

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

BY
Dhumnor Srinath
chandrakanthsrinath@gmail.com

Project Abstract

Dreams World Properties has strategically integrated Salesforce to enhance client management and streamline property-related services. This integration automates the capture of customer information and preferences from the company's website, creating comprehensive profiles in Salesforce and reducing manual data entry errors. Customers are categorized into approved and non-approved segments, with approved users receiving personalized property recommendations tailored to their preferences, significantly enhancing their experience. Non-approved users still access a wide range of listings, ensuring suitability for all clients. This seamless integration optimizes operational efficiency by automating processes and freeing staff for strategic tasks, ultimately improving customer satisfaction and supporting growth in the competitive real estate market. Through Salesforce, Dreams World Properties offers superior customer service, personalized recommendations, and efficient management, establishing itself as a leader in the industry.

INDEX

SL NO	TASK/TOPIC	page number
1	TASK 1	5
2	TASK 2	6
3	TASK 3	7-8
4	TASK 4	8-9
5	TASK 5	10-11
6	TASK 6	12-14
7	TASK 7	14-15
8	TASK 8	15-18
9	TASK 9	18-21
10	TASK 10	21-22
11	TASK 11	22-23
12	TASK 12	23-26
13	TASK 13	26-27
14	TASK 14	27-28

INTRODUCTION

Dreams World Properties has harnessed the power of Salesforce to revolutionize its client management and property-related services. By integrating Salesforce with its website, the company has automated the capture and organization of customer information and preferences, creating detailed profiles without the need for manual data entry. This sophisticated system categorizes users into approved and non-approved segments, allowing for personalized property recommendations that enhance the user experience for approved clients while still providing a broad range of listings for others. This integration not only optimizes operational efficiency but also frees up staff to focus on strategic tasks, ultimately improving customer satisfaction and driving growth in the competitive real estate market.

In an era where personalized customer service and efficient data management are crucial, Dreams World Properties' innovative approach sets it apart from competitors. The automation facilitated by Salesforce ensures that no customer detail is overlooked, and every interaction is recorded and utilized to enhance service delivery. The categorization system means that approved users receive tailored property suggestions, making their search more efficient and enjoyable, while non-approved users still have access to comprehensive listings, ensuring no potential client is left out. This seamless integration showcases Dreams World Properties' commitment to leveraging advanced technology to provide top-tier real estate services, demonstrating their leadership and forward-thinking in an ever-evolving industry. Through this innovative use of Salesforce, Dreams World Properties underscores its dedication to superior customer service, operational excellence, and effective management, solidifying its position as a frontrunner in the real estate market.

Task 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.

Activity 1

- Creating a form to get the customer details like Name, Phone, Email, Address and type of property the customer is interested in.

My form - <https://form.jotform.com/242103954717052>

Output:

The form is titled "Dreams World". It includes fields for Name (First Name and Last Name), Phone Number (with validation message "Please enter a valid phone number."), Email (example@example.com), and Address (Street Address, Street Address Line 2, City, State / Province, Postal / Zip Code). A section for "Which type of Property are you looking for?" includes radio buttons for RESIDENTIAL, COMMERCIAL, and RENTAL. A green "Submit" button is at the bottom.

Dreams World

Name

First Name

Last Name

Phone Number

(000) 000-0000

Please enter a valid phone number.

Email

example@example.com

Address

Street Address

Street Address Line 2

City State / Province

Postal / Zip Code

Which type of Property are you looking for?

RESIDENTIAL

COMMERCIAL

RENTAL

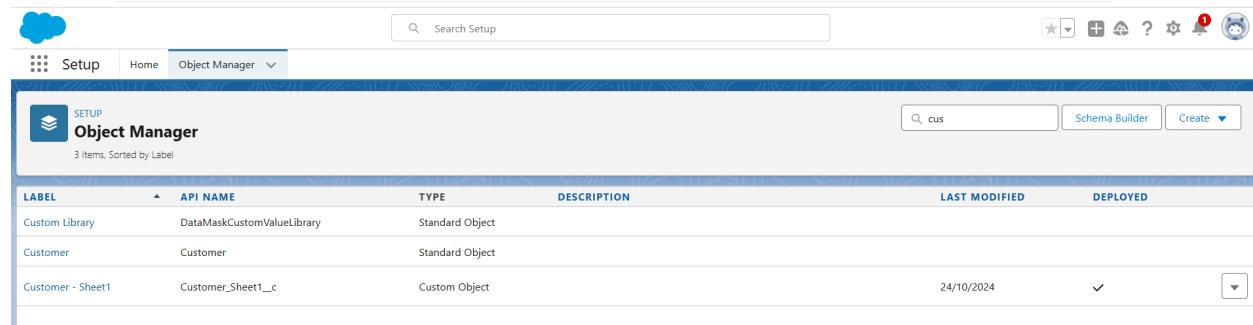
Submit

Task 2: Create Objects from Spreadsheet.

Directly Creating Objects from Spreadsheet in Salesforce

- Create Customer object

To create a Customer object, go to Object Manager, select "Create Object from Spreadsheet," click the provided link to download the Customer spreadsheet, upload the downloaded file, map the spreadsheet fields to the object's fields, and complete the upload to finish creating the Customer object.



The screenshot shows the Salesforce Object Manager interface. The top navigation bar includes 'Setup', 'Home', and 'Object Manager'. A search bar at the top right contains the text 'cus'. Below the header, there is a search bar with 'cus' and buttons for 'Schema Builder' and 'Create'. The main area displays a table titled 'Object Manager' with three items. The columns are labeled 'LABEL', 'API NAME', 'TYPE', 'DESCRIPTION', 'LAST MODIFIED', and 'DEPLOYED'. The data rows are:

Label	API Name	Type	Description	Last Modified	Deployed
Custom Library	DataMaskCustomValueLibrary	Standard Object			
Customer	Customer	Standard Object			
Customer - Sheet1	Customer_Sheet1__c	Custom Object		24/10/2024	✓

- Create Property object

To create a Property object, go to Object Manager, select "Create Object from Spreadsheet," download the Property spreadsheet via the provided link, upload the file, map the fields from the spreadsheet to the object's fields, and finalize the upload to complete the creation of the Property object.



The screenshot shows the Salesforce Object Manager interface. The top navigation bar includes 'Setup', 'Home', and 'Object Manager'. A search bar at the top right contains the text 'prop'. Below the header, there is a search bar with 'prop' and buttons for 'Schema Builder' and 'Create'. The main area displays a table titled 'Object Manager' with one item. The columns are labeled 'LABEL', 'API NAME', 'TYPE', 'DESCRIPTION', 'LAST MODIFIED', and 'DEPLOYED'. The data row is:

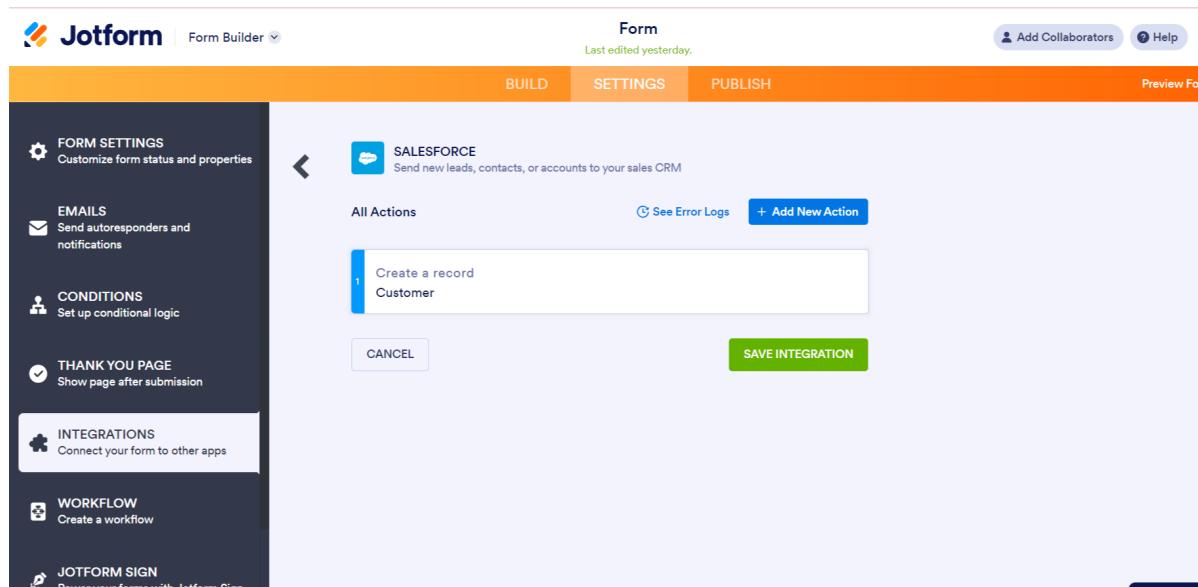
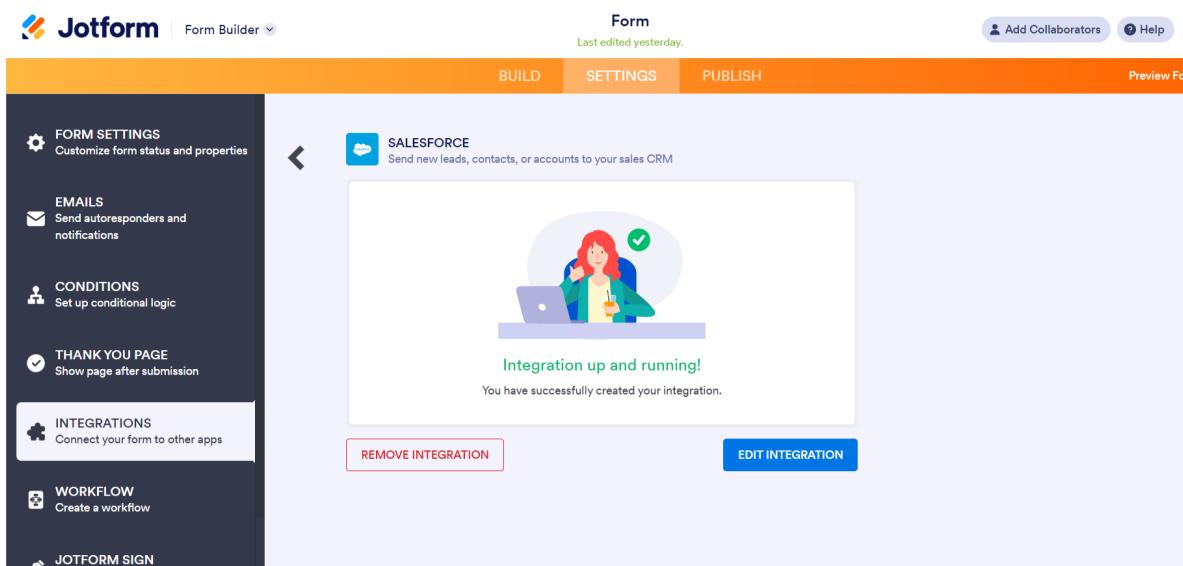
Label	API Name	Type	Description	Last Modified	Deployed
Property	Property__c	Custom Object		24/10/2024	✓

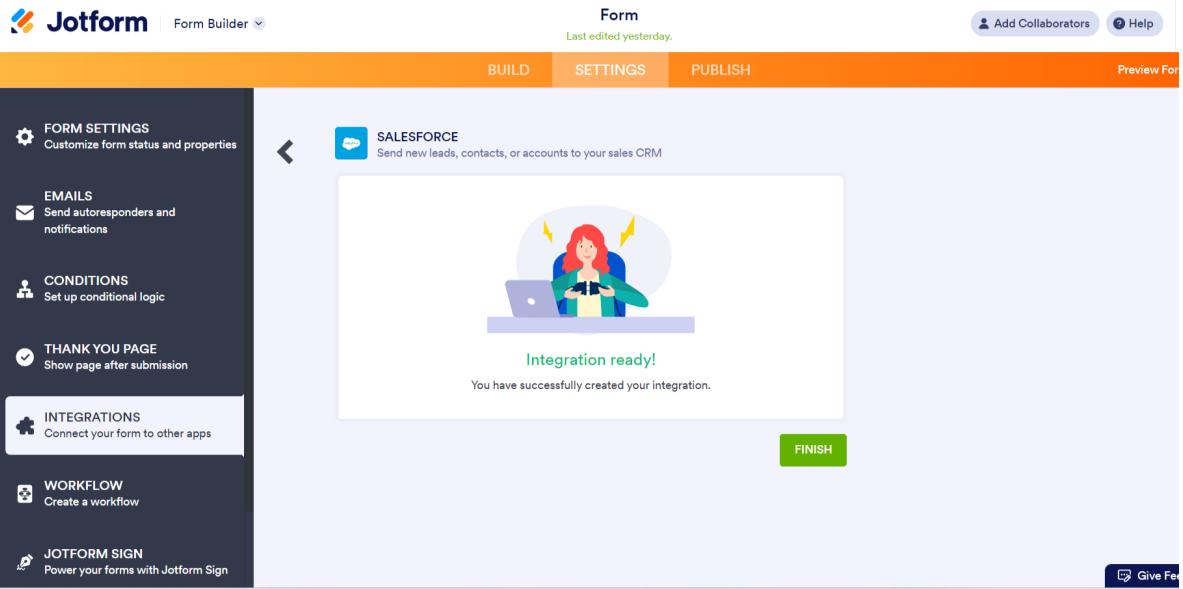
Task 3: Integrate JotForm with Salesforce Platform

In this Milestone we are going to integrate jotform with Salesforce

Activity 1

- Integrate Jotform with Salesforce to create Customer records.





Task 4: Create Roles

Create Roles as per business requirement

- Sales Executive Role

- Create a hierarchical role structure in Salesforce: Sales Executive below Sales Representative, Sales Manager below Sales Executive, and Customer below Sales Manager.

The screenshot shows the Salesforce Setup interface. At the top, there's a search bar with 'Search Setup' and a 'Setup' button. Below the search bar is a navigation bar with 'Home' and 'Object Manager'. The main content area has a sidebar on the left with 'Users' expanded, showing 'Roles' selected. The main pane displays a tree view of roles under 'Sales'. The hierarchy is as follows:

- Sales Representative** (Edit | Del | Assign)
 - Sales Executive** (Edit | Del | Assign)
 - Sales Manager** (Edit | Del | Assign)
 - Customer** (Edit | Del | Assign)
- SF Admin** (Edit | Del | Assign)
 - Add Role**
- SVP, Customer Service & Support** (Edit | Del | Assign)
 - Add Role**
 - Customer Support, International** (Edit | Del | Assign)
 - Add Role**
 - Customer Support, North America** (Edit | Del | Assign)
 - Add Role**
 - Installation & Repair Services** (Edit | Del | Assign)
 - Add Role**

The screenshot shows the Salesforce Setup interface with the 'Roles' page selected. The left sidebar shows navigation categories like Users, Feature Settings, Sales, Service, and Case Teams. The main content area displays the 'Sales Executive' role details. The 'Role Detail' section includes fields for Label (Sales Executive), This role reports to (Sales Representative), Modified By (Dhummor Srinath, 24/10/2024, 7:28 pm), Opportunity Access, Case Access, Role Name (Sales_Executive), Role Name as displayed on reports, Sharing Groups (Role: Role and Internal Subordinates), and a note about edit permissions. Below this is a table titled 'Users in Sales Executive Role' with no records displayed.

This screenshot shows the 'Sales Manager' role page. The left sidebar is identical to the previous one. The main content area displays the 'Sales Manager' role details. The 'Role Detail' section includes fields for Label (Sales Manager), This role reports to (Sales Executive), Modified By (Dhummor Srinath, 24/10/2024, 7:28 pm), Opportunity Access, Case Access, Role Name (Sales_Manager), Role Name as displayed on reports, Sharing Groups (Role: Role and Internal Subordinates), and a note about edit permissions. Below this is a table titled 'Users in Sales Manager Role' with no records displayed.

This screenshot shows the 'Customer' role page. The left sidebar is identical to the previous ones. The main content area displays the 'Customer' role details. The 'Role Detail' section includes fields for Label (Customer), This role reports to (Sales Manager), Modified By (Dhummor Srinath, 24/10/2024, 7:29 pm), Opportunity Access, Case Access, Role Name (Customer), Role Name as displayed on reports, Sharing Groups (Role: Role and Internal Subordinates), and a note about edit permissions. Below this is a table titled 'Users in Customer Role' with no records displayed.

Task 5: Create a Property Details App

An App where the objects will be displayed

Activity 1

- Create a new Lightning App named “Property Details” and add “Customer” and “Property” objects, then assign it to the “System Admin” profile.

The screenshot shows the 'Lightning Experience App Manager' interface. The 'App Manager' tab is selected. A search bar at the top right contains 'Search Setup'. Below the search bar, there are buttons for 'New Lightning App' and 'New Connected App'. The main area displays a table of apps:

App Name	Developer Name	Description	Last Modified ...	App Type	Vi...
15 Platform	Platform	The fundamental Lightning Platform	16/07/2024, 4:09 pm	Classic	▼
16 Playground Starter	Playground Starter	Get started with your Trailhead Playground.	16/07/2024, 4:09 pm	Lightning (Managed)	▼
17 Property Details	Property_Details		25/10/2024, 9:51 pm	Lightning	▼

The screenshot shows the 'App Settings' page for the 'Property Details' app. The 'App Details & Branding' tab is selected. On the left, a sidebar lists 'App Options', 'Utility Items (Desktop Only)', 'Navigation Items', and 'User Profiles'. The main area contains the following sections:

- App Details**: Fields for 'App Name' (Property Details) and 'Developer Name' (Property_Details).
- App Branding**: Fields for 'Image' (with an 'Upload' button) and 'Primary Color Hex Value' (#0070D2).
- Org Theme Options**: A checkbox labeled 'Use the app's image and color instead of the org's custom theme'.
- App Launcher Preview**: A preview showing a blue square with 'PD' and the text 'Property Details'.

The screenshot shows the 'App Settings' page for the 'Property Details' app. The 'App Options' tab is selected. On the left, a sidebar lists 'App Details & Branding', 'Utility Items (Desktop Only)', 'Navigation Items', and 'User Profiles'. The main area contains the following sections:

- Navigation and Form Factor**:
 - Navigation Style**: Radio buttons for 'Standard navigation' (selected) and 'Console navigation'.
 - Supported Form Factors**: Radio buttons for 'Desktop and phone' (selected), 'Desktop', and 'Phone'.
- Setup and Personalization**:
 - Setup Experience**: Radio buttons for 'Setup (full set of Setup options)' (selected) and 'Service Setup'.
 - App Personalization Settings**: Checkboxes for 'Disable end user personalization of nav items in this app', 'Disable temporary tabs for items outside of this app', and 'Use Omni-Channel sidebar'.

Lightning App Builder | App Settings | Pages | Property Details | ? Help

App Settings

App Details & Branding
App Options
Utility Items (Desktop Only)

Navigation Items
User Profiles

Utility Items (Desktop Only)

Utility Items (Desktop Only)
Give your users quick access to productivity tools and add background utility items to your app.

Add Utility Item Utility Bar Alignment: Default

The utility bar is a fixed footer that opens components in docked panels. Available only when the app is viewed in Lightning Experience on a desktop.

To enable the utility bar for this app, add a utility item.

Lightning App Builder | App Settings | Pages | Property Details | ? Help

App Settings

App Details & Branding
App Options
Utility Items (Desktop Only)

Navigation Items

User Profiles

Navigation Items

Navigation Items
Choose the items to include in the app, and arrange the order in which they appear. Users can personalize the navigation to add or move items, but users can't remove or rename the items that you add. Some navigation items are available only for phone or only for desktop. These items are dropped from the navigation bar when the app is viewed in a format that the item doesn't support.

Available Items Create

Type to filter list...
Accounts
All Sites
Alternative Payment Methods
Analytics
App Launcher

Selected Items

Customer - Sheet1
Property

Lightning App Builder | App Settings | Pages | Property Details | ? Help

App Settings

App Details & Branding
App Options
Utility Items (Desktop Only)

Navigation Items

User Profiles

User Profiles

User Profiles
Choose the user profiles that can access this app.

Available Profiles Selected Profiles

Type to filter list...
Analytics Cloud Integration User
Analytics Cloud Security User
Authenticated Website
Authenticated Website
B2B Reordering Portal Buyer Profile
Contract Manager
Custom: Sales Profile
Customer Community Login User
Customer Community Plus Login User

Customer
Manager

Task 6: Create Profiles

Create profiles as per business requirement

- Customer

- Clone the Salesforce Platform User profile, name it “Customer,” adjust custom app settings to include only Property Details, remove all standard object permissions, and set read and view all permissions for the Property object.

Profile
Customer

Name: Customer
User License: Salesforce Platform
Description: Created By: Dhumor Srinath, 24/10/2024, 7:34 pm
Modified By: Dhumor Srinath, 24/10/2024, 7:40 pm

Page Layouts

Customer	[View]	Property	[View]		
Custom App Settings	Visible Default	Visible Default			
Analytics Studio (standard__Insights)	<input type="checkbox"/>	<input type="checkbox"/>	Property Details (Property_Details)	<input checked="" type="checkbox"/>	<input type="checkbox"/>
App Launcher (standard__AppLauncher)	<input type="checkbox"/>	<input type="checkbox"/>	User Access & Permissions Assistant (Profile2PermSet_Profiles_to_Permission_Sets)	<input type="checkbox"/>	<input type="checkbox"/>
Platform (standard__Platform)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	WDC (standard__Work)	<input type="checkbox"/>	<input type="checkbox"/>

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

The screenshot shows the Salesforce Setup interface with the 'Profiles' page selected. The left sidebar shows 'Users' and 'Profiles' under 'Setup'. The main content area displays profile settings:

- Communication Subscription Consents:** Options for Communication Subscription Timings, Contacts, Contact Point Addresses, Contact Point Consents, and Contact Point Emails.
- Locations:** Options for Party Consents, Push Topics, Sellers, Streaming Channels, and User External Credentials.
- Custom Object Permissions:** For AppLogs and Customer objects, showing permissions for Basic Access (Read, Create, Edit, Delete), Data Administration (View All, Modify All), and System Admin (Read, Create, Edit, Delete, View All, Modify All).
- Platform Event Permissions:** For AppLogEvents, showing permissions for Basic Access (Read, Create) and System Admin (Read, Create).
- Session Settings:** Session Times Out After: 2 hours of inactivity; Session Security Level Required at Login.
- Password Policies:** User passwords expire in 90 days, 3 passwords remembered, Minimum password length 8, and Password complexity requirement: Must include alpha and numeric characters.

- Manager

- Clone the Salesforce Platform User profile, name it "Manager," then ensure only "Property Details" is checked in Custom App Settings, remove all StandardObject Permissions, uncheckall Custom Object Permissions, and enable "Modify All" for "Property" and "Customer."

The screenshots show the Salesforce Setup interface with the 'Profiles' page selected. The left sidebar shows 'Hyperforce Assistant' and 'Prospector Preferences' under 'Setup'.

Profile Detail:

- Name: Manager
- User License: Salesforce Platform
- Description:
- Created By: Dhumor Srinath, 24/10/2024, 7:36 pm
- Modified By: Dhumor Srinath, 24/10/2024, 7:40 pm

Custom Field-Level Security:

Object	Action	[View]
Invoice	[View]	
Invoice Line	[View]	
Lead	[View]	
Location	[View]	
Location Group	[View]	
Location Group Assignment	[View]	
Object Milestone	[View]	
ApplLog	[View]	
Customer	[View]	
Customer	[View]	
Property	[View]	

Custom App Settings:

App	Visible	Default
Analytics Studio (standard__Insights)	<input type="checkbox"/>	<input type="checkbox"/>
App Launcher (standard__AppLauncher)	<input type="checkbox"/>	<input type="checkbox"/>
Platform (standard__Platform)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

The screenshot shows the Salesforce Setup interface with the search bar set to "Search Setup". The main content area is titled "SETUP Profiles". It displays "Standard Object Permissions" for "Profiles". There are two large tables: one for "Standard Object Permissions" and one for "Custom Object Permissions".

Standard Object Permissions:

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

Custom Object Permissions:

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

Task 7: Create a CheckBoxfield on user

Create Field on the User as per the business requirement.

Activity 1

In Object Manager, search for "User," go to Fieldsand Relationships, and create a new field named "Verified" with the data type "Check Box."

The top screenshot shows the 'Fields & Relationships' section for the 'User' object. A new field 'Verified' has been created, defined as a checkbox type.

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Verified	Verified__c	Checkbox		

The bottom screenshot shows the 'Custom Field Definition Detail' page for the 'Verified' field. It provides detailed information such as Field Label (Verified), Field Name (Verified), API Name (Verified__c), Data Type (Checkbox), and Description (Help Text).

Task 8: Create Users

Create three different users with three different Roles and profiles as we have mentioned above.

- User 1

- In Setup, navigate to Administration > Users > New User, fill in the Last Name as "Executive," set the Role to "Sales Executive," select the License as "Salesforce,"

The screenshot shows the 'Users' section in the Setup menu. The 'All Users' list is displayed, showing various users with their details like Full Name, Alias, Username, Role, Active status, and Profile assigned.

Action	Full Name	Alias	Username	Role	Active	Profile
<input type="checkbox"/>	Brown, Courtney	cbrown	cbrown@pink2872024.com	Manager	<input checked="" type="checkbox"/>	Standard Platform User
<input type="checkbox"/>	Chatter Expert	Chatter	chatty.00dm00000007u4zbuua.uojc3b0uhm2@chatter.salesforce.com		<input checked="" type="checkbox"/>	Chatter Free User
<input type="checkbox"/>	Customer	cust	chandrakanthsrinath@gmail.com		<input type="checkbox"/>	Customer
<input type="checkbox"/>	Customer2	cust	chandrakanthsrinath@gmail.com		<input checked="" type="checkbox"/>	Customer
<input type="checkbox"/>	Executive	exec	chandrakanthsrinath568@gmail.com		<input type="checkbox"/>	System Administrator
<input type="checkbox"/>	Manager	mana	chandrakanthsrinath6@gmail.com		<input type="checkbox"/>	Manager
<input type="checkbox"/>	Srinath, Dhumnor	Dsrm	chandrakanthsrinath@mindful-unicorn-1f90r.com		<input checked="" type="checkbox"/>	Analytics Cloud Integration User
<input type="checkbox"/>	User_Integration	integ	integration@00dm00000007u4zbuua.com		<input checked="" type="checkbox"/>	Analytics Cloud Security User
<input type="checkbox"/>	User_Security	sec	insightssecurity@00dm00000007u4zbuua.com		<input checked="" type="checkbox"/>	Standard Platform User
<input type="checkbox"/>	Wheeler, Allison	awhee	chandrakanth@black2872024.com			

choose the Profile as "System Administrator," and save.

The screenshot shows the Salesforce Setup interface with the 'Users' tab selected. On the left, a sidebar lists various user management settings. The main area displays the 'User Executive' profile. The 'User Detail' section shows the following information:

Name	Executive	Role
Alias	exec	User License
Email	chandrankanthsrinath8975@gmail.com [Verify]	Profile
Username	chandrankanthsrinath5986@gmail.com	System Administrator
Nickname	User1729793506728660201	Active
Title	User1729793506728660201	Marketing User
Company		Offline User
		Knowledge User

- User 2

- In Setup, navigate to Administration > Users > New User, fill in the Last Name as "Manager," set the Role to "Sales Manager," select the License as "Salesforce Platform," choose the Profile as "Manager," and save.

The screenshot shows the Salesforce Setup interface with the 'Users' tab selected. On the left, a sidebar lists various user management settings. The main area displays a table titled 'All Users' with one record listed:

Action	Full Name	Alias	Username	Role	Active	Profile
<input type="checkbox"/>	Manager	mana	chandrankanthsrinath06@gmail.com		<input checked="" type="checkbox"/>	Manager

The screenshot shows the Salesforce Setup interface with the 'Users' tab selected. On the left, a sidebar lists various user management settings. The main area displays the 'User Manager' profile. The 'User Detail' section shows the following information:

Name	Manager	Role
Alias	mana	User License
Email	chandrankanthsrinath06@gmail.com [Verify]	Profile
Username	chandrankanthsrinath06@gmail.com	Manager
Nickname	User1729794705943153822	Active
Title	User1729794705943153822	Marketing User
Company		Offline User
		Knowledge User

User 3

In Setup, navigate to Administration > Users > New User, fill in the Last Name as "Customer," set the Role to "Customer," select the License as "Salesforce Platform," choose the Profile as "Customer," ensure the "Verified" check box is unchecked, and save.

The screenshots show the Salesforce Setup interface for creating a new user:

- Screenshot 1: All Users List**
Shows a list of existing users. One user named 'Customer' is listed with the following details:

Action	Full Name	Alias	Username	Role	Active	Profile
<input type="checkbox"/>	Chatter Expert	Chatter	chatty.0dddm000007u4zbua.ujpx3b0uhm2@chatter.salesforce.com		<input checked="" type="checkbox"/>	Chatter Free User
<input type="checkbox"/>	Customer	cust	chandakanthsrinath@gmail.com		<input checked="" type="checkbox"/>	Customer
<input type="checkbox"/>	Customer2	cust	chandakanthfghihsrinath@gmail.com		<input checked="" type="checkbox"/>	Customer
- Screenshot 2: User Detail - Customer**
Shows the detailed view for the user 'Customer'. The 'User Detail' section includes fields for Name (Customer), Alias (cust), Email (chandakanthsrinath@gmail.com), Username (chandakanthsrinath@gmail.com), Nickname (User17297795749876667900), Title (null), Company (null), and Role (Customer). The 'User License' is set to 'Salesforce Platform' and 'Active' is checked. Other tabs like 'Sharing', 'Reset Password', and 'View Summary' are also visible.
- Screenshot 3: New User Creation Form**
Shows the 'New User' creation form. It includes sections for 'Single Sign On Information' (Federation ID) and 'Additional Information' (Verified checkbox). Fields for City, Zip/Postal Code, State/Province, and Country are also present.

User 4

- In Setup, navigate to Administration > Users > New User, fill in the Last Name as

"Customer2," set the Role to "Customer," select the License as "Salesforce Platform," choose the Profile as "Customer," ensure the "Verified" check box is checked, and save.

The first screenshot shows the 'All Users' list. A new user 'Customer2' has been created with the following details:

Action	Full Name	Alias	Username	Role	Active	Profile
<input type="checkbox"/>	Chatter Expert	Chatter	chatty.00dm000007u4zbuua.ujoic3b0uhm2@chatter.salesforce.com		<input checked="" type="checkbox"/>	Chatter Free User
<input type="checkbox"/>	Customer	cust	chandrankanthsrinath@gmail.com		<input checked="" type="checkbox"/>	Customer
<input type="checkbox"/>	Customer2	cust	chandrankantdfghjhsrinath@gmail.com		<input checked="" type="checkbox"/>	Customer

The second screenshot shows the 'User Detail' page for 'Customer2'. The user's details are as follows:

Name	Customer2	Role	Salesforce Platform
Alias	cust	User License	Customer
Email	chandrankanthsrinath@gmail.com [Verify]	Profile	
Username	chandrankantdfghjhsrinath@gmail.com	Active	<input checked="" type="checkbox"/>
Nickname	User17297796429899976560	Marketing User	<input type="checkbox"/>
Title		Offline User	<input type="checkbox"/>
Company		Knowledge User	<input type="checkbox"/>
User Customer2 ~ Salesforce - Developer Edition		Flow User	<input type="checkbox"/>
		Service Cloud User	<input type="checkbox"/>

Task 9: Create an Approval Process for Property Object

An Approval process to approve or reject the records as according

- Activity 1

- In Setup, navigate to Process Automation > Approval Process, create a "Property Approval" process with criteria "Location is not equal to blank" and "Verified equals

false," set the next automated approver to "Manager," enable record editability for administrators or current approvers, display Property, Owner, Location, and Type on the approval page layout, set initial submitters as Property Owner and Sales Manager, save, add an approval step "Executive Approval" for all records with the approver "Sales Executive," save, add a field update "Verified Property" to set the Verified checkbox to True, add a field update "UnVerified Property" to set the Verified checkbox to False, and activate the approval process.

The top screenshot shows the 'Approval Processes' page with the process details:

- Process Name:** Property Approval
- Unique Name:** Property_Approval
- Description:** (Property: Location NOTEQUALTO blank) AND (Property: Verified EQUALS False)
- Entry Criteria:** Record Editability: Administrator OR Current Approver
- Active:** Active (checkbox checked)
- Next Automated Approver Determined By:** Manager of Record Submitter
- Approval Assignment Email Template:** Initial Submitters: Property Owner, Role, Sales Manager
- Created By:** Dhumnor Srinath, 24/10/2024, 7:57 pm
- Modified By:** Dhumnor Srinath, 24/10/2024, 7:57 pm
- Allow Submitters to Recall Approval Requests:** (checkbox unchecked)

The bottom screenshot shows the 'Step 1. Enter Name and Description' screen of the Approval Process Editor:

- Step 1 of 6**
- Enter Name and Description:** Enter a name and description for your new approval process.
- Process Name:** Property Approval
- Unique Name:** Property_Approval
- Description:** (Required Information)
- Buttons:** Save, Next, Cancel

Step 2. Specify Entry Criteria

If only certain types of records should enter this approval process, enter that criteria below. For example, only expense reports from employees at headquarters should use this approval process.

Specify Entry Criteria

Use this approval process if the following criteria are met

Field	Operator	Value	Logic
Property Location	not equal to	blank	AND
Property Verified	equals	false	AND
--None--	--None--	--None--	AND
--None--	--None--	--None--	AND
--None--	--None--	--None--	

Add Filter Logic...

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

- Created By
- Last Modified By
- Property Name
- Verified

Selected Fields

- Property Owner
- Location
- Type

Up Down

Click here to view an example

Approval Page Fields

Display approval history information in addition to the fields selected above

Security Settings

Step 6. Specify Initial Submitters

Using the options below, specify which users are allowed to submit the initial request for approval. For example, expense reports should normally be submitted for approval only by their owners.

Initial Submitters

Submitter Type	Search	for	Find
Available Submitters	--None--		
Allowed Submitters	Role: Sales Manager	Property Owner	

Add Remove

Submission Settings

Allow submitters to recall approval requests

The screenshot shows the Salesforce Setup interface. The left sidebar is expanded to show the 'Field Service' section, specifically the 'Field Updates' sub-section under 'Workflow Actions'. The main content area is titled 'All Workflow Field Updates' and displays a table of field updates for the 'Case' object. The table has columns for Action, Name, Field to Update, Operation, Value, and Last Modified Date. There are three entries: one for changing case priority to high, and two for setting 'Verified' status.

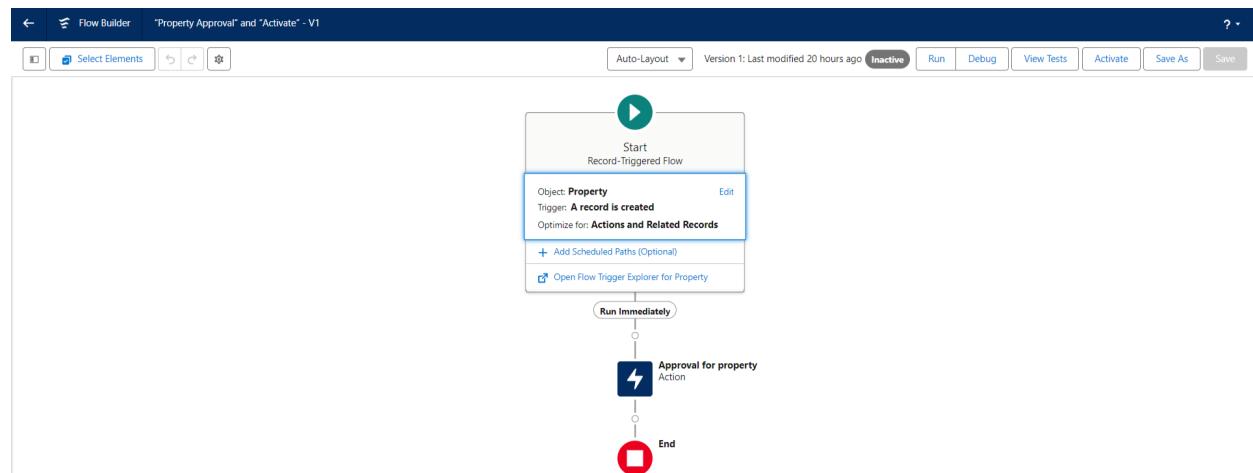
Action	Name	Field to Update	Operation	Value	Last Modified Date
Edit Del	Changes the case priority to high.	Case: Priority	Value	High	08/06/2024
Edit Del	Unverified	Property: Verified	Value		31/07/2024
Edit Del	Verified	Property: Verified	Value		31/07/2024

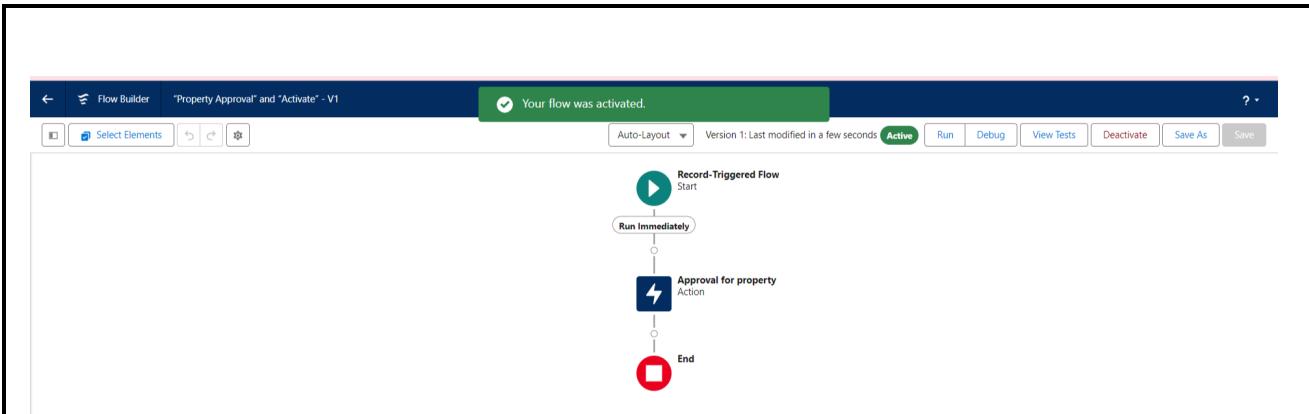
Task 10: Create a Record trigger flow to submit the Approval Process Automatically.

A flow that can submit the records directly for approval

- Activity 1

- In Setup, search for Flows, click on New, select "Record Trigger Flow," choose the Property object, set the trigger to "A record is created," add an action "Submit for Approval" with the label "Approval for property" and Record Id as {!\$Record.Id}, save the flow with the label "Property Approval," and activate.



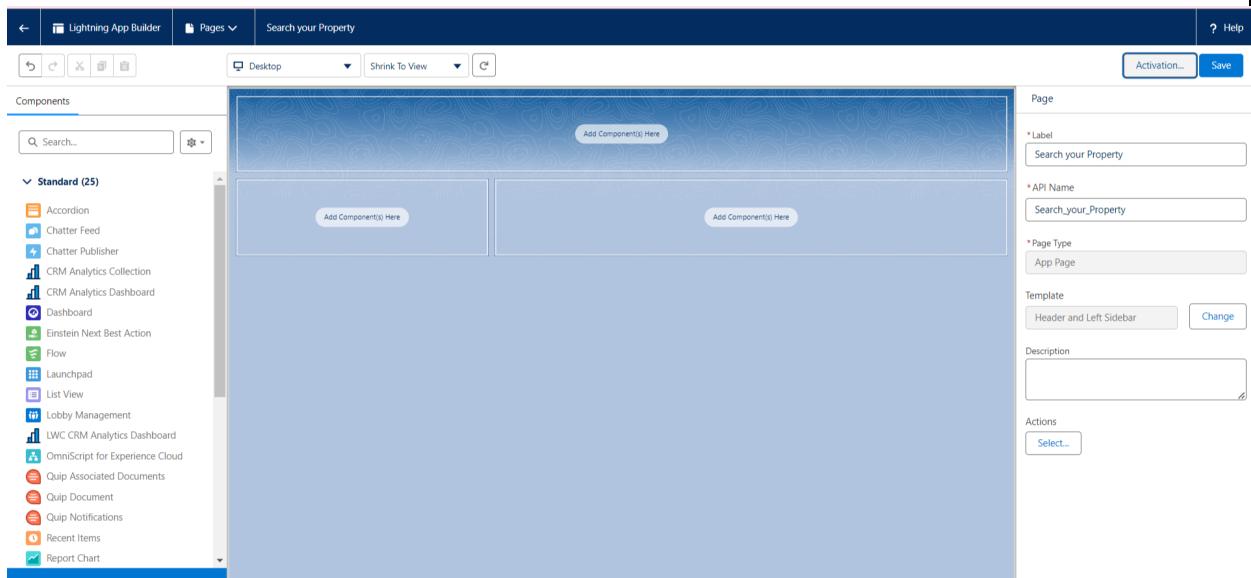


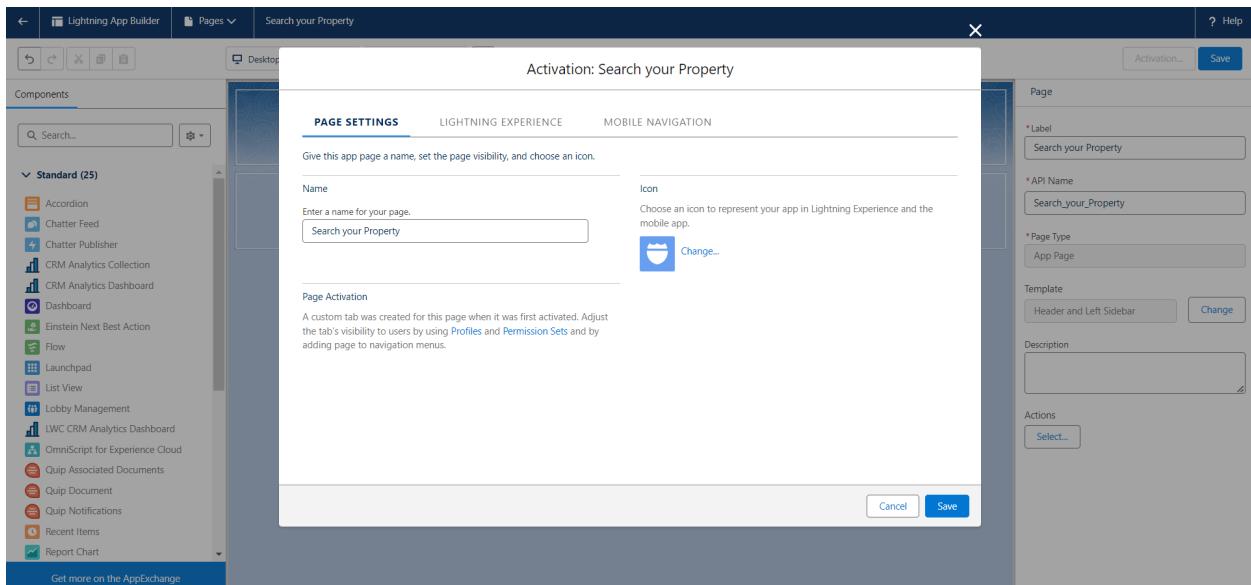
Task 11: Create an App Page

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

Activity 1

In Setup, navigate to Lightning App Builder, create a new App Page with the label "Search your Property," choose "Header and Left Sidebar" layout, save and activate the page, select "Activate for all Users" from Page Settings, and in Lightning Experience, add the page to "Property Details," then save.





Task 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”

- Activity 1

- Create an Apex class named "PropertyHandler_LWC" and make it Aura enabled, then in VSCode, authorize your org, create a Lightning Web Component, write the provided code in the HTML, JS, and metafile, and deploy the component to the org.

```

File ▾ Edit ▾ Debug ▾ Test ▾ Workspace ▾ Help ▾ < >
PropertyHandler_LWC.apex
Code Coverage: None | API Version: 61 | Go To
1 * public class PropertyHandler_LWC {
2
3     @AuraEnabled(cacheable=true)
4     public static List<Property__c> getProperty(String type, Boolean verified) {
5         String verifiedString = verified ? 'true' : 'false';
6
7         return [
8             SELECT Id, Location__c, Property_Name__c, Type__c, Verified__c
9             FROM Property__c
10            WHERE Type__c =: type
11            AND Verified__c =: verifiedString
12        ];
13    }
14 }

```

Logs	Tests	Checkpoints	Query Editor	View State	Progress	Problems
User: gourangi.srinidhi	Application: ApexTestHandler	Operation: ApexTestHandler	Time: 7/31/2024, 9:50:13 PM	Status: Success	Read: Unread	Size: 9.13 KB
gourangi.srinidhi	ApexTestHandler	ApexTestHandler	7/31/2024, 9:50:12 PM	Success	Unread	11.34 KB
gourangi.srinidhi	ApexTestHandler	ApexTestHandler	7/31/2024, 9:50:12 PM	Success	Unread	23.34 KB
gourangi.srinidhi	ApexTestHandler	ApexTestHandler	7/31/2024, 9:50:11 PM	Success	Unread	953 bytes
gourangi.srinidhi	ApexTestHandler	ApexTestHandler	7/31/2024, 9:50:11 PM	Success	Unread	5.11 KB
gourangi.srinidhi	ApexTestHandler	ApexTestHandler	7/31/2024, 9:50:10 PM	Success	Unread	12.54 KB



```
<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 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" alternativeLabel="Search" onclick={handleClick}></lightning-button-icon>
                    </div>
                </div>
                <template if:true={isttrue}>
                    <div class="slds-box">
                        <lightning-datatable key-field="id" data={propertylist} columns={columns}></lightning-datatable>
                    </div>
                </template>
                <template if:false={isfalse}>

```



The screenshot shows a code editor interface with the following details:

- File Menu:** File, Edit, Selection, View, Go, ...
- Toolbar:** Back, Forward, Search bar (salesforce), Minimize, Maximize, Close.
- Left Sidebar (EXPLORER):**
 - OPEN EDITORS:** Welcome (selected), new.js, new.html, new.js-meta.xml.
 - SALESFORCE:** force-app > main > default > lwc > new > new.html > template > lightning-card > template.
- Code Editor:** Displays the template for a Lightning component named "new". The code includes logic for handling true and false conditions using if:true and if:false directives. It also includes styling for the card using SLDs classes like "slds-box".

```
<template>
    <lightning-card>
        <div class="slds-box">
            <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>
```

```
force-app > main > default > lwc > new > JS new.js > C_01_Property_Management
1 import { LightningElement, api, track, wire } from 'lwc';
2 import getProperty from '@salesforce/apex/PropertyHandler_LWC.getProperty';
3 import { getRecord } from 'lightning/uiRecordApi';
4 import USER_ID from '@salesforce/user/Id';
5 export default class C_01_Property_Management extends LightningElement {
6     @api recordId;
7     userId = USER_ID;
8     verifiedvar;
9     typevar;
10    isfalse = true;
11    istrue = false;
12    @track propertylist = [];
13    columns = [
14        { label: 'Property Name', fieldName: 'Property_Name__c' },
15        { label: 'Property Type', fieldName: 'Type__c' },
16        { label: 'Property Location', fieldName: 'Location__c' },
17        { label: "Property link", fieldName: "Property_link__c" }
18    ]
19    propertyoptions = [
20        { label: "Commercial", value: "Commercial" },
21        { label: "Residential", value: "Residential" },
22        { label: "rental", value: "rental" }
23    ]
24    @wire(getRecord, { recordId: "$userId", fields: ['User.Verified__c'] })
25    recordFunction({ data, error }) {
26        if (data) {
```

EXPLORER ... Welcome JS new.js new.html new.js-meta.xml

force-app > main > default > lwc > new > JS new.js > C_01_Property_Management

```

5  export default class C_01_Property_Management extends LightningElement {
25    recordFunction({ data, error }) {
26      if (data) {
27        console.log(data)
28        console.log("This is the User Id ---> " + this.userId);
29        this.verifiedvar = data.fields.Verified__c.value;
30      } else {
31        console.error(error)
32        console.log('this is error')
33      }
34    }
35    changehandler(event) {
36      console.log(event.target.value);
37      this.typevar = event.target.value;
38    }
39    handleClick() {
40      getOwnProperty({ type: this.typevar, verified: this.verifiedvar })
41      .then(result => {
42        this.isfalse = true;
43        console.log(result)
44        console.log('This is the User id ---> ' + this.userId);
45        console.log('This is the verified values ---> ' + this.verifiedvar);
46        if (result != null && result.length != 0) {
47          this.isTrue = true;
48          this.propertylist = result;
49          console.log(this.verifiedvar);
50        }
51      })
52      .catch(error => {
53        console.log(error)
54      })
55    }
56  }
57}
58}
59}
60

```

EXPLORER ... Welcome JS new.js new.html new.js-meta.xml

force-app > main > default > lwc > new > JS new.js > C_01_Property_Management

```

5  export default class C_01_Property_Management extends LightningElement {
39    handleClick() {
40      .then(result => {
41        this.isfalse = true;
42        console.log(result)
43        console.log('This is the User id ---> ' + this.userId);
44        console.log('This is the verified values ---> ' + this.verifiedvar);
45        if (result != null && result.length != 0) {
46          this.isTrue = true;
47          this.propertylist = result;
48          console.log(this.verifiedvar);
49          console.log(this.typevar)
50        }
51      })
52      .else {
53        this.isfalse = false;
54        this.isTrue = false;
55      }
56    }
57    .catch(error => {
58      console.log(error)
59    })
60  }
61

```

EXPLORER ... Welcome JS new.js new.html new.js-meta.xml

force-app > main > default > lwc > new > JS new.js-meta.xml > XML

```

1  <?xml version="1.0" encoding="UTF-8"?>
2  <LightningComponentBundle xmlns="http://soap.sforce.com/2006/04/metadata">
3    <apiVersion>59.0</apiVersion>
4    <isExposed>true</isExposed>
5    <targets>
6      <target>lightning__RecordPage</target>
7      <target>lightning__AppPage</target>
8      <target>lightning__HomePage</target>
9      </targets>
10   </LightningComponentBundle>

```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS Salesforce CLI    ... ^ x  
18:13:22.864 ended SFDX: Deploy This Source to Org  
18:51:17.450 Starting SFDX: Deploy This Source to Org  
  
==== Deployed Source  


| STATE   | FULL NAME | TYPE                     | PROJECT PATH                                   |
|---------|-----------|--------------------------|------------------------------------------------|
| Changed | new       | LightningComponentBundle | force-app\main\default\lwc\new\new.html        |
| Changed | new       | LightningComponentBundle | force-app\main\default\lwc\new\new.js          |
| Changed | new       | LightningComponentBundle | force-app\main\default\lwc\new\new.js-meta.xml |

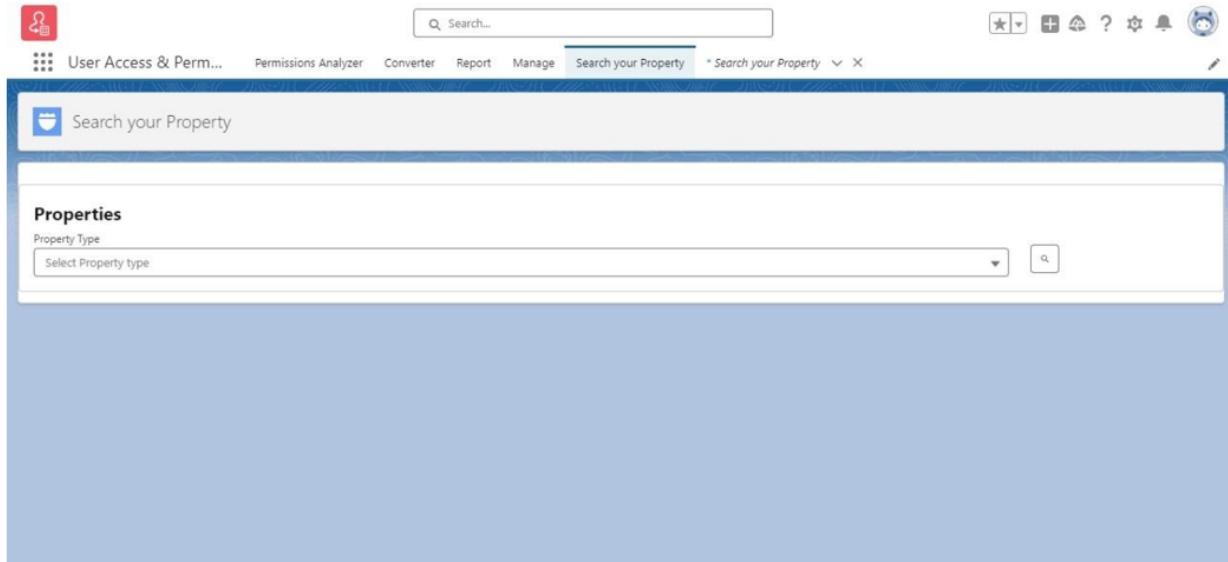
  
18:51:19.391 ended SFDX: Deploy This Source to Org
```

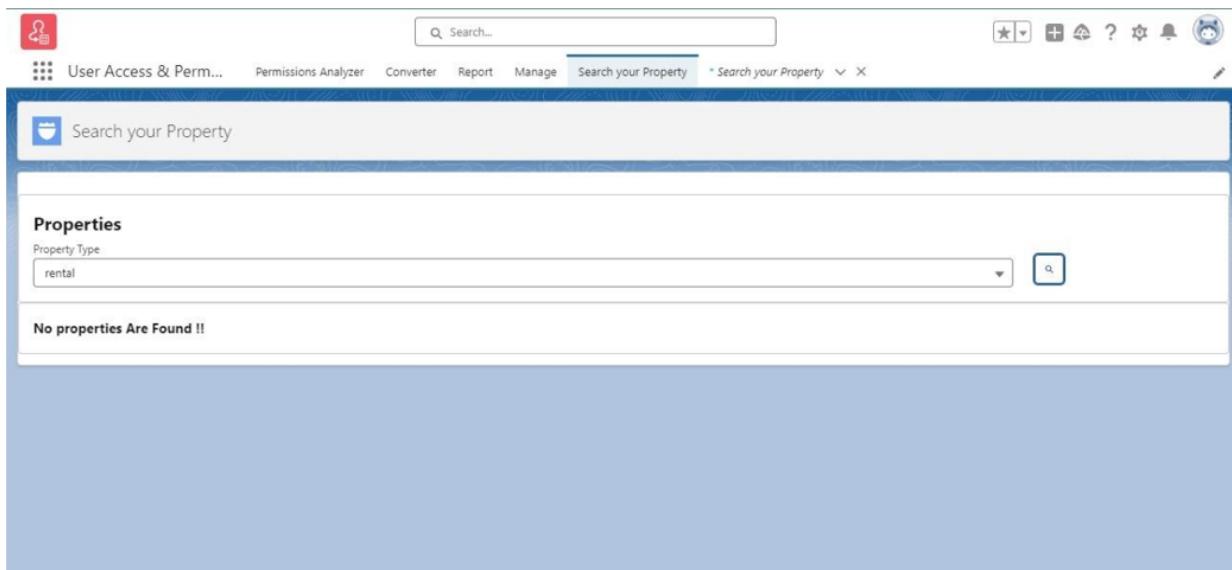
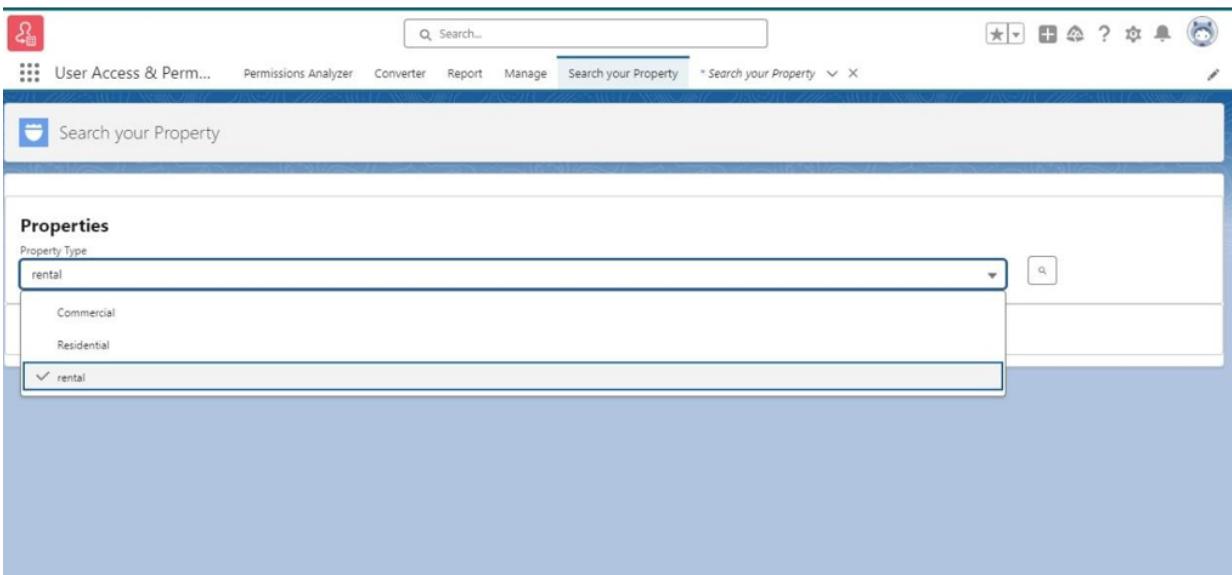
Task 13: Drag this Component to your App Page

Adding the Component to your Page

-Activity 1

-In Setup, go to App Launcher, search for Property Details, click the gear icon to Edit Page, drag the component to your App Page, and save the page.





Task 14: GiveAccessofApex Classes to Profiles

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

Activity 1

In Setup, search for Apex Classes, click on "Security" next to "PropertyHandler_LWC," add the "Manager" and "Customer" profiles, and save.

The image displays two screenshots of the Salesforce Setup interface.

Screenshot 1: Apex Classes

The left sidebar shows the navigation menu with "Apex Classes" selected under "Custom Code". The main content area is titled "Apex Classes" and lists various Apex classes with their details:

Action	Name	Version	Status	Last Modified	Owner
Edit Del	AnalyticsAuditorTest	61.0	Active	322	Dhumnor Srinath
Edit Del Security	SyncParkService	61.0	Active	1,434	Dhumnor Srinath
Edit Del Security	ParkLocator	61.0	Active	217	Dhumnor Srinath
Edit Del	ParkLocatorTest	61.0	Active	418	Dhumnor Srinath
Edit Del Security	parkService	61.0	Active	2,122	Dhumnor Srinath
Edit Del	ParkServiceMock	61.0	Active	612	Dhumnor Srinath
Edit Del Security	parksService	61.0	Active	2,130	Dhumnor Srinath
Edit Del Security	PropertyHandler_LWC	62.0	Active	36	Dhumnor Srinath

Screenshot 2: Profiles

The left sidebar shows the navigation menu with "Profiles" selected under "Custom Code". The main content area is titled "Profiles" and shows the configuration for the "PropertyHandler_LWC" class:

Enable Profile Access for Apex Class
PropertyHandler_LWC

Available Profiles: Analytics Cloud Integration User, Analytics Cloud Security User, Authenticated Website, B2B Reordering Portal Buyer Profile, Contract, Cross Org Data Proxy User, Custom Sales Profile, Customer Community Login User, Customer Community Plus Login User, Customer Community Plus User, Customer Community User, Customer Portal Manager Custom, Customer Portal Manager Standard, External Apps Login User

Enabled Profiles: Customer Manager, System Administrator

Add Remove

THANK YOU