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

PREPARED BY:

Harshvardhan Singh Chauhan

harshsc25@gmail.com

Abstract

Dreams World Properties has carefully integrated Salesforce into its processes, improving customer management and optimising property-related services. This innovative integration streamlines the capture of information about customers directly from the company's website, minimising manual data entry errors and producing detailed customer profiles in Salesforce. Dreams World Properties presents a more specific experience to its let users by breaking them down into approved and non-approved categories, recommending properties based on their unique tastes and requirements. At the same time, non-approved users are not left behind, since they can still view a wide range of property listings that meet general suitability requirements.

This seamless Salesforce integration not only optimizes operational efficiency by automating repetitive processes but also liberates the company's workforce from manual tasks, allowing staff to focus on more strategic objectives. As a result, customer satisfaction has improved, and the business has positioned itself for growth in the highly competitive real estate market. By leveraging Salesforce technology, Dreams World Properties has significantly elevated its ability to deliver superior customer service, provide tailored property recommendations, and streamline internal workflows for enhanced operational performance. The integration demonstrates the company’s commitment to utilizing cutting-edge technology to maintain its leadership in the industry, establishing a strong foundation for future success.

Introduction

Dreams World Properties has harnessed the power of Salesforce to revolutionise 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 system categorises 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 optimises 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.

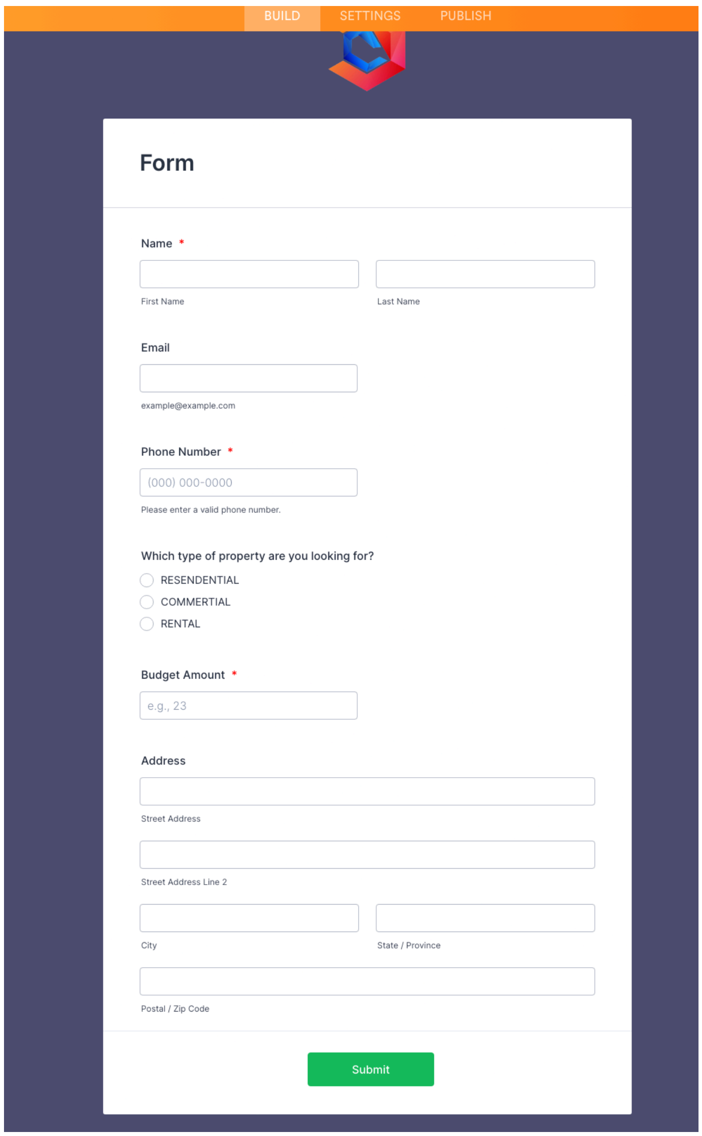
Approved users receive tailored property suggestions, making their search more efficient and enjoyable, while non-approved users have access to comprehensive listings, ensuring no potential client is left out. This seamless integration showcases Dreams World Properties' commitment to leveraging advanced 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.

SL NO.	TASK
1.	Create a Jotform and integrate it with the org to create a record of customers automatically.
2.	Create Objects from Spreadsheet.
3.	Integrate Jotform with Salesforce Platform
4.	Create Roles
5.	Create a Property Details App
6.	Create Profiles
7.	Create a CheckBox field on user
8.	Create Users
9.	Create an Approval Process for Property Object
10.	Create a Record trigger flow to submit the Approval Process Automatically.
11.	Create an App Page
12.	Create a LWC Component
13.	Drag this Component to your App Page
14.	Give Access of Apex Classes to Profiles

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

Objective: Create a form to capture customer details automatically and create a record in Salesforce.

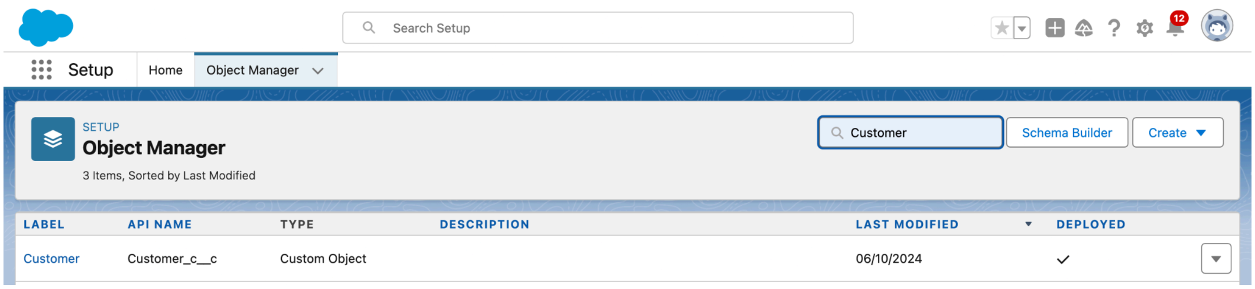
- **Activity**:
  a. Develop a JotForm to capture details such as Name, Phone, Email, Address, and the type of property the customer is interested in.
     MyForm: https://form.jotform.com/242782679233466



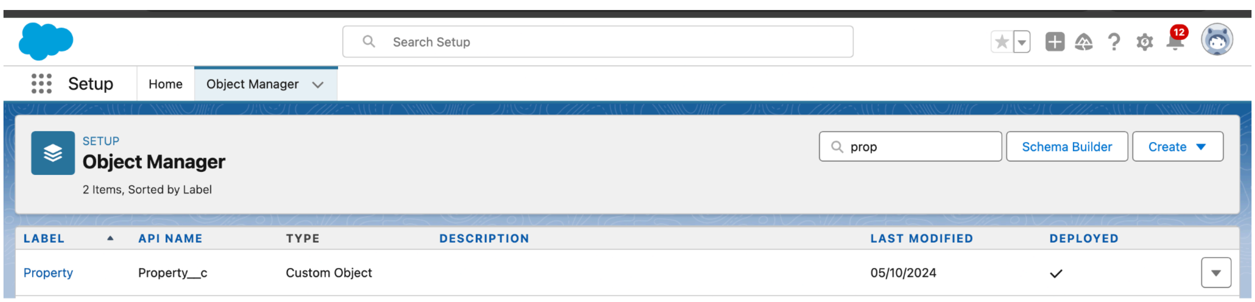
Task 2: Create Objects from Spreadsheet

Objective: Directly create objects in Salesforce from a spreadsheet.

- **Activity**:
    a. Create a `Customer` object.



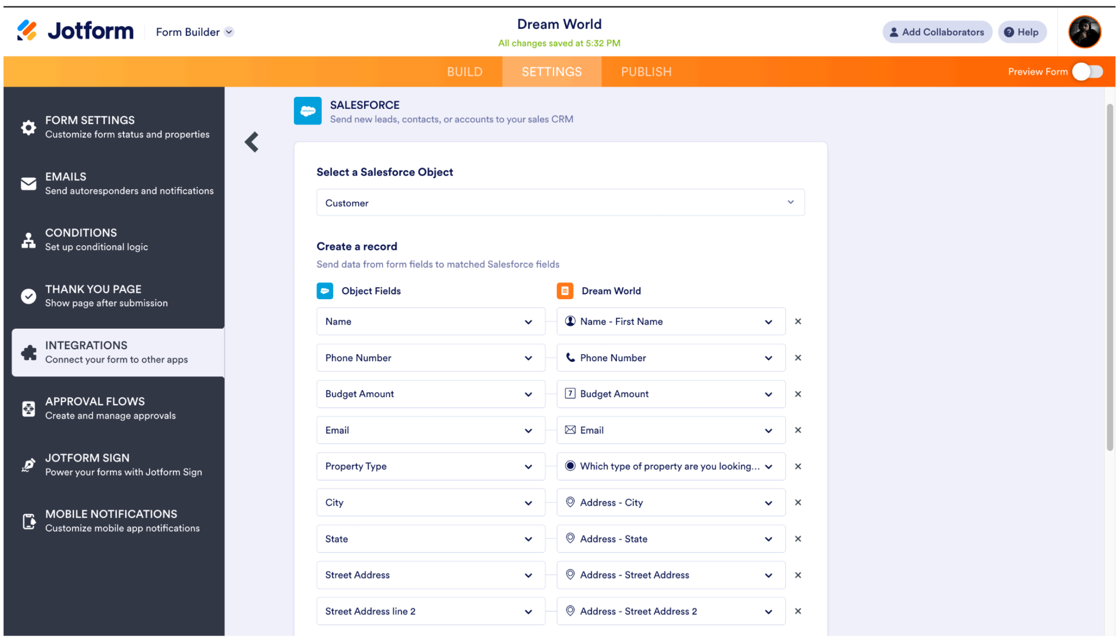
    b. Create a `Property` object.

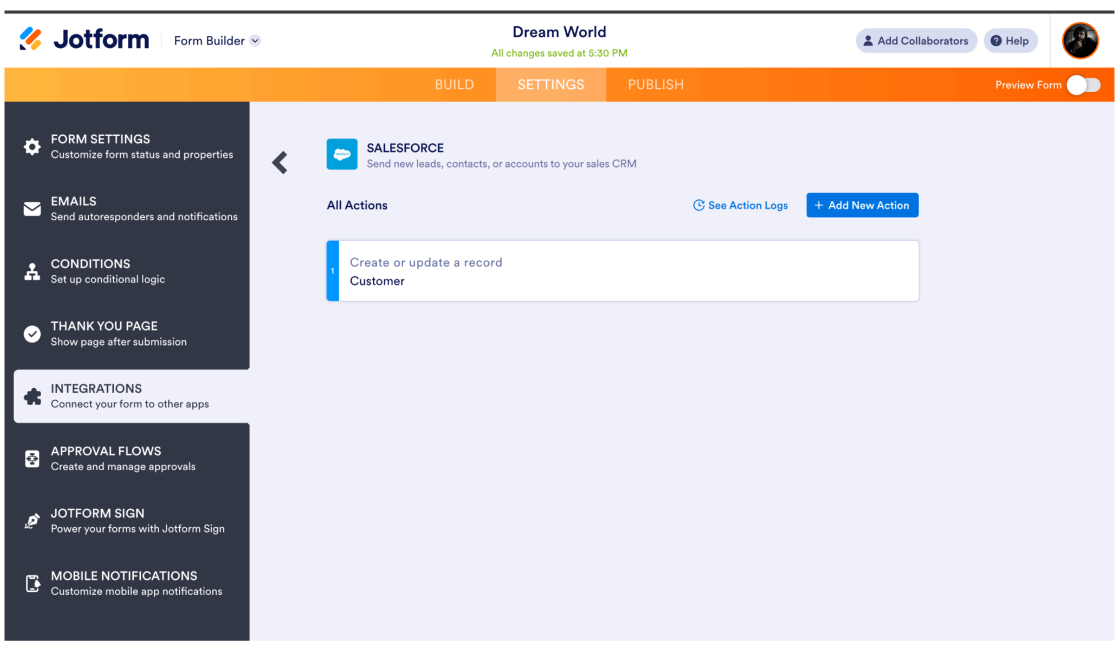


Task 3: Integrate JotForm with Salesforce

Objective: Integrate JotForm with Salesforce to create customer records automatically.

- **Activity**:
    a. Integrate JotForm with Salesforce to automate customer record creation.

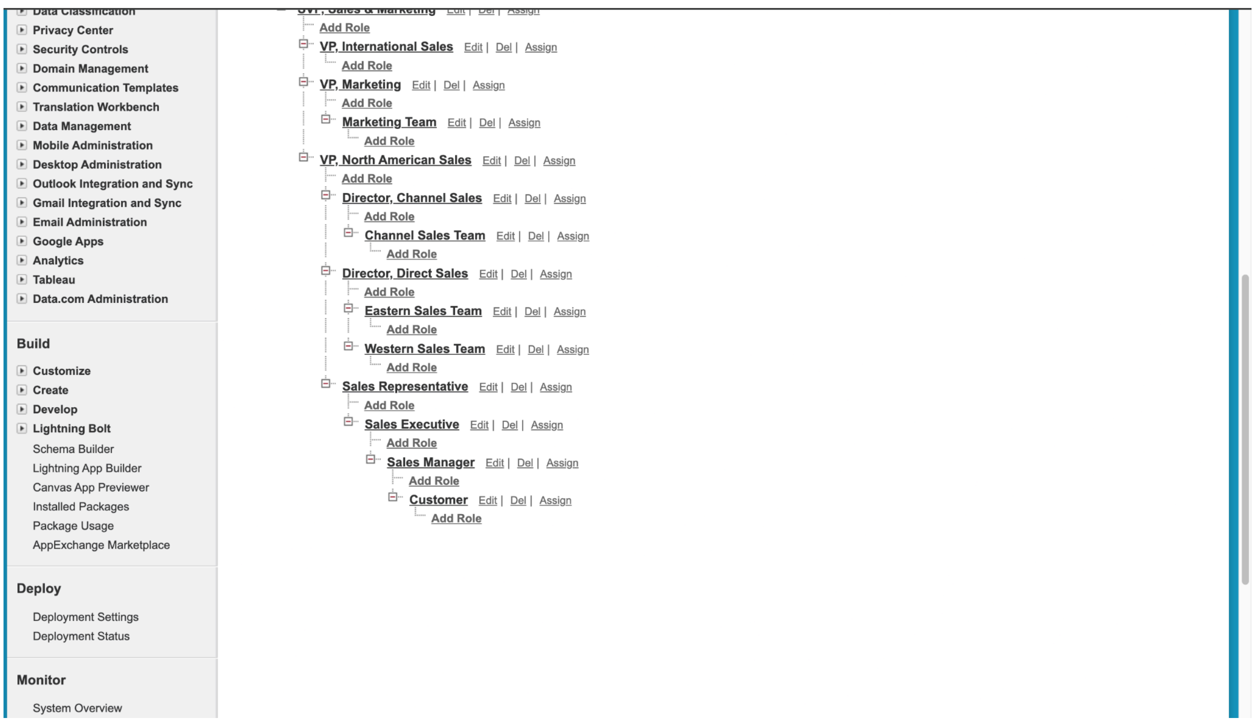




Task 4: Create Roles

Objective: Create roles as per the business requirements.

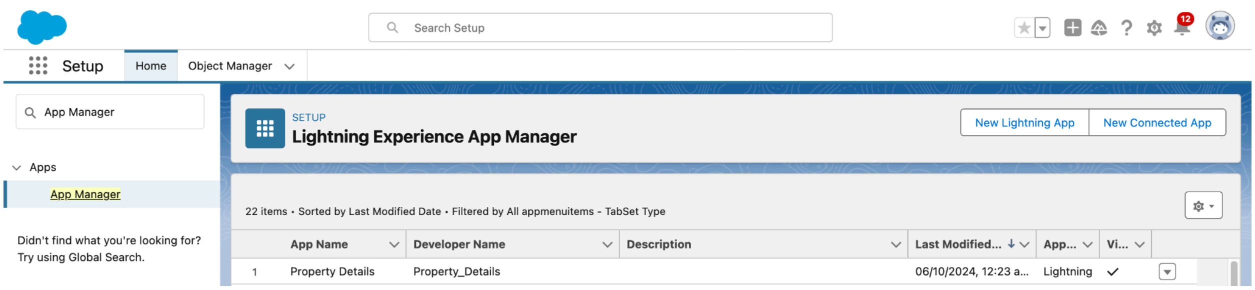
- **Activity**:
  a. Establish a hierarchical role structure where Sales Managers report to Sales Executives.
  b. Establish a hierarchical role structure where Customer report to Sales Managers.

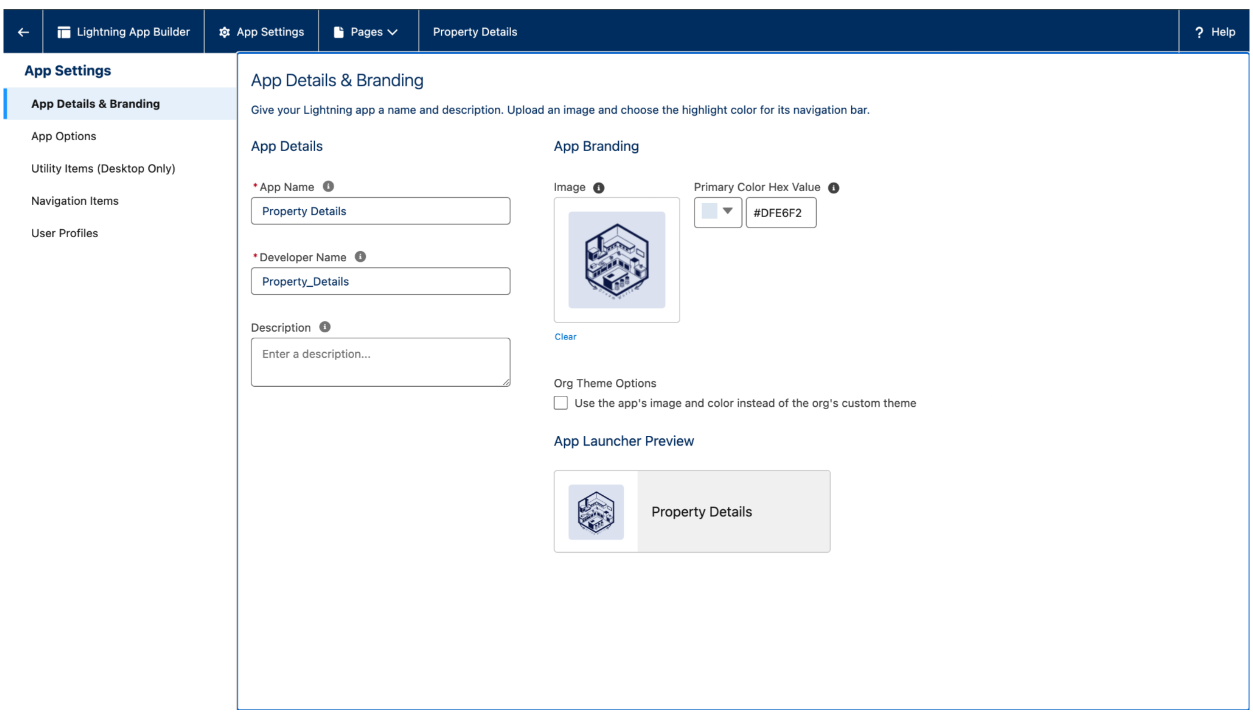


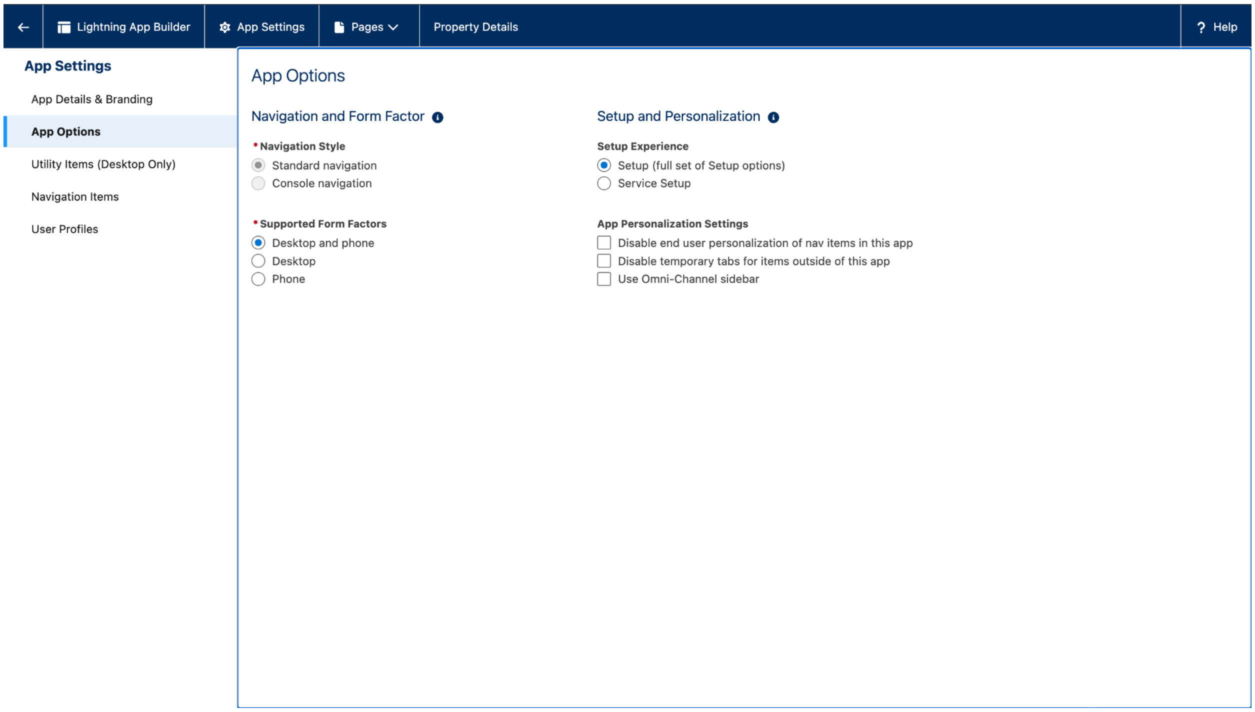
Task 5: Create a Property Details App

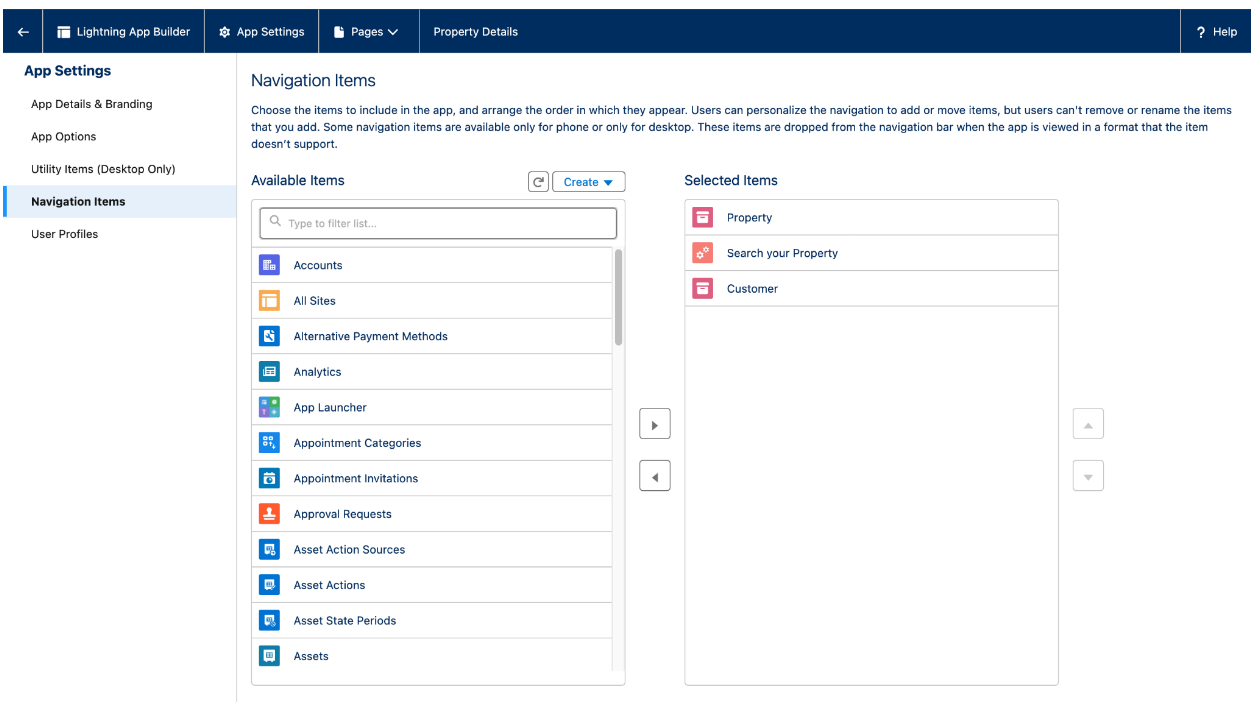
Objective: Create an app to display the created objects.

- **Activity**:
    a. Create a new Lightning App named "Property Details" and add `Customer` and `Property` objects.





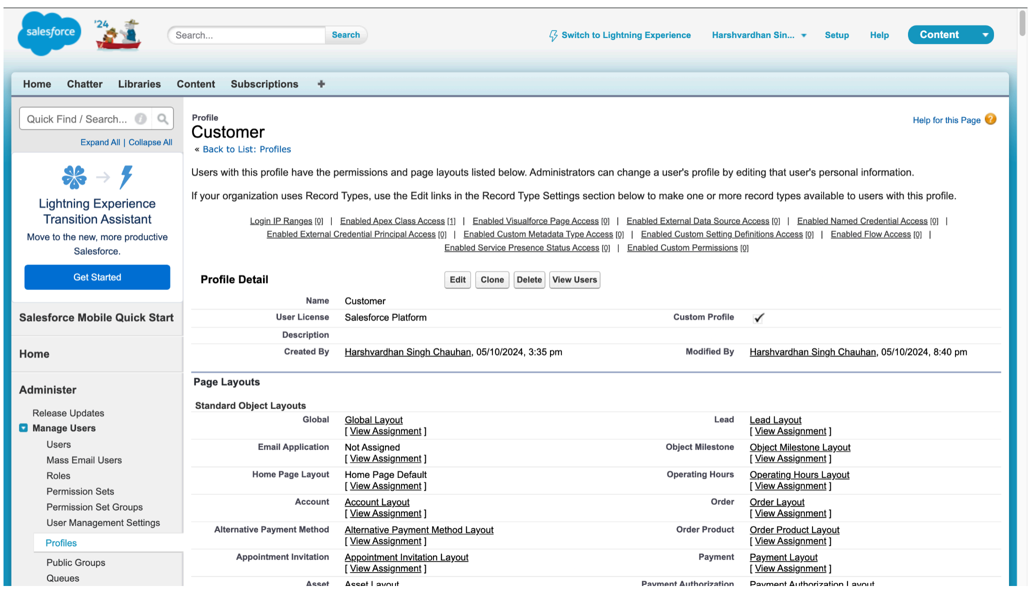

Lightning App Builder
App Settings
Pages
Property Details
Help
App Settings
App Details & Branding
App Options
Utility Items (Desktop Only)
Navigation Items
User Profiles
App Options
Navigation and Form Factor
Setup and Personalization
*Navigation Style
Standard navigation
Console navigation
Setup Experience
Setup (full set of Setup options)
Service Setup
*Supported Form Factors
Desktop and phone
Desktop
Phone
App Personalization Settings
Disable end user personalization of nav items in this app
Disable temporary tabs for items outside of this app
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
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
Appointment Categories
Appointment Invitations
Approval Requests
Asset Action Sources
Asset Actions
Asset State Periods
Assets
Selected Items
Property
Search your Property
Customer

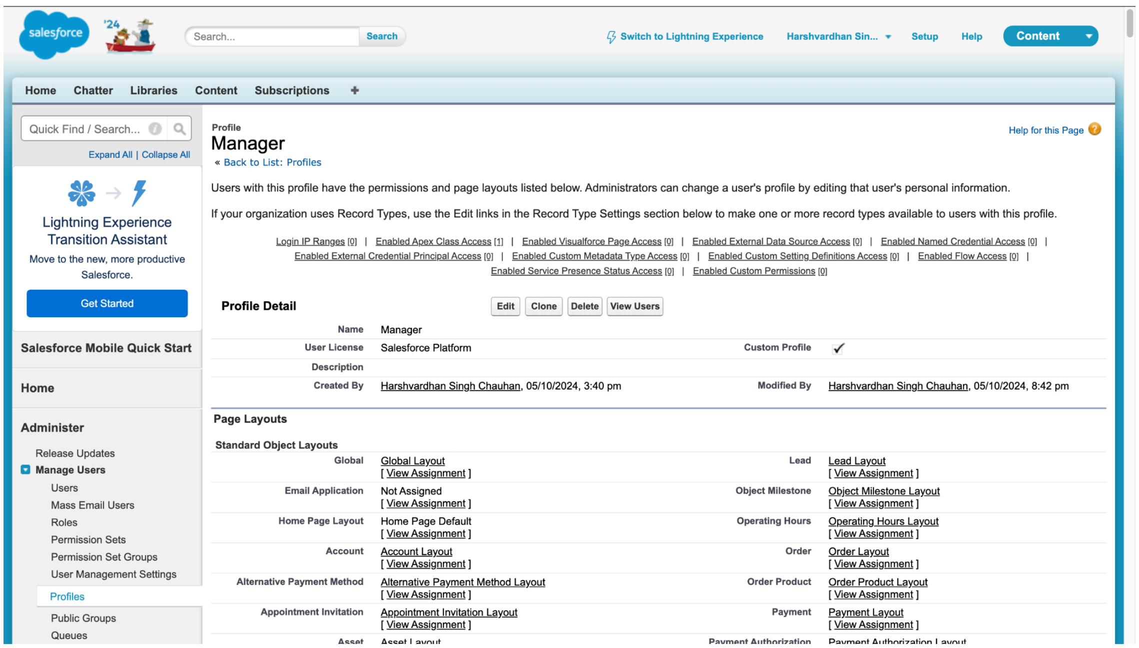
Task 6: Create Profiles

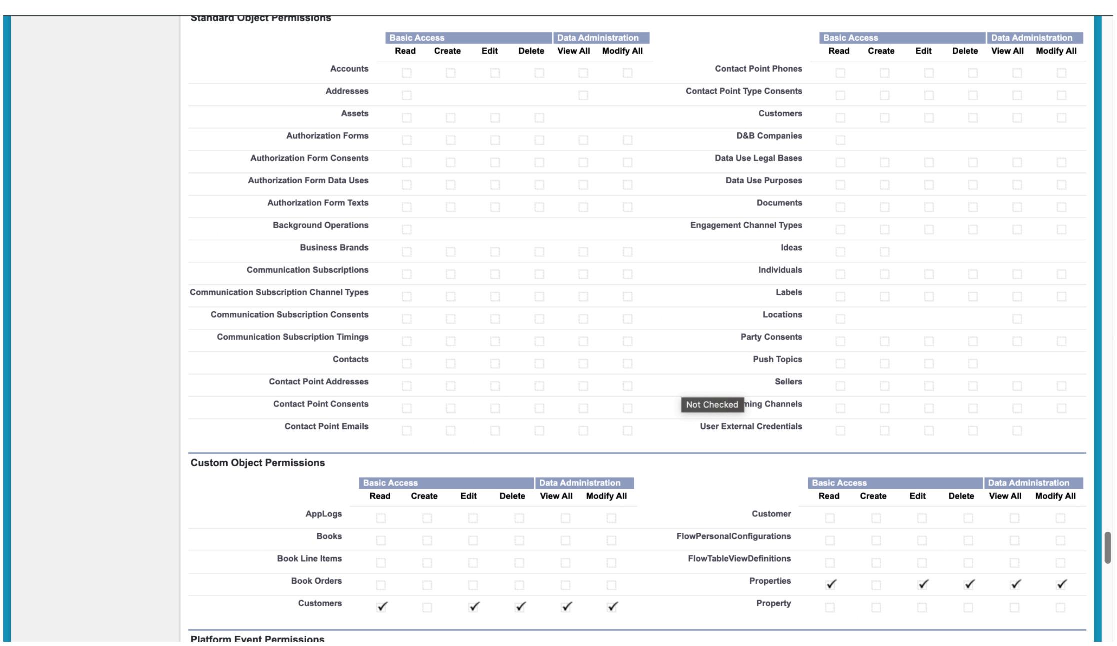
Objective: Set up profiles as per the business requirements.

- **Activity**:
    a. **Customer Profile**: Clone the Salesforce Platform User profile, name it "Customer," and adjust custom app settings to include only `Property Details`.



b. **Manager Profile**: Clone the Salesforce Platform User profile, name it "Manager," and enable permissions to modify both `Property` and `Customer` records.

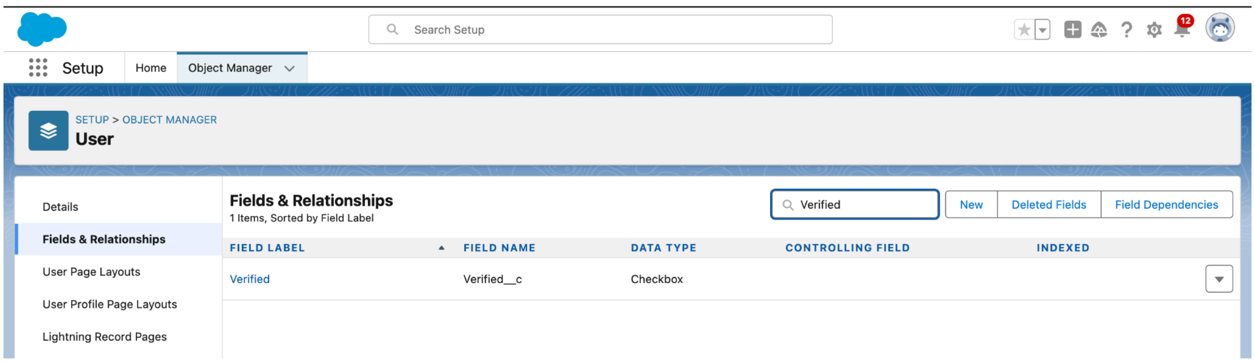


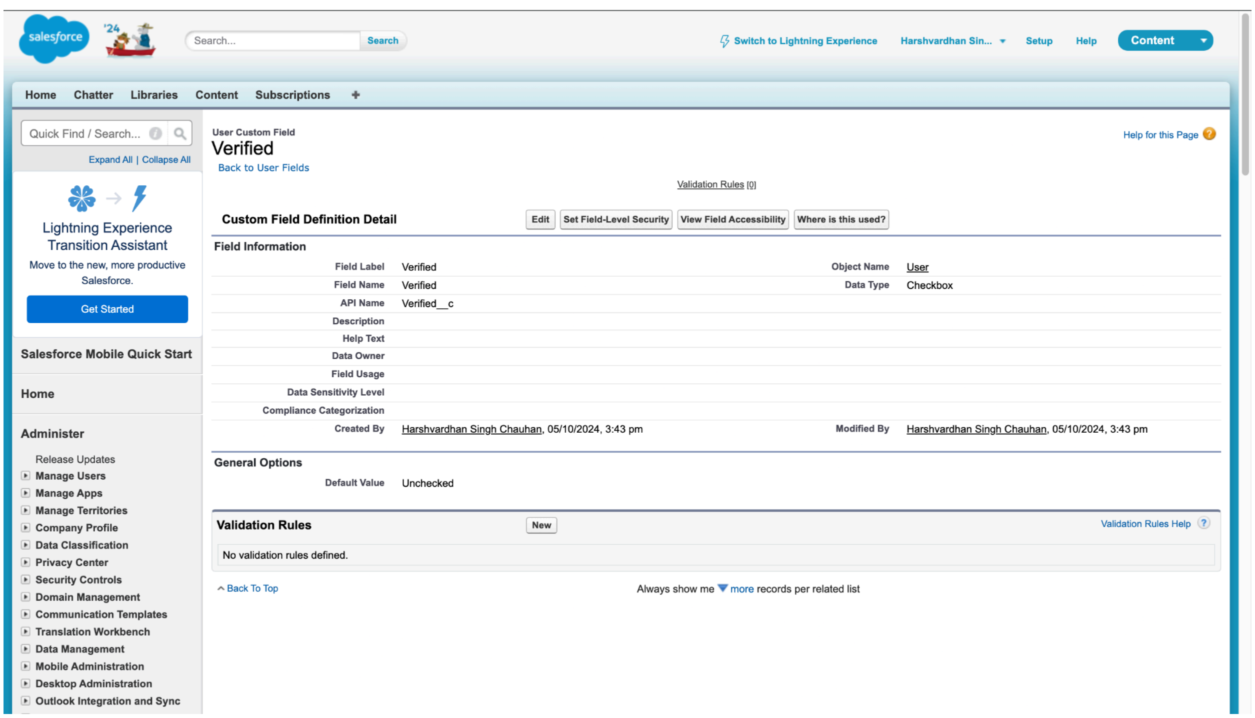


Task 7: Create a Checkbox Field on User

Objective: Create a checkbox field on the user object based on business requirements.

- **Activity**:
    a. Create a new field named "Verified" on the `User` object with the data type "Checkbox."

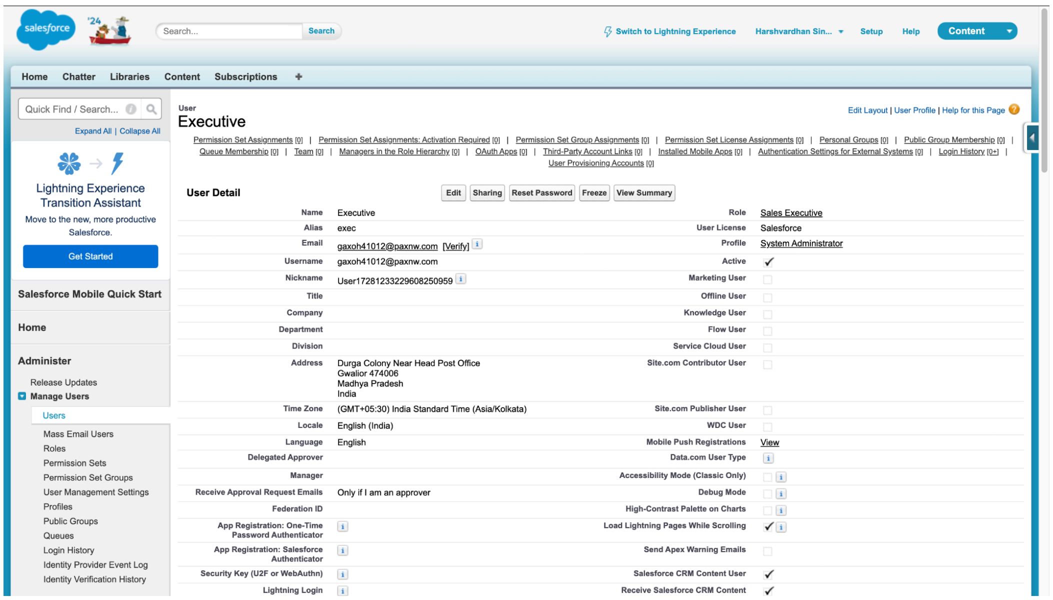




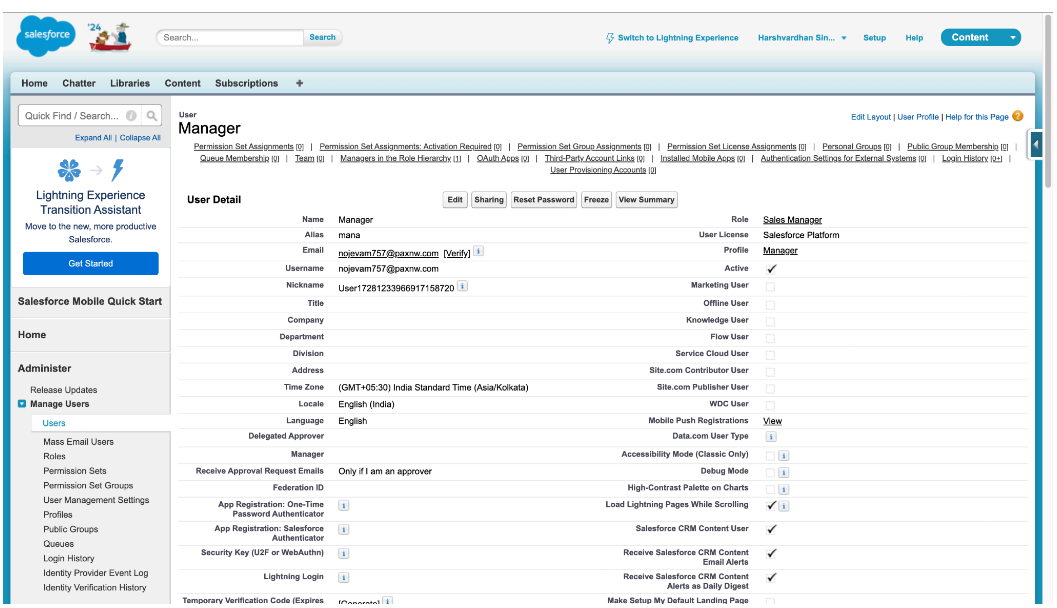
Task 8: Create Users

Objective: Create users with different roles and profiles.

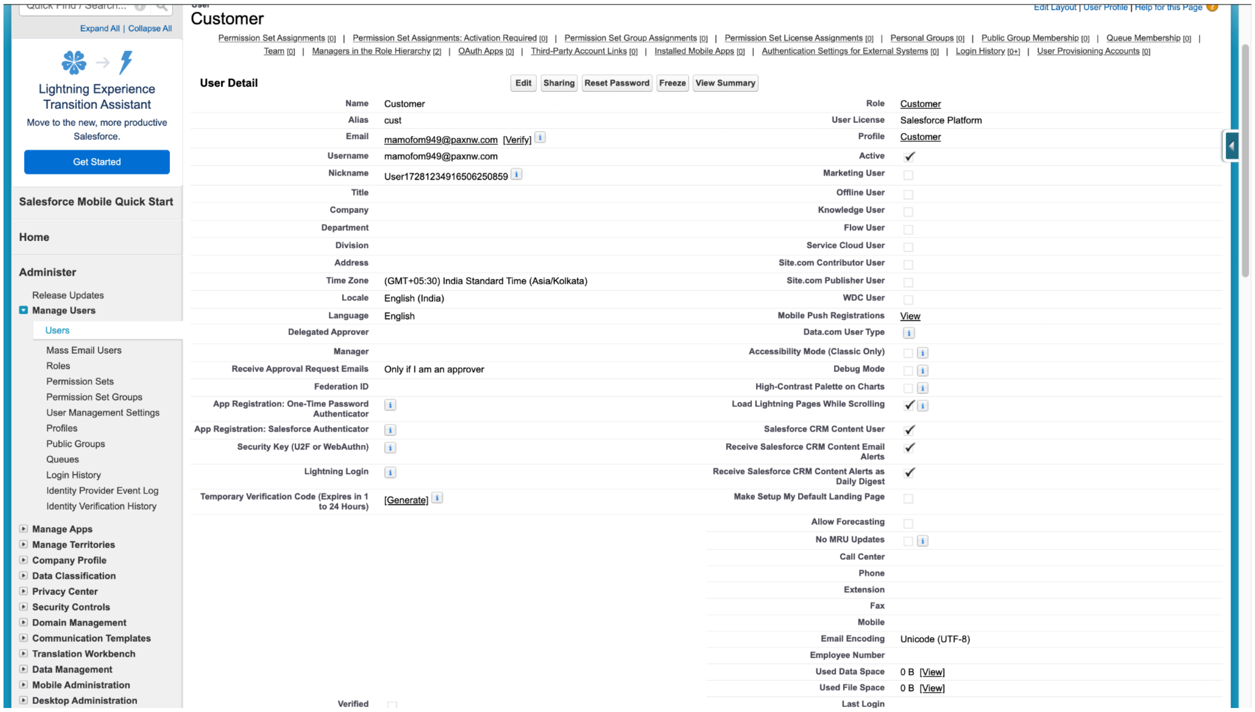
- **Activity**:
    a. **User 1**: Sales Executive with the System Administrator profile.



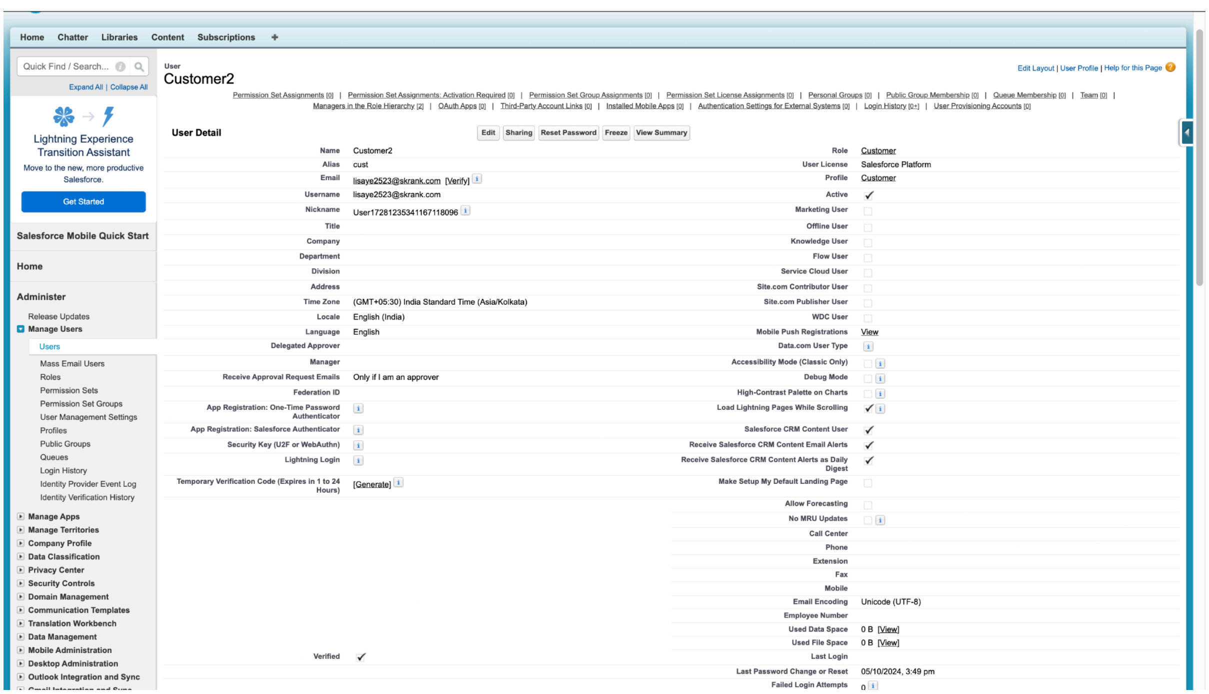
b. **User 2**: Manager with the Manager profile.



c. **User 3**: Customer with the Customer profile, "Verified" checkbox unchecked.



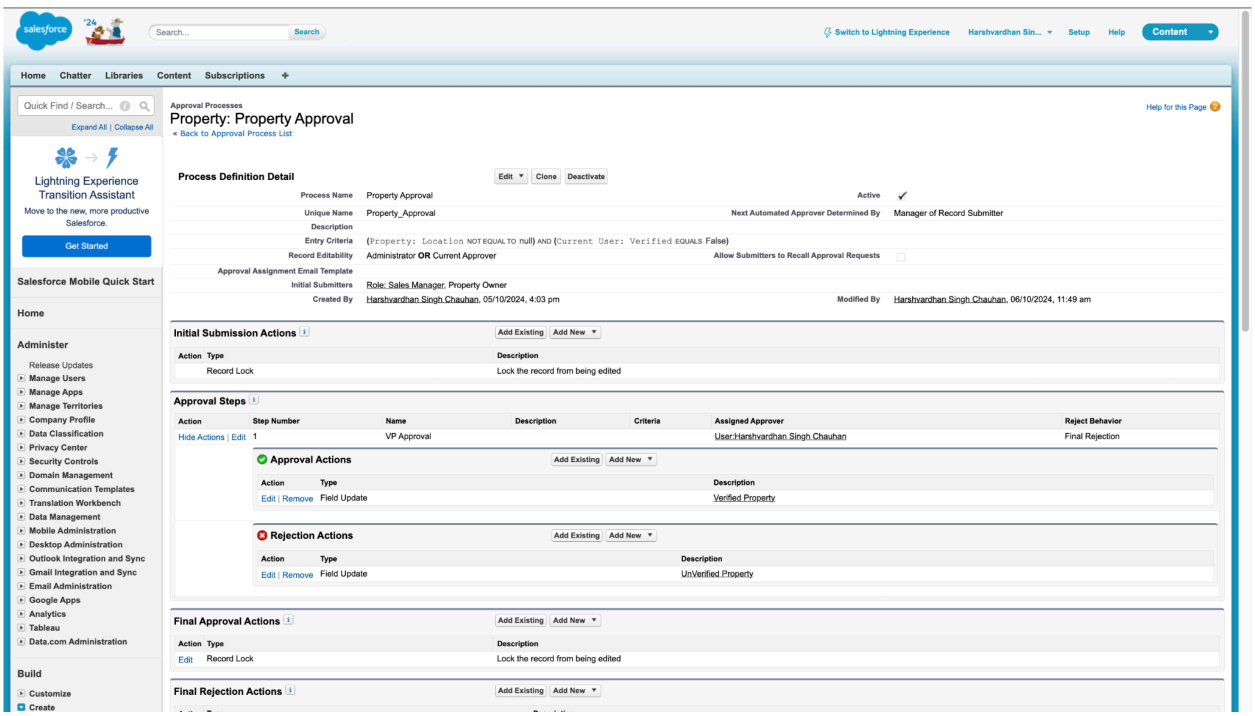
d. **User 4**: Customer2 with the Customer profile, "Verified" checkbox checked.

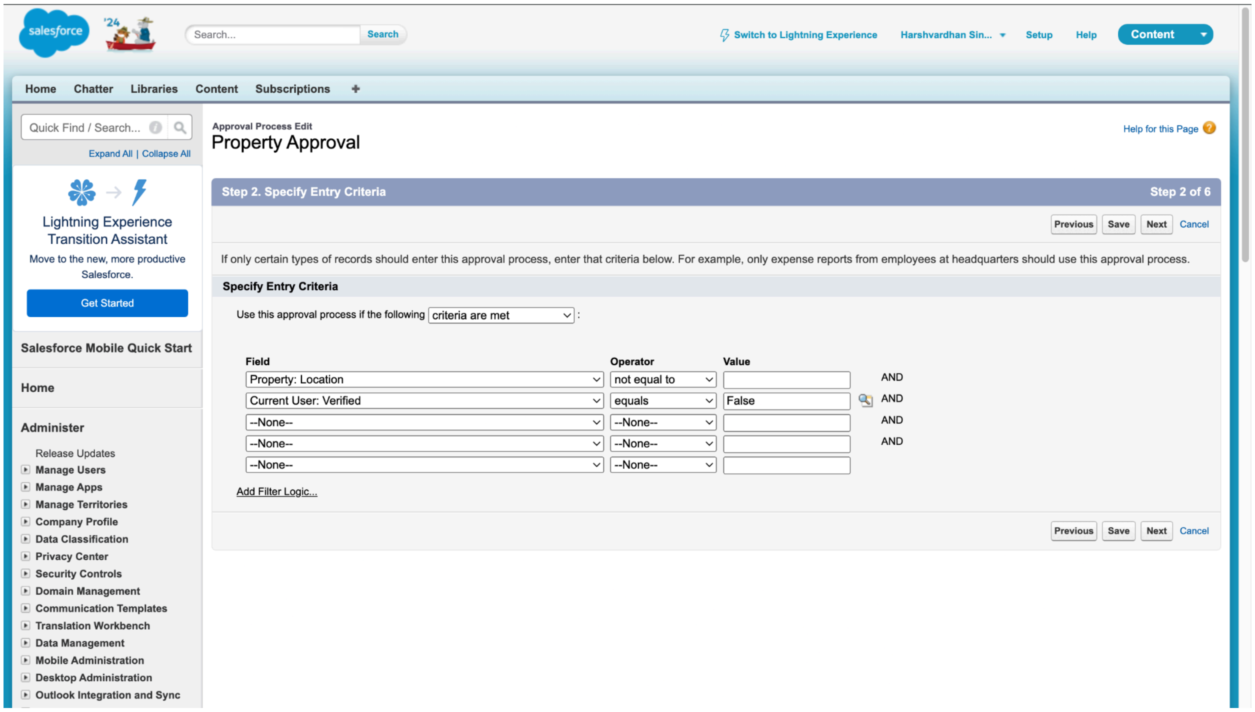


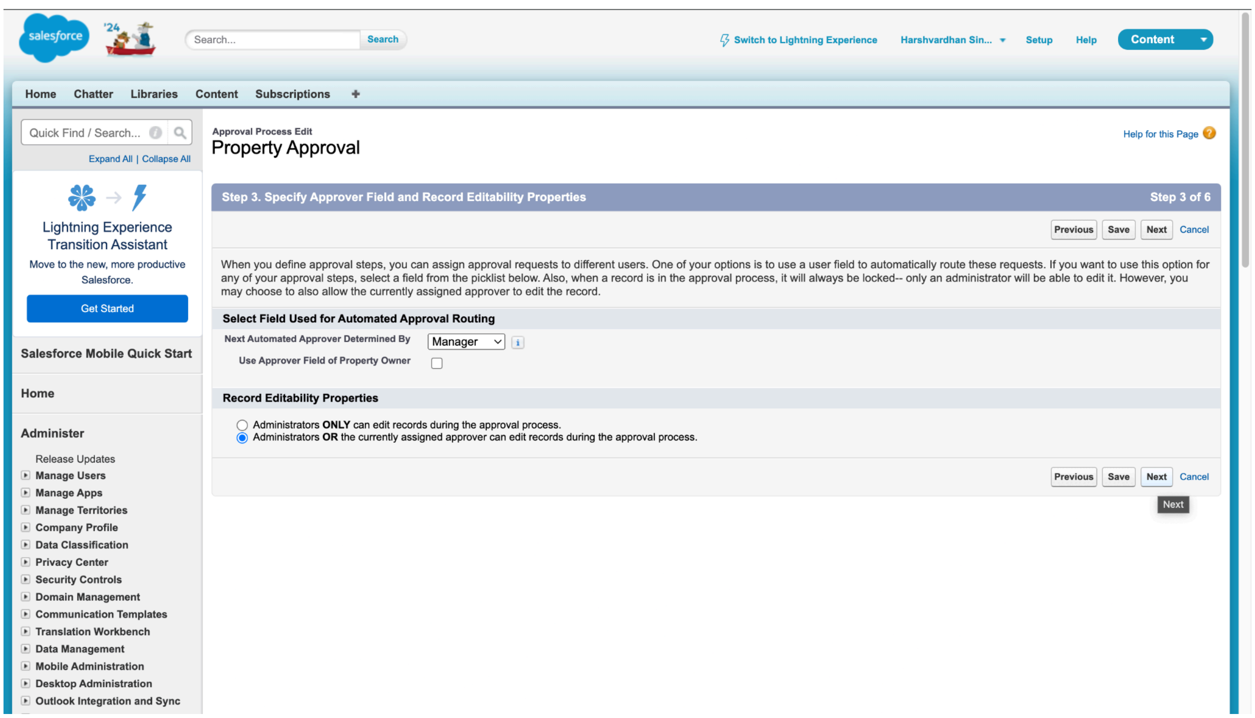
Task 9: Create an Approval Process for the Property Object

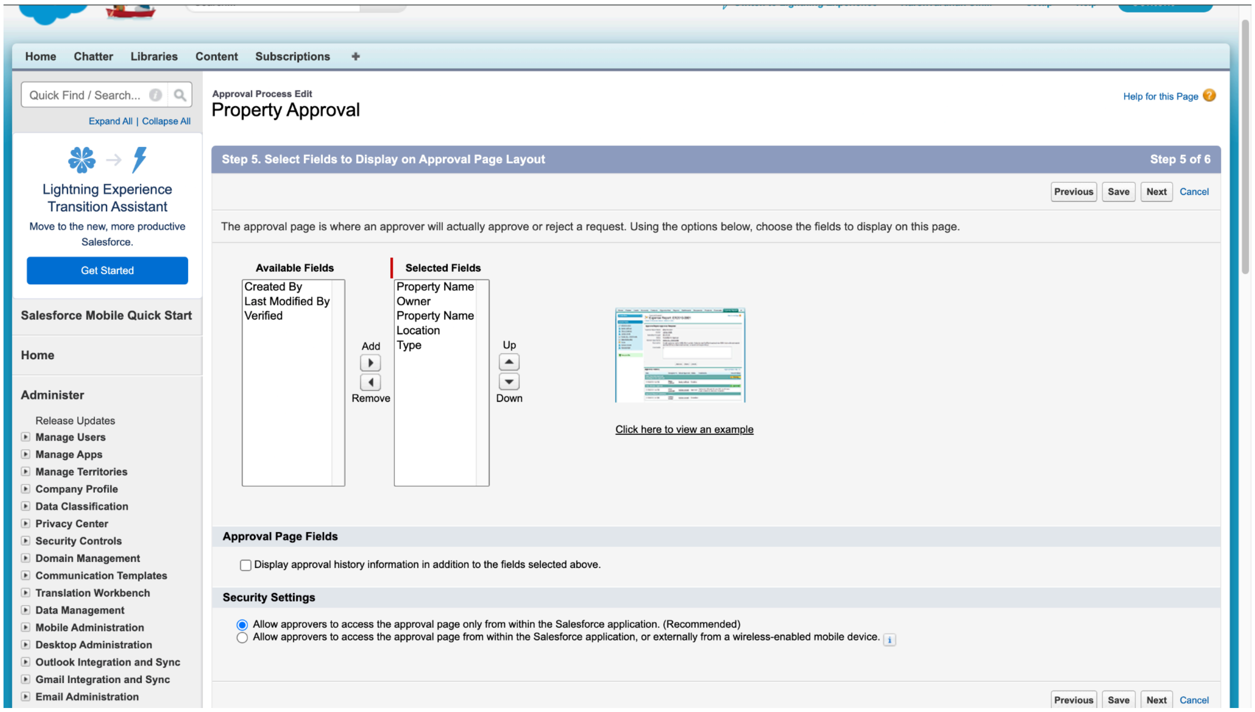
Objective: Establish an approval process for the Property object.

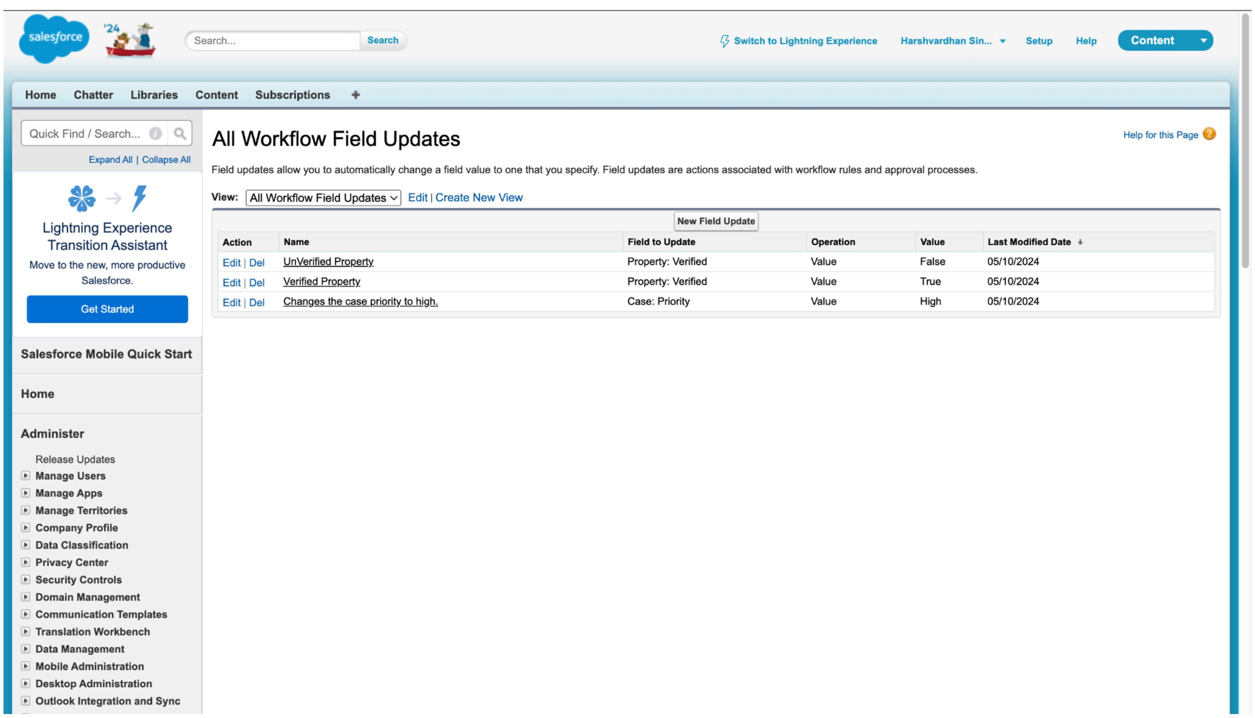
- **Activity**:
  a. Create a "Property Approval" process with criteria such as "Location is not blank" and "Verified equals false." Assign Managers and Sales Executives as approvers, and update the Verified status based on approval outcomes.

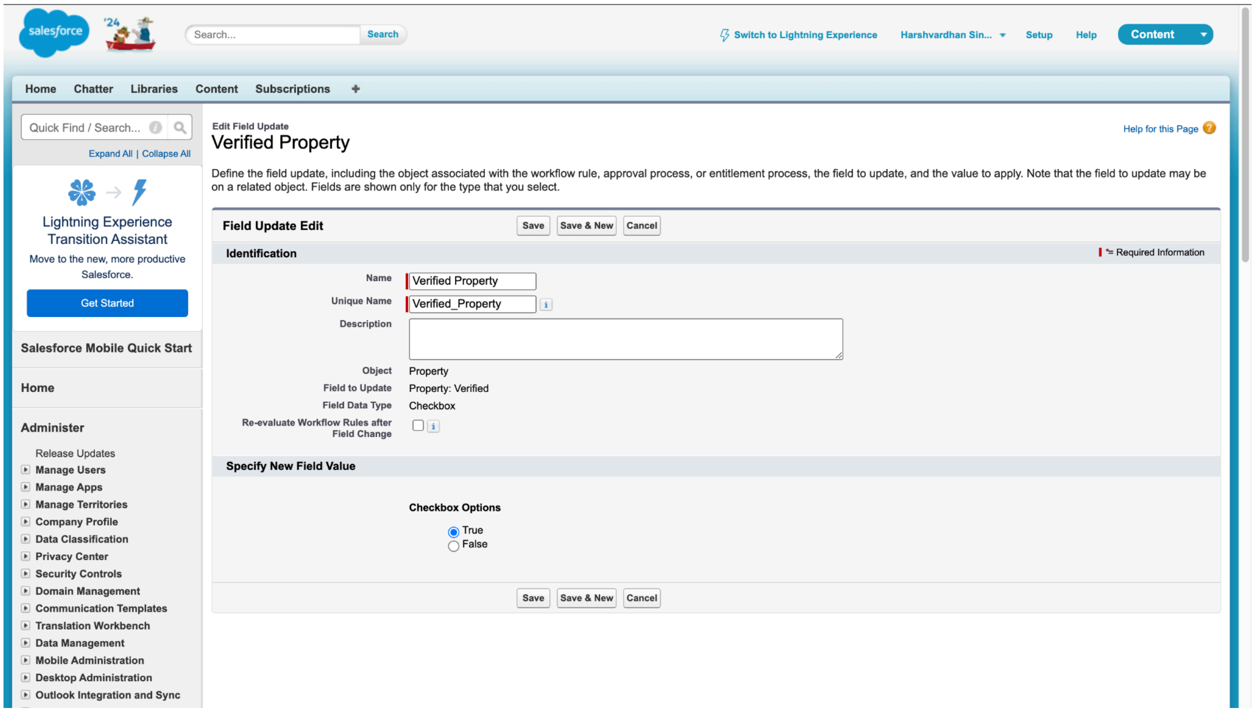


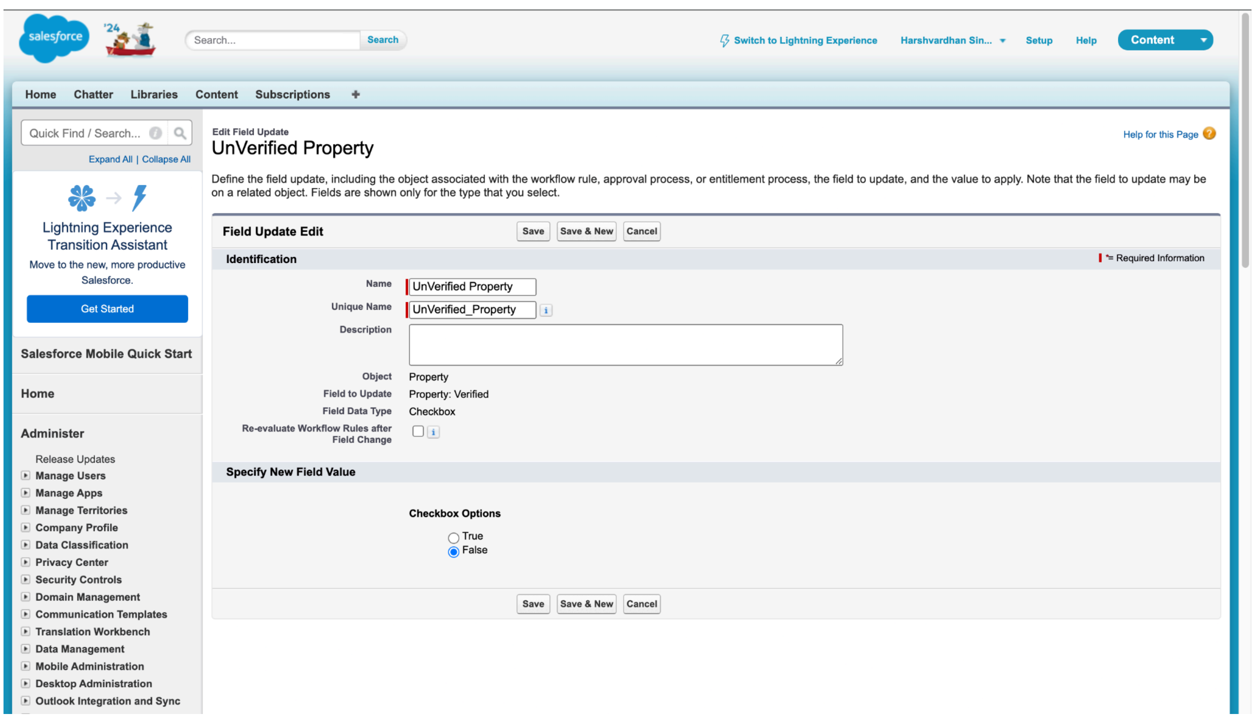

salesforce
'24
Search...
Search
Switch to Lightning Experience
Harshvardhan Sin...
Setup
Help
Content
Home
Chatter
Libraries
Content
Subscriptions
Quick Find / Search...
Expand All | Collapse All
Lightning Experience Transition Assistant
Move to the new, more productive Salesforce.
Get Started
Salesforce Mobile Quick Start
Home
Administer
Release Updates
Manage Users
Manage Apps
Manage Territories
Company Profile
Data Classification
Privacy Center
Security Controls
Domain Management
Communication Templates
Translation Workbench
Data Management
Mobile Administration
Desktop Administration
Outlook Integration and Sync
Approval Process Edit
Property Approval
Help for this Page
Step 2. Specify Entry Criteria
Step 2 of 6
Previous
Save
Next
Cancel
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
Property: Location
not equal to
AND
Current User: Verified
equals
False
AND
--None--
--None--
AND
--None--
--None--
AND
--None--
--None--
Add Filter Logic...
Previous
Save
Next
Cancel


salesforce
'24
Search...
Search
Switch to Lightning Experience
Harshvardhan Sin...
Setup
Help
Content
Home
Chatter
Libraries
Content
Subscriptions
Quick Find / Search...
Expand All | Collapse All
Lightning Experience Transition Assistant
Move to the new, more productive Salesforce.
Get Started
Salesforce Mobile Quick Start
Home
Administer
Release Updates
Manage Users
Manage Apps
Manage Territories
Company Profile
Data Classification
Privacy Center
Security Controls
Domain Management
Communication Templates
Translation Workbench
Data Management
Mobile Administration
Desktop Administration
Outlook Integration and Sync
Approval Process Edit
Property Approval
Help for this Page
Step 3. Specify Approver Field and Record Editability Properties
Step 3 of 6
Previous
Save
Next
Cancel
When you define approval steps, you can assign approval requests to different users. One of your options is to use a user field to automatically route these requests. If you want to use this option for any of your approval steps, select a field from the picklist below. Also, when a record is in the approval process, it will always be locked-- only an administrator will be able to edit it. However, you may choose to also allow the currently assigned approver to edit the record.
Select Field Used for Automated Approval Routing
Next Automated Approver Determined By
Manager
Use Approver Field of Property Owner
Record Editability Properties
Administrators ONLY can edit records during the approval process.
Administrators OR the currently assigned approver can edit records during the approval process.
Previous
Save
Next
Cancel
Next


Home
Chatter
Libraries
Content
Subscriptions
Quick Find / Search...
Expand All | Collapse All
Lightning Experience Transition Assistant
Move to the new, more productive Salesforce.
Get Started
Salesforce Mobile Quick Start
Home
Administer
Release Updates
Manage Users
Manage Apps
Manage Territories
Company Profile
Data Classification
Privacy Center
Security Controls
Domain Management
Communication Templates
Translation Workbench
Data Management
Mobile Administration
Desktop Administration
Outlook Integration and Sync
Gmail Integration and Sync
Email Administration
Approval Process Edit
Property Approval
Help for this Page
Step 5. Select Fields to Display on Approval Page Layout
Step 5 of 6
Previous
Save
Next
Cancel
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
Verified
Add
Remove
Selected Fields
Property Name
Owner
Property Name
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
Allow approvers to access the approval page only from within the Salesforce application. (Recommended)
Allow approvers to access the approval page from within the Salesforce application, or externally from a wireless-enabled mobile device.
Previous
Save
Next
Cancel


salesforce
'24
Search...
Search
Switch to Lightning Experience
Harshvardhan Sin...
Setup
Help
Content
Home
Chatter
Libraries
Content
Subscriptions
Quick Find / Search...
Expand All | Collapse All
Lightning Experience Transition Assistant
Move to the new, more productive Salesforce.
Get Started
Salesforce Mobile Quick Start
Home
Administer
Release Updates
Manage Users
Manage Apps
Manage Territories
Company Profile
Data Classification
Privacy Center
Security Controls
Domain Management
Communication Templates
Translation Workbench
Data Management
Mobile Administration
Desktop Administration
Outlook Integration and Sync
All Workflow Field Updates
Help for this Page
Field updates allow you to automatically change a field value to one that you specify. Field updates are actions associated with workflow rules and approval processes.
View: All Workflow Field Updates
Edit | Create New View
New Field Update
Action Name Field to Update Operation Value Last Modified Date
Edit | Del UnVerified Property Property: Verified Value False 05/10/2024
Edit | Del Verified Property Property: Verified Value True 05/10/2024
Edit | Del Changes the case priority to high. Case: Priority Value High 05/10/2024

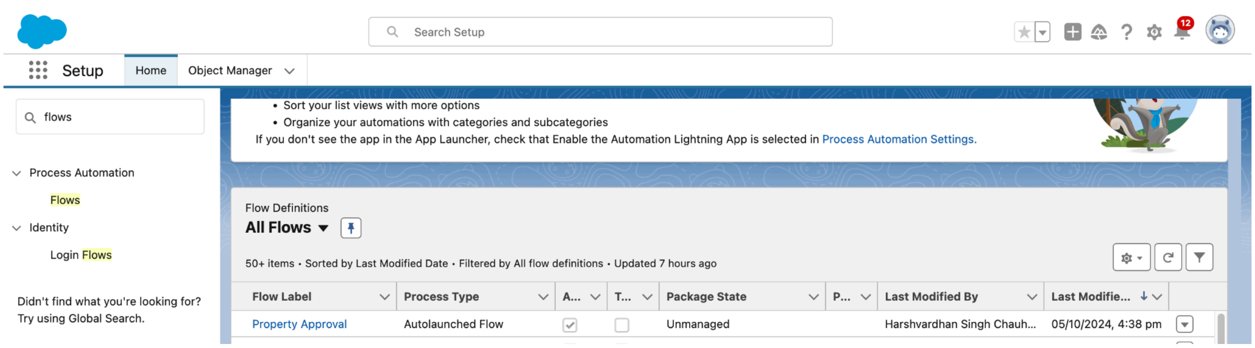

salesforce
'24
Search...
Search
Switch to Lightning Experience
Harshvardhan Sin...
Setup
Help
Content
Home
Chatter
Libraries
Content
Subscriptions
Quick Find / Search...
Expand All | Collapse All
Lightning Experience Transition Assistant
Move to the new, more productive Salesforce.
Get Started
Salesforce Mobile Quick Start
Home
Administer
Release Updates
Manage Users
Manage Apps
Manage Territories
Company Profile
Data Classification
Privacy Center
Security Controls
Domain Management
Communication Templates
Translation Workbench
Data Management
Mobile Administration
Desktop Administration
Outlook Integration and Sync
Edit Field Update
Verified Property
Help for this Page
Define the field update, including the object associated with the workflow rule, approval process, or entitlement process, the field to update, and the value to apply. Note that the field to update may be on a related object. Fields are shown only for the type that you select.
Field Update Edit
Save
Save & New
Cancel
Identification
= Required Information
Name
Verified Property
Unique Name
Verified_Property
Description
Object
Property
Field to Update
Property: Verified
Field Data Type
Checkbox
Re-evaluate Workflow Rules after Field Change
Specify New Field Value
Checkbox Options
True
False
Save
Save & New
Cancel

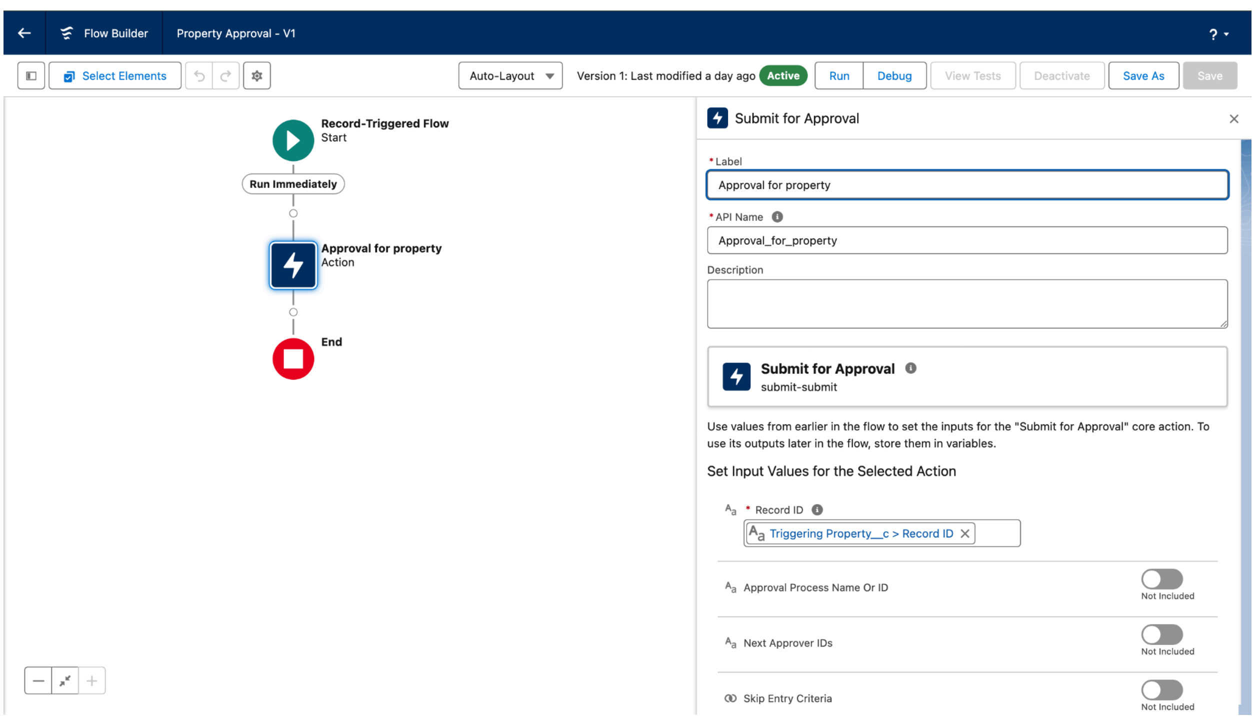

salesforce
'24
Search...
Search
Switch to Lightning Experience
Harshvardhan Sin...
Setup
Help
Content
Home
Chatter
Libraries
Content
Subscriptions
Quick Find / Search...
Expand All | Collapse All
Lightning Experience Transition Assistant
Move to the new, more productive Salesforce.
Get Started
Salesforce Mobile Quick Start
Home
Administer
Release Updates
Manage Users
Manage Apps
Manage Territories
Company Profile
Data Classification
Privacy Center
Security Controls
Domain Management
Communication Templates
Translation Workbench
Data Management
Mobile Administration
Desktop Administration
Outlook Integration and Sync
Edit Field Update
UnVerified Property
Help for this Page
Define the field update, including the object associated with the workflow rule, approval process, or entitlement process, the field to update, and the value to apply. Note that the field to update may be on a related object. Fields are shown only for the type that you select.
Field Update Edit
Save
Save & New
Cancel
Identification
= Required Information
Name
UnVerified Property
Unique Name
UnVerified_Property
Description
Object
Property
Field to Update
Property: Verified
Field Data Type
Checkbox
Re-evaluate Workflow Rules after Field Change
Specify New Field Value
Checkbox Options
True
False
Save
Save & New
Cancel

Task 10: Create a Record Trigger Flow for Approval

Objective: Automatically submit the approval process when a property record is created.

- **Activity**:
    a. Develop a Record Trigger Flow that triggers when a new property record is created and submits the record for approval.

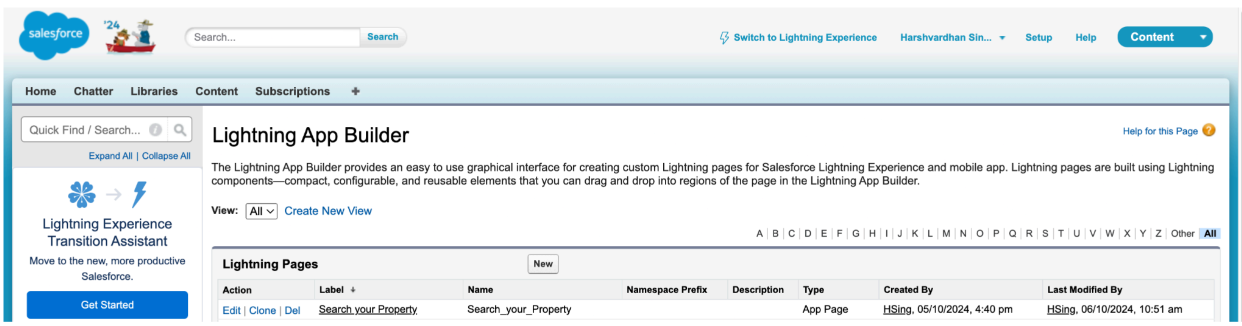


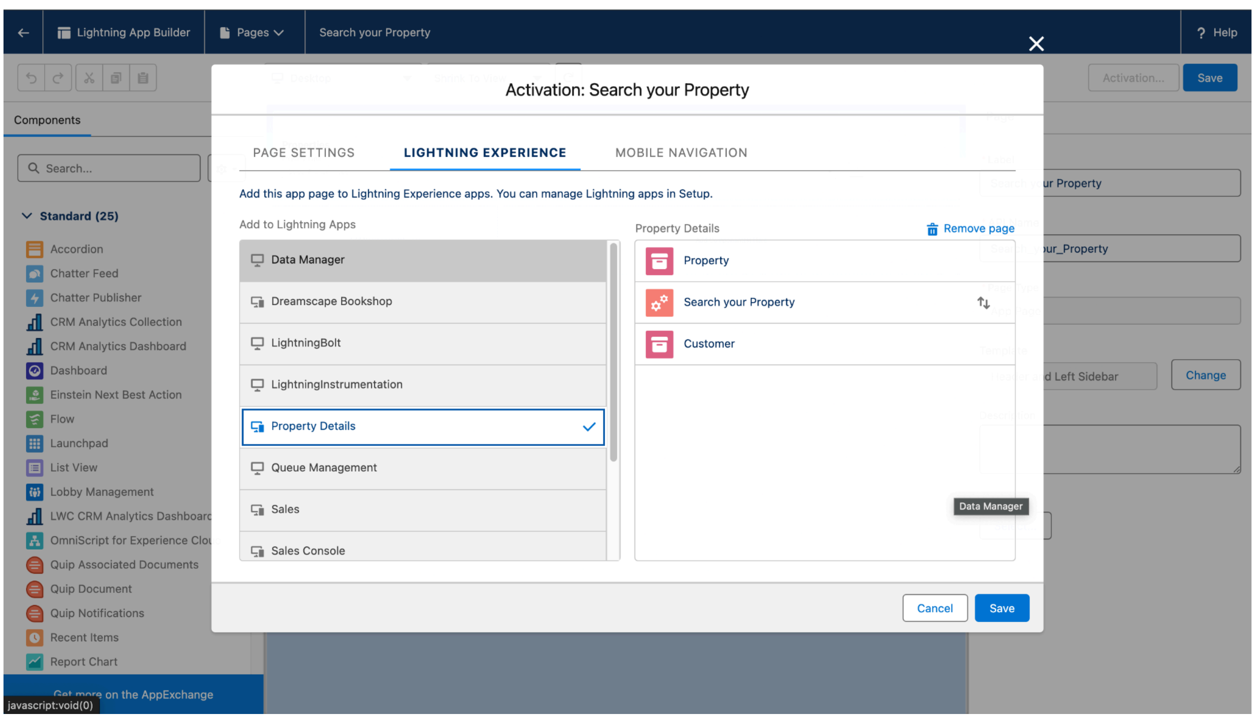


Task 11: Create an App Page

Objective: Build an app page for property search functionality.

- **Activity**:
  a. Create an app page labeled "Search Your Property" and add it to the `Property Details` app.



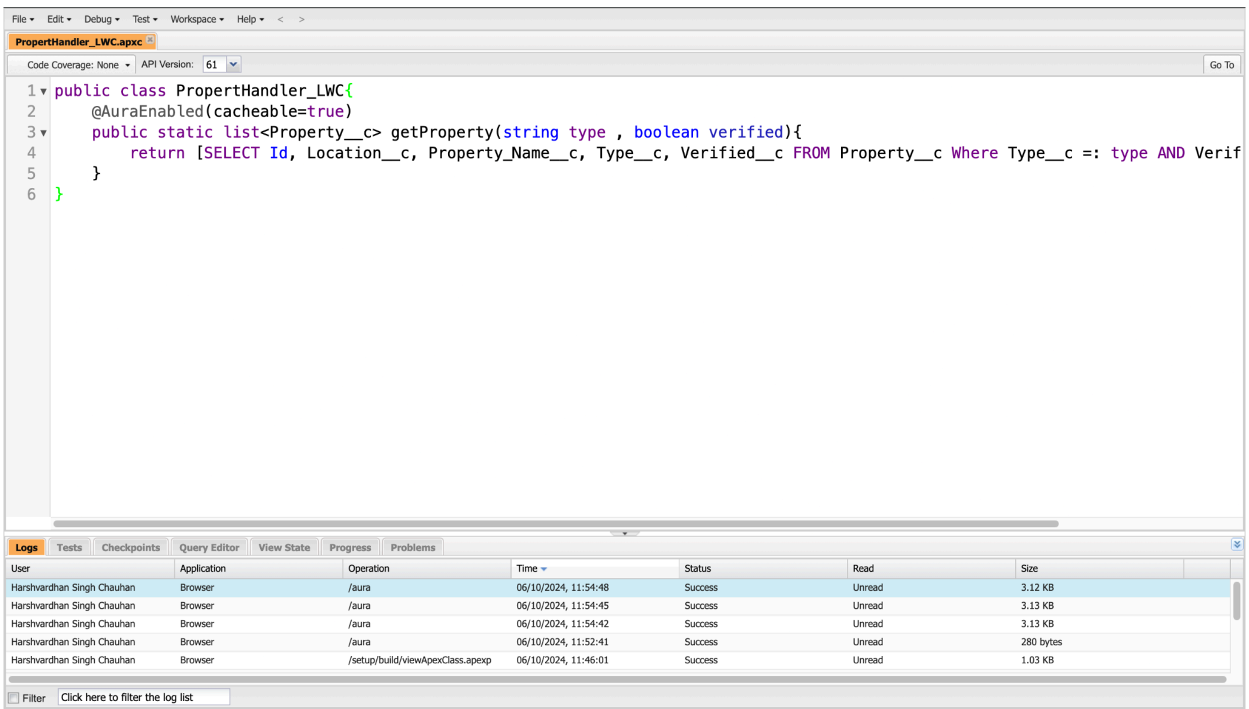


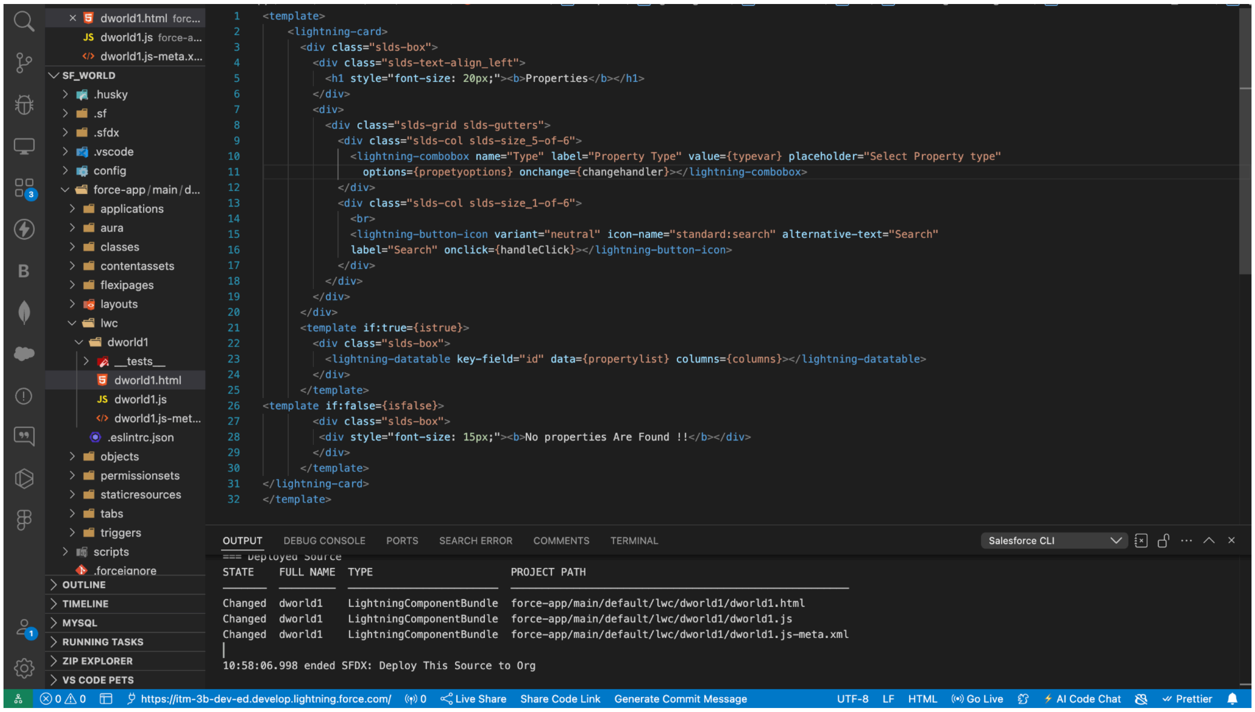
Task 12: Create a Lightning Web Component (LWC)

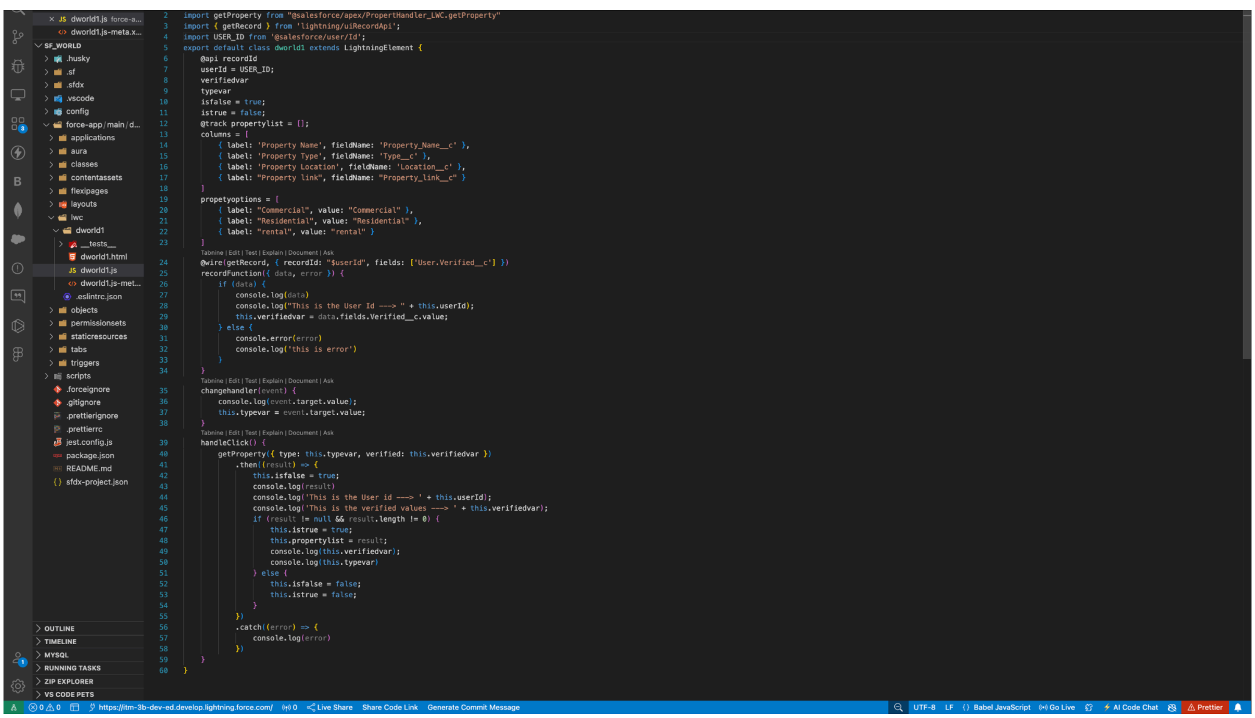
Objective: Develop an LWC component to manage property access for verified and unverified customers.

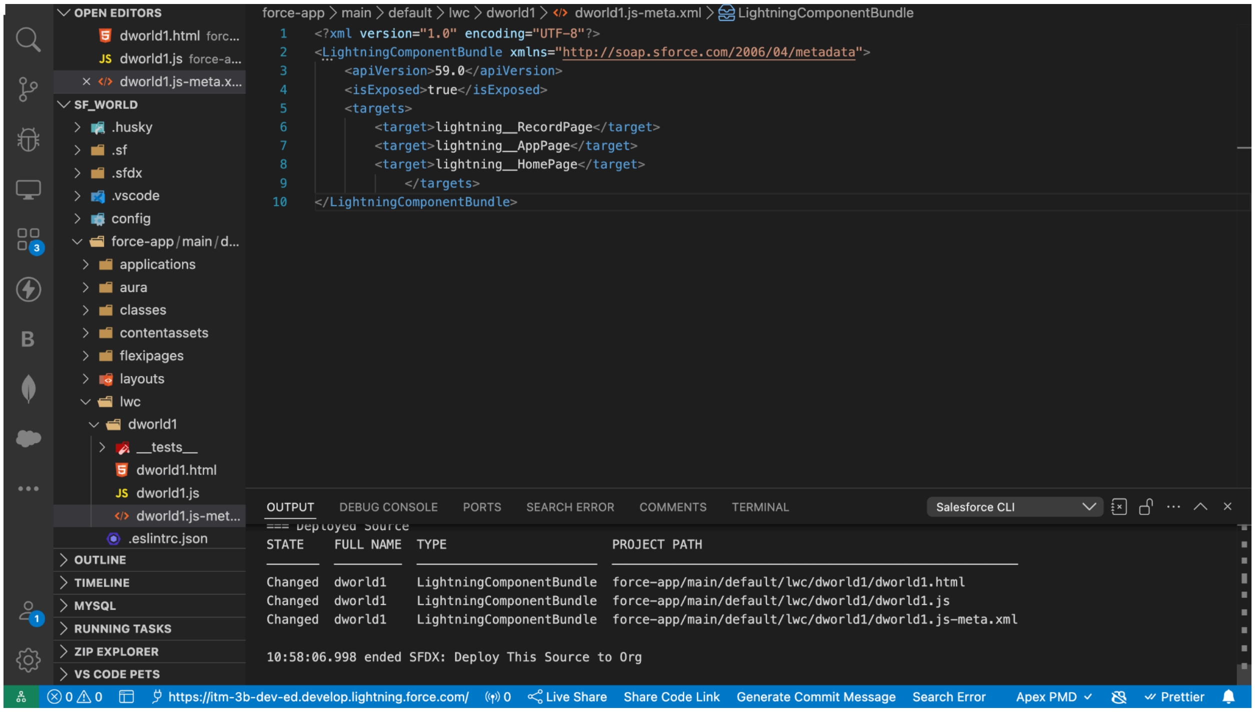
- **Activity**:
    a. Develop the LWC component with Apex logic to handle property visibility based on customer verification status.

- APEX CLASSES




dworld1.html
dworld1.js
dworld1.js-meta.xml
SF_WORLD
```
<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={istrue}>
            <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>
```
OUTPUT
STATE FULL NAME TYPE PROJECT PATH
Changed dworld1 LightningComponentBundle force-app/main/default/lwc/dworld1/dworld1.html
Changed dworld1 LightningComponentBundle force-app/main/default/lwc/dworld1/dworld1.js
Changed dworld1 LightningComponentBundle force-app/main/default/lwc/dworld1/dworld1.js-meta.xml
10:58:06.998 ended SFDX: Deploy This Source to Org
https://itm-3b-dev-ed.develop.lightning.force.com/

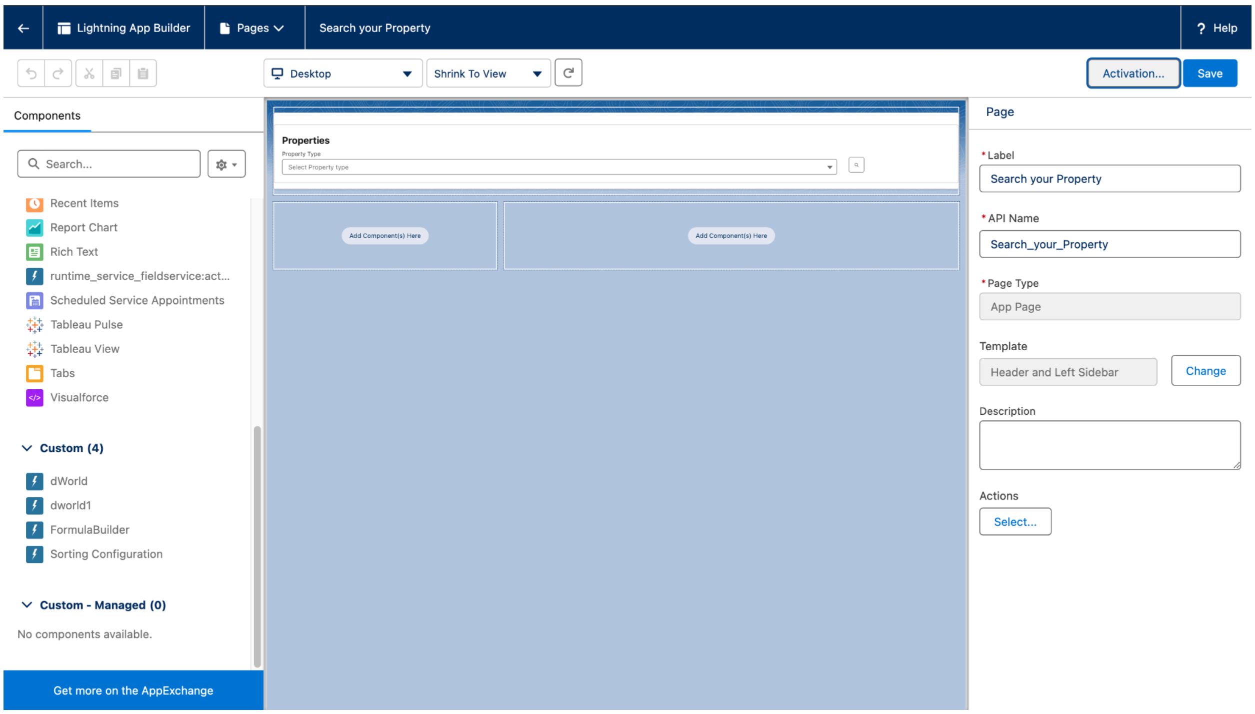

import getProperty from "@salesforce/apex/PropertHandler_LWC.getProperty"
import { getRecord } from 'lightning/uiRecordApi';
import USER_ID from '@salesforce/user/Id';
export default class dworld1 extends LightningElement {
    @api recordId
    userId = USER_ID;
    verifiedvar
    typevar
    isfalse = 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)
            })
    }
}

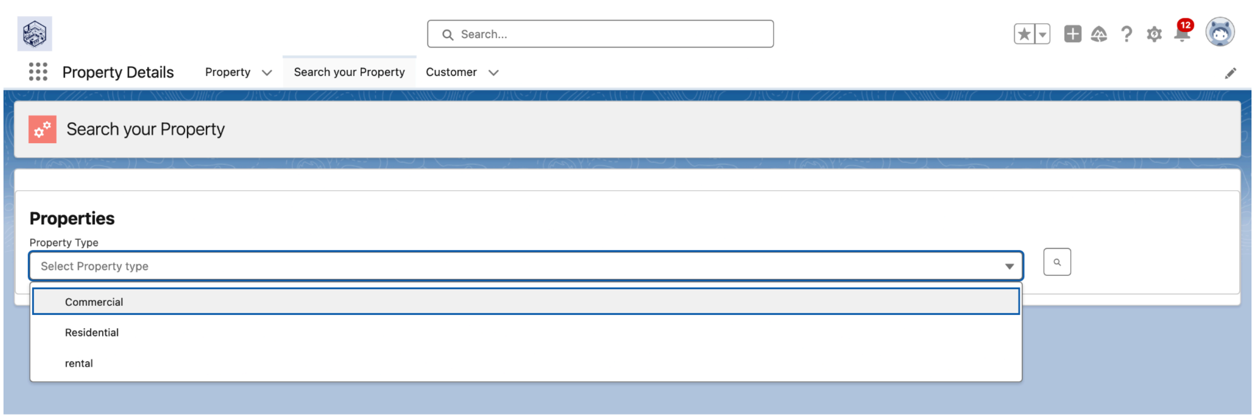

OPEN EDITORS
dworld1.html
dworld1.js
dworld1.js-meta.x...
SF_WORLD
.husky
.sf
.sfdx
.vscode
config
force-app/main/d...
applications
aura
classes
contentassets
flexipages
layouts
lwc
dworld1
__tests__
dworld1.html
dworld1.js
dworld1.js-met...
.eslintrc.json
OUTLINE
TIMELINE
MYSQL
RUNNING TASKS
ZIP EXPLORER
VS CODE PETS
force-app > main > default > lwc > dworld1 > dworld1.js-meta.xml > LightningComponentBundle
<?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>
OUTPUT DEBUG CONSOLE PORTS SEARCH ERROR COMMENTS TERMINAL
Salesforce CLI
STATE FULL NAME TYPE PROJECT PATH
Changed dworld1 LightningComponentBundle force-app/main/default/lwc/dworld1/dworld1.html
Changed dworld1 LightningComponentBundle force-app/main/default/lwc/dworld1/dworld1.js
Changed dworld1 LightningComponentBundle force-app/main/default/lwc/dworld1/dworld1.js-meta.xml
10:58:06.998 ended SFDX: Deploy This Source to Org
https://itm-3b-dev-ed.develop.lightning.force.com/
Live Share
Share Code Link
Generate Commit Message
Search Error
Apex PMD
Prettier

Task 13: Deploy the LWC Component to the App Page

Objective: Add the LWC component to the app page.

- **Activity**:
  a. Drag and drop the LWC component onto the "Search Your Property" app page and save.





Task 14: Assign Apex Class Access to Profiles

Objective: Set appropriate security access to Apex classes for necessary profiles.

- **Activity**:
  a. Assign access to the Apex class `PropertyHandler_LWC` to both Manager and Customer profiles.