A CRM Application to Handle the Clients and their property Related Requirements

Project Description:

Dreams World Properties integrates Salesforce to streamline customer interactions. Website engagement triggers automated record creation in Salesforce, capturing customer details and preferences. Salesforce categorizes users as approved or non-approved, offering tailored property selections to approved users. This enhances user experience and efficiency, providing personalized recommendations and broader listings. Seamless integration optimizes operations, improving customer engagement and facilitating growth in the real estate market.

Objectives:

1. Automate customer data capture from the website into Salesforce.
2. Classify customers into approved and non-approved categories for personalized services.
3. Provide tailored property recommendations to enhance user engagement.
4. Centralize property and customer data for streamlined management.

Key Features

1 Customer Data Management

- Automated capture of customer details (e.g., Name, Contact, Preferences).
- Centralized data storage in Salesforce for easy access and updates.

2 User Segmentation

- Categorization into:
    - **Approved Users**: Access to personalized property suggestions.
    - **Non-Approved Users**: Access to broader property listings.

3 Property Listings

- Dynamic and centralized property management.
- Tailored property suggestions for approved users based on preferences.

4 Integration

- Real-time synchronization between website interactions and Salesforce.
- Custom workflows to automate data processing and record creation.

5 Reporting and Analytics

- Real-time dashboards to track customer engagement and property interests.

- Reports on user segmentation, property performance, and operational metrics.

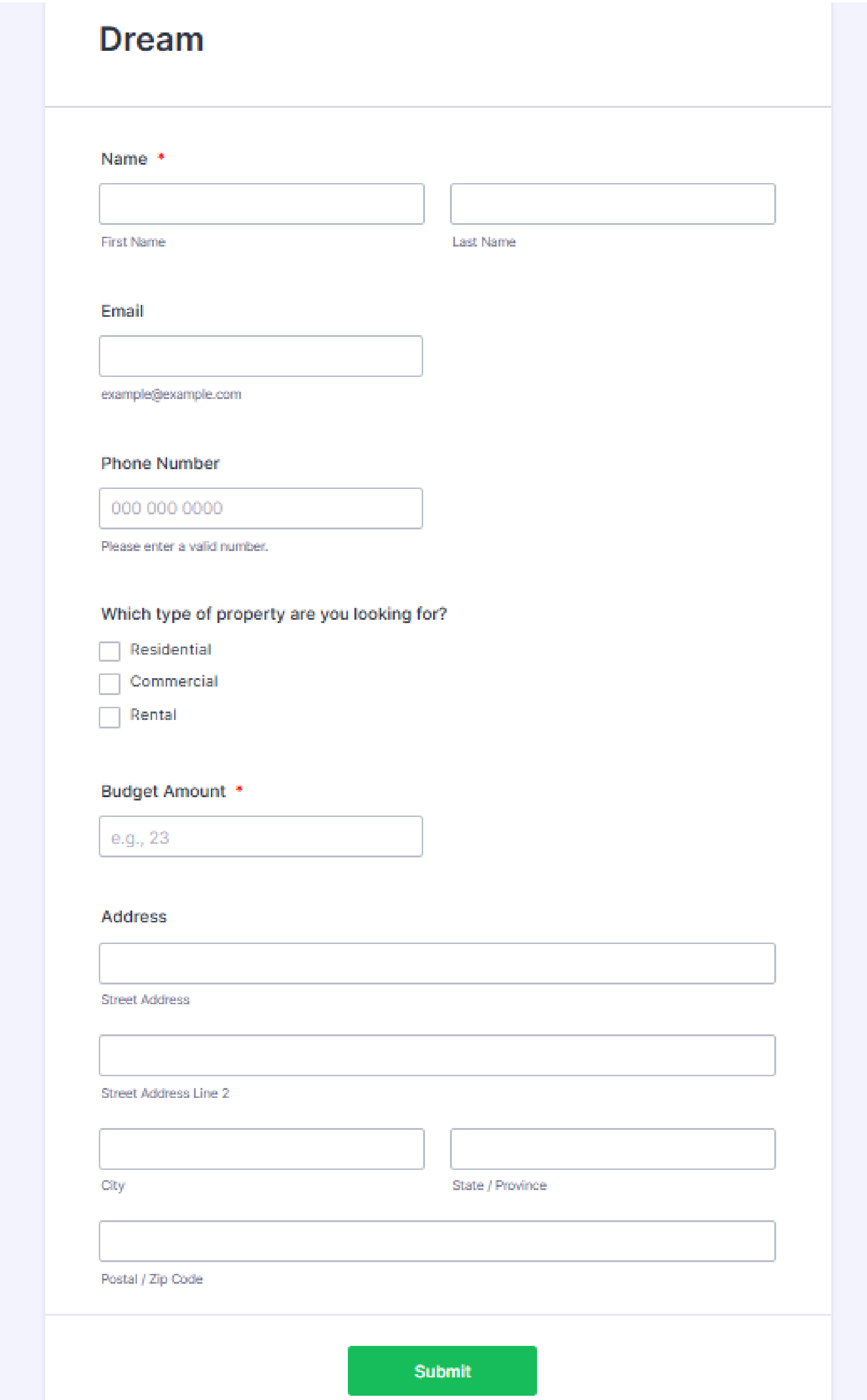
Key Milestones and Activities

Milestone 1: Jotform Integration

Goal: Create a Jotform and integrate it with Salesforce to automate the creation of customer records.

Activities:

1. Created Jotform is mentioned in below



Dream

Name *

First Name

Last Name

Email

example@example.com

Phone Number

000 000 0000

Please enter a valid number.

Which type of property are you looking for?

- [ ] Residential
- [ ] Commercial
- [ ] Rental

Budget Amount *

e.g., 23

Address

Street Address

Street Address Line 2

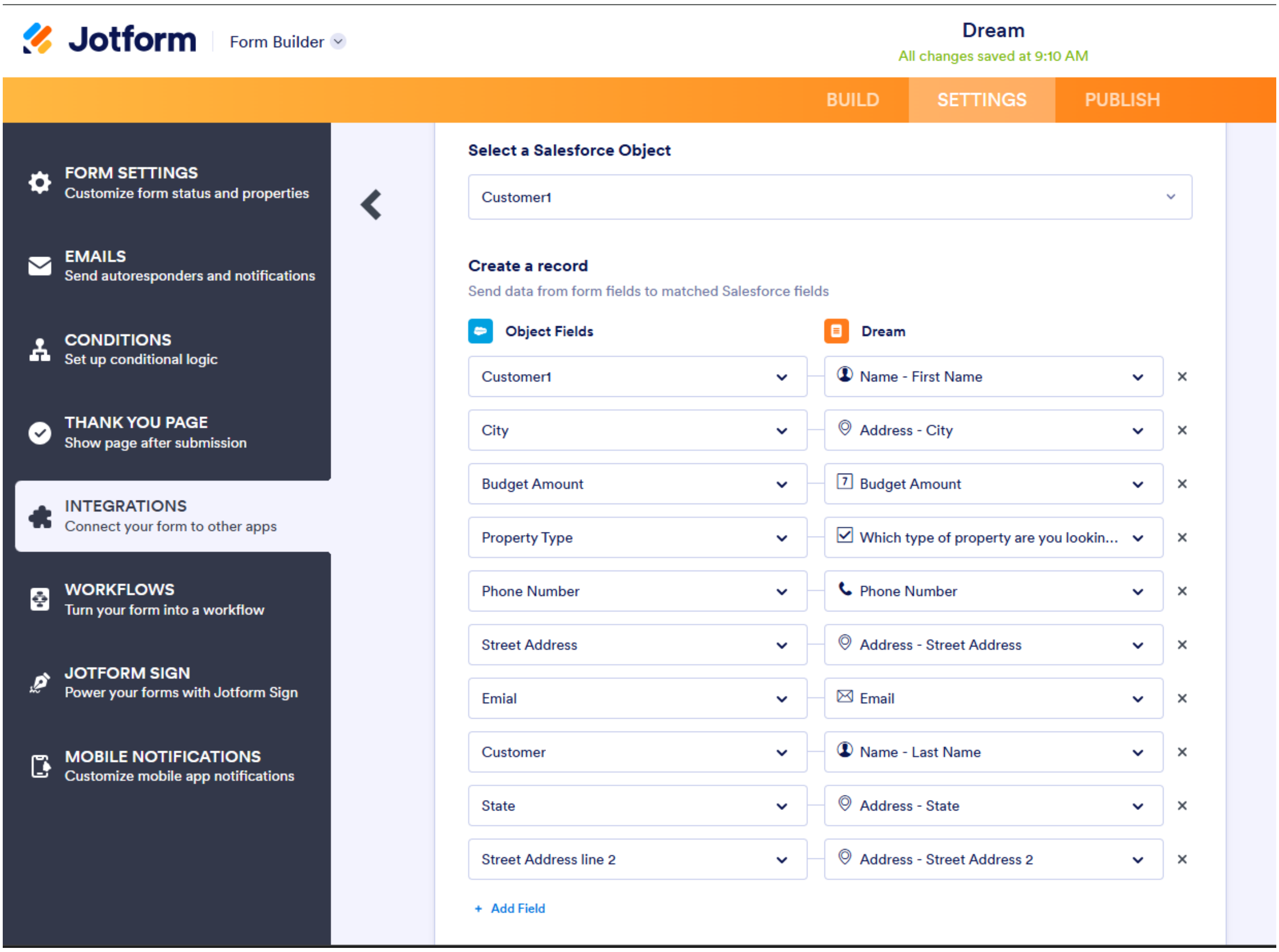
City

State / Province

Postal / Zip Code

Submit

2. **Create Objects from Spreadsheet:**
   - Import data from spreadsheets to Salesforce.
   - Set up necessary custom objects (e.g., Client, Property).
3. **Integrate Jotform with Salesforce Platform:**
   - Establish a connection between Jotform and Salesforce using APIs or integration tools.
   - Ensure automated record creation in Salesforce when a Jotform is submitted.



Milestone 2: Role and Profile Management

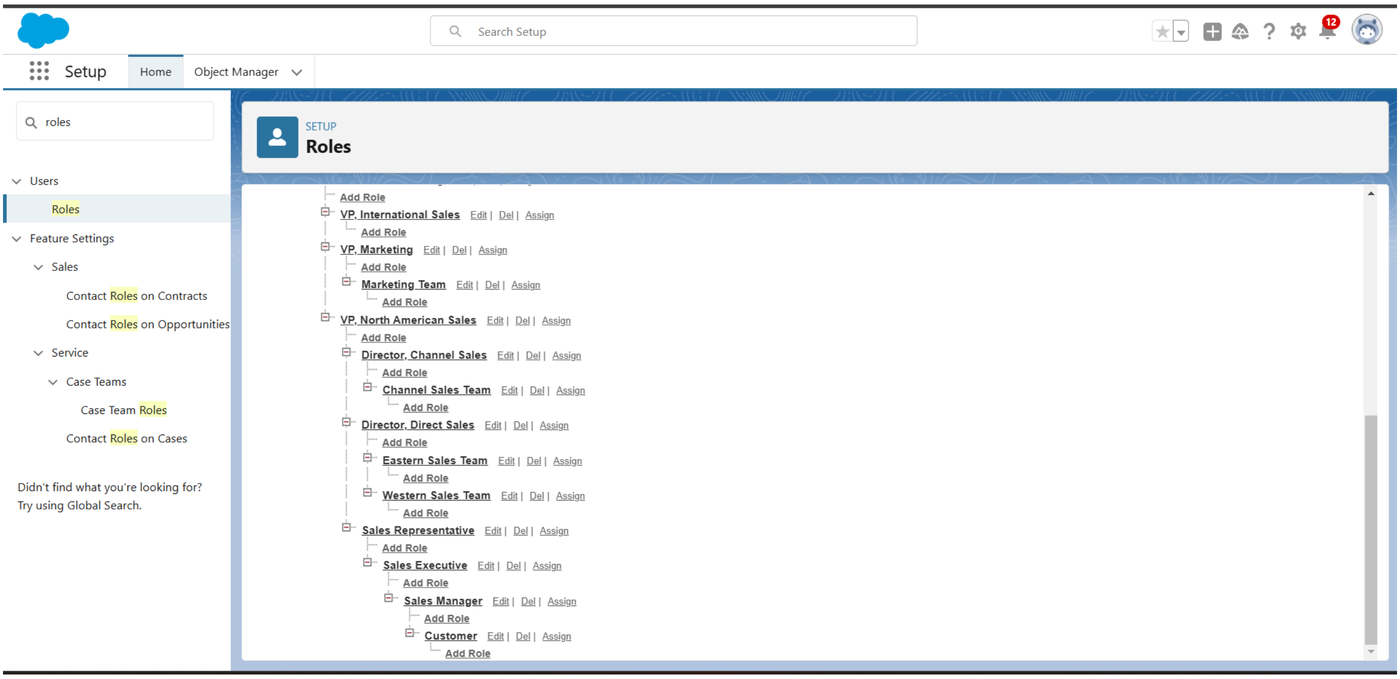
Goal: Establish roles and profiles for users to manage access and security within the CRM system.

Activities:

1. **Create Roles:**

- **Sales Executive**
  - **Position:** Top-level role in the hierarchy.

        - **Responsibilities:**
            - Oversees sales operations and strategy.
            - Monitors team performance and approves high-level decisions.
        - **Reports To:** No one (top-level).
- **Sales Manager**
        - **Position:** Mid-level role, directly below Sales Executive.
        - **Responsibilities:**
            - Manages the sales team and handles customer relationships.
            - Assigns leads and tracks team performance.
        - **Reports To:** Sales Executive.
- **Customer**
        - **Position:** Entry-level role, below Sales Manager.
        - **Responsibilities:**
            - Submits property requirements and preferences.
            - Collaborates with Sales Managers for assistance.
        - **Reports To:** Sales Manager.



Milestone 3: Property Management

Goal: Create a property management system within Salesforce for tracking and approval processes.

Steps to Create the "Property Details" Lightning App

1. **Go to App Manager**
        - From **Salesforce Setup**, type **"App Manager"** in the Quick Find box.
        - Click on **App Manager**.
2. **Create a New Lightning App**

- In the App Manager, click on the **New Lightning App** button.

3. **Define App Details**
    - **App Name**: Enter "**Property Details**".
    - **Developer Name**: Automatically populated (can leave as is).
    - **Description**: Add a brief description, e.g., "App for managing Customers and Property data."
    - Click **Next**.
4. **App Branding** (Optional)
    - Upload a custom logo if required.
    - Select a color scheme for the app.
    - Click **Next**.
5. **Navigation Style**
    - Choose **Standard Navigation** (default option for most apps).
    - Click **Next**.
6. **Add Objects**
    - In the **Select Items** step, search for and add the following objects:
        - **Customer**
        - **Property**
    - Drag and drop them into the **Selected Items** section.
    - Click **Next**.
7. **Set Profiles**
    - Assign the app to the appropriate user profiles (e.g., **Sales Manager**, **Sales Executive**).
    - Select profiles by checking the boxes next to their names.
    - Click **Save & Finish**.

Accessing the Lightning App

- Go to the **App Launcher** (grid icon in the top-left corner of Salesforce).
- Search for **"Property Details"** and open the app.
- The app will display the **Customer** and **Property** objects for easy access.

Steps to Create and Configure the "Customer" Profile

1. **Navigate to Profiles**
    - From **Setup**, type **"Profiles"** in the Quick Find box.
    - Click on **Profiles**.
2. **Clone the Salesforce Platform User Profile**
    - Locate the **Salesforce Platform User** profile in the list.
    - Click the **Clone** link next to it.
3. **Name the New Profile**
    - **Profile Name**: Enter "**Customer**".
    - Click **Save**.

1: Modify Custom App Settings

1. In the "**Customer**" profile, scroll to the **Custom App Settings** section.
2. **Uncheck All Custom Apps** except "**Property Details**".

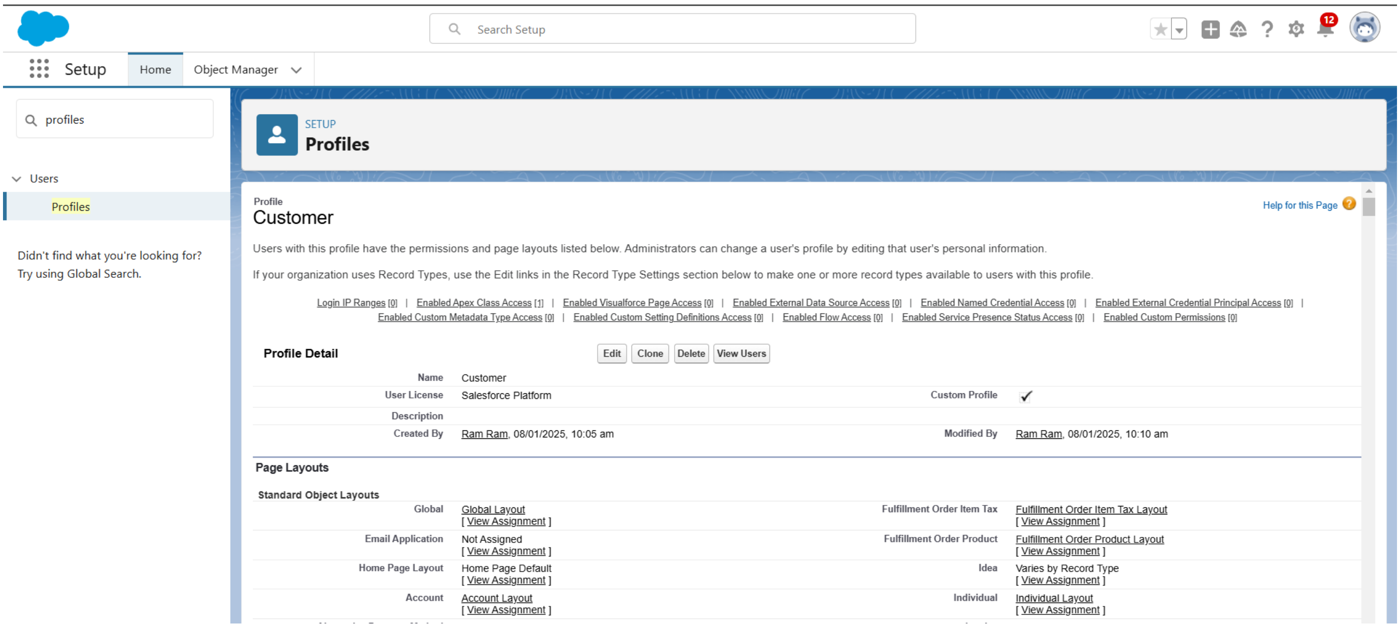
■ Ensure only the **Property Details** app is checked.

2: Remove Standard Object Permissions

1. Scroll to the **Standard Object Permissions** section.
2. Uncheck all permissions (Read, Create, Edit, Delete) for all standard objects.

3: Modify Custom Object Permissions

1. Scroll to the **Custom Object Permissions** section.
2. **Uncheck All Permissions** for every custom object except "**Property**".
   - For the **Property** object, check only:
     - **Read**
     - **View All**



Result

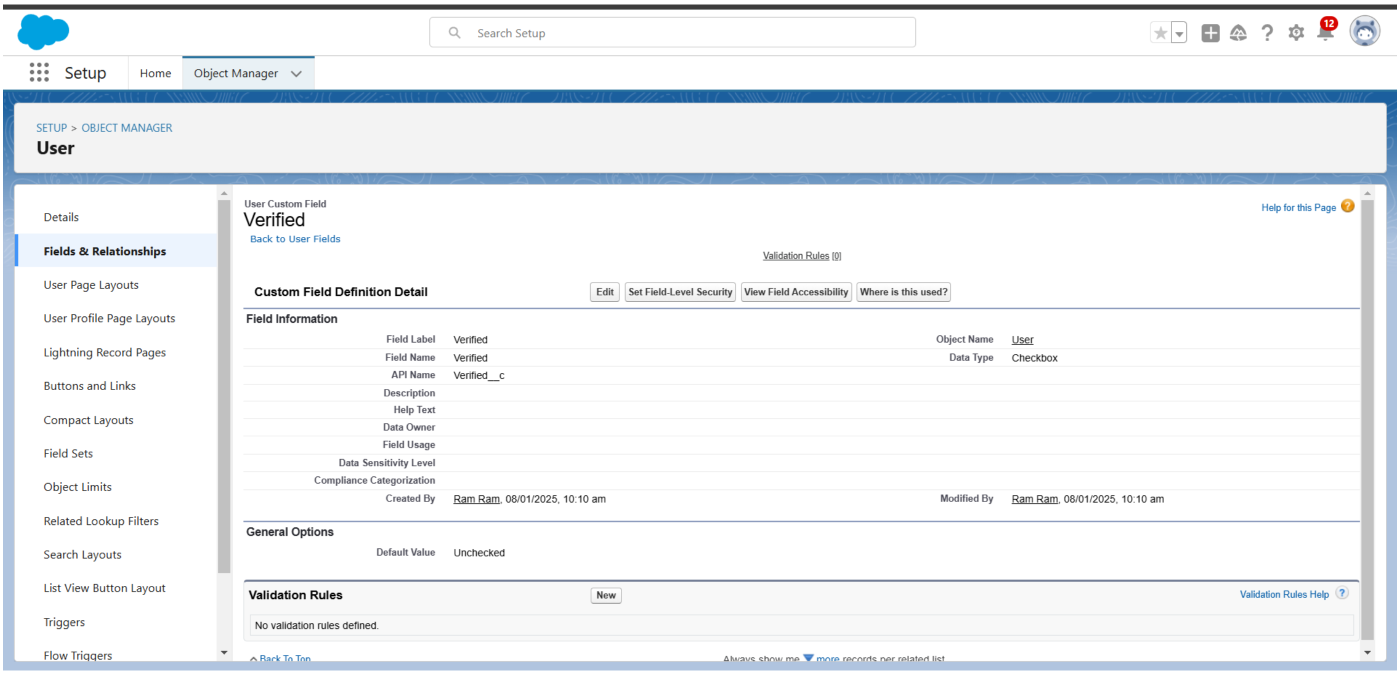
- The **Customer** profile now has access only to the **Property Details** app and the **Property** object with **Read** and **View All** permissions.
- It has no permissions for any other standard or custom objects.

Similarly create the **Manager** profile.

Steps to Create the "Verified" Checkbox Field

1. **Navigate to Object Manager**
   - From **Setup**, type "**Object Manager**" in the Quick Find box.
   - Click on **Object Manager**.

2. **Search for the User Object**
   - In the Object Manager search bar, type **"User"**.
   - Select the **User** object from the results.
3. **Go to Fields & Relationships**
   - In the User object setup page, click on **Fields & Relationships** in the left-hand menu.
4. **Create a New Field**
   - Click on the **New** button at the top of the Fields & Relationships list.
5. **Choose Field Type**
   - Select **Checkbox** as the **Data Type**.
   - Click **Next**.
6. **Define Field Details**
   - **Field Label**: Enter **"Verified"**.
   - **Default Value**:
     - Leave unchecked (default is "False").
   - **Field Name**: This will auto-populate as **"Verified"**.
   - Click **Next**.
7. **Set Field-Level Security**
   - Select the profiles that should have access to this field (e.g., **Sales Executive**, **Sales Manager**, or **Customer**).
   - Ensure **Read-Only** or **Editable** access is configured as needed.
   - Click **Next**.
8. **Add to Page Layouts**
   - Select the page layouts where the **Verified** field should appear.
   - Click **Save**.



Result

You now have a **"Verified"** checkbox field on the **User** object. Users can toggle this field to indicate whether a user is verified or not.

Steps to Create the "Users"

Create 4 separate users following the same steps, making sure to fill in unique details like **Username**, **First Name**, and **Email** for each new user.

- Go to **Setup**.
- Under **Administration**, select **Users**.
- Click on **New User**.

User 1:

- Last Name: Executive
- Role: Sales Executive
- License: Salesforce
- Profile: System Administrator
- Click **Save**.

User 2:

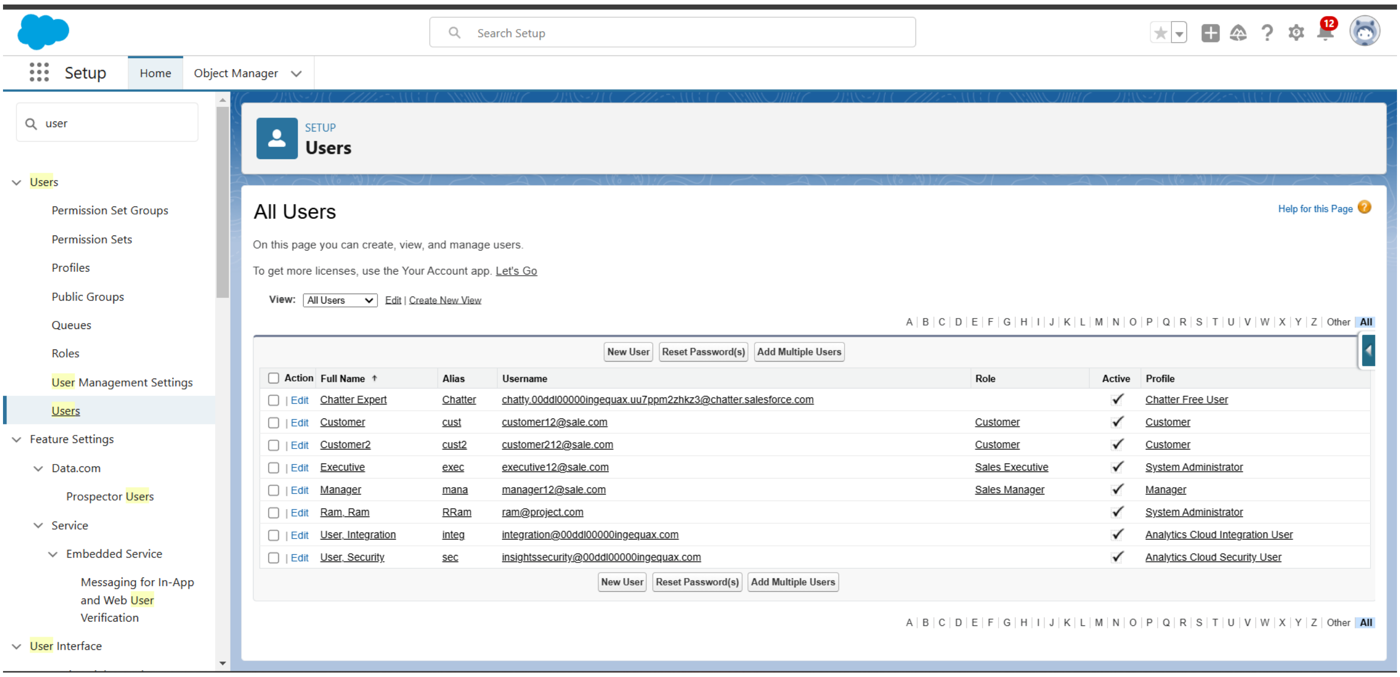
- Last Name: Manager
- Role: Sales Manager
- License: Salesforce Platform
- Profile: Manager
- Click **Save**.

User 3:

- Last Name: Customer
- Role: Customer
- License: Salesforce Platform
- Profile: Customer
- Click **Save**.

User 4:

- Last Name: Customer2
- Role: Customer
- License: Salesforce Platform
- Profile: Customer
- Click **Save**.



Steps to Create Approval Process for Property Object:

1. Log in to Salesforce:

- Ensure you have the necessary admin permissions to create the approval process.

2. Navigate to Setup:

- Click the **Gear icon** (⚙) in the top-right corner.
- Select **Setup**.

3. Access Approval Processes:

- In the **Quick Find** search box, type **Approval Processes**.
- Under **Process Automation**, select **Approval Processes**.

4. Select the Property Object:

- Choose the **Property** object from the list of available objects.

5. Create New Approval Process:

- Click the **Create New Approval Process** button.
- Select **Use Standard Setup Wizard**.
- Enter **Process Name**: `Property Approval`.

6. Define Criteria for Entry:

- Add the following criteria:
  - **Location is not equal to blank**.
  - **Verified Equals false**.
- This ensures the process only starts when the location is filled in and the record has not been verified.

7. Select the Automated Approver:

- Choose **Next Automated Approver Determined By**: Select **Manager**.
- This will automatically route the approval to the manager.

8. Record Editability Properties:

- From **Record Editability Properties**, select **Administrators or the currently assigned approver can edit records during the approval process**.
- This ensures only the approver or administrator can make changes while the record is being approved.

9. Select Fields to Display on Approval Page Layout:

- Select the following fields to display on the approval page layout:
  - **Property**
  - **Owner**
  - **Location**
  - **Type**
- These fields will be visible to the approvers when reviewing the records.

10. Select Initial Submitters:

- Choose the following initial submitters for the approval process:
  - **Owner**: Property Owner.
  - **Roles**: Sales Manager.

11. Click Next and Save:

- Click **Next** and review the settings.
- Click **Save**.

Approval Step: Executive Approval

12. Add an Approval Step:

- Click **Add Approval Step**.
- Name the step: **Executive Approval**.

13. Define Criteria for Approval Step:

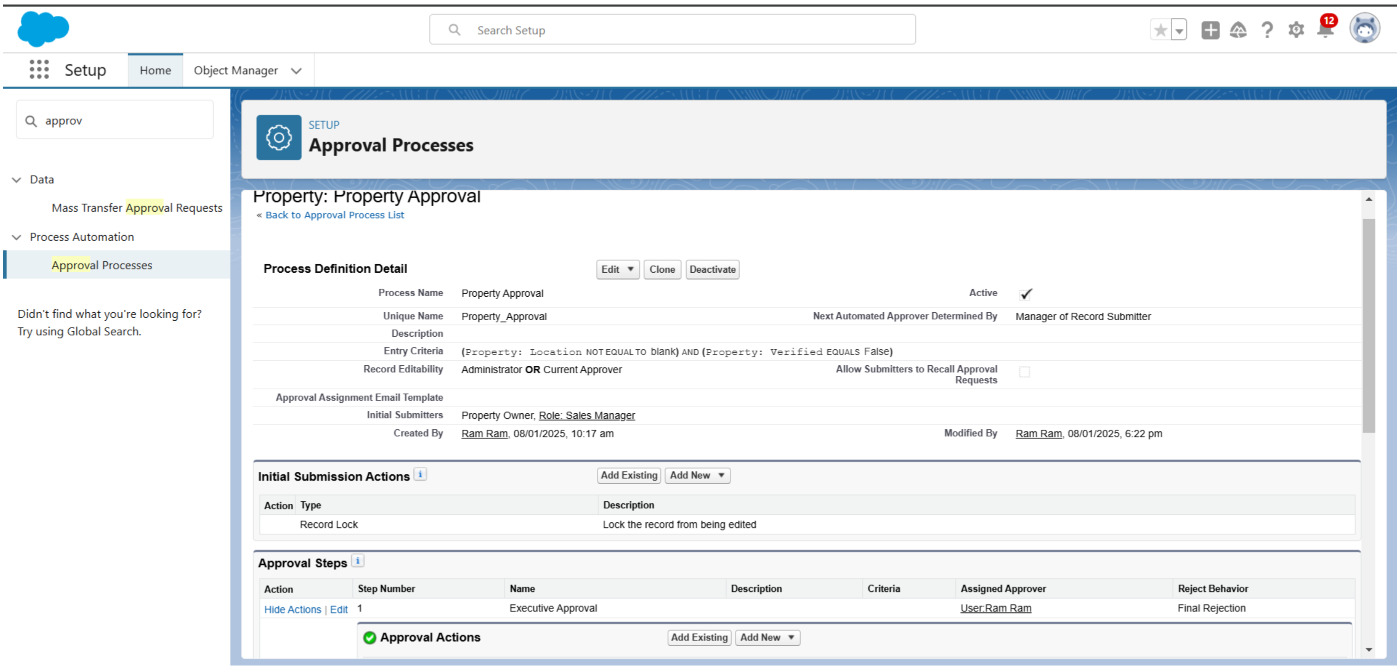
- Specify the criteria: **All records should enter this step**.
- This ensures that every record enters the **Executive Approval** step, regardless of other conditions.

14. Select Approvers:

- Select the approvers for this step:
    - **Sales Executive**.

15. Save the Step:

- After selecting the approvers, click **Save** to add the step to the approval process.



Final Steps:

16. Activate the Approval Process:

- Once you have set up the approval process and all the steps, click **Activate** to make it active.

17. Review the Approval Process:

- Go back to the **Approval Processes** page under the **Property** object and verify that the process is listed and active.

Steps to Create a Record Trigger Flow to Submit the Approval Process Automatically for the Property Object:

1. Log in to Salesforce:

- Make sure you have the necessary permissions to create a flow.

2. Navigate to Flows:

- Click on the **Gear icon** (⚙) in the top-right corner.
- Select **Setup**.
- In the **Quick Find** box, search for **Flows**.

3. Create a New Flow:

- Click **New Flow**.
- Choose **Record Trigger Flow**.

4. Select the Object:

- Select the **Object**: **Property**.

5. Define Trigger Criteria:

- Choose **Trigger the Flow When**: **A record is created**.
    - This ensures that the flow is triggered whenever a new Property record is created.

6. Set Entry Conditions:

- For this scenario, choose **None** for entry conditions. This means the flow will trigger on every record creation without specific conditions.

7. Add Action to Submit for Approval:

- Click **+ Add Element**.
- Select **Action** from the options.
- In the **Action** dropdown, choose **Submit for Approval**.

8. Configure the Submit for Approval Action:

- **Label**: Give the action a name like **Approval for Property**.
- **Record ID**: Set the Record ID to **{!$Record.Id}**.
    - This ensures the flow will submit the **current Property record** for approval.

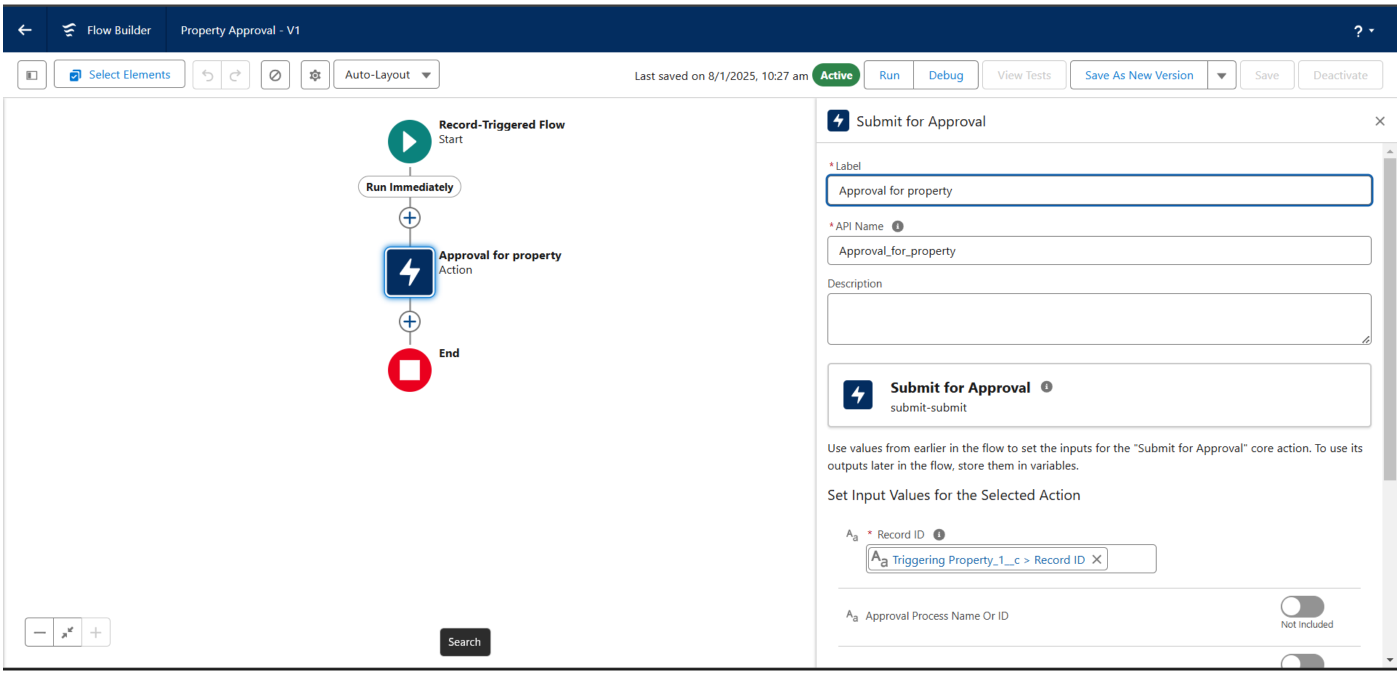
9. Save the Flow:

- After configuring the action, click **Done**.
- Click **Save** to save the flow.

- **Label**: Name the flow **Property Approval**.

10. Activate the Flow:

- After saving the flow, click **Activate** to make the flow live.



Steps to Create an App Page for the Property Object:

1. Log in to Salesforce:

- Ensure you have the necessary permissions to create and configure pages.

2. Navigate to Lightning App Builder:

- Click the **Gear icon** (⚙) in the top-right corner and select **Setup**.
- In the **Quick Find** search box, type **Lightning App Builder**.
- Click on **Lightning App Builder**.

3. Create a New App Page:

- Click on the **New** button.
- Select **App Page** and then click **Next**.

4. Set App Page Details:

- **Label**: Enter the name **Search Your Property**.
- Click **Next** to continue.

5. Choose the Layout:

- Select the **Header and Left Sidebar** layout (this will display the necessary components in the header and sidebar).
- Click **Done**.

6. Save the Page:

- After configuring the layout, click **Save**.

7. Activate the Page:

- Click **Activate** to make the page available for use.

8. Page Activation Settings:

- In the **Page Settings** panel, select **Activate for All Users**.
- This ensures that all users will have access to the app page.

9. Add App Page to Property Details:

- In **Lightning Experience**, go to the **Property Details** object page.
- Click on **Add Page** to associate the new page with this object.
- Once done, click **Save** to finalize the addition.

Steps to Create an LWC Component for Verified and Non-Verified Customers:

Here is a detailed guide to create a **Lightning Web Component (LWC)** that restricts access to properties based on customer verification status, and deploy it on the **Search Your Property Page**:

1. Set Up the LWC Component:

You will need to create an LWC component that will:

- Check the verification status of the customer.
- Display only verified properties to verified customers and non-verified properties to non-verified customers.

2. Create the LWC Component:

A. Prepare the Component Structure:

1. **Navigate to Developer Console** or use **VS Code** to create the LWC.
2. Create a new Lightning Web Component (LWC) with the following files:
    - **HTML File** (e.g., `propertyDisplay.html`)
    - **JS File** (e.g., `propertyDisplay.js`)

- **Meta File** (e.g., `propertyDisplay.js-metaxml`)

c_01_Property_Management.html × c_01_Property_Management.js c_01_Property_Management.js-meta.xml

force-app > main > default > lwc > c_01_Property_Management > c_01_Property_Management.html > template > lightning-card > template

```html
<template>
  <lightning-card>
    <div class="slds-box">
      <div class="slds-text-align_left">
        <h1 style="font-size: 20px;"><b>Properties</b></h1>
      </div>
      <div>
        <div class="slds-grid slds-gutters">
          <div class="slds-col slds-size_5-of-6">
            <lightning-combobox name="Type" label="Property Type" value={typevar} placeholder="Select Property type"
              options={propetyoptions} onchange={changehandler}></lightning-combobox>
          </div>
          <div class="slds-col slds-size_1-of-6">
            <br>
            <lightning-button-icon variant="neutral" icon-name="standard:search" alternative-text="Search"
            label="Search" onclick={handleClick}></lightning-button-icon>
          </div>
        </div>
      </div>
```

Steps to Add the LWC Component to Your App Page ("Search Your Property"):

Here is the detailed guide to add the **LWC Component** (PropertyDisplay) to the **Search Your Property Page** within Salesforce:

1. Go to App Launcher:

- Click on the **App Launcher** (Grid icon) in the top-left corner of Salesforce.
- In the search box, type **Property Details** and select the corresponding page.

2. Edit the Page:

- Once you're on the **Property Details** page, click the **Gear icon** (⚙) in the top-right corner of the page.
- Select **Edit Page** from the dropdown. This will open the **Lightning App Builder** in edit mode.

3. Drag the Component to the Page:

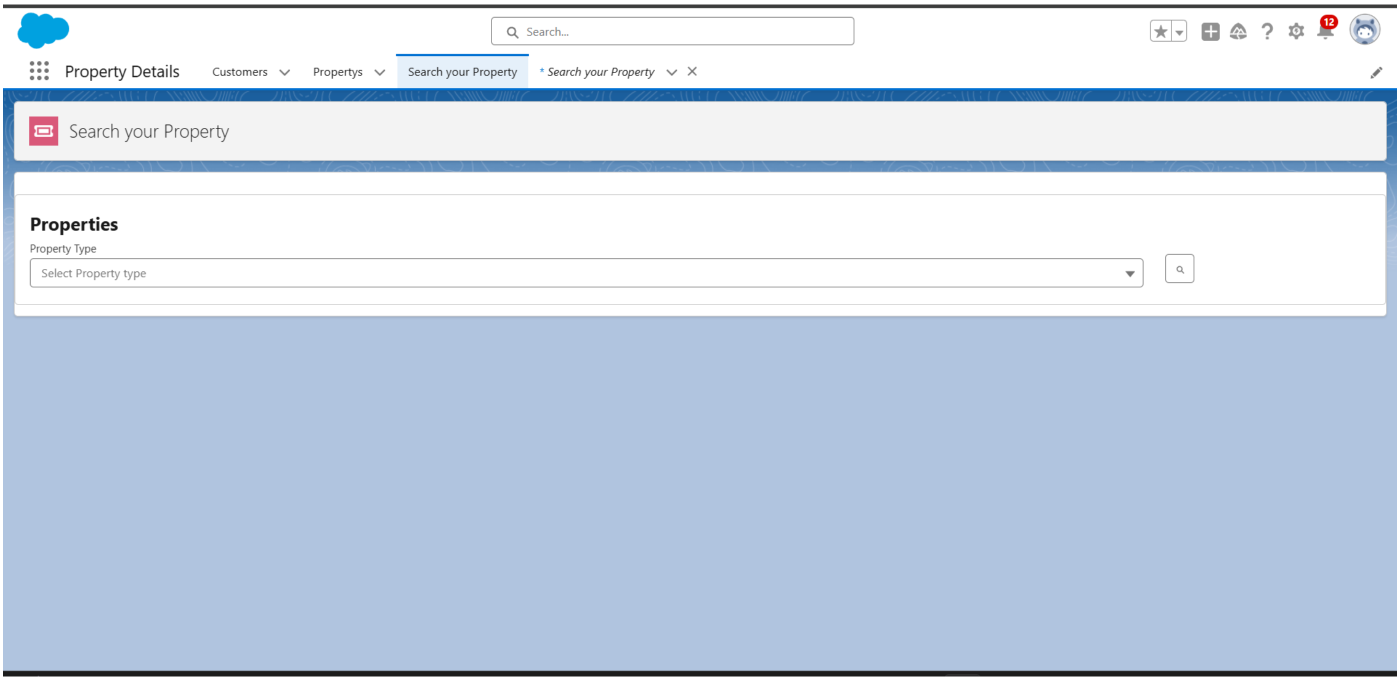
- In the **Lightning App Builder**, on the left side, you should see the list of components available (including your custom components).
- Find your **LWC Component** (PropertyDisplay) under the **Custom Components** section.
- Drag the **PropertyDisplay** component onto the desired location on the page layout.

4. Save the Page:

- After adding the component, click **Save** to save the changes to the page.

5. Activate the Page:

- Once saved, you can activate the page to make it available to all users or select specific user profiles. This can be done in the **Page Activation** settings within the App Builder.



Steps to Give Access to Apex Classes for Specific Profiles:

Here's how to grant access to the **Apex Class** (`PropertyHandler__LWC`) to the **Manager** and **Customer** profiles:

1. Navigate to Apex Classes:

- In Salesforce, click the **Gear icon** (⚙) in the top-right corner and select **Setup**.
- In the **Quick Find** search box, type **Apex Classes**.
- Click on **Apex Classes** under the **Develop** section.

2. Find the Apex Class:

- Locate the **Apex Class** you want to give access to (e.g., **`PropertyHandler__LWC`**).
- Click on the **Security** link next to the class name (this will open the class' security settings).

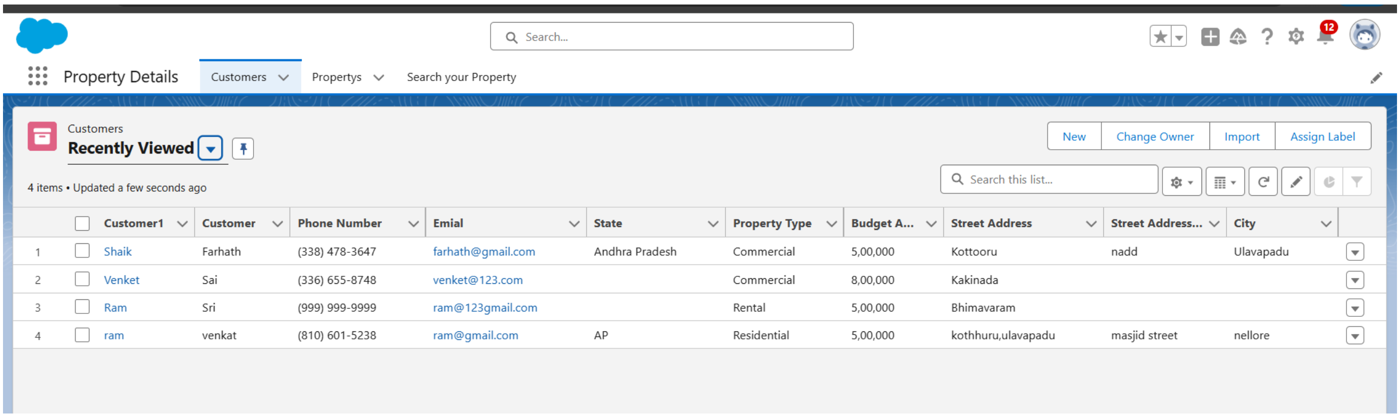
3. Edit Access for Profiles:

- Under **Apex Class Access** settings, you will see a list of profiles.
- To give access to the class for specific profiles, check the boxes for the **Manager** and **Customer** profiles.

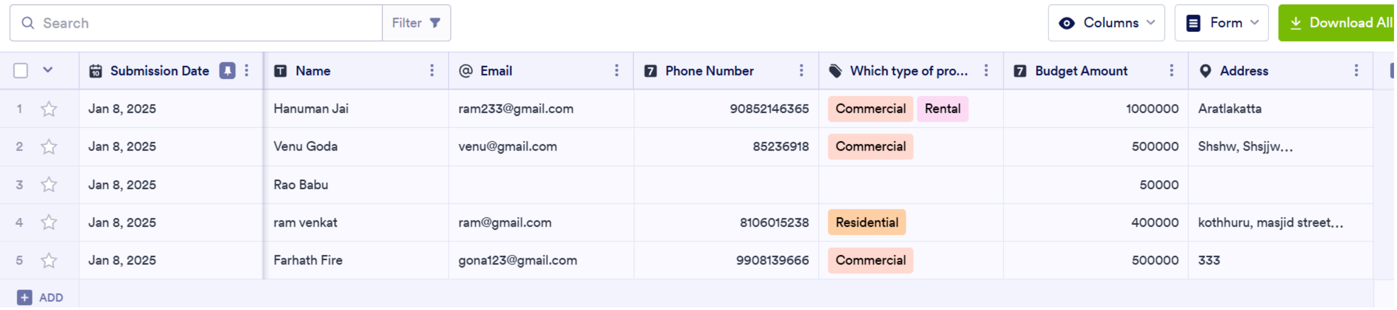
4. Save the Changes:

- After selecting the profiles, click **Save** to apply the access settings.

Final Outcome:



	Customer1	Customer	Phone Number	Emial	State	Property Type	Budget A...	Street Address	Street Address...	City
1	Shaik	Farhath	(338) 478-3647	farhath@gmail.com	Andhra Pradesh	Commercial	5,00,000	Kottooru	nadd	Ulavapadu
2	Venket	Sai	(336) 655-8748	venket@123.com		Commercial	8,00,000	Kakinada		
3	Ram	Sri	(999) 999-9999	ram@123gmail.com		Rental	5,00,000	Bhimavaram		
4	ram	venkat	(810) 601-5238	ram@gmail.com	AP	Residential	5,00,000	kothhuru,ulavapadu	masjid street	nellore



	Submission Date	Name	Email	Phone Number	Which type of pro...	Budget Amount	Address
1	Jan 8, 2025	Hanuman Jai	ram233@gmail.com	90852146365	Commercial Rental	1000000	Aratlakatta
2	Jan 8, 2025	Venu Goda	venu@gmail.com	85236918	Commercial	500000	Shshw, Shsjjw...
3	Jan 8, 2025	Rao Babu				50000	
4	Jan 8, 2025	ram venkat	ram@gmail.com	8106015238	Residential	400000	kothhuru, masjid street...
5	Jan 8, 2025	Farhath Fire	gona123@gmail.com	9908139666	Commercial	500000	333

Testing and Validation:

1. Unit Testing:

- Verify the functionality of individual components such as data capture flows, approval processes, and LWCs.
- Ensure test cases cover all possible user scenarios.

2. Integration Testing:

- Test end-to-end functionality, ensuring seamless interaction between the website and Salesforce.
- Validate the accuracy of data synchronization.

3. User Acceptance Testing (UAT):

- Collaborate with stakeholders to confirm that the system meets their requirements.
- Address any feedback or adjustments needed.

4. Performance Testing:

- Assess the system's performance under various loads to ensure reliability.
- Optimize areas with potential bottlenecks.

5. Security Testing:

- Verify data access permissions based on roles and profiles.
- Ensure sensitive customer data is securely stored and transmitted.

Key Scenarios:

1. Milestone 1: Automating Customer Record Creation
- Integrate JotForm with Salesforce to capture customer data and preferences automatically.

2. Creating and Managing Objects:
- Create objects from spreadsheets to import initial data.
- Use Salesforce to manage and update these objects.

3. User and Role Management:
- Create user roles and profiles to control data access and permissions.
- Add a checkbox field to manage user approval status.

4. Property Details App:
- Develop an app to store, display, and manage property details.

5. Approval Process:
- Implement an approval process for property objects to streamline decision-making.
- Use record-triggered flows to automate approval submissions.

6. LWC Component:
- Create and drag an LWC component to an app page to enhance UI functionality.

7. Apex Class Access:
- Grant Apex class access to profiles for advanced functionality.

Conclusion

This CRM application will provide a comprehensive solution for managing clients and their property-related requirements. With automated workflows, streamlined processes, and an intuitive interface, Dreams World Properties will achieve greater efficiency and customer satisfaction.