**LEASE MANAGEMENT**

SREE NARAYANA GURU COLLEGE,

K. G. CHAVADI, COIMBATORE

**College Code** : bru36

**TEAM ID** : NM2025TMID26347

**TEAM:  
      Team Leader :  
 PRAJAKTHA V** [**praju261005@gmail.com**](mailto:praju261005@gmail.com)

**Team Member :**

**ARYA PU**

[**aryadaibl@gmail.com**](mailto:aryadaibl@gmail.com)

**Team Member :  
 ATHIRA H** [**priyapkd4@gmail.com**](mailto:priyapkd4@gmail.com)

**Team Member :  
 MOHAMMED SAJJAD NS**

[**mohammedsajjad18021@gmail.com**](mailto:mohammedsajjad18021@gmail.com)

**Team Member :**

**SARATH K**

[**ksarath0805@gmail.com**](mailto:ksarath0805@gmail.com)

**INTRODUCTION**

1. **Project Overview**

The Lease Management System leverages Salesforce to simplify and automate real estate leasing operations. From managing tenants and lease contracts to handling payments and communications, it uses tools like flows, approval processes, and email alerts to ensure efficiency and accuracy. The platform also provides real-time visibility into lease performance, financial records, and tenant activities. This makes it easier for organizations to make informed decisions, improve operational efficiency, and deliver seamless leasing services.

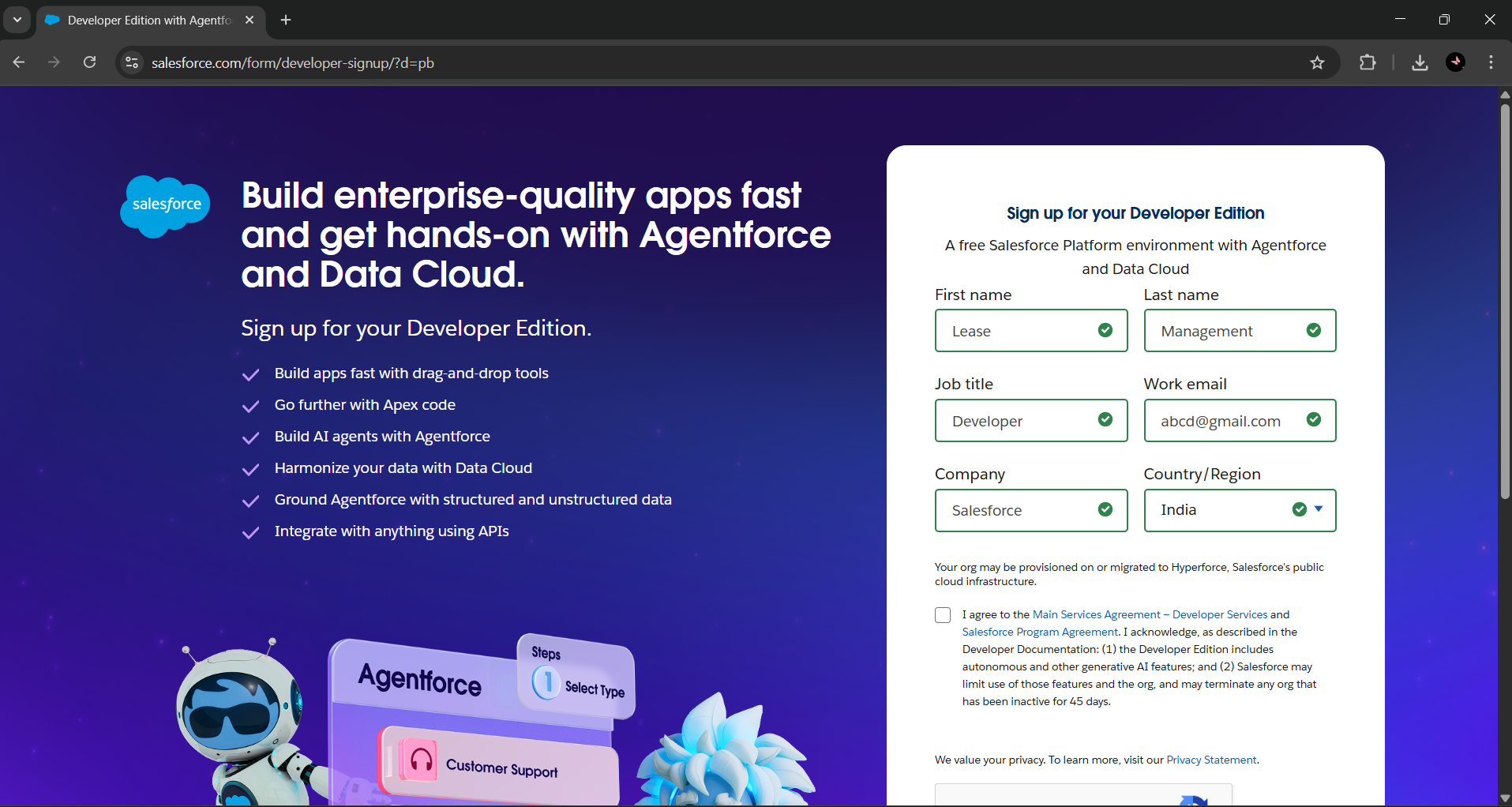
1. **Purpose**

The primary objective of this project is to empower organizations to efficiently manage properties, tenants, and lease-related activities within a unified system. By introducing automation, the solution reduces the need for manual intervention, thereby saving time and minimizing the chances of human error. It improves accuracy across leasing processes, ensuring that financial transactions, contracts, and tenant records remain consistent and reliable. In addition, the system enhances compliance by providing standardized workflows and timely alerts for approvals, renewals, and payments. Communication between property managers and tenants is also streamlined through built-in tools such as notifications and email alerts. Overall, the project is designed to create a seamless, efficient, and transparent leasing process that benefits both organizations and tenants.

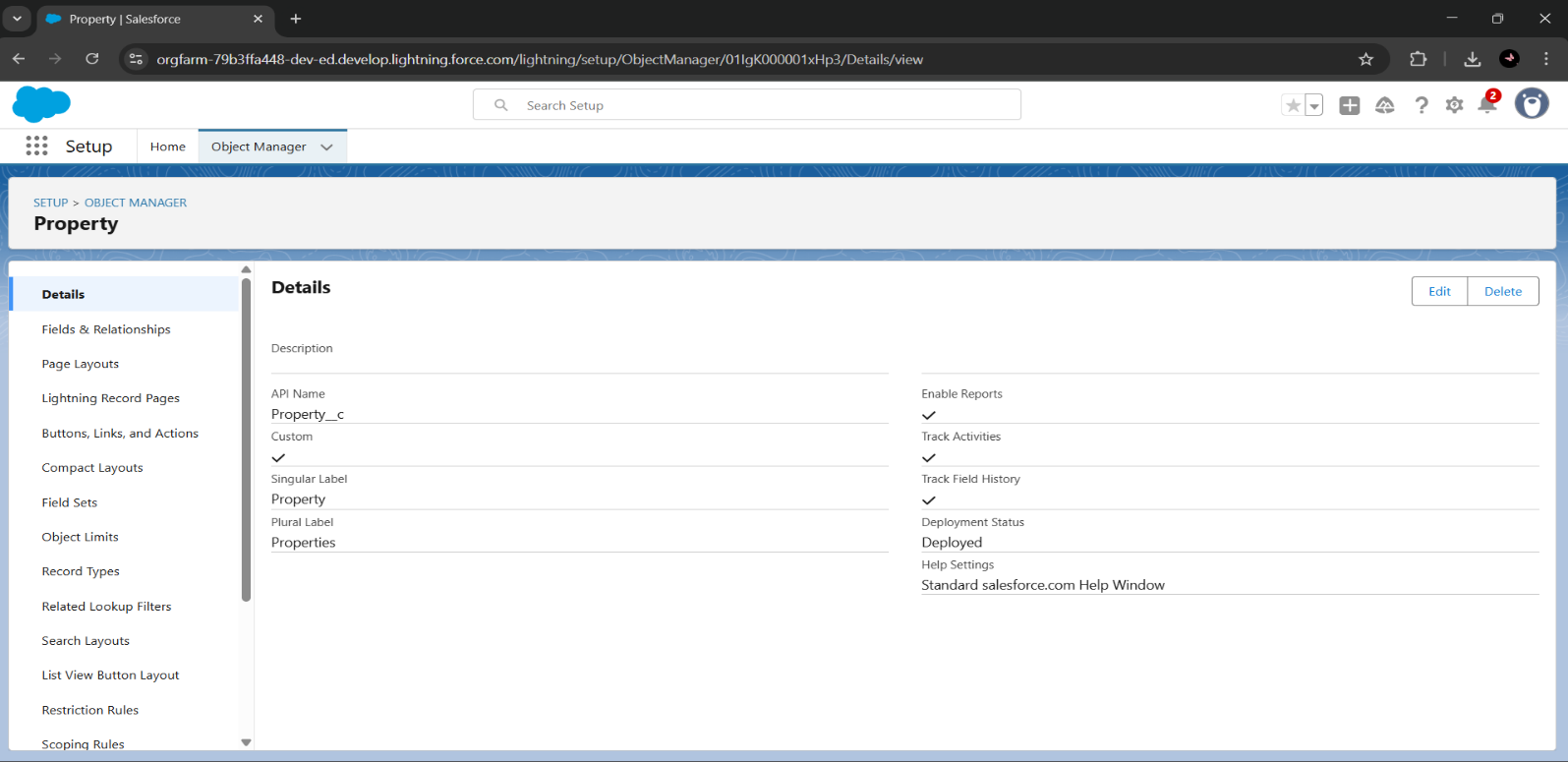
**DEVELOPMENT PHASE**

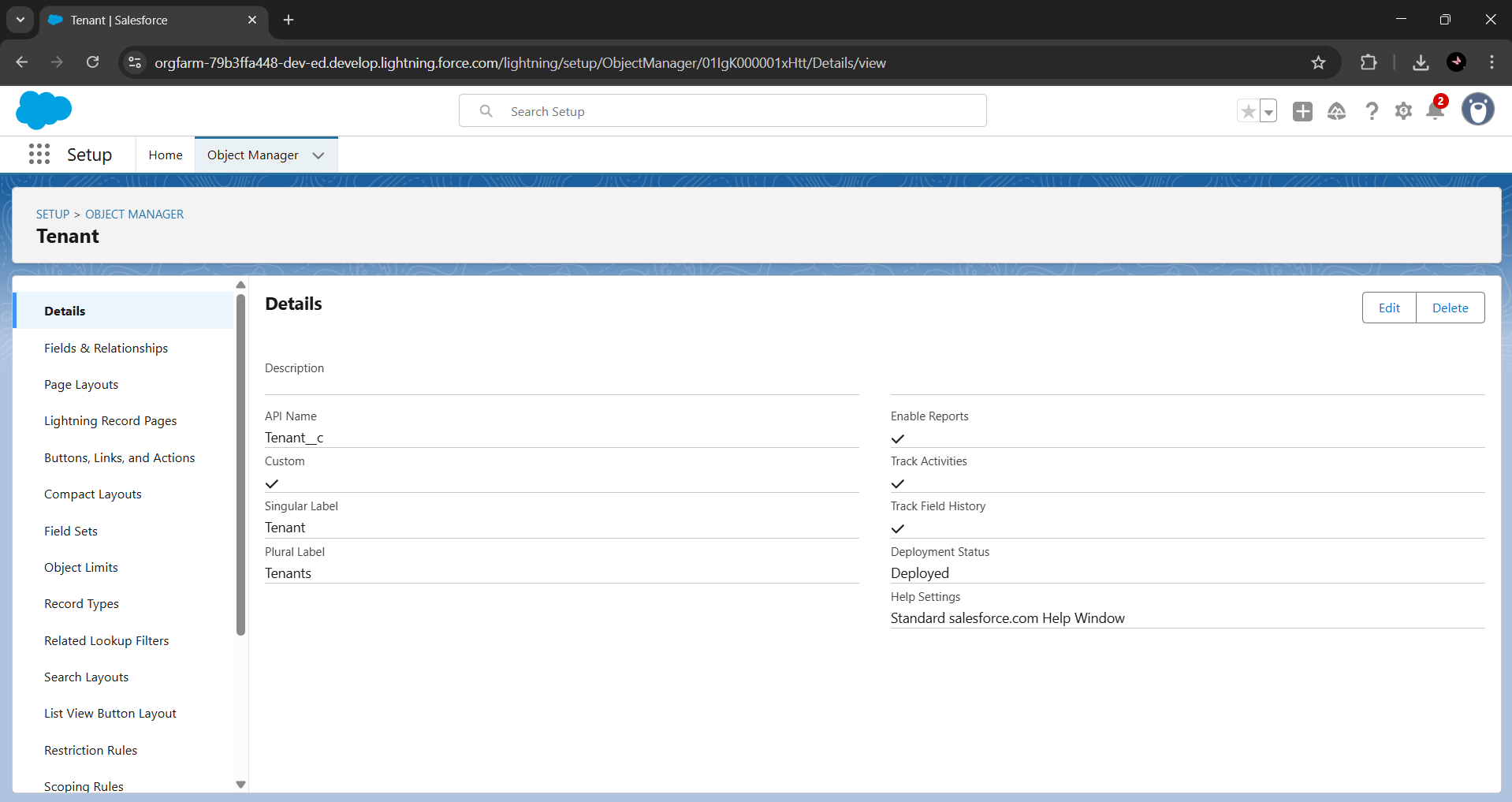
1. **Creating Developer Account:**

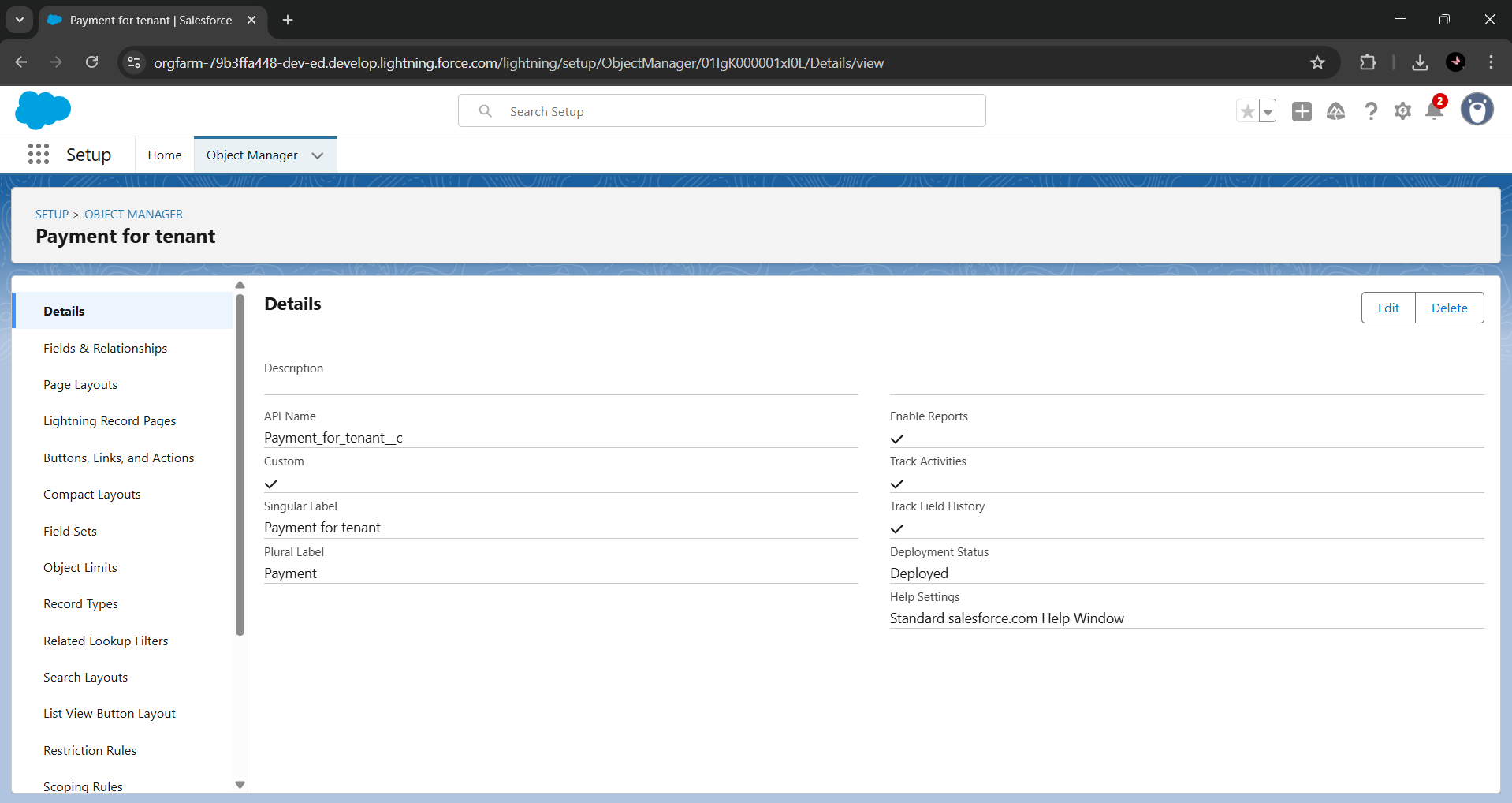
**By using this URL -** [**https://www.salesforce.com/form/developer-signup/?d=pb**](https://www.salesforce.com/form/developer-signup/?d=pb)

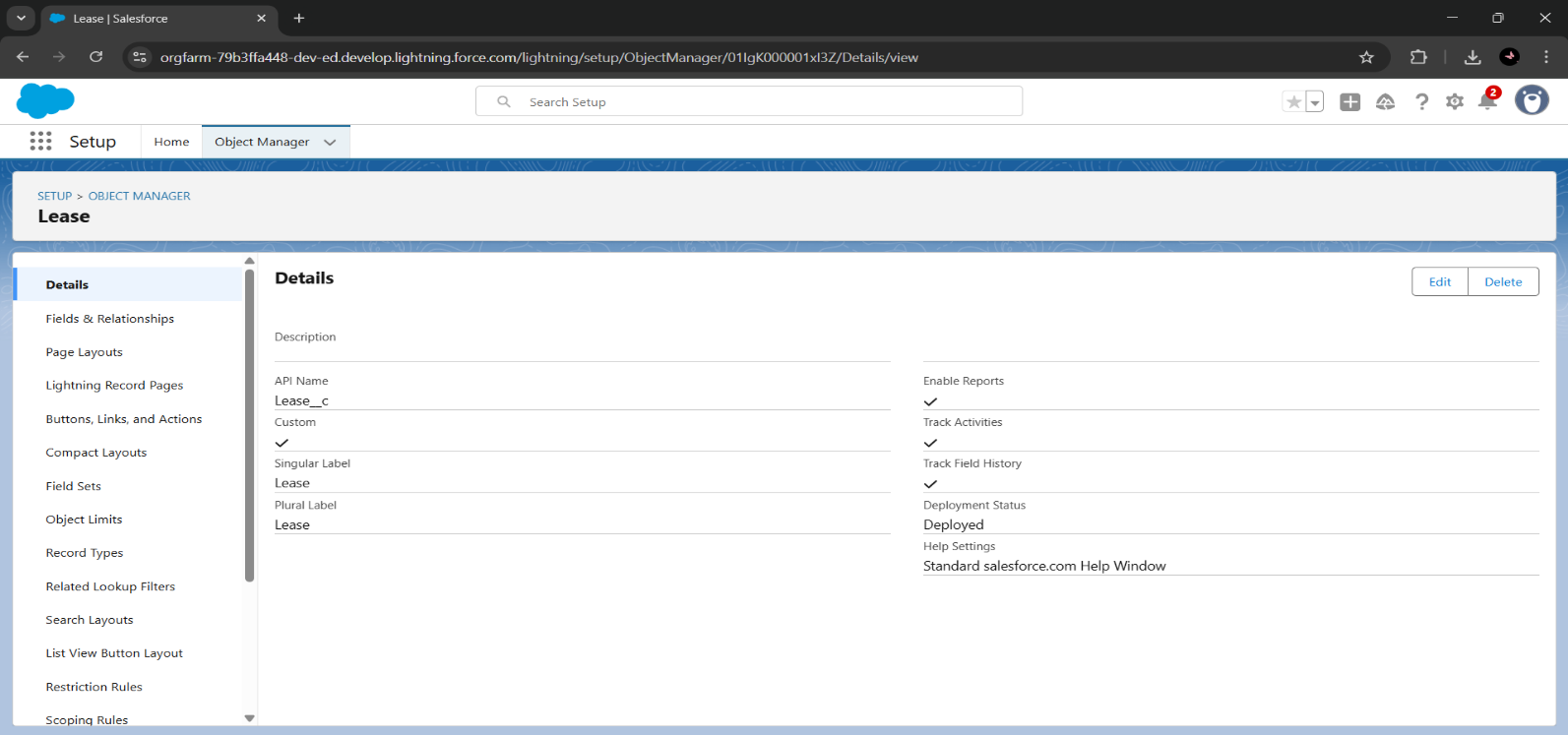
****

1. **Created objects: Property, Tenant, Lease, Payment**

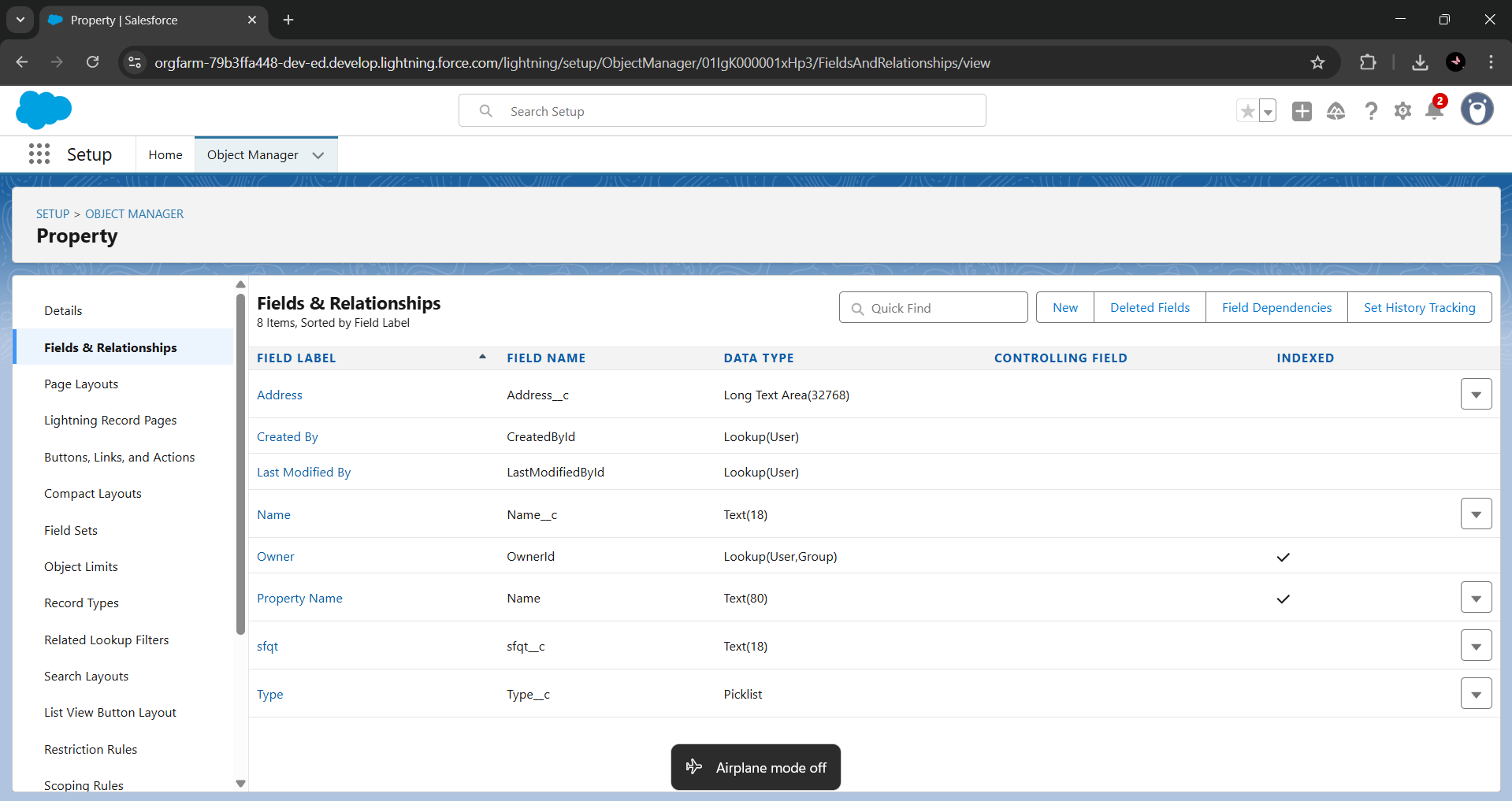
****

****

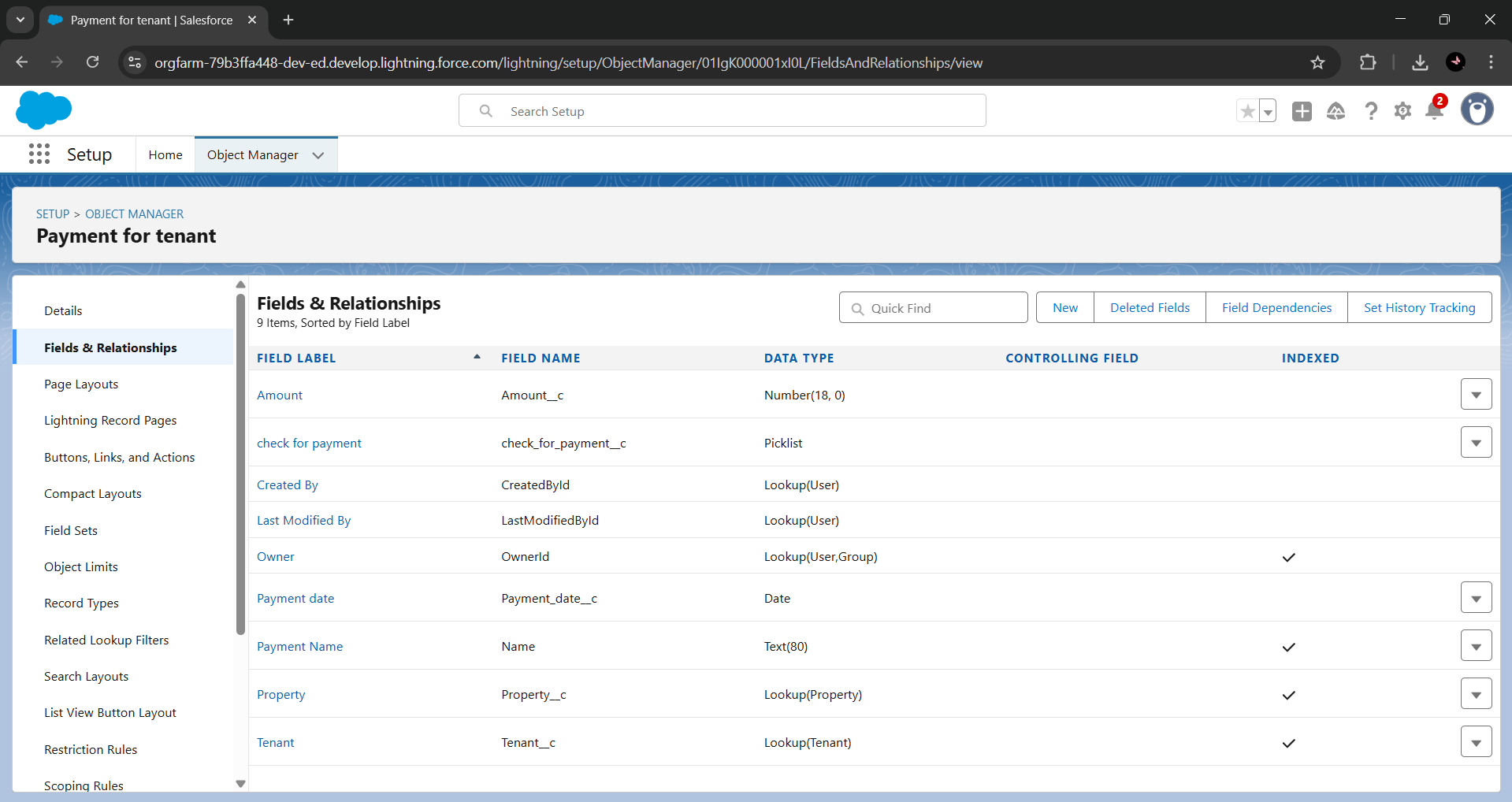
****

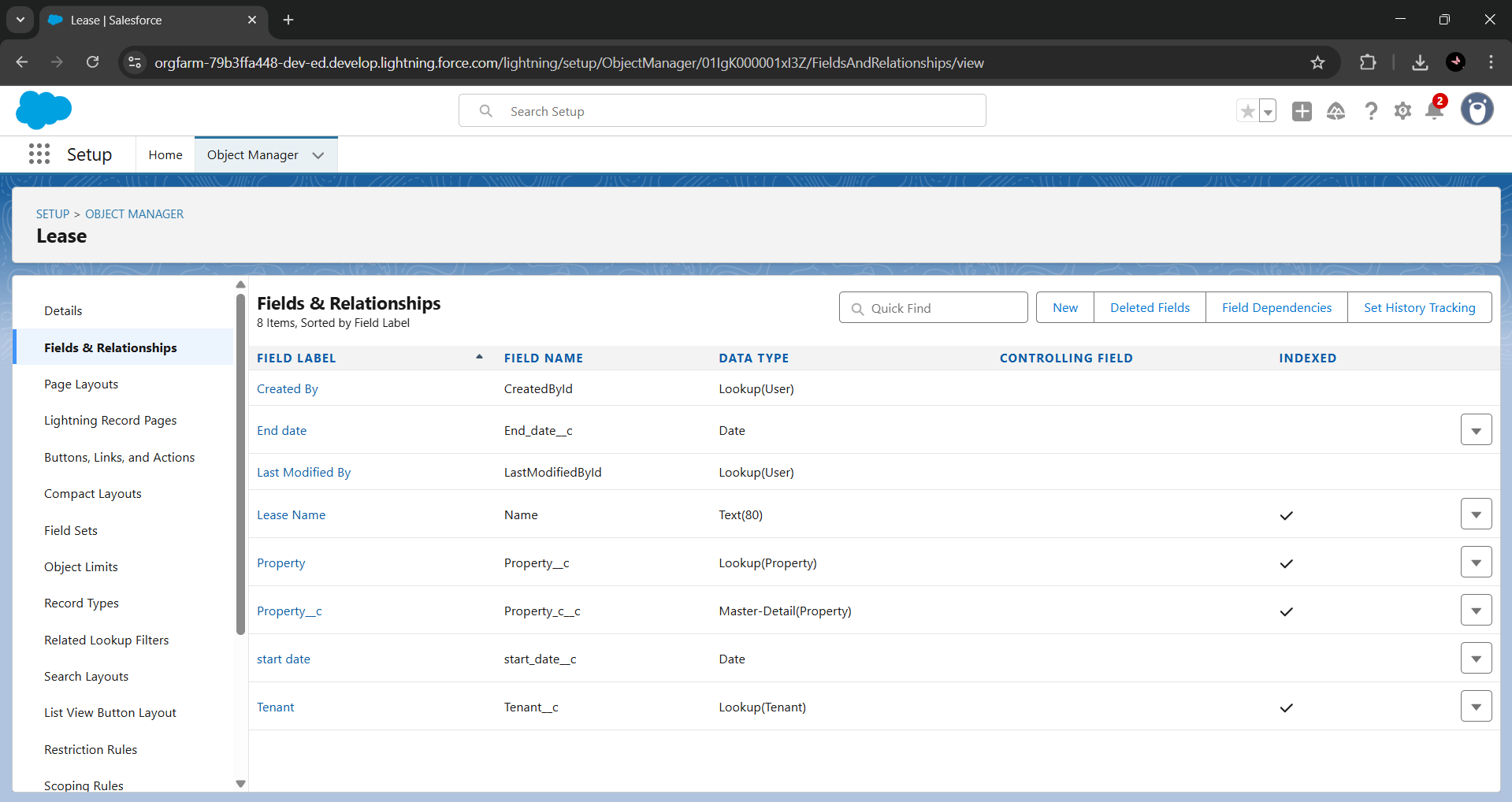
****

1. **Configured fields and relationships**

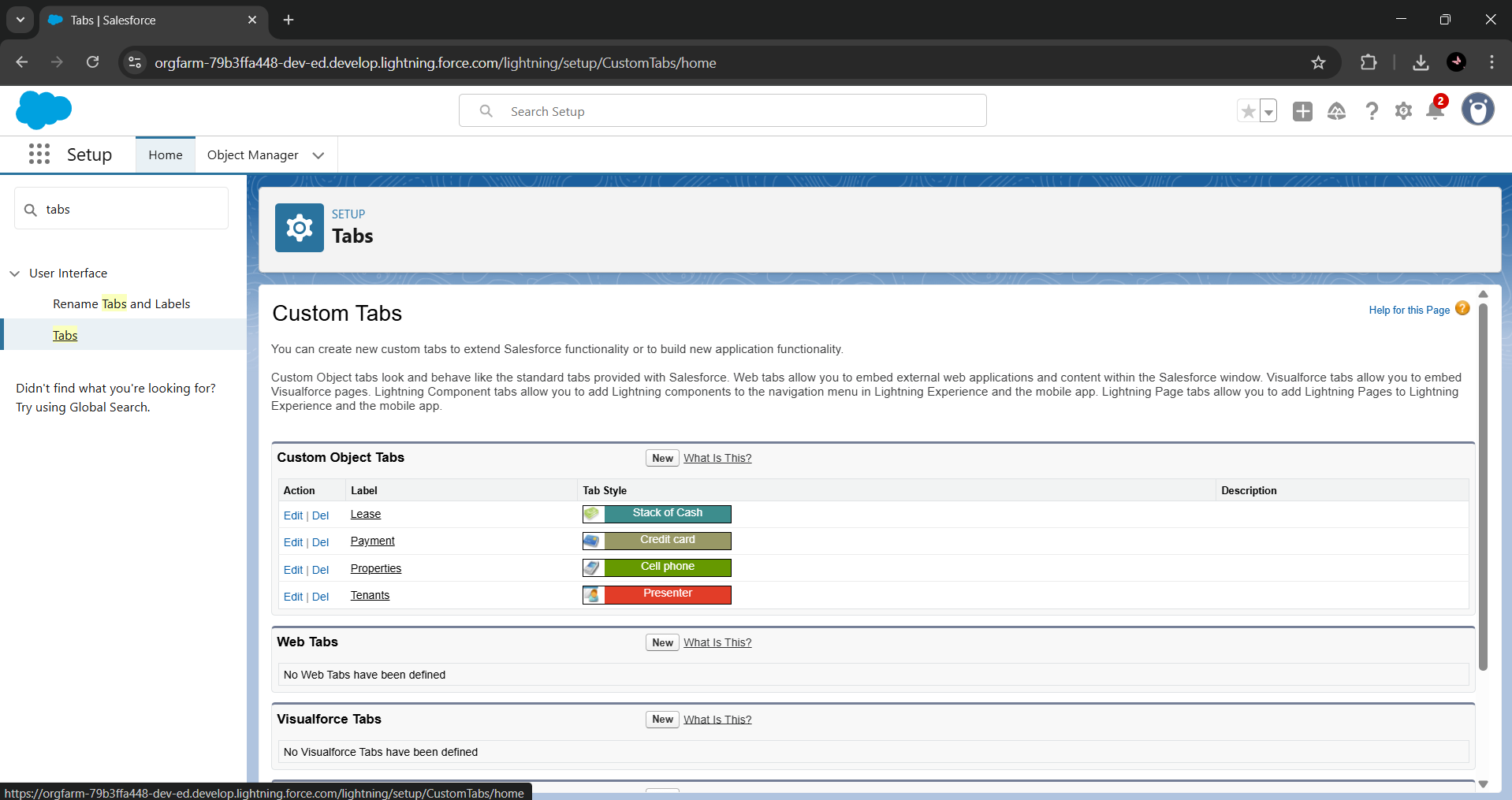
****



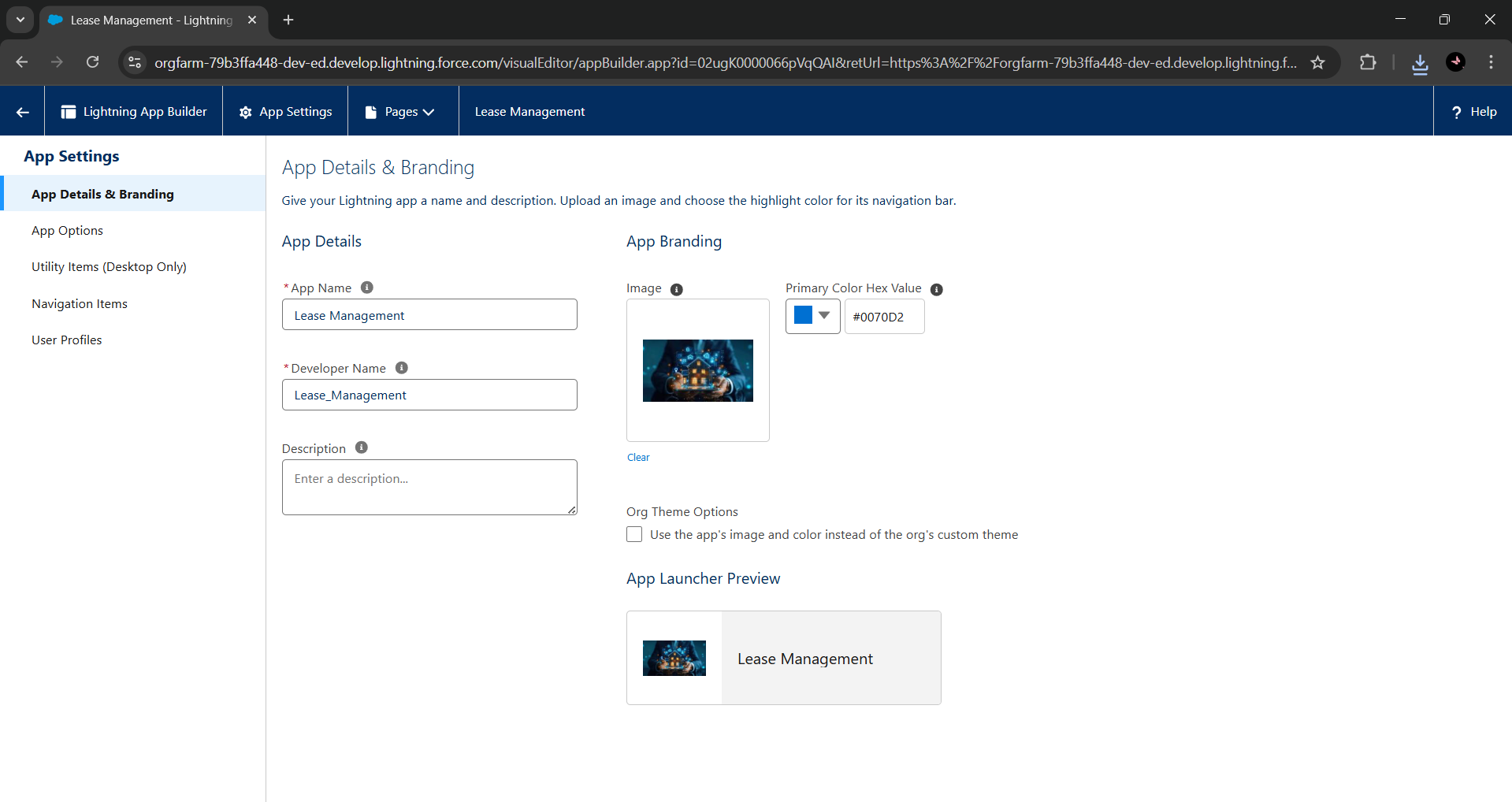




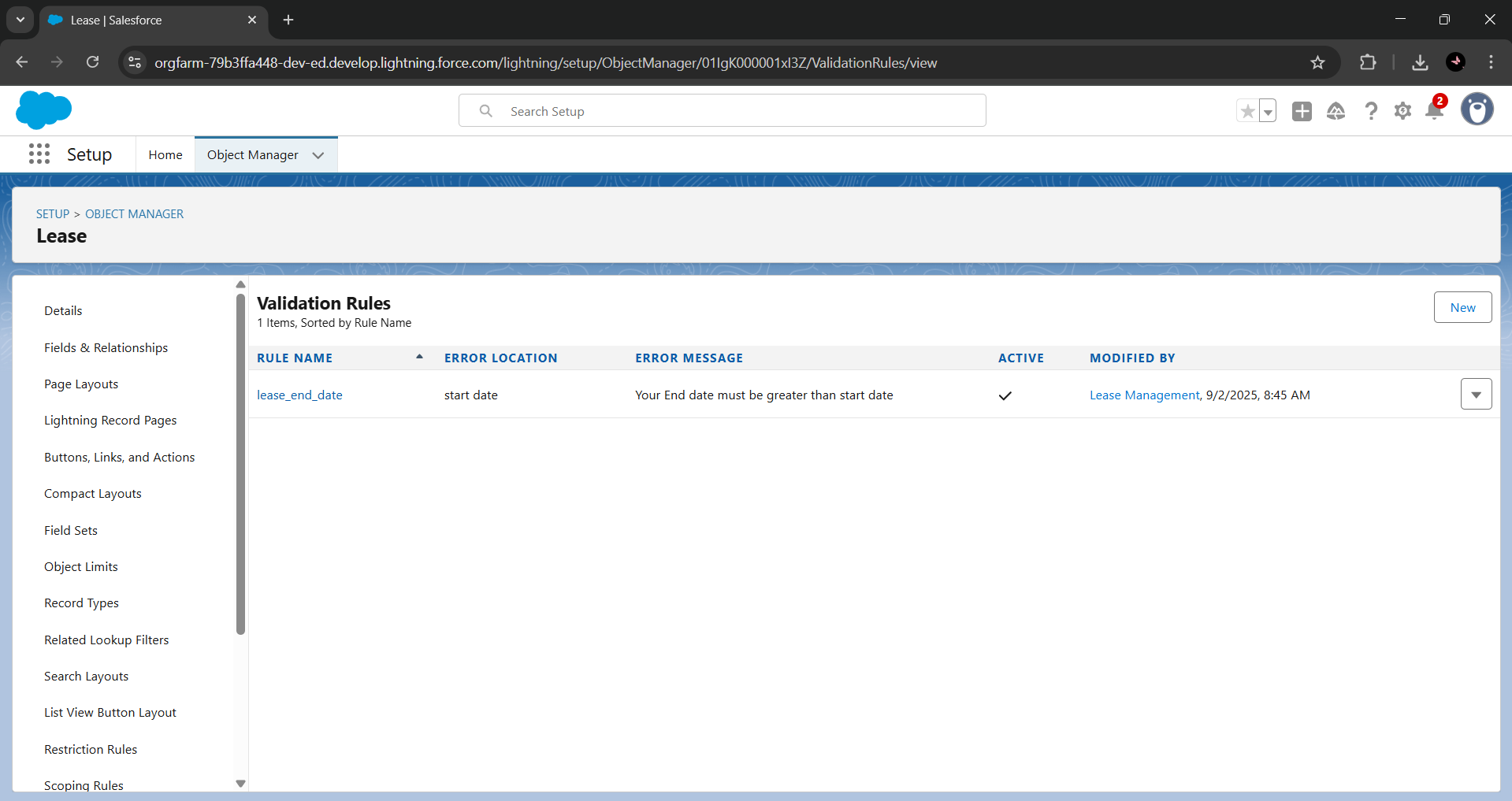
1. **Tabs**

****

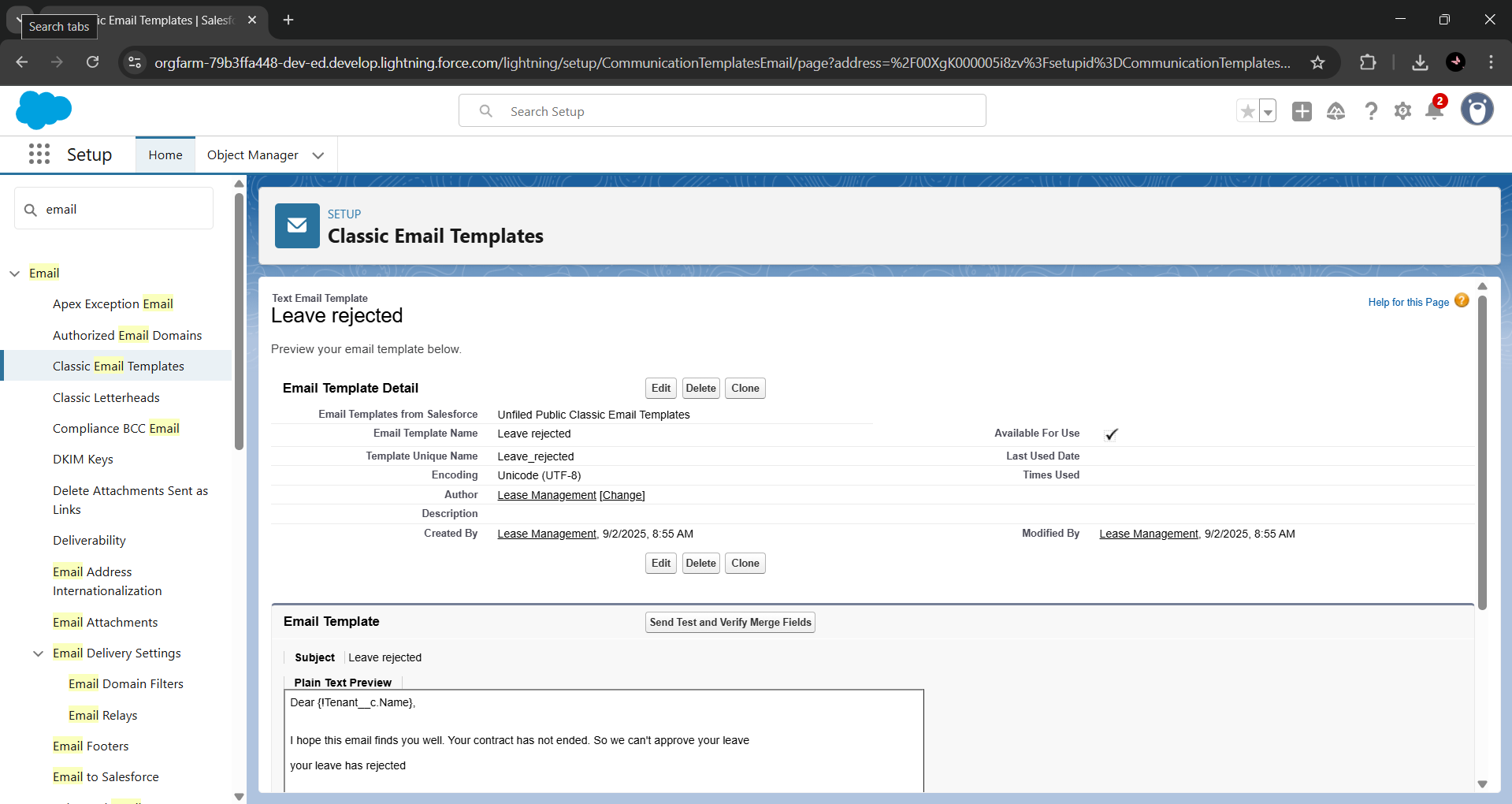
1. **Developed Lightning App**

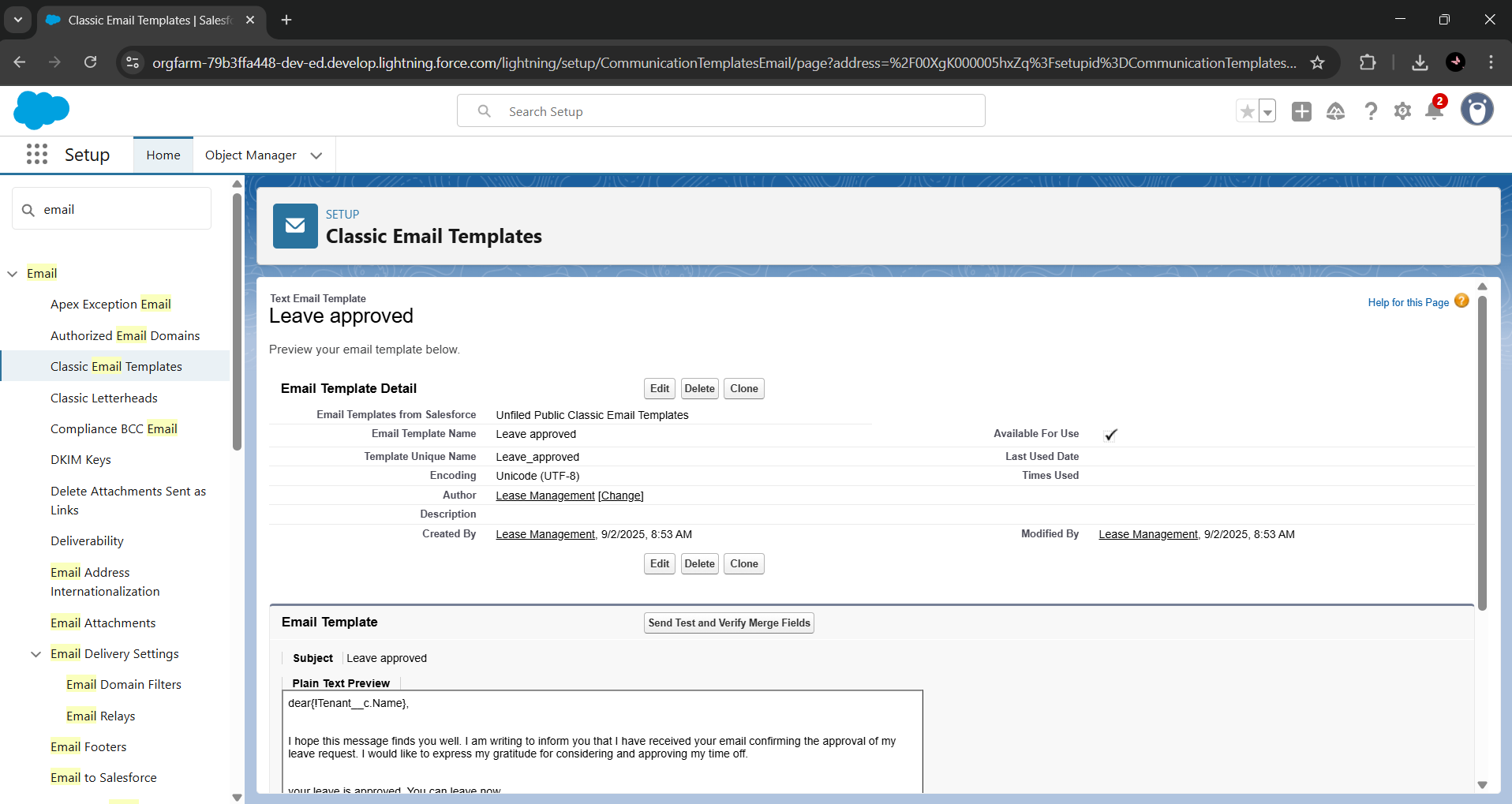
****

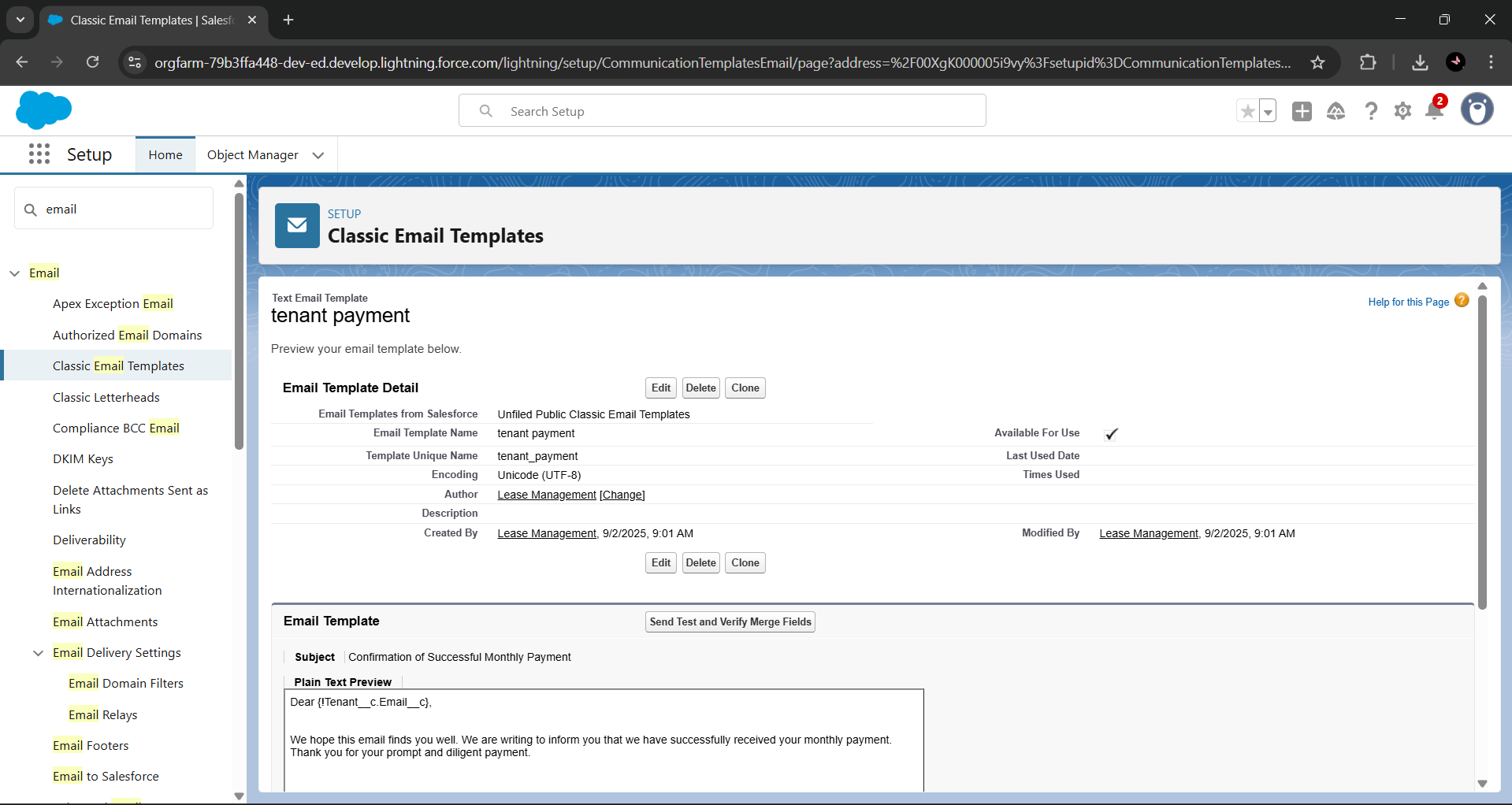
1. **To create a validation rule to a Lease Object**

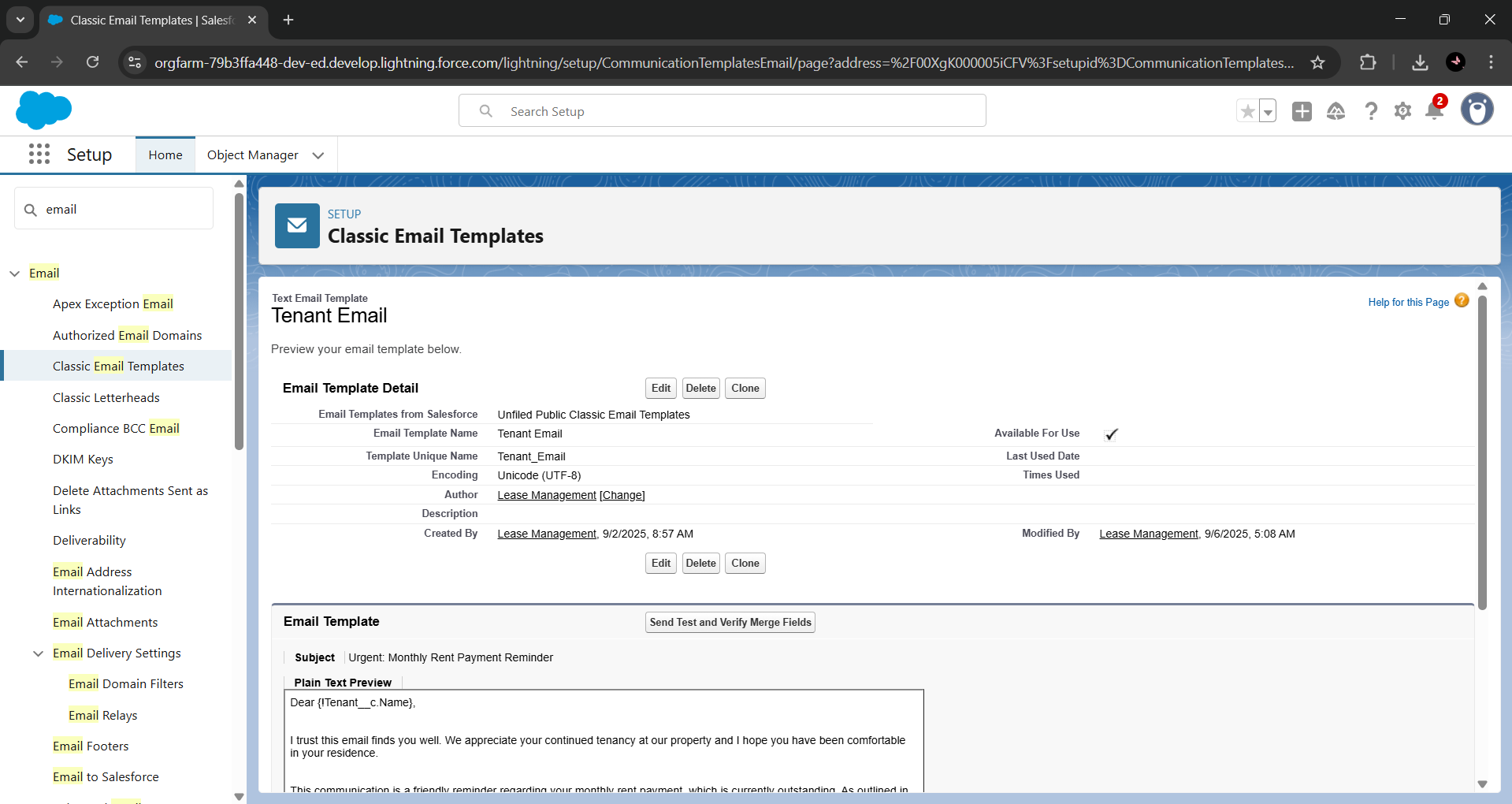
****

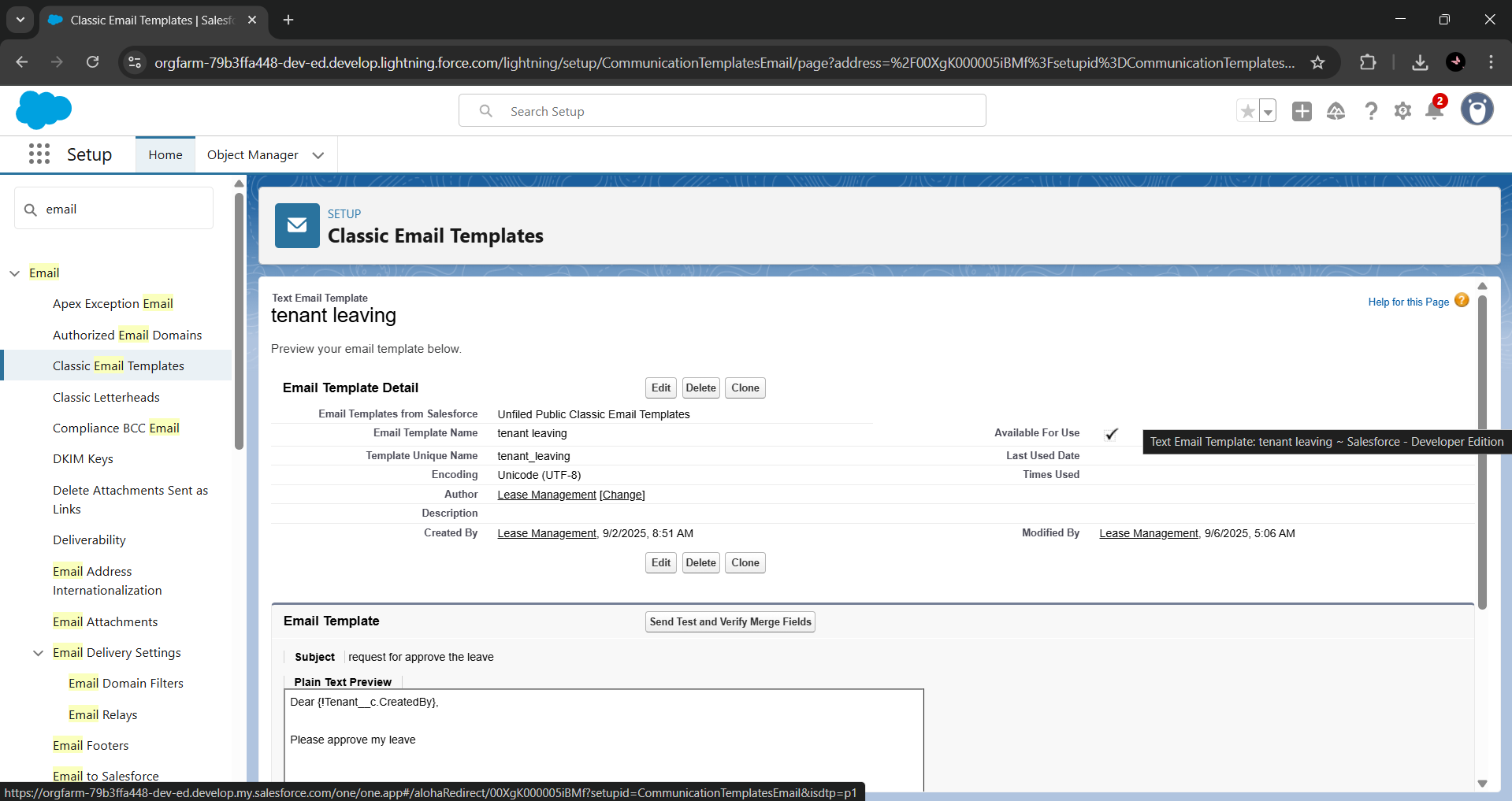
1. **Built and tested email templates for leave request, approval, rejection, payment, and reminders**

****

****

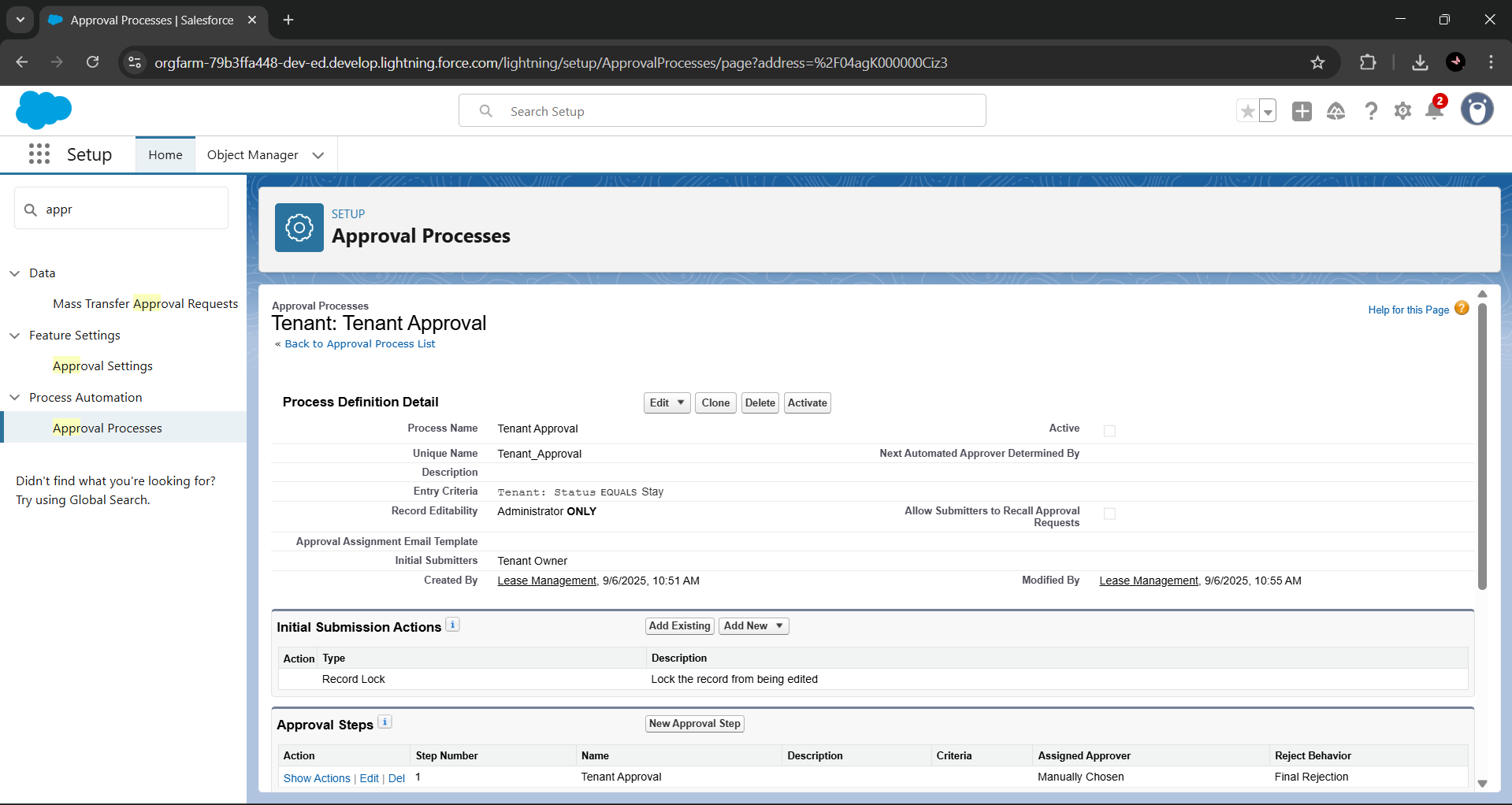
****

****

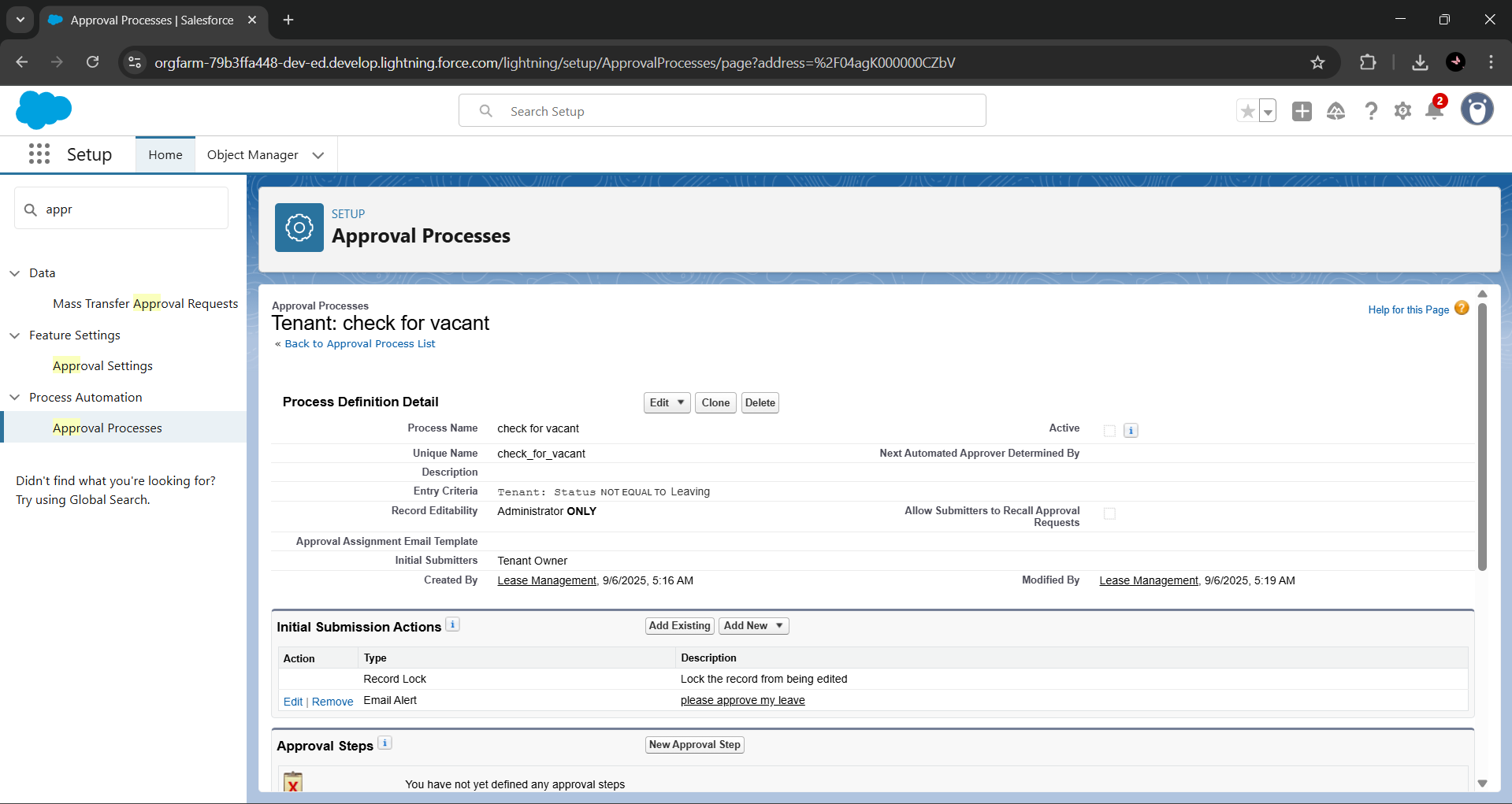
****

1. **Approval Process creation**

For Tenant Leaving:

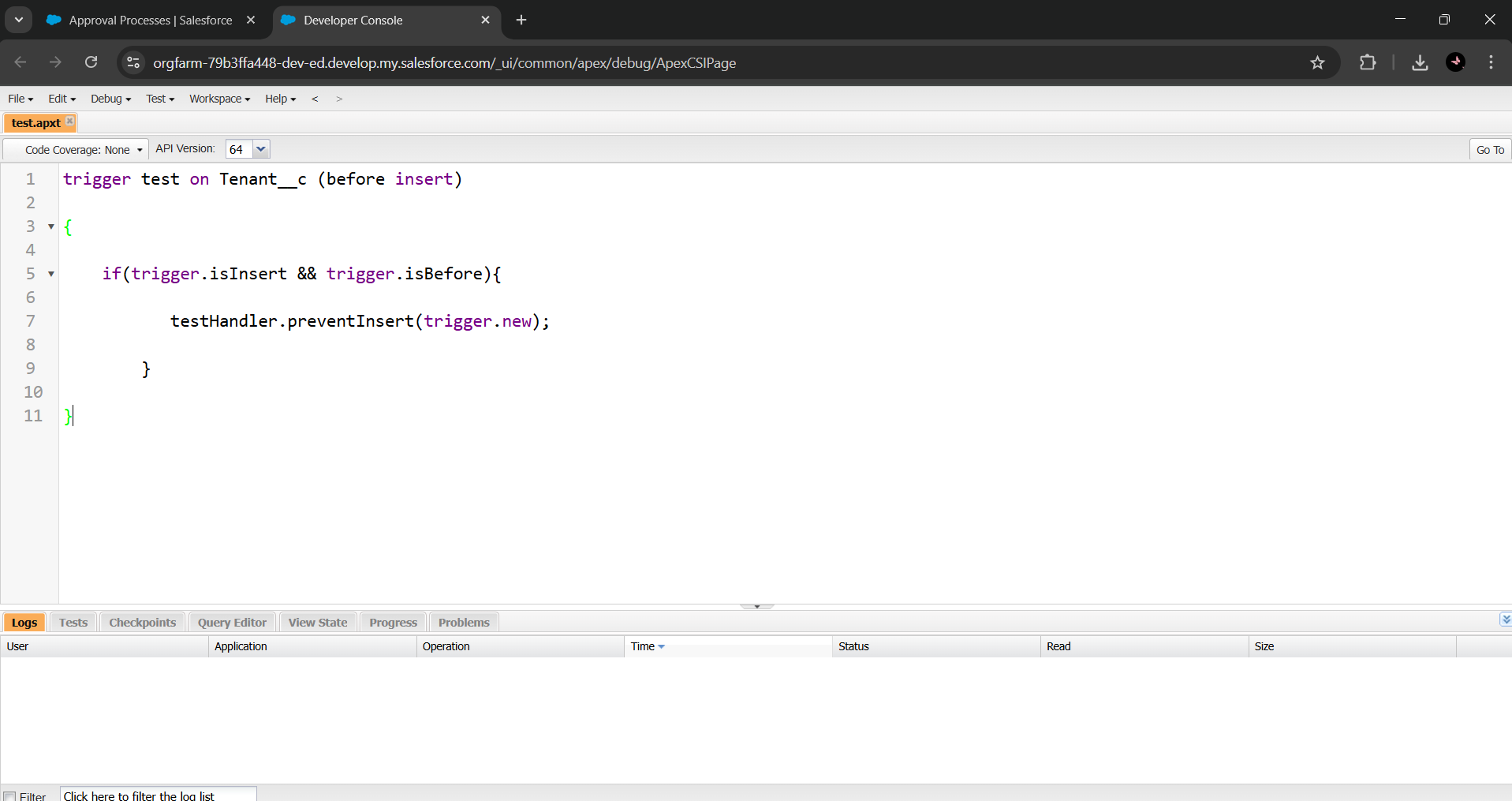


For Check for Vacant:

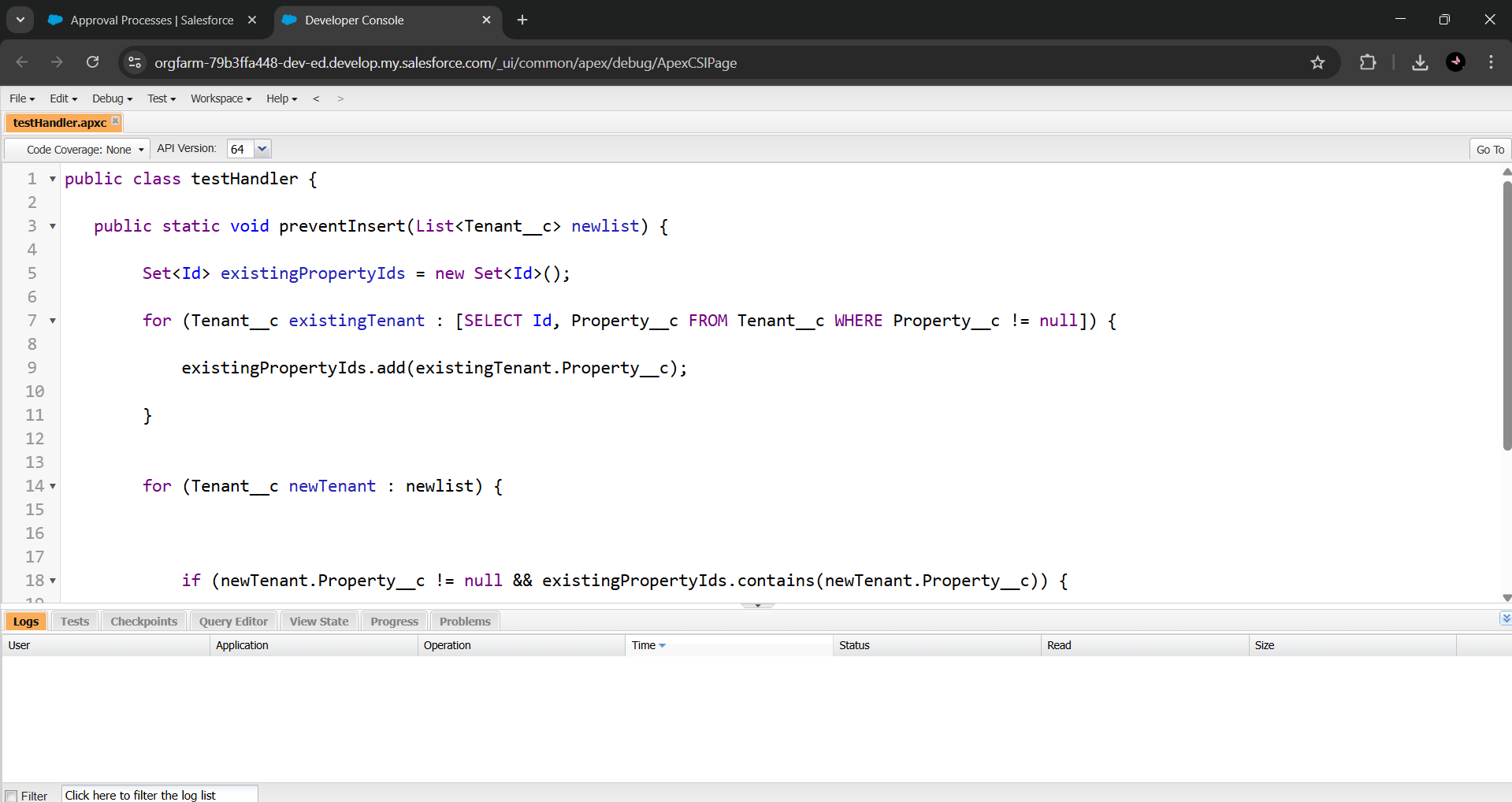


1. **Apex Trigger**

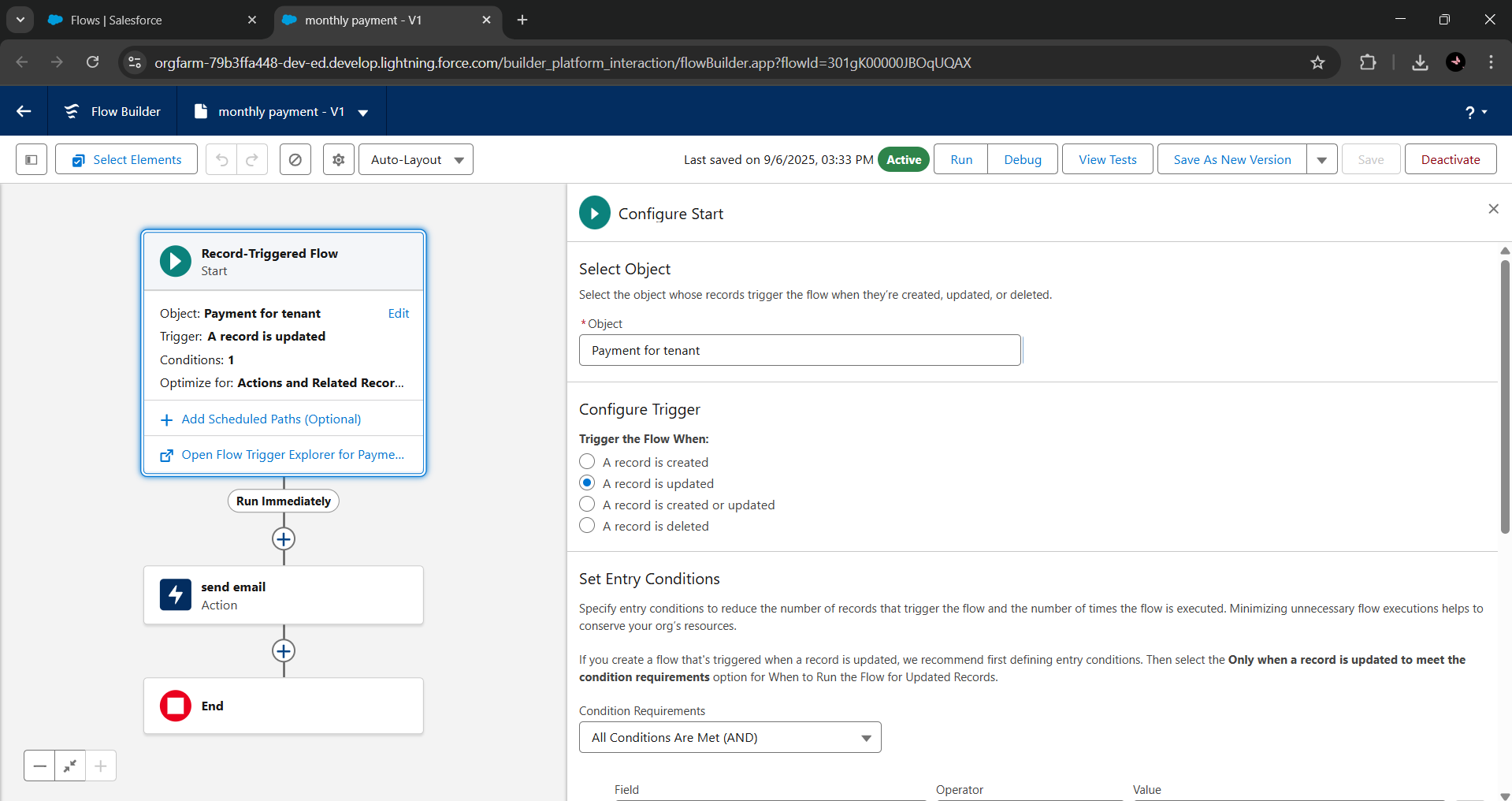
Create an Apex Trigger

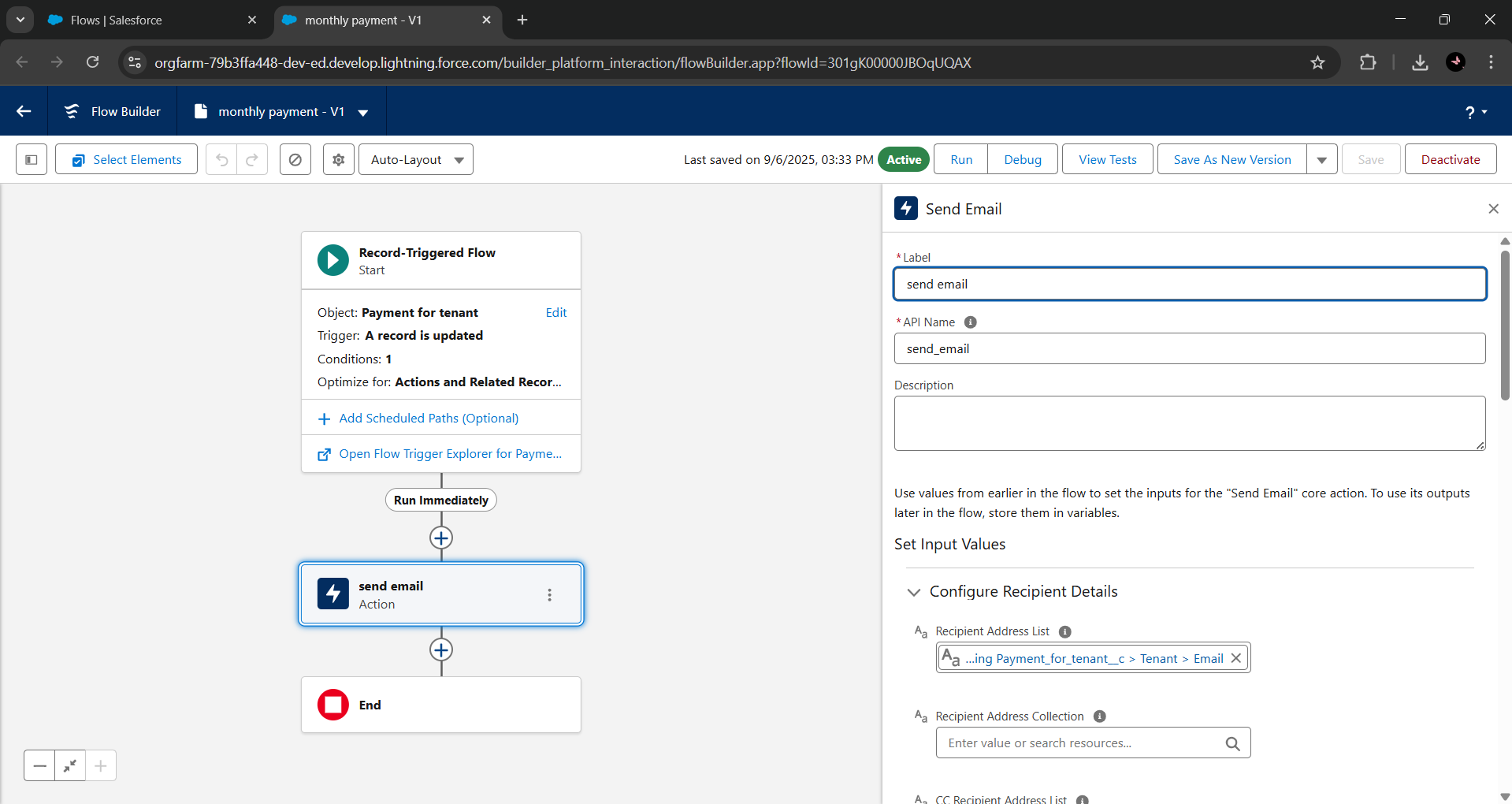


Create an Apex Handler class



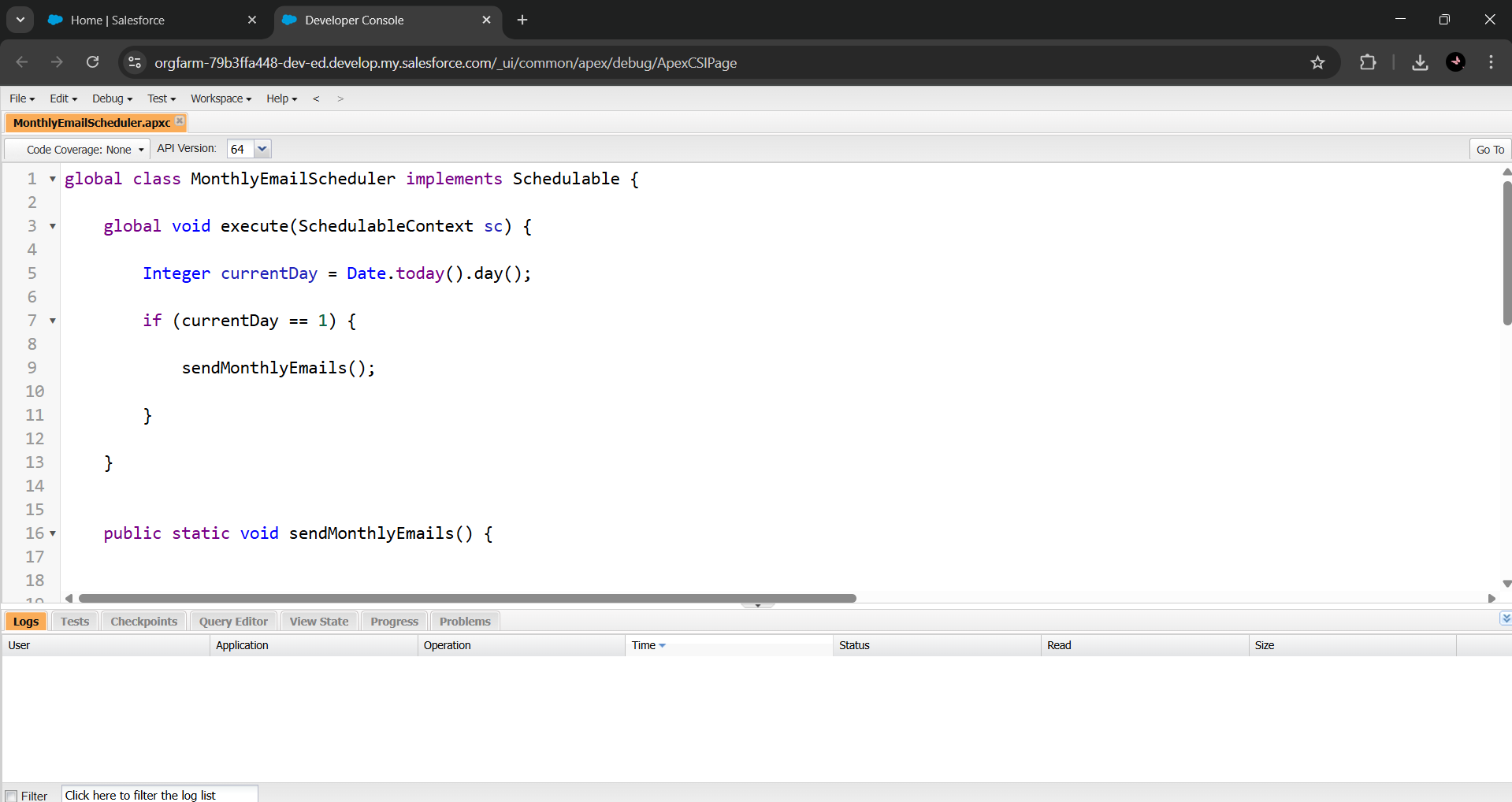
1. **FLOWS**

****

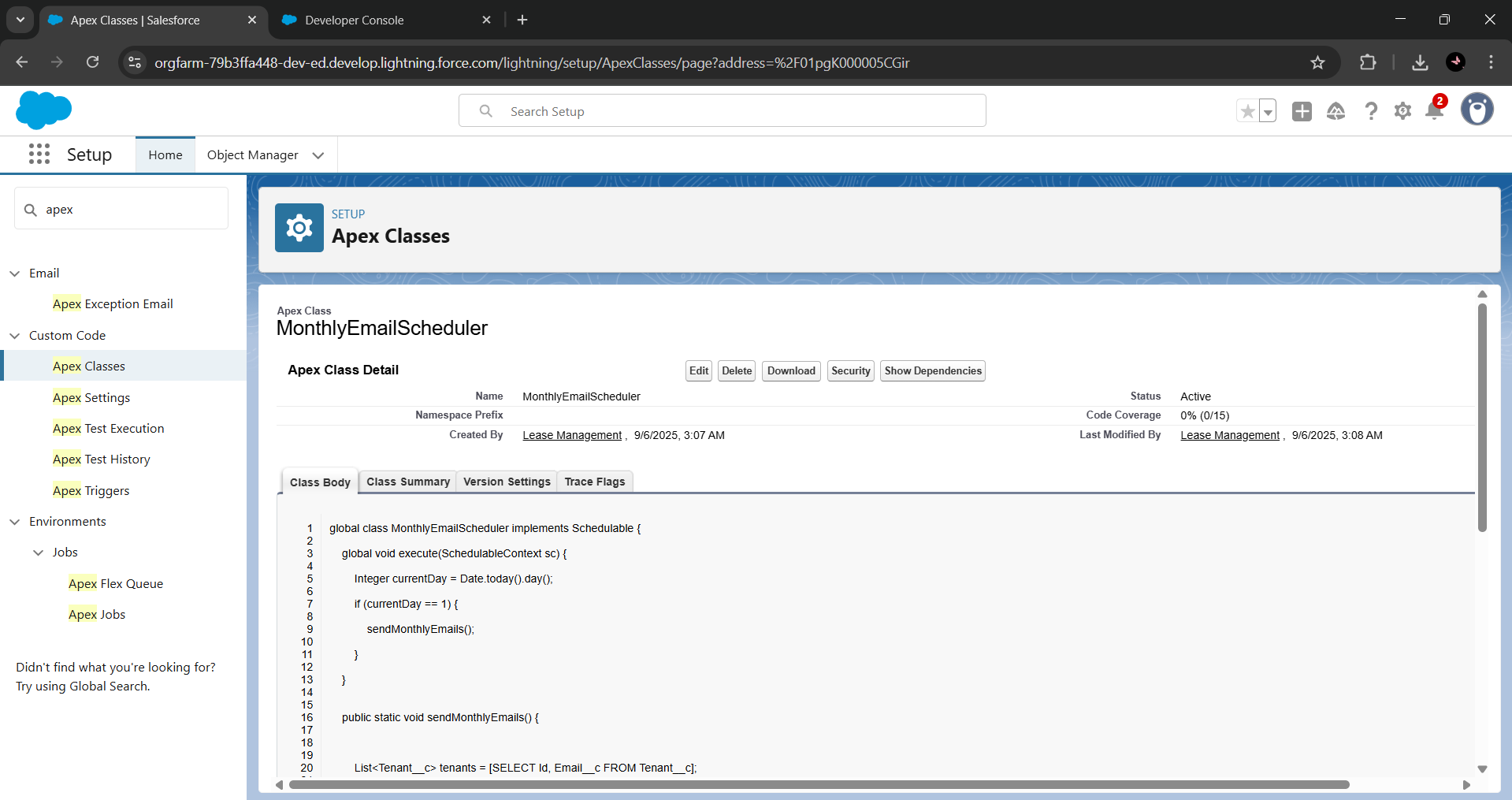
****

1. **Schedule class:**

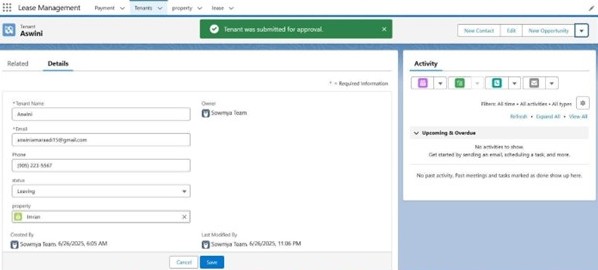
Create an Apex Class

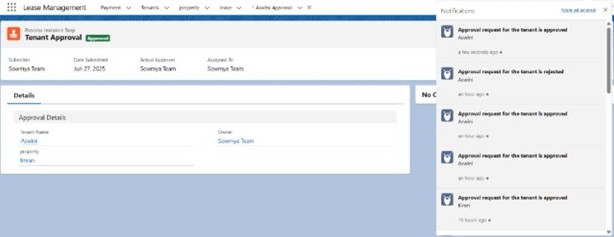


Schedule Apex class





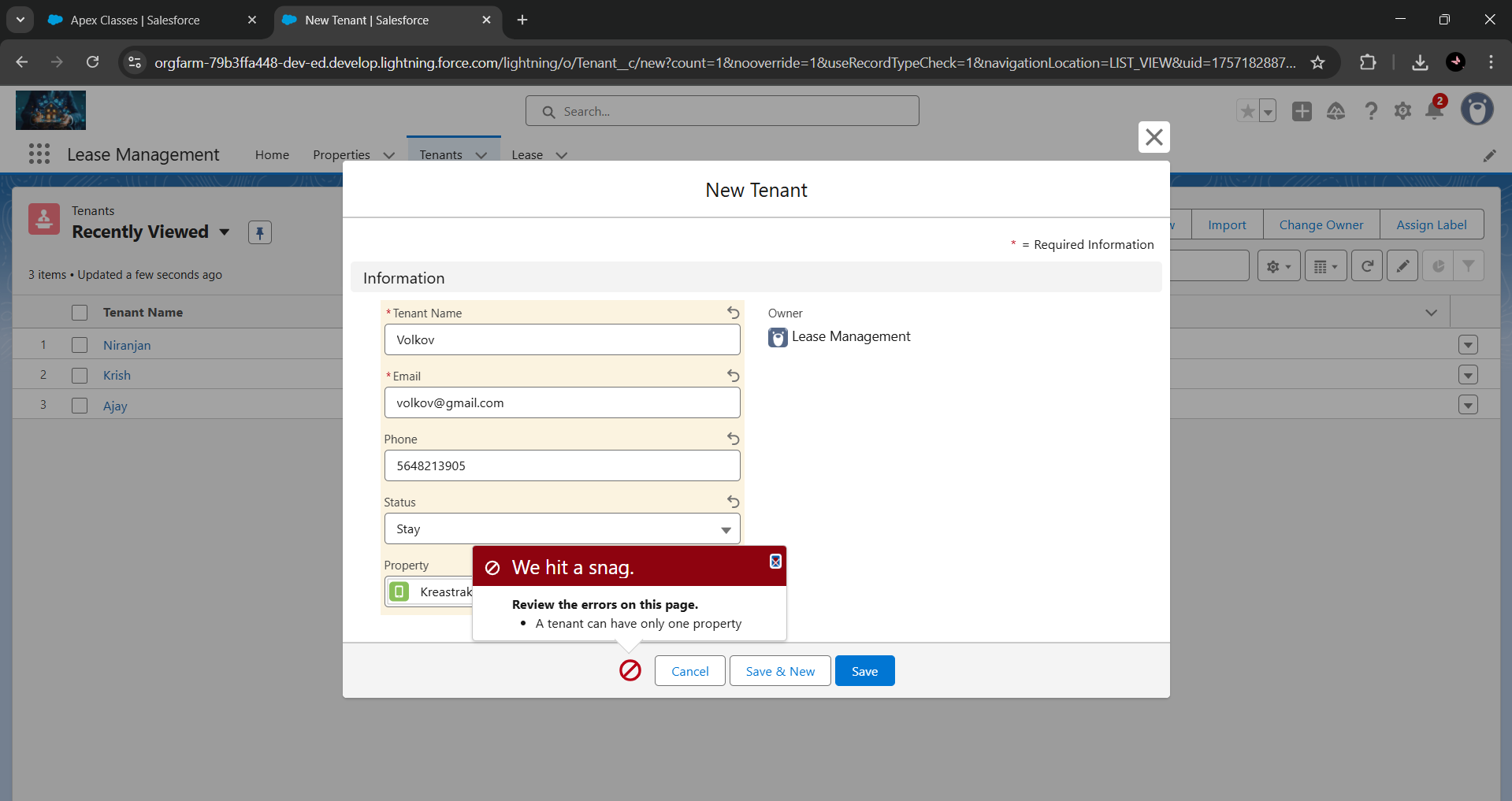




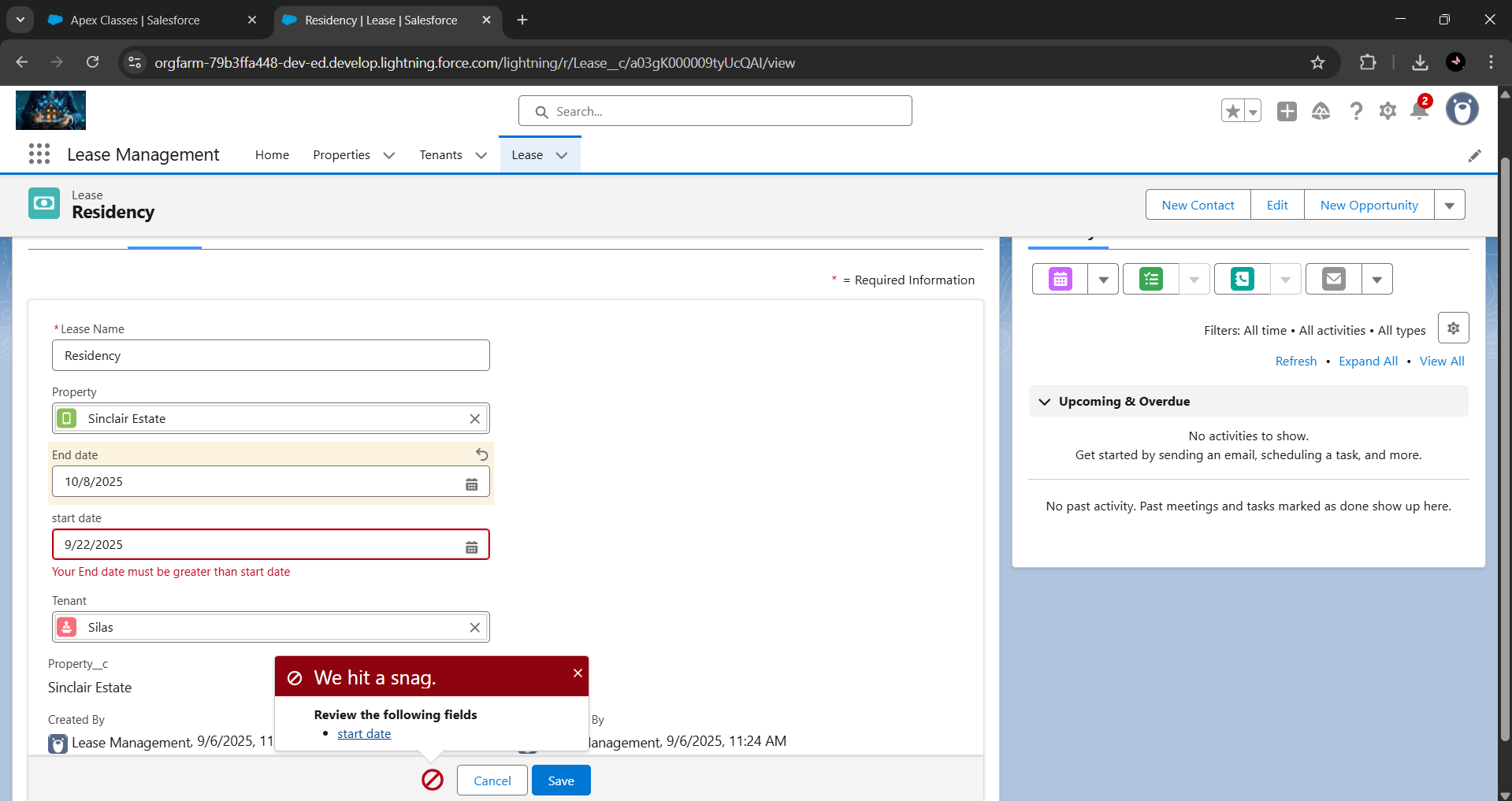
**FUNCTIONAL AND PERFORMANCE TESTING**

1. **Performance Testing**

Trigger validation by entering duplicate tenant-property records



Validation Rule checking



**ADVANTAGES & DISADVANTAGES**

**✅ Advantages**

1. **Operational Efficiency** – Automates routine tasks such as lease renewals, payment tracking, and tenant communication, reducing manual workload and saving time.
2. **Improved Accuracy** – Minimizes human errors in contracts, financial records, and compliance requirements through automated workflows and validations.
3. **Enhanced Compliance** – Ensures adherence to lease terms, legal obligations, and organizational policies with built-in approval processes and alerts.
4. **Centralized Data Management** – Provides a single platform to store and manage tenant information, lease documents, and financial data, improving accessibility and transparency.
5. **Better Communication** – Strengthens engagement between property managers and tenants through automated notifications, reminders, and email alerts.
6. **Scalability** – Supports organizational growth by handling an increasing number of properties, tenants, and contracts without additional manual effort.
7. **Data-Driven Insights** – Offers reporting and analytics features for informed decision-making and strategic planning.

**⚠️ Disadvantages**

1. **High Initial Setup Cost –** Implementation may require significant investment in customization, licensing, and integration with existing systems.
2. **Training Requirements –** Users may need training to fully understand and utilize Salesforce features, which can take time and resources.
3. **Dependency on Internet and Platform –** As a cloud-based system, it relies heavily on stable internet connectivity and the Salesforce platform’s uptime.
4. **Customization Complexity –** Advanced customization to meet unique business needs may require technical expertise, potentially increasing dependency on developers.
5. **Data Security Concerns –** Although Salesforce provides strong security, organizations must still manage data privacy risks, especially when handling sensitive tenant and financial information.
6. **Change Management Challenges –** Transitioning from traditional methods to an automated system may face resistance from staff who are accustomed to manual processes.

**CONCLUSION**

The Lease Management System is a Salesforce-driven solution that transforms the way organizations manage properties, tenants, and lease-related activities. By automating key processes, centralizing data, and improving communication, it reduces manual effort, enhances accuracy, and ensures stronger compliance with regulatory and organizational standards. While the project may involve challenges such as implementation costs, training requirements, and change management, these are outweighed by its long-term advantages, including operational efficiency, scalability, and improved tenant satisfaction. Overall, the system not only streamlines leasing operations but also equips organizations with the tools to make data-driven decisions, strengthen relationships, and achieve sustainable growth in an increasingly competitive real estate landscape.

**APPENDIX**

Source Code: Provided in Apex Classes and Triggers

**Test.apxt:**

Trigger Code:

trigger test on Tenant\_\_c (before insert)

{

    if(trigger.isInsert && trigger.isBefore){

        testHandler.preventInsert(trigger.new);

    }

}

**testHandler.apxc:**

public class testHandler {

   public static void preventInsert(List<Tenant\_\_c> newlist) {

        Set<Id> existingPropertyIds = new Set<Id>();

        for (Tenant\_\_c existingTenant : [SELECT Id, Property\_\_c FROM Tenant\_\_c WHERE Property\_\_c != null]) {

            existingPropertyIds.add(existingTenant.Property\_\_c);

        }

        for (Tenant\_\_c newTenant : newlist) {

            if (newTenant.Property\_\_c != null && existingPropertyIds.contains(newTenant.Property\_\_c)) {

                newTenant.addError('A tenant can have only one property');

            }

        }

    }

}

**MothlyEmailScheduler.apxc:**

global class MonthlyEmailScheduler implements Schedulable {

global void execute(SchedulableContext sc) {

Integer currentDay = Date.today().day();

if (currentDay == 1) {

sendMonthlyEmails();

}

}

public static void sendMonthlyEmails() {

List<Tenant\_\_c> tenants = [SELECT Id, Email\_\_c FROM Tenant\_\_c];

for (Tenant\_\_c tenant : tenants) {

String recipientEmail = tenant.Email\_\_c;

String emailContent = 'I trust this email finds you well. I am writing to remind you that the monthly rent is due Your timely payment ensures the smooth functioning of our rental arrangement and helps maintain a positive living environment for all.';

String emailSubject = 'Reminder: Monthly Rent Payment Due';

Messaging.SingleEmailMessage email = new Messaging.SingleEmailMessage();

email.setToAddresses(new String[]{recipientEmail});

email.setSubject(emailSubject);

email.setPlainTextBody(emailContent);

Messaging.sendEmail(new Messaging.SingleEmailMessage[]{email});

}

}

}