

SALESFORCE PROJECT

PROJECT TITLE :RealEstate CRM (Property & Agent Management)

PROBLEM: Real estate firms struggle with property listings, agent performance, and client leads. Solution

Phase 6: User Interface Development

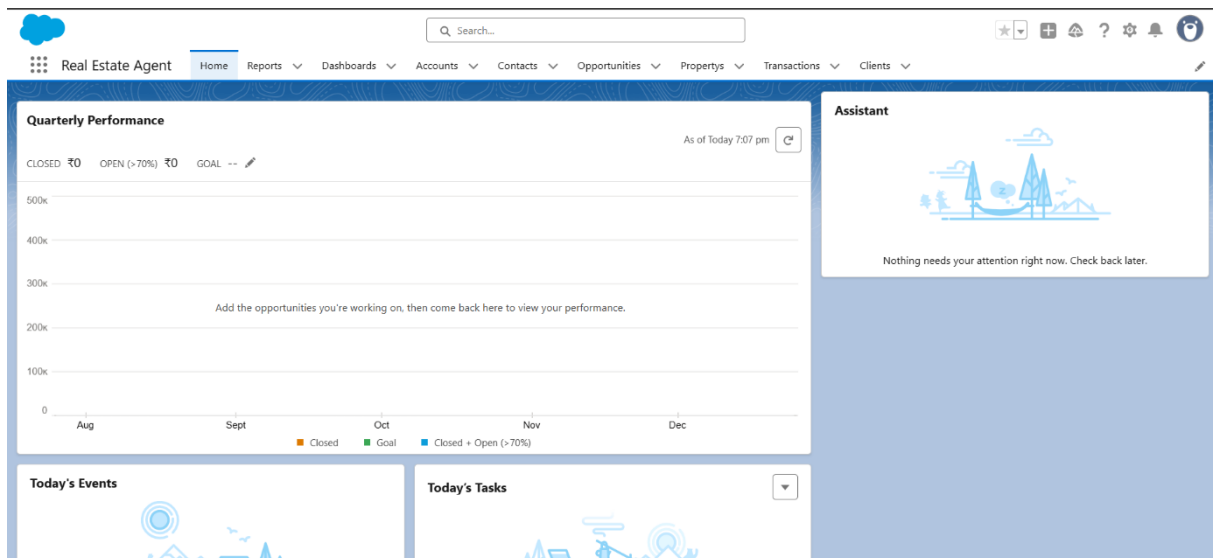
A. Lightning App Builder

A. Lightning App Builder

Goal: Build custom Lightning Apps for internal users and portal users.

Internal Sales App

- App Name: Real Estate CRM
- Components:
 - Home: Dashboard component (pipeline, top agents, top properties)
 - Navigation: Tabs for Properties, Deals, Clients, Contracts, Visits
- Branding: Company logo, theme color



B. Record Pages

Use **Lightning Record Pages** to design pages per object.

1. Property__c Record Page

- Sections:
 - Property Details
 - Owner / Agent Info
 - Related Deals
 - Related Visits
 - Google Map component (LWC or Lightning Component)
- Tabs:
 - Details, Related, Map
- Highlights Panel: Name, Price, Status, Bedrooms

2. Deal__c Record Page

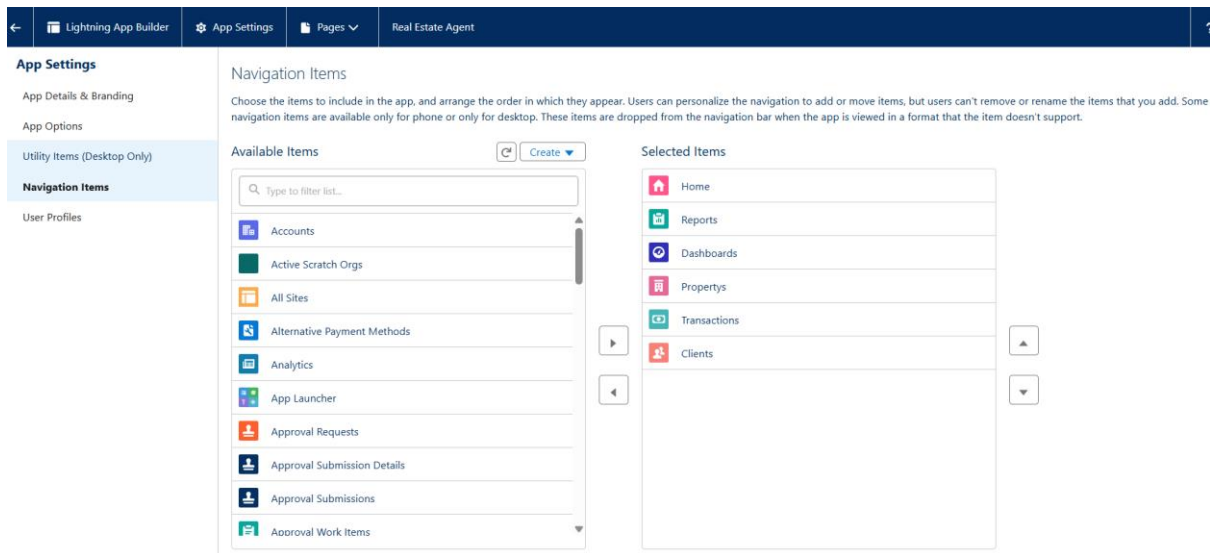
- Sections:
 - Deal Details (Amount, Stage, Probability)
 - Linked Client and Property
 - Related Contract
- Highlights Panel: Name, Stage, Close Date, Amount

3. Client__c Record Page

- Sections:
 - Contact Info
 - Interested Properties
 - Assigned Agent
 - Related Deals
- Highlights Panel: Name, Email, Status

C. Tabs

- ☐ **Internal App Tabs:** Properties, Deals, Clients, Contracts, Visits, Dashboards
- ☐ **Portal Tabs:** Property Search, My Leads, Contact Support



D. Home Page Layouts

- **Internal Home Page**
 - Components:
 - Dashboard (Sales Pipeline)
 - Top Properties by Views
 - My Upcoming Visits
 - Recent Deals
 - Personalized for roles: Agents see only their data; Managers see team data.

E. Utility Bar

Add Utility Bar components for internal app:

- Quick Search (Properties & Clients)
- Today's Tasks / Visits
- Notifications
- Global Create (New Deal, New Visit)

F. Lightning Web Components (LWC)

Core Components:

1. **PropertySearch**

- Inputs: City, Price Range, Bedrooms, Property Type
- Output: List of properties (cards)
- Features: Filter, sort, pagination

2. **PropertyCard**

- Shows: Name, Price, Status, Image, Quick link to Property Page
- Click → Navigate to Property__c Record Page

3. **PropertyMap**

- Displays properties on Google Map
- Markers clickable → navigate to Property details

4. **LeadForm**

- Screen flow embedded as LWC or standalone LWC
- Inputs: Name, Email, Phone, Preferred City, Budget
- Action: Creates Client__c + Deal__c

G. Apex with LWC

- **Purpose:** Fetch data or handle actions LWC cannot do declaratively
- Examples:
 1. Search Properties dynamically (SOQL in Apex)
 2. Fetch Agent performance stats for dashboard cards
 3. Trigger Contract creation & send for e-sign

H. Imperative Apex Calls

- For search filters, API integrations, e-signature triggers

```
import searchProperties from '@salesforce/apex/PropertyController.searchProperties';

searchProperties({ city: this.city, minPrice: this.minPrice, maxPrice: this.maxPrice })
  .then(result => { this.properties = result; })
  .catch(error => { console.error(error); });
```

K. Navigation Service

- Use **NavigationMixin** to programmatically navigate:
 - To record pages (Property, Deal, Client)
 - To standard or custom URLs
 - To Experience Cloud pages

```
this[NavigationMixin.Navigate]({
  type: 'standard__recordPage',
  attributes: {
    recordId: this.propertyId,
    objectApiName: 'Property__c',
    actionName: 'view'
  }
});
```