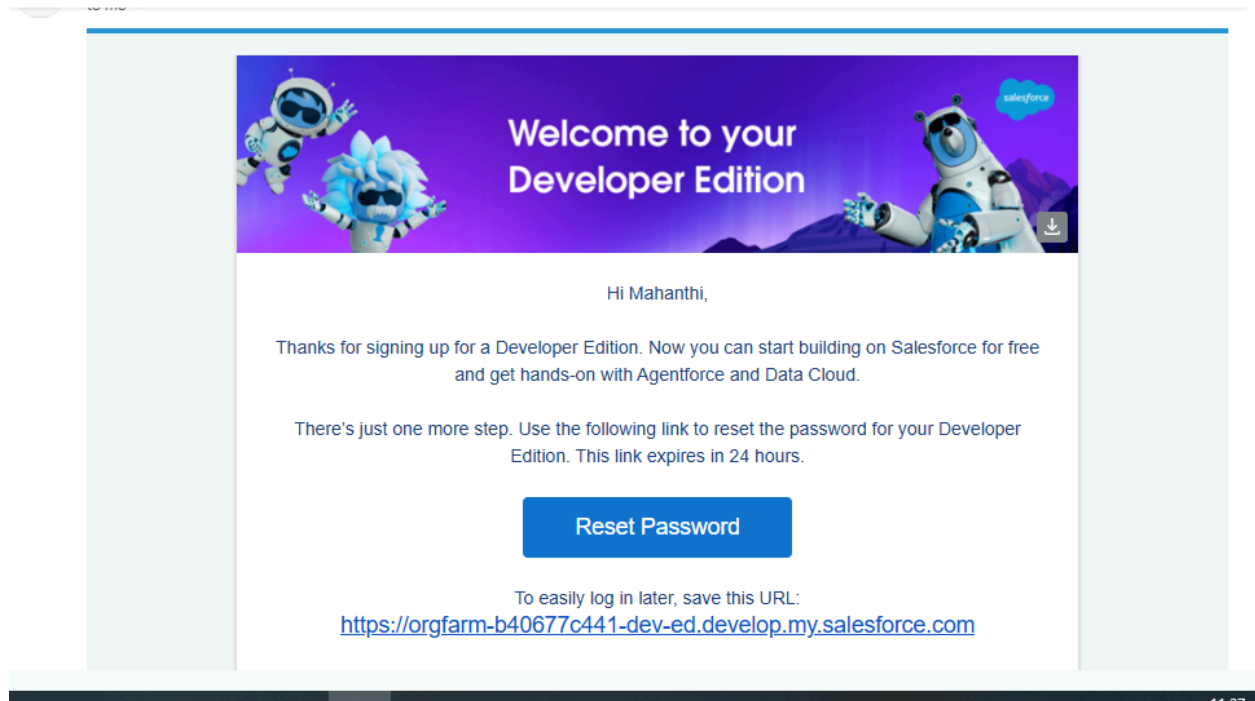
2.Create Custom Objects :

The next step is to create the custom objects that will store the core data for the application.

Four custom objects:

1• Buyer__c: Stores customer details and manages their fuel purchases.
2.Fuel_details__c: Records fuel transactions linking buyers, suppliers, and gas stations. •
3. Gas_Station__c: Maintains gas station information, fuel availability, and usage.
4.Supplier__c: Tracks suppliers and the total fuel supplied to stations.
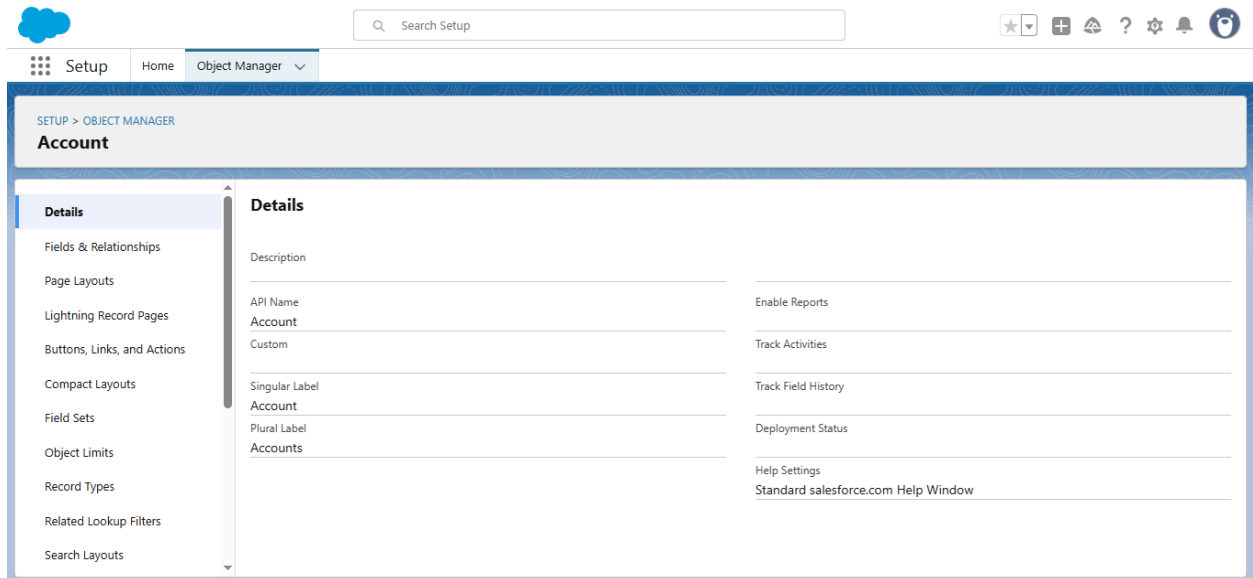
3. Custom Tabs:

 They are a key feature of the Salesforce application's user interface. They are designed to give users easy access and a way to navigate to the various custom objects created for the project, such as "Buyers," "Fuel Details," "Gas Stations," and "Suppliers".
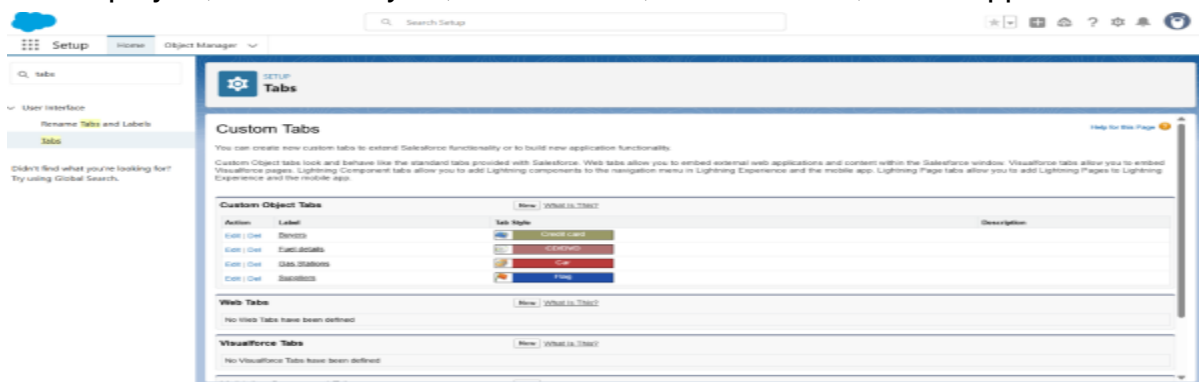
4.Create Lightning App:

- **App Creation:** A Lightning App named "**GAS STATION**" was created to manage all CRM operations.
- **Configuration:** The app was configured with the necessary custom objects to ensure smooth usage.

- **Profile Assignment:** The app was assigned to the **System Administrator** profile, giving administrators access to the application.
- **Objects and Fields:** The app incorporates the following custom objects, which are populated with necessary fields to manage the Gas Station CRM:
    a. **Buyer**
    b. **Fuel Details**
    c. **Gas Station**
    d. **Supplier**
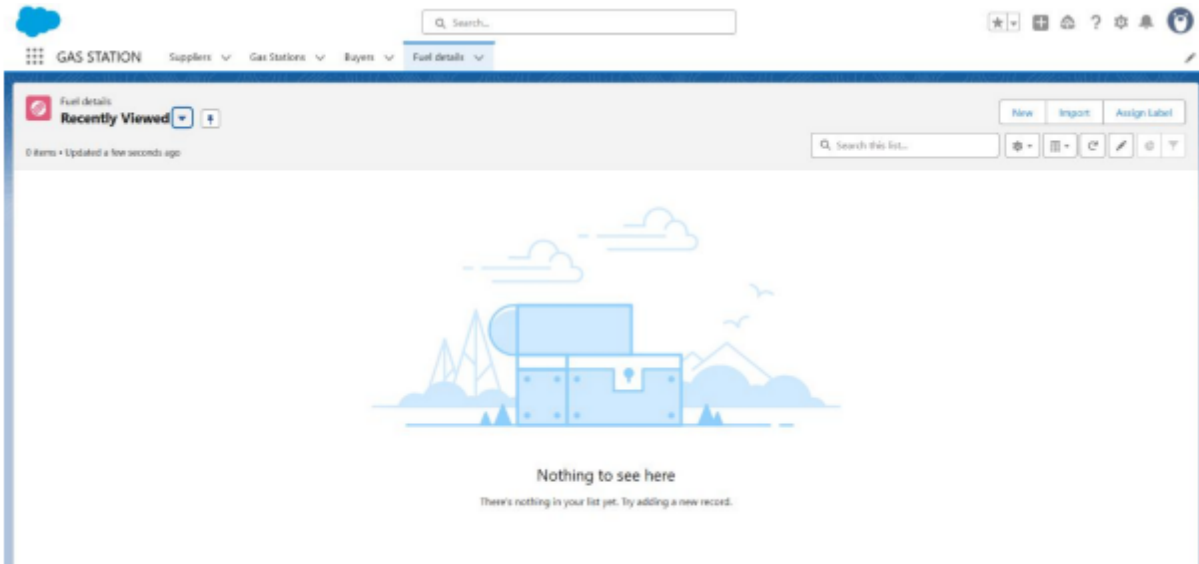
5. Fields to be created:

1. Buyer Object:
- First Name
- Last Name
- Email
- Phone Number
- Vehicle Number
- Vehicle Model

2. Fuel Details Object:
- Fuel Type (e.g., Petrol, Diesel, CNG)
- Price Per Liter
- Quantity (in Liters)
- Total Amount
- Date and Time of Transaction
- Payment Method (e.g., Cash, Credit Card, UPI)
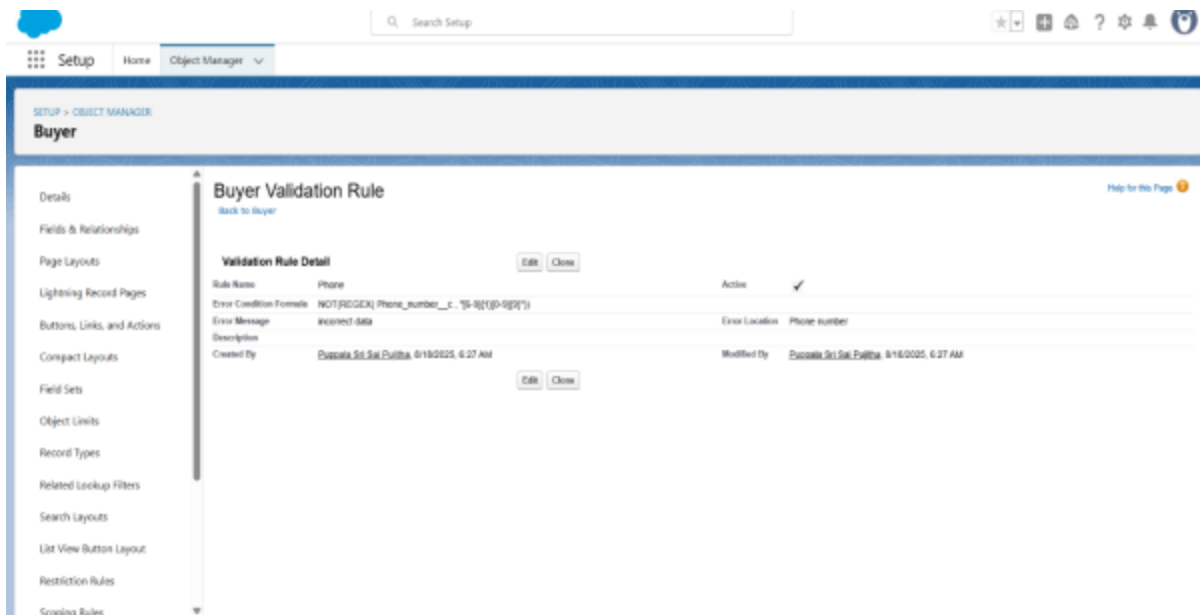
3. Gas Station Object:
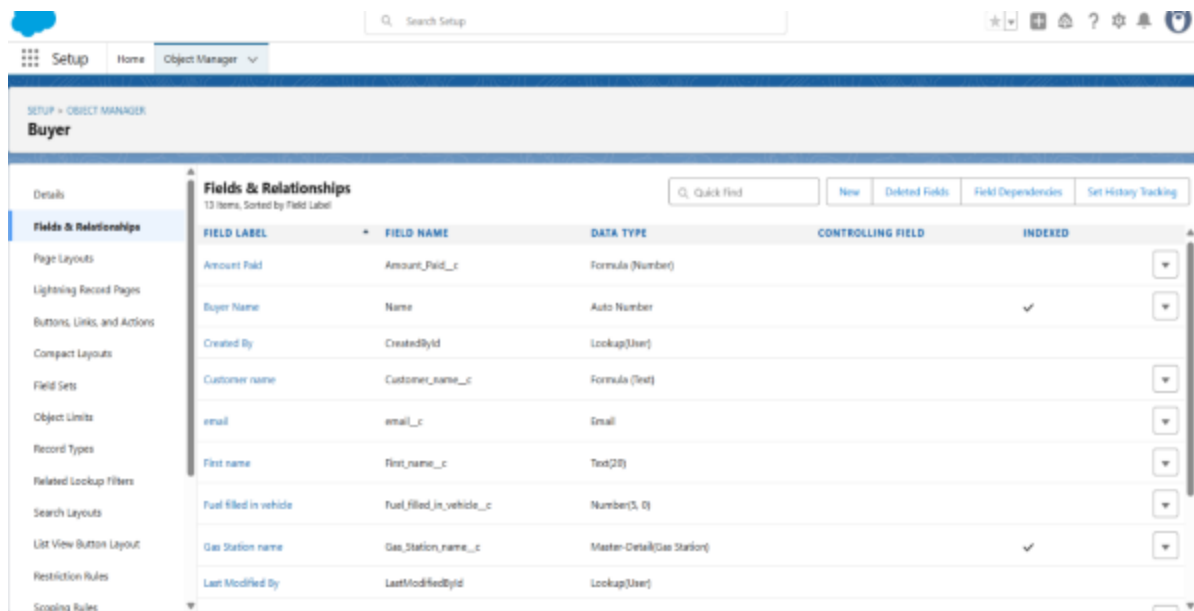
- Name
- Address
- Contact Person
- Phone Number

4. Supplier Object:

- Name
- Contact Person
- Phone Number
- Address
- Contract Expiration Date

Fuel details

Fields & Relationships
6 items, Sorted by Field Label

New | Deleted Fields | Field Dependencies | Set History Tracking

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED	
Created By	CreatedById	Lookup(User)			
Fuel details id	Name	Auto Number		✓	▼
Fuel Supplied	Fuel_Supplied__c	Number(5, 0)			▼
Gas Station	Gas_Station__c	Master-Detail(Gas Station)		✓	▼
Last Modified By	LastModifiedById	Lookup(User)			
Supplier Name	Supplier_Name__c	Master-Detail(Supplier)		✓	▼

Sidebar: 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, Scoping Rules

---

Gas Station

Fields & Relationships
8 items, Sorted by Field Label

New | Deleted Fields | Field Dependencies | Set History Tracking

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED	
Created By	CreatedById	Lookup(User)			
Fuel Available in bunk	Fuel_Available_in_bunk__c	Formula (Number)			▼
Fuel Price/litre	Fuel_Price_litre__c	Number(5, 0)			▼
Fuel supplied to bunk	Fuel_supplied_to_bunk__c	Roll-Up Summary (SUM Fuel details)			▼
Fuel used	Fuel_used__c	Roll-Up Summary (SUM Buyer)			▼
Gas Station	Name	Auto Number		✓	▼
Last Modified By	LastModifiedById	Lookup(User)			
Owner	OwnerId	Lookup(User,Group)		✓	

Sidebar: 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, Scoping Rules

---

Supplier

Fields & Relationships
5 items, Sorted by Field Label

New | Deleted Fields | Field Dependencies | Set History Tracking

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED	
Created By	CreatedById	Lookup(User)			
Last Modified By	LastModifiedById	Lookup(User)			
Owner	OwnerId	Lookup(User,Group)		✓	
sum of fuel supplied	sum_of_Fuel_supplied__c	Roll-Up Summary (SUM Fuel details)			▼
Supplier Name	Name	Text(80)		✓	▼

Sidebar: 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

6.Create Page Layout for Objects:

- **Select the Object:** Navigate to the Object Manager in Salesforce Setup and choose the object you want to create a page layout for (e.g., Buyer, Fuel Details, Gas Station, Supplier).
- **Access Page Layouts:** From the object's details page, click on "Page Layouts."
- **Create or Edit:** You can either create a new page layout or edit an existing one.
- **Arrange Fields:** Drag and drop fields, sections, and related lists from the palette onto the canvas to organize the layout. The document suggests grouping related fields into sections for a cleaner look.
- **Save and Assign:** After arranging the fields, save the layout and then assign it to the relevant user profiles (e.g., Sales Executive, Sales Person, System Administrator) so they can view and use it.
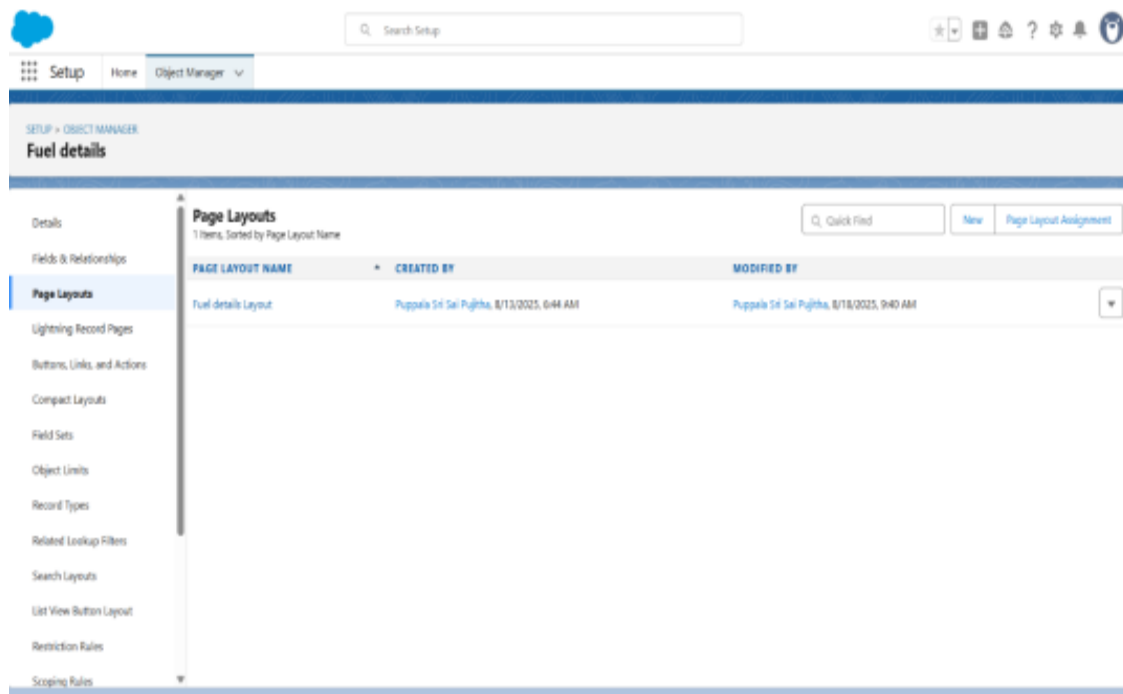
7.Create profiles:

- Click the **Gear (Setup)** icon (top-right) → **Setup**.
- In **Quick Find**, type **Profiles** and select **Profiles** under **Users**.
- Click **New Profile**.
- In the **New Profile** dialog/page, choose an existing profile to clone (pick a base like **Standard User**).
- Enter the **Profile Name** (e.g., `Manager Profile`) and any description.

- Click **Save**.
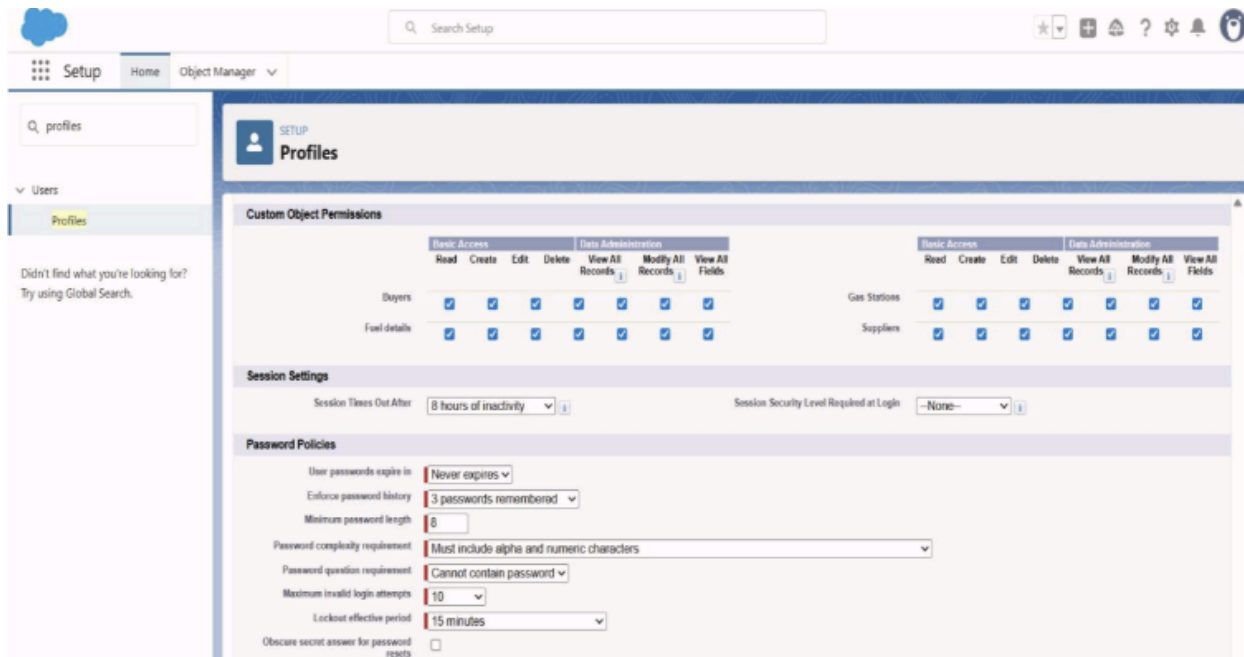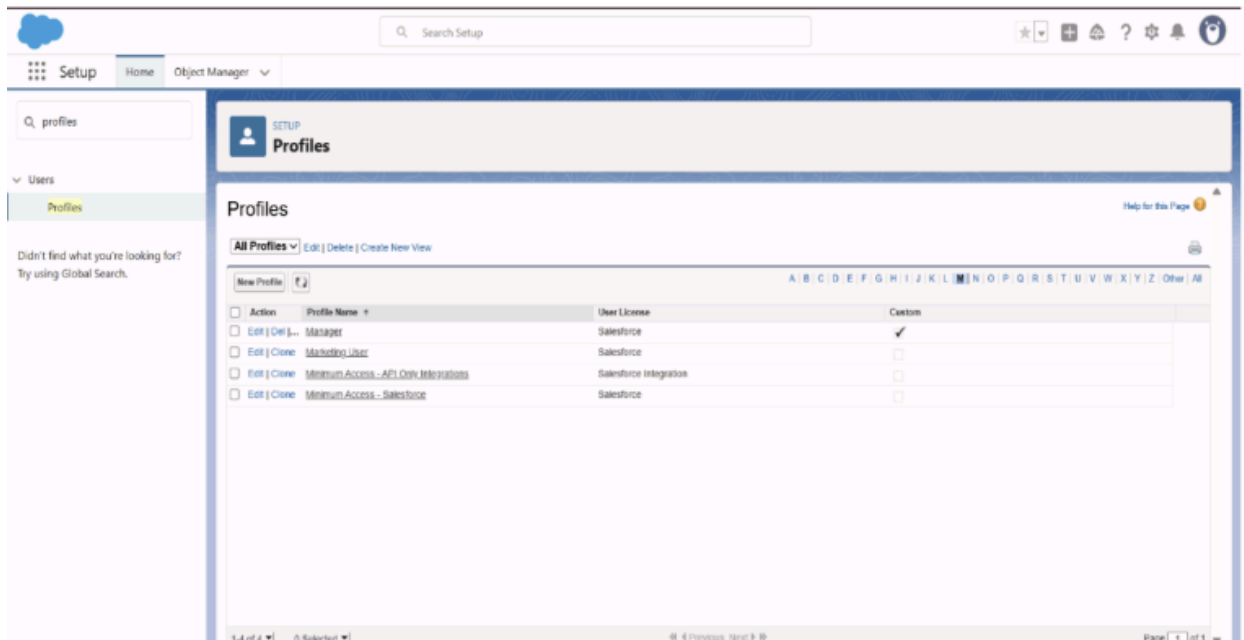
7. Creating Roles & Hierarchy:

- Go to **Setup** → search **Roles**.
- Click **Set Up Roles**.
- Click **Add Role** → name it **Manager** → Save.

- Under Manager, click **Add Role** → name it **Sales Executive** → Save.
- Under Sales Executive, click **Add Role** → name it **Sales Person** → Save.
- Go to **Users** → open a user → **Edit** → assign the correct Role → Save.
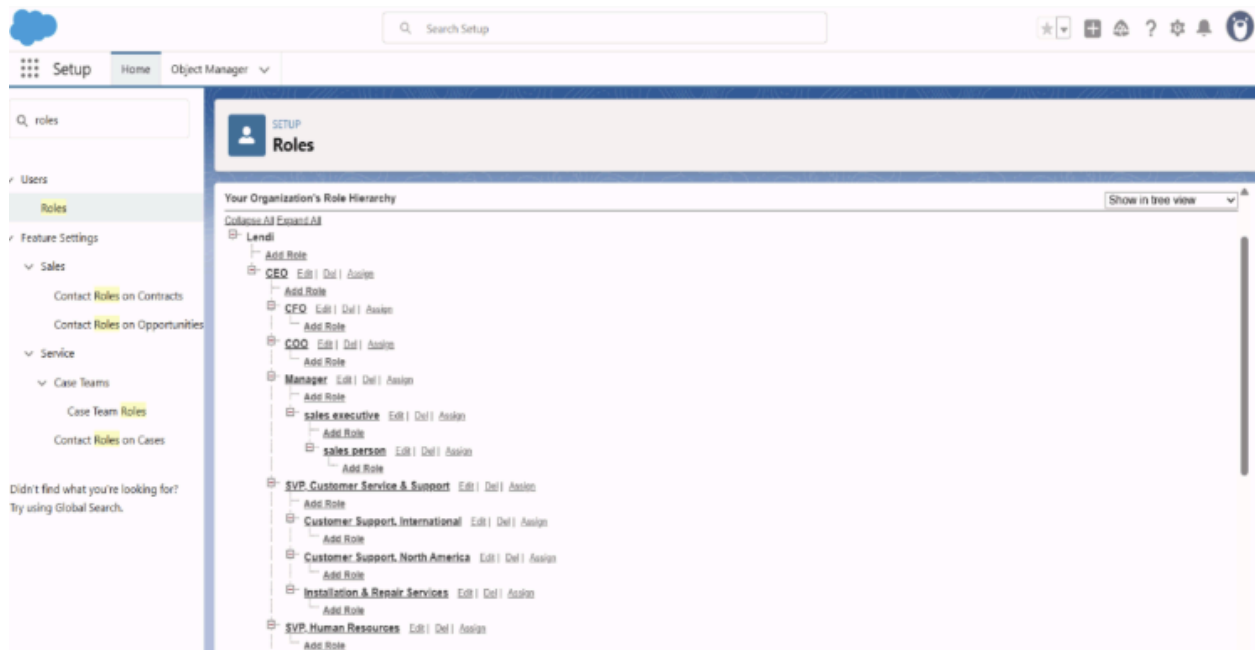
8. Creating users:

- Go to **Setup** → search **Users**.
- Click **Users** → then click **New User**.
- Fill in details: **First Name, Last Name, Email, Username** (must be in email)
-
- Select **Role** (Manager / Sales Executive / Sales Person).
- Select **Profile** (Manager Profile / Sales Executive Profile / Sales Person Profile).
- Choose **User License** (usually Salesforce).
- Tick **Generate new password** if you want Salesforce to email login details.
- Click **Save**.
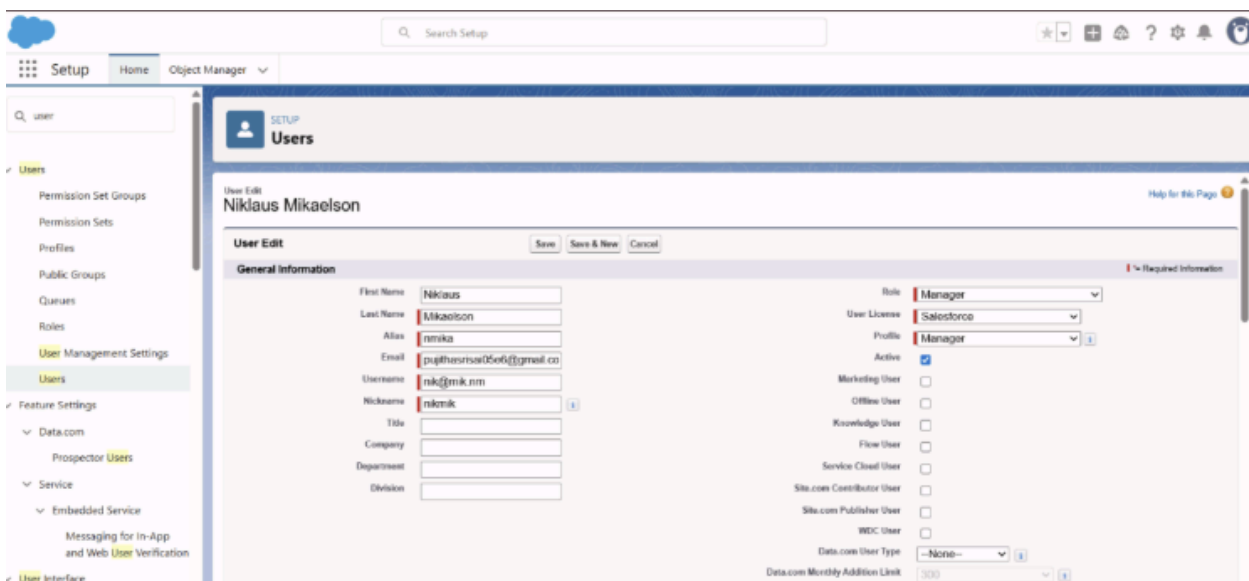
Created users the displayed

9. Create Permission Sets:

- Go to **Setup** → search **Permission Sets**.
- Click **Permission Sets** → then click **New**.
- Enter **Label** (e.g., "Report Access"), API Name fills automatically.
- Select a **User License** (or leave as "None" for all users).
- Click **Save**.
- Inside the Permission Set, set what you need:
  - **Object Settings** → adjust CRUD (Read, Create, Edit, Delete).
  - **Field Permissions** → mark fields Visible/Read Only.
  - **App Permissions** / **System Permissions** → check boxes (e.g., "Run Reports").
- Click **Manage Assignments** → **Add Assignments** → choose users → **Assign**.

10. OWD (Organization-Wide Defaults):

- Go to **Setup** → search **Sharing Settings**.
- Click **Sharing Settings**.
- In **Organization-Wide Defaults**, click **Edit**.
- For each object, set the **Default Internal Access** (and External if needed).
  Example for your Gas Station CRM:
    - **Buyer__c** → Private
    - **Fuel_Details__c** → Private
    - **Gas_Station__c** → Public Read Only
    - **Supplier__c** → Public Read Only
- Click **Save**.
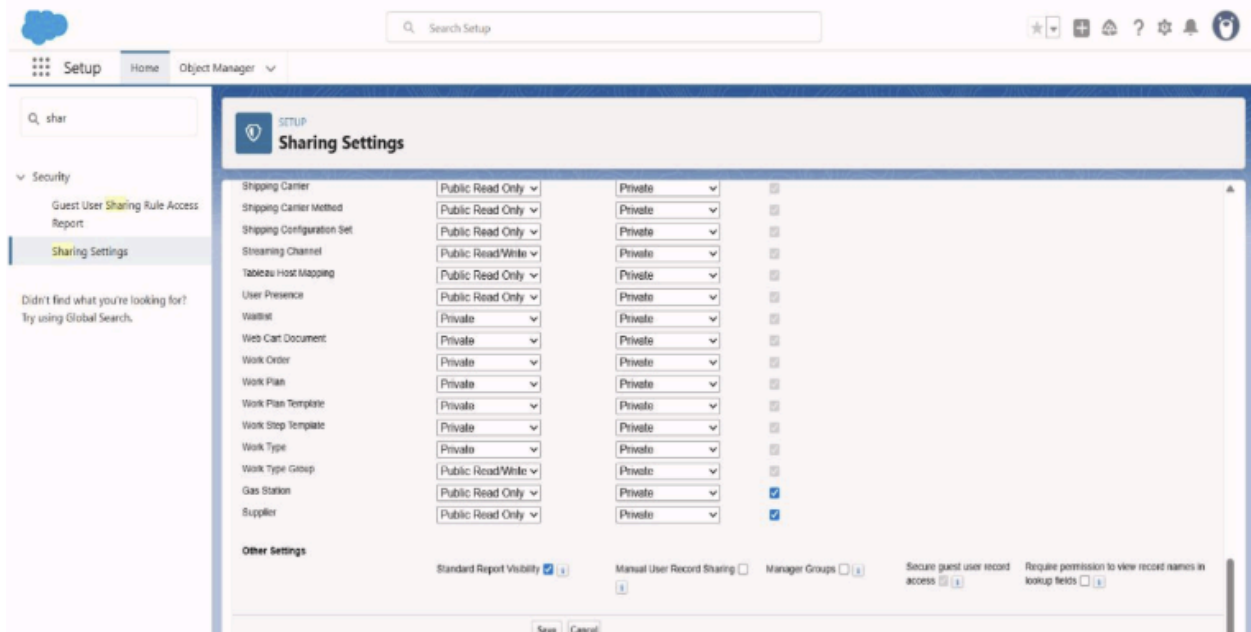
11. Create, View, and Delete Records:

Create a Record

1. Go to the **App Launcher** → select your app (e.g., *Gas Station*).
2. Click the **tab** (e.g., Buyer, Fuel Details, Gas Station, Supplier).
3. Click **New**.
4. Fill in the fields (Name, Phone, Amount Paid, etc.).
5. Click **Save**.

View a Record

1. Go to the object's **tab** (e.g., Buyer).
2. Select a **record** from the list view.
3. Record details open → check fields, related lists, and activities.

Delete a Record

1. Open the record you want to delete.
2. Click the **Dropdown / Delete** button (top-right).
3. Confirm deletion by clicking **Delete** again.
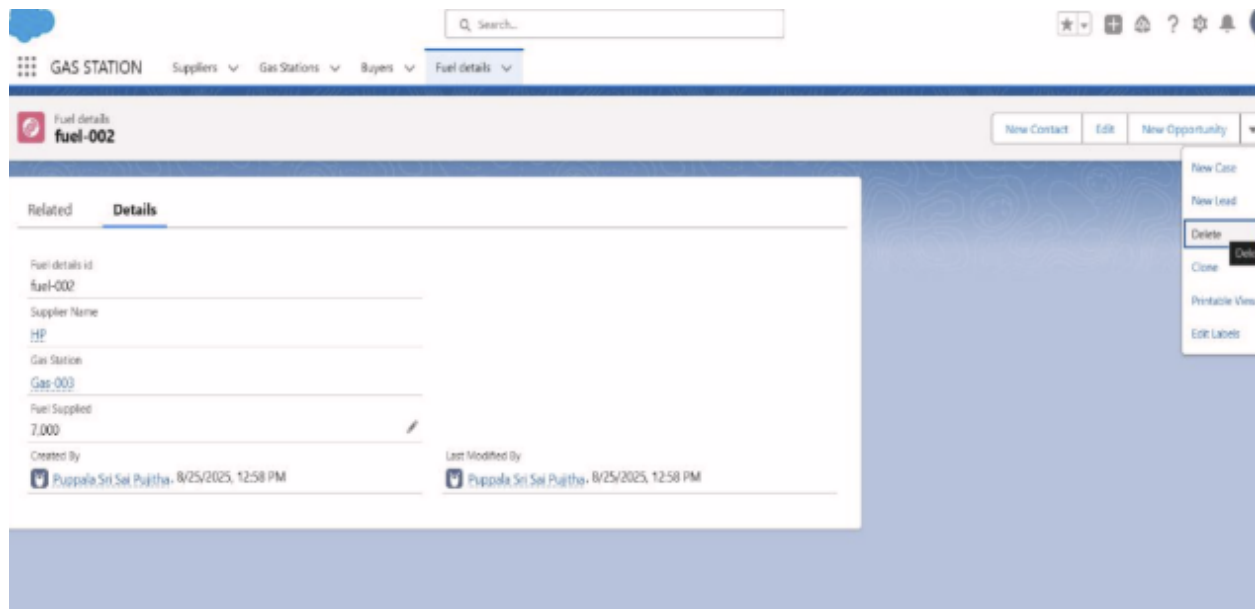
12. Creating Reports:

- Go to **App Launcher** → search and open **Reports**.
- Click **New Report**.
- Choose a **Report Type** (e.g., Buyer with Fuel Details, Gas Station, Supplier).
- Click **Continue**.
- In the **Report Builder**:
  - Drag and drop fields you want (e.g., Customer Name, Fuel Supplied, Amount Paid).
  - Add **Filters** (e.g., Date = THIS MONTH).
  - Add **Group Rows/Columns** if needed (e.g., group by Gas Station).
- Click **Run** to preview the report.
- Click **Save & Run**.
- Enter **Report Name** and **Folder** (e.g., "Monthly Fuel Usage Report").
- Click **Save**.

13. Creating Dashboards:

Dashboards visually display key metrics and report data in charts and graphs.
• In this project, dashboards were created to provide insights on fuel estimation, sales, and customer activity
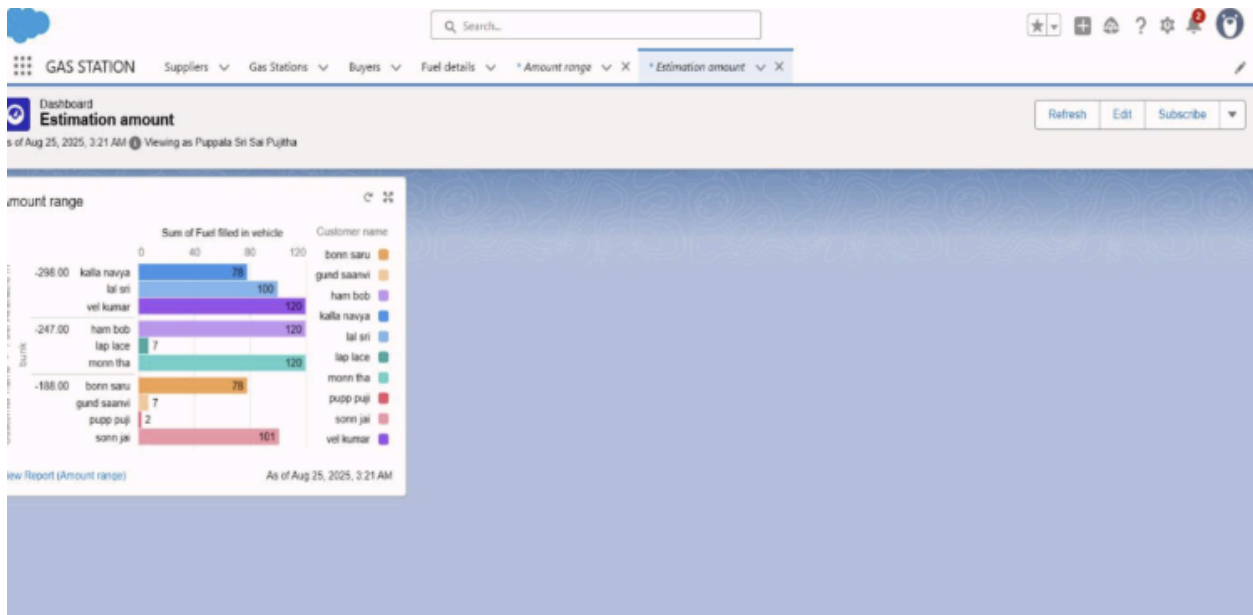
14. Creating Flow:
A record-triggered flow was created on the Buyer object.
• The flow triggers when a record is created or updated.

• It automatically sends an email receipt to the customer.
• The email includes Customer Name, Amount Paid, Vehicle Type, and Fuel Intake.

15. Creating Apex Triggers:

Apex triggers are custom code that execute before or after events on records.
Handler (FuelRecordHandler.apxc)
• beforeDeleteInfo → Stops deletion of Fuel Details if supplier qty > 500.
• beforeDeleteGas → Ensures Gas Station fuel price > 50 before insert.

File ▾ Edit ▾ Debug ▾ Test ▾ Workspace ▾ Help ▾ ＜ ＞

FuelRecordHandler.apxc ×

Code Coverage: None ▾   API Version 64 ▼   Go To

```apex
1 ▾ public class FuelRecordHandler {
2
3 ▾     public static void beforeDeleteInfo(list<Fuel_details__c> fuelList){
4
5           //fuelList = [select Id from Fuel_details__c];
6
7 ▾         for(Fuel_details__c ful : fuelList){
8
9 ▾             if(ful.Fuel_supplied__c > 500){
10
11                  ful.addError('you cannot delete the fuel details record because it is associated with supplier and Gas station records');
12
13              }
14
15          }
16
17      }
18
19 ▾     public static void beforeDeleteGas(list<Gas_Station__c> gasList){
20
```

Logs Tests Checkpoints Query Editor View State Progress Problems

User   Application   Operation   Time ▾   Status   Read   Size

Filter   Click here to filter the log list

File ▾ Edit ▾ Debug ▾ Test ▾ Workspace ▾ Help ▾ ＜ ＞

FuelRecordHandler.apxc ×   beforeDelete.apxt ×

Code Coverage: None ▾   API Version 64 ▼   Go

```apex
1 ▾ trigger beforeDelete on Fuel_details__c  (before Delete) {
2
3 ▾     if(trigger.isbefore && trigger.isDelete){
4
5           FuelRecordHandler.beforeDeleteInfo(trigger.old);
6
7      }
8
9 }
```

Logs Tests Checkpoints Query Editor View State Progress Problems

User   Application   Operation   Time ▾   Status   Read   Size

---

File ▾ Edit ▾ Debug ▾ Test ▾ Workspace ▾ Help ▾ ＜ ＞

FuelRecordHandler.apxc ×   beforeDelete.apxt ×   beforeInsert.apxt ×

Code Coverage: None ▾   API Version: 64 ▼   Go

```apex
1 ▾ trigger beforeInsert on Gas_Station__c (before insert ) {
2
3 ▾     if(trigger.isbefore && trigger.isinsert){
4
5           FuelRecordHandler.beforeDeleteGas(trigger.new);
6
7      }
8 }
```

Logs Tests Checkpoints Query Editor View State Progress Problems

User   Application   Operation   Time ▾   Status   Read   Size

Feature Enhancements for Gas Station App:

1.Customer Management

- Maintain a database of regular and new customers.
- Track customer visits, fuel purchases, and preferences.
- Store contact information for promotions and loyalty programs.

2. Fuel Inventory Management

- Monitor real-time fuel stock levels.
- Set alerts for low inventory to prevent shortages.
- Manage different fuel types (Petrol, Diesel, CNG, etc.).

3. Transaction & Billing Management

- Automatic billing for each fuel purchase.
- Generate invoices and receipts for customers.
- Record payment methods (cash, card, digital payments).

4. Loyalty & Reward Programs

- Implement points or cashback for frequent customers.
- Track redemption of loyalty rewards.
- Send notifications for offers or discounts.

5. Employee Management

- Track employee shifts and attendance.
- Assign roles for cashier, supervisor, or manager.
- Monitor employee performance and sales.

6. Reports & Analytics

- Generate daily, weekly, and monthly sales reports.
- Analyze fuel consumption patterns.
- Identify peak hours and high-demand fuel types.

7. Notification & Alerts

- Send SMS/email notifications for promotions or updates.
- Alert managers to stock shortages or irregular transactions.

8. Integration

- Integrate with digital payment systems for faster transactions.
- Optionally integrate with government fuel monitoring systems.

9. Security & Access Control

- Role-based access for employees (admin, cashier, supervisor).
- Protect sensitive customer and financial data.

Additional screenshots:
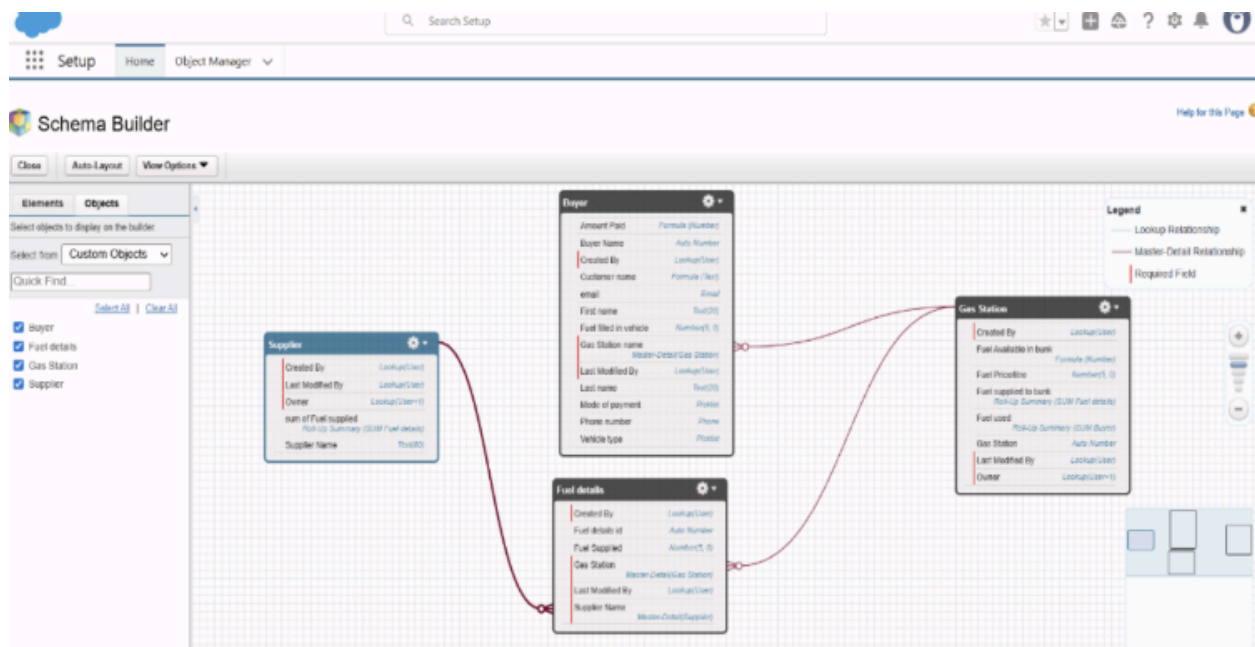
Setup | Home | Object Manager ⌄

Q tabs

User Interface

Rename Tabs and Labels

Tabs

Didn't find what you're looking for?
Try using Global Search.

SETUP
Tabs

Custom Tabs

Help for this Page ❓

You can create new custom tabs to extend Salesforce functionality or to build new application functionality.

Custom Object tabs look and behave like the standard tabs provided with Salesforce. Web tabs allow you to embed external web applications and content within the Salesforce window. Visualforce tabs allow you to embed Visualforce pages. Lightning Component tabs allow you to add Lightning components to the navigation menu in Lightning Experience and the mobile app. Lightning Page tabs allow you to add Lightning Pages to Lightning Experience and the mobile app.

Custom Object Tabs [New] What Is This?

Action	Label	Tab Style	Description
Edit | Del	Buyers	Credit card	
Edit | Del	Fuel details	CD/DVD	
Edit | Del	Gas Stations	Car	
Edit | Del	Suppliers	Flag	

Web Tabs [New] What Is This?

No Web Tabs have been defined

Visualforce Tabs [New] What Is This?

No Visualforce Tabs have been defined

SETUP > OBJECT MANAGER
Buyer

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	
Scoping Rules	

Field Name: Amount_Paid
Description:
Help Text:
Data Owner: User ⌄
Field Usage: --None-- ⌄
Data Sensitivity Level: --None-- ⌄
Compliance Categorization: Available / Chosen
  PII
  HIPAA
  GDPR
  PCI

Formula Options

Formula Return Type: Number ⌄
Decimal Places: 2 ⌄

Enter your formula and click Check Syntax to check for errors. Click the Advanced Formula subtab to use additional fields, operators, and functions.
Example: Fahrenheit = 1.8 * Celsius__c + 32 | More Examples...

Simple Formula | Advanced Formula

[Insert Field] [Insert Operator ⌄]

Functions
-- All Function Categories -- ⌄
ABS
ACOS

Amount Paid (Number) =
Fuel_filled_in_vehicle__c * Gas_Station_name__r.Fuel_Price_litre__c

Quick Tips
Getting Started
Operators & Functions

| | Contact Point Consents | ✓ | ✓ | ✓ | ✓ | ☐ | ☐ | ☐ | | Scheduling Overview | ✓ | ✓ | ✓ | ✓ | ☐ | | ☐ |
| | Contact Point Emails | ✓ | ✓ | ✓ | ✓ | ☐ | ☐ | ☐ | | User External Credentials | ✓ | ✓ | ✓ | ✓ | ☐ | | ☐ |

Custom Object Permissions

	Basic Access				Data Administration					Basic Access				Data Administration		
	Read	Create	Edit	Delete	View All Records	Modify All Records	View All Fields			Read	Create	Edit	Delete	View All Records	Modify All Records	View All Fields
Buyers	✓	☐	☐	☐	☐	☐	☐	Gas Stations		✓	☐	✓	☐	☐	☐	☐
Fuel details	✓	☐	☐	☐	☐	☐	☐	Suppliers		✓	☐	☐	☐	☐	☐	☐

Session Settings

| Session Times Out After | 2 hours of inactivity | Session Security Level Required at Login | |

Password Policies

User passwords expire in	90 days
Enforce password history	3 passwords remembered
Minimum password length	8
Password complexity requirement	Must include alpha and numeric characters
Password question requirement	Cannot contain password
Maximum invalid login attempts	10
Lockout effective period	15 minutes
Obscure secret answer for password resets	☐
Require a minimum 1 day password lifetime	☐
Don't immediately expire links in forgot password emails	☐

| | Contact Point Emails | ✓ | ✓ | ✓ | ✓ | ☐ | ☐ | ☐ | | User External Credentials | ✓ | ✓ | ✓ | ✓ | ☐ | | ☐ |

Custom Object Permissions

	Basic Access				Data Administration					Basic Access				Data Administration		
	Read	Create	Edit	Delete	View All Records	Modify All Records	View All Fields			Read	Create	Edit	Delete	View All Records	Modify All Records	View All Fields
Buyers	✓	✓	☐	☐	☐	☐	☐	Gas Stations		✓	☐	☐	☐	☐	☐	☐
Fuel details	✓	✓	☐	☐	☐	☐	☐	Suppliers		✓	☐	☐	☐	☐	☐	☐

Session Settings

| Session Times Out After | 2 hours of inactivity | Session Security Level Required at Login | |

Password Policies

User passwords expire in	90 days
Enforce password history	3 passwords remembered
Minimum password length	8
Password complexity requirement	Must include alpha and numeric characters

Conclusion

The CRM application for gas filling stations provides an efficient and automated solution to manage daily operations, customer interactions, and fuel inventory. By integrating customer management, transaction tracking, employee monitoring, and analytics, the system streamlines business processes, reduces manual errors, and enhances overall customer satisfaction. The implementation of loyalty programs and real-time reporting helps in building strong customer relationships and making informed business decisions. Overall, this project demonstrates how technology can optimize operational efficiency, improve service quality, and support the growth of fuel station businesses.