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

1. Project Overview

This CRM Application to Handle the Clients and their property Related Requirements is integrating Salesforce to streamline client interactions and manage property-related requirements. The objective is to enhance operational efficiency, automate data handling, and improve customer experience through:

- Automated record creation upon website interactions.
- Categorization of users into *approved* and *non-approved* groups.
- Personalized property recommendations for approved users.

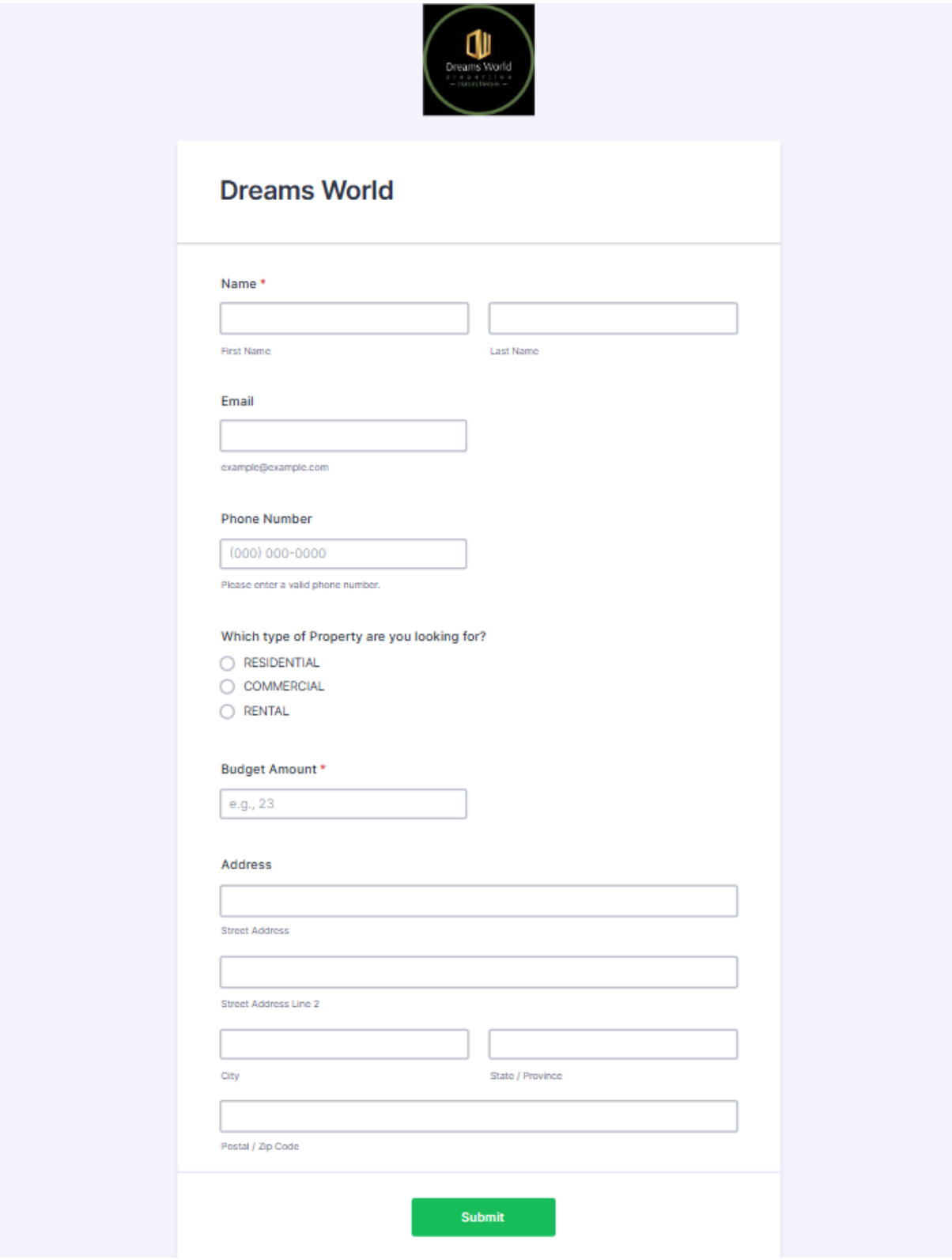
2. Objectives

1. **Automate Client Data Management**: Automatically generate customer records from website interactions.
2. **User Categorization**: Classify users as approved or non-approved to tailor property recommendations.
3. **Approval Workflow**: Implement an approval process for client access to specific property listings.
4. **Scalability**: Ensure the system can scale with increasing customer and data volume.

3. Project Milestones

Milestone 1:Create a Jotform and integrate it with the org to create a record of customers automatically.

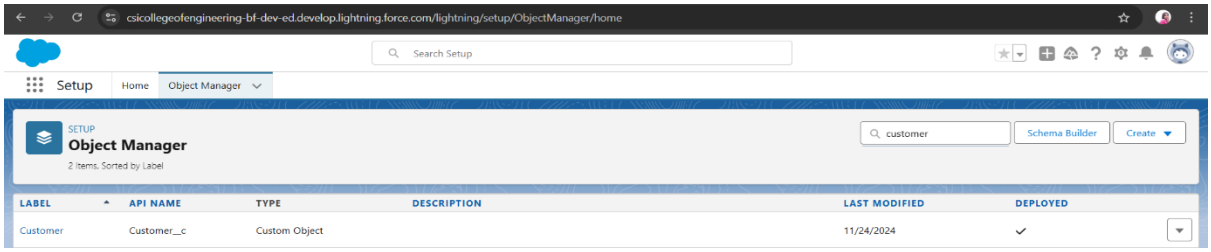
- Create a form to collect customer data and link it to Salesforce.
- Automatically generate customer records.



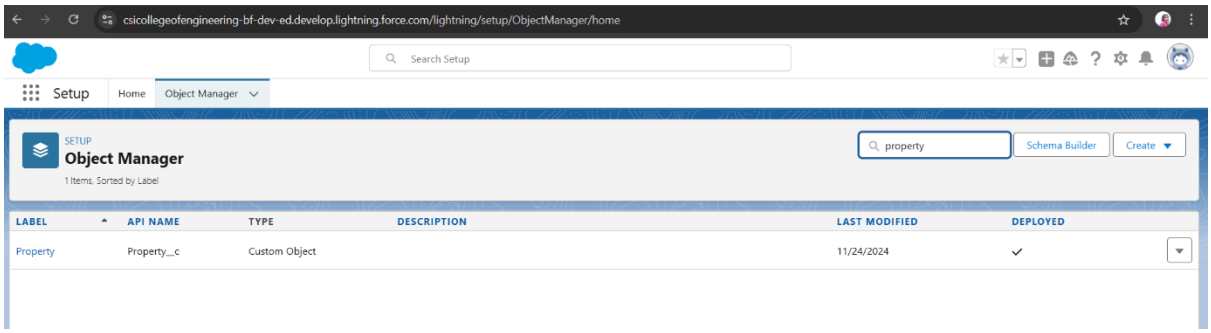
Milestone 2: Create Objects from Spreadsheet.

Directly Creating Objects from Spreadsheet in Salesforce

- Create Customer Object

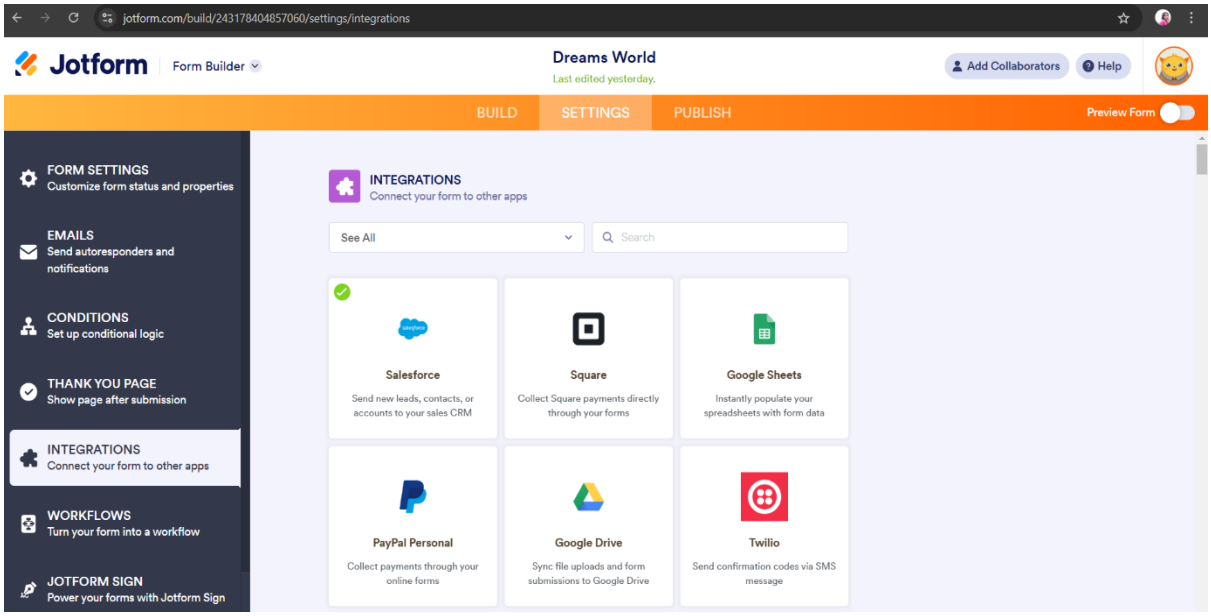


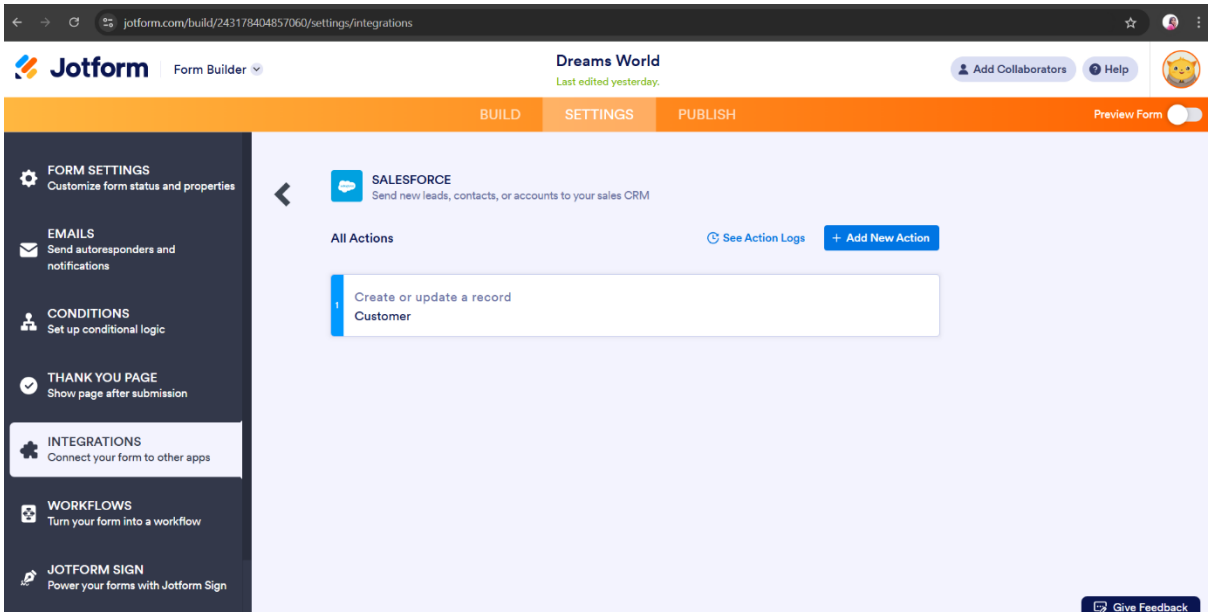
- Create Property object



Milestone 3: Integrate Jotform with Salesforce Platform

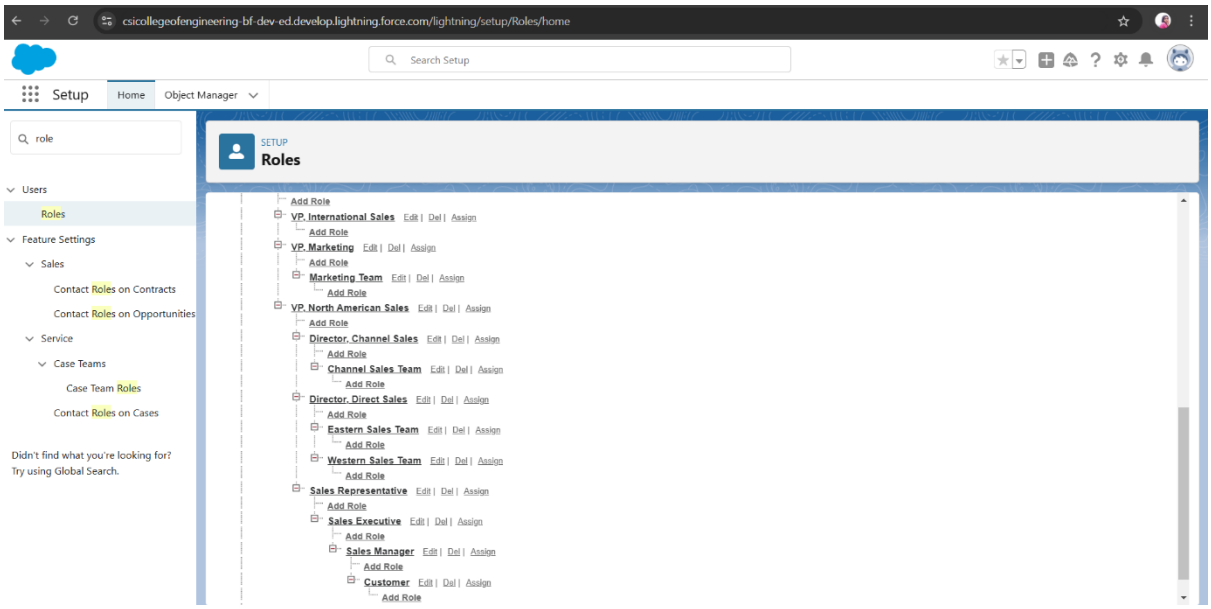
Integrate jotform with salesforce to create customer records





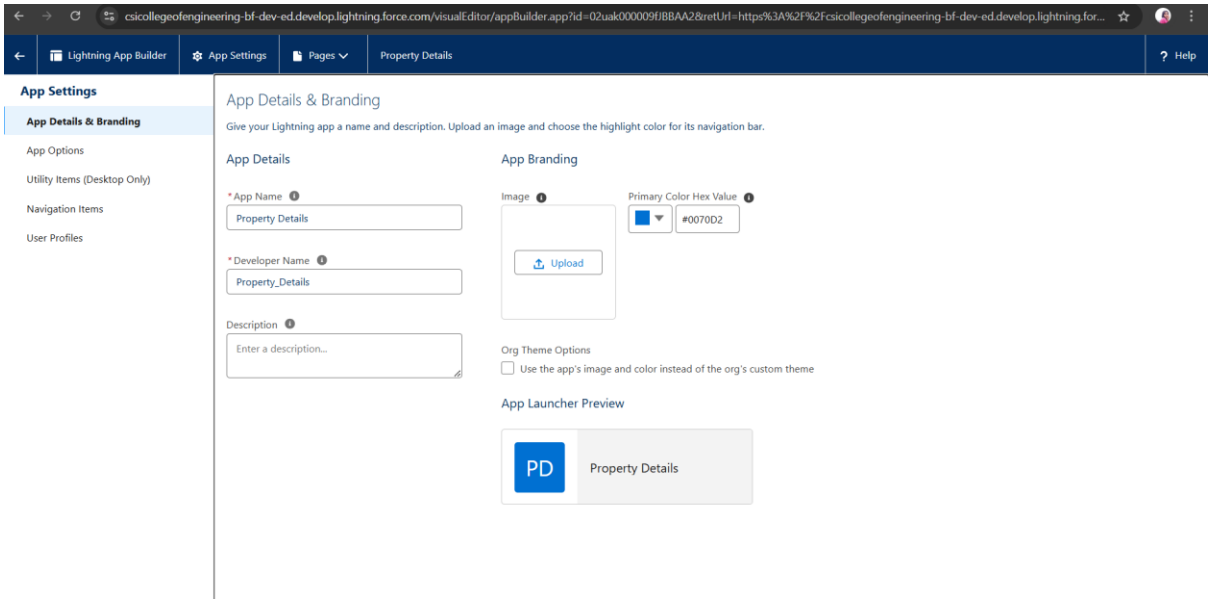
Milestone 4: Create Roles

Create and assign roles for different user levels Sales Executive, Manager, Customer



Milestone 5: Create a Property Details App

Create a Lightning App named *Property Details* in Salesforce, integrating the *Customer* and *Property* objects. Assign the System Administrator profile for app access and management .

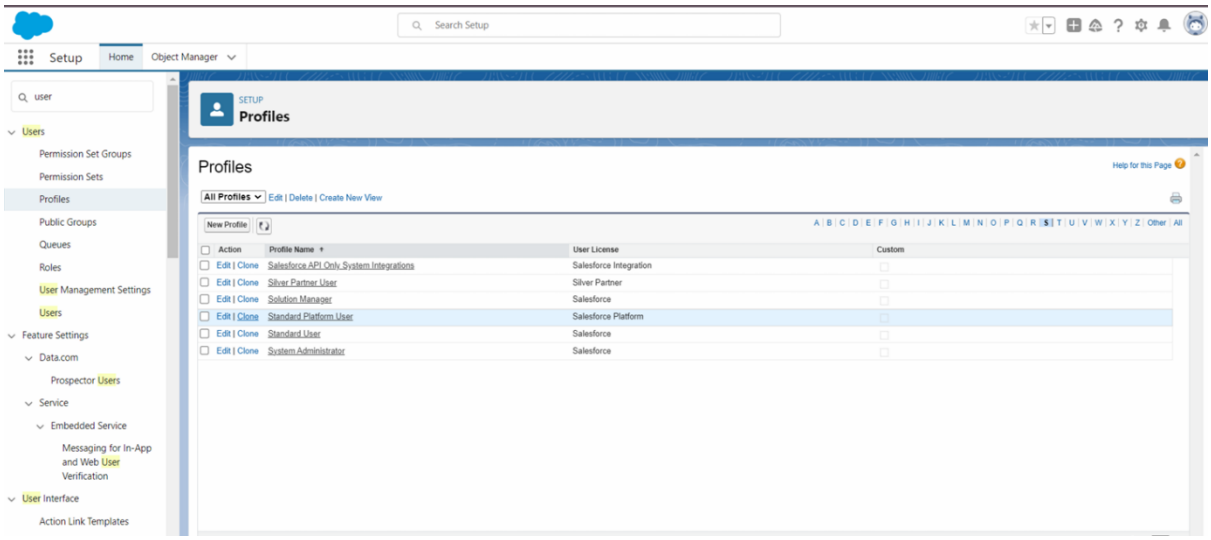


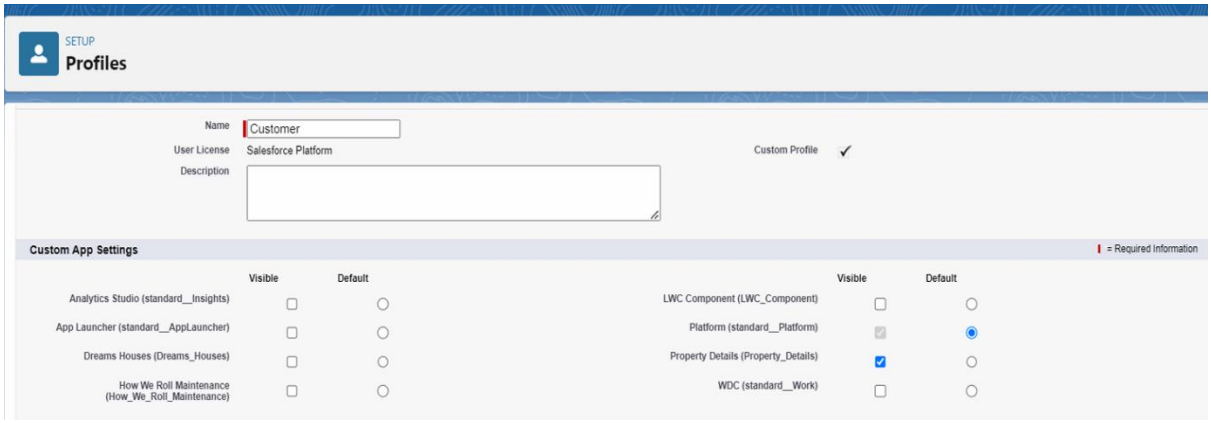
Milestone 6: Create Profiles

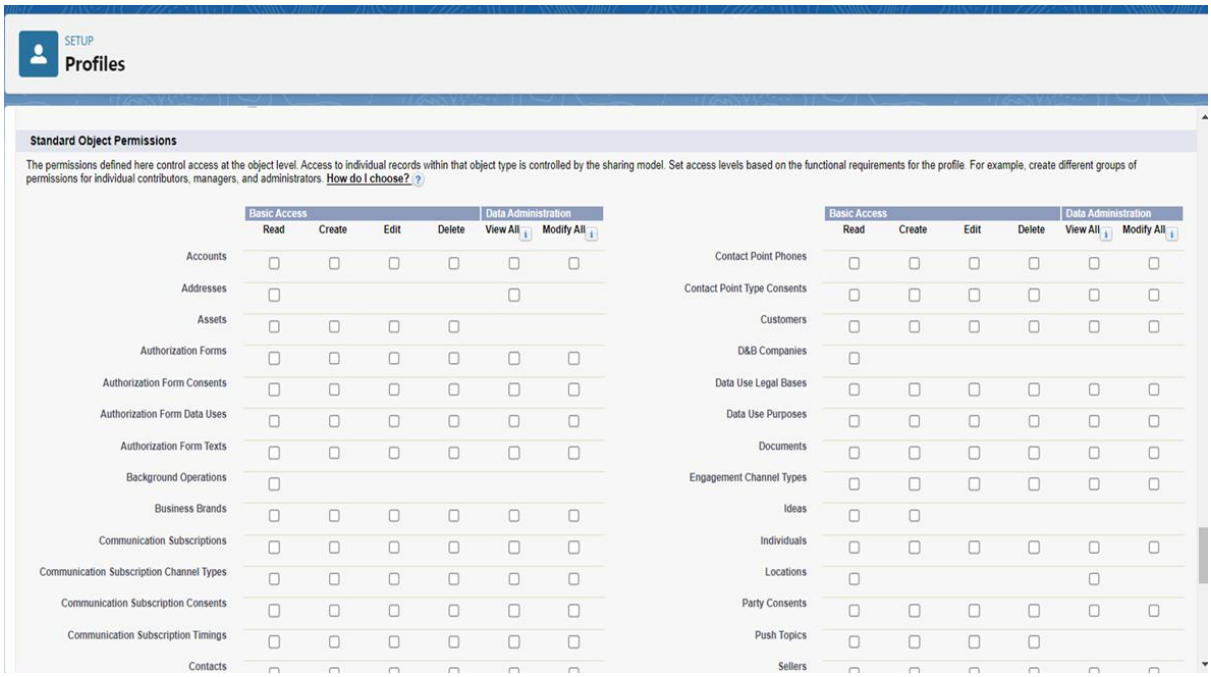
Clone the Salesforce Platform User profile to create Customer and Manager profiles.

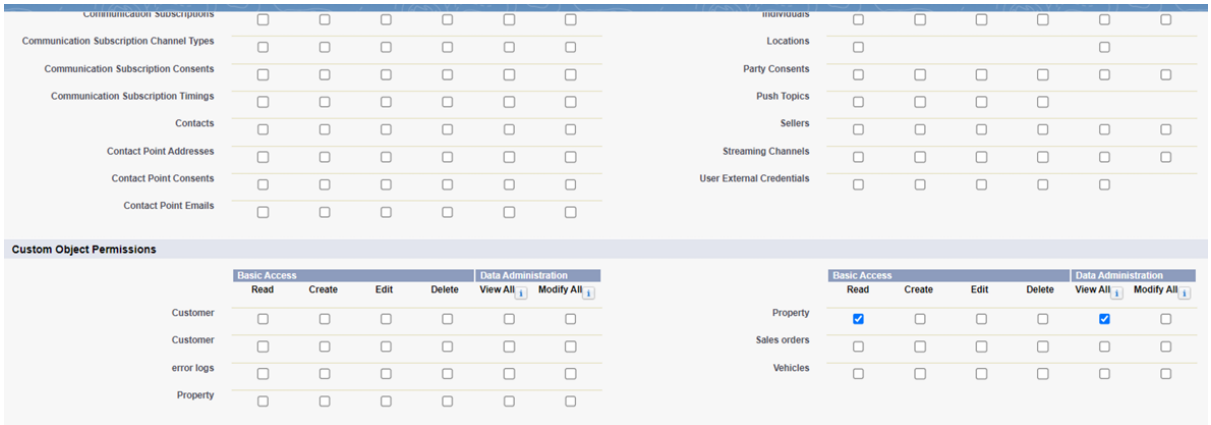
Customer:

For Customer, restricted access to only the *Property Details* app, removed standard object permissions, and allowed read-only access to the *Property* object.



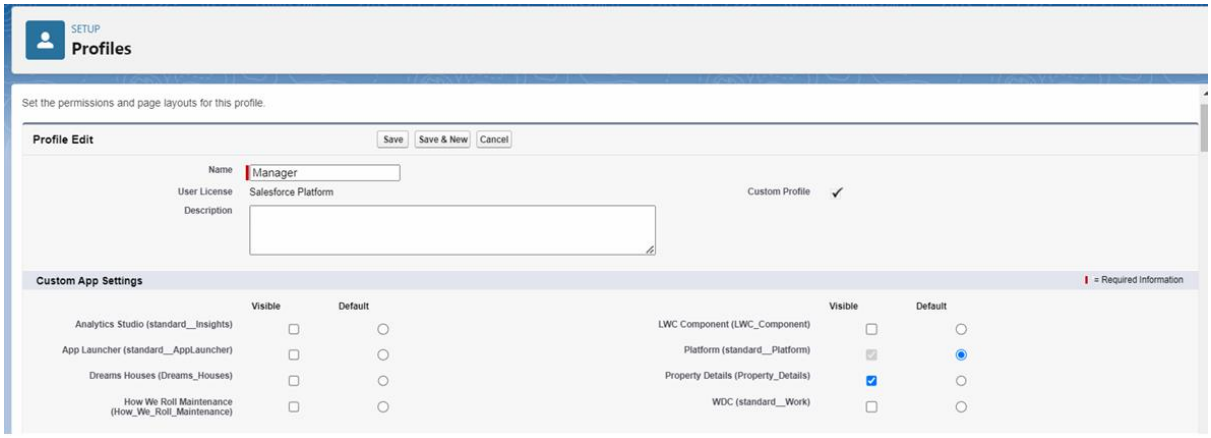


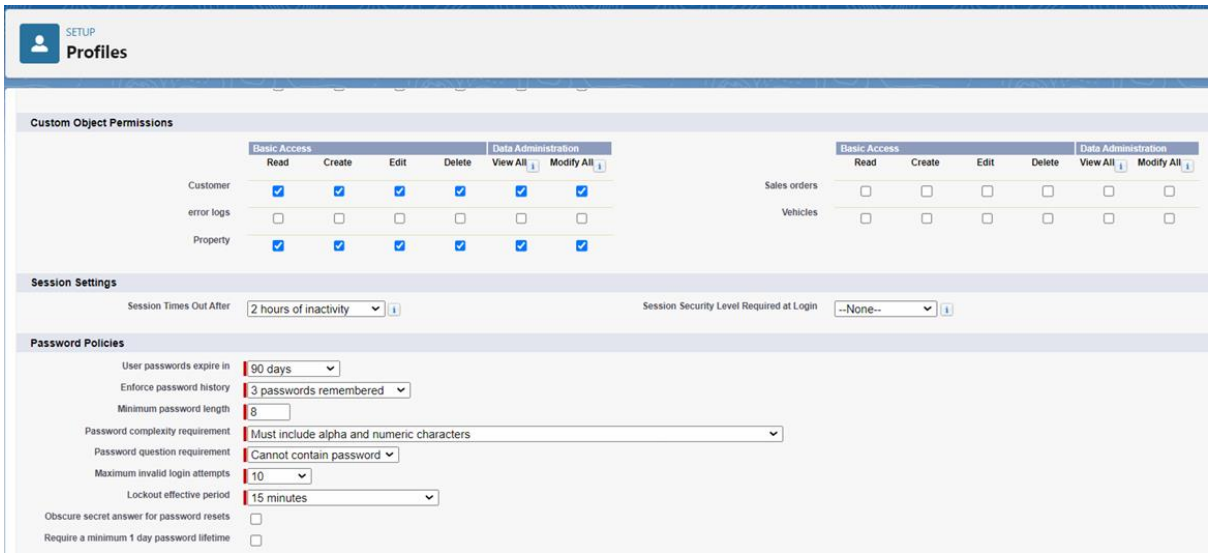




Manager:

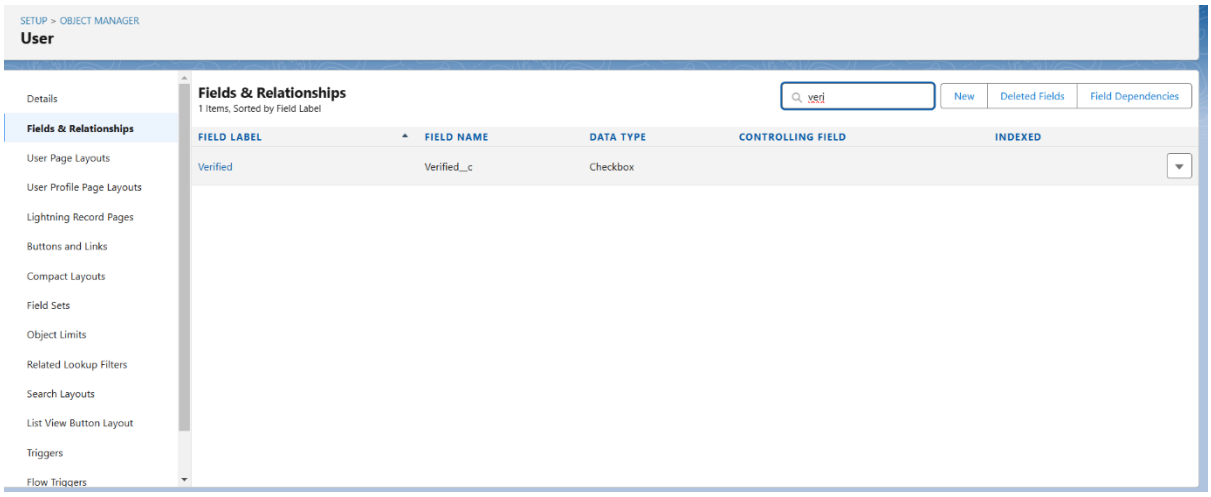
For Manager, restricted access to the *Property Details* app, removed standard object permissions, and granted full access (modify all) to both *Property* and *Customer* objects.

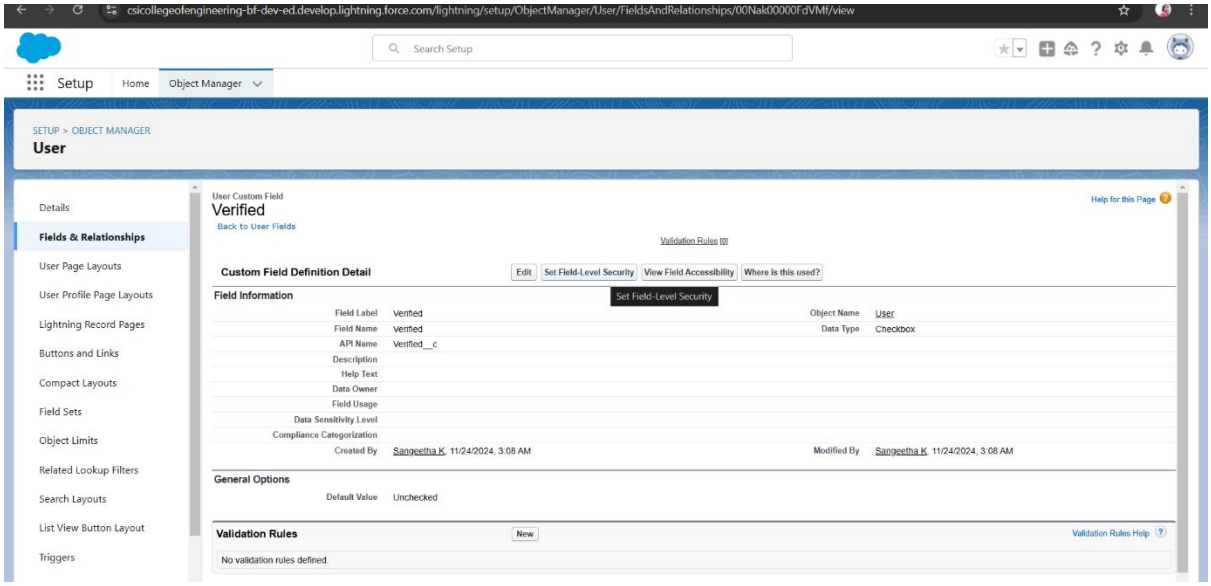




Milestone 7: Create a CheckBox field on user

Create a new custom field "Verified" in the User object under Fields and Relationships with the data type set to Checkbox.

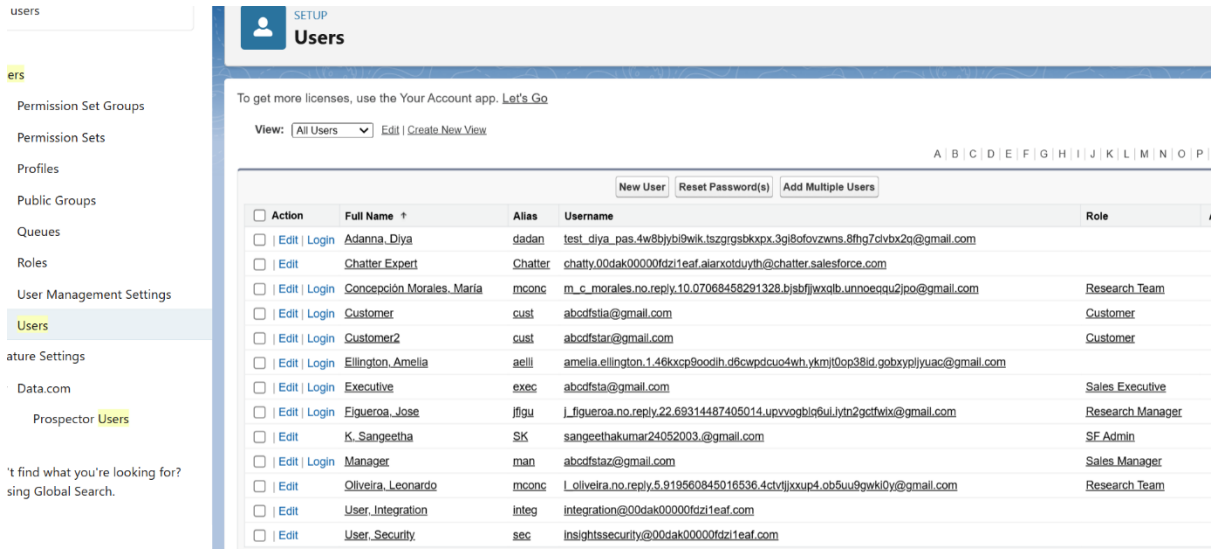




Milestone 8: Create Users

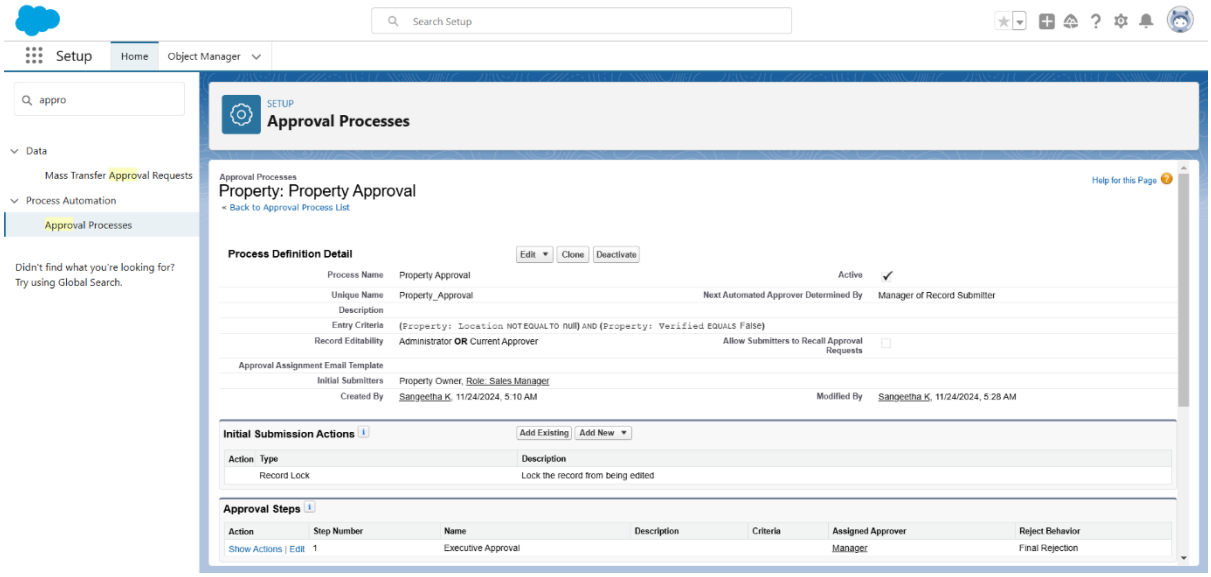
Create four users with different roles and profiles in Salesforce:

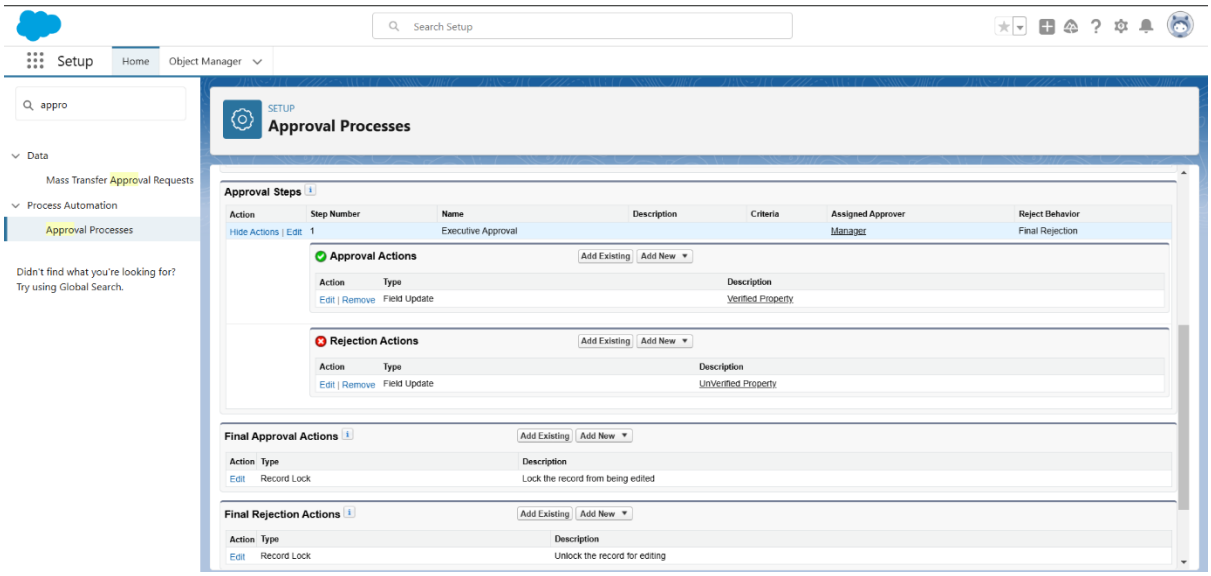
1. User 1 (Executive): Assigned Sales Executive role, System Administrator profile, and Salesforce license.
2. User 2 (Manager): Assigned Sales Manager role, Manager profile, and Salesforce Platform license.
3. User 3 (Customer): Assigned Customer role, Customer profile, and Salesforce Platform license, with the Verified checkbox unchecked.
4. User 4 (Customer2): Assigned Customer role, Customer profile, and Salesforce Platform license, with the Verified checkbox checked.



Milestone 9: Create an Approval Process for Property Object

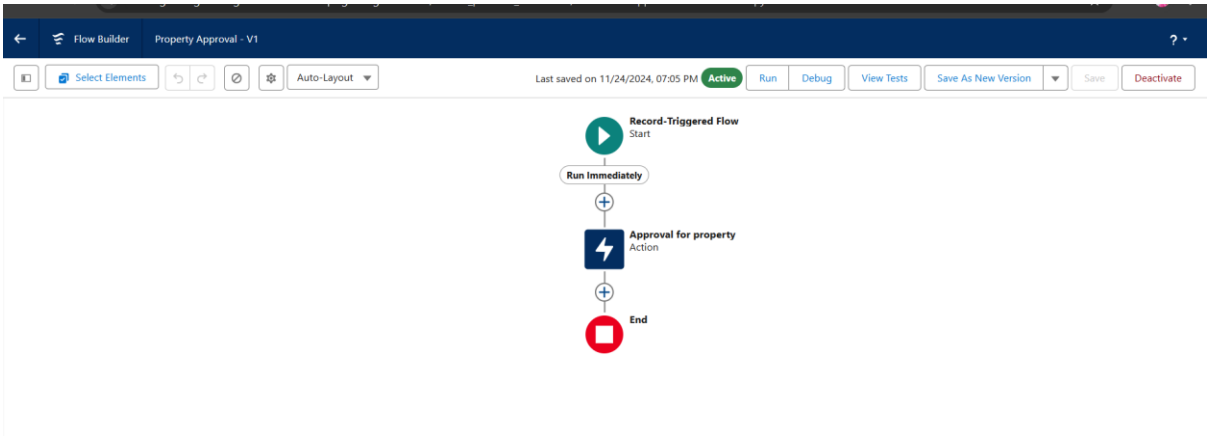
An Approval Process named *Property Approval* was created for the *Property* object in Salesforce. The process was configured with two key criteria: Location is not blank and Verified is false. The Sales Manager was designated as the automated approver, with permission for Administrators or the approver to edit records during the process. The approval page layout was customized to display essential fields like Property, Owner, Location, and Type. Sales Manager was set as the initial submitter, and Sales Executive as the final approver. Field updates were added to mark properties as Verified (True) or Unverified (False), and the approval process was activated for use.





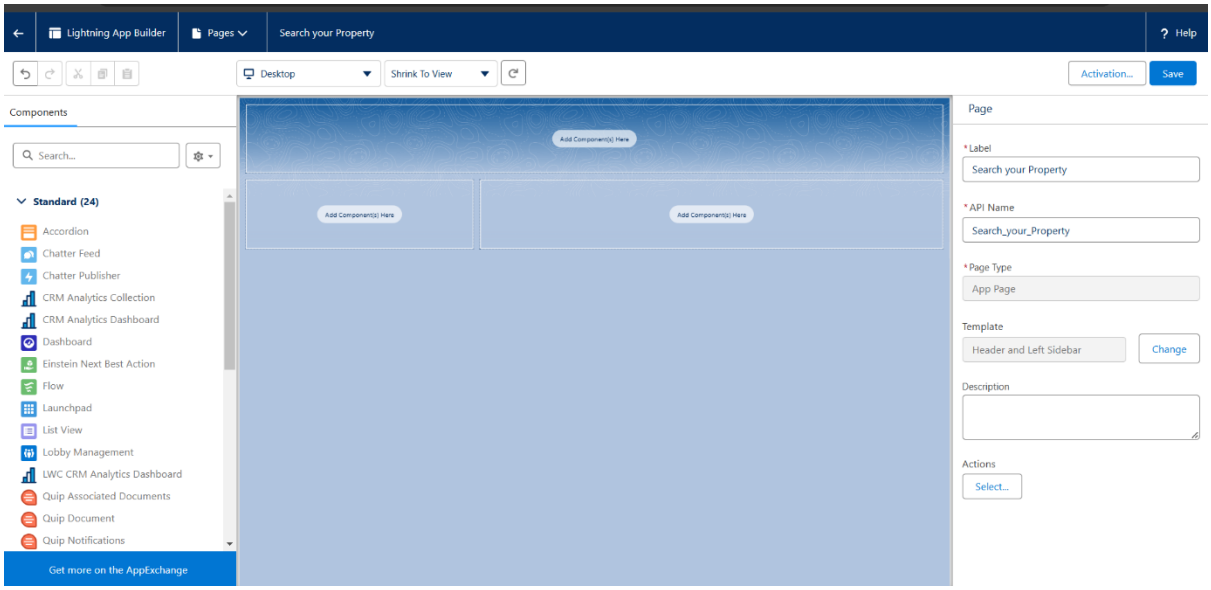
Milestone 10: Create a Record trigger flow to submit the Approval Process Automatically

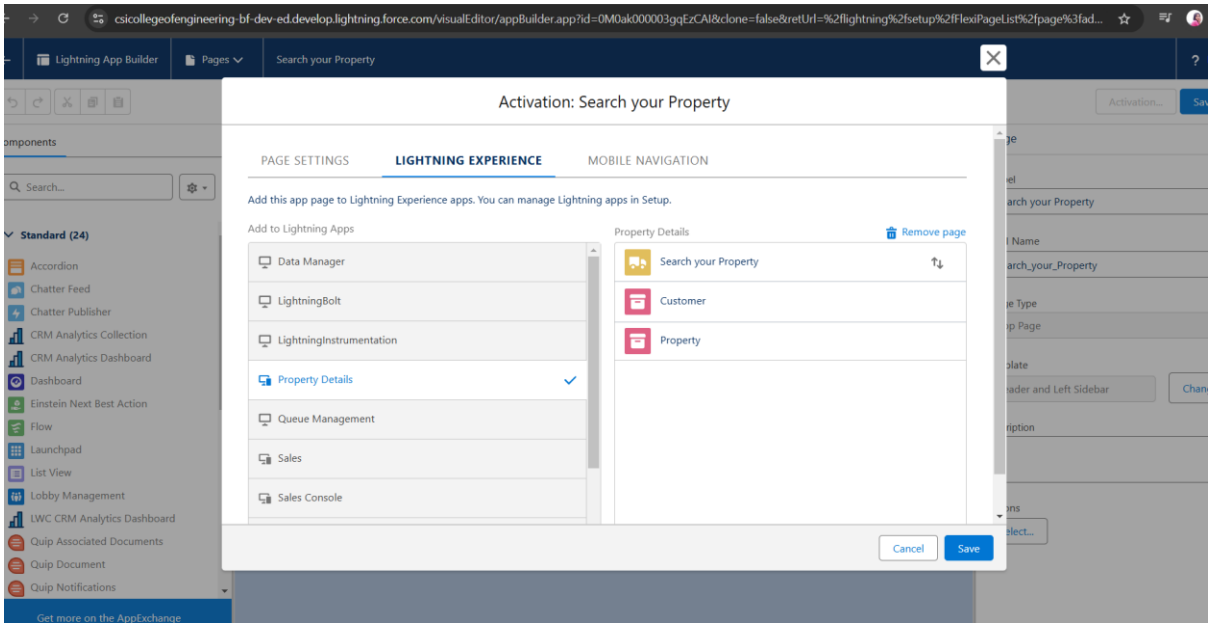
Created a Record Trigger Flow for the *Property* object in Salesforce to automate the submission of properties for approval. The flow was triggered when a record is created, with no entry conditions specified. The action added was "Submit for Approval", with the label "Approval for Property" and the Record ID set to {!$Record.Id}. After configuring the flow, it was saved with the label "Property Approval" and activated for use. This automation ensures properties are automatically submitted for approval when they are created.



Milestone 11: Create an App Page

Created an App Page for the *Property Details* object, titled "Search Your Property", using Lightning App Builder. The page was configured with the Header and Left Sidebar components. After saving, it was activated and set to be available for all users. The page was added to the Property Details app in Lightning Experience. The page setup allows users to search properties directly within the app. All configurations were saved and the app was made accessible to users, streamlining property searches. This integration enhances user navigation and property management.





Milestone 12: Create a LWC Component

Create an Lwc Component for the customers so that only verified customers can access the verified properties and non Verified customers can access non verified properties, and deploy it on “Search your Property Page”

1.Create an Apex Class and make it aura enabled and name it “PropertHandler_LWC” and make it aura enabled,then in VSCode,authorize your org,create a Lighting Web Component and write the provided code.

File Edit Debug Test Workspace Help < >

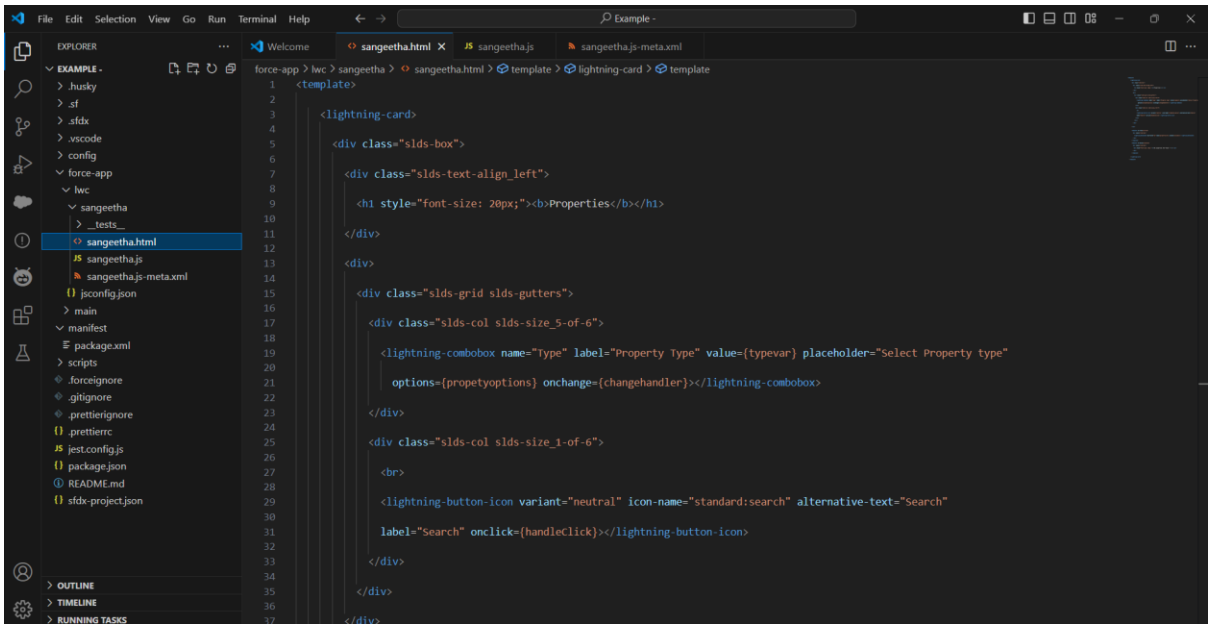
PropertHandler_LWC.apxc

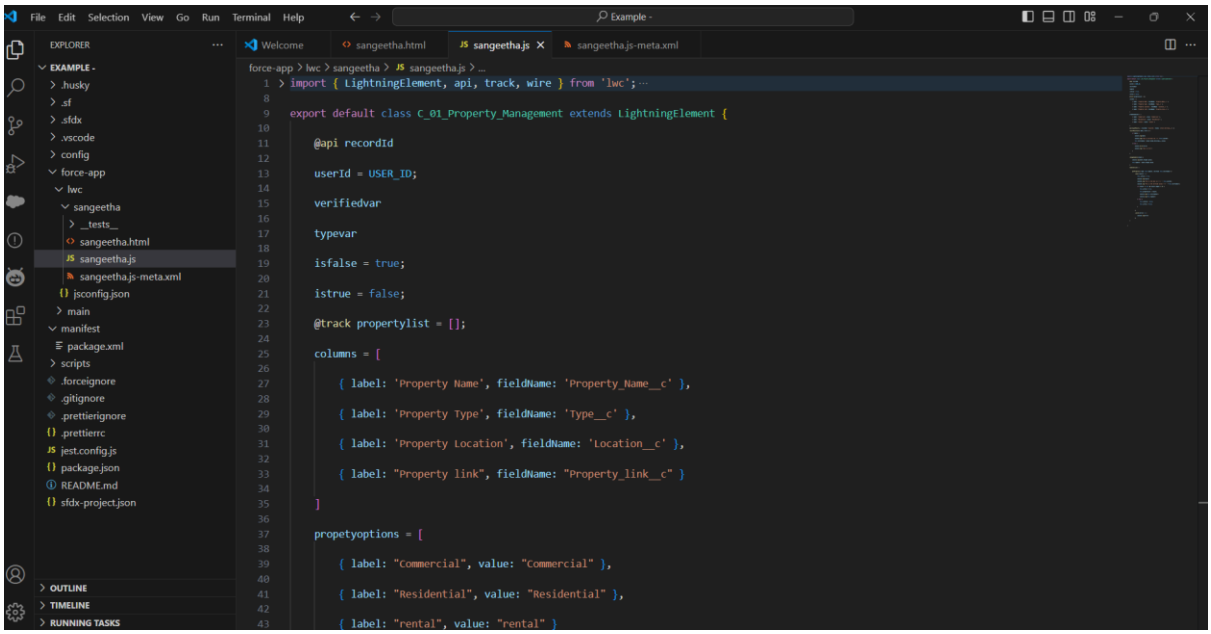
Code Coverage: None API Version: 59 Go To

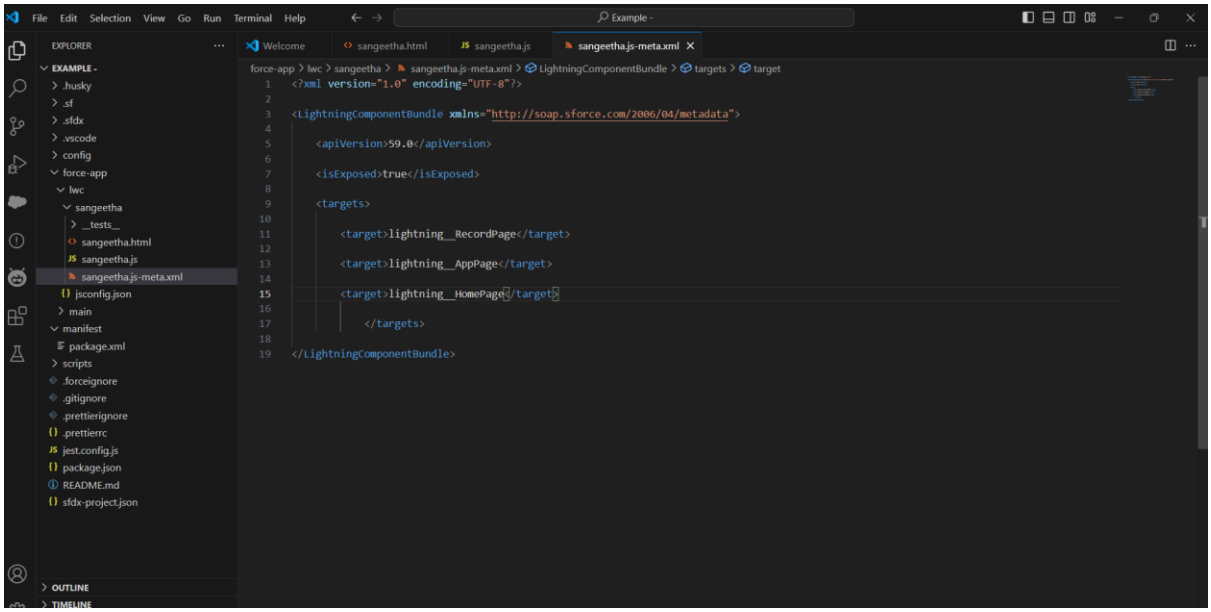
```
public class PropertHandler_LWC{
    @AuraEnabled(cacheable=true)
    public static list<Property__c> getProperty(string type , boolean verified){
        return [SELECT Id, Location__c, Property_Name__c, Type__c, Verified__c FROM Property__c Where Type__c =: type AND Verified__c =: verified];
    }
}
```

Logs Tests Checkpoints Query Editor View State Progress Problems

User	Application	Operation	Time	Status	Read	Size

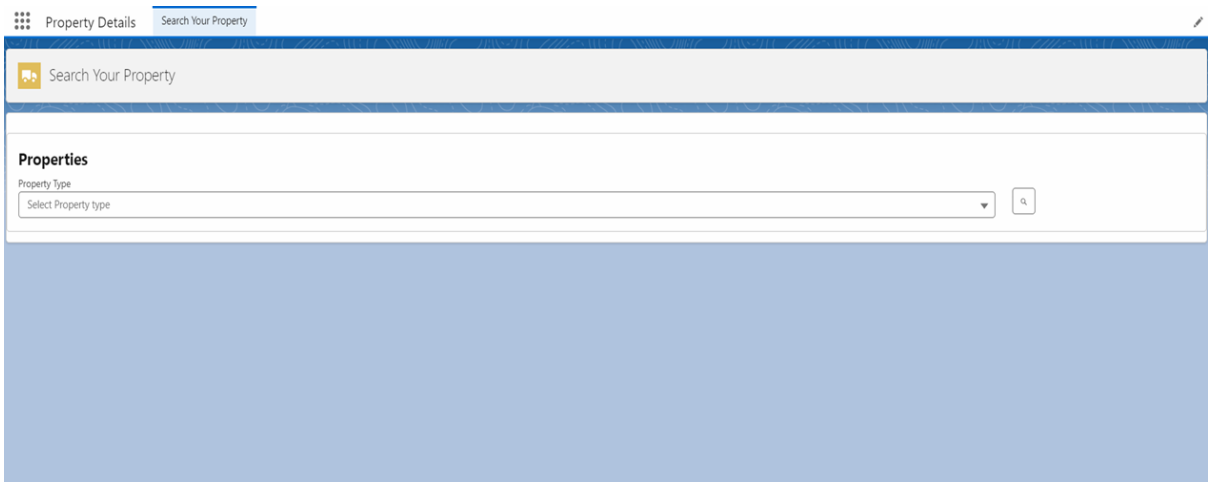






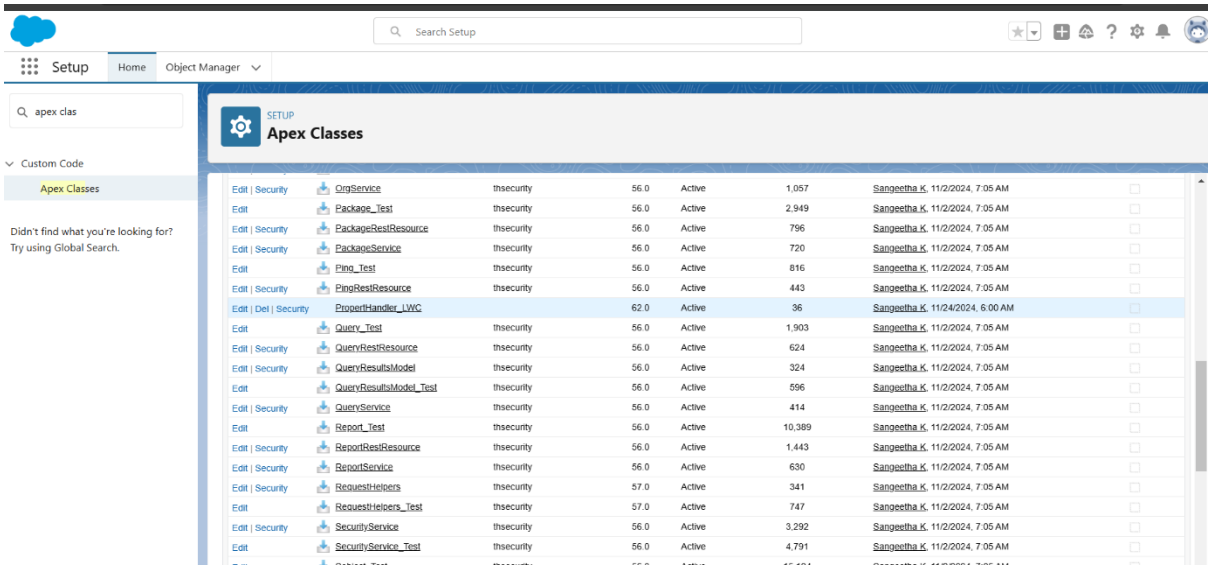
Milestone 13: Drag this Component to your App Page

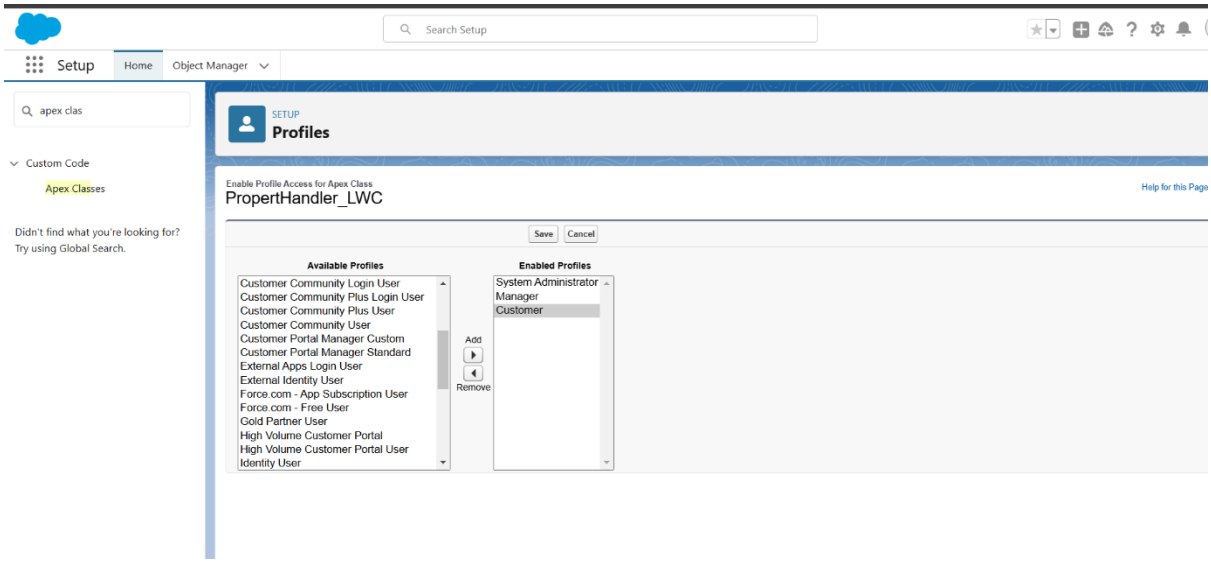
Navigate to App Launcher in Salesforce and searched for Property Details.
Click the gear icon and selected Edit Page.
Drag the Lightning Web Component to the app page and save the changes.



Milestone 14: Give Access of Apex Classes to Profile

Navigate to Setup Apex Classes and click on "Security" next to PropertyHandler__LWC.In the Profiles section, added Manager and Customer profiles for access.Save the changes to grant the necessary profiles access to the PropertyHandler__LWC Apex class.





Conclusion

The CRM application to Handle the Clients and their property Related Requirements aims to streamline customer interactions, enhance user experience, and optimize property management operations through Salesforce integration. With scalable architecture and automated workflows, the platform will support future growth and improve customer satisfaction.