# Smart Property Portal CRM – Salesforce Implementation: Final Documentation

Submitted by: Turaka Lavanya, Pragati Engineering College

Mentor: Annapurna Gurram, TCS Lastmile Project – Salesforce CRM

## 1. Project Overview

The Smart Property Portal CRM is a Salesforce-based customer relationship management system designed to streamline property listings, customer interactions, and sales operations for real estate businesses. It enables seamless coordination between buyers, sellers, and agents by integrating lead management, property booking, and payment tracking modules. The CRM enhances business efficiency, transparency, and decision-making through automation and real-time analytics dashboards.

### Objectives

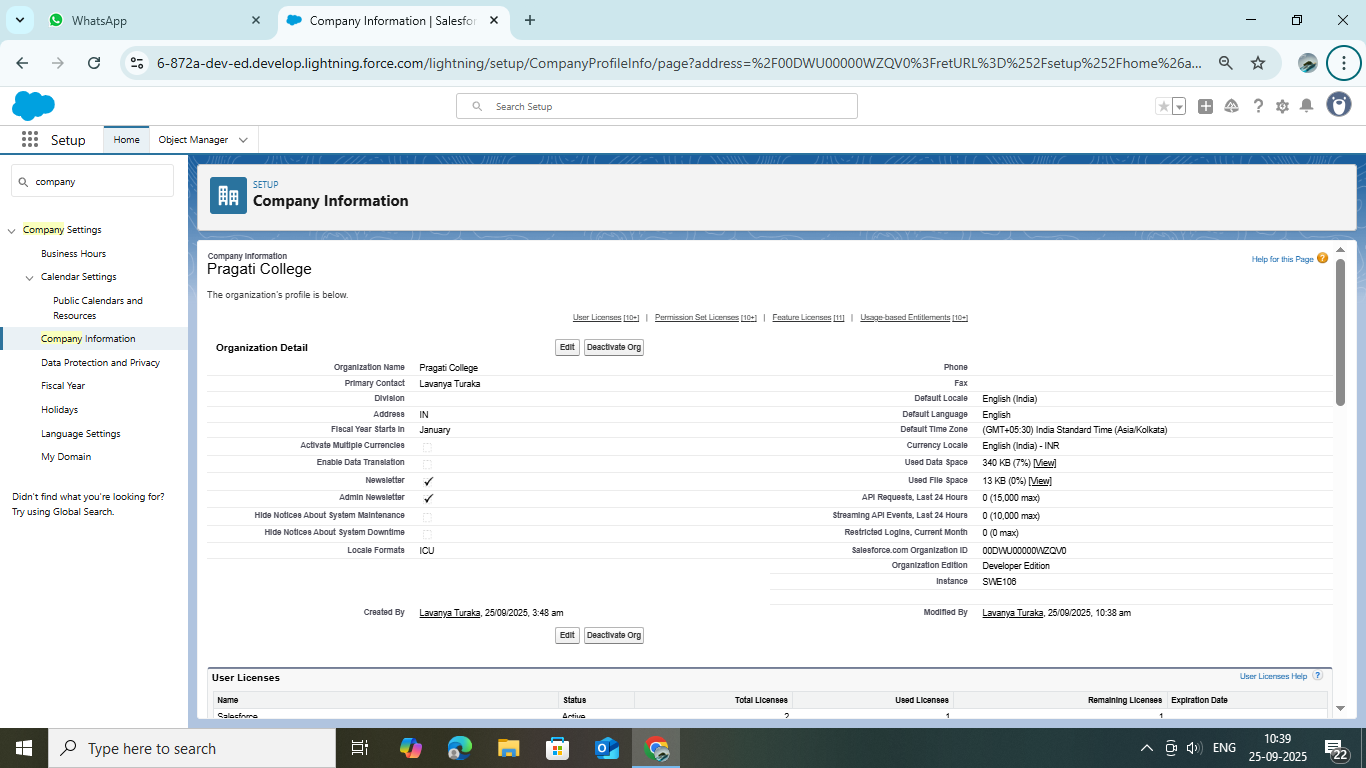
The main objectives of this CRM project are to enhance property management efficiency, automate repetitive tasks, and improve customer satisfaction. It provides real-time insights into sales performance, automates approval processes for property booking, and ensures accurate data management for better business decisions. The system aims to reduce manual work, increase response speed, and provide a user-friendly platform for agents and administrators.

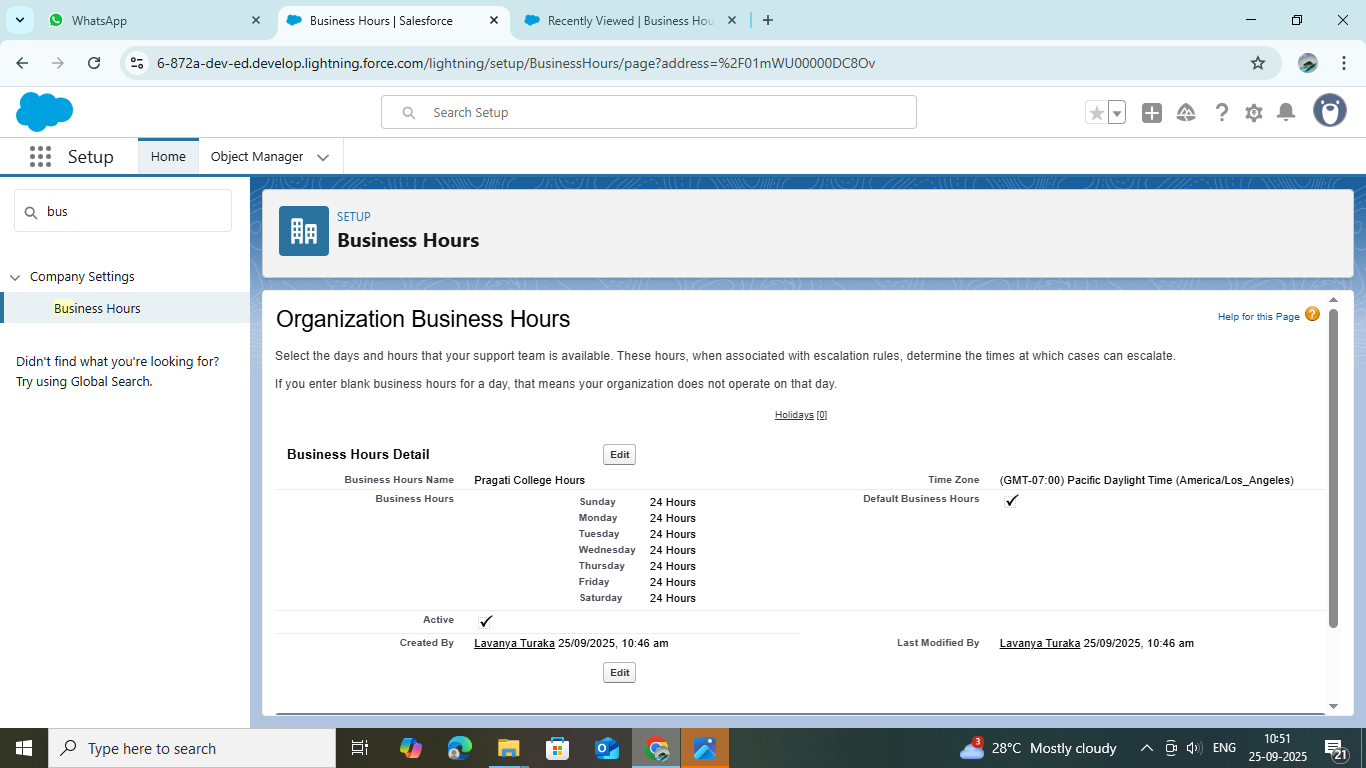
## 2. Project Implementation Phases

### Phase 1: Problem Understanding & Industry Analysis

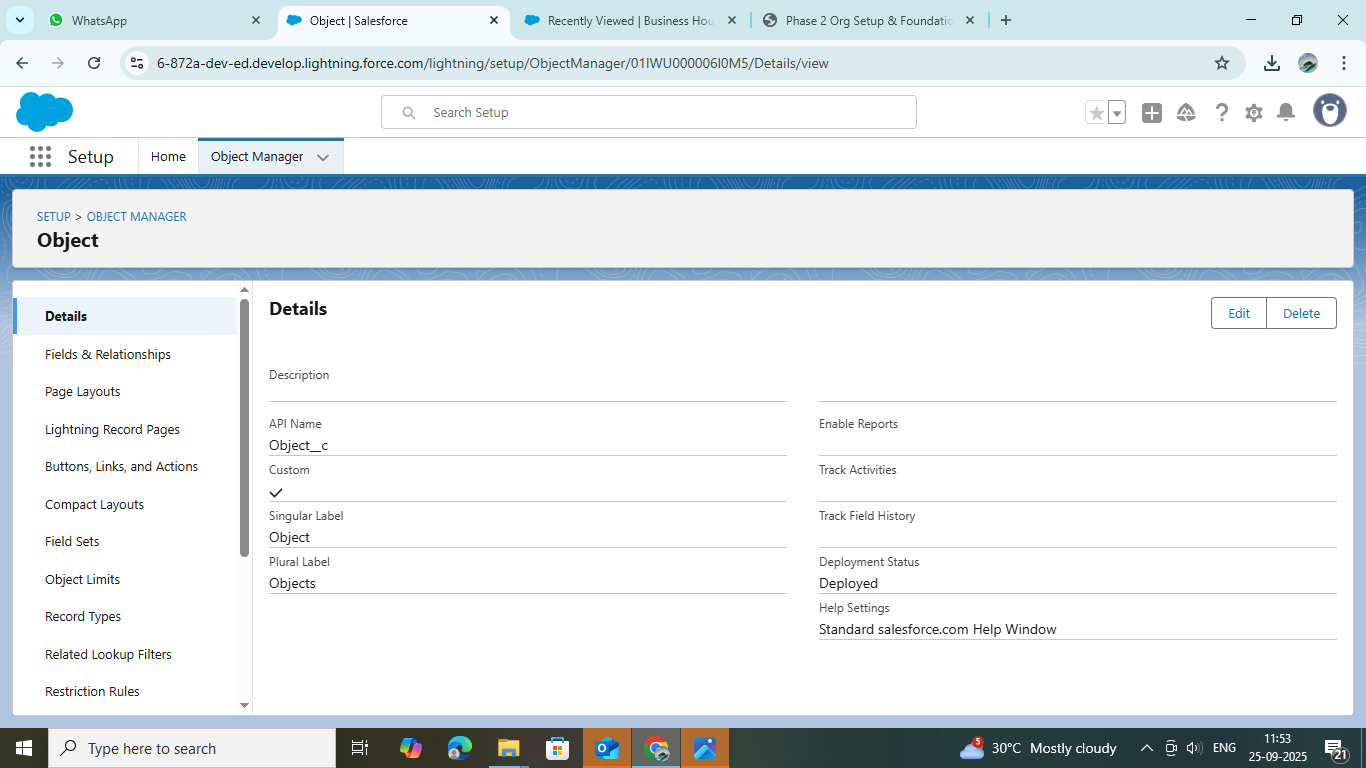
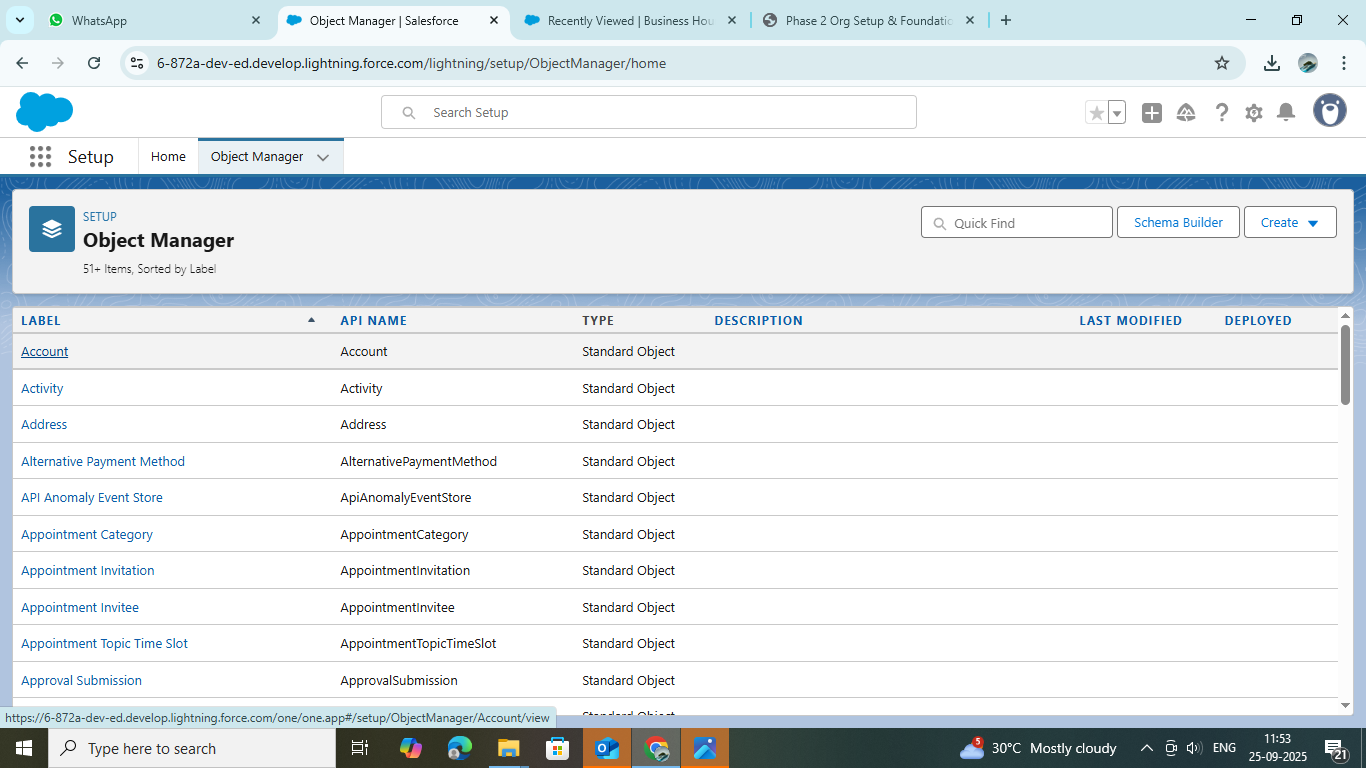
* **Requirement Gathering:** Collected requirements from real estate agencies and property managers regarding lead tracking, payment management, and booking confirmation.
* **Stakeholder Analysis:** Identified key stakeholders including buyers, sellers, agents, and admins.
* **Business Process Mapping:** Designed workflow diagrams showing customer inquiry, booking, and deal closure.
* **Industry-specific Use Case Analysis:** Studied the property management industry trends and pain points to align the CRM solution.
* **AppExchange Exploration:** Explored Salesforce AppExchange apps for property and booking management to compare features.

### Phase 2: Org Setup & Foundational Configuration

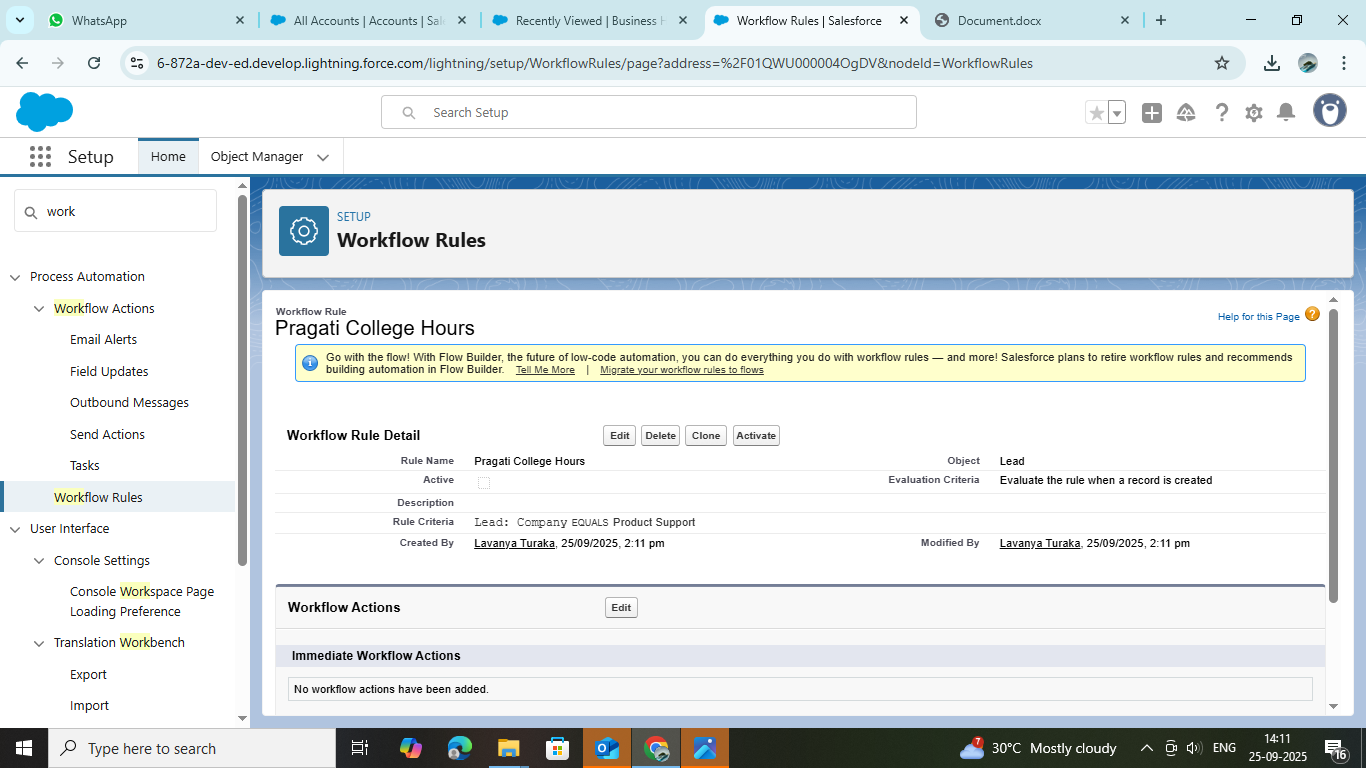
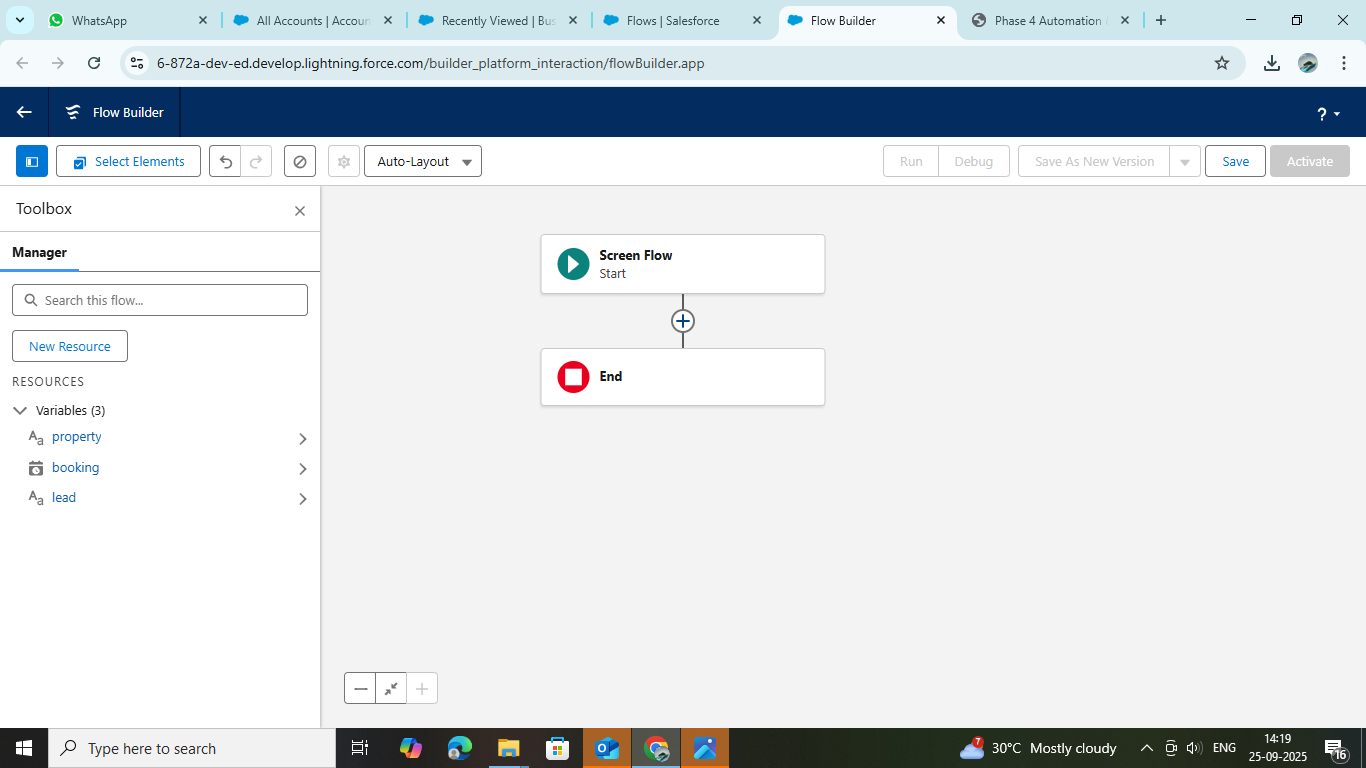
* **Salesforce Edition Used:** Developer Edition.
* **Company Profile Setup:** Configured business hours, fiscal year, and company information for the Smart Property Portal.
* **User Setup & Licenses:** Created Admin, Agent, and Customer Support profiles with required permissions.
* **Roles & Profiles:** Designed a role hierarchy (CEO → Sales Manager → Sales Agent) and created custom profiles.
* **Sharing & Security:** Set object-level security to Private and applied sharing rules for team-based visibility.
* **Sandbox Usage:** Used a Developer Sandbox for testing configurations and automation.
* **Deployment Basics:** Prepared for deployment using Change Sets and version control.



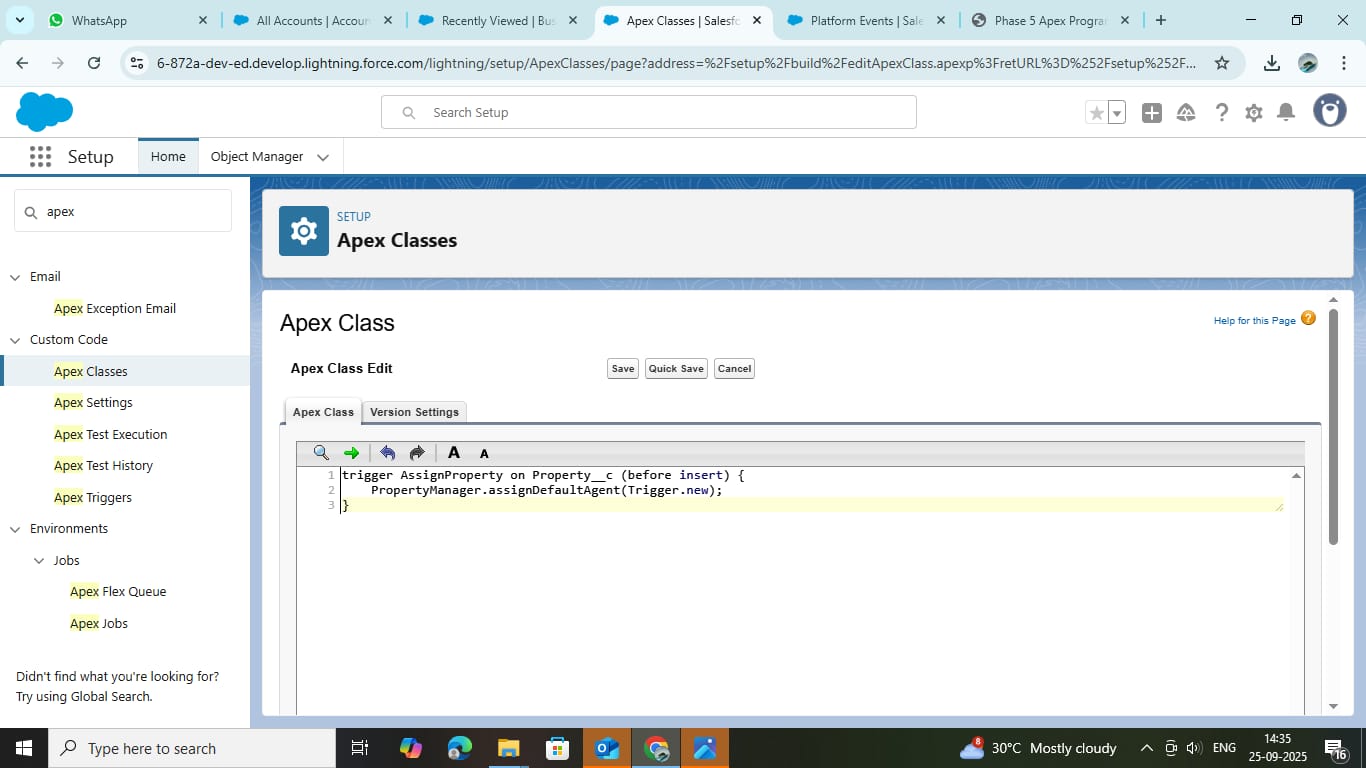
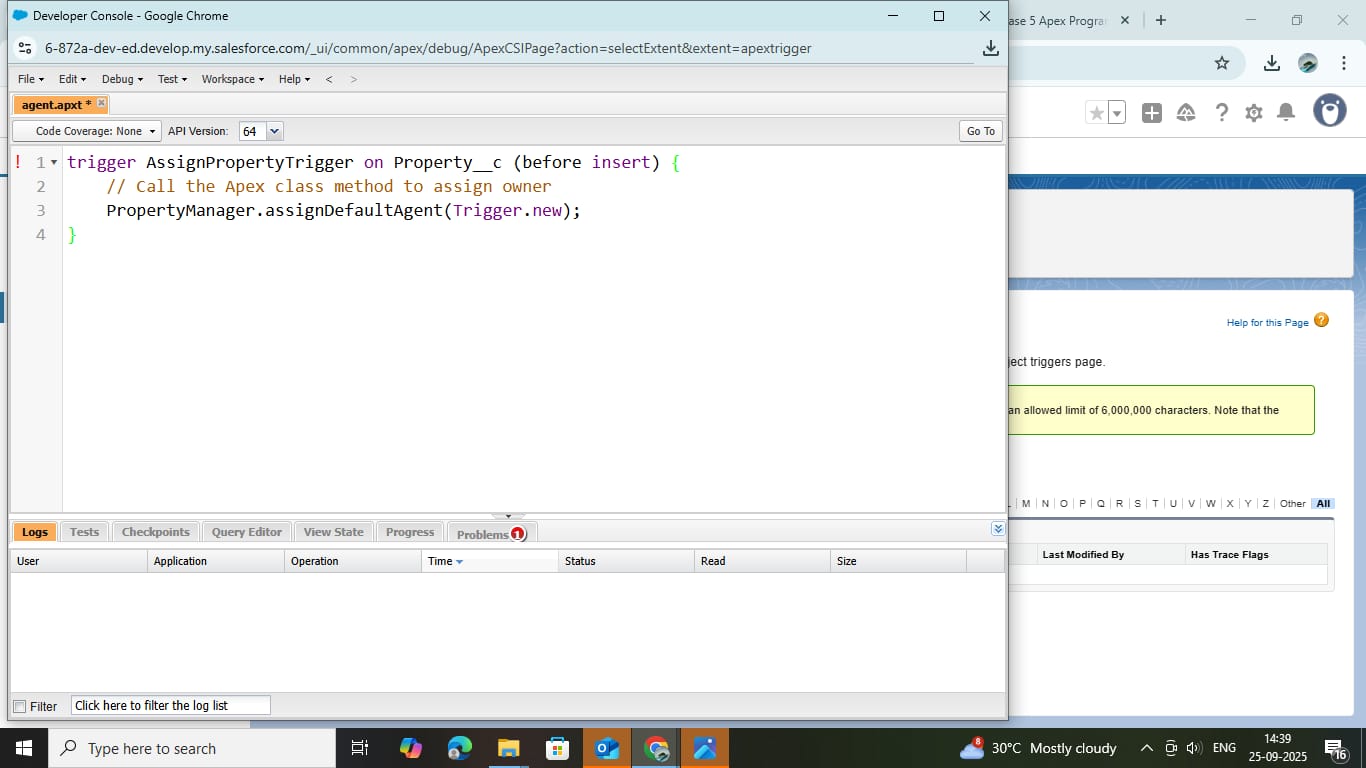
### Phase 3: Data Model & UI Design

* **Standard Objects:** Utilized Account, Contact, and Opportunity for managing customers and deals.
* **Custom Objects:** Created Property\_\_c, Booking\_\_c, and Payment\_\_c to handle real estate data.
* **Relationships:**
  + Master-Detail relationship between Property\_\_c and Booking\_\_c.
  + Lookup relationship between Booking\_\_c and Payment\_\_c.
* **Schema Builder:** Visualized data model relationships.
* **Record Types and Page Layouts:** Created layouts for different property types (e.g., Residential, Commercial).
* **User Interface:**
  + Arranged navigation and tabs for easy access to objects.
  + Created Lightning Record Pages with tabs for Details, Related Lists, and Activities.
  + Tested the user experience for different profiles to ensure usability.

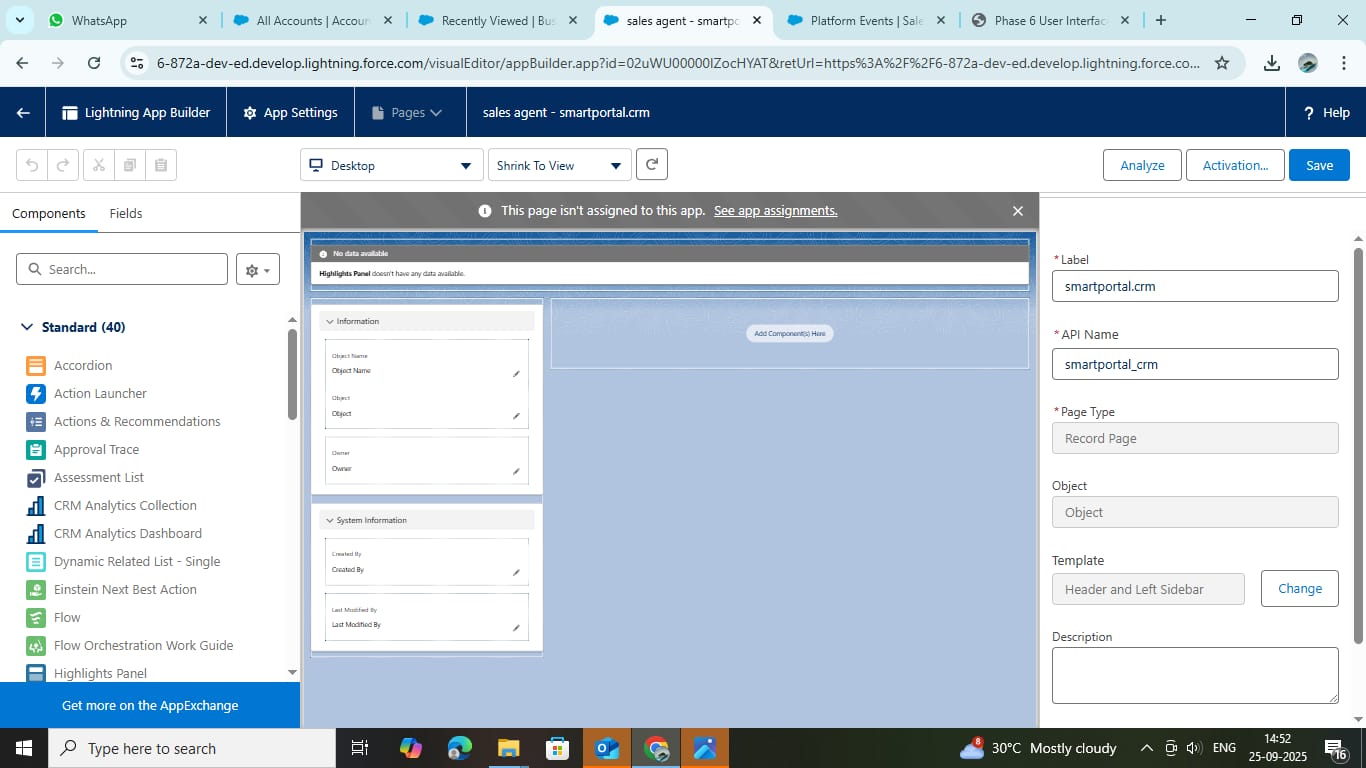
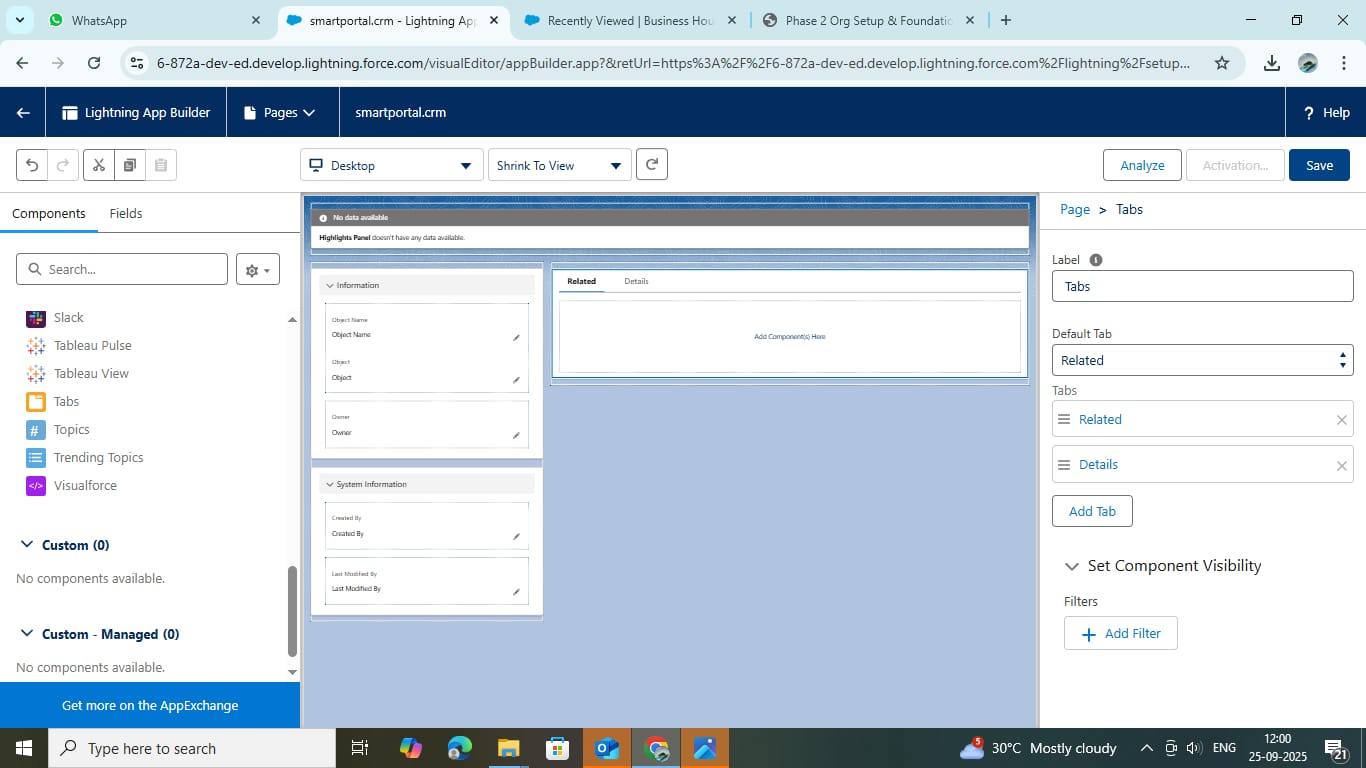
### Phase 4: Process Automation (Admin)

* **Validation Rules:** Ensured mandatory fields like Payment Amount and Booking Date are entered.
* **Workflow Rules:** Automated email alerts to customers upon successful booking.
* **Process Builder:** Updated booking status when a payment is completed.
* **Approval Process:** Implemented for high-value property deals requiring manager approval.
* **Flow Builder:** Automated record creation for follow-up tasks after bookings.
* **Inbound Integrations:** Configured Apex REST endpoints and Platform Events to receive data from external systems.

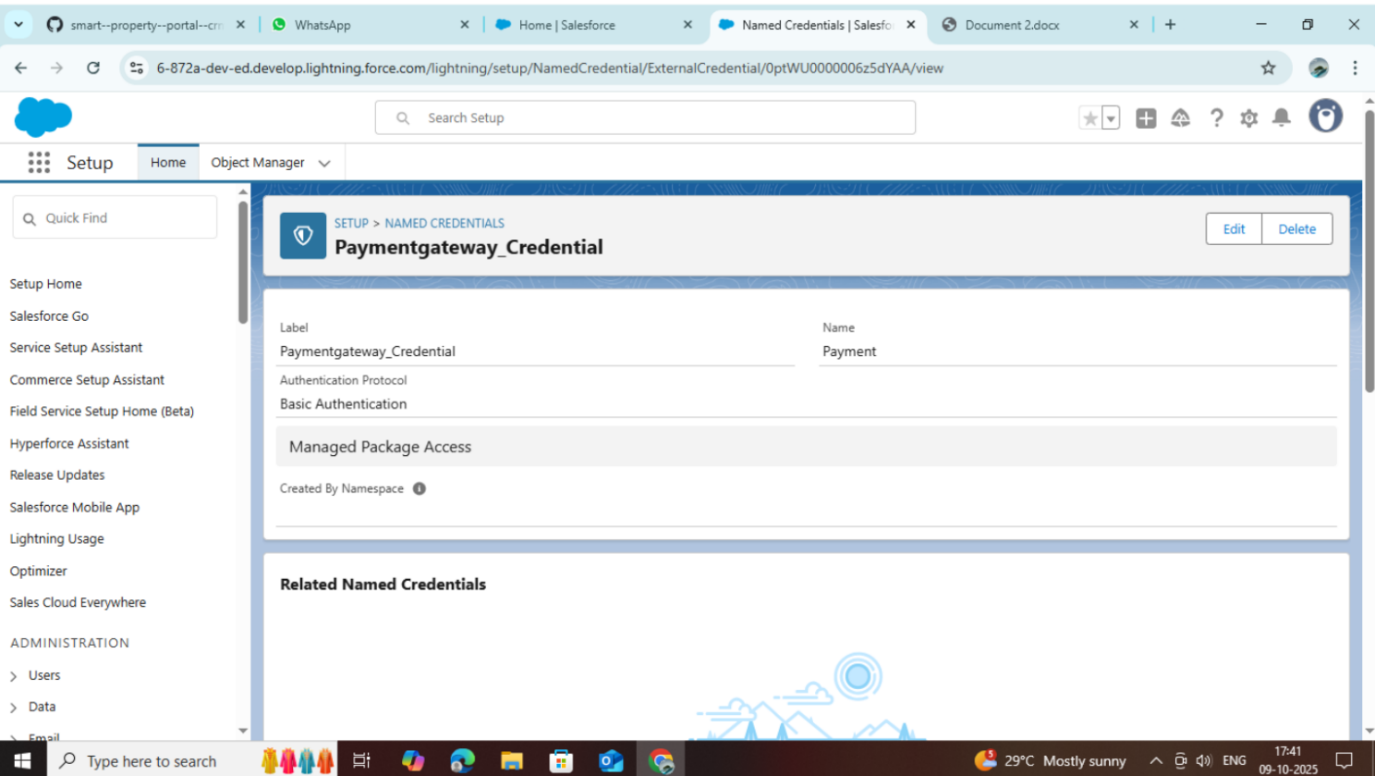
### Phase 5: Apex Programming (Developer)

* **Apex Triggers:** Created before-insert triggers on Booking\_\_c to auto-generate booking numbers.
* **Trigger Design Pattern:** Used a one-trigger-per-object approach with handler classes for maintainability.
* **Batch Apex:** Used for periodic updating of property statuses.
* **Asynchronous Apex:** Implemented future methods for asynchronous callouts for payment confirmation.
* **Exception Handling:** Added try-catch blocks for safer transaction processing.
* **Test Classes:** Wrote test methods to validate trigger logic and ensure code coverage.

### Phase 6: User Interface Development

* **Lightning App Builder:** Designed the Smart Property App with tabs for Properties, Bookings, and Payments.
* **Record Pages:** Customized record pages for an enhanced agent experience.
* **Lightning Web Components (LWC):** Developed a dynamic booking form for customer inputs.
* **Apex with LWC:** Retrieved real-time property availability using SOQL queries in Apex methods called from LWCs.
* **Events in LWC:** Used custom events for communication between parent and child components.
* **Navigation Service:** Enabled smooth page navigation between property records.

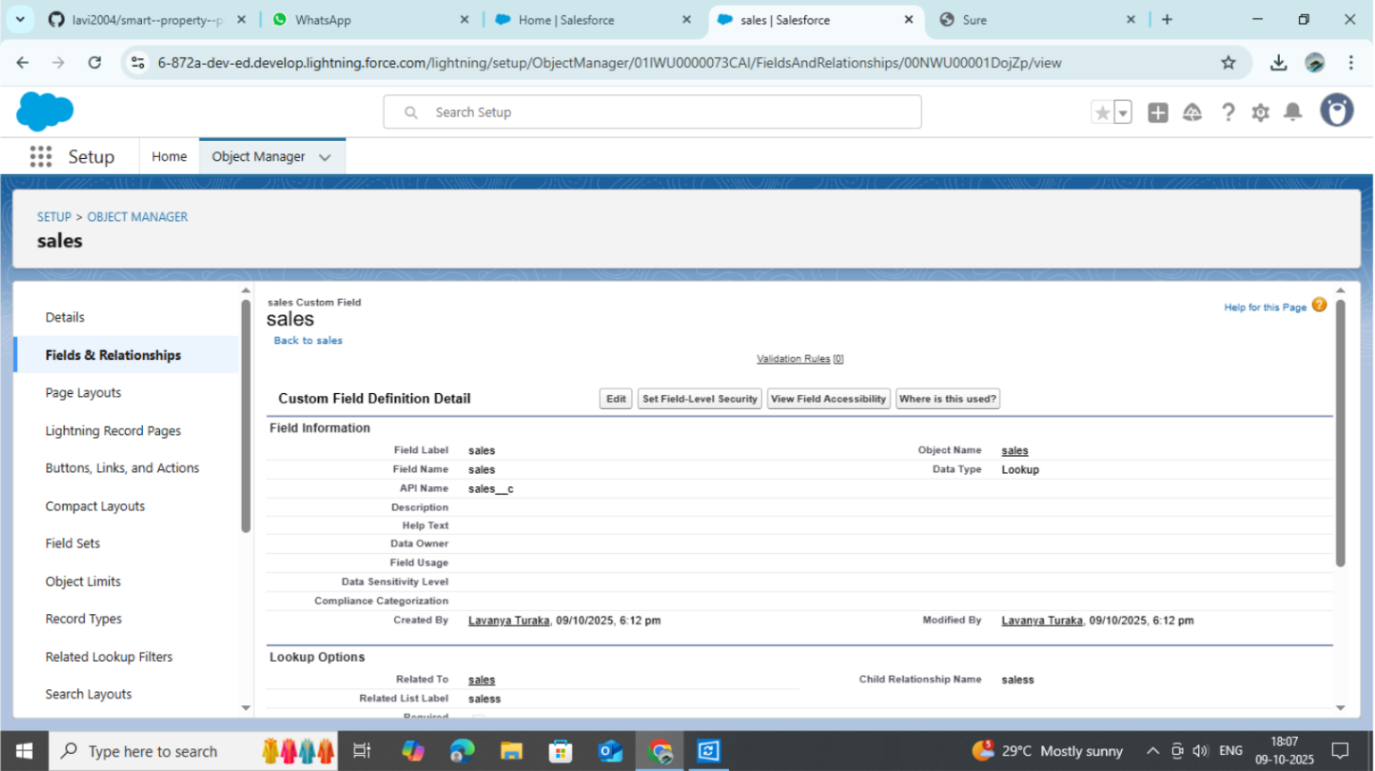
### Phase 7: Integration & External Access

* **Named Credentials:** Configured for external API integration with a payment gateway.
* **REST Callouts:** Used to verify online payments and update payment status.
* A screenshot of a computer

  AI-generated content may be incorrect.**Platform Events:** Implemented notifications for booking approvals.
* **OAuth Authentication:** Ensured secure data access for integrations.
* **Remote Site Settings:** Configured endpoints for third-party services.

### A screenshot of a computer AI-generated content may be incorrect.

### Phase 8: Data Management & Deployment

* **Data Import Wizard:** Imported sample property and client data.
* **Data Loader:** Used for bulk updates of payment records.
* **Duplicate Rules:** Prevented duplicate property listings.

A screenshot of a computer

AI-generated content may be incorrect.

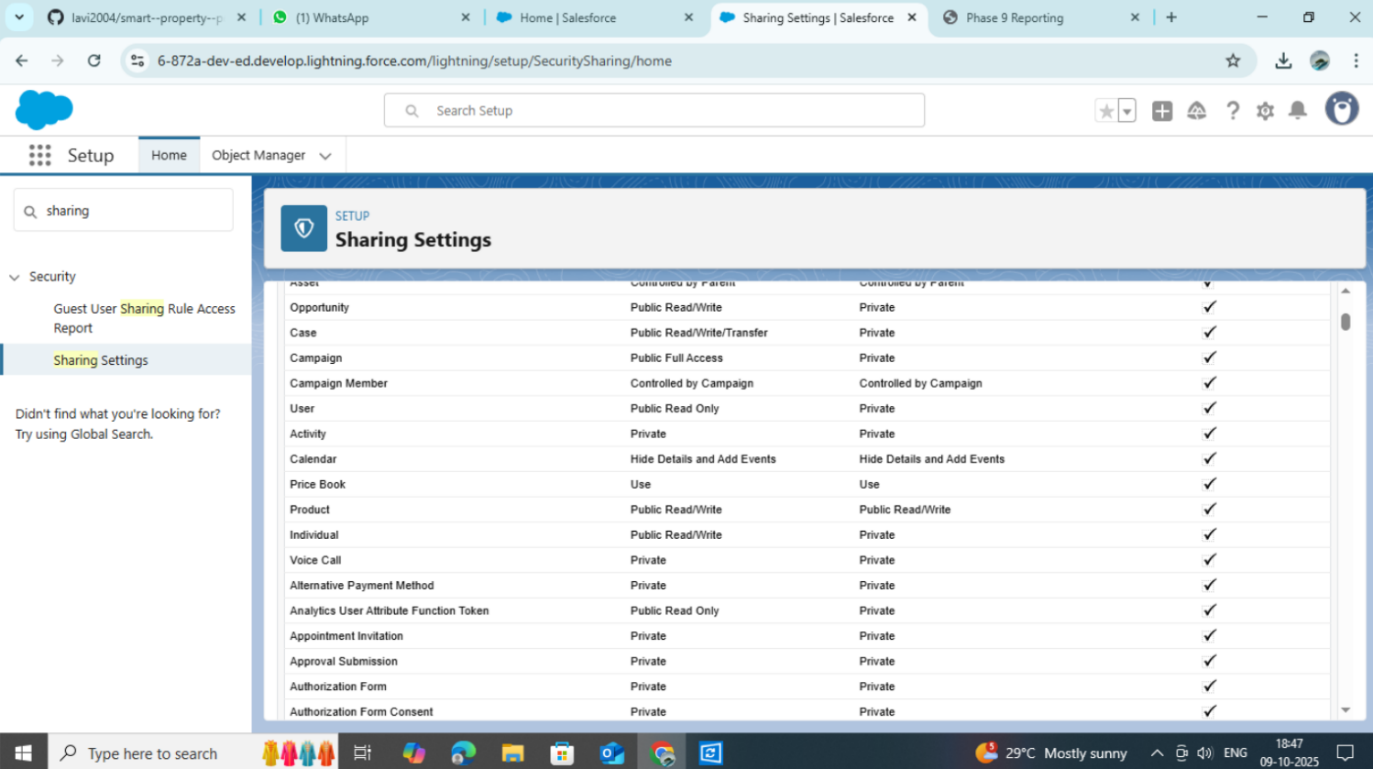
* A screenshot of a computer

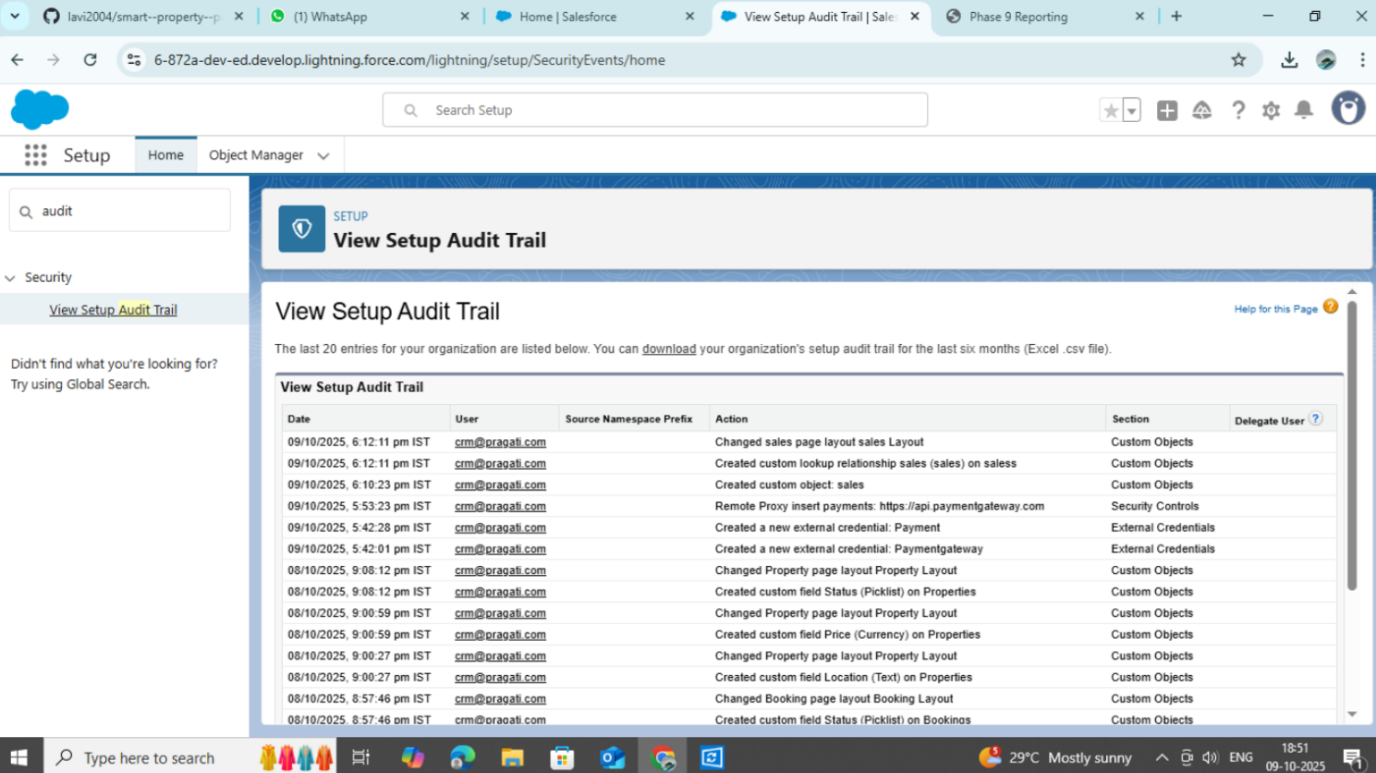
  AI-generated content may be incorrect.**Change Sets:** Used for deployment between different Salesforce environments
* **SFDX:** Utilized Visual Studio Code for version control and metadata deployment.

### Phase 9: Reporting, Dashboards & Security Review

* **Reports:** Created tabular and summary reports for property sales and pending payments.



* **Dashboards:** Built graphical dashboards for a monthly sales overview.
* **Profiles & Roles:** Set access levels for Admin, Agent, and Customer Support users.
* **Sharing Rules:** Defined record-level access control for agents

.

* **Field Level Security:** Restricted sensitive information like payment details.

### Phase 10: Quality Assurance Testing

* Conducted test cases for Booking creation, Payment updates, and Automation flows.
* Verified expected vs. actual outputs using test data.
* Captured screenshots for passed test cases.
* Ensured a smooth data flow between modules and validated trigger logic.
* Resolved minor validation and record duplication issues.
* Final Video: https://drive.google.com/file/d/1gCnB4kcHUJdpAXFcfGmg-2tr8mINYcCZ/view?usp=drivesdk

## 3. Conclusion

The Smart Property Portal CRM successfully demonstrates how Salesforce can be leveraged to digitize and automate real estate operations. From property listing to booking and payment management, the solution ensures efficient data handling, customer engagement, and business analytics. It enhances productivity and provides a reliable framework for future scalability.