|  |
| --- |
| A CRM Application to Handle the Clients and their property Related Requirements |
|  |

    BY

SYAMALA SAI NADH REDDY

[sainadhreddy\_s@srmap.edu.in](mailto:sainadhreddy_s@srmap.edu.in)

Abstract:

This project focuses on developing a CRM application to manage clients and their property-related requirements. The project involves integrating JotForm with Salesforce to automatically create customer records, creating objects from a spreadsheet, defining roles and profiles, developing a property details app, and setting up an approval process for property objects. Additionally, the project includes creating an app page with a Lightning Web Component (LWC) and providing profile access to Apex classes.

**INDEX**

|  |  |
| --- | --- |
| Topic | Page No |
| Project Abstract | 1 |
| Introduction | 3 |
| Task 1: Create a JotForm and Integrate It with the Org | 4 |
| Task 2: Create Objects from Spreadsheet | 6 |
| Task 3: Create Roles | 7 |
| Task 4: Create a Property Details App | 8 |
| Task 5: Create Profiles | 9 |
| Task 6: Create a Checkbox Field on User | 10 |
| Task 7: Create Users | 11 |
| Task 8: Create an Approval Process for Property Object | 12 |
| Task 9: Create a Record Trigger Flow to Submit the Approval Process Automatically | 13 |
| Task 10: Create an App Page | 14 |
| Task 11: Create a LWC Component | 15 |
| Task 12: Drag This Component to Your App Page | 16 |
| Task 13: Give Access of Apex Classes to Profiles | 17 |

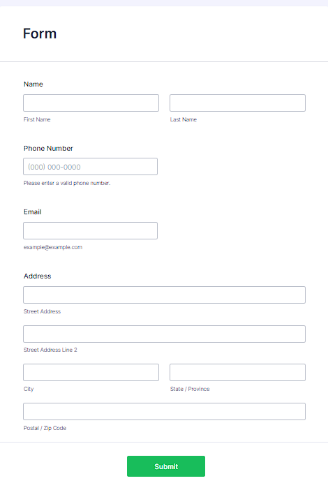
**INTRODUCTION:**

The CRM application aims to streamline the management of clients and their property requirements. It integrates various tools and technologies to automate processes, enhance data management, and improve user experience. This project will guide through the steps of setting up and configuring the CRM application, ensuring efficient and automated handling of customer records and property details.

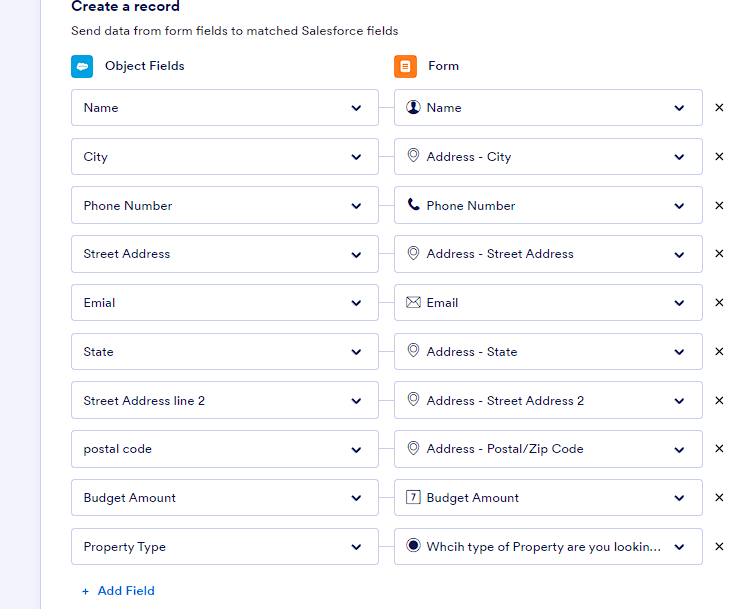
**Task 1: Create a JotForm and Integrate It with the Org to Create a Record of Customers**

**Automatically**

1. **Create a JotForm:**
   * Design a form to capture customer details such as name, contact information, and property requirements.
   * Save the form and get the form URL.
2. **Integrate JotForm with Salesforce:**
   * Navigate to the JotForm integration settings.
   * Select Salesforce and authenticate with your Salesforce credentials.
   * Map the JotForm fields to the Salesforce customer object fields.
   * Save the integration settings.
3. **Verification:**
   * Submit a test form entry.
   * Check Salesforce to ensure the customer record is created automatically.

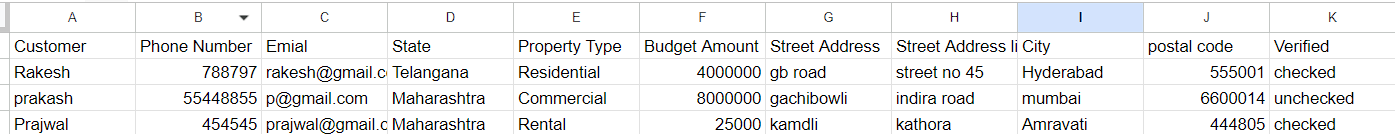
JotForm Design:

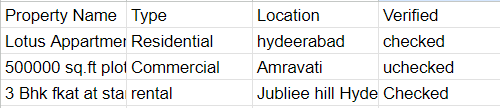
JotForm Integration Settings:



**Task 2: Create Objects from Spreadsheet**

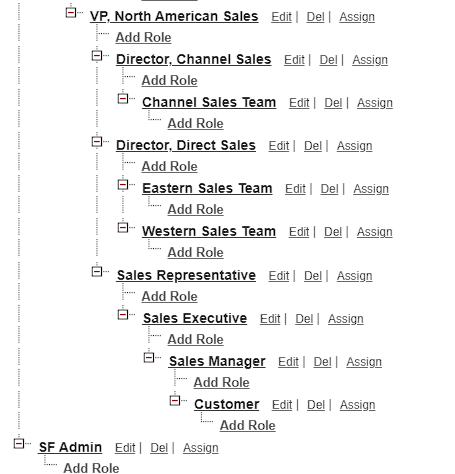
1. **Prepare the Spreadsheet:**
   * Create a spreadsheet with the necessary columns for customer and property details.
   * Save the spreadsheet in CSV format.
2. **Import Data into Salesforce:**
   * Use the Data Import Wizard or Data Loader to upload the CSV file.
   * Map the spreadsheet columns to the corresponding Salesforce object fields.
   * Start the import process and verify the records in Salesforce.





**Task 3: Create Roles**

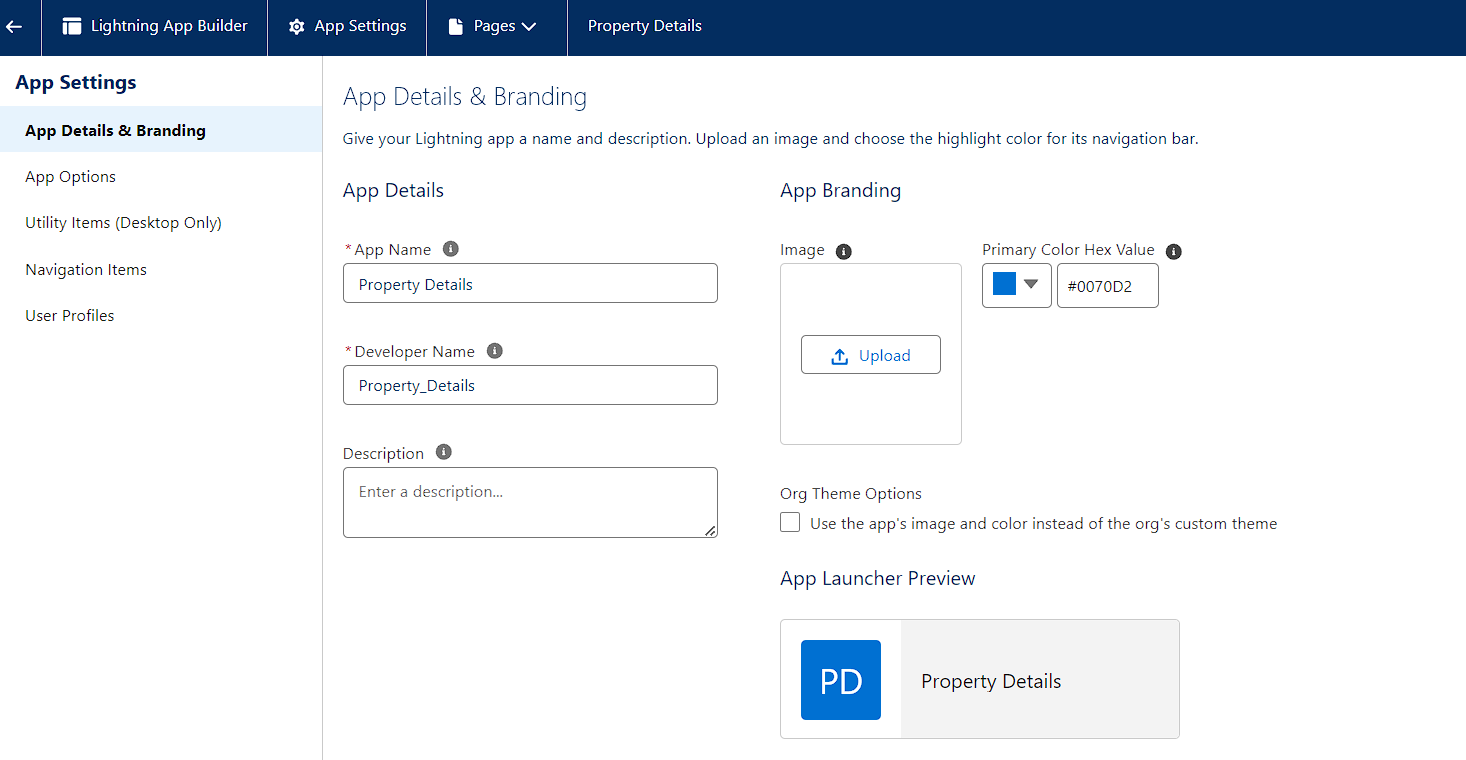
1. **Define Roles:**
   * Go to Salesforce Setup.
   * Navigate to "Roles" under "Users".
   * Create new roles such as Sales Manager, Property Agent, and Customer Support.
   * Assign users to the respective roles.



**Task 4: Create a Property Details App**

**Build the App:**

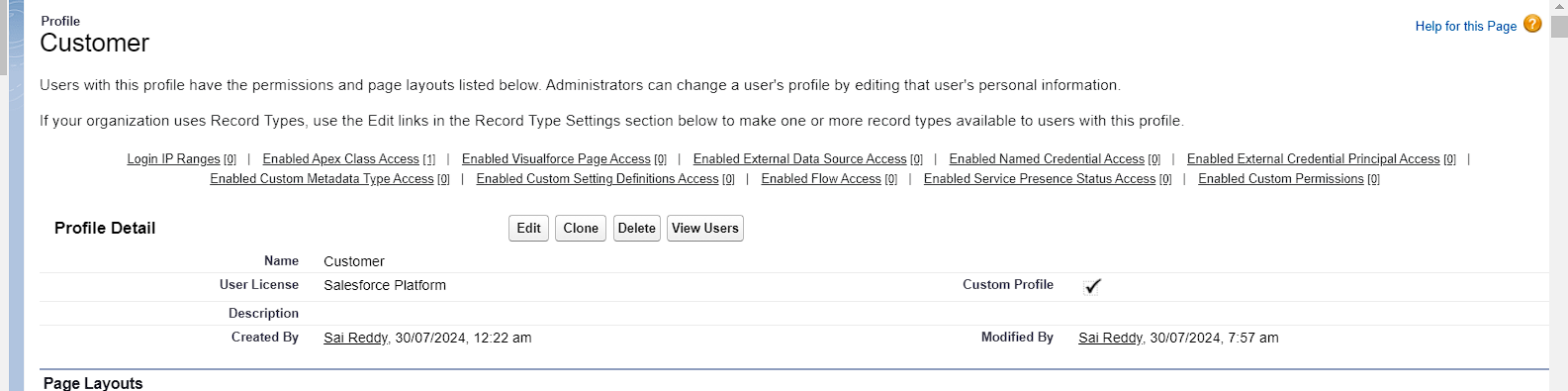
* + Navigate to the App Manager.
  + Create a new app and include the property object and relevant tabs.



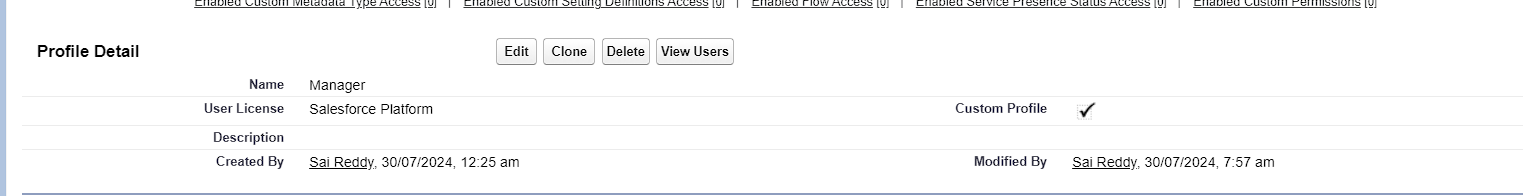
**Task 5: Create Profiles**

1. **Set Up Profiles:**
   * Go to Salesforce Setup.
   * Navigate to "Profiles".
   * Create custom profiles for different user types such as Admin, Sales, and Property Manager.
   * Configure field-level security and permissions for each profile.

CUSTOMER: -



MANAGER: -



**Task 6: Create a Checkbox Field on User**

1. **Add Checkbox Field:**
   * Go to Salesforce Setup.
   * Navigate to "Object Manager" and select the User object.
   * Add a custom checkbox field named "Active".

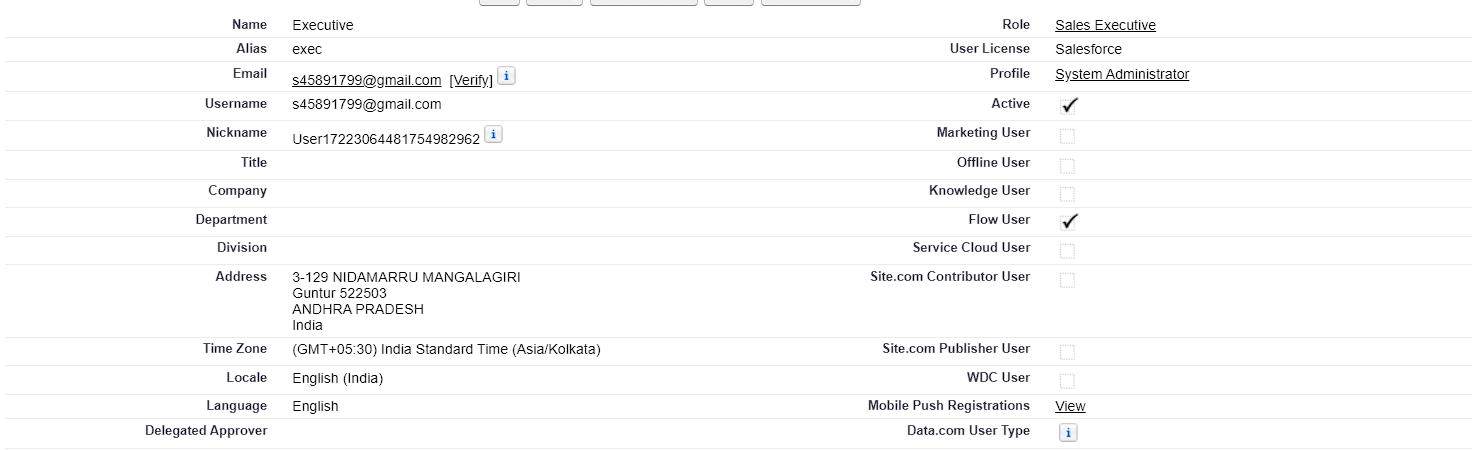
Checkbox Field Creation: -



**Task 7: Create Users**

1. **Add New Users:**
   * Go to Salesforce Setup.
   * Navigate to "Users" under "Administration".
   * Create new users and assign them to the appropriate profiles and roles.

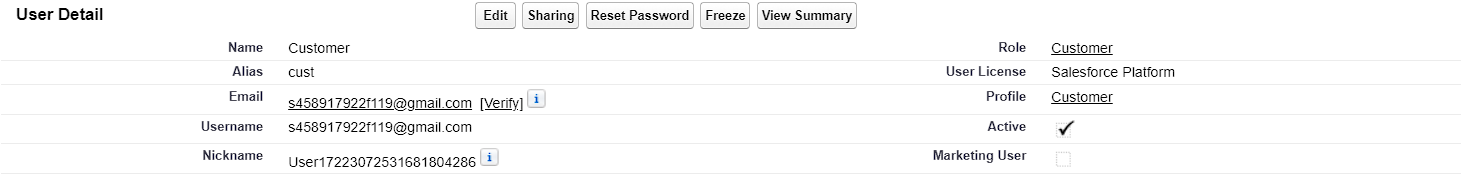
User1: -



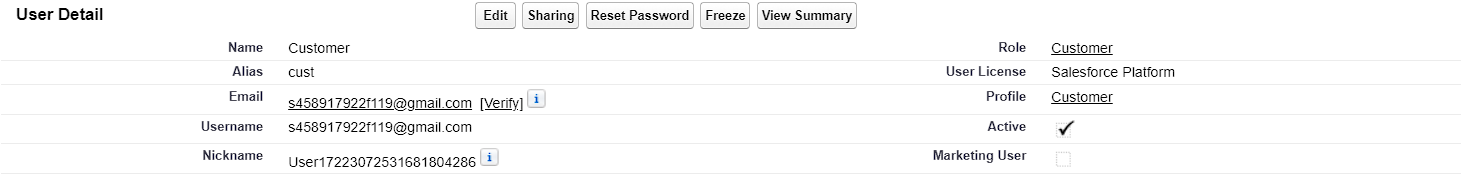
USER2: -



USER3: -



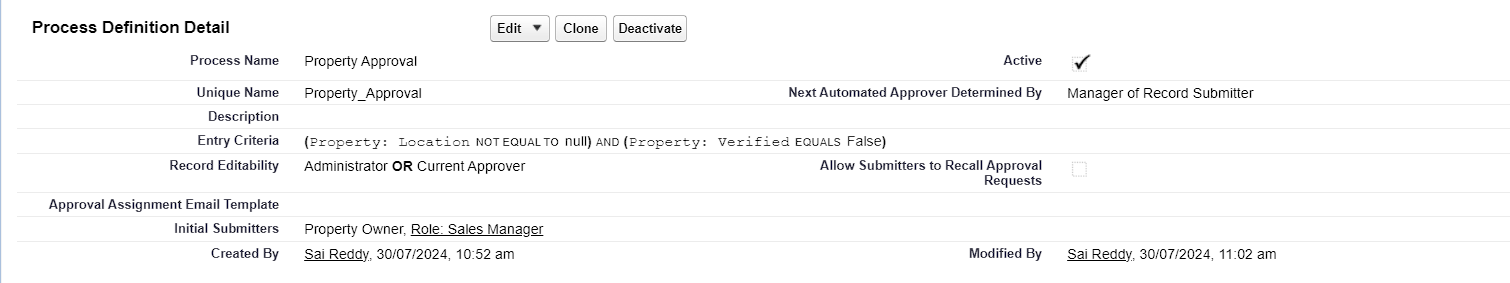
USER4: -

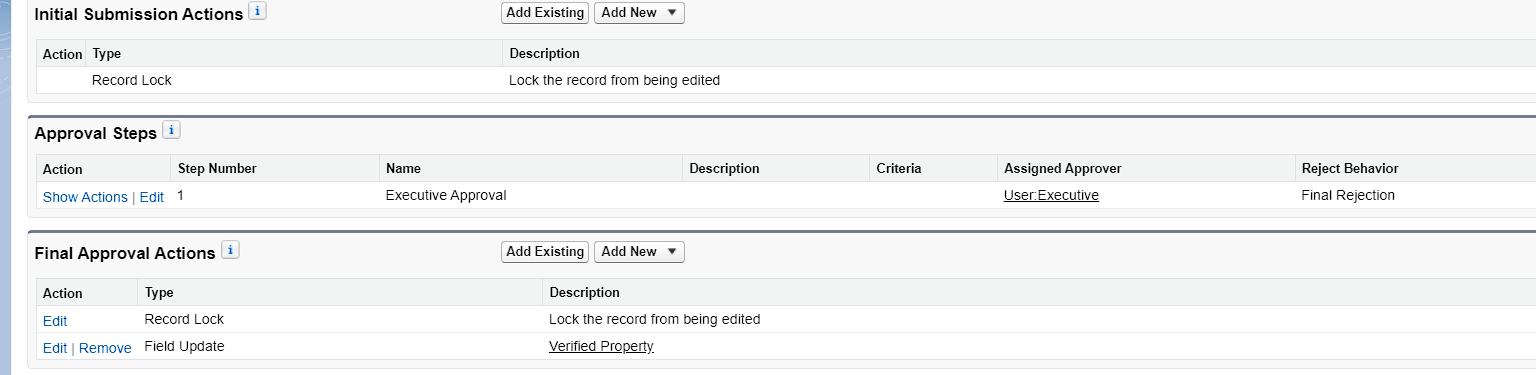


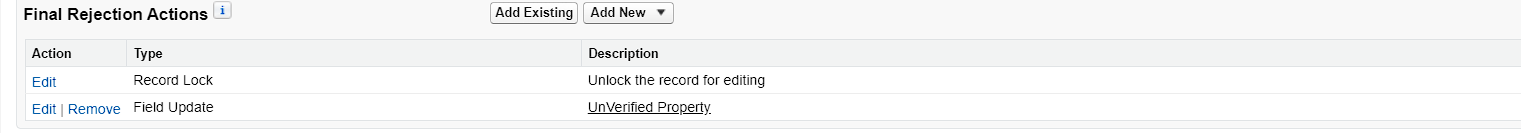
**Task 8: Create an Approval Process for Property Object**

1. **Define Approval Process:**
   * Go to Salesforce Setup.
   * Navigate to "Approval Processes" under "Process Automation".
   * Create a new approval process for the Property object.
   * Define entry criteria, approval steps, and actions.

Approval Process Setup:



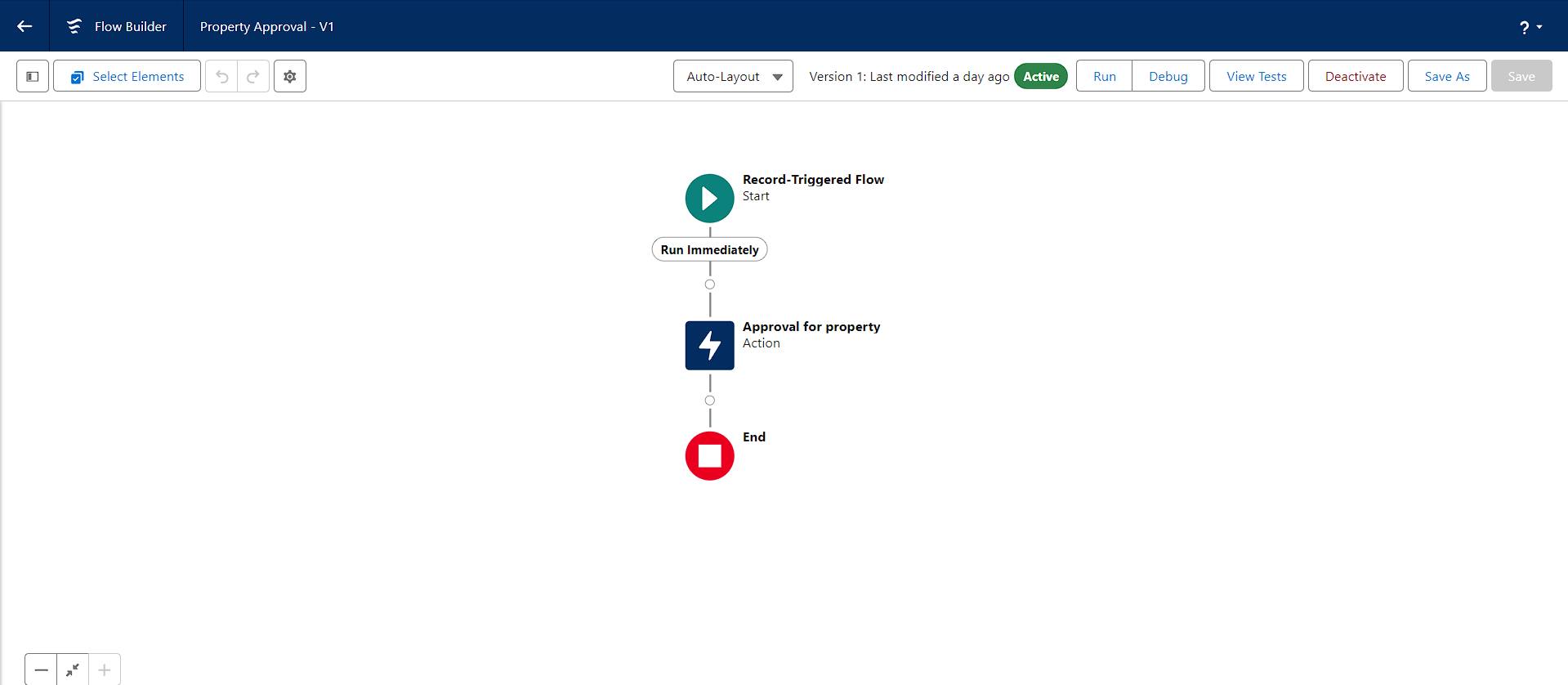




**Task 9: Create a Record Trigger Flow to Submit the Approval Process Automatically**

1. **Set Up Flow:**
   * Go to Salesforce Setup.
   * Navigate to "Flows" under "Process Automation".
   * Create a new record-triggered flow for the Property object.
   * Configure the flow to submit the approval process automatically when certain criteria are met.

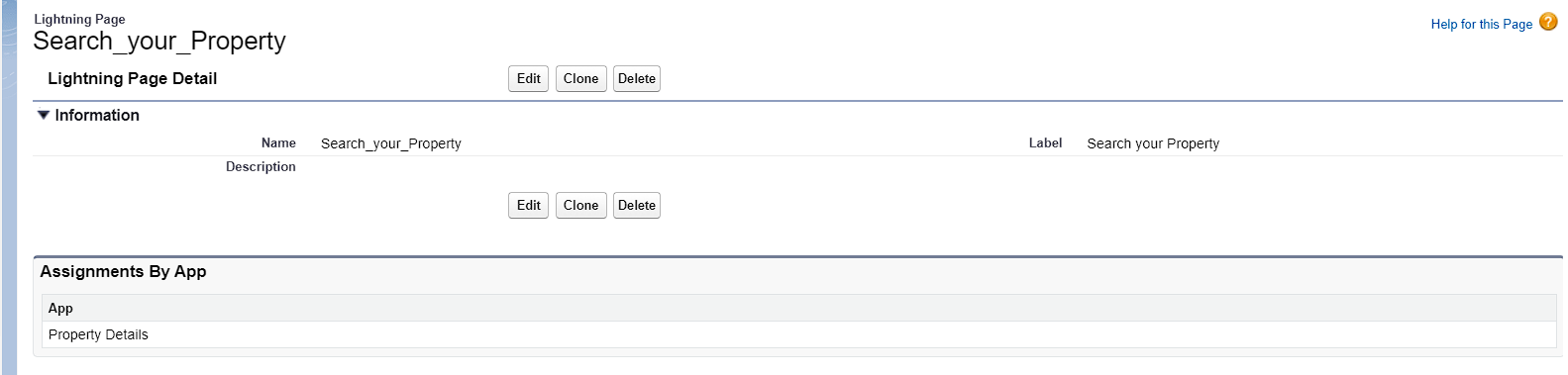
Flow Configuration: -



**Task 10: Create an App Page**

1. **Develop App Page:**
   * Use the Lightning App Builder to create a new app page.
   * Add relevant components and configure the layout.

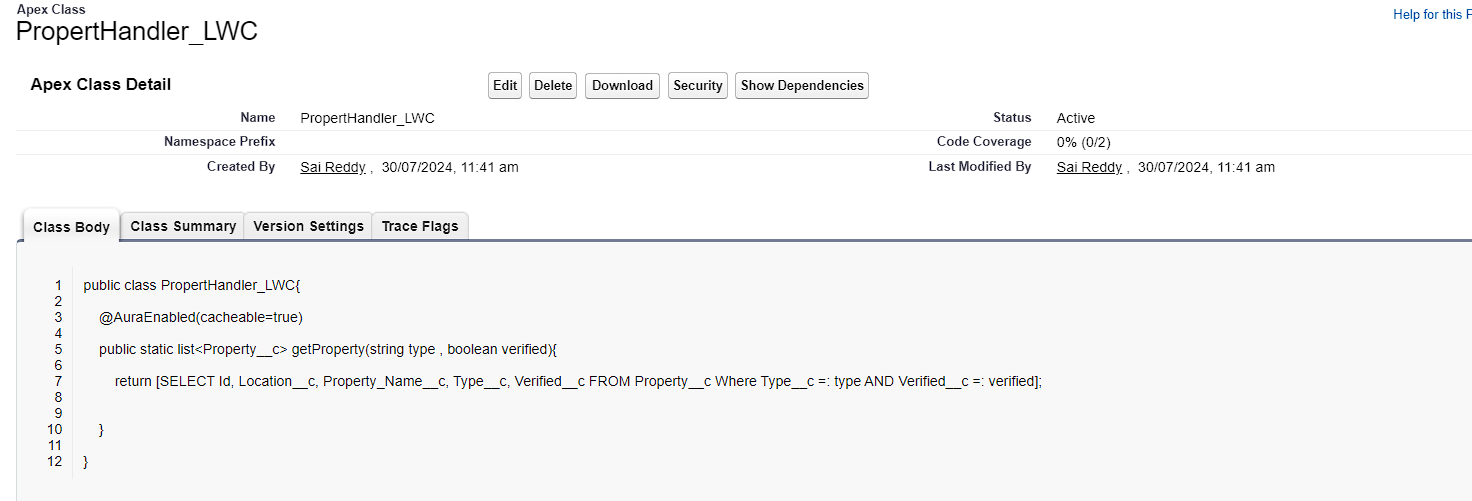
App Page Builder:



**Task 11: Create a LWC Component**

1. **Develop LWC Component:**
   * Use Visual Studio Code and Salesforce CLI to create a new Lightning Web Component.
   * Implement the component logic and design.

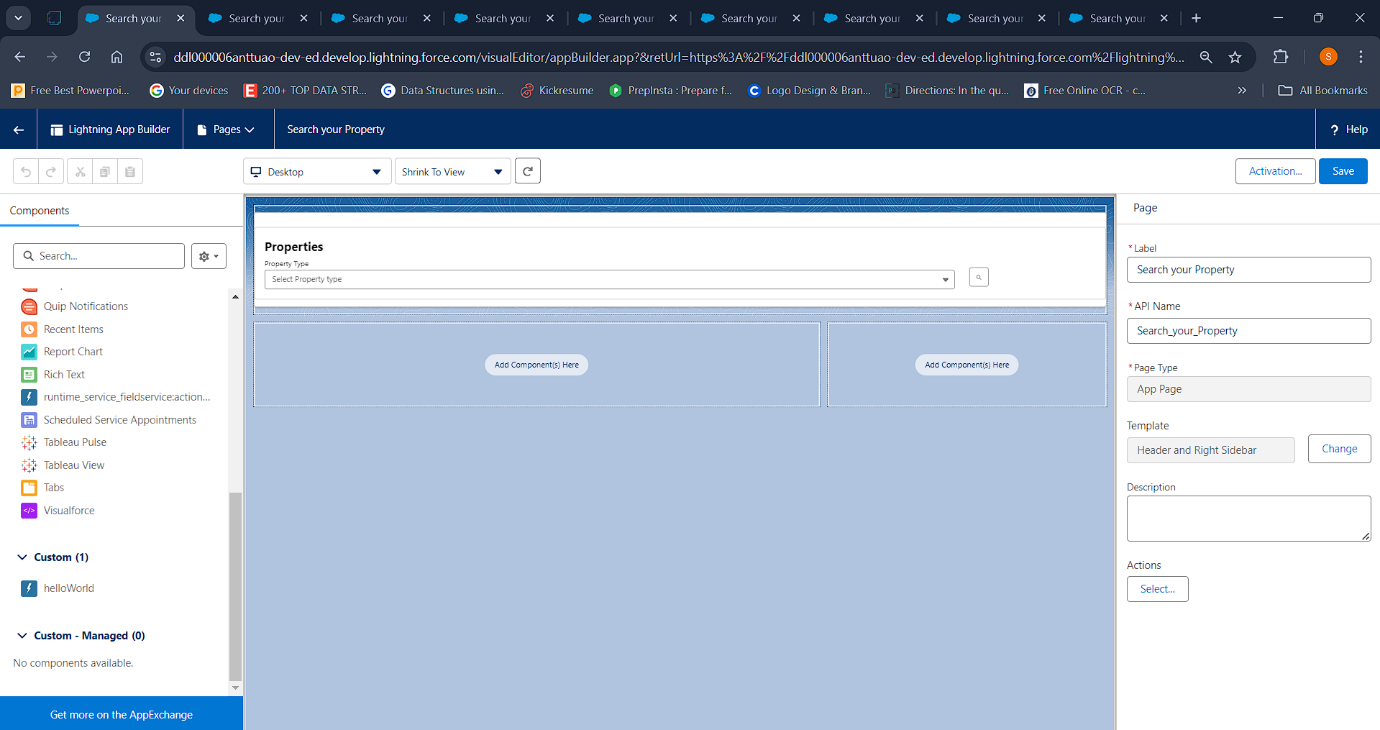
LWC Code:

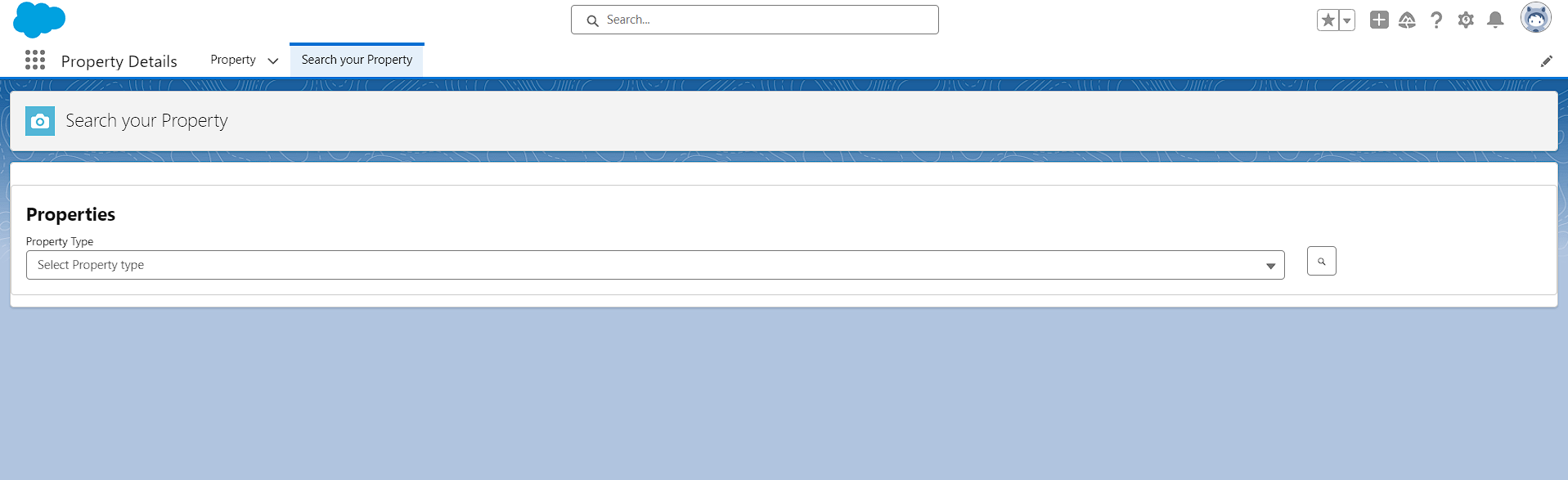


**Task 12: Drag This Component to Your App Page**

1. **Add LWC to App Page:**
   * Go to the Lightning App Builder.
   * Drag and drop the created LWC component onto the app page.

Adding LWC to App Page:

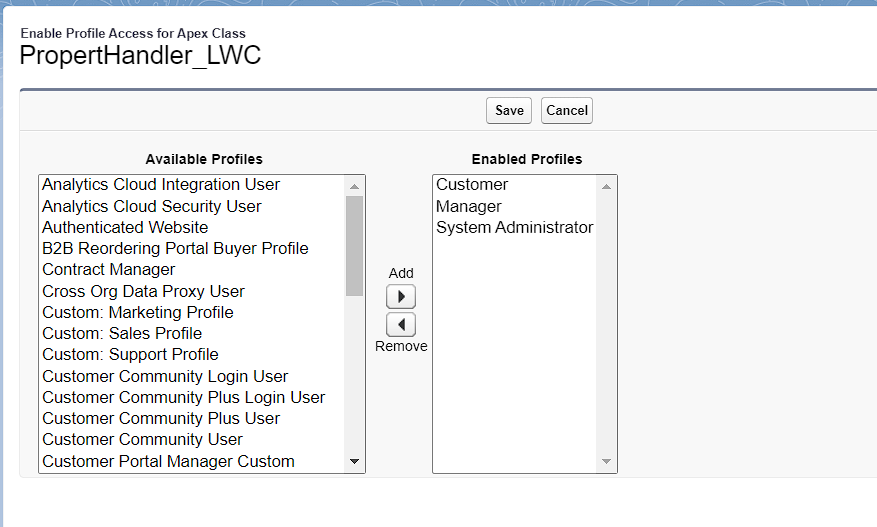




**Task 13: Give Access of Apex Classes to Profiles**

1. **Assign Apex Class Access:**
   * Go to Salesforce Setup.
   * Navigate to "Profiles".
   * Edit the profiles to include access to the necessary Apex classes.

Apex Class Access:



THANK YOU