Salesforce Virtual Internship Program

SmartInternz

A CRM APPLICATION TO HANDLE THE CLIENTS AND THEIR PROPERTY RELATED REQUIREMENTS

NAME: **SAI JOSHITHA PALAVALASA**

E-MAIL: **322103382046@gvpce.ac.in**

COLLEGE NAME: **GAYATRI VIDYA PARISHAD COLLEGE OF ENGINEERING, AUTONOMOUS**

PROJECT TITLE: *A CRM APPLICATION TO HANDLE THE CLIENTS AND THEIR PROPERTY RELATED REQUIREMENTS*

1.Project Overview

This project focuses on the development of a **CRM Application for managing clients and their property-related requirements** using Salesforce. The application aims to streamline and optimize client interactions, property tracking, and operational workflows. Designed to meet the specific needs of real estate businesses, this solution automates key processes such as capturing customer preferences, managing property listings, and providing tailored recommendations.

The primary challenge addressed by this project is the manual handling of client and property data, which often leads to inefficiencies and inaccuracies. By leveraging Salesforce's robust CRM platform, the project delivers a comprehensive, user-friendly solution that ensures efficient resource management, enhanced customer relationships, and seamless reporting.

Through this project, we will be able to achieve:

- **Operational Excellence:** Automating routine processes to save time and reduce errors.
- **Data-Driven Decision-Making:** Providing owners with detailed, real-time insights into client preferences, property status, and transaction trends.
- **Scalability and Efficiency:** Supporting long-term growth with a flexible, secure, and scalable solution.

2. Objectives

Business Goals:

1. **Streamlined Operations:** Automate client and property data management.
2. **Enhanced Customer Relationships:** Provide personalized insights and service recommendations.
3. **Data Security:** Implement robust access controls.

Specific Outcomes:

- Centralized platform for managing client and property data.
- Real-time dashboards for monitoring key metrics.
- Reduction in manual errors and increased efficiency.

3. Salesforce Features and Concepts Utilized

a) Custom Objects:

- **Customer Object:** Tracks customer details, including name, contact information, and preferences.
- **Property Object:** Maintains data on properties, such as type, location, and status.

b) Workflows and Automations:

- Automated email notifications for key events (e.g., property status updates).
- Triggers for data consistency and real-time updates.

c) Validation Rules:

- Ensure mandatory fields (e.g., client email, property ID) are filled.
- Prevent invalid data entries.

d) Roles and Profiles:

1. **Roles Created:**
    - **Sales Executive Role:** Limited access to assigned properties and customers.
    - **Sales Manager Role:** Access to manage team activities and property approvals.
    - **Customer Role:** View-only access to approved properties.
2. **Profiles Created:**
    - **System Administrator:** Full access to all objects and records.
    - **Manager:** Managerial data access.

- **Customer:** Limited access with "Verified" and "Unverified" distinctions.

e) Record Trigger Flow:

- Developed a flow to automatically submit property approval processes.

f) Approval Process:

- Configured an approval process for the Property object to ensure properties meet business requirements before publication.
- Automated submission of approvals using a record-triggered flow.

g) Lightning Web Components (LWC):

- Created a custom LWC to enhance user experience by providing a dynamic interface for property management.
- Integrated the LWC into the app page for streamlined navigation and interaction.

h) Apex Classes and Triggers

- Developed custom logic for automating processes, such as updating property statuses and submitting approvals.
- Example: An Apex trigger to manage property status changes dynamically.

4)Detailed Steps to Solution Design

Create a JotForm and Integrate with Salesforce:

- Designed a JotForm to collect customer details and preferences.
- Integrated JotForm with Salesforce to automatically create customer records upon form submission.

STEPS FOLLOWED TO CREATE JOTFORM:

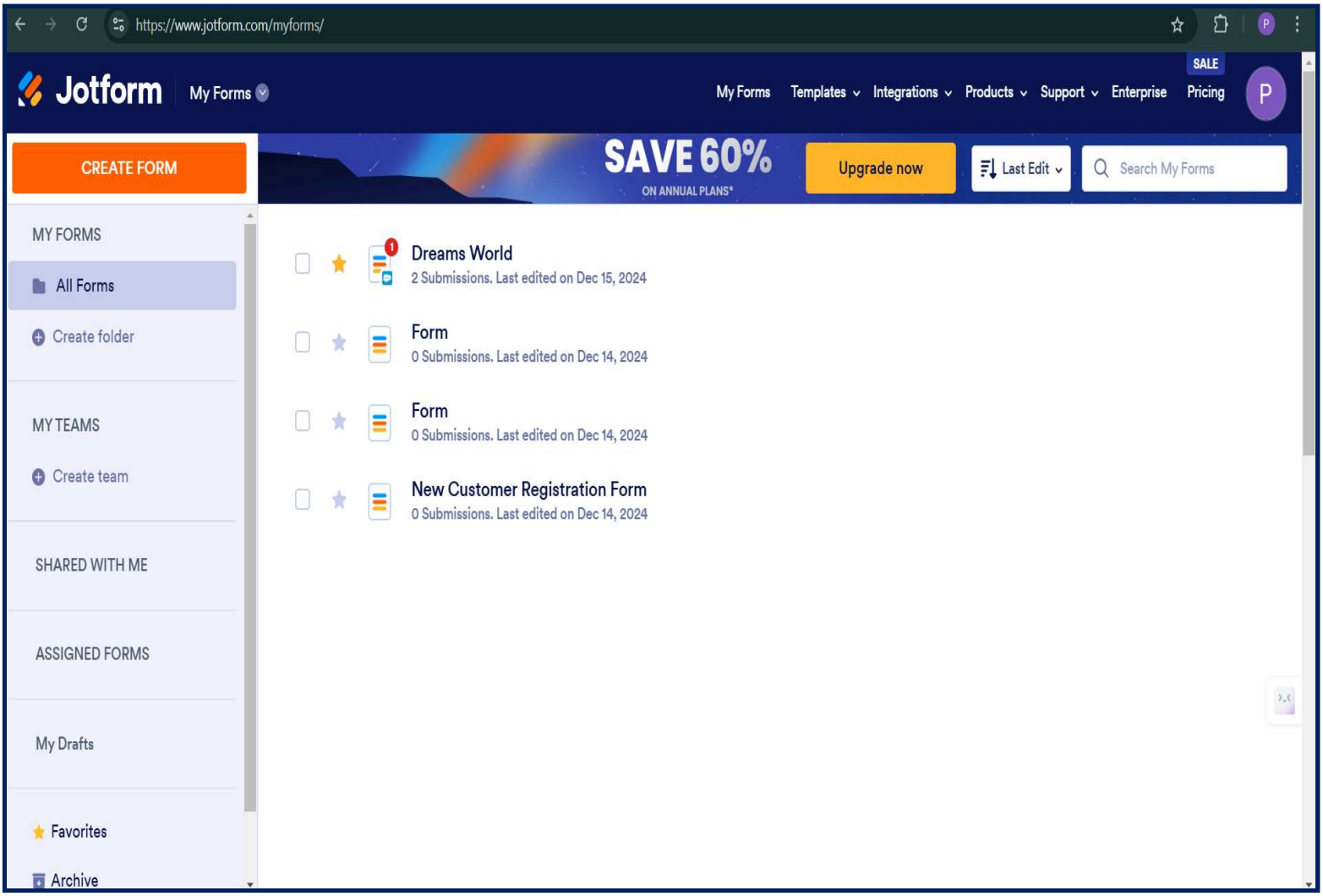
1) Go to any browser and search for jotform. If you donot have an account we need to

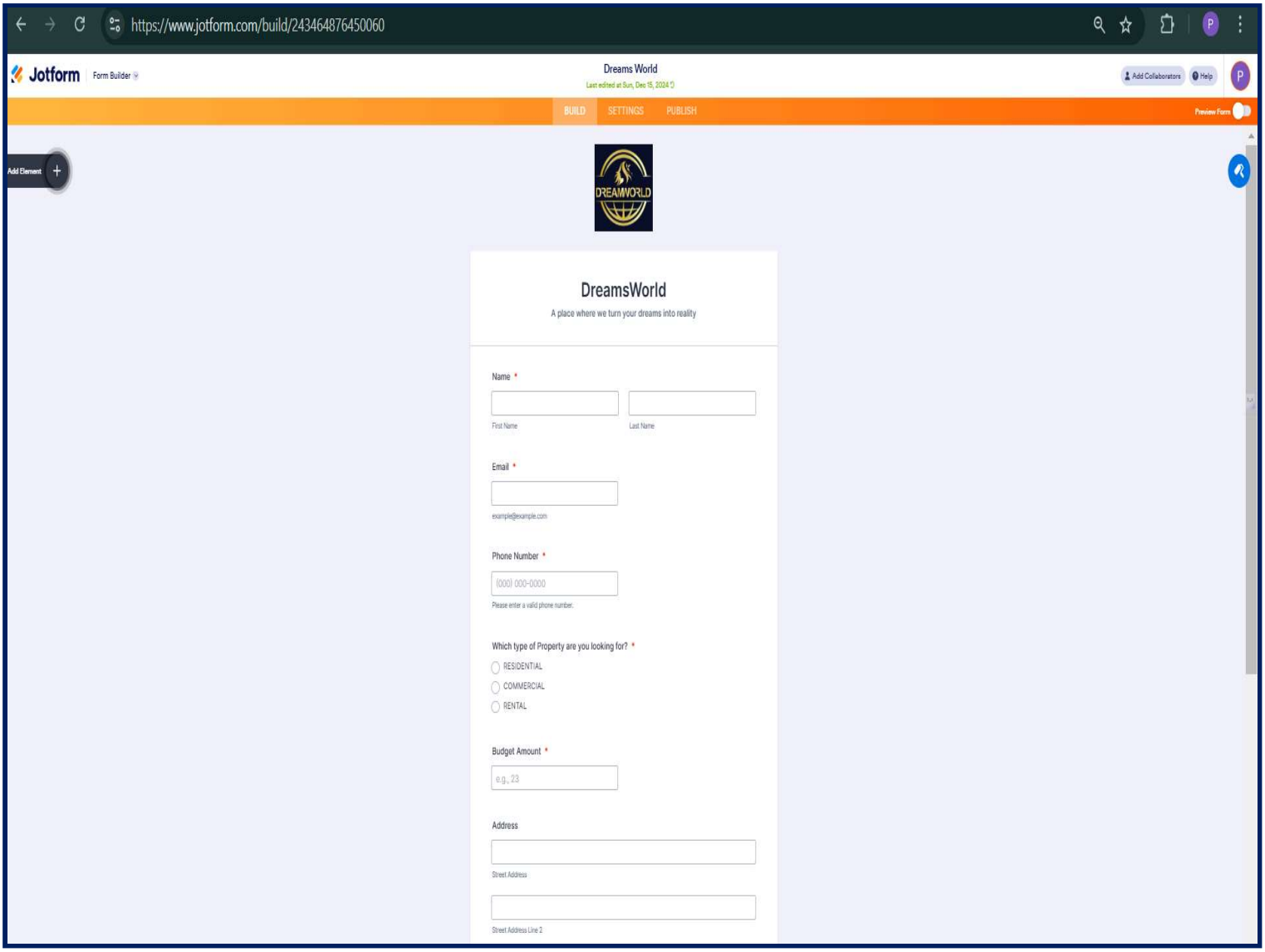
signup. Then, after signing up, log into it.

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.





- **Create Objects from Spreadsheet:**
  - Imported data from a spreadsheet to create custom objects such as Customer and Property.

STEPS FOLLOWED TO CREATE OBJECTS:

1) Go to your object manager in salesforce and click on create object from spreadsheet.

2) Then, we need to click on the link to get the spreadsheet.

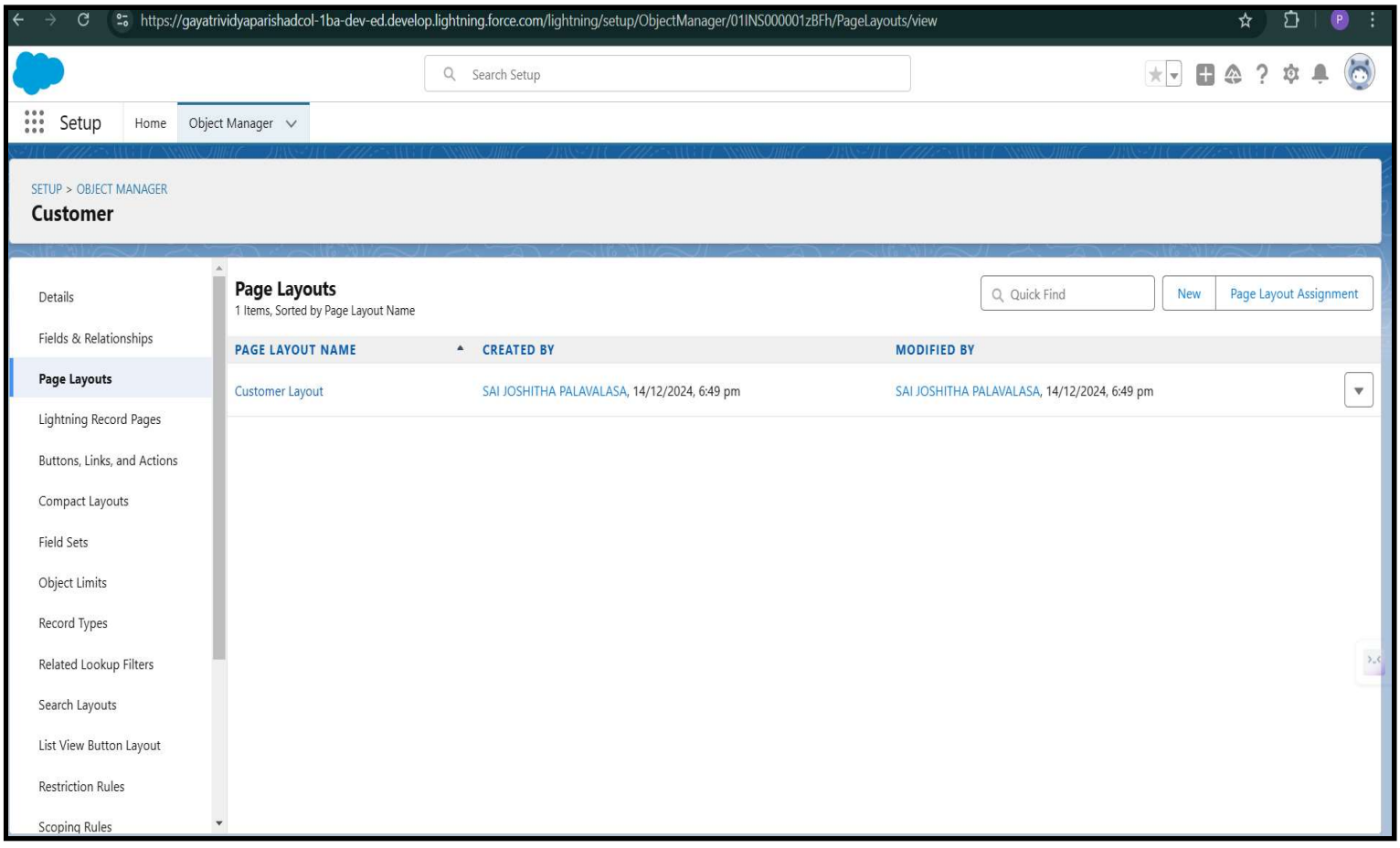
3) For **Customer, click** on the Customer link. Similarly, for **Property ,** we need to click on

Property Link.

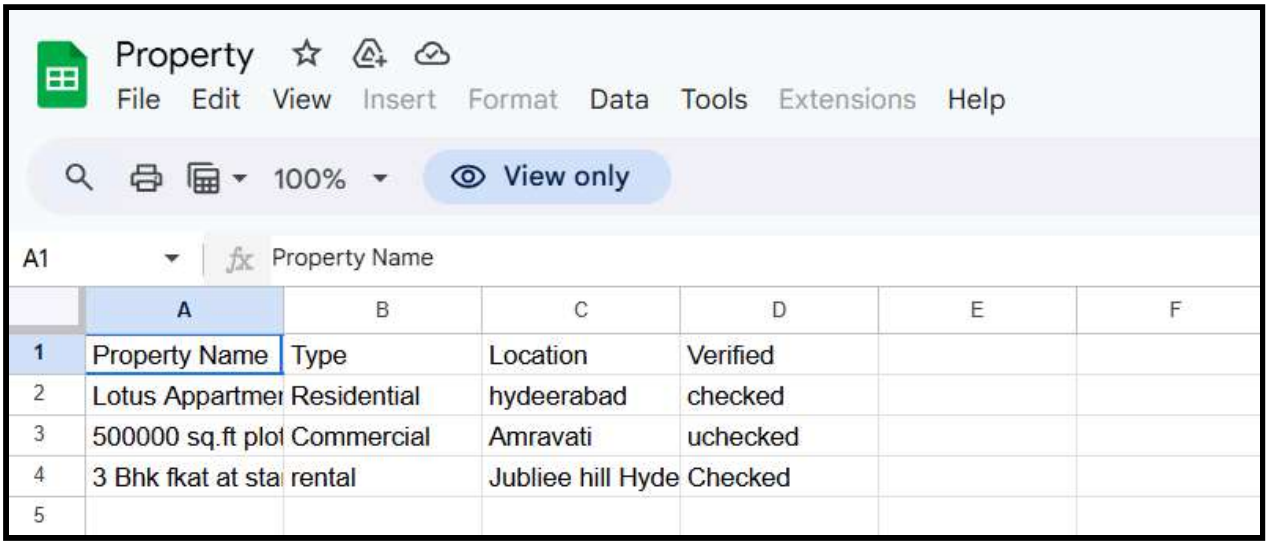
4) After downloading, upload the file, map the fields, and upload to create an object.

CUSTOMER OBJECT CREATION

	A	B	C	D	E	F	G	H	I	J	K
1	Customer	Phone Number	Emial	State	Property Type	Budget Amount	Street Address	Street Address li	City	postal code	Verified
2	Rakesh	788797	rakesh@gmail.c	Telangana	Residential	4000000	gb road	street no 45	Hyderabad	555001	checked
3	prakash	55448855	p@gmail.com	Maharashtra	Commercial	8000000	gachibowli	indira road	mumbai	6600014	unchecked
4	Prajwal	454545	prajwal@gmail.c	Maharashtra	Rental	25000	kamdli	kathora	Amravati	444805	checked
5											



PROPERTY OBJECT CREATION:



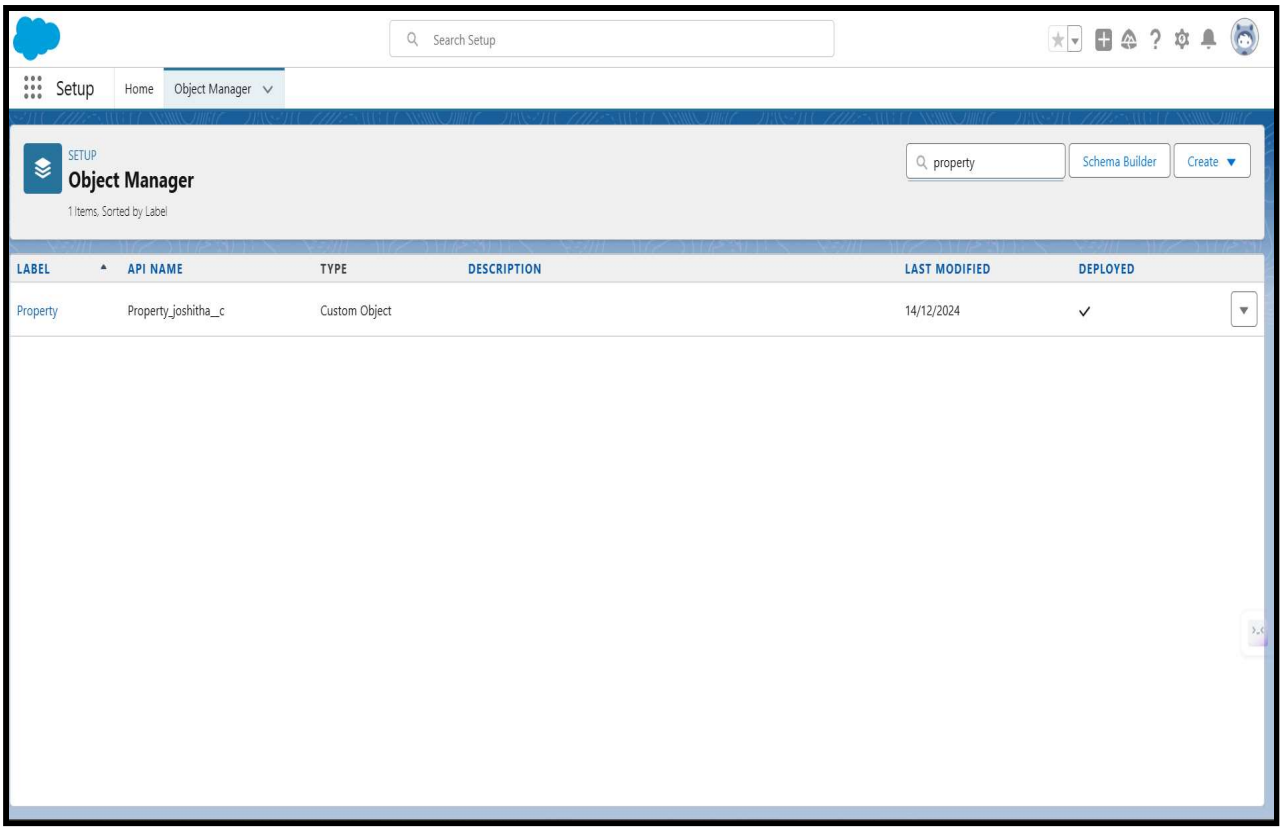
Property

File Edit View Insert Format Data Tools Extensions Help

100% View only

A1 fx Property Name

	A	B	C	D	E	F
1	Property Name	Type	Location	Verified		
2	Lotus Appartmer	Residential	hydeerabad	checked		
3	500000 sq.ft plot	Commercial	Amravati	uchecked		
4	3 Bhk fkat at sta	rental	Jubliee hill Hyde	Checked		
5						



Search Setup

Setup Home Object Manager

SETUP

Object Manager

1 Items, Sorted by Label

property Schema Builder Create

LABEL	API NAME	TYPE	DESCRIPTION	LAST MODIFIED	DEPLOYED
Property	Property_joshitha__c	Custom Object		14/12/2024	✓

- **Integrate JotForm with Salesforce Platform:**
  Enabled seamless data flow between the frontend (JotForm) and backend (Salesforce).

STEPS FOLLOWED TO INTEGRATE:

1) On the Jotform Platform, Click on Integration and choose Salesforce.

2) Click on User Integration and choose “Add to From”.

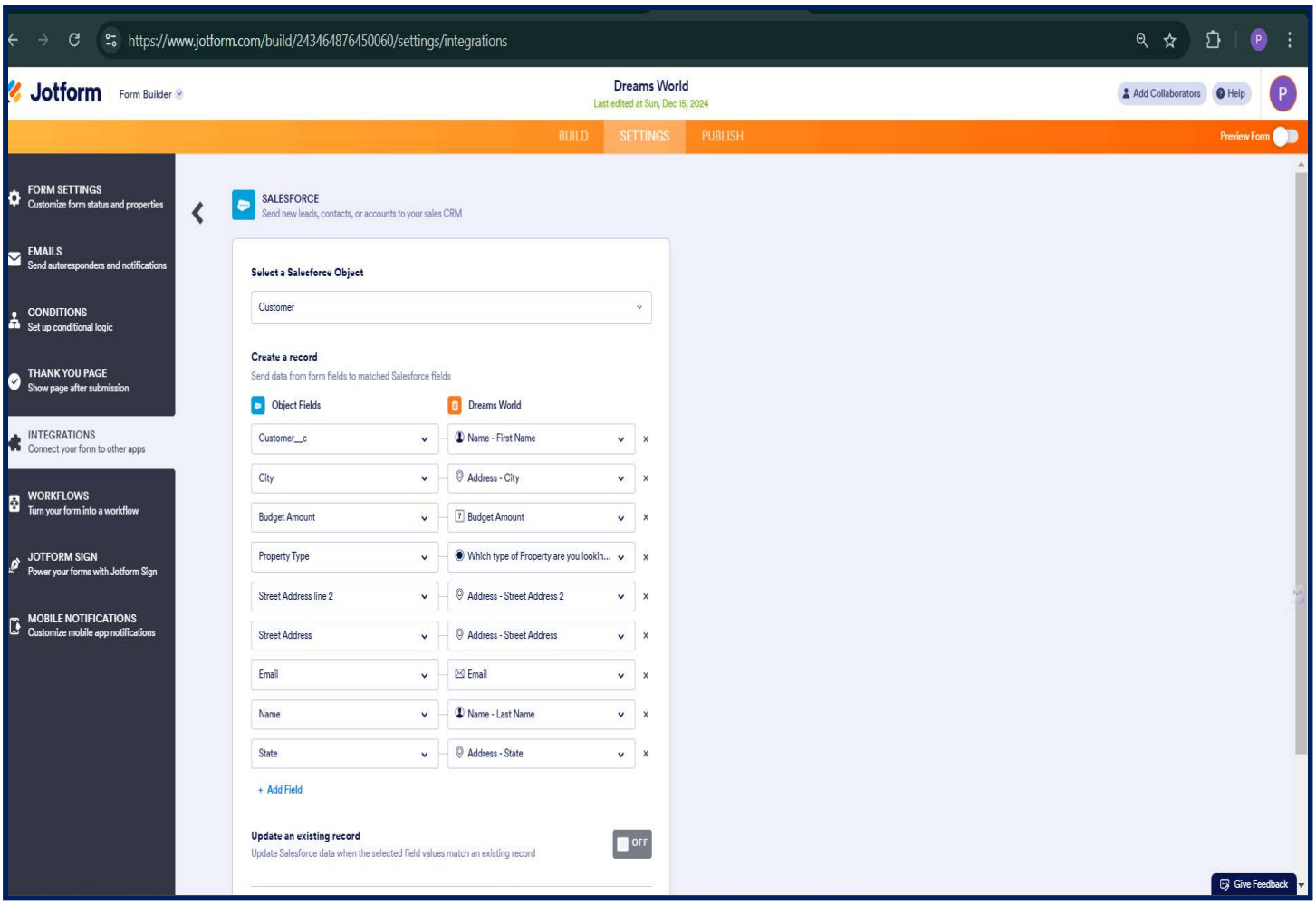
3) Select the Org with which you want to Integrate your jotform with.

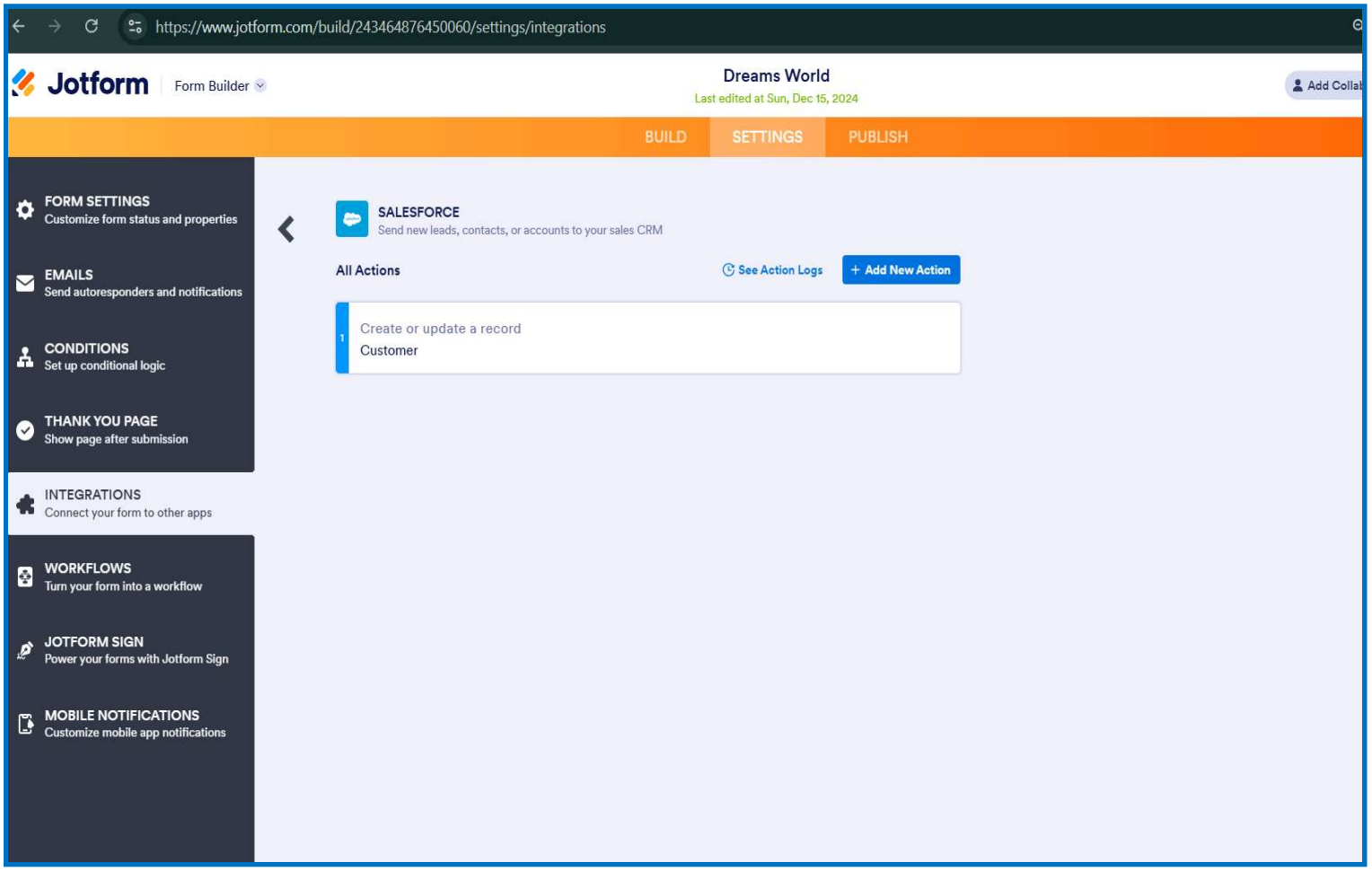
4) Select an Action - Create a record.

Select a Salesforce Object : - Customer

5) Map Each and every field on the Object with the fields on the form and “Save Action”.

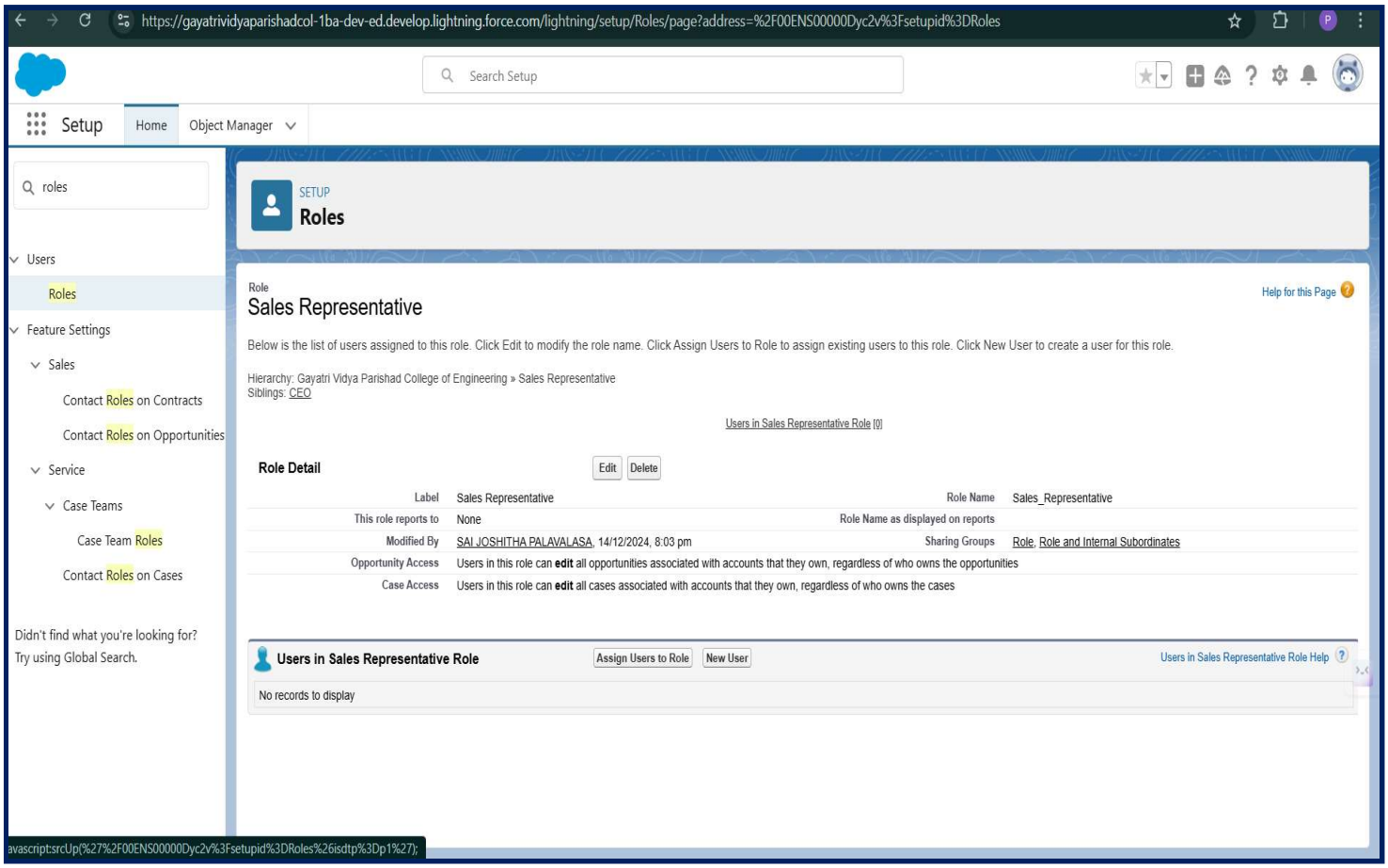
6) Then "Save the Integration and click on Finish.

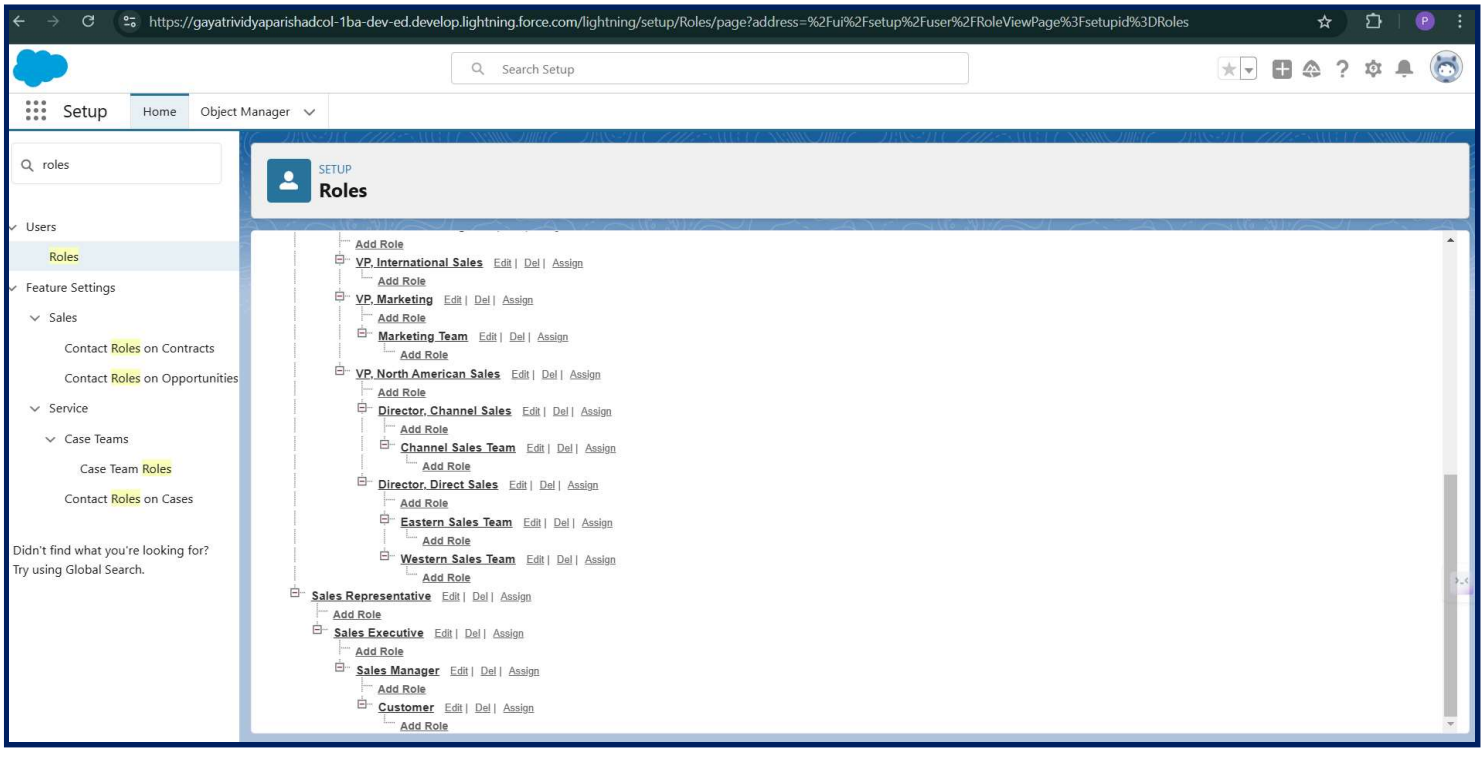




Roles and Hierarchy

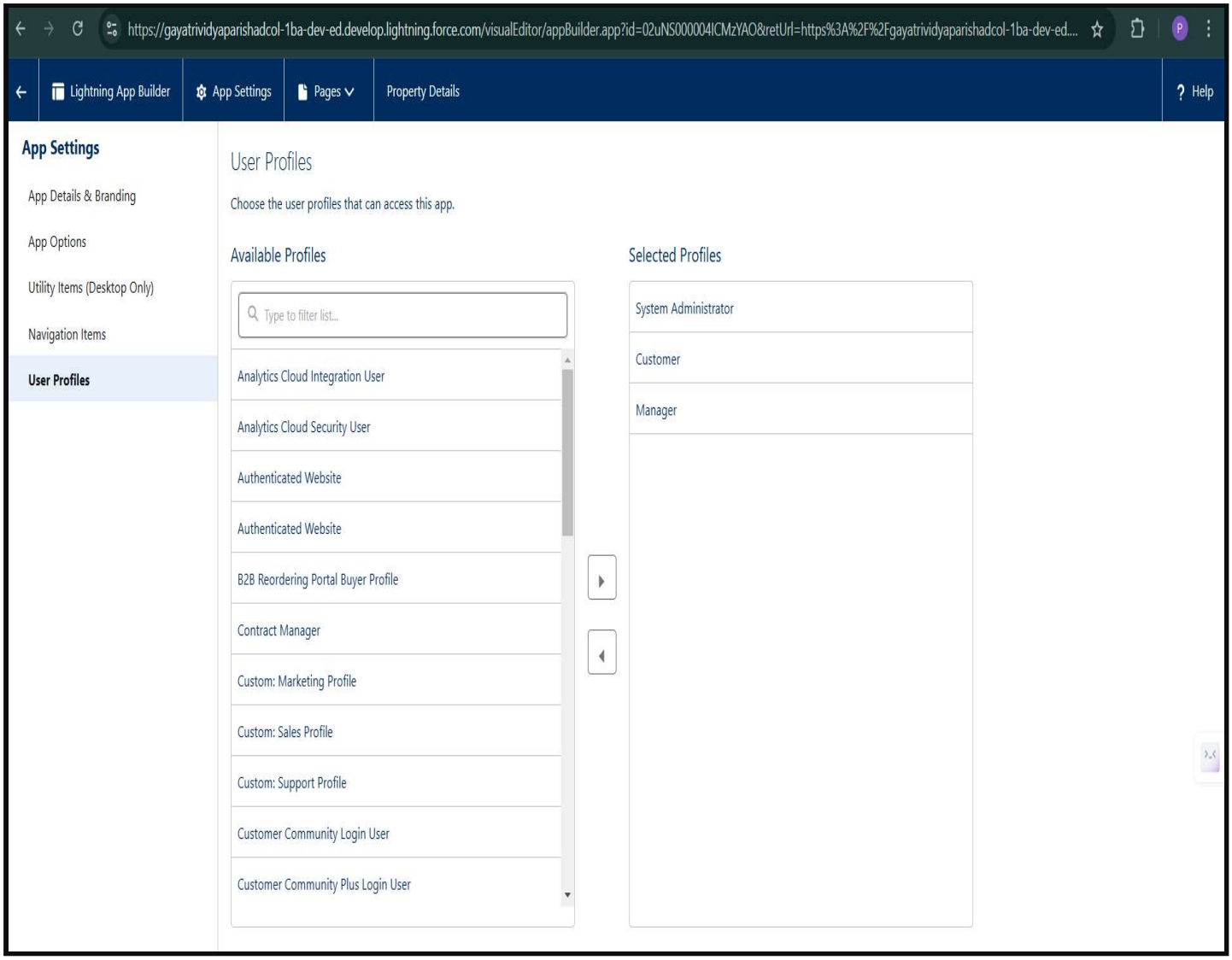
- Created roles and defined a hierarchy for streamlined user access and management:
    1. **Sales Representative:** Added a new role just below this level.
    2. **Steps:**
        i. Navigated to **Setup** > **Roles**.
        ii. Clicked on **Expand All** to view the hierarchy.
        iii. Added a new role under the **Sales Representative** to define reporting relationships.
        iv. Assigned the role to users based on their organizational positions.


https://gayatrividyaparishadcol-1ba-dev-ed.develop.lightning.force.com/lightning/setup/Roles/page?address=%2F00ENS00000Dyc2v%3Fsetupid%3DRoles
Search Setup
Setup
Home
Object Manager
roles
Users
Roles
Feature Settings
Sales
Contact Roles on Contracts
Contact Roles on Opportunities
Service
Case Teams
Case Team Roles
Contact Roles on Cases
Didn't find what you're looking for?
Try using Global Search.
SETUP
Roles
Role
Sales Representative
Help for this Page
Below is the list of users assigned to this role. Click Edit to modify the role name. Click Assign Users to Role to assign existing users to this role. Click New User to create a user for this role.
Hierarchy: Gayatri Vidya Parishad College of Engineering » Sales Representative
Siblings: CEO
Users in Sales Representative Role [0]
Role Detail
Edit
Delete
Label
Sales Representative
Role Name
Sales_Representative
This role reports to
None
Role Name as displayed on reports
Modified By
SAI JOSHITHA PALAVALASA, 14/12/2024, 8:03 pm
Sharing Groups
Role, Role and Internal Subordinates
Opportunity Access
Users in this role can edit all opportunities associated with accounts that they own, regardless of who owns the opportunities
Case Access
Users in this role can edit all cases associated with accounts that they own, regardless of who owns the cases
Users in Sales Representative Role
Assign Users to Role
New User
Users in Sales Representative Role Help
No records to display
avascript:srcUp(%27%2F00ENS00000Dyc2v%3Fsetupid%3DRoles%26isdtp%3Dp1%27);


https://gayatrividyaparishadcol-1ba-dev-ed.develop.lightning.force.com/lightning/setup/Roles/page?address=%2Fui%2Fsetup%2Fuser%2FRoleViewPage%3Fsetupid%3DRoles
Search Setup
Setup
Home
Object Manager
roles
Users
Roles
Feature Settings
Sales
Contact Roles on Contracts
Contact Roles on Opportunities
Service
Case Teams
Case Team Roles
Contact Roles on Cases
Didn't find what you're looking for?
Try using Global Search.
SETUP
Roles
Add Role
VP, International Sales Edit | Del | Assign
Add Role
VP, Marketing Edit | Del | Assign
Add Role
Marketing Team Edit | Del | Assign
Add Role
VP, North American Sales Edit | Del | Assign
Add Role
Director, Channel Sales Edit | Del | Assign
Add Role
Channel Sales Team Edit | Del | Assign
Add Role
Director, Direct Sales Edit | Del | Assign
Add Role
Eastern Sales Team Edit | Del | Assign
Add Role
Western Sales Team Edit | Del | Assign
Add Role
Sales Representative Edit | Del | Assign
Add Role
Sales Executive Edit | Del | Assign
Add Role
Sales Manager Edit | Del | Assign
Add Role
Customer Edit | Del | Assign
Add Role

Property Details App

- **Purpose:** A dedicated app for managing property data, enabling seamless data entry, retrieval, and approval processes for sales representatives and managers.
- **Steps to Access:**
  - ➤ Open the **App Launcher** (grid icon in the top-left corner).
  - ➤ Search for and open the **Property Details App**.
  - ➤ Use predefined Lightning Pages to:
    - View and edit property details.
    - Access custom components for data entry (e.g., **Property Form**).
    - Review reports and dashboards showing property trends, approvals, and sales.
  - ➤ Assign access to the app to specific roles like **Sales Executive** or **Manager**.



Purpose of Creating Profiles

Profiles in Salesforce control what users can see and do in the system. By creating custom profiles for "Customer" and "Manager," we define specific permissions and access tailored to their roles:

- **Customer Profile:** Allows customers to view only their property details without access to other data.
- **Manager Profile:** Grants managers higher access to modify property and customer details, ensuring they can oversee and manage the data effectively.

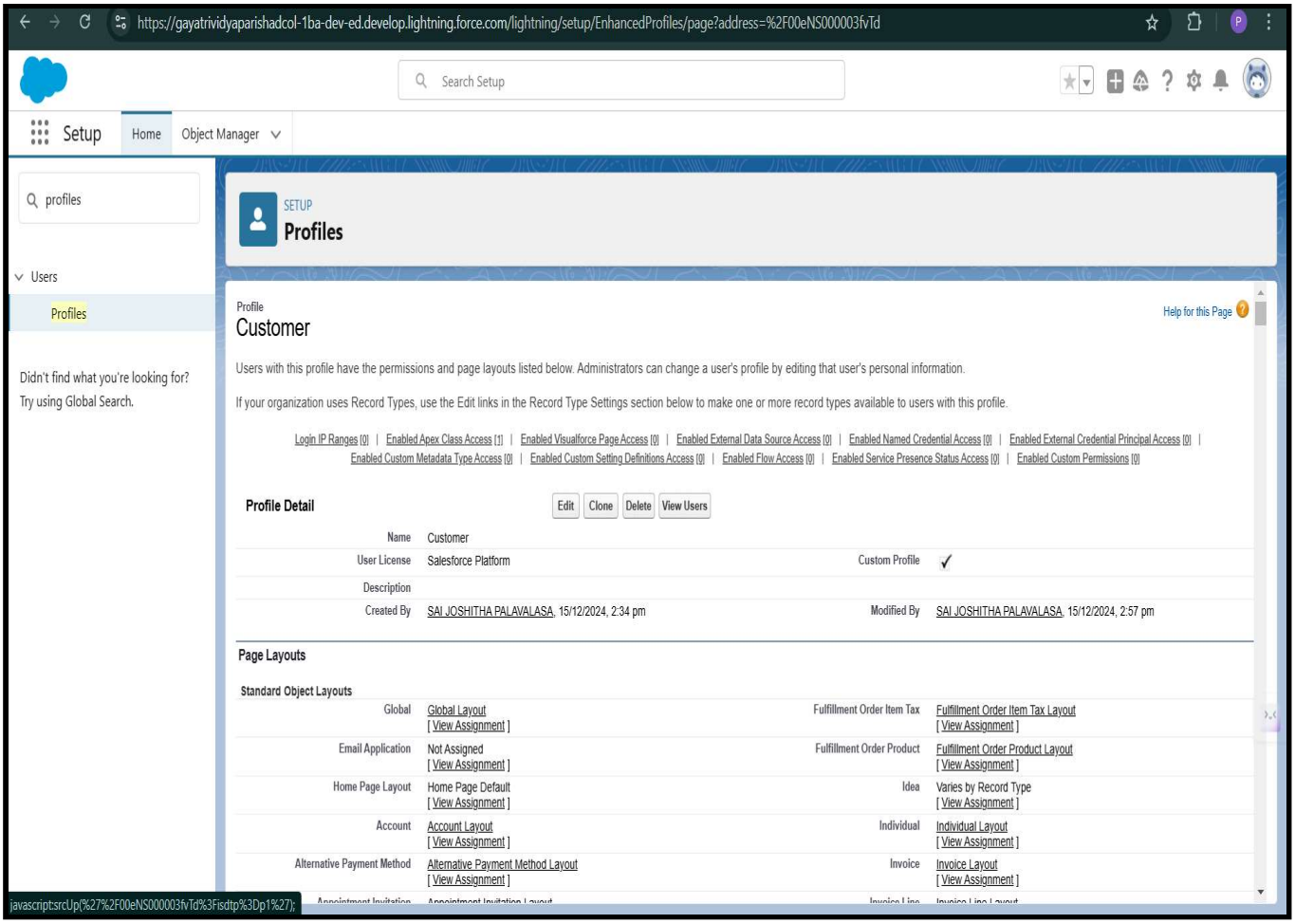
Steps to Create Profiles

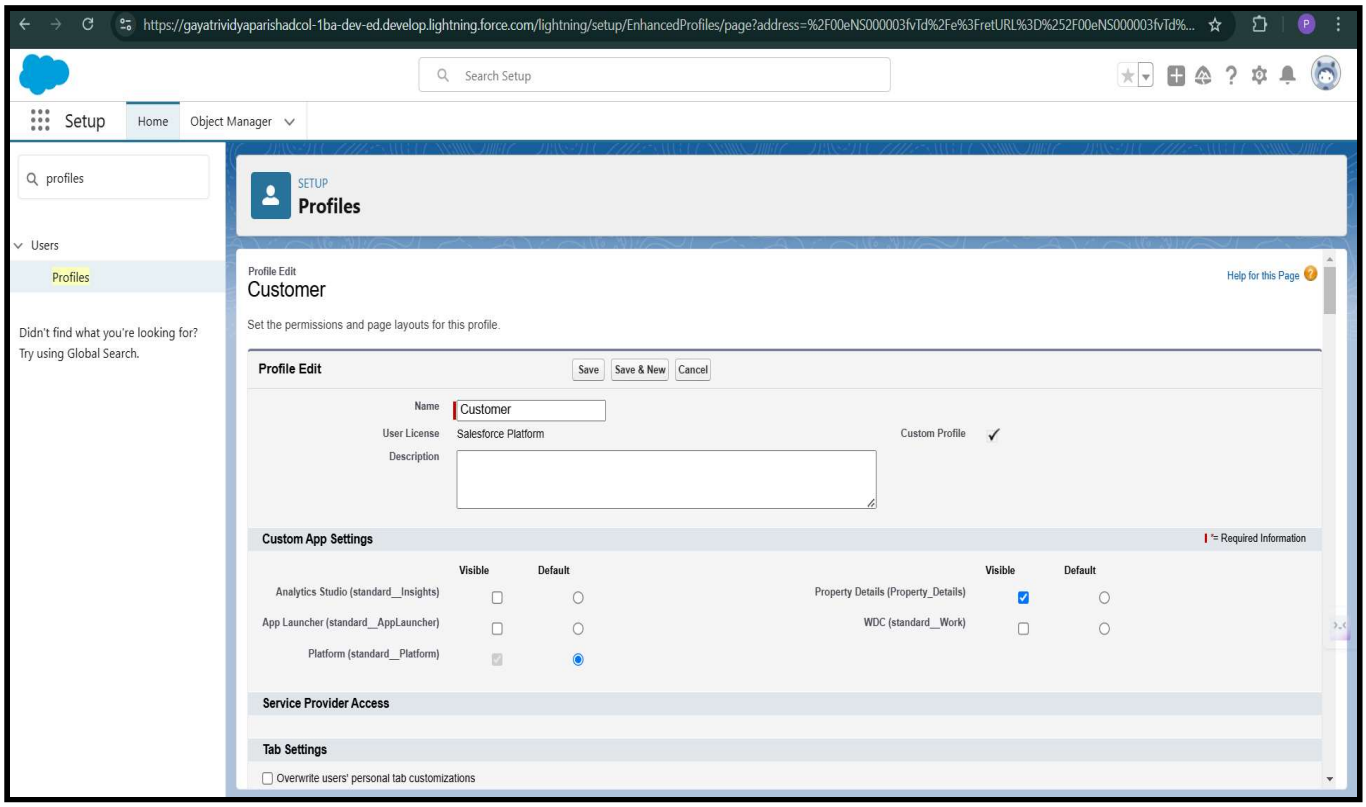
1. Customer Profile

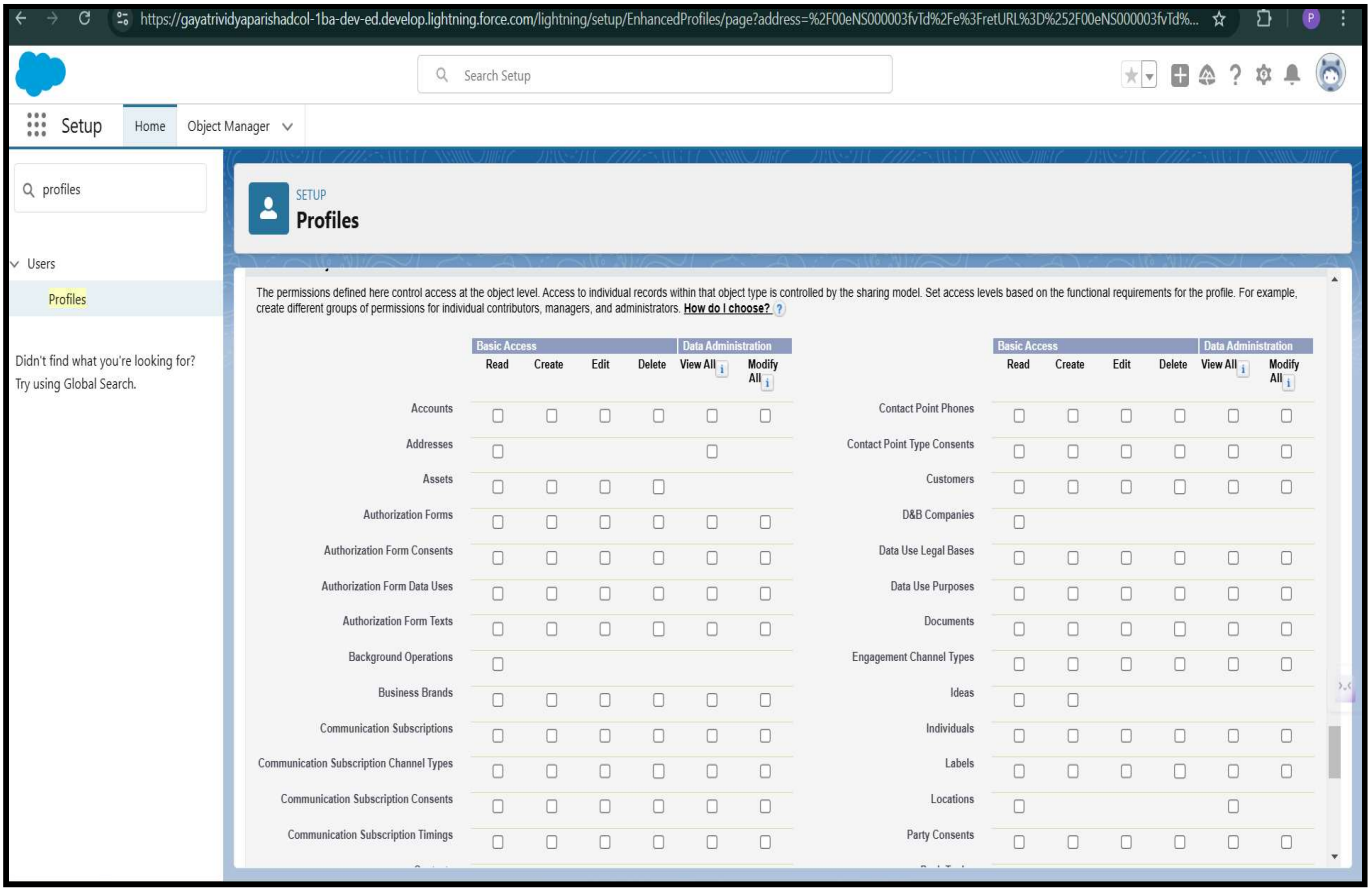
Purpose: To allow users with the "Customer" role to view and access only property details, ensuring restricted access to sensitive data.

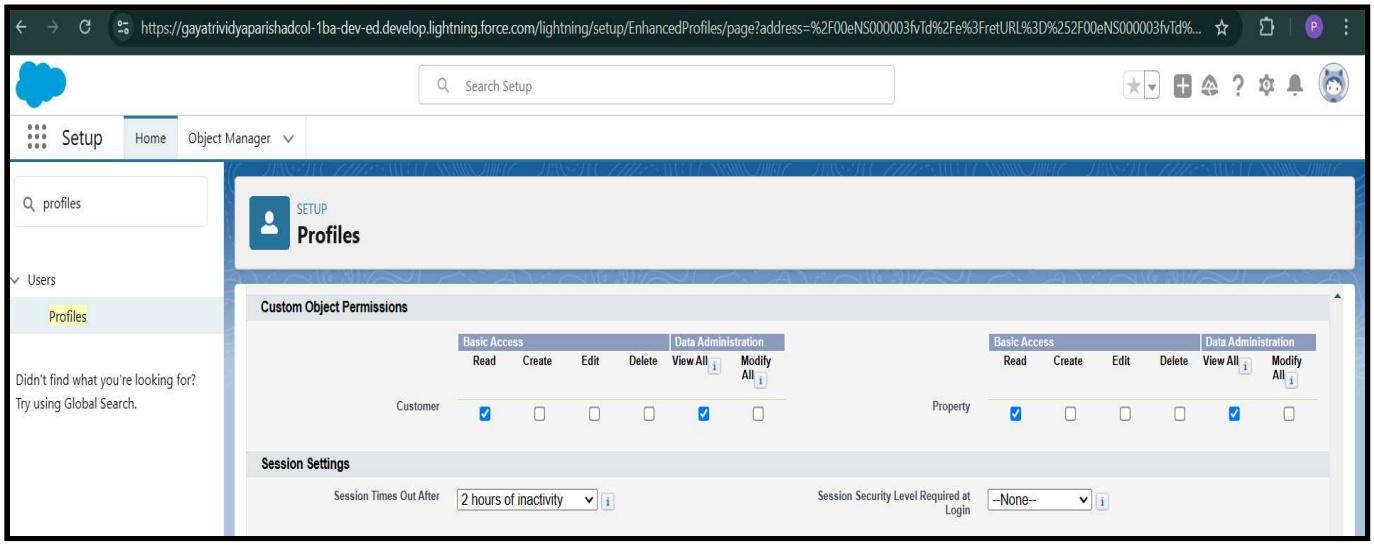
Steps to Create:

1. Navigate to **Setup** in Salesforce.
2. In the **Quick Find box**, search for and select **Profiles**.
3. Locate the **Salesforce Platform User** profile and click **Clone**.
4. Name the new profile **Customer**.
5. **Custom App Settings:**
    - Uncheck all custom objects except **Property Details**.
6. **Standard Object Permissions:**
    - Remove all permissions (Read, Create, Edit, Delete) for standard objects.
7. **Custom Object Permissions:**
    - Uncheck all permissions except:
        - **Read**
        - **View All** for the **Property** object.
8. Save the profile.


https://gayatrividyaparishadcol-1ba-dev-ed.develop.lightning.force.com/lightning/setup/EnhancedProfiles/page?address=%2F00eNS000003fvTd
Search Setup
Setup
Home
Object Manager
profiles
Users
Profiles
Didn't find what you're looking for?
Try using Global Search.
SETUP
Profiles
Profile
Customer
Help for this Page
Users with this profile have the permissions and page layouts listed below. Administrators can change a user's profile by editing that user's personal information.
If your organization uses Record Types, use the Edit links in the Record Type Settings section below to make one or more record types available to users with this profile.
Login IP Ranges [0] | Enabled Apex Class Access [1] | Enabled Visualforce Page Access [0] | Enabled External Data Source Access [0] | Enabled Named Credential Access [0] | Enabled External Credential Principal Access [0] | Enabled Custom Metadata Type Access [0] | Enabled Custom Setting Definitions Access [0] | Enabled Flow Access [0] | Enabled Service Presence Status Access [0] | Enabled Custom Permissions [0]
Profile Detail
Edit
Clone
Delete
View Users
Name
Customer
User License
Salesforce Platform
Custom Profile
Description
Created By
SAI JOSHITHA PALAVALASA, 15/12/2024, 2:34 pm
Modified By
SAI JOSHITHA PALAVALASA, 15/12/2024, 2:57 pm
Page Layouts
Standard Object Layouts
Global
Global Layout
[View Assignment]
Email Application
Not Assigned
[View Assignment]
Home Page Layout
Home Page Default
[View Assignment]
Account
Account Layout
[View Assignment]
Alternative Payment Method
Alternative Payment Method Layout
[View Assignment]
Fulfillment Order Item Tax
Fulfillment Order Item Tax Layout
[View Assignment]
Fulfillment Order Product
Fulfillment Order Product Layout
[View Assignment]
Idea
Varies by Record Type
[View Assignment]
Individual
Individual Layout
[View Assignment]
Invoice
Invoice Layout
[View Assignment]
javascript:srcUp(%27%2F00eNS000003fvTd%3Fisdtp%3Dp1%27);


https://gayatrividyaparishadcol-1ba-dev-ed.develop.lightning.force.com/lightning/setup/EnhancedProfiles/page?address=%2F00eNS000003fvTd%2Fe%3FretURL%3D%252F00eNS000003fvTd%...
Search Setup
Setup
Home
Object Manager
profiles
Users
Profiles
Didn't find what you're looking for?
Try using Global Search.
SETUP
Profiles
Profile Edit
Customer
Help for this Page
Set the permissions and page layouts for this profile.
Profile Edit
Save
Save & New
Cancel
Name
Customer
User License
Salesforce Platform
Custom Profile
Description
Custom App Settings
= Required Information
Visible
Default
Analytics Studio (standard__Insights)
App Launcher (standard__AppLauncher)
Platform (standard__Platform)
Property Details (Property_Details)
WDC (standard__Work)
Service Provider Access
Tab Settings
Overwrite users' personal tab customizations


https://gayatrividyaparishadcol-1ba-dev-ed.develop.lightning.force.com/lightning/setup/EnhancedProfiles/page?address=%2F00eNS000003fvTd%2Fe%3FretURL%3D%252F00eNS000003fvTd%...
Search Setup
Setup
Home
Object Manager
profiles
Users
Profiles
Didn't find what you're looking for?
Try using Global Search.
SETUP
Profiles
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
Addresses
Assets
Authorization Forms
Authorization Form Consents
Authorization Form Data Uses
Authorization Form Texts
Background Operations
Business Brands
Communication Subscriptions
Communication Subscription Channel Types
Communication Subscription Consents
Communication Subscription Timings
Contact Point Phones
Contact Point Type Consents
Customers
D&B Companies
Data Use Legal Bases
Data Use Purposes
Documents
Engagement Channel Types
Ideas
Individuals
Labels
Locations
Party Consents

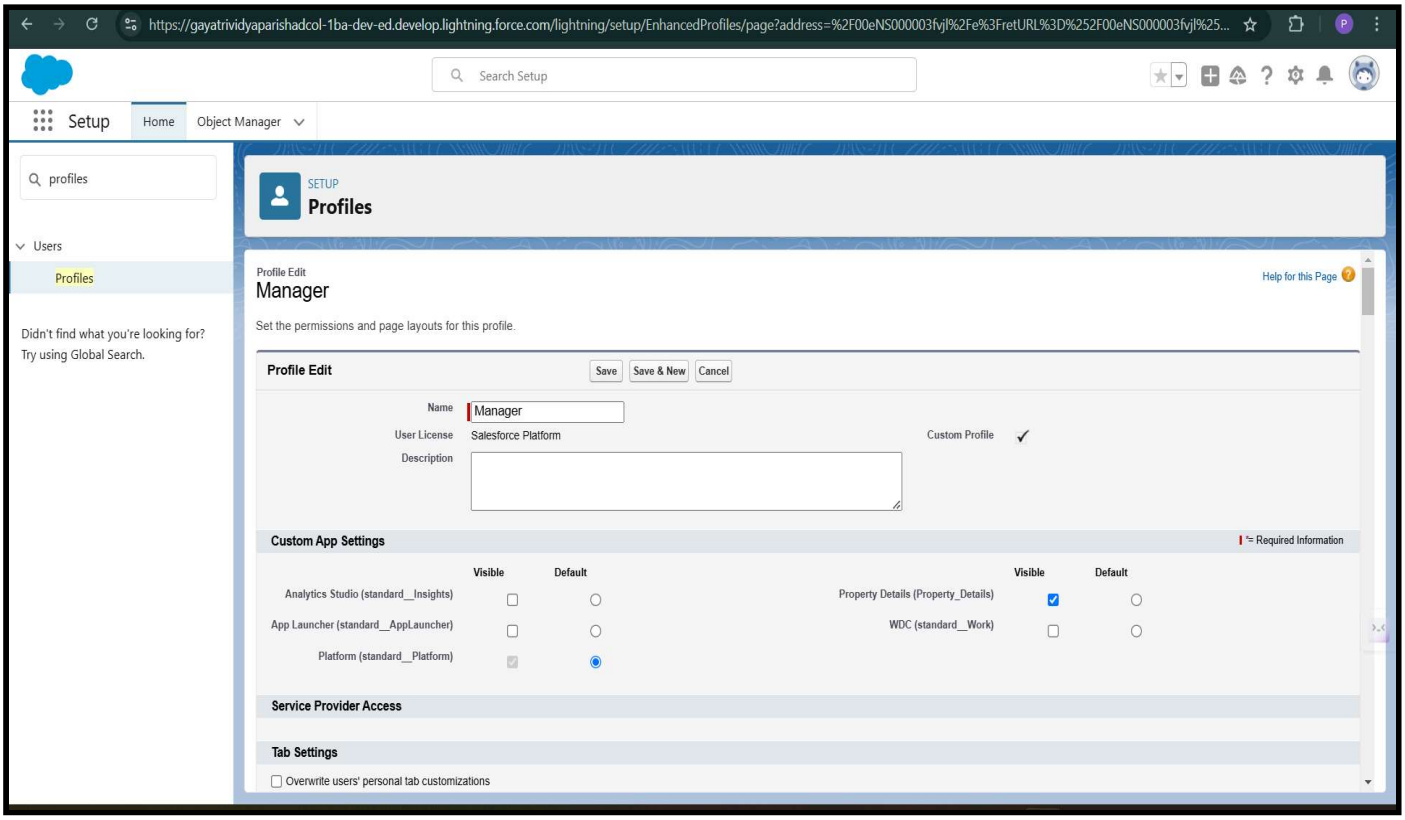

https://gayatrividyaparishadcol-1ba-dev-ed.develop.lightning.force.com/lightning/setup/EnhancedProfiles/page?address=%2F00eNS000003fvTd%2Fe%3FretURL%3D%252F00eNS000003fvTd%...
Search Setup
Setup
Home
Object Manager
profiles
Users
Profiles
Didn't find what you're looking for?
Try using Global Search.
SETUP
Profiles
Custom Object Permissions
Basic Access
Data Administration
Read
Create
Edit
Delete
View All
Modify All
Customer
Property
Session Settings
Session Times Out After
2 hours of inactivity
Session Security Level Required at Login
--None--

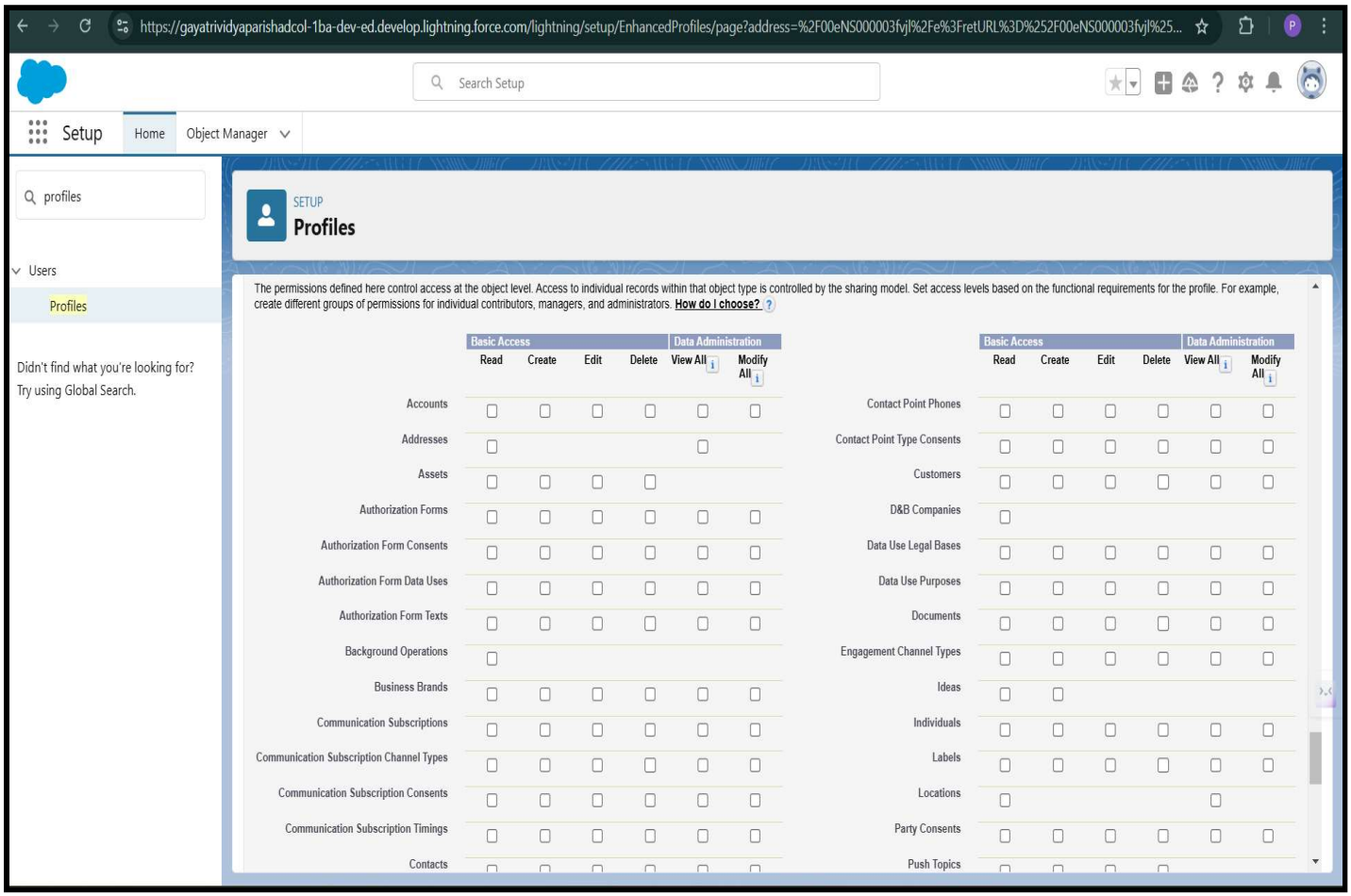
2.Manager Profile

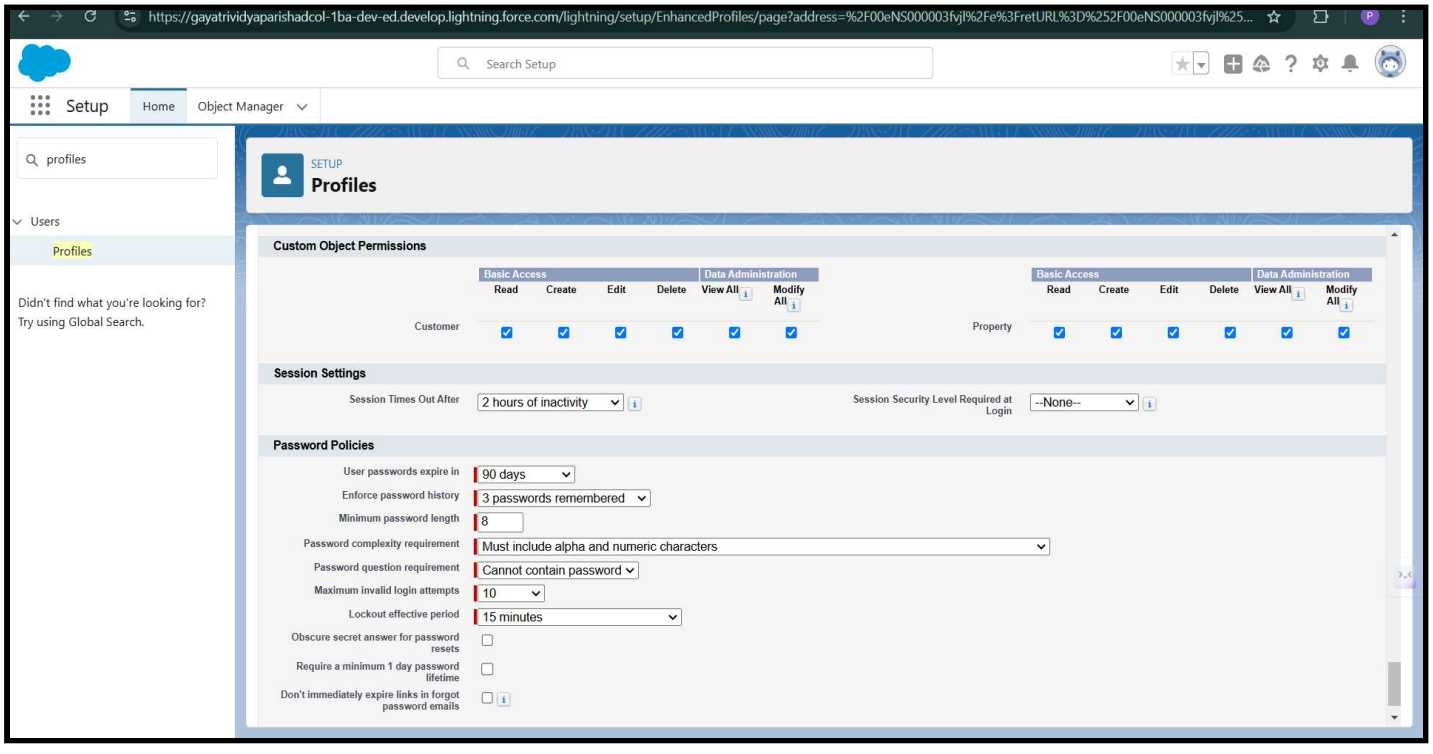
Purpose: To enable users with the "Manager" role to modify both property and customer details, ensuring managerial control over key data.

Steps to Create:

1. Navigate to **Setup** in Salesforce.
2. In the **Quick Find box**, search for and select **Profiles**.
3. Locate the **Salesforce Platform User** profile and click **Clone**.
4. Name the new profile **Manager**.
5. **Custom App Settings:**
   - Uncheck all custom objects except **Property Details**.
6. **Standard Object Permissions:**
   - Remove all permissions (Read, Create, Edit, Delete) for standard objects.
7. **Custom Object Permissions:**
   - Uncheck all permissions except:
     - **Modify All** for the **Property** and **Customer** objects.
8. Save the profile.




https://gayatrividyaparishadcol-1ba-dev-ed.develop.lightning.force.com/lightning/setup/EnhancedProfiles/page?address=%2F00eNS000003fvjl%2Fe%3FretURL%3D%252F00eNS000003fvjl%25...
Search Setup
Setup
Home
Object Manager
profiles
Users
Profiles
Didn't find what you're looking for?
Try using Global Search.
SETUP
Profiles
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
Addresses
Assets
Authorization Forms
Authorization Form Consents
Authorization Form Data Uses
Authorization Form Texts
Background Operations
Business Brands
Communication Subscriptions
Communication Subscription Channel Types
Communication Subscription Consents
Communication Subscription Timings
Contacts
Contact Point Phones
Contact Point Type Consents
Customers
D&B Companies
Data Use Legal Bases
Data Use Purposes
Documents
Engagement Channel Types
Ideas
Individuals
Labels
Locations
Party Consents
Push Topics


https://gayatrividyaparishadcol-1ba-dev-ed.develop.lightning.force.com/lightning/setup/EnhancedProfiles/page?address=%2F00eNS000003fvjl%2Fe%3FretURL%3D%252F00eNS000003fvjl%25...
Search Setup
Setup
Home
Object Manager
profiles
Users
Profiles
Didn't find what you're looking for?
Try using Global Search.
SETUP
Profiles
Custom Object Permissions
Basic Access
Data Administration
Read
Create
Edit
Delete
View All
Modify All
Customer
Property
Session Settings
Session Times Out After
2 hours of inactivity
Session Security Level Required at Login
--None--
Password Policies
User passwords expire in
90 days
Enforce password history
3 passwords remembered
Minimum password length
8
Password complexity requirement
Must include alpha and numeric characters
Password question requirement
Cannot contain password
Maximum invalid login attempts
10
Lockout effective period
15 minutes
Obscure secret answer for password resets
Require a minimum 1 day password lifetime
Don't immediately expire links in forgot password emails

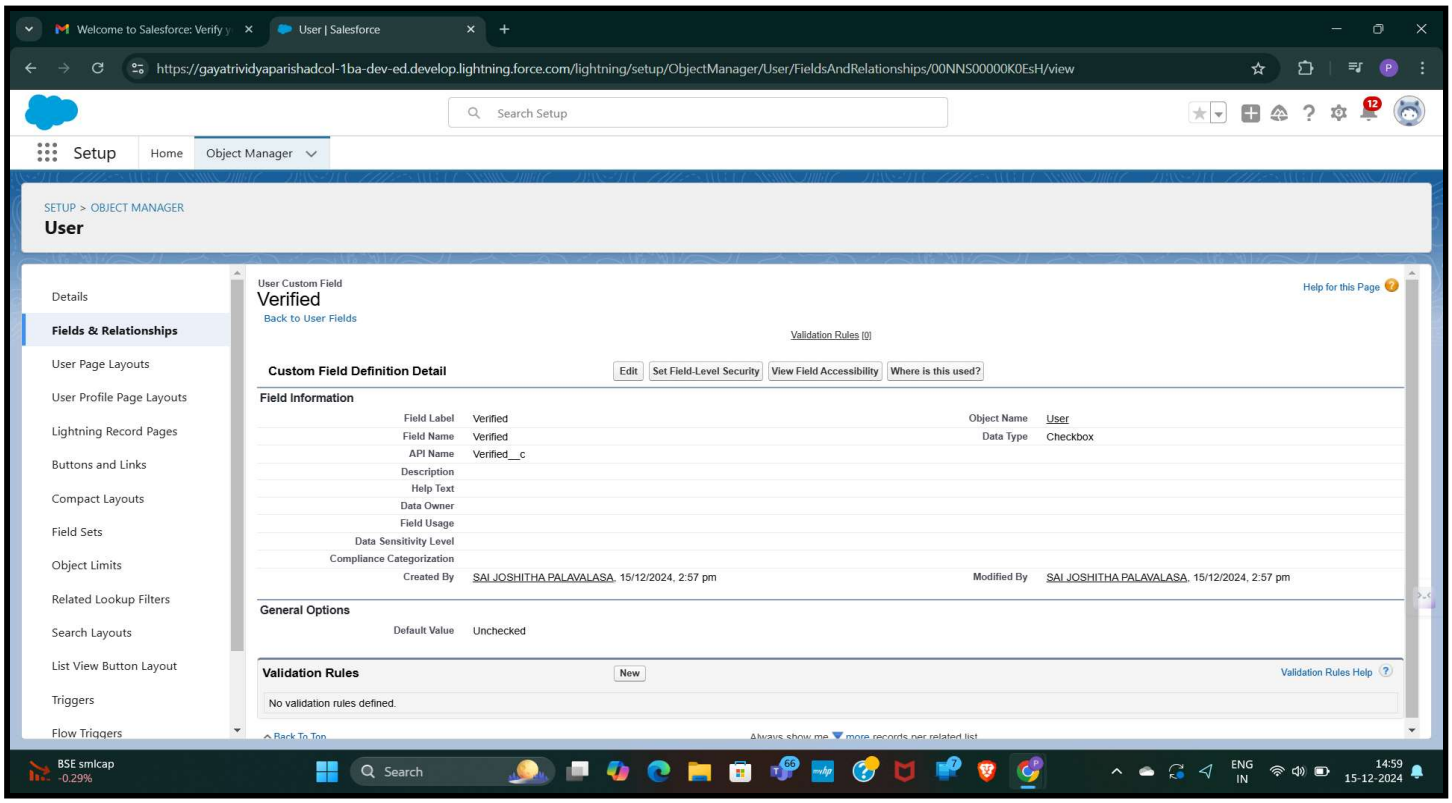
Purpose of the "Verified" Checkbox Field

The "Verified" checkbox field is used to identify whether a user has been verified in the system. It helps in controlling access, tracking user verification status, or enabling certain features based on whether the user is verified. For example:

- **Use Case:** A verified user might have access to restricted data or functionality, while unverified users may be limited to basic features.

Steps to Create a Checkbox Field Named "Verified" on the User Object

1. **Access Object Manager:**
   - Go to **Setup**, search for **Object Manager**, and select the **User** object.
2. **Create a New Field:**
   - Navigate to **Fields and Relationships**, click **New**, and select **Checkbox** as the field type.
3. **Configure Field Details:**
   - Name the field **Verified**, set the default value as **Unchecked**, and proceed to the next step.
4. **Set Permissions and Layout:**
   - Define field-level security for profiles and add the field to the required page layouts.
5. **Save:** Click **Save** to complete the creation.
   -



Purpose of Users in Salesforce

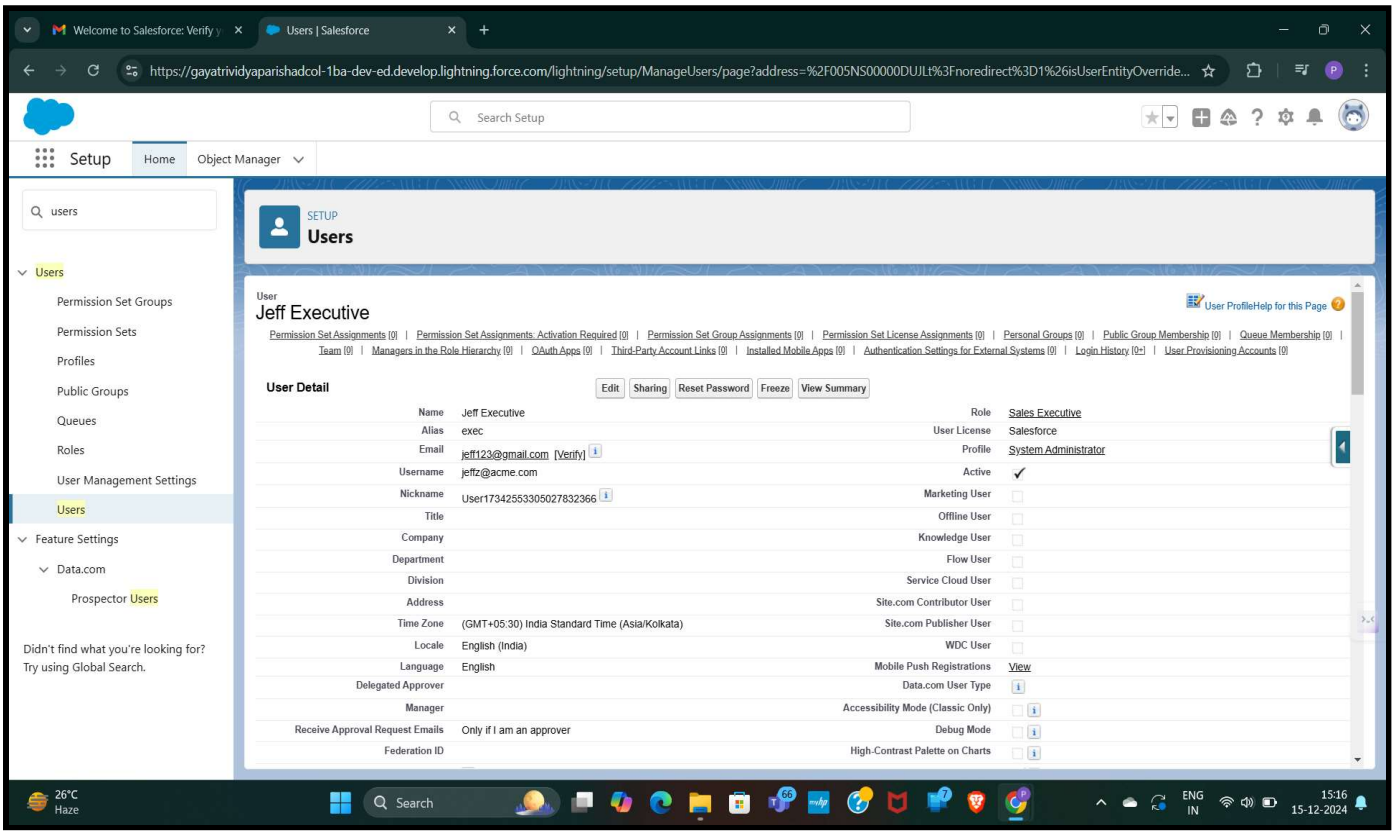
Users in Salesforce represent individuals who log into the system. Each user is assigned a role, profile, and license to control their data access, permissions, and functionality within the organization.

Steps to Create Users

User 1: Sales Executive

Purpose: Represents a system administrator with access to manage all data and configurations.

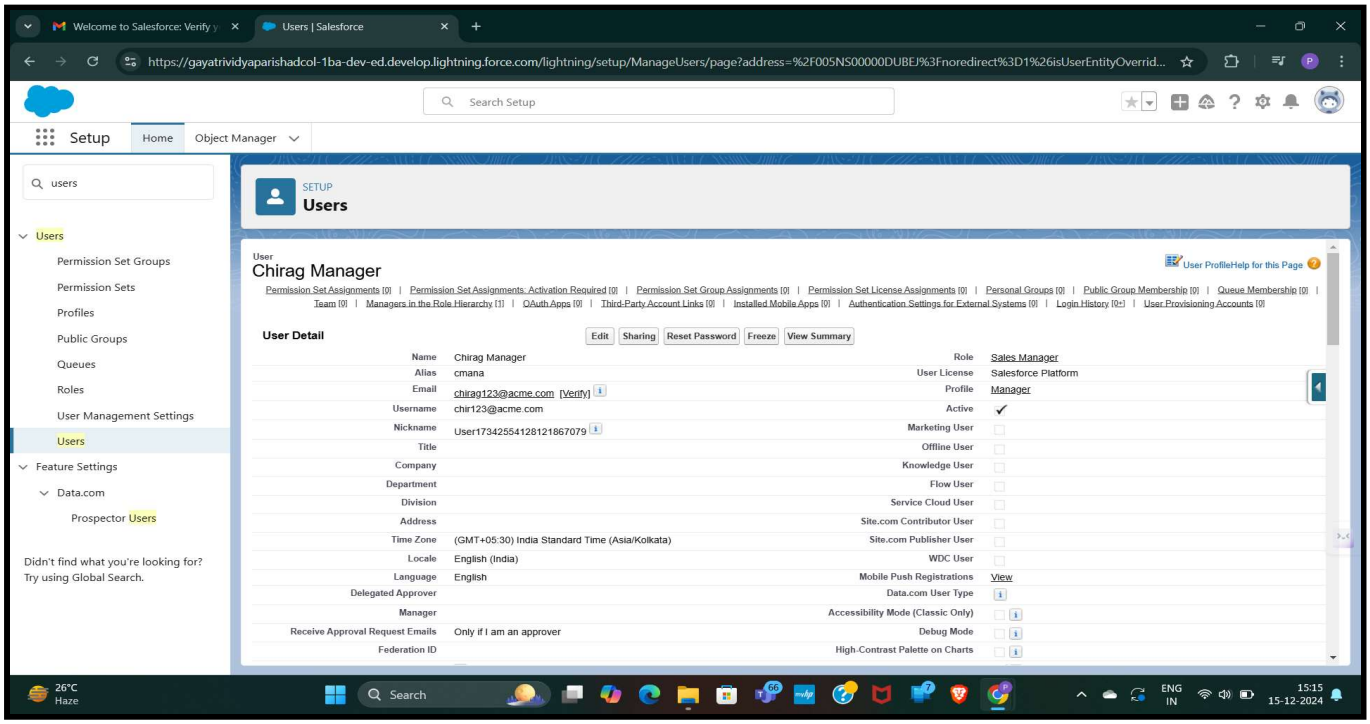
1. Go to **Setup** > **Administration** > **Users** > **New User**.
2. Enter **Last Name:** Executive, select **Role:** Sales Executive, and **License:** Salesforce.
3. Assign **Profile:** System Administrator and click **Save**.



User 2: Manager

Purpose: Manages sales operations and oversees data for their team.

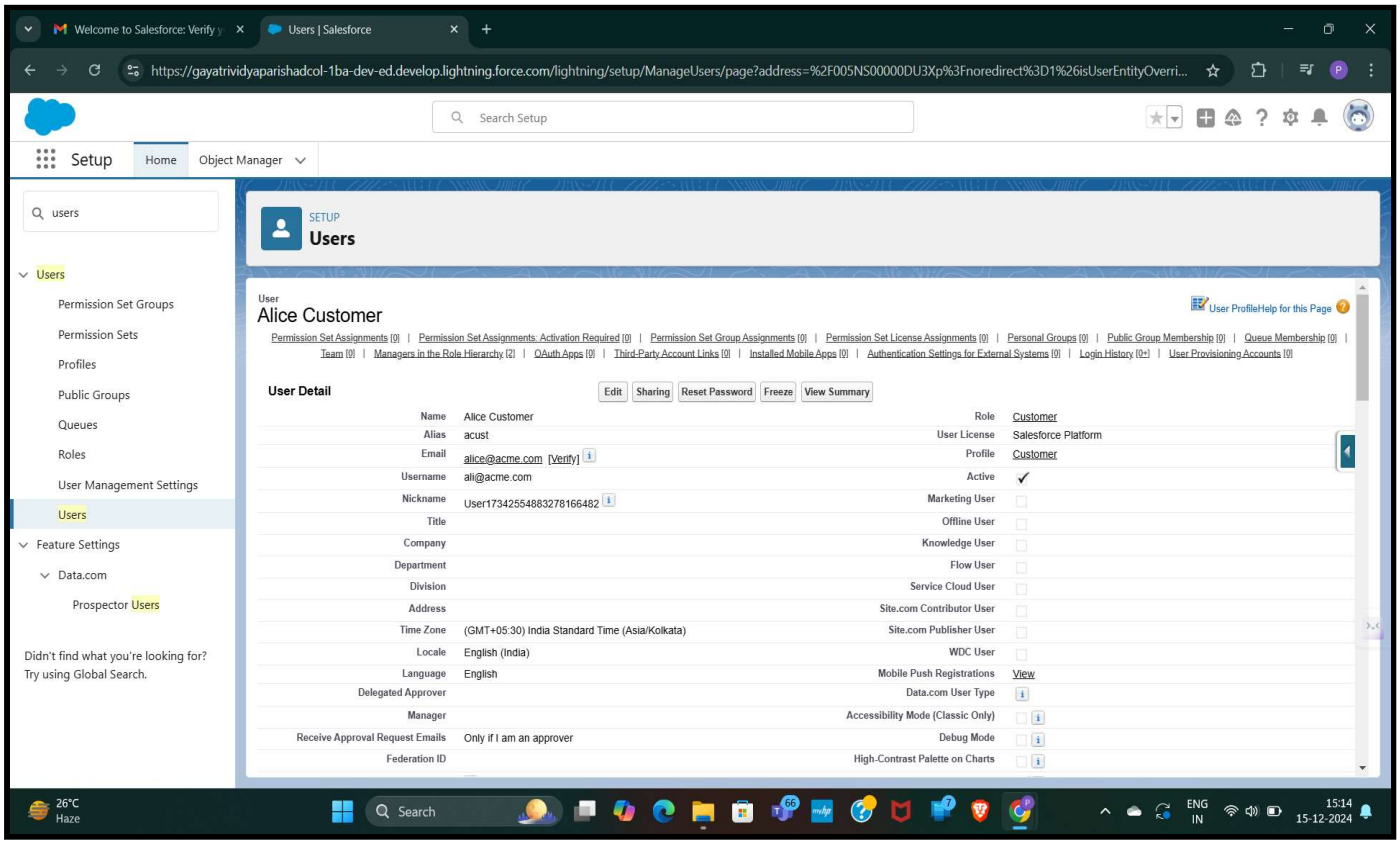
1. Go to **Setup** > **Administration** > **Users** > **New User**.
2. Enter **Last Name:** Manager, select **Role:** Sales Manager, and **License:** Salesforce Platform.
3. Assign **Profile:** Manager and click **Save**.



User 3: Customer (Unverified)

Purpose: Represents a customer with limited access to property-related data.

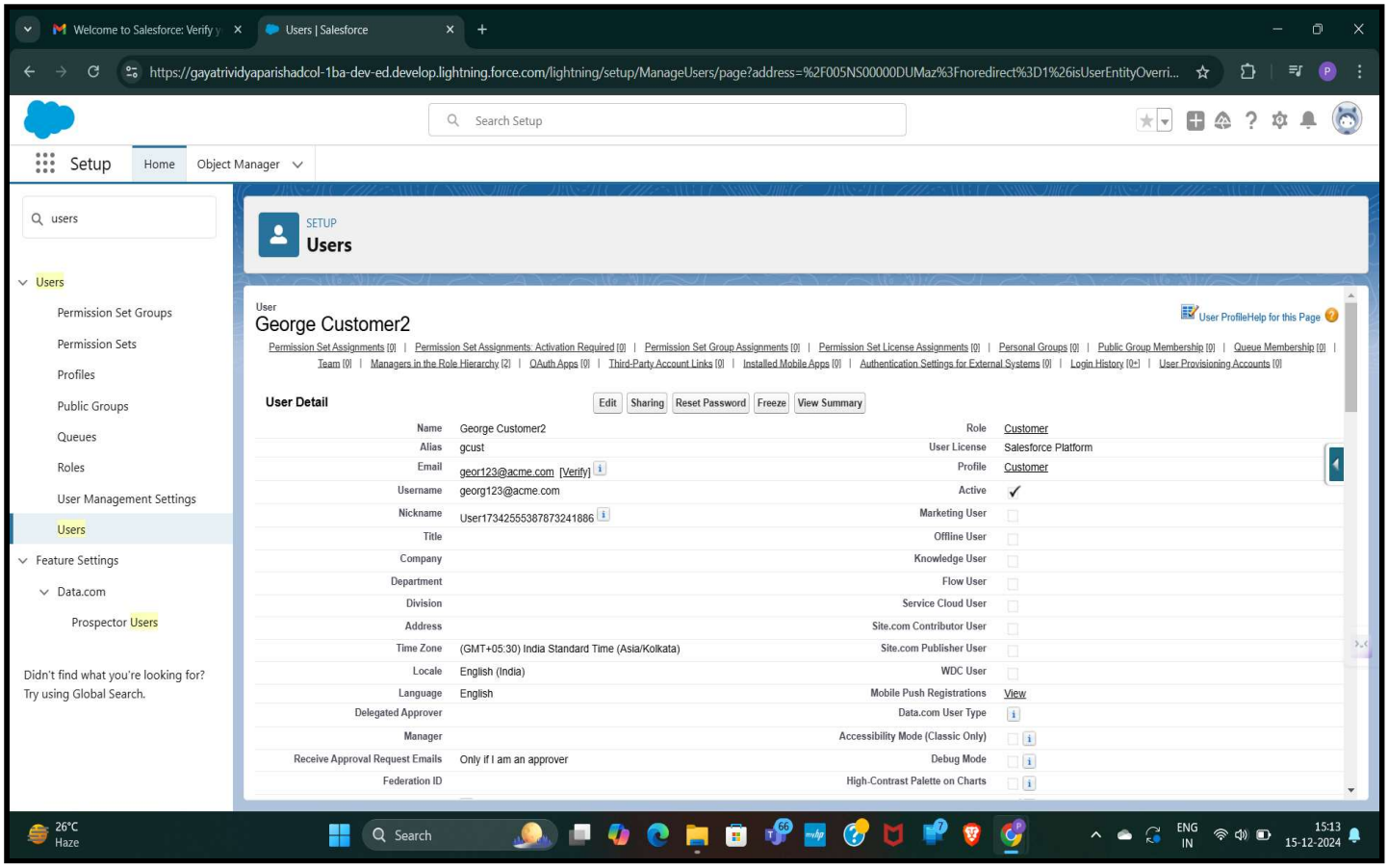
1. Go to **Setup** > **Administration** > **Users** > **New User**.
2. Enter **Last Name:** Customer, select **Role:** Customer, and **License:** Salesforce Platform.
3. Assign **Profile:** Customer, ensure the **Verified** checkbox is **Unchecked**, click save.



User 4: Customer2 (Verified)

Purpose: Represents a verified customer with access to view approved property details.

1. Go to **Setup** > **Administration** > **Users** > **New User**.
2. Enter **Last Name:** Customer2, select **Role:** Customer, and **License:** Salesforce Platform.
3. Assign **Profile:** Customer, ensure the **Verified** checkbox is **Checked**, and click **Save**.



Property Approval Process

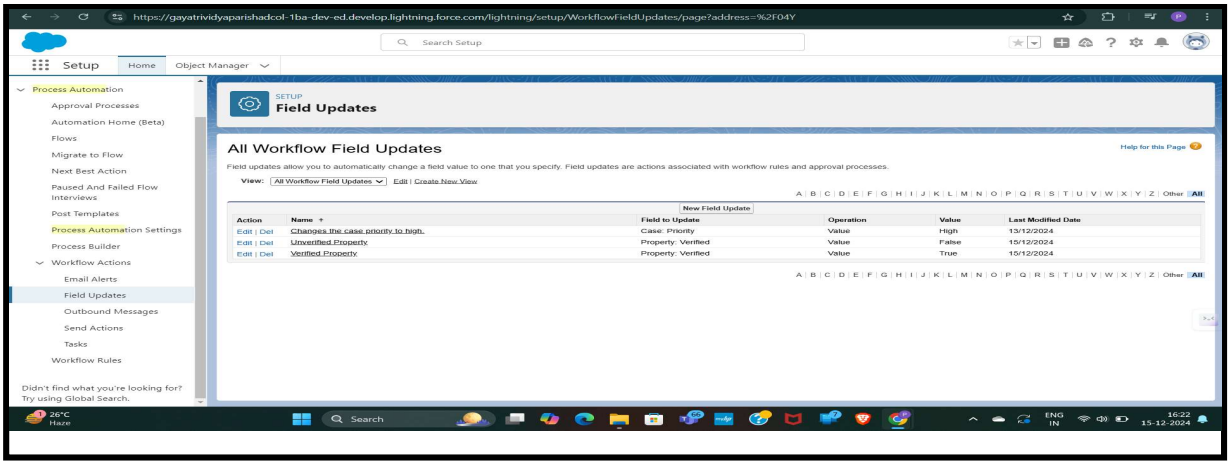
Purpose: The Property Approval process ensures that property records are reviewed and approved based on their verification status. It streamlines decision-making by automating approval routing and updating the verification status of properties.

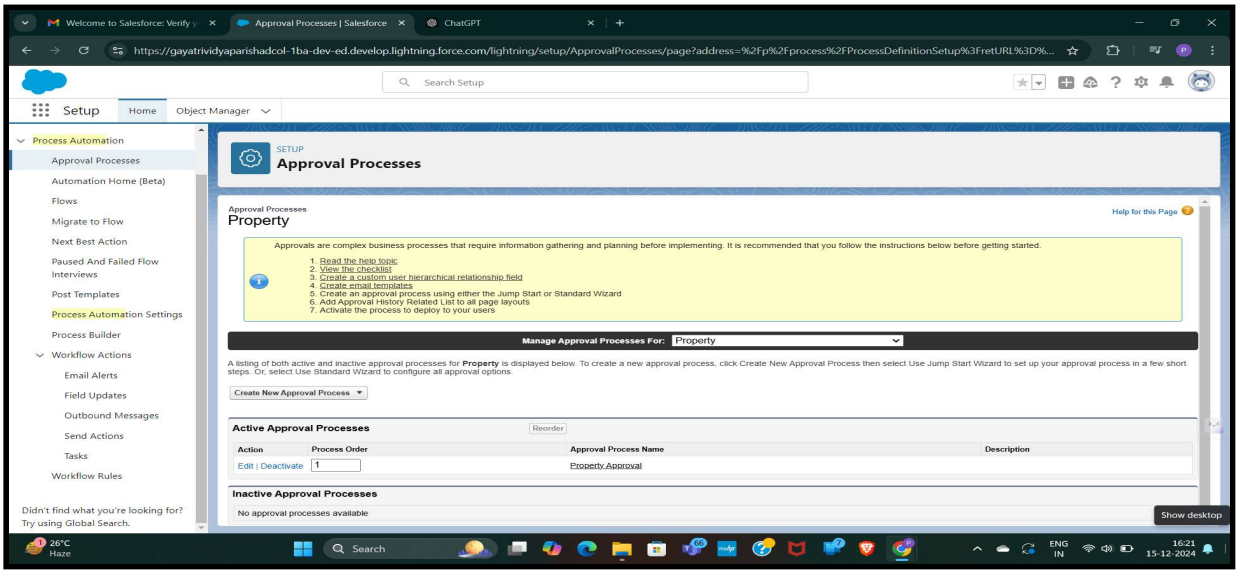
Steps to Create the Property Approval Process:

1. **Create Approval Process:**
   - Navigate to **Setup** → **Process Automation** → **Approval Processes**.
   - Select the **Property** object and create a new approval process. Name it **Property Approval**.
2. **Set Criteria for Approval:**
   - **Criteria 1:** Location is not blank.
   - **Criteria 2:** Verified equals **false**.
3. Click **Next**.
4. **Define Approver and Record Editability:**

- Under **Next Automated Approver Determined By**, select **Manager**.
- For **Record Editability**, select **Administrators or the currently assigned approver can edit records during the approval process**.

5. **Initial Submitters:**
   - Add **Property Owner** and **Sales Manager** as initial submitters.
   - Click **Save**.
6. **Add Approval Step - Executive Approval:**
   - Name the step **Executive Approval** and set it to include **all records**.
   - Select **Sales Executive** as the approver.
7. **Field Updates:**
   - Add two field updates:
     - **Verified Property**: Update the **Verified** field to **True**.
     - **UnVerified Property**: Update the **Verified** field to **False**.
8. **Activate** the process and save





What is a Trigger in Salesforce?

A **trigger** in Salesforce is a piece of automation that executes specific actions when certain events occur, such as when a record is created, updated, or deleted. Triggers are essential for enforcing business logic, automating workflows, and ensuring data consistency.

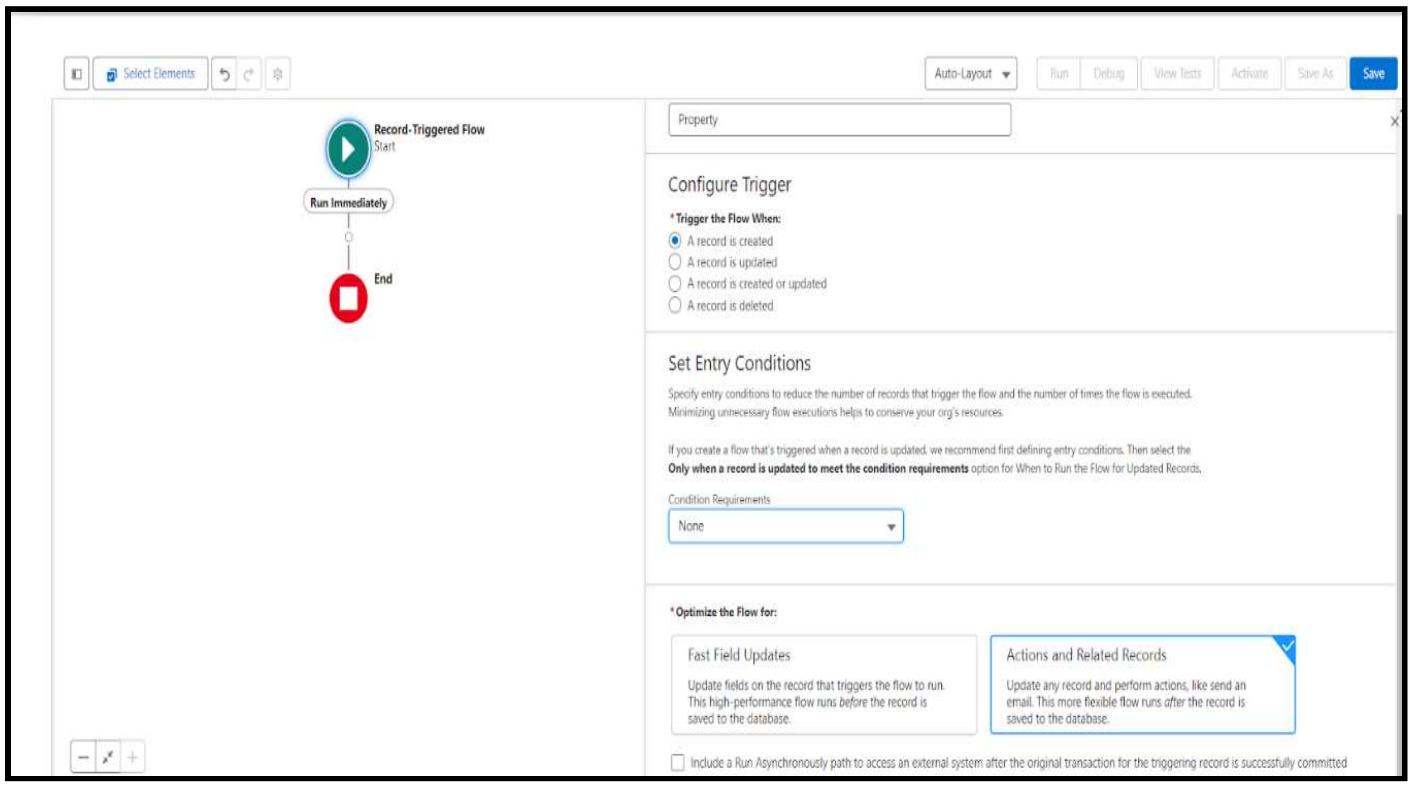
In this case, a **Record-Triggered Flow** acts as a declarative alternative to programmatic triggers, enabling automated submission of records for approval without writing code.

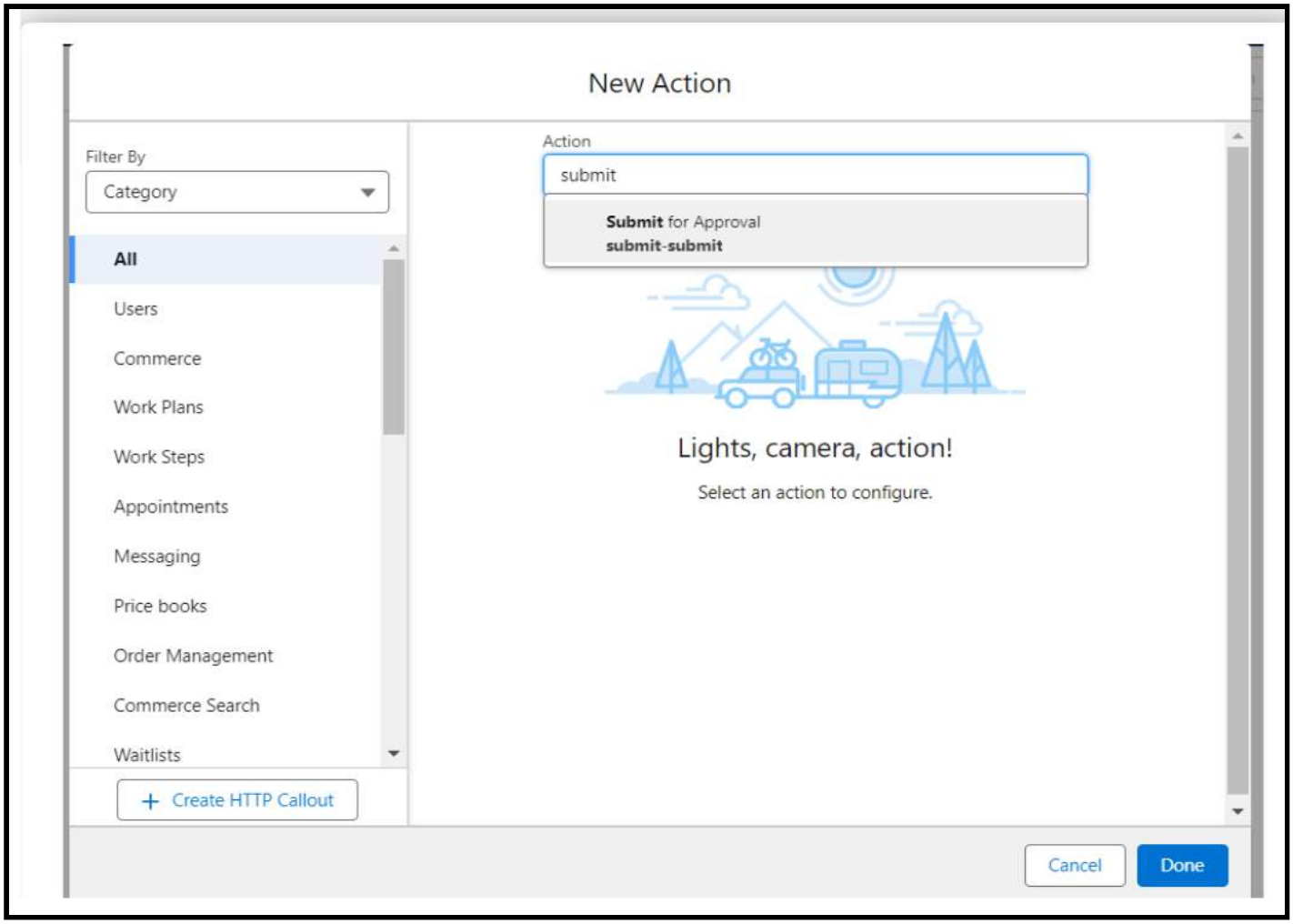
Purpose of the Flow:

This **Record-Triggered Flow** automatically submits property records for approval as soon as they are created, streamlining the approval process and eliminating the need for manual intervention.

Steps to Create the Record-Triggered Flow:

1. **Create a New Record-Triggered Flow:**
   - From **Setup**, search for **Flows** and click on **New Flow**.
   - Select **Record-Triggered Flow** as the flow type.
2. **Configure Trigger Details:**
   - **Object:** Select **Property**.
   - **Trigger the flow when:** Choose **A record is created**.
3. **Set Entry Conditions:**
   - Leave the **Entry Conditions** as **None**, so the flow triggers for all newly created Property records.
4. **Add an Action to Submit for Approval:**
   - Click on the **+ (Add Element)** and select **Action**.
   - In the **Action Type**, search for **Submit for Approval**.
   - Configure the action to automatically submit the record for the appropriate approval process.
   - Save the flow.
5. **Activate the Flow:**
   - After configuring the flow, click **Save** and then **Activate** to enable it.


Select Elements
Auto-Layout
Run
Debug
View Tests
Activate
Save As
Save
Record-Triggered Flow
Start
Run Immediately
End
Property
Configure Trigger
*Trigger the Flow When:
A record is created
A record is updated
A record is created or updated
A record is deleted
Set Entry Conditions
Specify entry conditions to reduce the number of records that trigger the flow and the number of times the flow is executed.
Minimizing unnecessary flow executions helps to conserve your org's resources.
If you create a flow that's triggered when a record is updated, we recommend first defining entry conditions. Then select the
Only when a record is updated to meet the condition requirements option for When to Run the Flow for Updated Records.
Condition Requirements
None
*Optimize the Flow for:
Fast Field Updates
Update fields on the record that triggers the flow to run. This high-performance flow runs before the record is saved to the database.
Actions and Related Records
Update any record and perform actions, like send an email. This more flexible flow runs after the record is saved to the database.
Include a Run Asynchronously path to access an external system after the original transaction for the triggering record is successfully committed


New Action
Filter By
Category
All
Users
Commerce
Work Plans
Work Steps
Appointments
Messaging
Price books
Order Management
Commerce Search
Waitlists
+ Create HTTP Callout
Action
submit
Submit for Approval
submit-submit
Lights, camera, action!
Select an action to configure.
Cancel
Done

What is an App Page in Salesforce?

An **App Page** in Salesforce is a custom Lightning page designed to display specific information and components tailored to a particular business use case. It can combine standard and custom Lightning components, making it a powerful tool for organizing and presenting data in a user-friendly manner.

Purpose of an App Page:

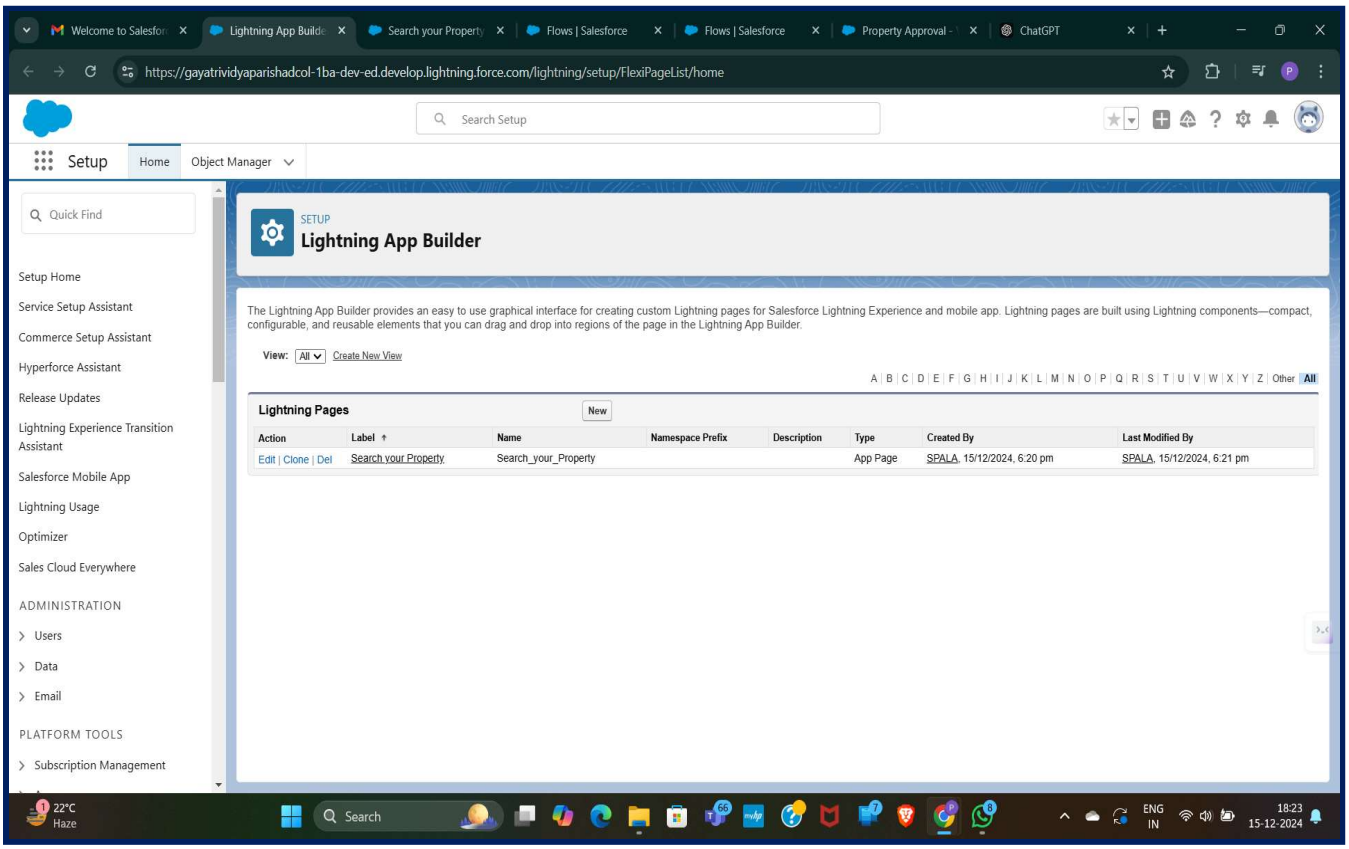
- To provide a consolidated view of related data, tools, or actions for specific objects or business functions.
- To enhance user productivity by creating a personalized and intuitive interface.
- To allow users to interact with records, dashboards, and other resources in a streamlined way.

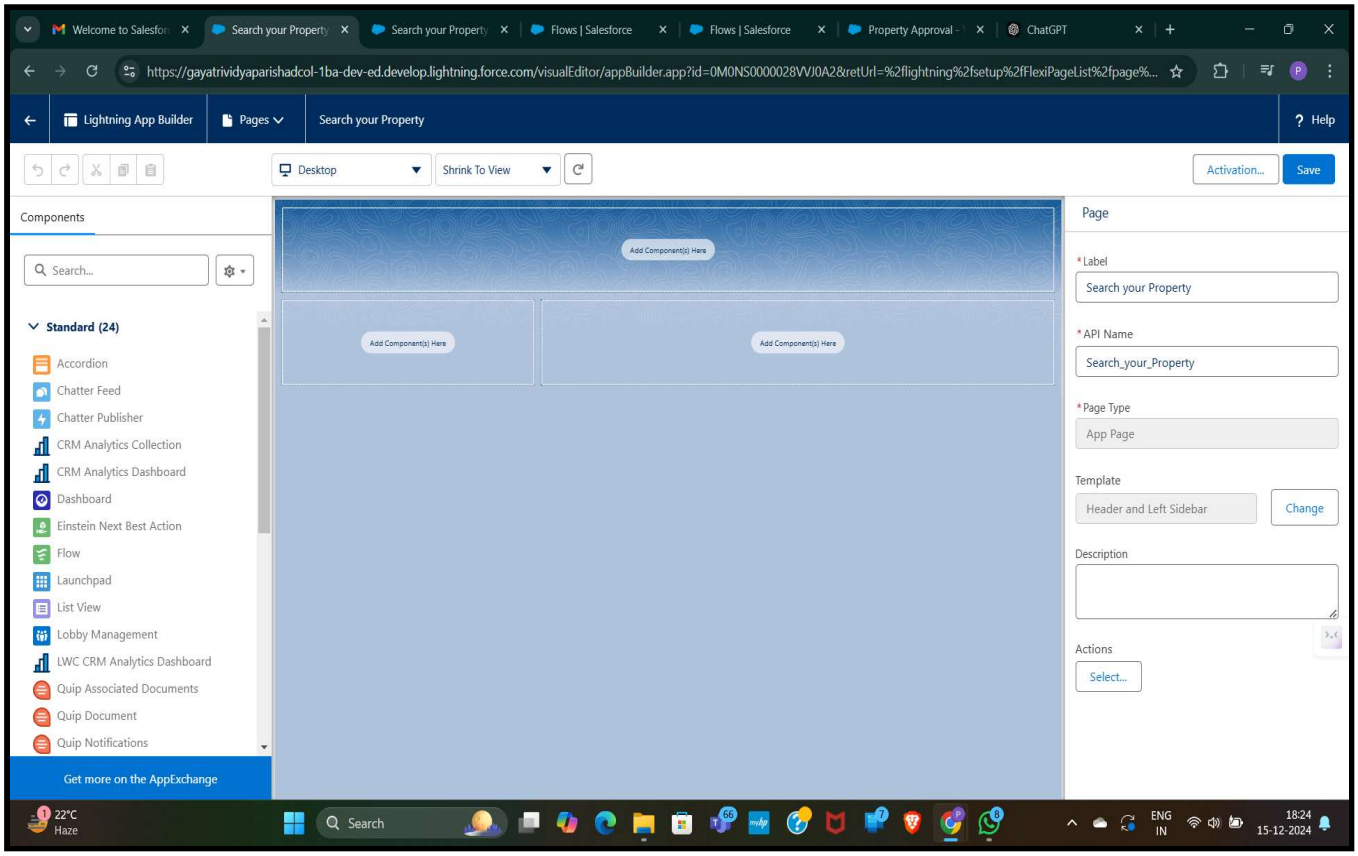
Steps to Create an App Page:

1. **Navigate to the Lightning App Builder:**
    - From **Setup**, search for **Lightning App Builder** and click on it.
    - Click on **New** to create a new Lightning page.
2. **Choose the Page Type:**
    - Select **App Page** and click **Next**.
3. **Define Page Label:**
    - Enter the label as **Search Your Property**.
    - Click **Next**.
4. **Select Page Layout:**
    - Choose the layout **Header and Left Sidebar**.
    - Click **Done**.
5. **Customize the Page:**
    - Add components such as:
        - **Search Component:** To search for property records.
        - **Record List:** To display property records.
        - **Charts or Dashboards:** For visual insights.
    - Arrange components in the layout as needed.

6. **Save and Activate the Page:**
    - Click **Save**.
    - Click **Activate** to make the page available.
7. **Activate for All Users:**
    - In the **Page Activation Settings**, select **Activate for All Users**.
    - Confirm the activation.

NOTE: Creation of app builder is shown in the figures(mentioned in next page)





What is a Lightning Web Component (LWC)?

LWC is a modern framework for building user interfaces on the Salesforce platform. It uses standard web technologies such as HTML, CSS, and JavaScript, and provides optimized performance. LWCs are used to create reusable components for Salesforce applications.

Steps to create an LWC app:

1) Create an Apex Class and make it aura enabled and name it “PropertHandler_LWC”.

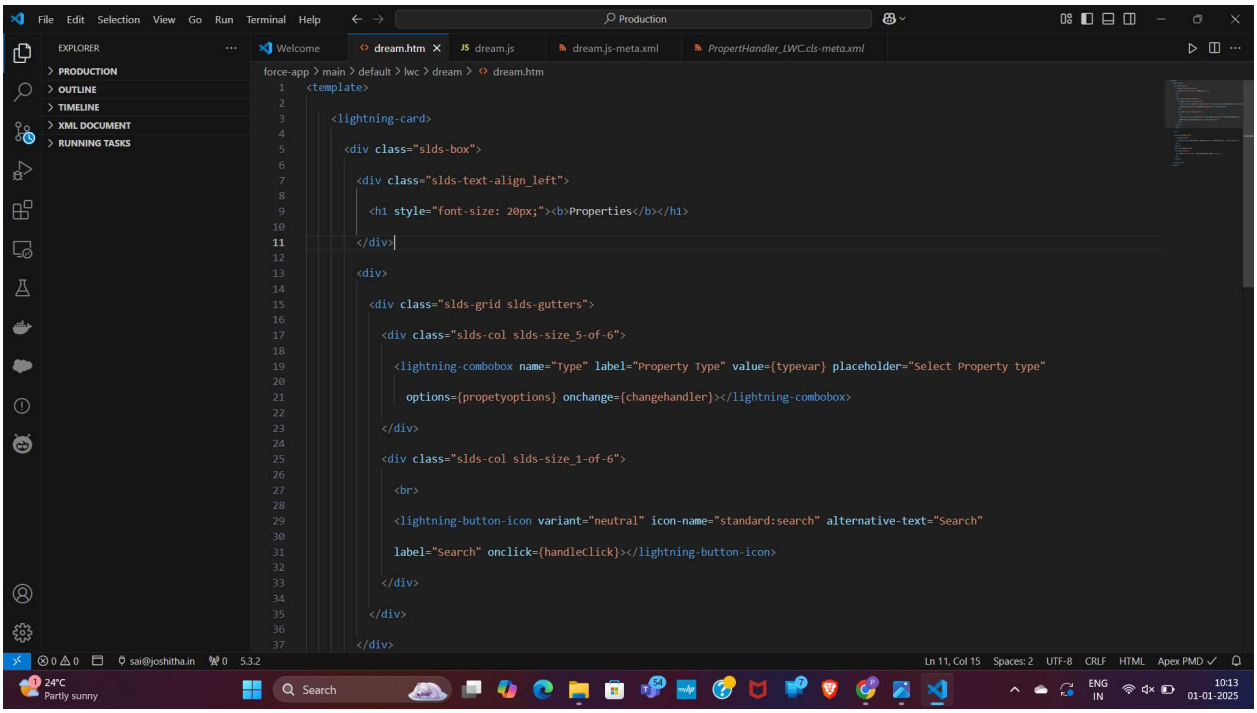
2) Create a Lightning Web Component in your VsCode, and (ctrl+shift +P) and click on authorize an org.

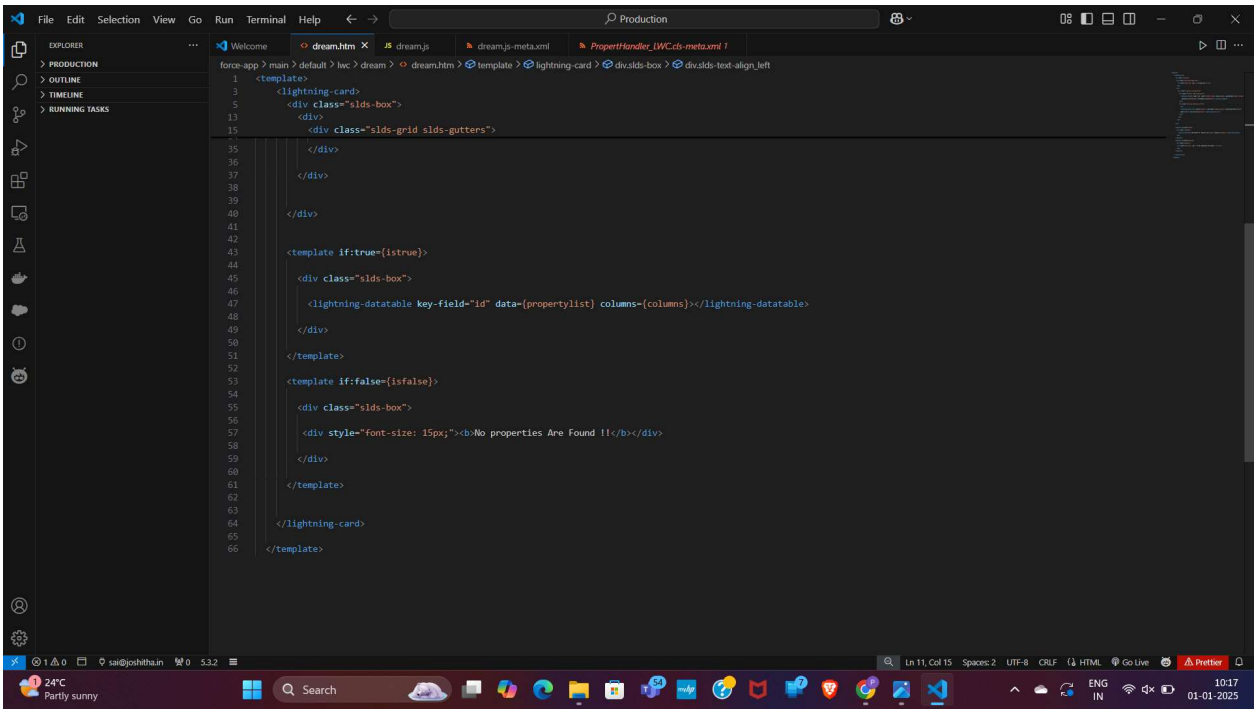
3) Enter your login id and password to authorize your org. Then, click (ctrl+shift +P) and

Create a lightning Web Component and Name it.

HTML CODE

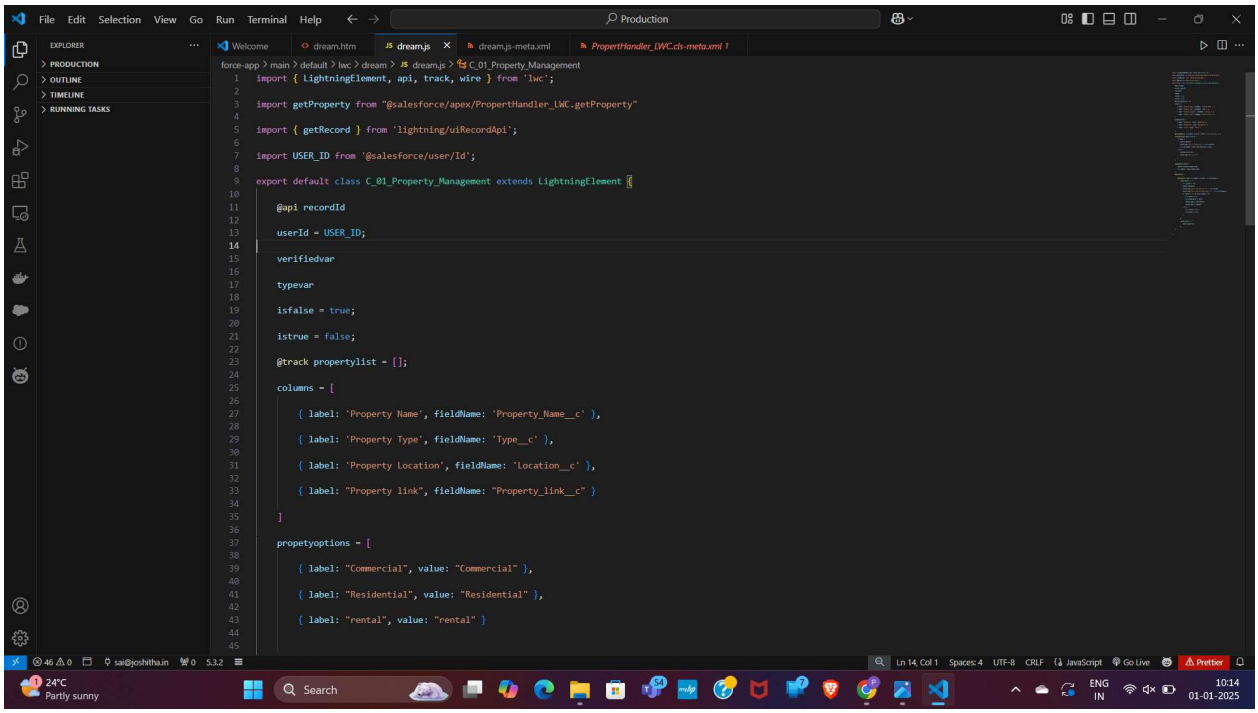
The code below is a HTML code with the name **dream.htm**

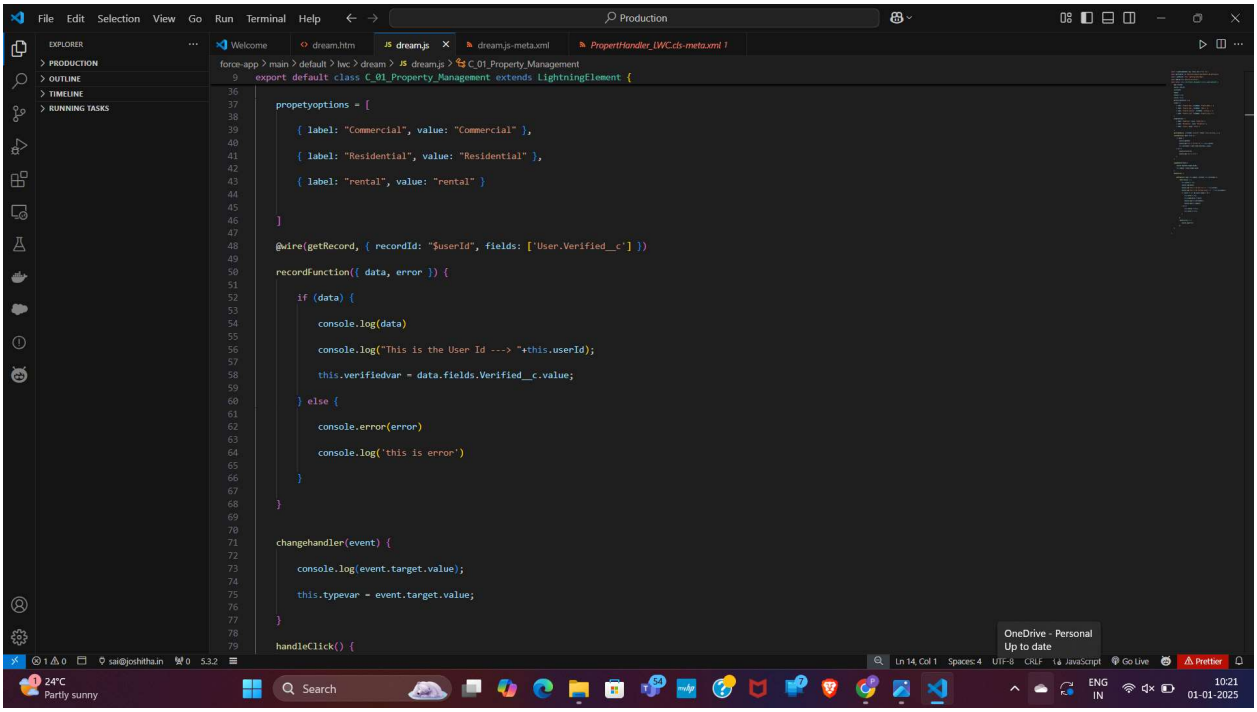




Now we need to write the js code as follows:

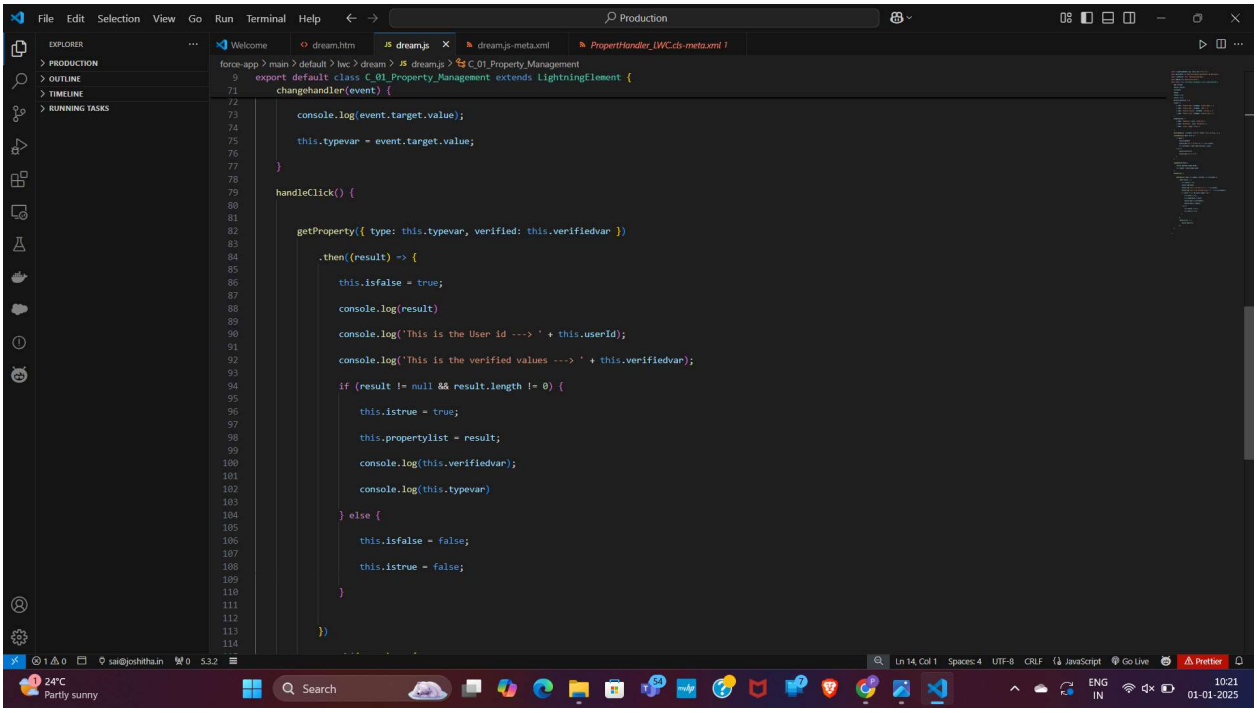
It is saved as **dream.js**



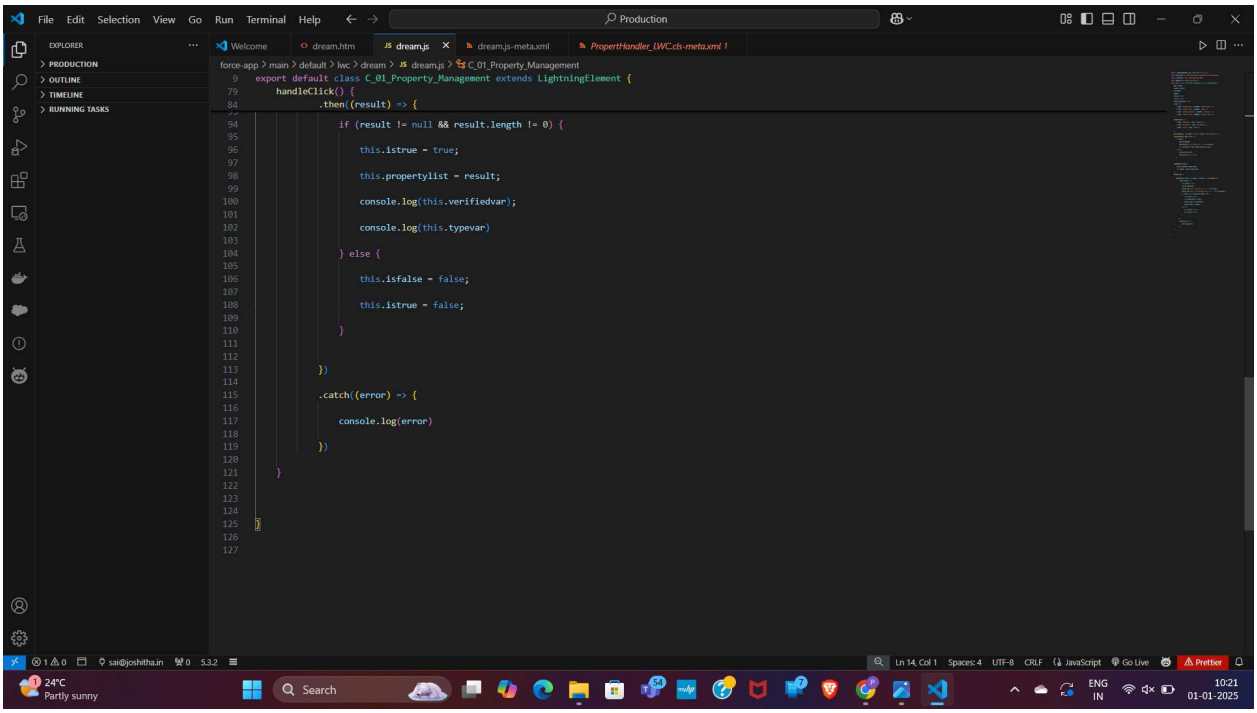


```
export default class C_01_Property_Management extends LightningElement {

    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() {
```

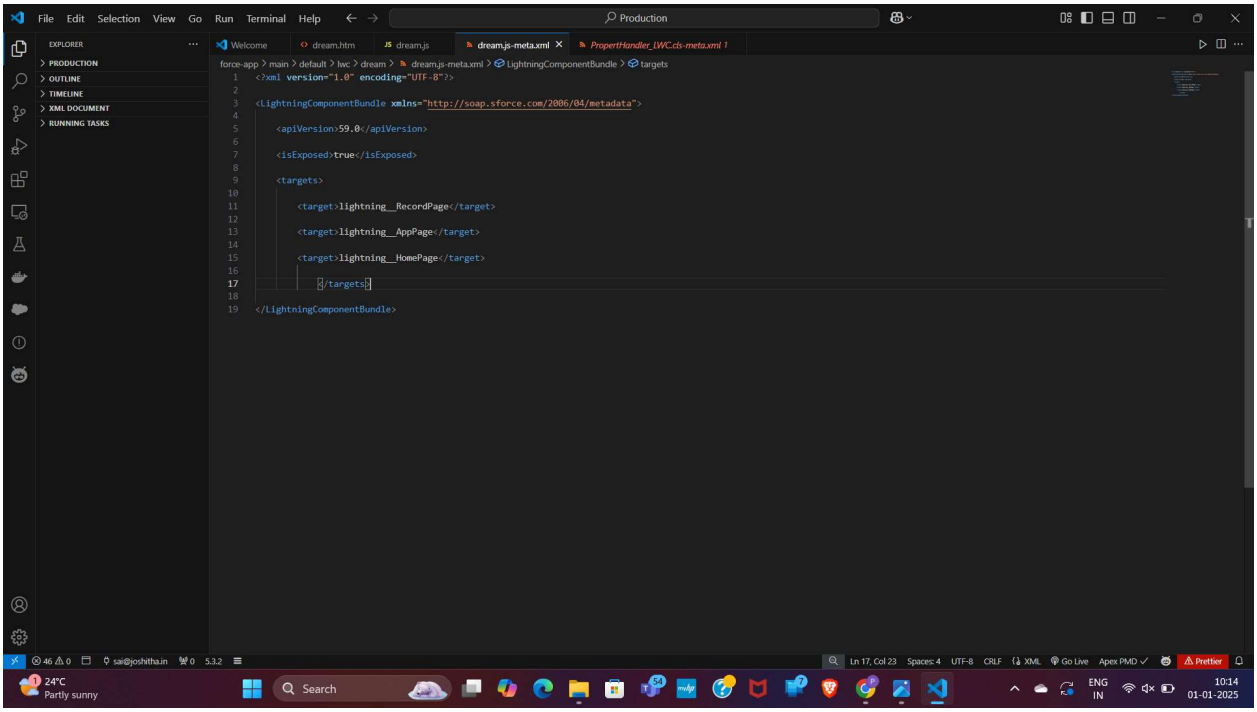
js code



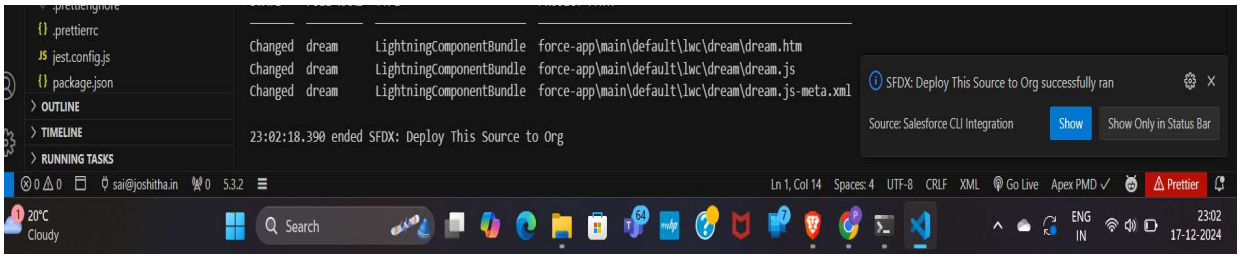
```
export default class C_01_Property_Management extends LightningElement {
    changehandler(event) {

        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;

                }


            })
```



Next>>In the metafile, we need to give our targets to deploy the component. The code is saved as **dream.js-meta.xml**

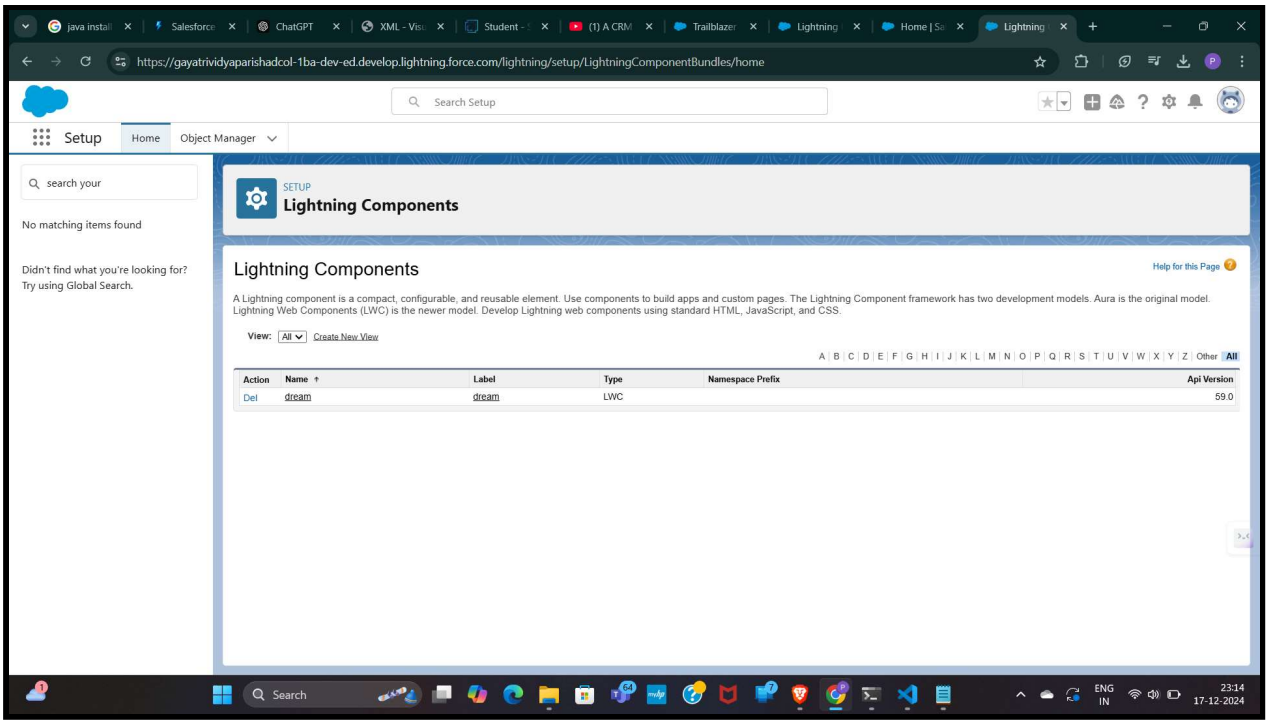


After Saving all the three Codes , Right Click and deploy this component to the org. This looks like (as below):



It clearly displays on the right that SFDX: Deploy This Source to Org successfully ran.This means that we successfully deployed our source into salesorce org.

Go to Salesforce>>check if dream is present or not(for verification of salesforce cli integration):



This clearly shows the integration of dream into the org. So, it is successfully deployed.

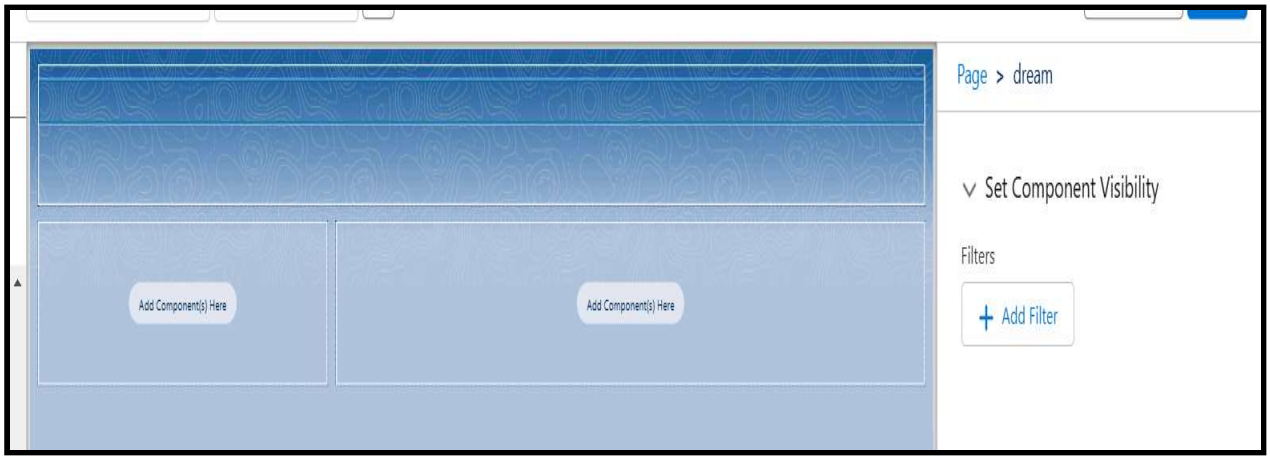
Then drag and drop all the components to app page.

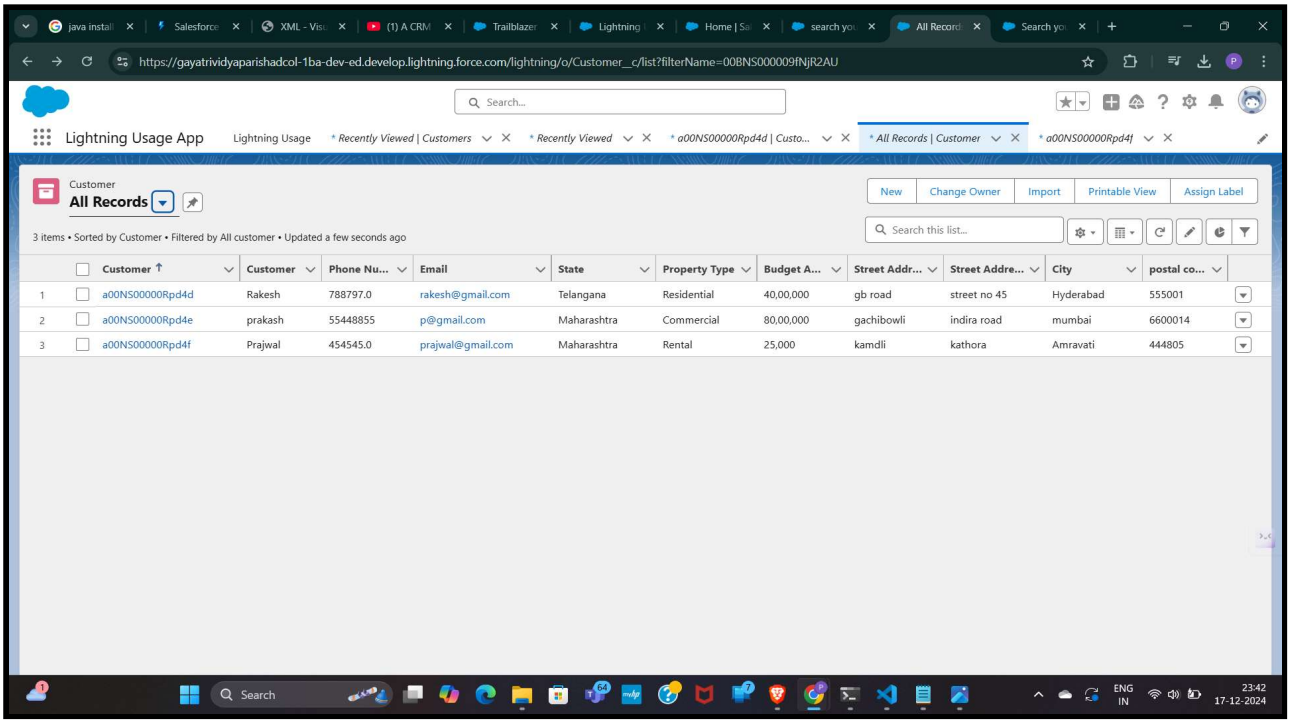
Steps:

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

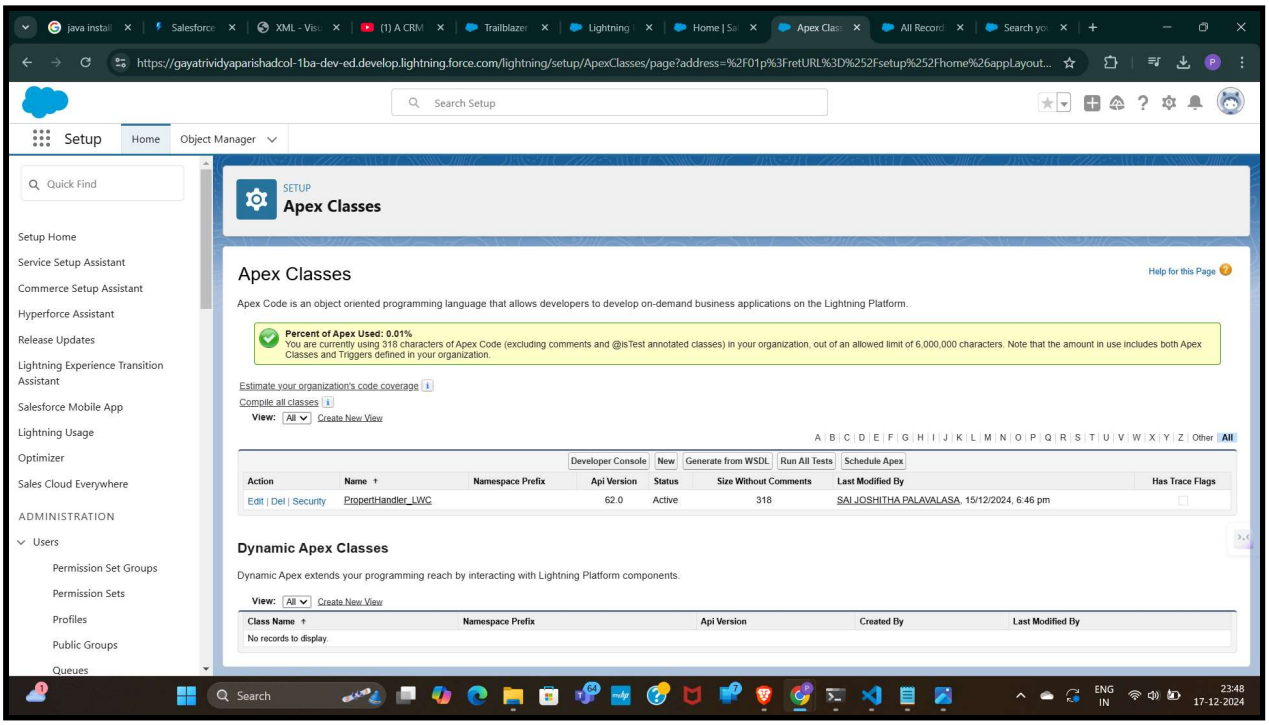




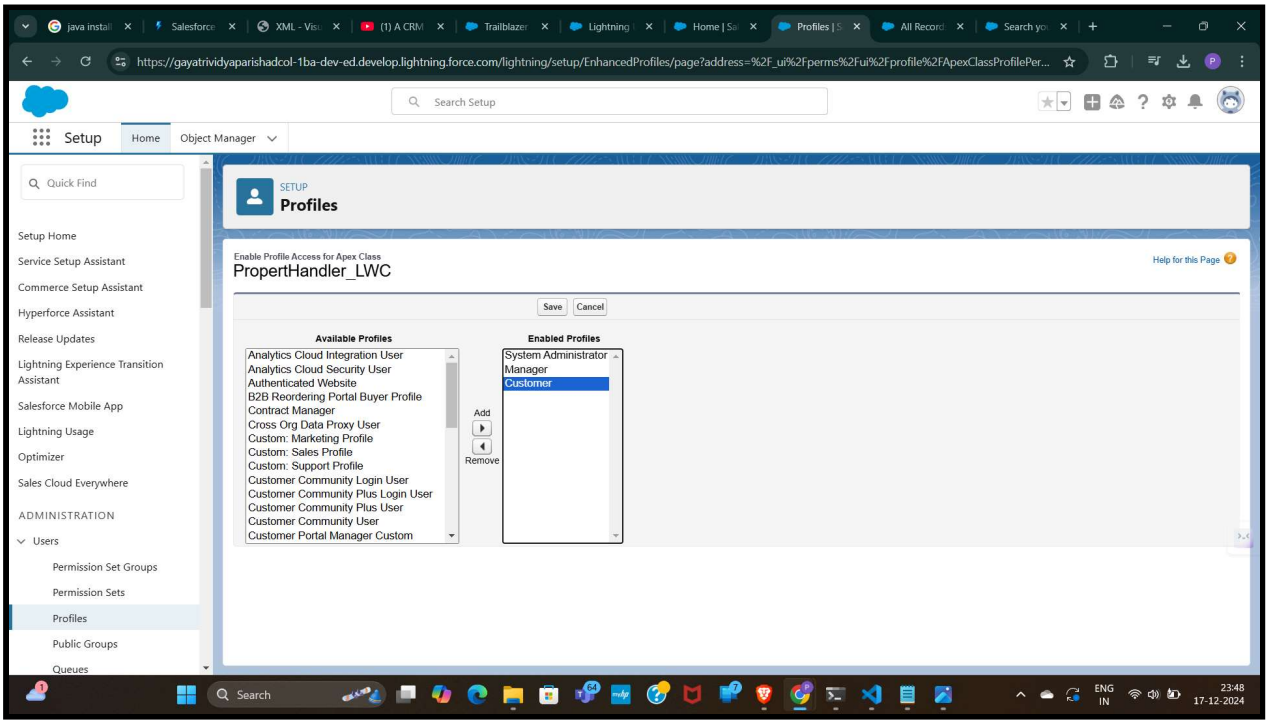
Finally, give access of apex classes to profiles

Steps:

1) From Setup >> Search For Apex Classes >> Click on “Security” behind “PropertyHandler__LWC”.



2)Then, From Profiles Add “Manager” and “Customer” and “Save”.



5.TESTING AND VALIDATION

Unit Testing:

- Apex classes and triggers tested to ensure they handle edge cases and return accurate results.
- Achieved >90% code coverage to meet Salesforce standards.

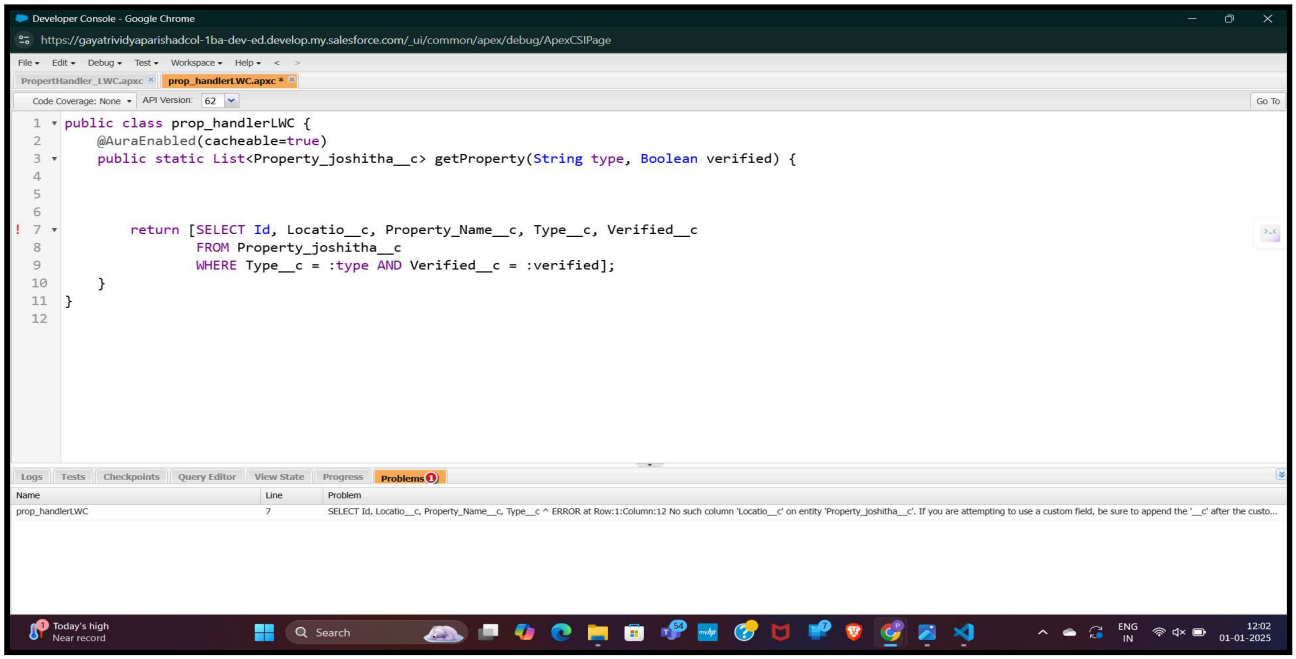
User Interface Testing:

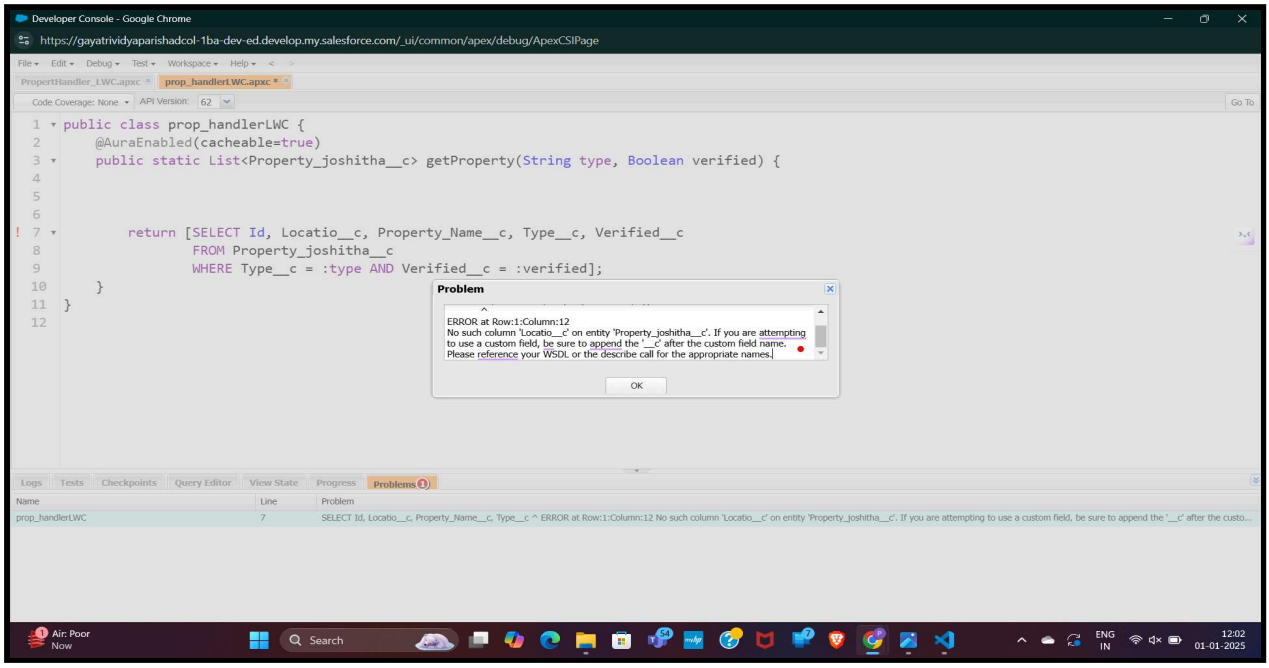
- Validated all forms and pages across different browsers and devices.
- Ensured consistent user experience and accurate data presentation.

End-to-End Testing:

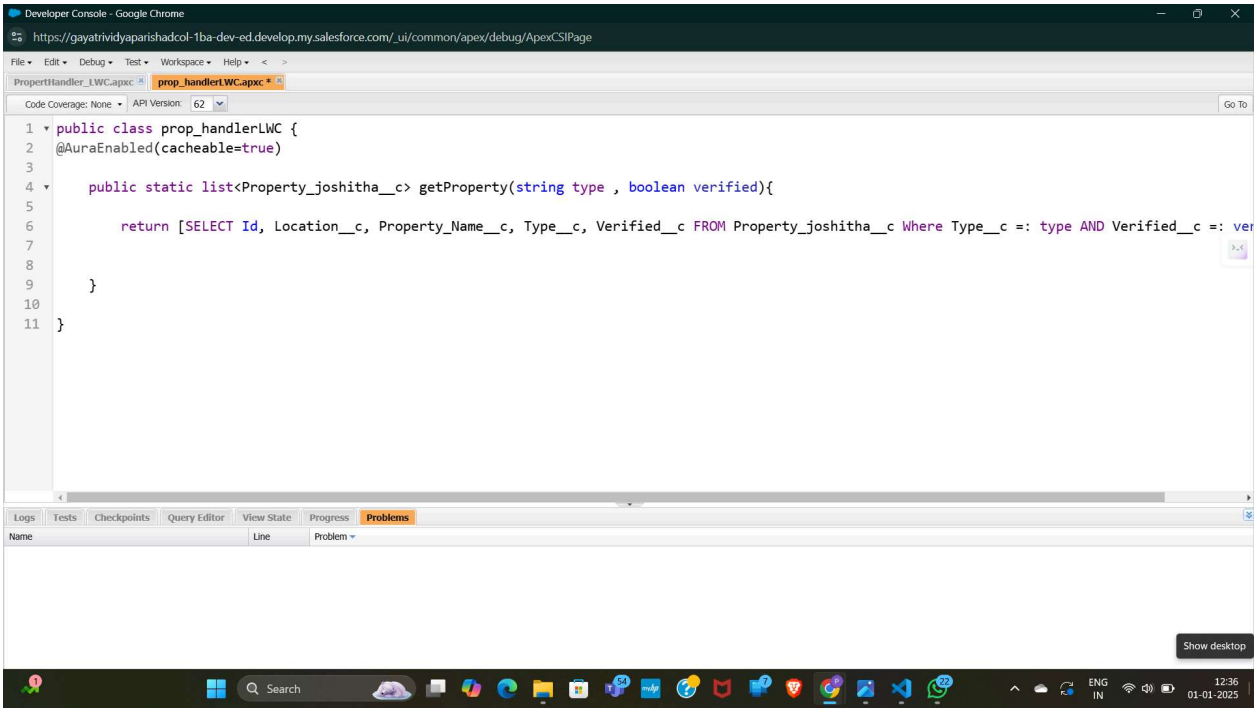
- Simulated real-world scenarios, such as creating customer records, submitting property approvals, and managing property data, to confirm seamless functionality.

Case-1: Sample error testing of apex class is shown as below

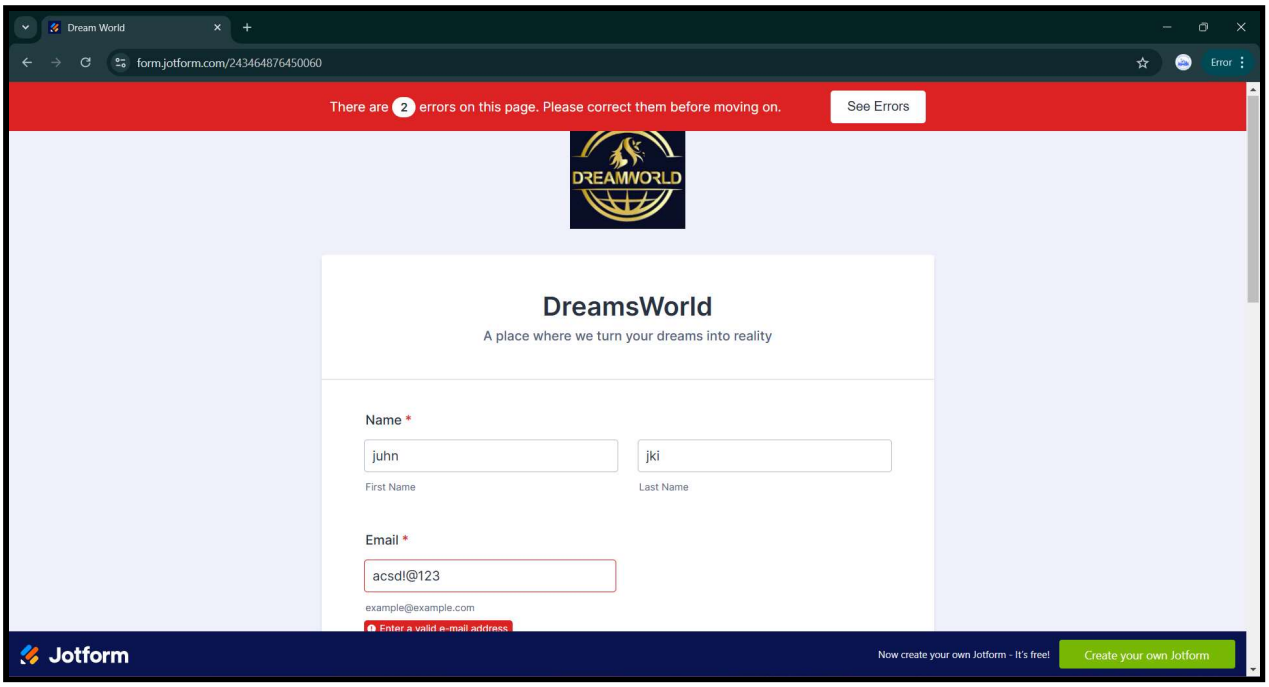


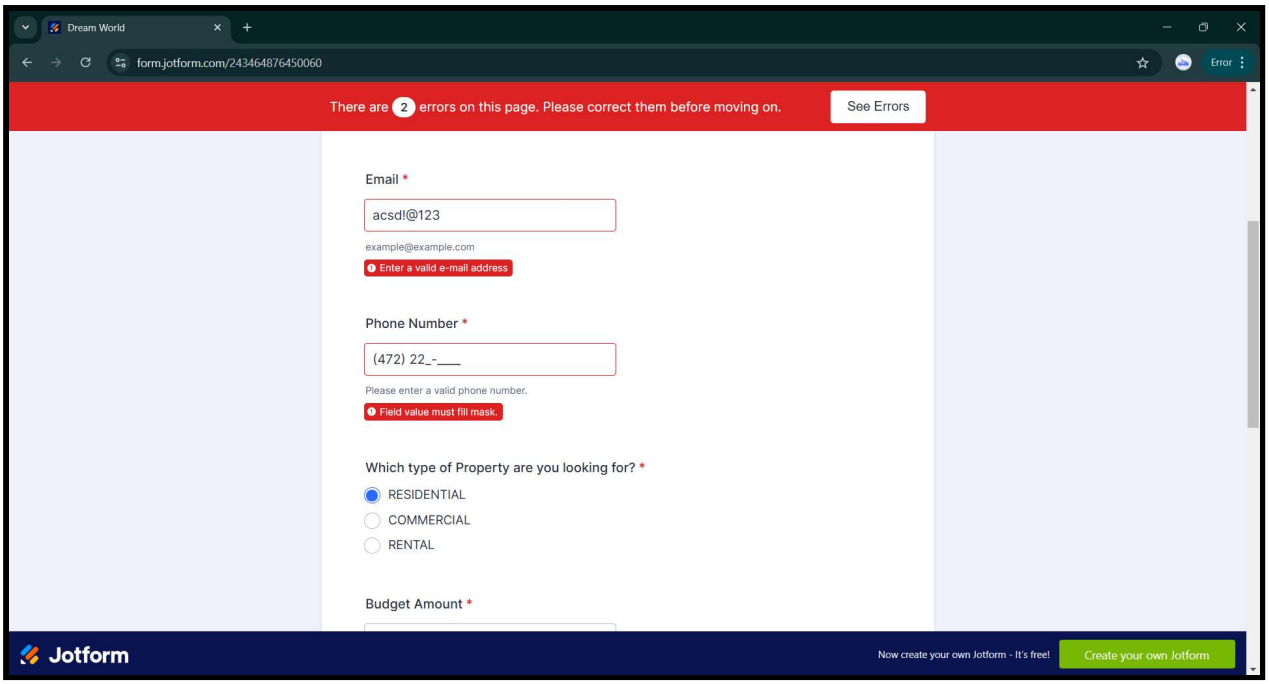


CORRECTED CODE(AFTER REMOVING ERROR):

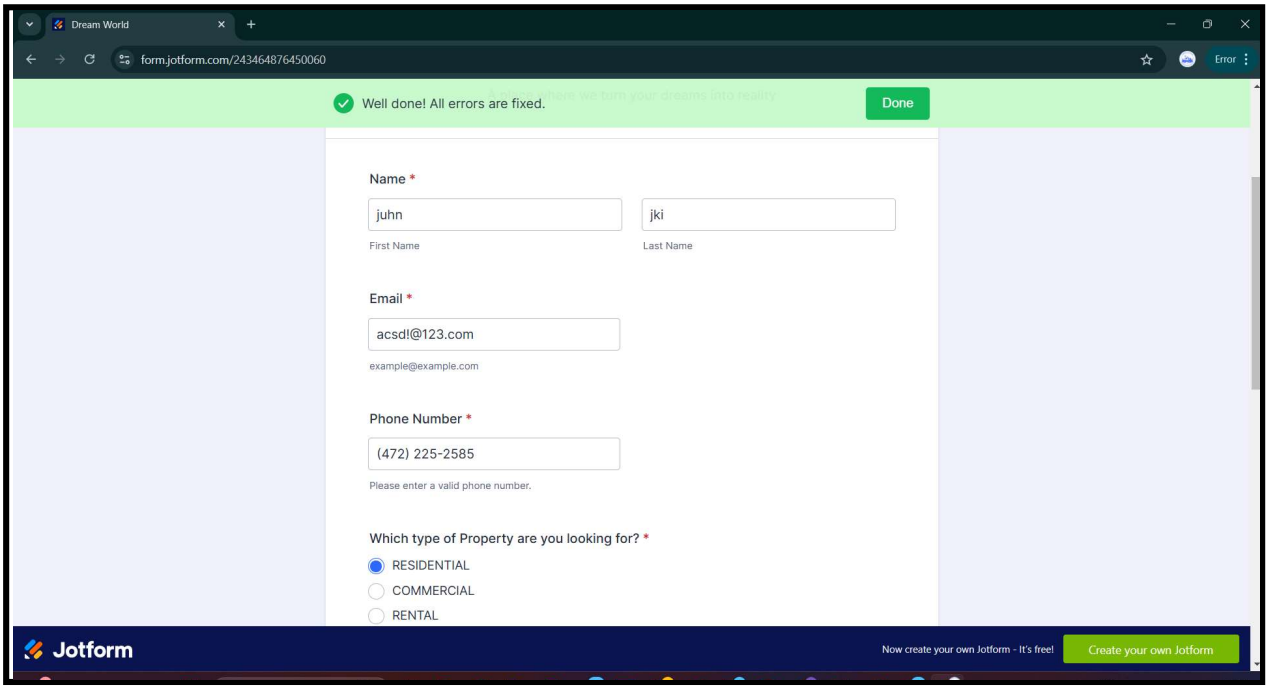


Case-2: Jotform invalid details





Corrected the errors and now able to submit form(error-free)



6.KEY SCENARIOS ADDRESSED BY SALESFORCE

1. **Automated Customer Record Creation**
   - Website interactions trigger automated record creation in Salesforce, capturing essential customer details like name, contact information, and property preferences without manual intervention.
2. **Customer Categorization and Tailored Experiences**
   - Salesforce classifies customers as "approved" or "non-approved" based on predefined criteria, enabling personalized property recommendations for approved users and broader listings for non-approved users.
3. **Streamlined Customer Interaction**
   - Centralized CRM functionality in Salesforce ensures all customer interactions are tracked and managed efficiently, providing a unified view of customer requirements and history.
4. **Dynamic Property Filtering and Recommendations**
   - By leveraging Lightning Web Components (LWCs) and Apex, Salesforce

offers tailored property options, ensuring customers can easily find listings that match their preferences.

5. **Real-Time Data Synchronization**
    - Seamless integration between the website and Salesforce ensures real-time updates, reflecting the latest customer preferences and property availability.
6. **Enhanced Customer Engagement**
    - Automated workflows and email notifications keep customers informed about their application status, property recommendations, and upcoming opportunities, fostering better engagement.
7. **Operational Efficiency**
    - Salesforce optimizes internal operations by automating repetitive tasks, such as data entry and user categorization, reducing the workload for the sales team.
8. **Scalability and Growth Enablement**
    - The scalable architecture supports the addition of new features like predictive analytics, customer feedback tracking, and advanced property recommendation engines to drive business growth.

By addressing these scenarios, Salesforce empowers **Dreams World Properties** to enhance user experience, streamline operations, and improve overall efficiency in managing customer and property-related requirements.

7. CONCLUSION

Summary of Achievements

This project successfully implemented a Salesforce-based CRM application to handle client and property-related requirements. The key accomplishments include:

1. **Streamlined Operations:** Automated workflows for property management, approvals, and notifications, reducing manual efforts and errors.
2. **Custom Data Management:** Created custom objects and relationships to effectively manage client and property data.
3. **Role-Based Access Control:** Defined roles, profiles, and hierarchies to ensure secure and efficient data access.

4. **Dynamic User Interfaces:** Developed Lightning Web Components and app pages to enhance user experience.
5. **Automation and Validation:** Implemented record-triggered flows, validation rules, and approval processes for seamless operations.
6. **Enhanced Decision-Making:** Configured reports and dashboards (if applicable) for insights into property trends, customer interactions, and sales performance.

These achievements demonstrate the effectiveness of Salesforce in building scalable, user-friendly, and efficient CRM solutions tailored to business needs.