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 optimizesoperations, improving customer engagement and facilitating growth in the real estate market.

1. Client Management
    a. Add,update, and delete client details.
    b. Trackclient preferences, budget, and location interests.
    c. Maintain contact details and communication history.

2. Property Management
    a. Manage property listings with details like type, price, location, and features.
    b. Trackproperties available for sale, rent, or lease.
    c. Upload photos and documents for properties.

3. Requirement Matching
    a. Match client requirements with available properties using filters.
    b. Notify clients about new properties that fit their criteria.

4. Lead Tracking
    a. Manage inquiries and follow upwith potential clients.
    b. Schedulemeetings and site visits.
    c. Assign leads to specific team members.

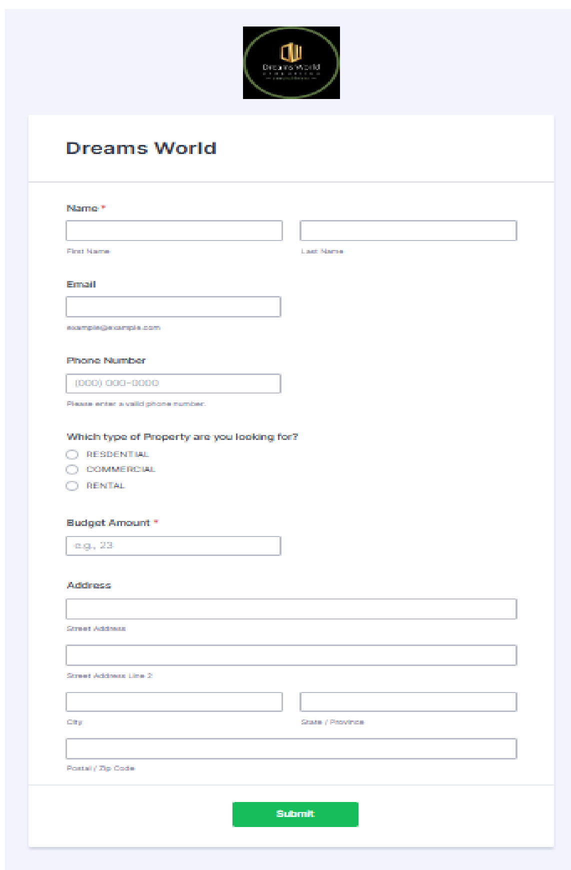
Milestone 1: Create a Jotform and integrate it with theorg to create a record of customers automatically.

Client wants a form for the customers to get the details directly into the salesforce so that the admins can create a user in the org.Client wants a form for the customers to get the details directly into the salesforce so that the admins can create a user in the org.

Activity1

Open your browser and search for jotform and log in.

1. After login click on create form andclick on start from scratch
2. Now create a form to get the customer details like Name, Phone,EmailI, Address andtype of property the customer is interested in.
3. Once the form is created, publish it by clicking on publish.
4. form link :https://www.jotform.com/build/243218794664063



Create Objects from Spreadsheet

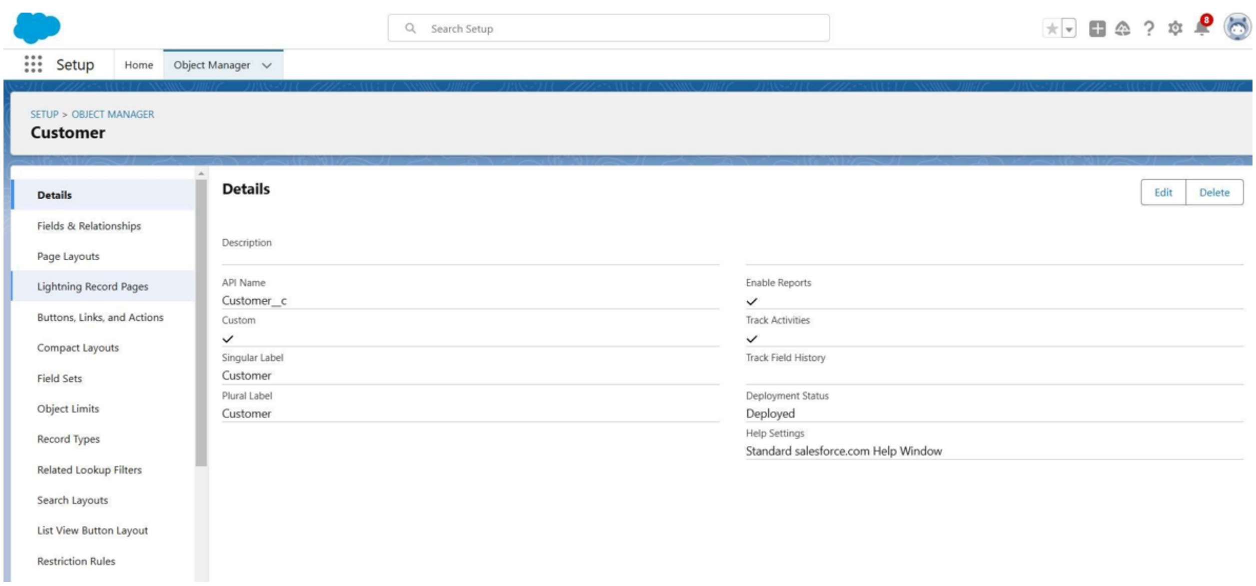
Directly Creating Objects from Spreadsheet in Salesforce

Creating Customer Object :

1. Go to your object manager and and click oncreate object from spreadsheet.
2. Click on the link to get the spreadsheet
3. customer

Customer	Phone Number	Emial	State	Property Type	Budget Amount	Street Address	Street Address	City	postal code	Verified
Rakesh	788797	rakesh@gmail.	Telangana	Residential	4000000	gb road	street no 45	Hyderabad	555001	checked
prakash	55448855	p@gmail.com	Maharashtra	Commercial	8000000	gachibowli	indira road	mumbai	6600014	unchecked
Prajwal	454545	prajwal@gmail.	Maharashtra	Rental	25000	kamdli	kathora	Amravati	444805	checked

After downloading, upload the file, map the fields and upload to create an object.

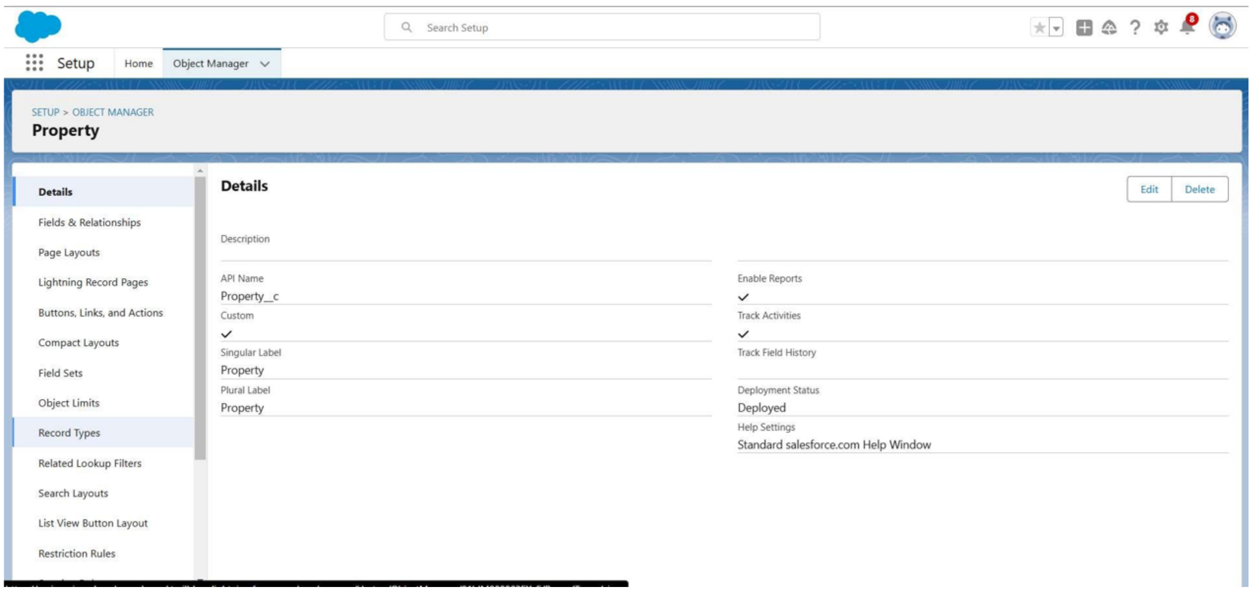


Creating Property Object :

1. Follow the same from the customer object tocreate theProperty Object
2. Property

A	B	C	D
Property Name	Type	Location	Verified
Lotus Appartme	Residential	hydeerabad	checked
500000 sq.ft pl	Commercial	Amravati	uchecked
3 Bhk fkat at st	rental	Jubliee hill Hyd	Checked

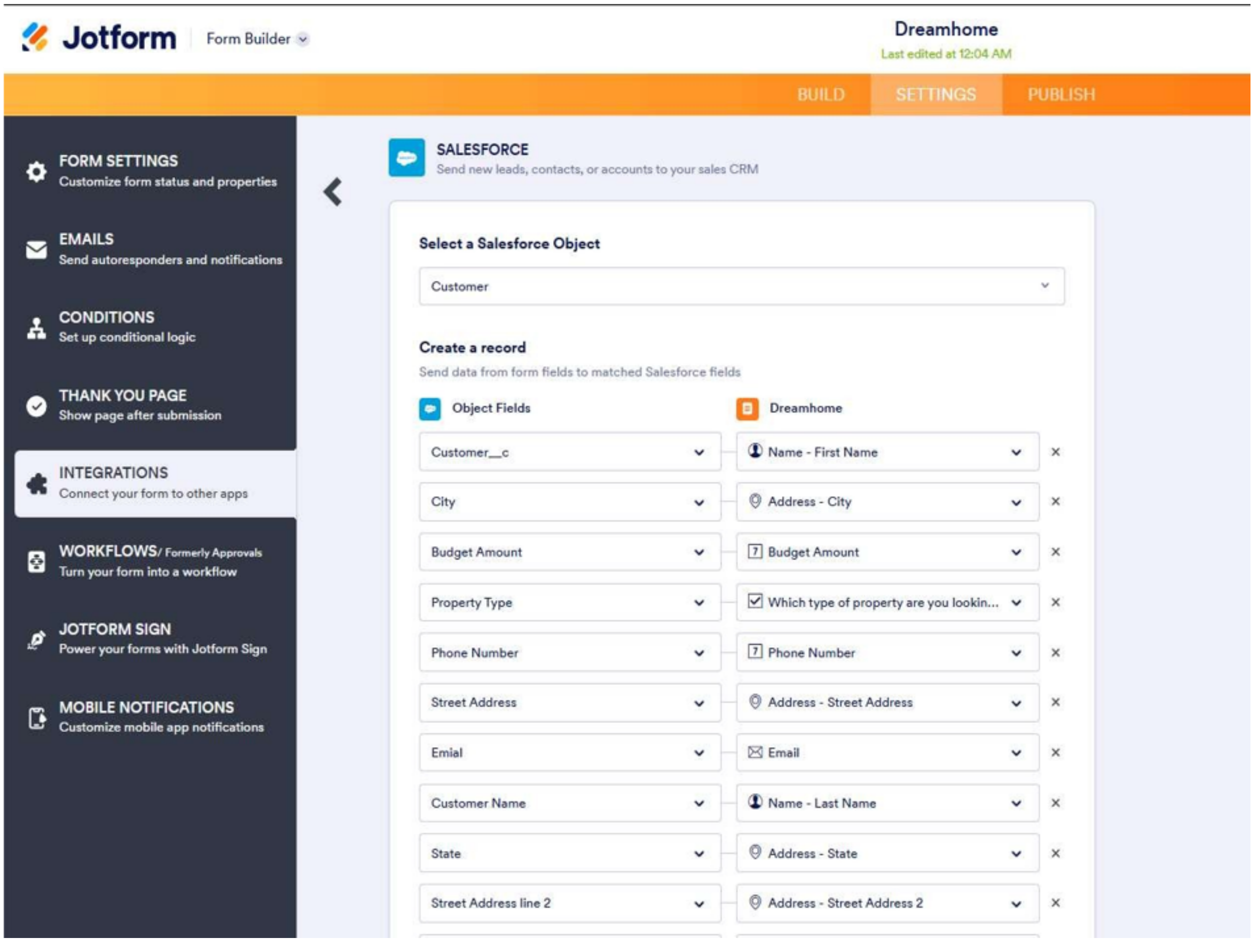
After downloading, upload the file, map the fields and upload to create an object. the filedsas follows



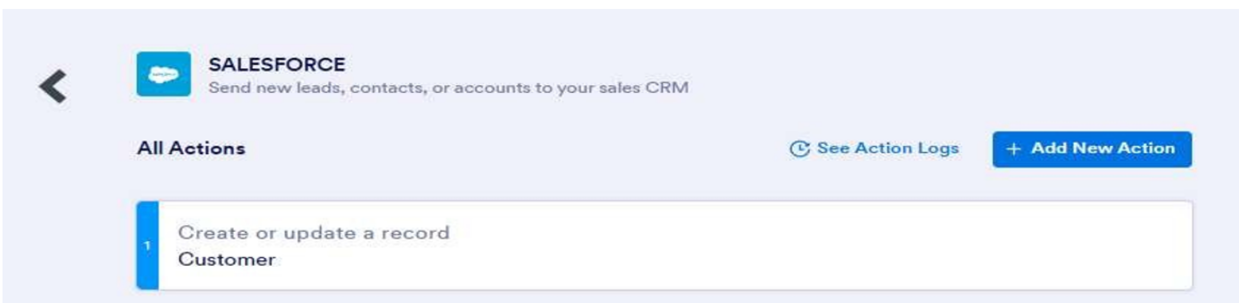
Activity

1. On the Jotform Platform, Click on Integration and choose Salesforce
2. Click on User Integration and choose "Add to From"
3. Select the Org with which you want to Integrate your jotform with and select your account
4. Select an Action -Create a record.
5. Select a Salesforce Object : - Customer

Map Each and every field on the Object with the fields on the form and "Save Action".



Then “Save the Integration” and “Finish”.

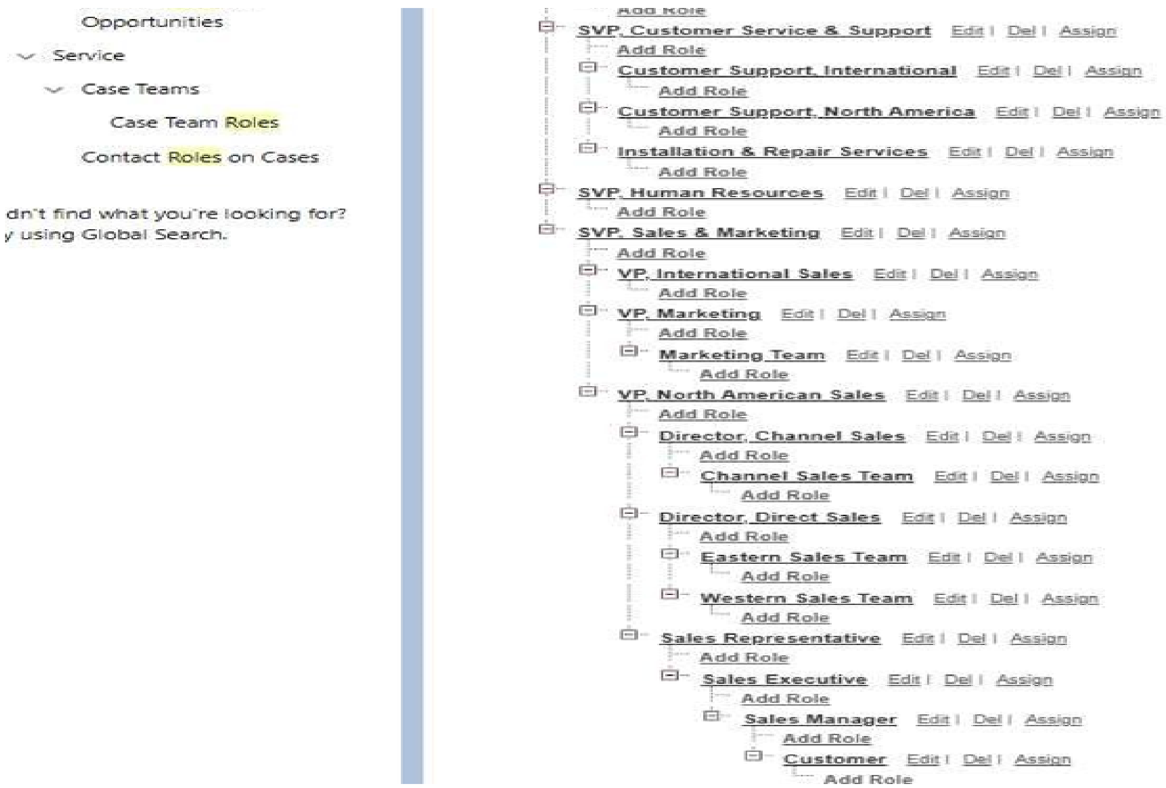


Create Roles

here we need to Create Roles as per business requirenment

Activity:- 1

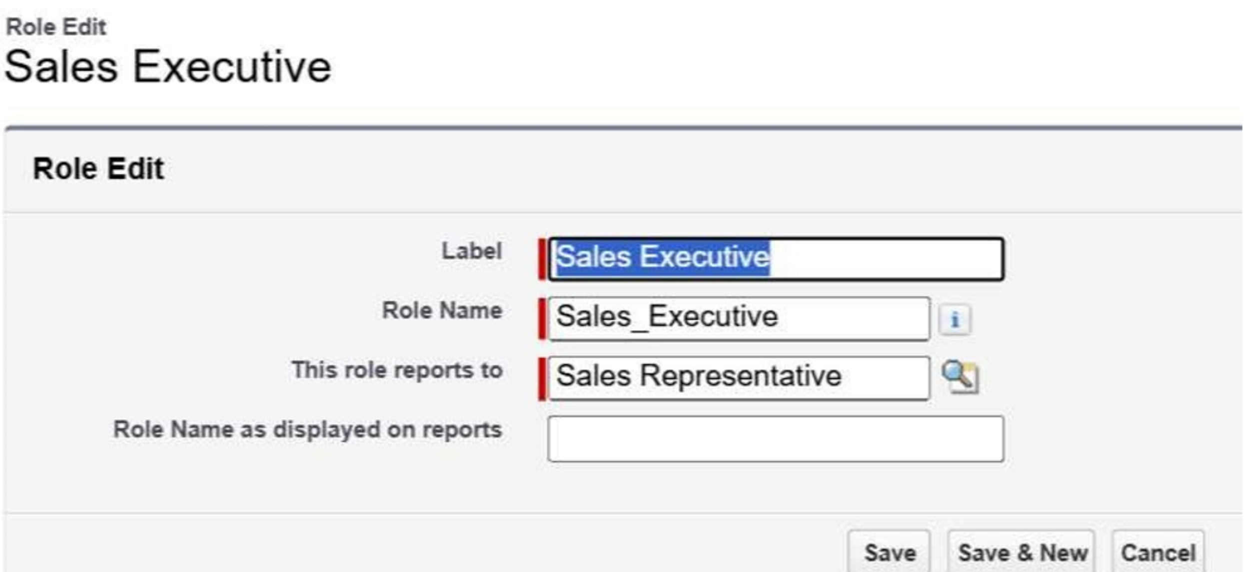
1) Go to Setup and Click on Roles, then click on Expand all and Add a Role just below the Sales Representative



* It will use the "System Administrator Profile".

2) Label - Sales Executive

Reports to - Sales Representative



Similarly Create a Role Name “Sales Manager” below Sales Executive which reports to Sales Executive, Also Add a Role below Sales Manager labeled as “Customer” which reports to Sales Manager.

Create a Property Details App

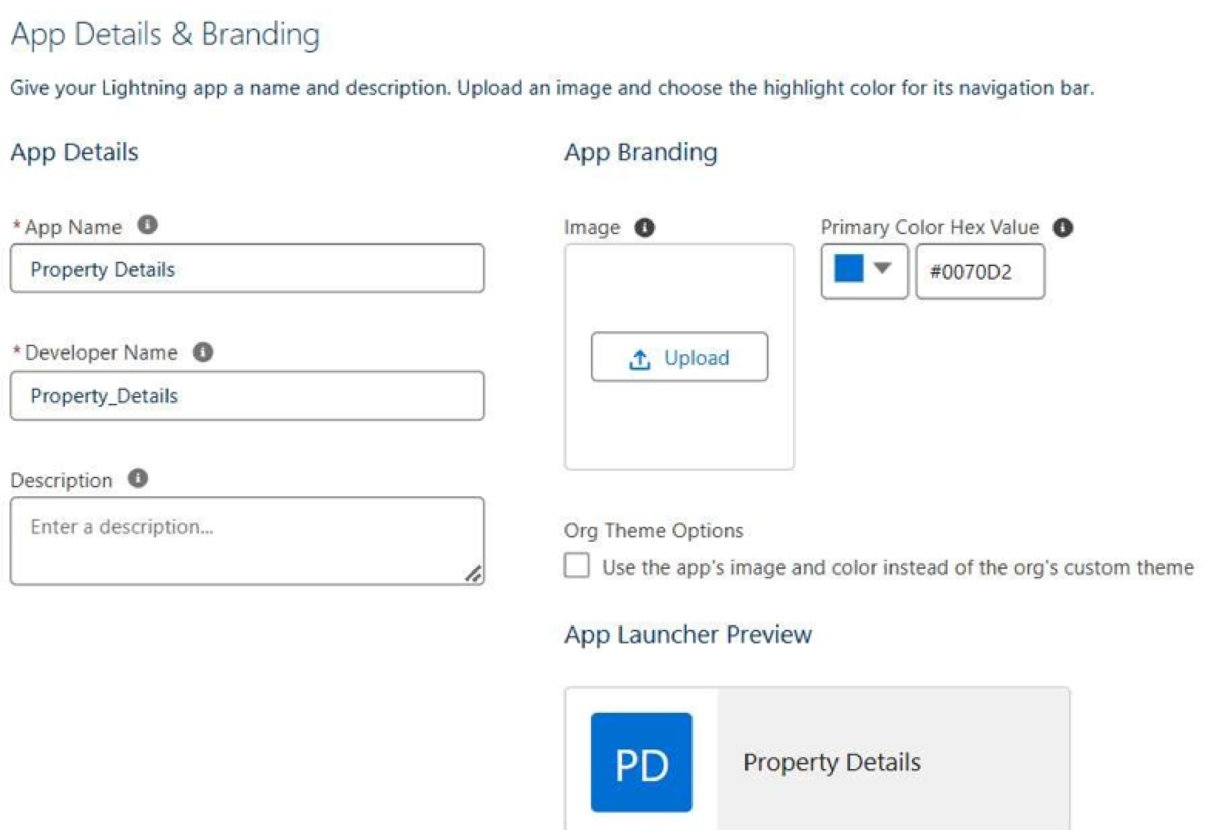
An App where the objectswill be displayed

Activity1

1. From Setup>» Go to AppManager and click on New Lightning App and Nameit as "Property Details" and add "Customer" and "Property" Object.
2. Click Next >» Next >» Save and Add “System Admin "Profile.

App Details & Branding

Give your Lightning app a name and description. Upload an image and choose the highlight color for its navigation bar.

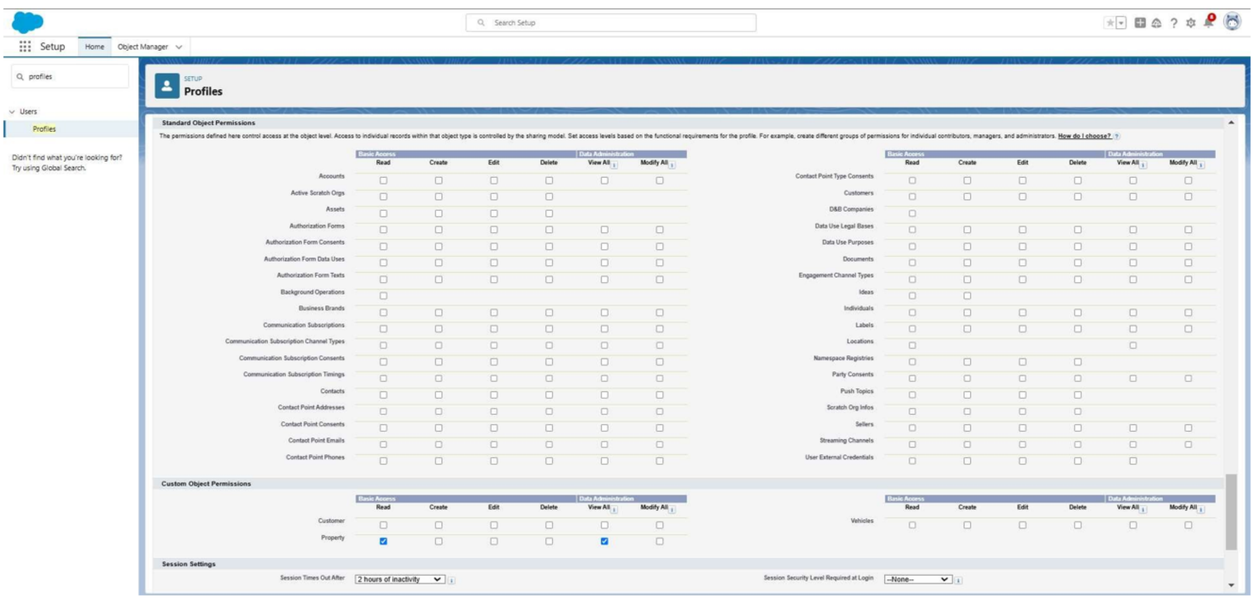


Create Profiles

Create profiles as per business requirement

Creating Customer Profiles

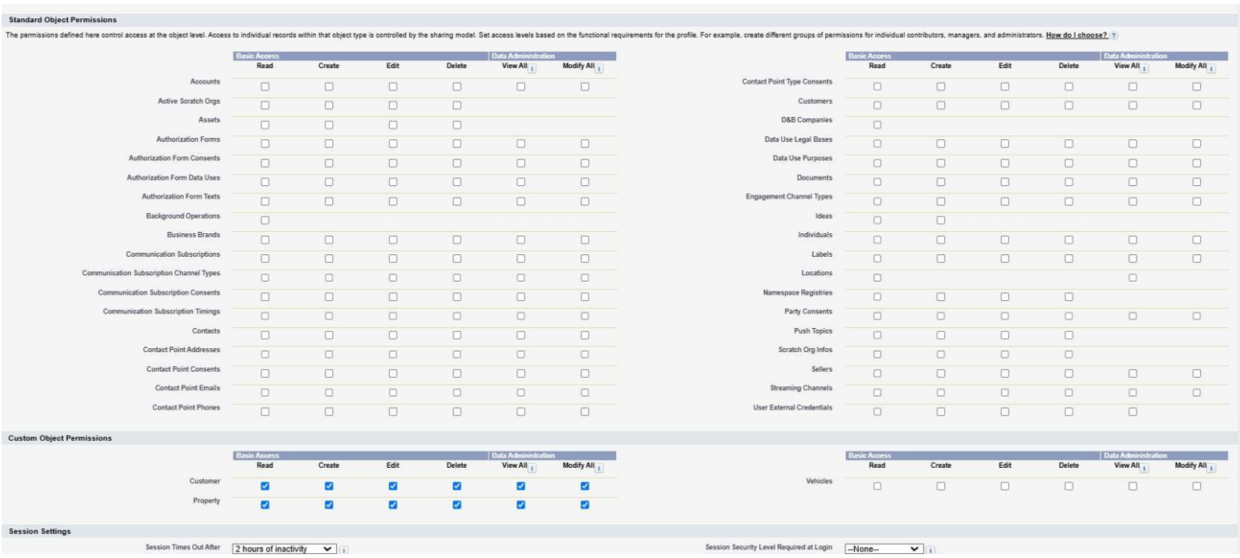
1. From Setup? Go to Profiles and Clone (standard pIatform)SaIesforce Platform User and Name it “Customer”..
2. Uncheck allthe Custom Objects and Check onlyProperty Details From Custom App Settings.
3. so RemovealltheStandard Object Permissions
4. Uncheck all the Custom Object Permissions and check read and view all in “Property”
5. make sure every submissionobject permissions are unselected and then save



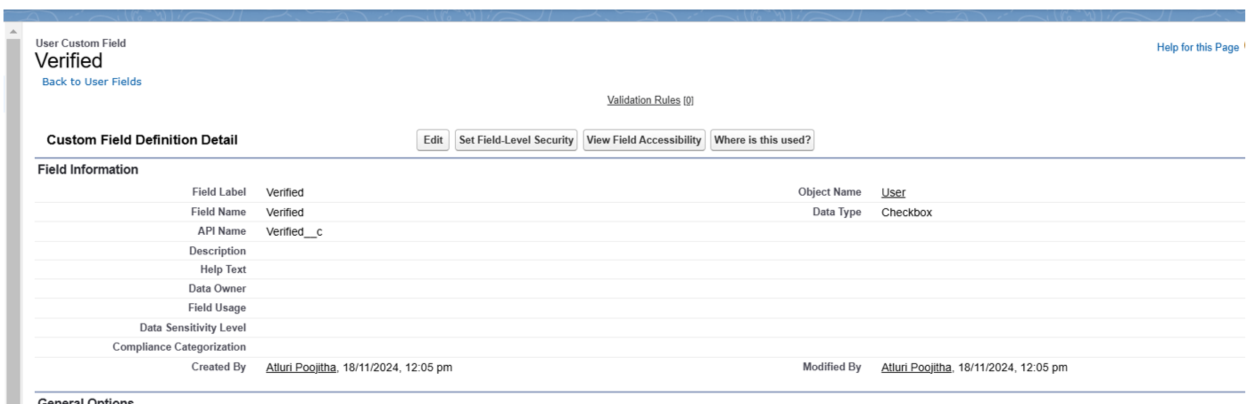
Creating ManagerProfiles :-

1. From Setup » Go to Profiles andClone Salesforce Platform User and Name it “Manager”.
2. Uncheck allthe Custom Objects and Check onlyProperty Details From Custom App Settings.
3. Also Remove all the Standard Object Permissions.

4. Uncheck allthe Custom ObjectPermissions and check only“modify all” from “Property” and“Customer”.



1. Setup >> Object Manager >> Search for User >> Fields and Relationships
2. select the Data type “Check Box”
3. Create new Field Named as “Verified”



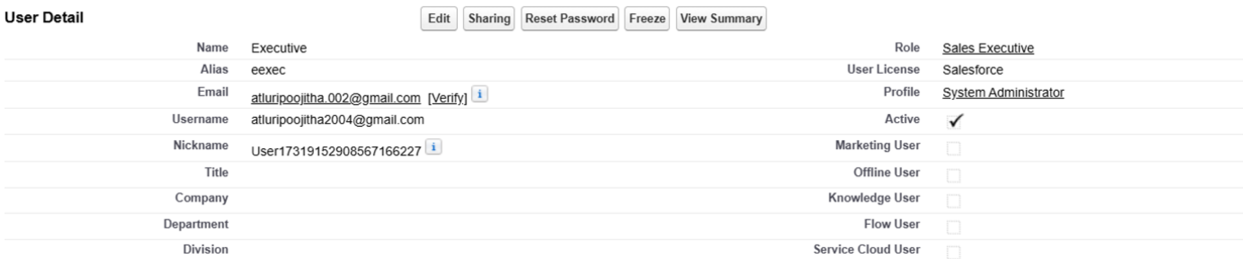
Create Users

Create three differentusers with three different Roles and profilesas we have mentioned above. here we are going to create 4 users

User : 1

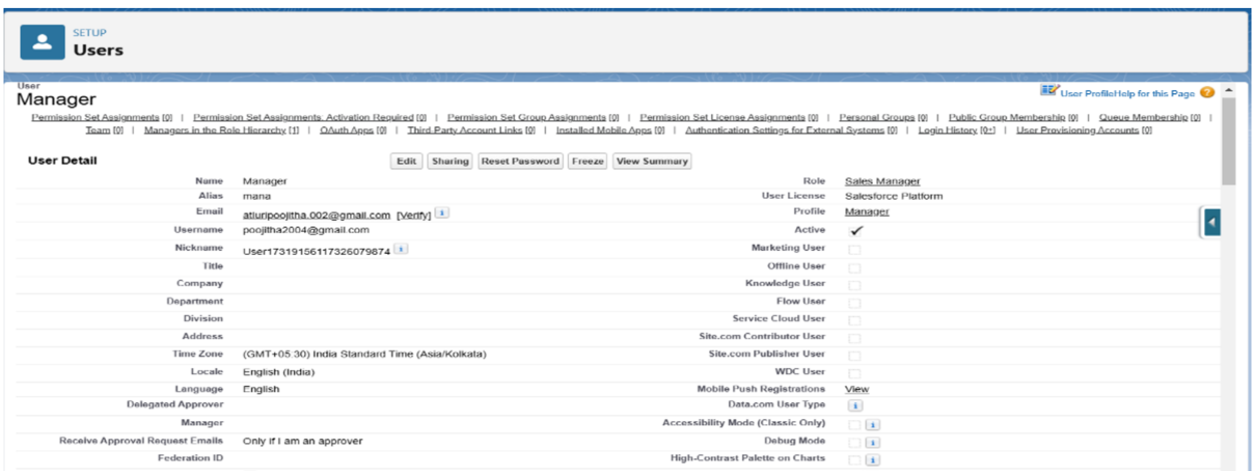
1. Goto Setup --> Administration --› Users --> New User
2. Last Name - Executive

3. Role - Sales Executive
4. License - Salesforce
5. Profile - System Administrator



User : 2

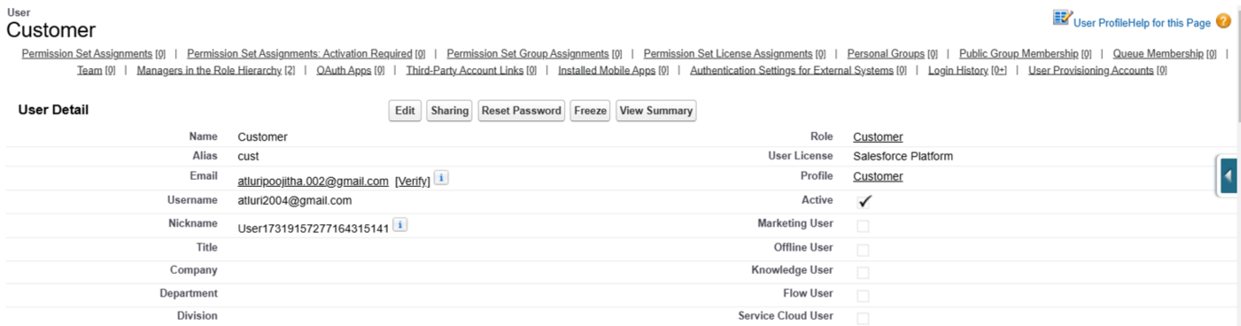
1. Goto Setup > Administration >› Users ›>New User
2. Last Name >> Manager
3. Role >> Sales Manager
4. License >> Salesforce Platform
5. Profile >> Manager
6. Save



User : 3

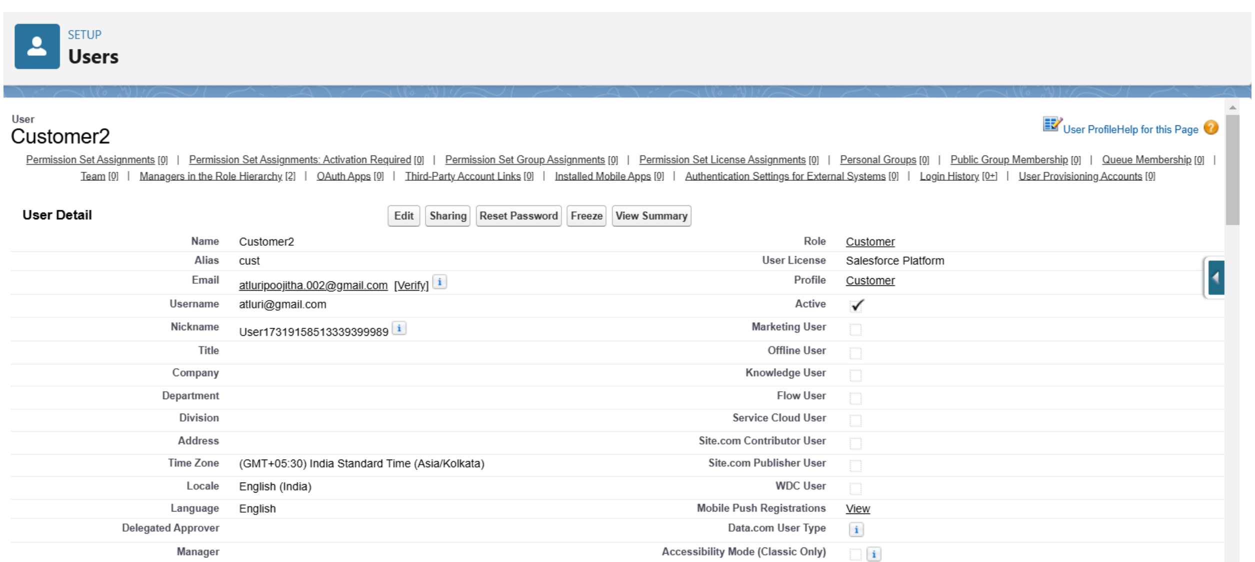
1. Go to Setup»>Administration »> Users »> New User
2. Last Name » Customer
3. Role >> Customer

4. License»>Salesforce Platform
5. Profile»>Customer
6. Make Sure the verifiedcheckboxis"Unchecked"
7. Save



User : 4

1. Goto Setup »> Administration >> Users >> New User
2. Last Name >> Customer2
3. Role >» Customer
4. License >> Salesforce Platform
5. Profile »> Customer
6. Make Sure the verified check box is "checked"
7. Save

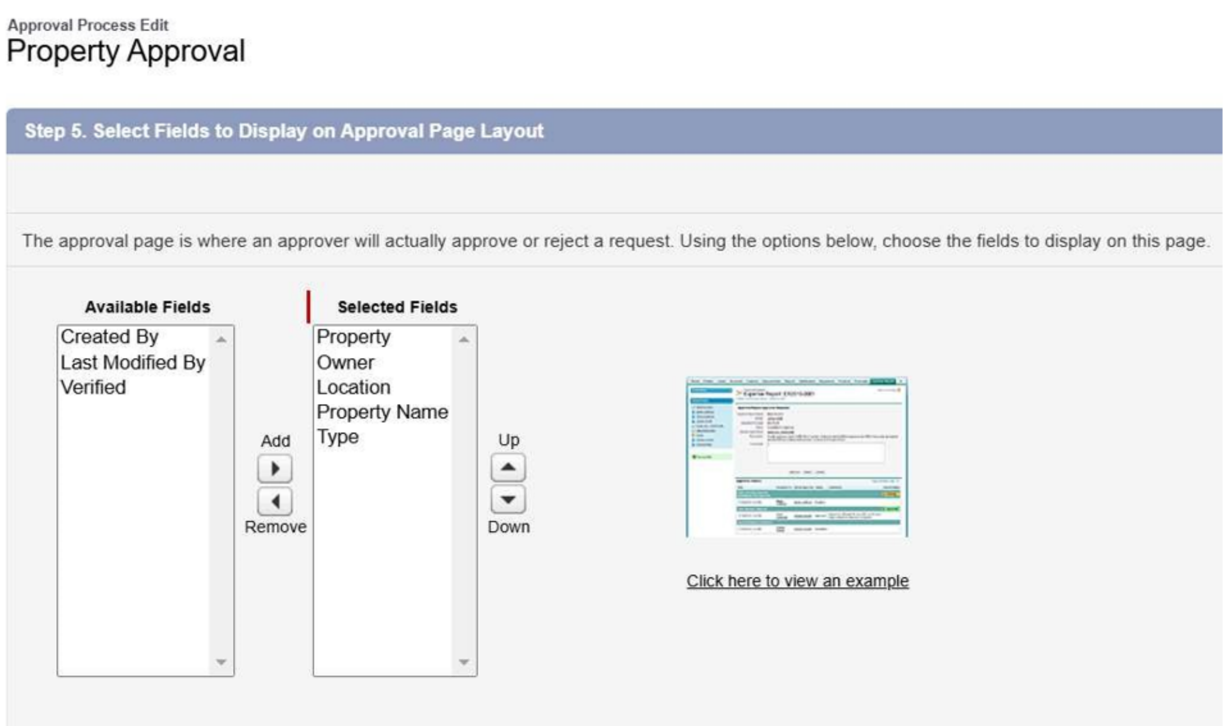


Create an ApprovalProcess for PropertyObject

An Approval processto approve or reject the records as according

Activity1

1. From Setup >>Process Automation >Approval Process
2. before proceeding we need to select property in tne manage approval process
3. Process Name - Property Approval
4. select 2 criteria -
5. Location- i not equal to- blank,
6. Verified- Equals- false
7. Click next and "Next Automated Approver Determined By" Select Manager
8. FromRecord Editability Properties >>Click on AdministratorsoRthe currently assigned approver can edit records during the approval process.
9. FromStep 5. Select Fields to Display on Approval Page Layout select Property, Owner, Location, Type.



1. Click Next and Select the initial Submiters »
2. Owner >> Property Owner
1. Roles>> Sales Manager
2. Save.

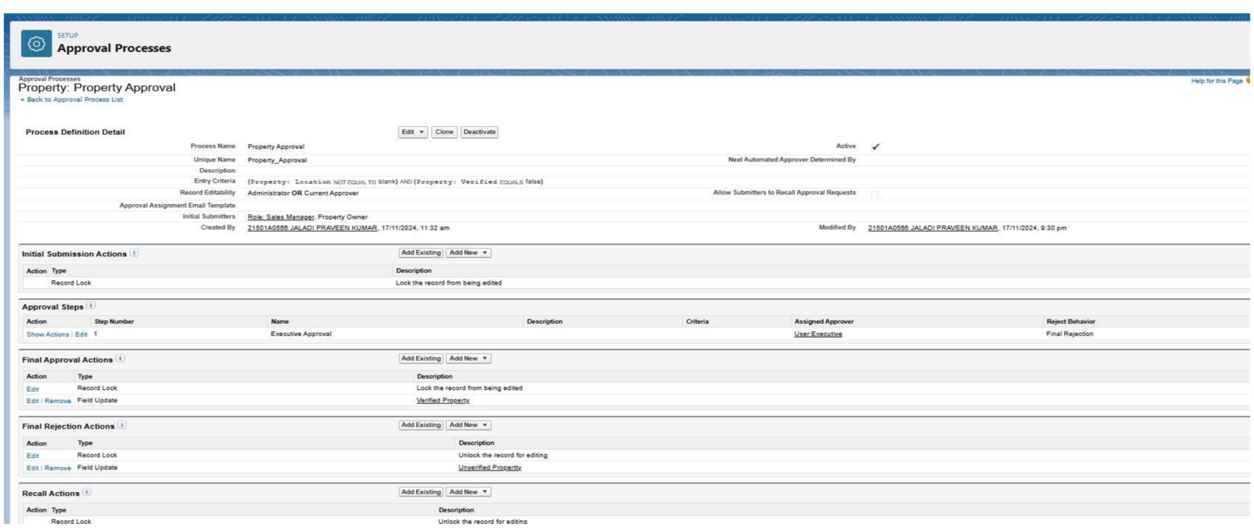
after saving we are directed to approval steps and we need to do as follows Add an approval step name "Executive Approval ”
click next and select the Approver as ”Sales Executive “and "Save" Add One field Update as "Verified Property"

1. Select Object »Property
2. Field to Update >> Verified
3. Field Data Type >» CheckBox
4. Select CheckBox Option as"True"
5. Save.

Add One fieldUpdate as "UnVerified Property"

1. Select Object » Property
2. Field to Update >»Verified
3. Field Data Type >» CheckBox
4. Select CheckBox Option as"False"
5. Save.

Activate the Approval Process.



Create a Recordtrigger flow to submit the Approval Process Automatically

A flow that can submit the recordsdirectly for approval

Activity1

a. FromSetup >>Search forFlows >>Click OnNew andSelect “Record Trigger Flow”.

b. SelectObject >>Property

c. Select“Trigger the flowwhen” >> “A record is created”

d. SetEntry Conditions>> “None”

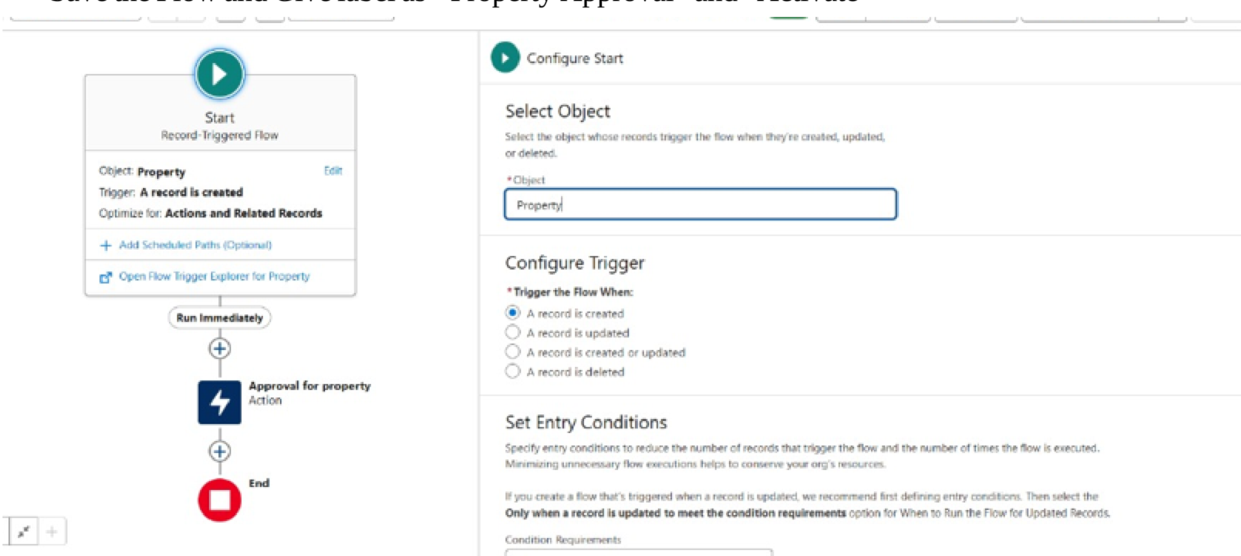
e. Add a“Action” >> “Submit for Approval”

f. Give Label >> Approval forproperty

g. Record Id>> (!SRecord.Id)

h. Done

Save the Flow and Give label as “Property Approval” and “Activate”

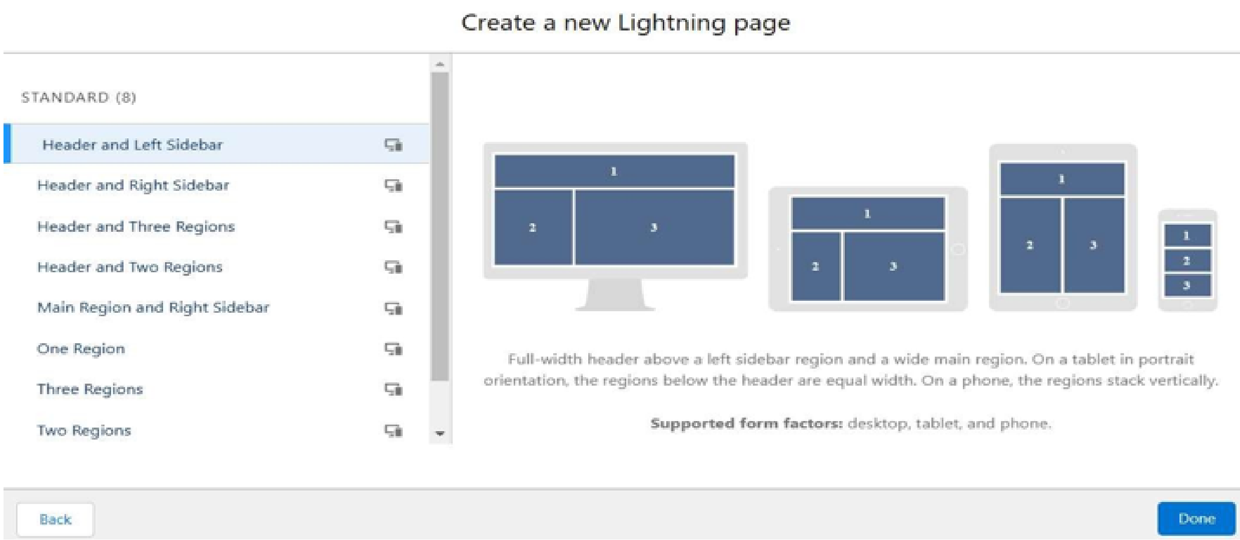


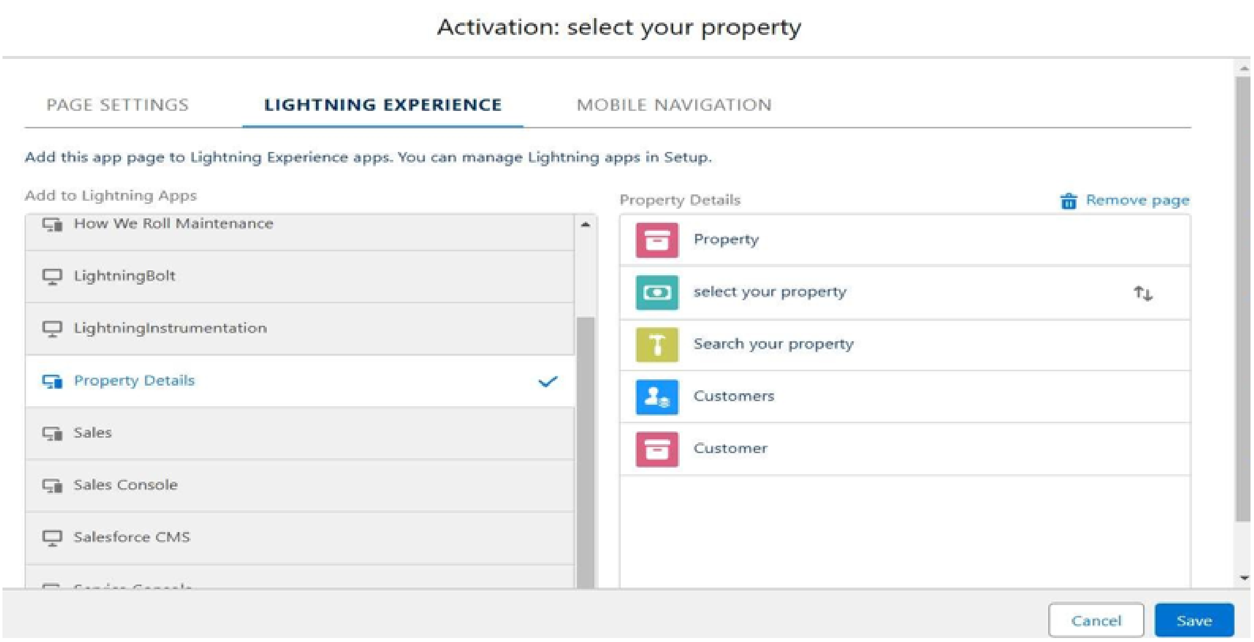
Create an App Page

Create an App Page on the Property details Object named as “Search Your Property”

Activity1

1. From Setup »Go to Lightning App Builder >> Click on New >> Select App Page and
2. Click on Next.
3. Give Labelas“Search your Property” click “Next”.
4. Click “header and Left Sidebar” and Click on “Done”
5. Click on “Save ”and then click on “Activate”.
6. From Page Settingselectpageactivationas “Activate for all Users”.
7. From Lightning ExperienceClick on “Property Details” and click on Add Page“.
8. Then Clickon“Save”





Create a LWC Component

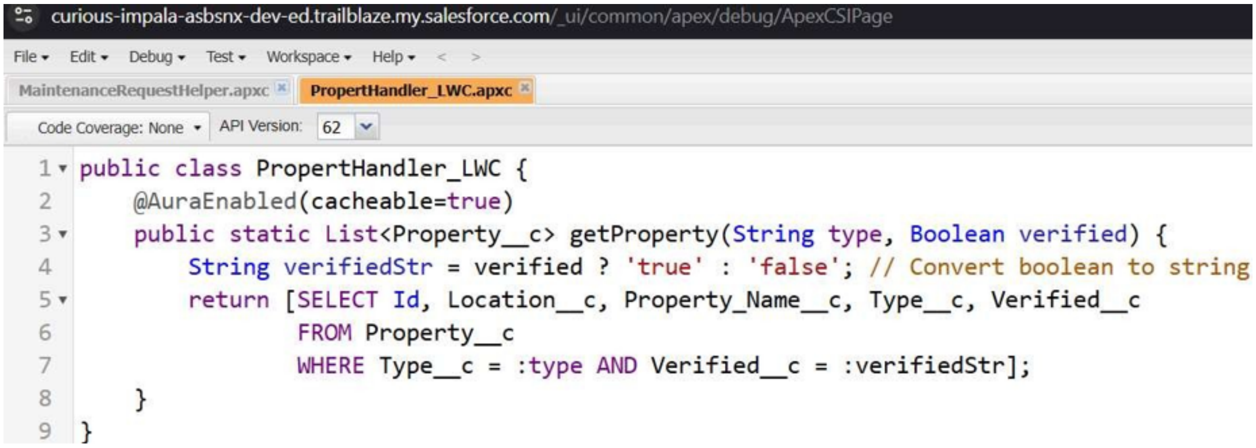
a. Create an LWCComponent for the customers so that only verified customers canaccess the verified properties and non Verified customers can access non verified properties, and deploy it on "Search your Property Page"

Activity1

1. Create an ApexClass and makeit aura enabled and name it "PropertHandIer_LWC"

Code: -

```
public class PropertHandIer_LWC (
@AuraEnabIed(cacheabIe=true)
public static List<Propertyc> getProperty(String type, Boolean verified) ( String
verifiedstr = verified ? 'true' : 'false' // Convert boolean to string return [SELECT Id,
Location c, Property_Namec, Type c, Verified c
FROM Property c
WHERE Type c = :type AND Verified c = :verifiedStr];
```



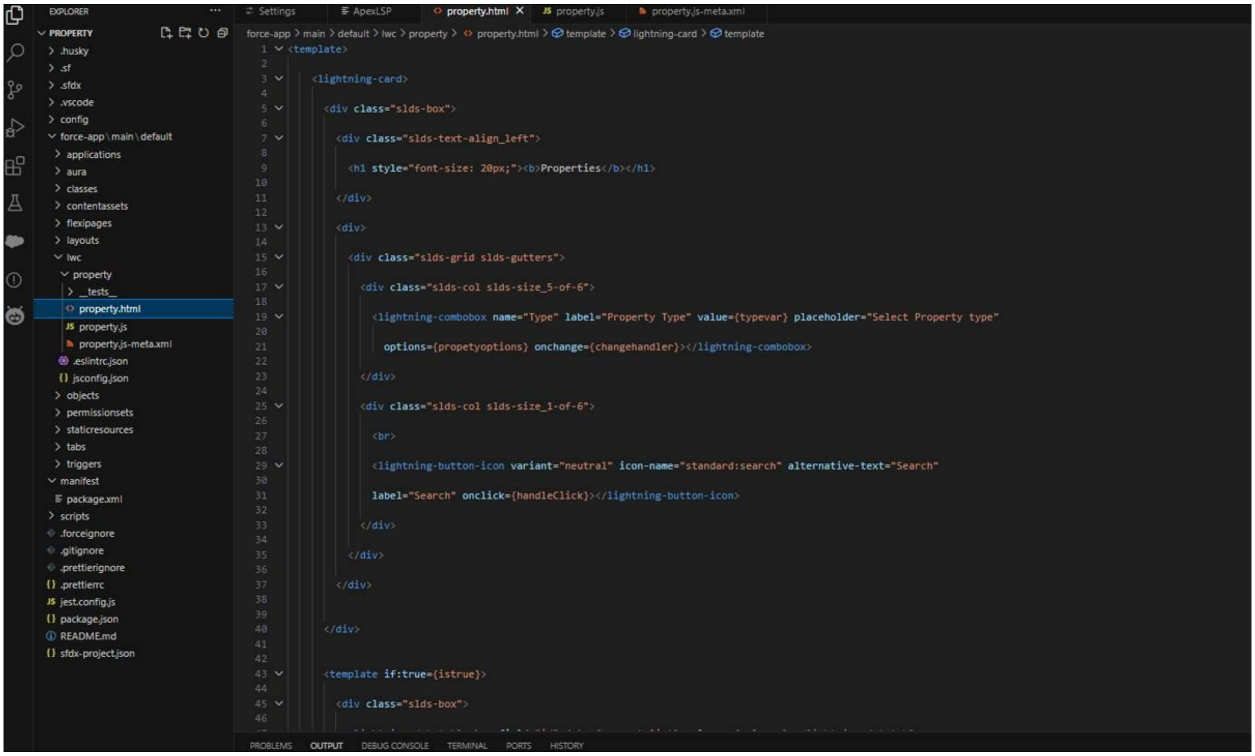

```
public class PropertHandler_LWC {
    @AuraEnabled(cacheable=true)
    public static List<Property__c> getProperty(String type, Boolean verified) {
        String verifiedStr = verified ? 'true' : 'false'; // Convert boolean to string
        return [SELECT Id, Location__c, Property_Name__c, Type__c, Verified__c
                FROM Property__c
                WHERE Type__c = :type AND Verified__c = :verifiedStr];
    }
}
```

1. Create a Lightning Web Component in your VsCode, and (ctrl+shift +P) andclick on authorize an org.
2. Enter your login id and password to authorize your org.
3. Now (ctrl+shift +P) and Create a lightning Web Component and Name it Anything you want to. (Example -
4. In yourHtml File Write this code : -

Code :-

```
<tempIate>
 <Iightning-card>
  <div class="sIds-box">
   <div class="sIds-text-aIign_Ieft">
    <h1 styIe="font-size: 20px;"><b>Properties</b></h1>
   </div>
   <div>
    <div cIass="sIds-grid sIds-gutters">
     <div class=”sIds-col sIds-size_5-of-6“>
                <lightning-combobox name="Type"  IabeI="Property  Type"  value=(typevar}
pIacehoIder="SeIect Property type”
       options=(propetyoptions}  onchange=(changehandIer}></lightning-combobox>
     </div>
     <div class=”sIds-col sIds-size_1-of-6”>
      <br>
```

```
                        <Iightning-button-icon  variant=”neutraI"  icon-name=”standard:search"  alternative-text="Search"
            IabeI="Search"  onclick=(handleCIick}></lightning-button-icon>
          </div>
        </div>
      </div>

    </div>

    <tempIate  if:true=(istrue)>
      <div class="sIds-box">
                  <lightning-datatable  key-fieId="id"  data=(propertyIist}  columns=(coIumns}></Iightning-
datatabIe>
      </div>
    </templates
    <tempIate  if:false=(isfalse}>
      <div class="sIds-box">
       <div styIe="font-size: 15px;"><b>No  properties  Are Found !!</b></div>
      </div>
    </template›
  </lightning-card>
</templates>
```

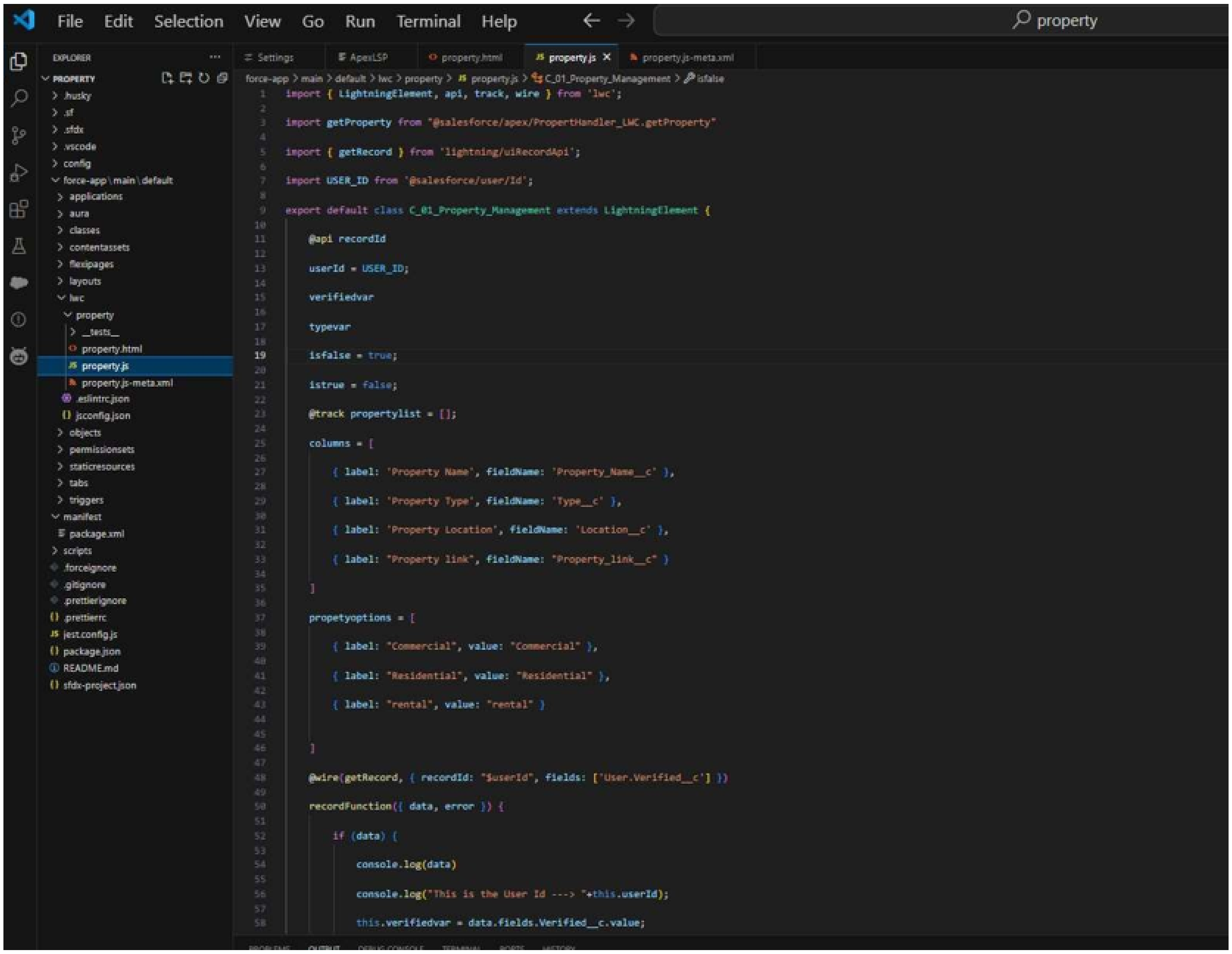


1. In YourJs File Write this code : -

Code :-

```
import ( LightningElement, api, track, wire ) from 'Iwc';

import getProperty from "@salesforce/apex/PropertHandIer_LWC.getProperty"

import ( getRecord } from 'lightning/uiRecordApi'; import USER

ID from '@salesforce/user/Id';

export default class C 01_Property_M anagementextends LightningElement ( @api recordld

  userld = USER_ID;

  verifiedvar typevar

  isfalse = true; istrue =

  false;

  @track propertylist = [];

  columns = [
    ( label: 'Property Name', fieldName: 'Property_Name c' ), ( label:

    'Property Type', fieldName: 'Type c' ),

    ( label: 'Property Location', fieldName: 'Locationc' ), ( label:

    "Property link", fieldName: "Property link c"}

  propetyoptions= [

    ( label: "Commercial", value: "Commercial" }, ( label:

    "Residential", value: "Residential" ),
```

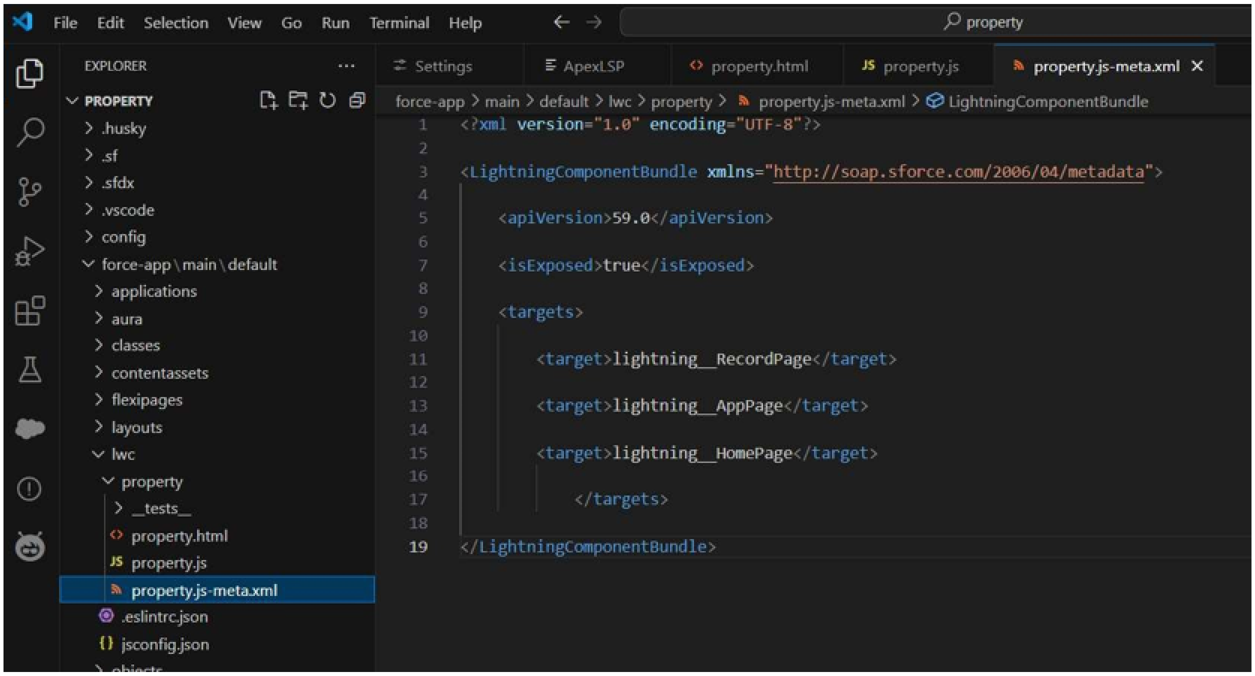
```
( label: "rental", value: "rental" }

@wire(getRecord, ( recordld: "$userld", fields: ['User.Verified  c'] )) recordFunction(( data,
error }) (
  if (data) ( console.log(data)
    console.log("This is the User Id ---> "+this.userId);
    this.verifiedvar = data.fields.Verified   c.value;
  } else (
    console.error(error)
    console.log('this is error')
changehandIer(event) (
  console.log(event.target.value); this.typevar =
  event.target.value;
handleClick() {
  getProperty(( type: this.typevar, verified: this.verifiedvar ))
    .then((result) => ( this.isfaIse =
      true; console.log(result)
      console.log('This is the User id ---> ' + this.userId);
      console.log('This is the verified values ---> ' + this.verifiedvar); if (result !=
      null && result.length != 0) (
        this.istrue = true; this.propertyIist =
        result; console.log(this.verifiedvar);
        console.log(this.typevar)
      ) else (
        this.isfaIse = false;
        this.istrue = false;
    .catch((error) => (
      console.log(error)
```



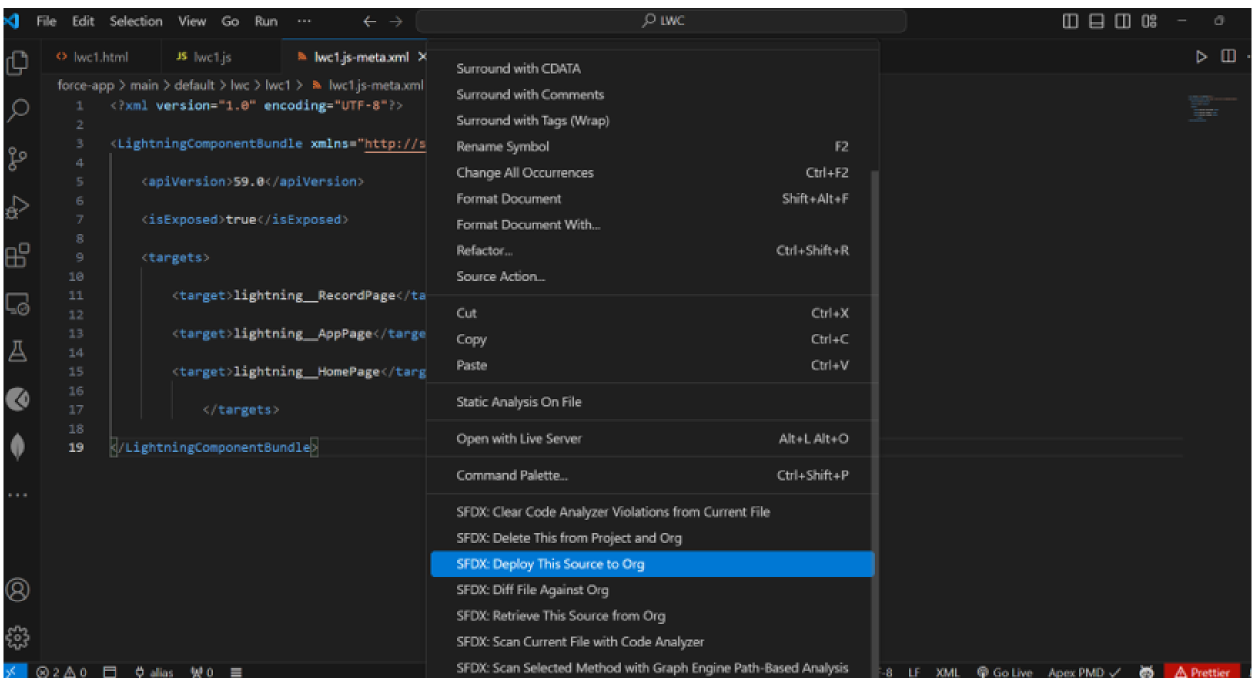
1. In Yourmetafile give your targets to deploy the component.

Code

```
<?xml version="1.0" encoding="UTF-8"?>
<LightningComponentBundle xmlns="http://soap.sforce.com/2006/04/metadata">
    <apiVersion>59.0</apiVersion>
    <isExposed>true</isExposed>
    <targets>
        <target>Iightning__RecordPage</targets
        <target>Iightning__AppPage</targets
        <target>Iightning__HomePage</targets
            </targets>
</LightningComponentBundle>
```



After Saving all the three Codes , Right Click and deploy this componentto the org



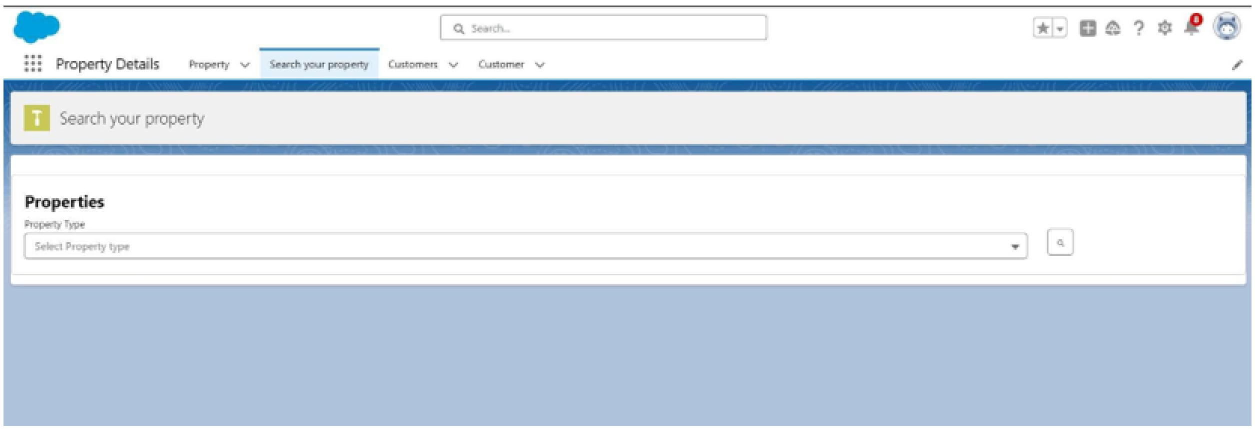
Drag this Component to your App Page

Adding the Componentto your Page

Activity1

1. From Setup >> Go to App Launcher >> Searchfor PropertyDetails
2. On thisPage click ongear icon andclick on EditPage
3. after clicking on edit pageit will bnedirected to apppages then

Drag the Component(properties) to your App Page and Save the Page.

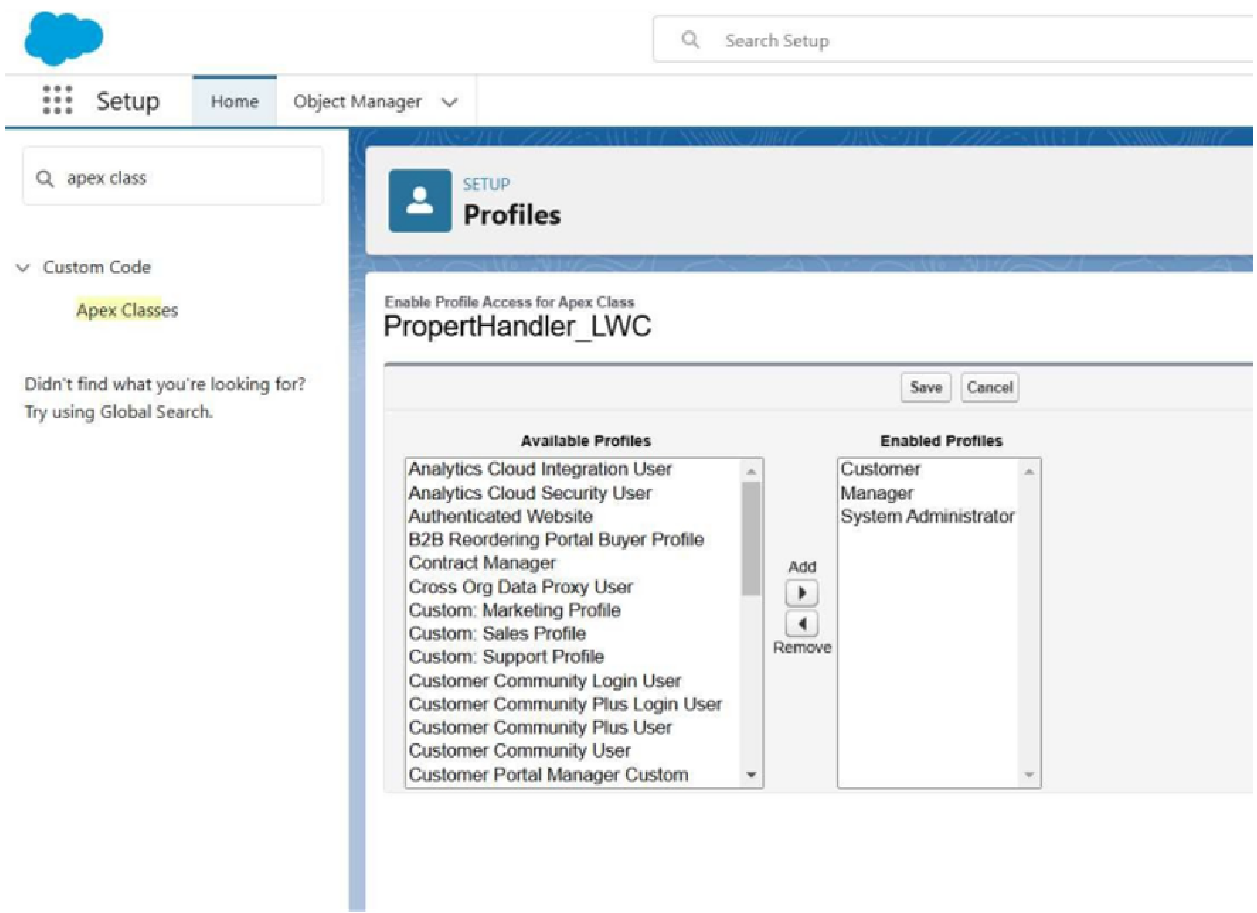


Give Access of Apex Classesto Profiles

The Apex Class has a Security, Enable the security for the profiles that needs to access this class.

1. Activity1

   From Setup>> Search For Apex Classes>> Click on "Security" behind "PropertyHandlerLWC".
2. From Profiles Add "Manager" and"Customer" and "Save".


Search Setup
Setup
Home
Object Manager
apex class
Custom Code
Apex Classes
Didn't find what you're looking for?
Try using Global Search.
SETUP
Profiles
Enable Profile Access for Apex Class
PropertHandler_LWC
Save
Cancel
Available Profiles
Analytics Cloud Integration User
Analytics Cloud Security User
Authenticated Website
B2B Reordering Portal Buyer Profile
Contract Manager
Cross Org Data Proxy User
Custom: Marketing Profile
Custom: Sales Profile
Custom: Support Profile
Customer Community Login User
Customer Community Plus Login User
Customer Community Plus User
Customer Community User
Customer Portal Manager Custom
Add
Remove
Enabled Profiles
Customer
Manager
System Administrator