

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.

1. Client Management

- Add, update, and delete client details.
- Track client preferences, budget, and location interests.
- Maintain contact details and communication history.

2. Property Management

- Manage property listings with details like type, price, location, and features.
- Track properties available for sale, rent, or lease.
- Upload photos and documents for properties.

3. Requirement Matching

- Match client requirements with available properties using filters.
- Notify clients about new properties that fit their criteria.

4. Lead Tracking

- Manage inquiries and follow up with potential clients.
- Schedule meetings and site visits.
- Assign leads to specific team members.

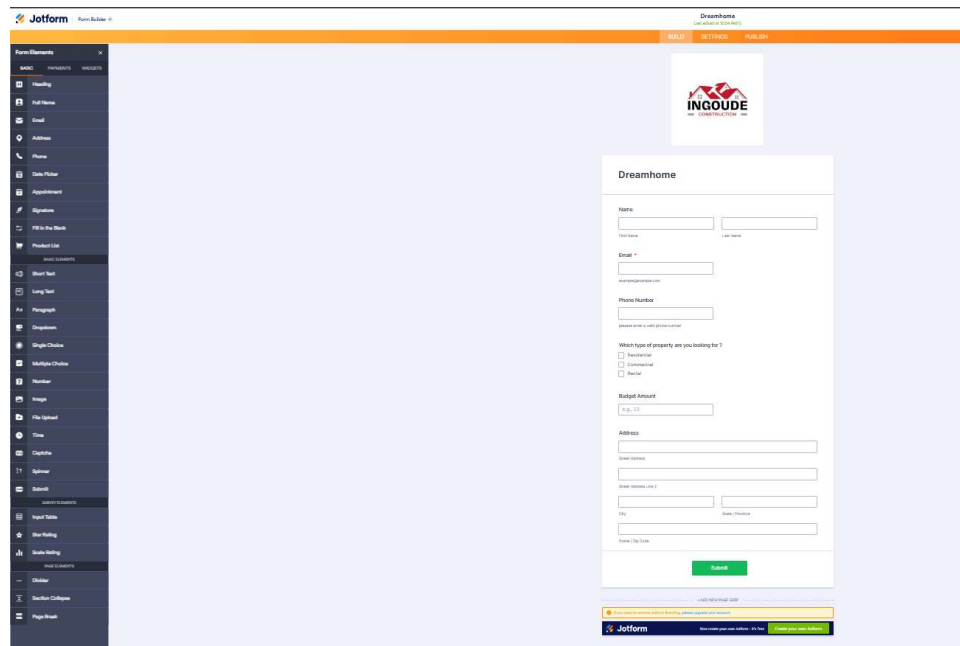
Milestone 1 :- Create a Jotform and integrate it with the org to create a record of customers automatically.

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.

Activity : 1

Open your browser and search for jotform and log in.

2. After login click on create form and click on start from scratch
3. Now create a form to get the customer details like Name, Phone, Email, Address and type of property the customer is interested in.
4. Once the form is created, publish it by clicking on publish.
5. form link :- <https://form.jotform.com/243208544225048>



The screenshot shows a Jotform interface with a dark sidebar on the left containing various form widgets like Text, Email, Address, Phone, Date Picker, etc. The main area displays a form titled 'Dreamhome' with the following fields and options:

- Name:** Two text input fields for 'First Name' and 'Last Name'.
- Email:** A text input field with a dropdown arrow.
- Phone Number:** A text input field with a placeholder '(999) 999-9999'.
- Which type of property are you looking for?** Three radio button options: 'Residential', 'Commercial', and 'Other'.
- Budget Amount:** A text input field with a placeholder '\$ up to \$'.
- Address:** A series of text input fields for 'Street Address', 'Apt, Suite, etc.', 'City', 'State', and 'Zip'.
- Submit:** A green button at the bottom of the form.

At the top right of the form, there is a logo for 'INGOUDE' and a 'Dreamhome' title. The Jotform logo and version number 'Jotform 5.10.0 (2020)' are visible at the bottom of the page.

Create Objects from Spreadsheet

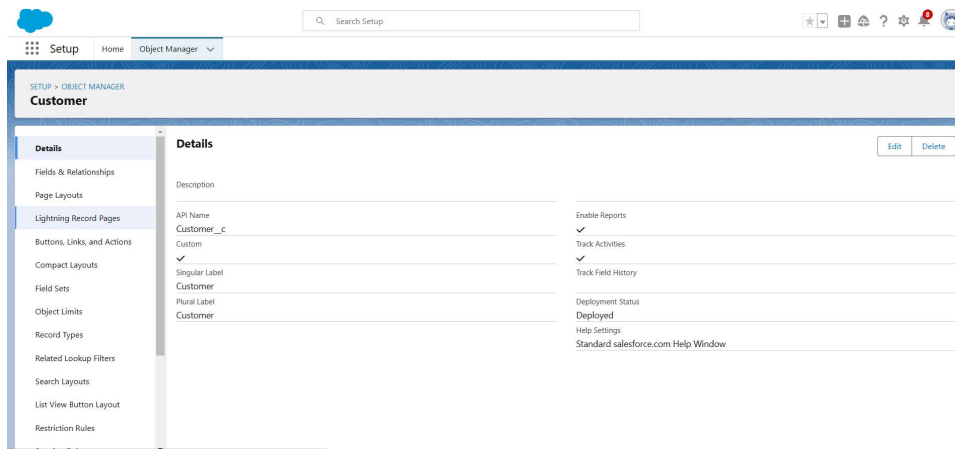
Directly Creating Objects from Spreadsheet in Salesforce

Creating Customer Object :

1. Go to your object manager and click on create 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

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

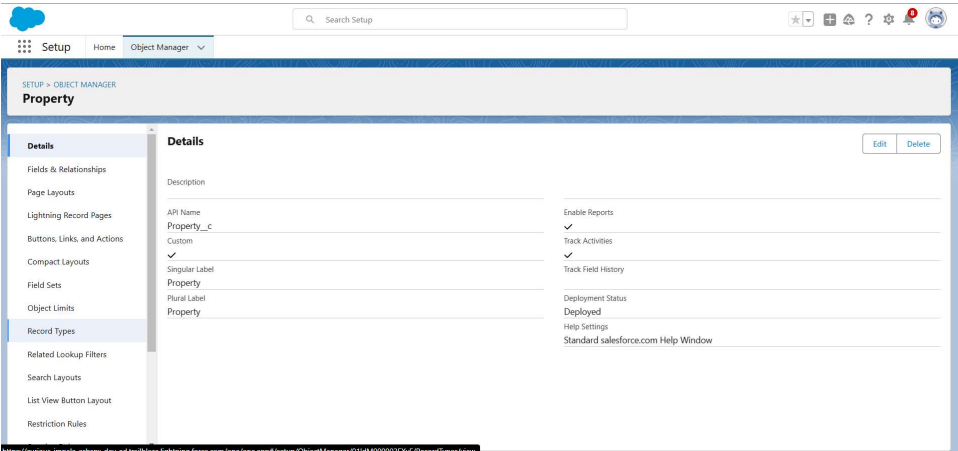


Creating Property Object :

- 1. Follow the same from the customer object to create the Property Object
- 2. [Property](#)

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

- 4. After downloading, upload the file, map the fields and upload to create an object.
- 5. the fileds as follows



Integrate Jotform with Salesforce Platform

In this Milestone we are going to integrate jotform with Salesforce

Activity :-1

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”.

Jotform Form Builder Dreamhome
Last edited at 12:04 AM

INTEGRATIONS Connect your form to other apps

SALESFORCE
Send new leads, contacts, or accounts to your sales CRM

Select a Salesforce Object
Customer

Create a record
Send data from form fields to matched Salesforce fields

Object Fields	Dreamhome
Customer__c	Name - First Name
City	Address - City
Budget Amount	Budget Amount
Property Type	Which type of property are you lookin...
Phone Number	Phone Number
Street Address	Address - Street Address
Email	Email
Customer Name	Name - Last Name
State	Address - State
Street Address line 2	Address - Street Address 2

Then “Save the Integration” and “Finish”.

SALESFORCE
Send new leads, contacts, or accounts to your sales CRM

All Actions See Action Logs + Add New Action

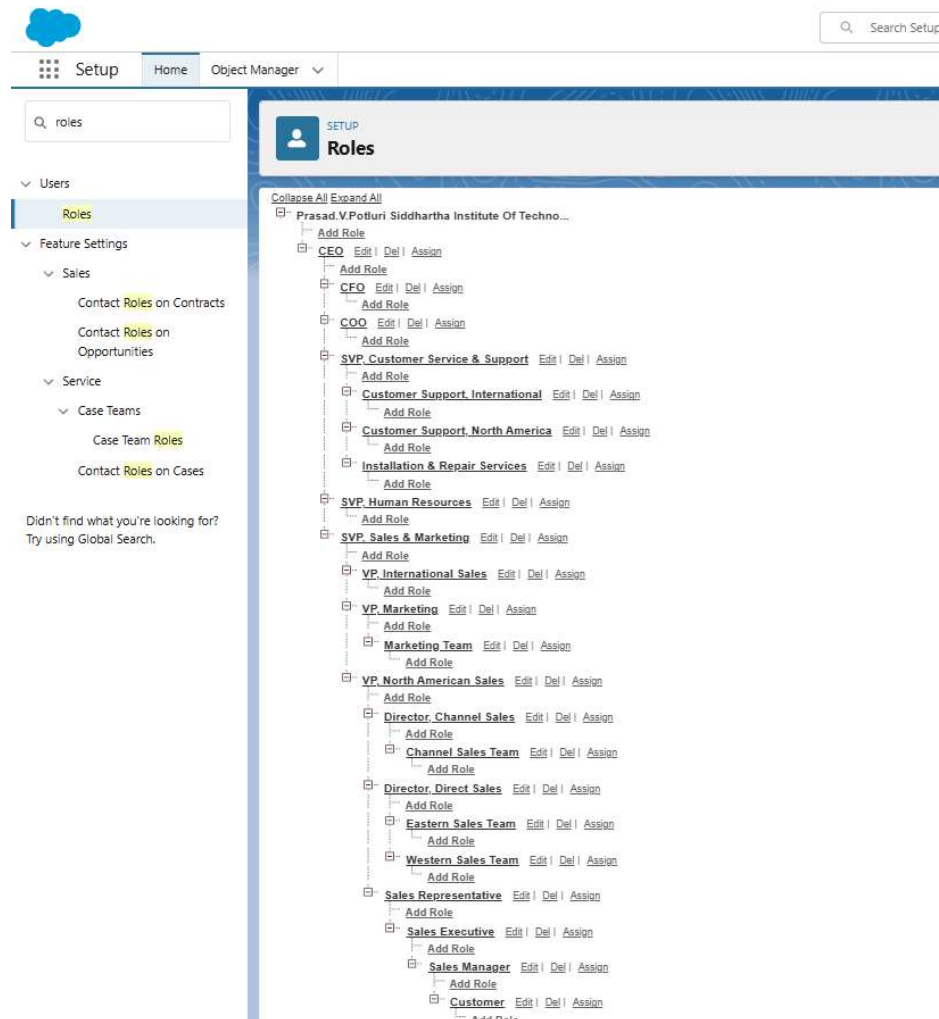
1 Create or update a record
Customer

Create Roles

here we need to Create Roles as per business requirement

Activity :- 1

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



The screenshot displays the Salesforce Setup interface. The left sidebar shows the navigation menu with 'Setup' selected. Under 'Users', the 'Roles' link is highlighted. The main content area shows the 'Roles' page with a search bar at the top. Below the search bar, there is a list of roles organized in a tree structure. The roles listed include: CEO, CFO, COO, SVP, Customer Service & Support, Customer Support, International, Customer Support, North America, Installation & Repair Services, SVP, Human Resources, SVP, Sales & Marketing, VP, International Sales, VP, Marketing, Marketing Team, VP, North American Sales, Director, Channel Sales, Channel Sales Team, Director, Direct Sales, Eastern Sales Team, Western Sales Team, Sales Representative, Sales Executive, Sales Manager, and Customer. Each role has an 'Add Role' link next to it. The 'Sales Representative' role is highlighted, and the 'Add Role' link is visible below it.

1. if we don't find sales representative we need to create it according to the need
2. It will use the "System Administrator Profile".
3. Label - Sales Executive
4. Reports to - Sales Representative

Role Edit

Sales Executive

Role Edit

Label	<input type="text" value="Sales Executive"/>
Role Name	<input type="text" value="Sales_Executive"/> 
This role reports to	<input type="text" value="Sales Representative"/> 
Role Name as displayed on reports	<input type="text"/>

- 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 objects will be displayed

Activity :- 1

1. From Setup>> Go to App Manager and click on New Lightning App and Name it 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.

App Details

*App Name ⓘ

Property Details

*Developer Name ⓘ

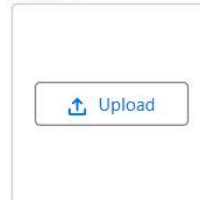
Property_Details

Description ⓘ

Enter a description...

App Branding

Image ⓘ



Primary Color Hex Value ⓘ



#0070D2

Org Theme Options

☐ Use the app's image and color instead of the org's custom theme

App Launcher Preview



Create Profiles

Create profiles as per business requirement

Creating Customer Profiles :-

1. From Setup? Go to Profiles and Clone (standard platform)Salesforce Platform User and Name it “Customer”..
2. Uncheck all the Custom Objects and Check only Property Details From Custom App Settings.
3. Also Remove all the Standard Object Permissions
4. Uncheck all the Custom Object Permissions and check read and view all in “Property”
5. make sure every submission object permissions are unselected and then save

Setup

Profiles

Standard Object Permissions

The permissions defined here control access at the object level. Access to individual records within that object type is controlled by the sharing model. Set access levels based on the functional requirements for the profile. For example, create different groups of permissions for individual contributors, managers, and administrators. [View the Object Permissions](#)

Object Name	Read	Create	Edit	Delete	View All	Modify All
Accounts	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Active Scratch Orgs	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Assets	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Authorization Forms	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Authorization Form Consents	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Authorization Form Data Uses	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Authorization Form Texts	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Background Operations	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Business Brands	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Communication Subscriptions	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Communication Subscription Channel Types	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Communication Subscription Consents	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Communication Subscription Texts	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Contacts	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Contact Point Addresses	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Contact Point Consents	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Contact Point Emails	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Contact Point Phones	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Contact Point Type Consents	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Customers	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
DMB Companies	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Data Use Legal Bases	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Data Use Purposes	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Documents	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Engagement Channel Types	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Identities	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Individuals	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Labels	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Locations	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Nonresponse Registries	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Party Consents	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Push Topics	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Scratch Org Infos	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Sellers	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Streaming Channels	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
User External Credentials	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Custom Object Permissions

Object Name	Read	Create	Edit	Delete	View All	Modify All
Customer	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Property	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Session Settings

Session Times Out After: 2 hours of inactivity

Session Security Level Required at Login: None

Creating Manager Profiles :-

1. From Setup >> Go to Profiles and Clone Salesforce Platform User and Name it "Manager".
2. Uncheck all the Custom Objects and Check only Property Details From Custom App Settings.
3. Also Remove all the Standard Object Permissions.
4. Uncheck all the Custom Object Permissions and check only "modify all" from "Property" and "Customer".

Standard Object Permissions

The permissions defined here control access at the object level. Access to individual records within that object type is controlled by the sharing model. Set access levels based on the functional requirements for the profile. For example, create different groups of permissions for individual contributors, managers, and administrators. [How do I choose?](#)

	Basic Access				Data Administration	
	Read	Create	Edit	Delete	View All	Modify All
Accounts	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Active Scratch Orgs	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Assets	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Authorization Forms	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Authorization Form Consents	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Authorization Form Data Uses	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Authorization Form Tests	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Background Operations	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Business Brands	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Communication Subscriptions	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Communication Subscription Channel Types	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Communication Subscription Consents	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Communication Subscription Timings	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Contacts	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Contact Point Addresses	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Contact Point Consents	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Contact Point Emails	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Contact Point Phones	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Contact Point Type Consents	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Customers	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
D&B Companies	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Data Use Legal Bases	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Data Use Purposes	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Documents	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Engagement Channel Types	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Ideas	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Individuals	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Labels	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Locations	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Homepage Registrations	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Party Consents	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Push Topics	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Scratch Org Infos	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Sellers	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Streaming Channels	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
User External Credentials	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Custom Object Permissions

	Basic Access				Data Administration	
	Read	Create	Edit	Delete	View All	Modify All
Customer	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Property	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Session Settings

Session Times Out After:

Session Security Level Required at Login:

Create a Check Box field on user

Create Field on the User as per the business requirement.

Activity :- 1

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

SETUP > OBJECT MANAGER

User

Details

Fields & Relationships

User Page Layouts

User Profile Page Layouts

Lightning Record Pages

Buttons and Links

Compact Layouts

Field Sets

Object Limits

Related Lookup Filters

Search Layouts

List View Button Layout

Triggers

Flow Triggers

...

User Custom Field

Verified

[Back to User Fields](#)

[Help for](#)

[Validation Rules \(2\)](#)

Custom Field Definition Detail

[Edit](#) [Set Field-Level Security](#) [View Field Accessibility](#) [Where is this used?](#)

Field Information

Field Label	Verified	Object Name	User
Field Name	Verified	Data Type	Checkbox
API Name	Verified_c		
Description			
Help Text			
Data Owner			
Field Usage			
Data Sensitivity Level			
Compliance Categorization			
Created By	Z1501A0566 JALADI PRAVEEN KUMAR, 17/11/2024, 10:58 am		Modified By
			Z1501A0566 JALADI PRAVEEN KUMAR, 17/11/2024, 10:58 am

General Options

Default Value	Unchecked
---------------	-----------

Validation Rules

[New](#) [Validation Rules](#)

No validation rules defined.

[Back To Top](#) Always show me fewer ▲ / ▼ more records per related list

Create Users

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

User : 1

1. Go to Setup --> Administration --> Users --> New User
2. Last Name - Executive
3. Role - Sales Executive
4. License - Salesforce
5. Profile - System Administrator
6. Save

User Edit
Executive

The screenshot shows the 'User Edit' form in Salesforce. At the top, there are buttons for 'Save', 'Save & New', and 'Cancel'. Below this is the 'General Information' section. On the left side, there are input fields for: First Name (empty), Last Name (Executive), Alias (exec), Email (21501a0566@pvpsit.ac.ion), Username (21501a0566@pvpsit.ac.ion), Nickname (User1731751478213757170), Title (empty), Company (empty), Department (empty), and Division (empty). On the right side, there are dropdown menus for: Role (Sales Executive), User License (Salesforce), and Profile (System Administrator). Below these are several checkboxes: Active (checked), Marketing User (unchecked), Offline User (unchecked), Knowledge User (unchecked), Flow User (unchecked), Service Cloud User (unchecked), and Site.com Contributor User (unchecked).

User : 2

1. Go to 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 verified check box is "Unchecked"
7. Save

User Edit
Customer Help for this Page

User Edit Save Save & New Cancel

General Information Required Information

First Name	<input type="text"/>	Role	<input type="text" value="Customer"/>
Last Name	<input type="text" value="Customer"/>	User License	<input type="text" value="Salesforce Platform"/>
Alias	<input type="text" value="cust"/>	Profile	<input type="text" value="Customer"/>
Email	<input type="text" value="21501a0566@pvpsit.ac.in"/>	Active	<input checked="" type="checkbox"/>
Username	<input type="text" value="566@sit.com"/>	Marketing User	<input type="checkbox"/>
Nickname	<input type="text" value="User173175161322663227"/>	Offline User	<input type="checkbox"/>
Title	<input type="text"/>	Knowledge User	<input type="checkbox"/>
Company	<input type="text"/>	Flow User	<input type="checkbox"/>
Department	<input type="text"/>	Service Cloud User	<input type="checkbox"/>
Division	<input type="text"/>	Site.com Contributor User	<input type="checkbox"/>
		Site.com Publisher User	<input type="checkbox"/>

User : 4

1. Go to 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 Approval Process for Property Object

An Approval process to approve or reject the records as according

Activity :- 1

1. From Setup >> Process Automation >> Approval Process
2. before proceeding we need to select property in the 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. From Record Editability Properties >> Click on AdministratorsORthe currently assigned approver can edit records during the approval process.
9. From Step 5. Select Fields to Display on Approval Page Layout select Property, Owner, Location, Type.

Approval Process Edit
Property Approval

Step 5. Select Fields to Display on Approval Page Layout

The approval page is where an approver will actually approve or reject a request. Using the options below, choose the fields to display on this page.

Available Fields		Selected Fields
Created By	Add Remove	Property
Last Modified By		Owner
Verified		Location
		Property Name
		Type

Up
Down



[Click here to view an example](#)

1. Click Next and Select the initial Submitters >>
2. Owner >> Property Owner
3. Roles >> Sales Manager
4. 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 field Update 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.

The screenshot shows the 'Approval Processes' setup page for 'Property Approval'. The page is divided into several sections:

- Process Definition Detail:** Includes fields for Process Name, Unique Name, Description, Entry Criteria, Record Editability, Approval Assignment Email Template, Initial Submitters, and Created By. The 'Active' checkbox is checked.
- Initial Submission Actions:** A table with columns 'Action' and 'Description'. It contains one action: 'Record Lock' with description 'Lock the record from being edited'.
- Approval Steps:** A table with columns 'Action', 'Step Number', 'Name', 'Description', 'Criteria', 'Assigned Approver', and 'Reject Behavior'. It contains one step: 'Executive Approval' with assigned approver 'User:Edoardo' and reject behavior 'Final Rejection'.
- Final Approval Actions:** A table with columns 'Action', 'Type', and 'Description'. It contains two actions: 'Record Lock' (description: 'Lock the record from being edited') and 'Field Update' (description: 'Verified Property').
- Final Rejection Actions:** A table with columns 'Action', 'Type', and 'Description'. It contains two actions: 'Record Lock' (description: 'Unlock the record for editing') and 'Field Update' (description: 'Unverified Property').
- Recall Actions:** A table with columns 'Action', 'Type', and 'Description'. It contains one action: 'Record Lock' with description 'Unlock the record for editing'.

At the bottom, there is a 'Back To Top' link and a note: 'Always show the fewer ▲ / ▼ more records per related list'.

Create a Record trigger flow to submit the Approval Process Automatically

A flow that can submit the records directly for approval

Activity :- 1

1. From Setup >> Search for Flows >> Click On New and Select "Record Trigger Flow".
2. Select Object >> Property
3. Select "Trigger the flow when" >> "A record is created"
4. Set Entry Conditions >> "None"
5. Add a "Action" >> "Submit for Approval"
6. Give Label >> Approval for property
7. Record Id >> {!\$Record.Id}
8. Done

Save the Flow and Give label as "Property Approval" and "Activate"

Last saved on 17/11/2024, 11:49 am **Active** [Run](#) [Debug](#) [View Tests](#) [Save As New Version](#) [Save](#) [Deactivate](#)

The screenshot displays the Salesforce Flow Builder interface for a 'Record Trigger Flow'. The flow is titled 'Submit for Approval' and is currently 'Active'. The flow diagram on the left shows a sequence of steps: 'Run Immediately' (green circle), followed by a connector (+), then the 'Submit for Approval' action (blue square with a lightning bolt), another connector (+), and finally an 'End' step (red square). The right-hand pane shows the configuration for the 'Submit for Approval' action. It includes fields for 'Label' (set to 'Approval for property'), 'API Name' (set to 'Approval_for_property'), and a 'Description' field. Below these fields, there is a section titled 'Set Input Values for the Selected Action' which shows a variable 'Record ID' (labeled A_a) being mapped to the 'Record ID' input of the 'Submit for Approval' core action.

Submit for Approval

* Label
Approval for property

* API Name
Approval_for_property

Description

Submit for Approval
submit-submit

Use values from earlier in the flow to set the inputs for the "Submit for Approval" core action. To use its outputs later in the flow, store them in variables.

Set Input Values for the Selected Action

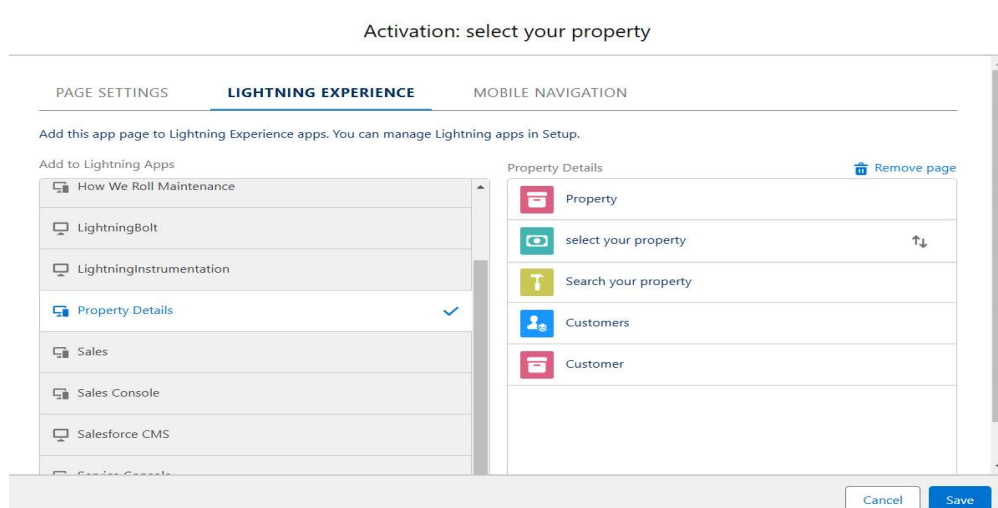
A_a * Record ID
A_a Triggering Property__c > Record ID

Create an App Page

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

Activity :- 1

1. From Setup >> Go to Lightning App Builder >> Click on New >> Select App Page and
2. Click on Next.
3. Give Label as "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 Setting select page activation as "Activate for all Users".
7. From Lightning Experience Click on "Property Details" and click on Add Page".
8. Then Click on "Save"



Create a LWC Component

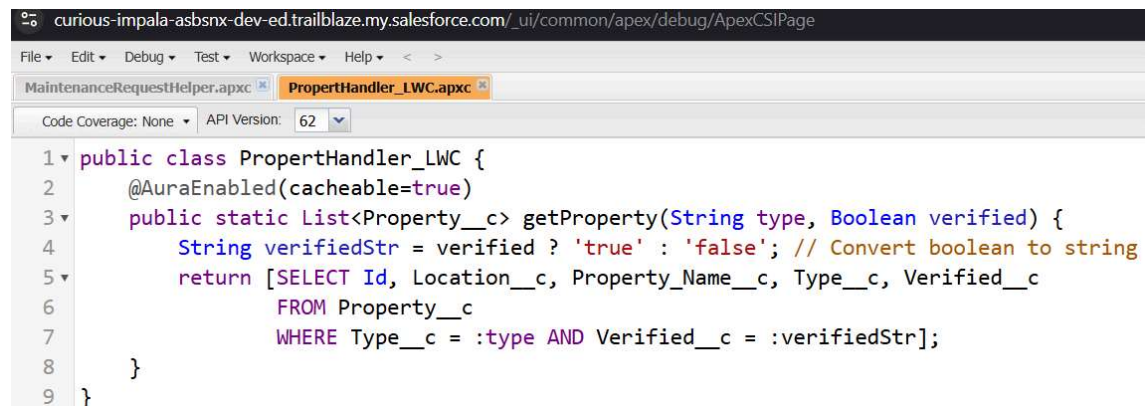
- Create an LWC Component for the customers so that only verified customers can access the verified properties and non Verified customers can access non verified properties, and deploy it on "Search your Property Page"

Activity :- 1

1. Create an Apex Class and make it aura enabled and name it "PropertHandler_LWC"

Code: -

```
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];  
    }  
}
```



2. Create a Lightning Web Component in your VsCode, and (ctrl+shift +P) and click on authorize an org.
3. Enter your login id and password to authorize your org.
4. Now (ctrl+shift +P) and Create a lightning Web Component and Name it Anything you want to. (Example -)

5. In your Html File Write this code :-

Code :-

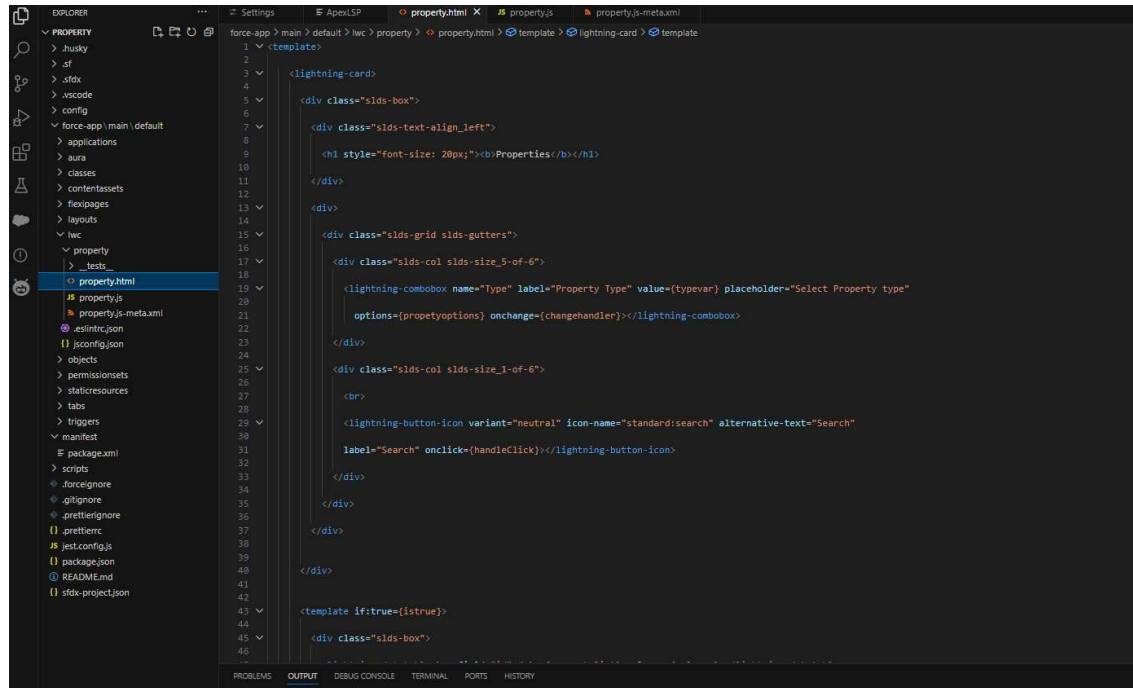
```
<template>
<lightning-card>
  <div class="slds-box">
    <div class="slds-text-align_left">
      <h1 style="font-size: 20px;"><b>Properties</b></h1>
    </div>
    <div>
      <div class="slds-grid slds-gutters">
        <div class="slds-col slds-size_5-of-6">
          <lightning-combobox name="Type" 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>
  </div>

  <template if:true={istru}>
    <div class="slds-box">
      <lightning-datatable key-field="id" data={propertylist} columns={columns}></lightning-
datatable>
    </div>
  </template>
  <template if:false={isfalse}>
    <div class="slds-box">
      <div style="font-size: 15px;"><b>No properties Are Found !!</b></div>
    </div>
  </template>
</lightning-card>
</template>
```

</template>

</lightning-card>

</template>



6. In Your Js File Write this code :-

Code :-

```
import { LightningElement, api, track, wire } from 'lwc';
import getProperty from '@salesforce/apex/PropertyHandler_LWC.getProperty';
import { getRecord } from 'lightning/uiRecordApi';
import USER_ID from '@salesforce/user/Id';

export default class C_01_Property_Management extends LightningElement {
  @api recordId
  userId = 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: 'Location__c' },
    { label: "Property link", fieldName: "Property_link__c" }
  ]
  propetyoptions = [
    { label: "Commercial", value: "Commercial" },
    { label: "Residential", value: "Residential" },
    { label: "rental", value: "rental" }

  ]
  @wire(getRecord, { recordId: "$userId", 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')
    }
  }
}

changehandler(event) {
  console.log(event.target.value);
  this.typevar = event.target.value;
}

handleClick() {

  getProperty({ type: this.typevar, verified: this.verifiedvar })
    .then((result) => {
      this.isfalse = 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.propertylist = result;
        console.log(this.verifiedvar);
      }
    })
}

```

```

        console.log(this.typevar)
    } else {
        this.isfalse = false;
        this.istrue = false;
    }

})

.catch((error) => {
    console.log(error)
})

}
}

```

```

force-app > main > default > lwc > property > property.js
1  import { LightningElement, api, track, wire } from 'lwc';
2
3  import getRecord from "@salesforce/apex/PropertyHandler_LWC.getRecord";
4
5  import { getRecord } from 'lightning/uiRecordApi';
6
7  import USER_ID from '@salesforce/user/id';
8
9  export default class C_01_Property_Management extends LightningElement {
10
11      @api recordId;
12
13      userId = USER_ID;
14
15      verifiedvar;
16
17      typevar;
18
19      isfalse = true;
20
21      istrue = false;
22
23      @track propertylist = [];
24
25      columns = [
26
27          { label: 'Property Name', fieldName: 'Property_Name__c' },
28
29          { label: 'Property Type', fieldName: 'Type__c' },
30
31          { label: 'Property Location', fieldName: 'Location__c' },
32
33          { label: 'Property link', fieldName: 'Property_link__c' }
34      ]
35
36      propertyoptions = [
37
38          { label: 'Commercial', value: 'Commercial' },
39
40          { label: 'Residential', value: 'Residential' },
41
42          { label: 'rental', value: 'rental' }
43      ]
44
45      @wire(getRecord, { recordId: "$userId", fields: ["User.Verified__c"] })
46      recordFunction({ data, error }) {
47
48          if (data) {
49
50              console.log(data)
51
52              console.log("This is the User Id ----> " + this.userId);
53
54              this.verifiedvar = data.fields.Verified__c.value;
55
56          }
57
58      }
59
60  }

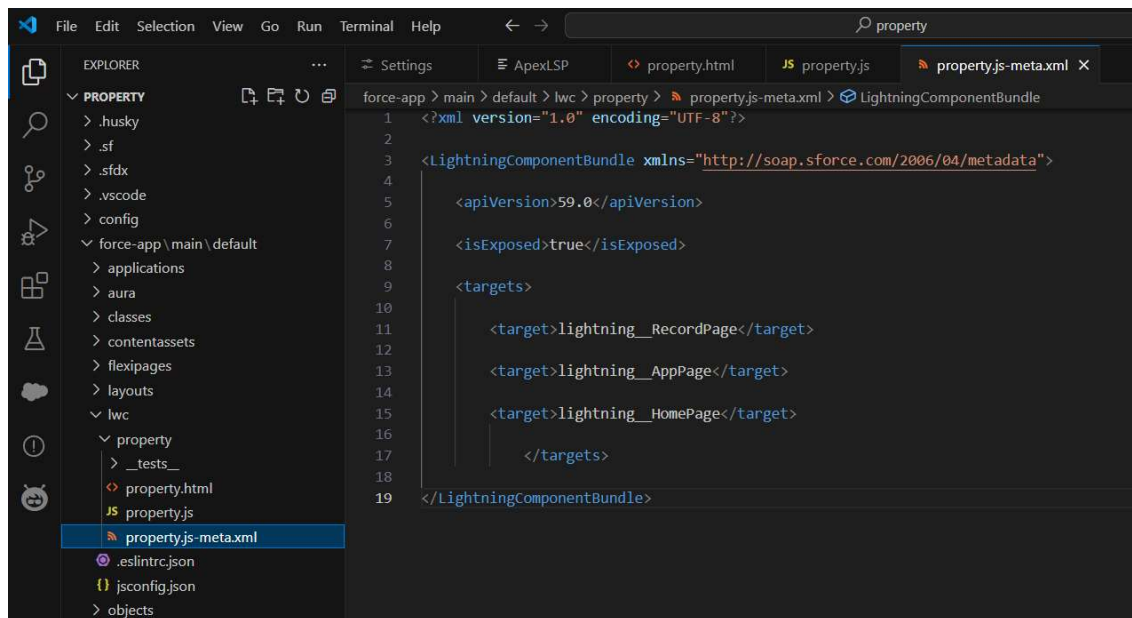
```

7. In Your metafile give your targets to deploy the component.

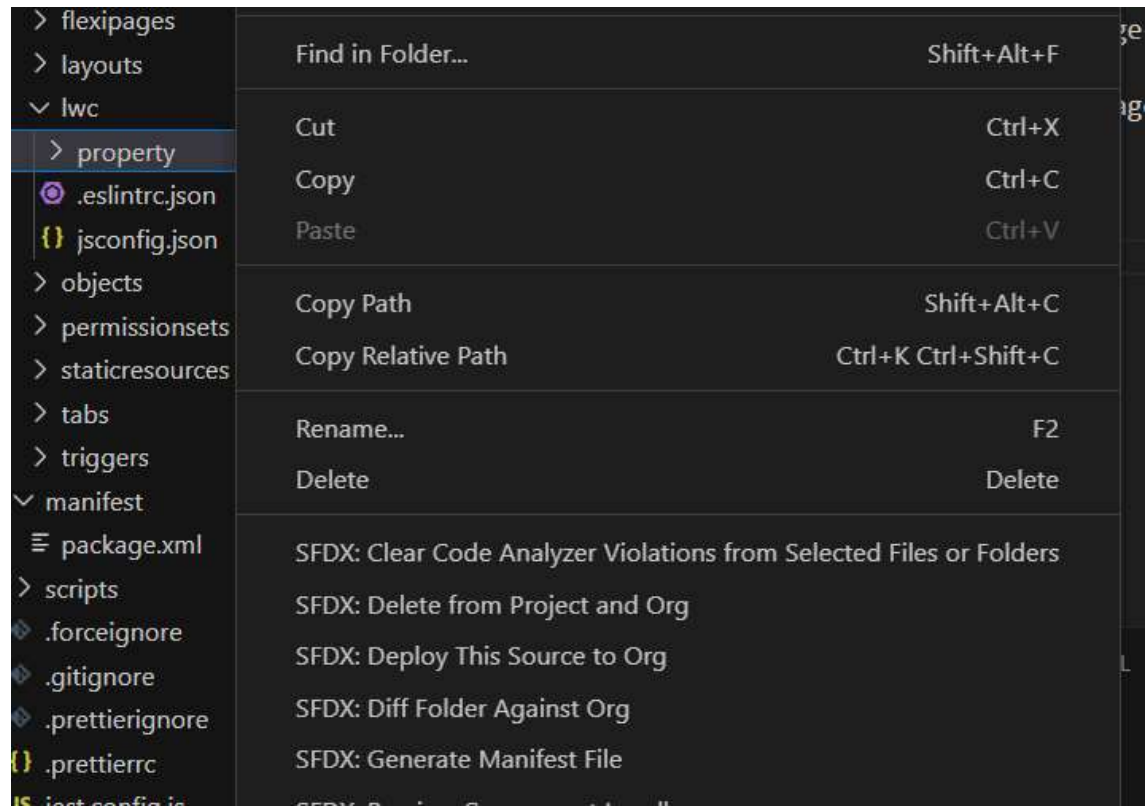
Code

:-

```
<?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>lightning__RecordPage</target>
    <target>lightning__AppPage</target>
    <target>lightning__HomePage</target>
  </targets>
</LightningComponentBundle>
```



After Saving all the three Codes , Right Click and deploy this component to the org.

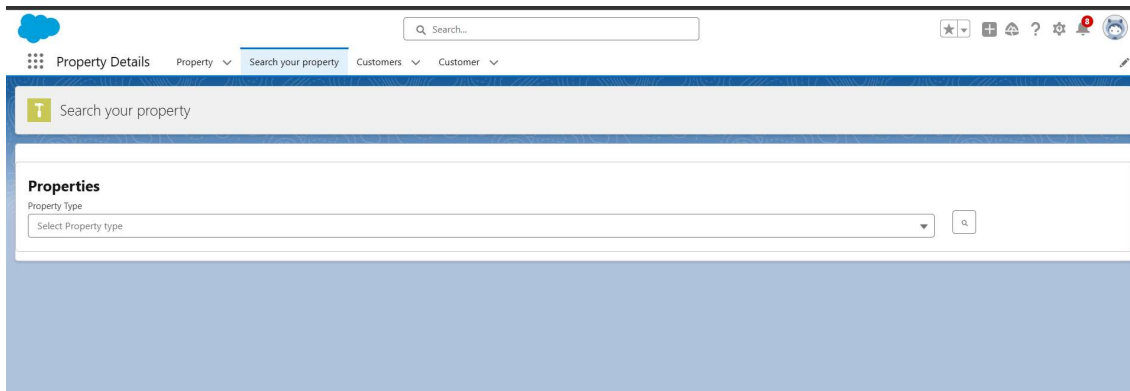


Drag this Component to your App Page

Adding the Component to your Page

Activity :- 1

1. From Setup >> Go to App Launcher >> Search for Property Details
2. On this Page click on gear icon and click on Edit Page
3. after clicking on edit page it will be directed to app pages then
3. Drag the Component(properties) to your App Page and Save the Page.



Give Access of Apex Classes to Profiles

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

1. Activity :- 1
From Setup >> Search For Apex Classes >> Click on "Security" behind "PropertyHandler_LWC".
2. From Profiles Add "Manager" and "Customer" and "Save".

The screenshot shows the Salesforce Setup interface. At the top, there's a search bar labeled 'Search Setup'. Below it, the navigation bar includes 'Setup', 'Home', and 'Object Manager'. A left sidebar contains a search bar with 'apex class' and a list of categories: 'Custom Code' and 'Apex Classes'. The main content area is titled 'SETUP Profiles' and 'Enable Profile Access for Apex Class PropertyHandler_LWC'. It features two columns: 'Available Profiles' and 'Enabled Profiles'. The 'Available Profiles' list includes various system and custom profiles. The 'Enabled Profiles' list currently contains 'Customer', 'Manager', and 'System Administrator'. Between the lists are 'Add' and 'Remove' buttons. 'Save' and 'Cancel' buttons are at the top right of the configuration area.

Available Profiles	Enabled Profiles
Analytics Cloud Integration User	Customer
Analytics Cloud Security User	Manager
Authenticated Website	System Administrator
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	