PROJECTTITLE

**CollegeName: AVP COLLEGE OF ARTS AND SCIENCE**

**CollegeCode:asbruaj**

**TEAMID:NM2025TMID25405**

**TEAMMEMBERS:**

**TeamLeaderName: Annapoorani.V**

**Email:annapooraniais2006@gmail.com**

**TeamMember1: Gladish Blessy Riyal.K**

**Email: gladishblessyriyal@gmail.com**

**TeamMember2: Thenmozhi.R**

**Email:thenmozhi142005@gmail.com**

**TeamMember3: Samyuktha.P**

**Email:sampandiyan78 @gmail.com**

# 1.INTRODUCTION

## ProjectOverview

TheLeaseManagementSystemisaSalesforce-basedapplicationdesignedtostreamlinethe processesassociatedwith leasingreal estateproperties.Ithandlestenantmanagement, lease

contracts,payments,andcommunicationwithautomationfeaturessuchasflows,approval processes, and email alerts.

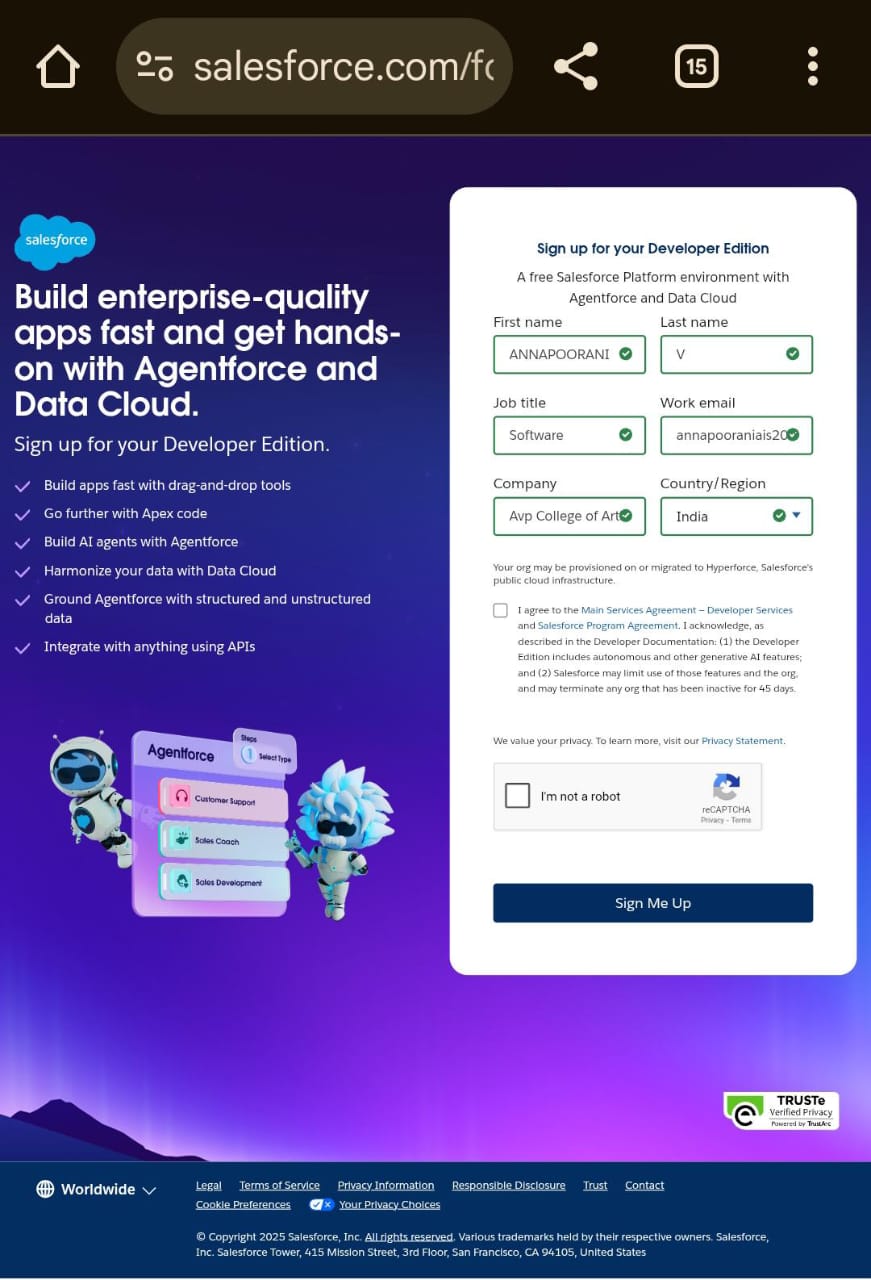
## Purpose

The main objective of the project is to enable organizations to efficiently manageproperties, tenants, and lease-related activities. It reduces manual intervention, improves accuracy, and ensures better compliance and communication.

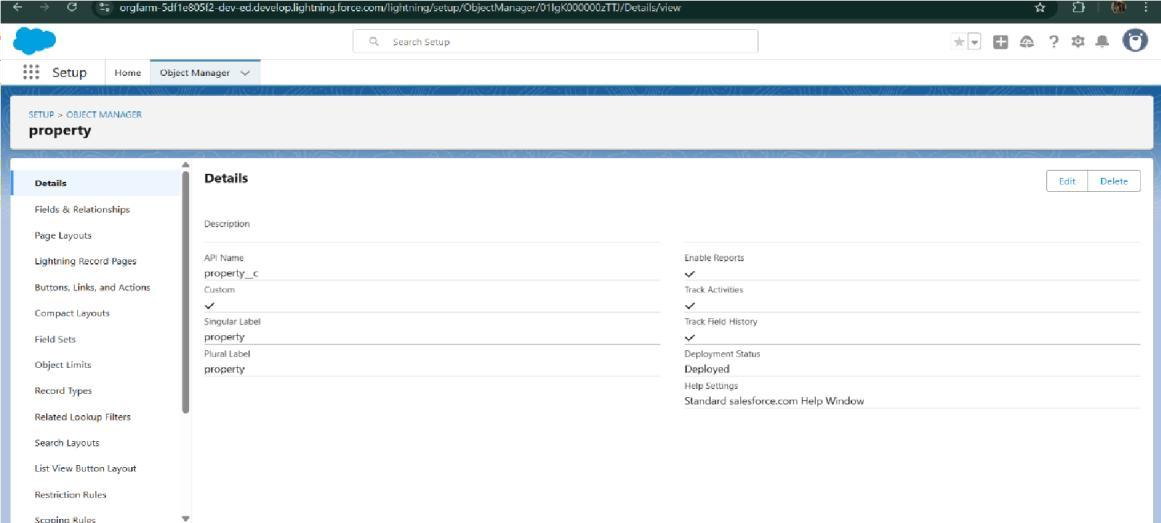
# DEVELOPMENTPHASE

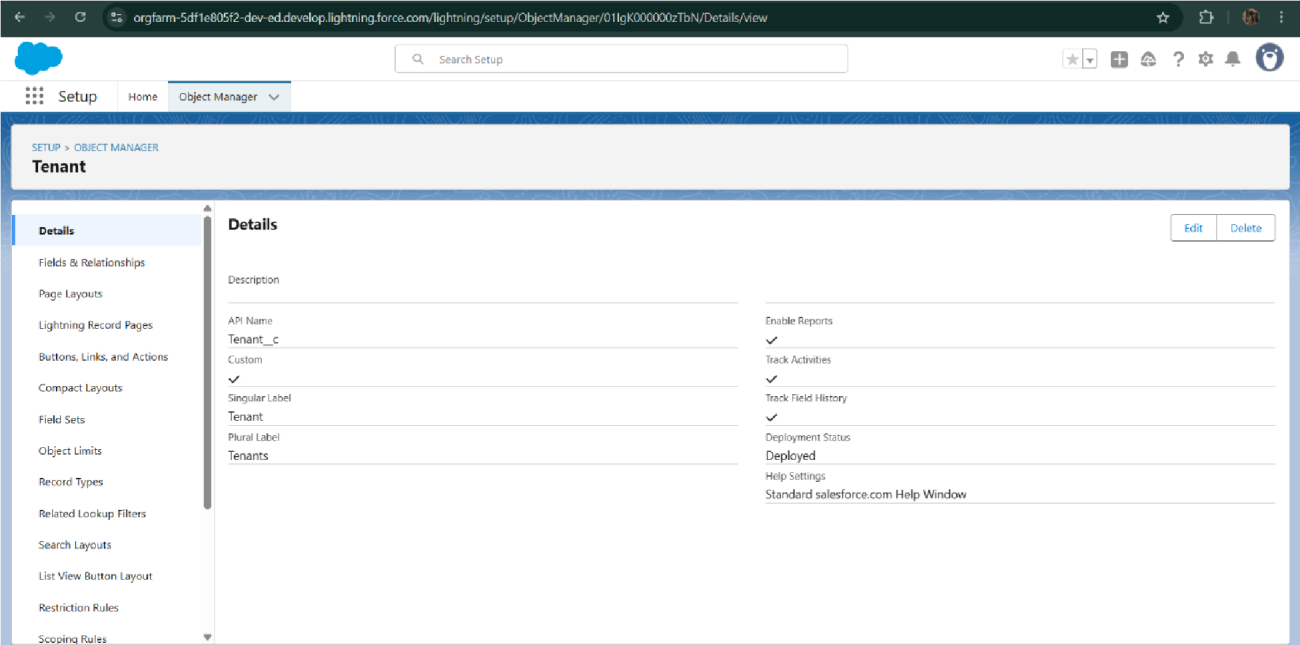
### CreatingDeveloperAccount:

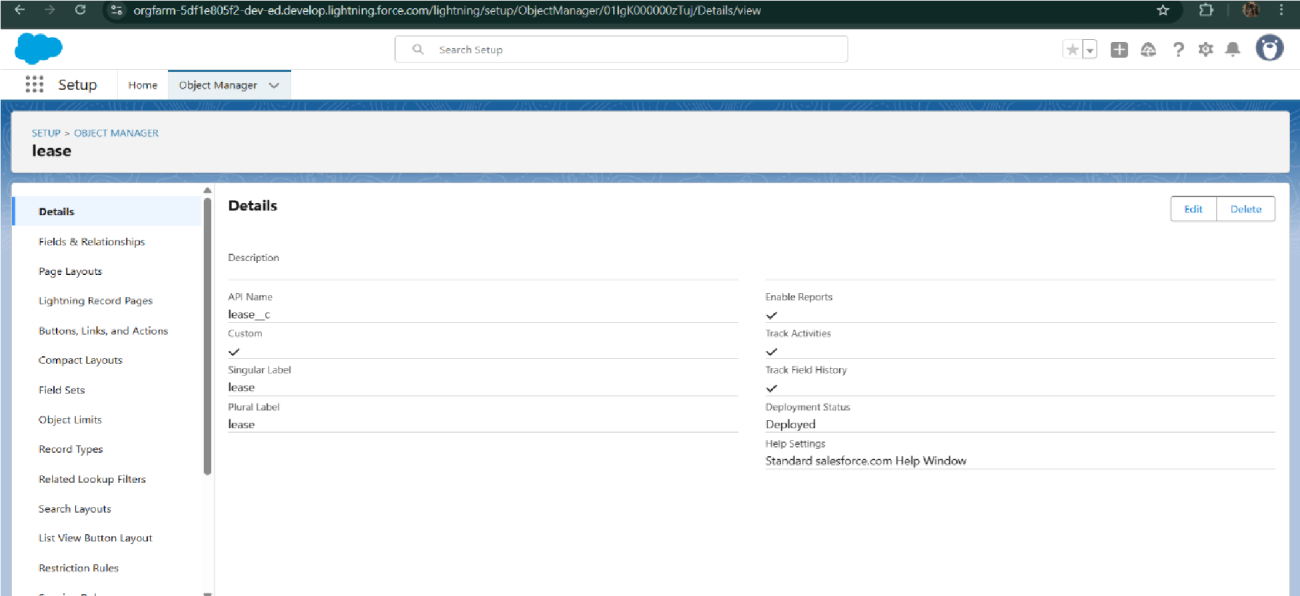
ByusingthisURL**-**[**https://www.salesforce.com/form/developer-signup/?d=pb**](https://www.salesforce.com/form/developer-signup/?d=pb)

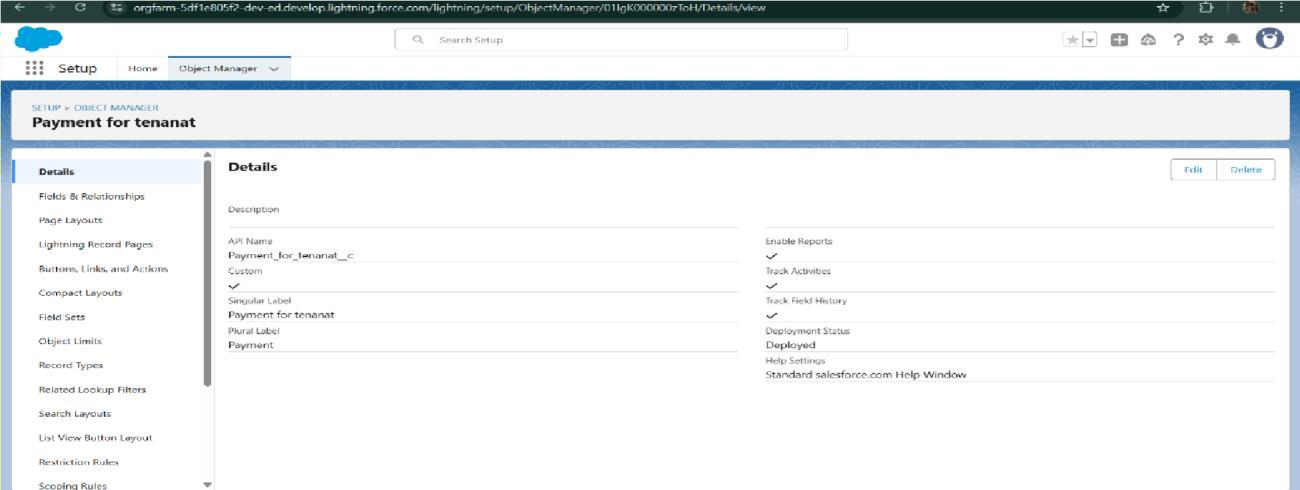


* + - Createdobjects:Property,Tenant,Lease,Payment

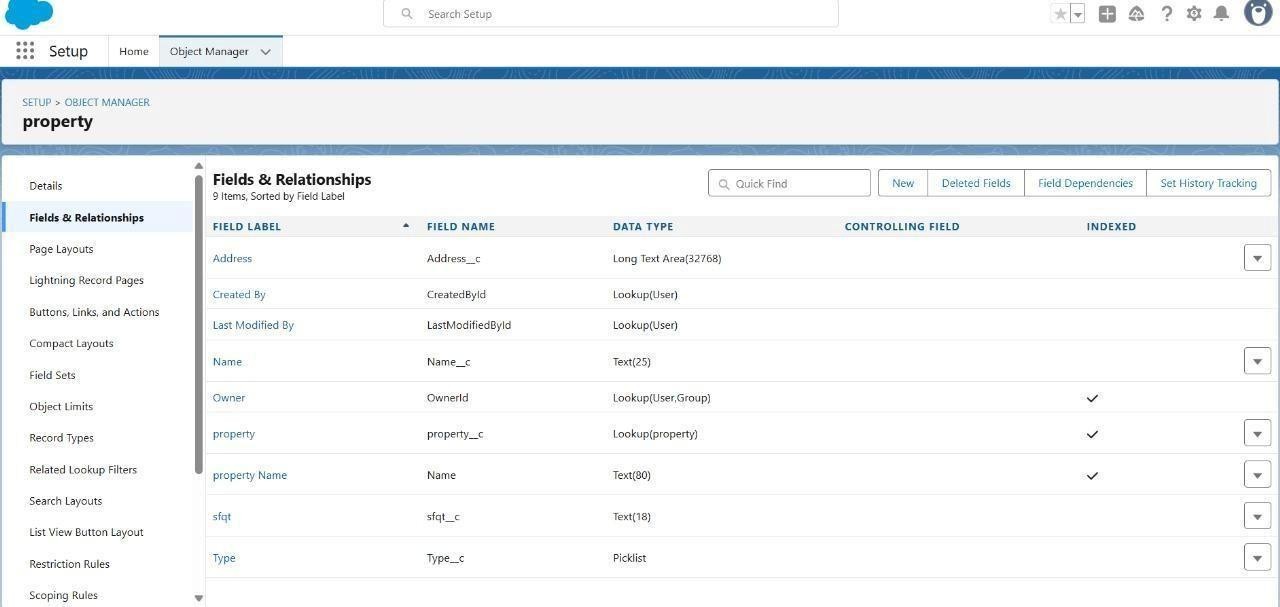


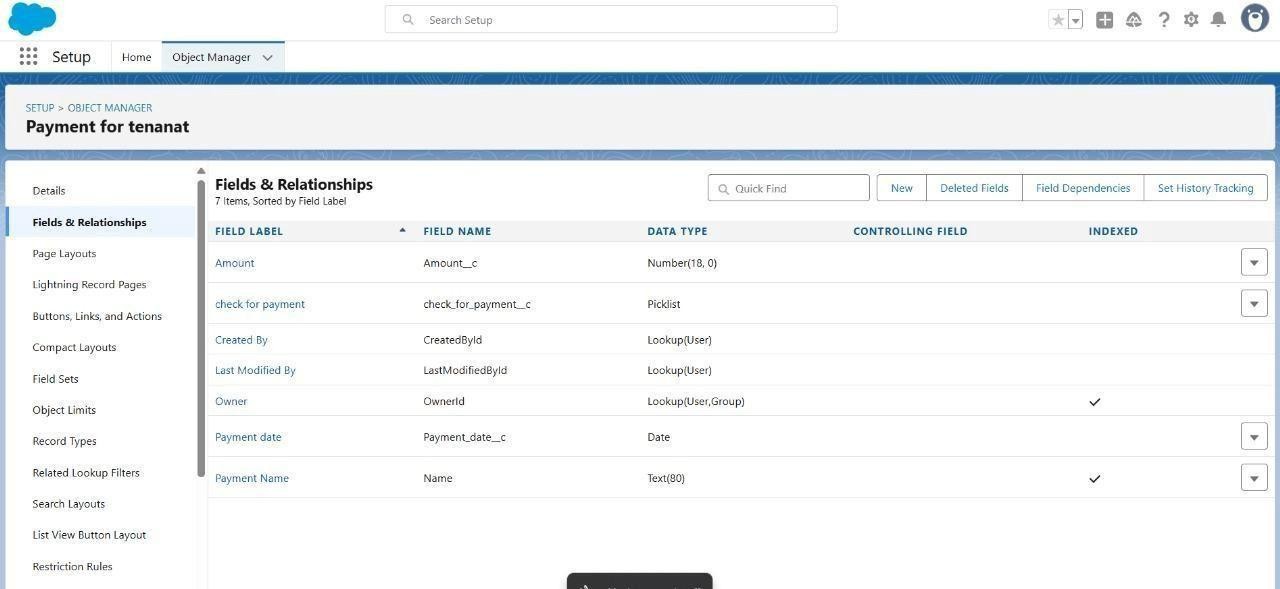


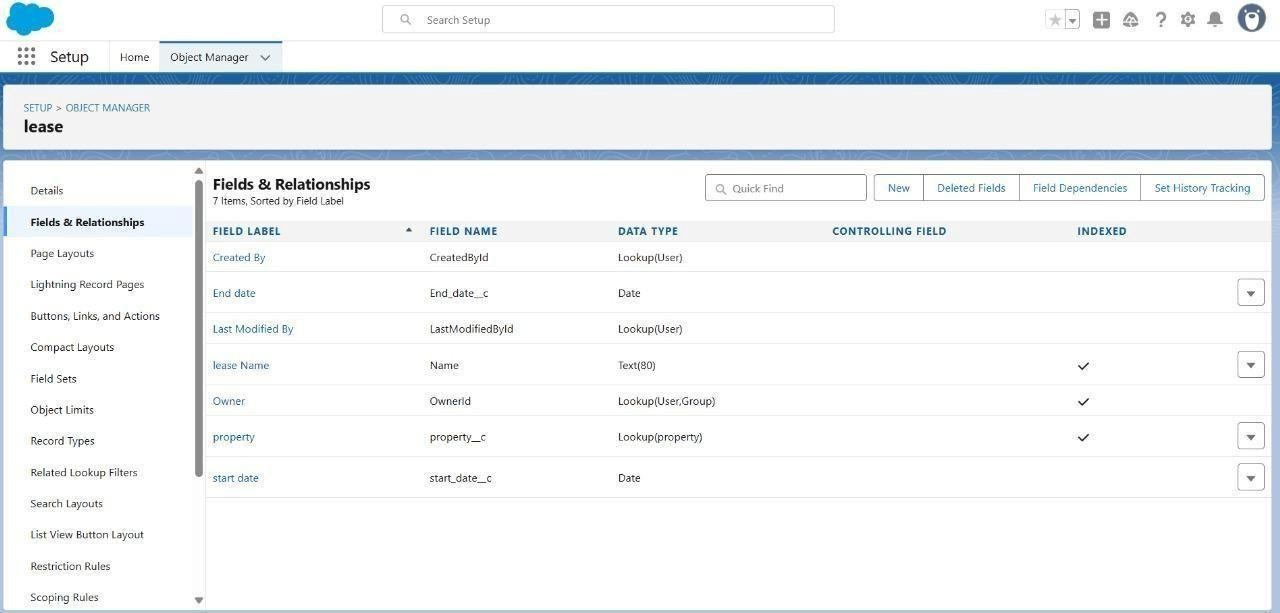


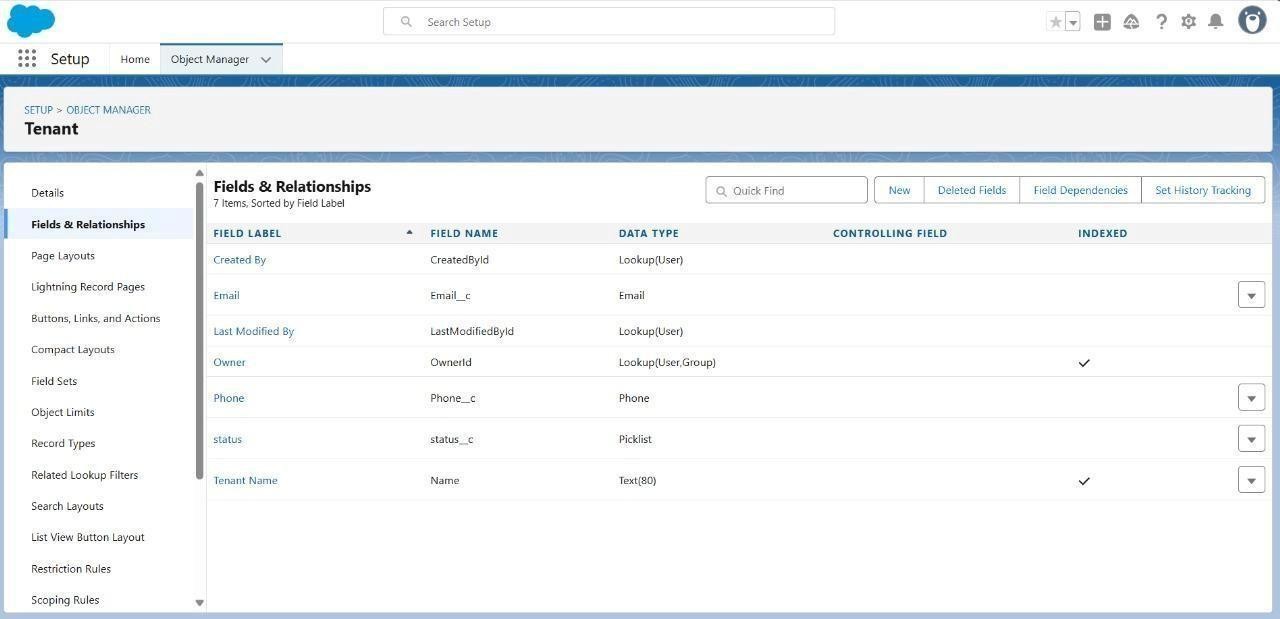


* + - Configuredfieldsand relationships

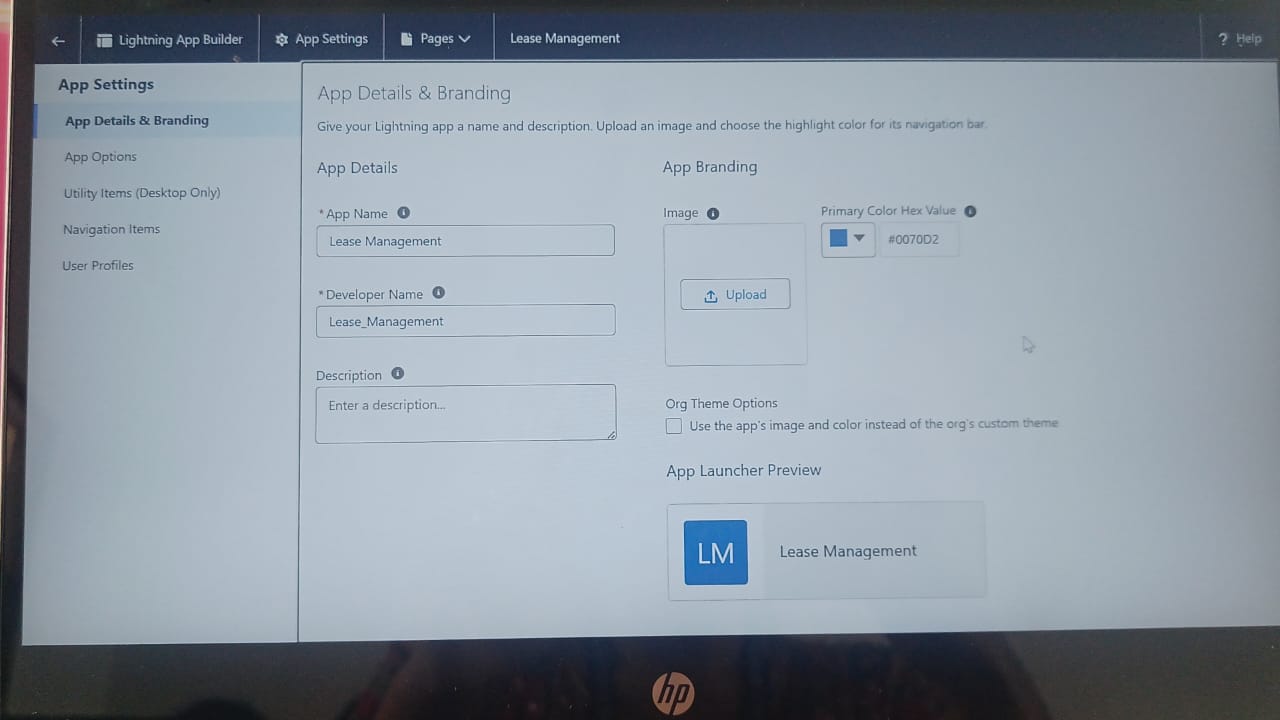


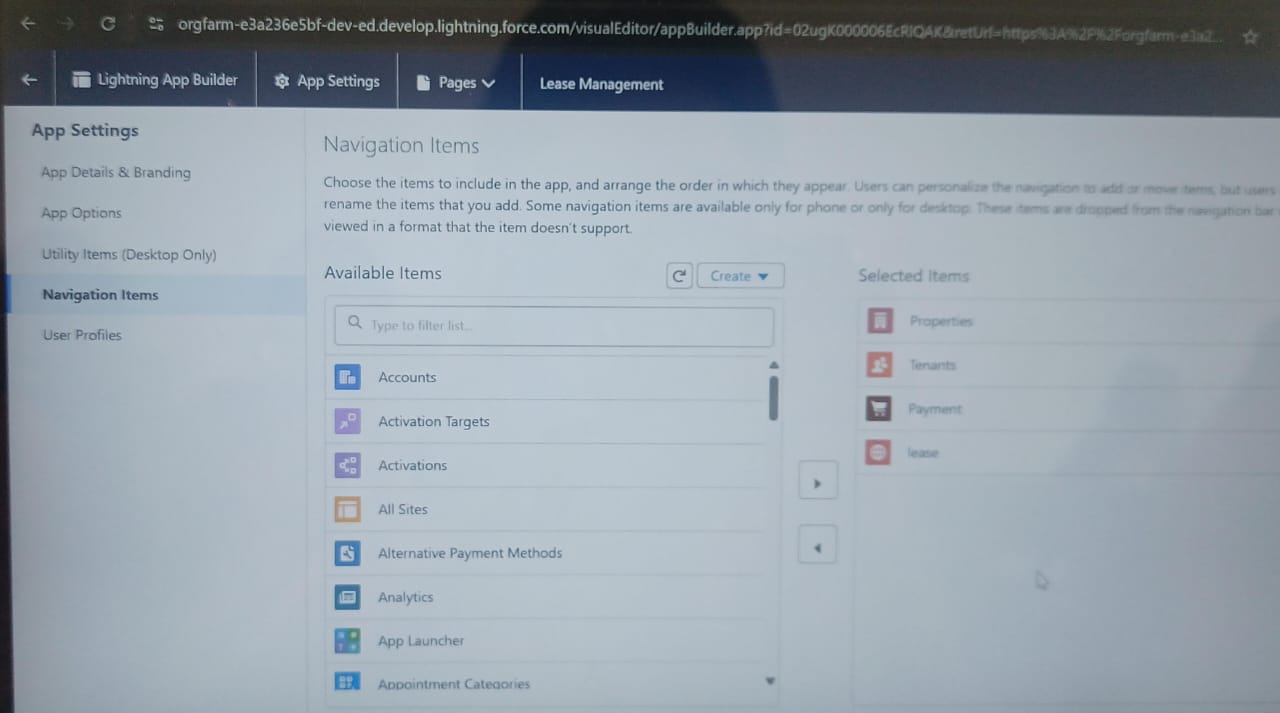


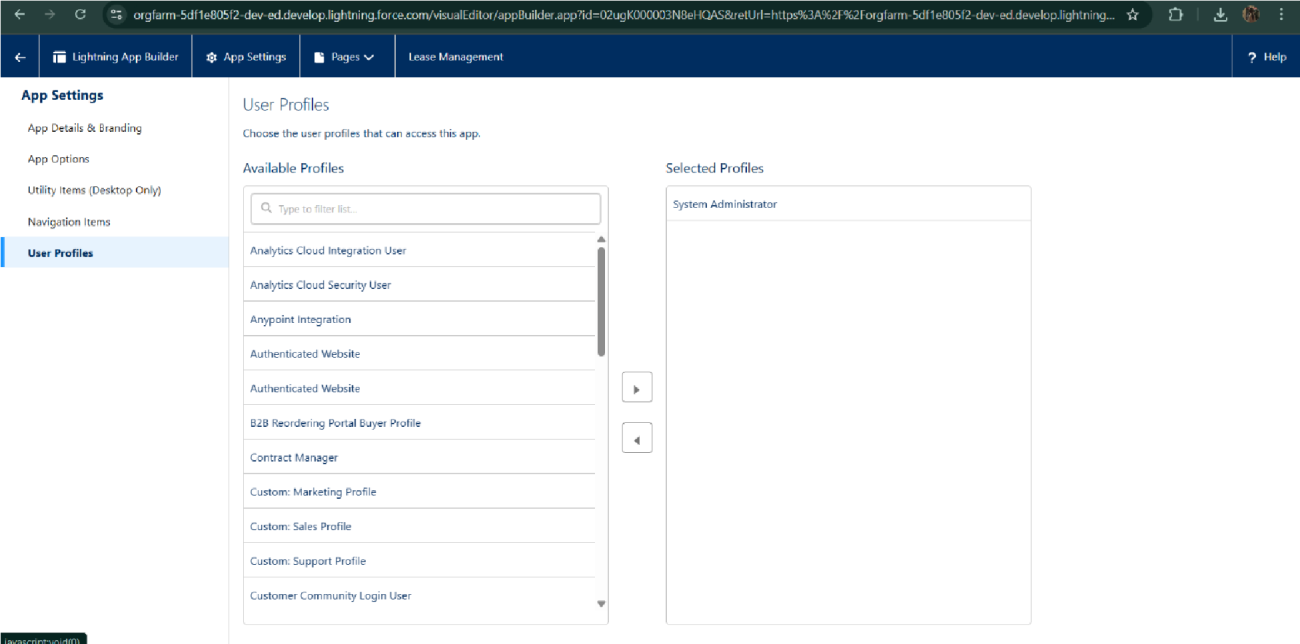


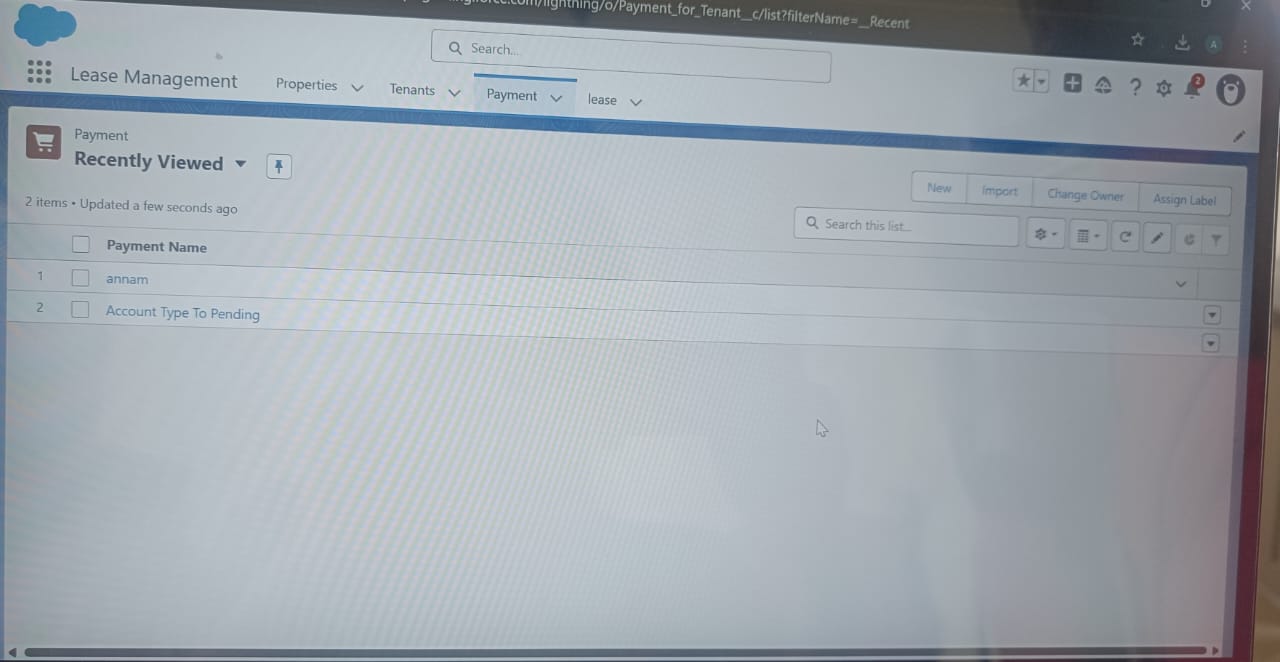


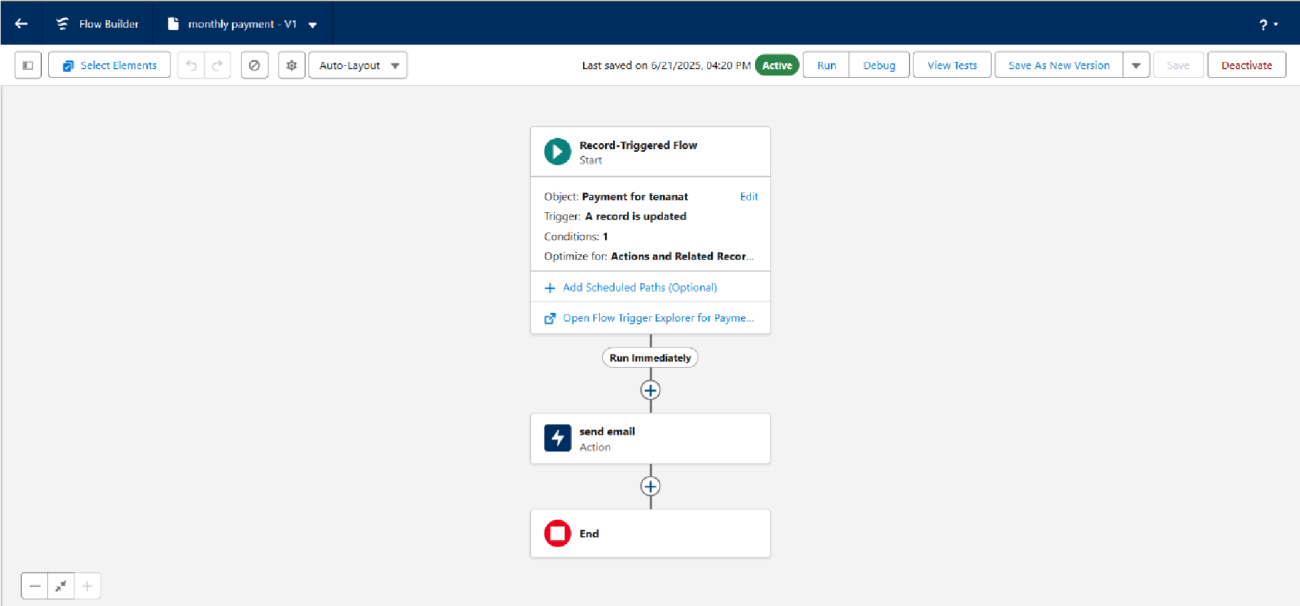
* + - DevelopedLightningAppwithrelevanttabs

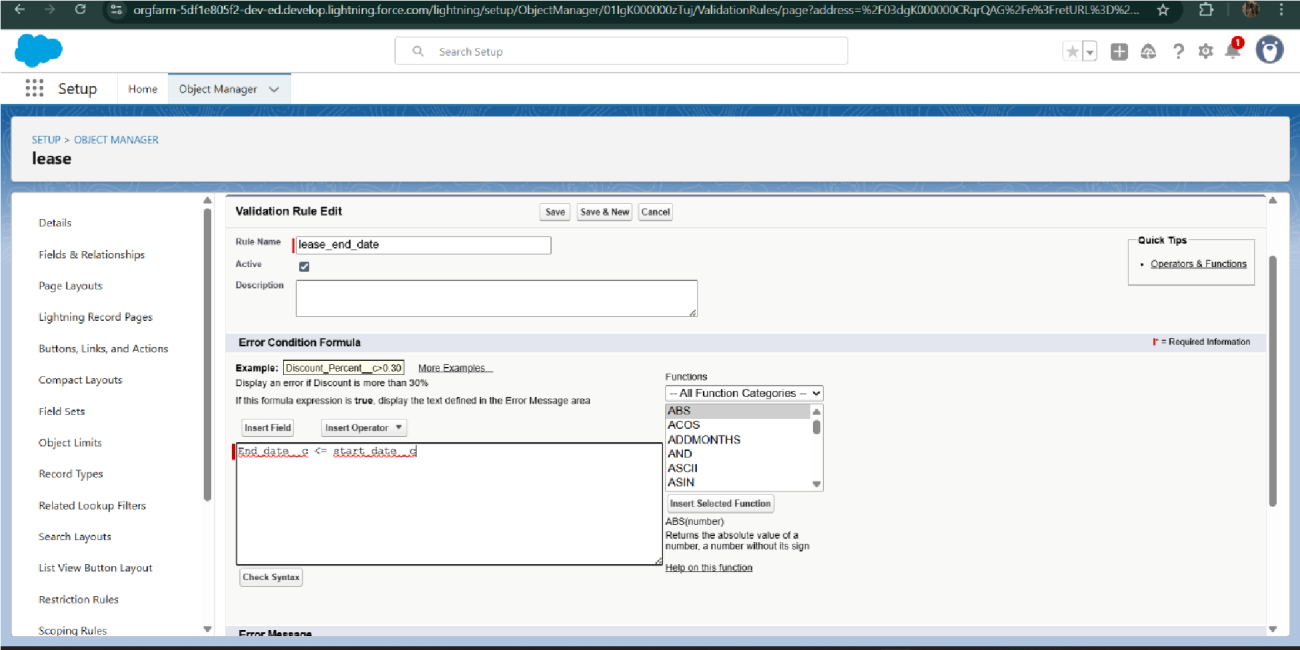


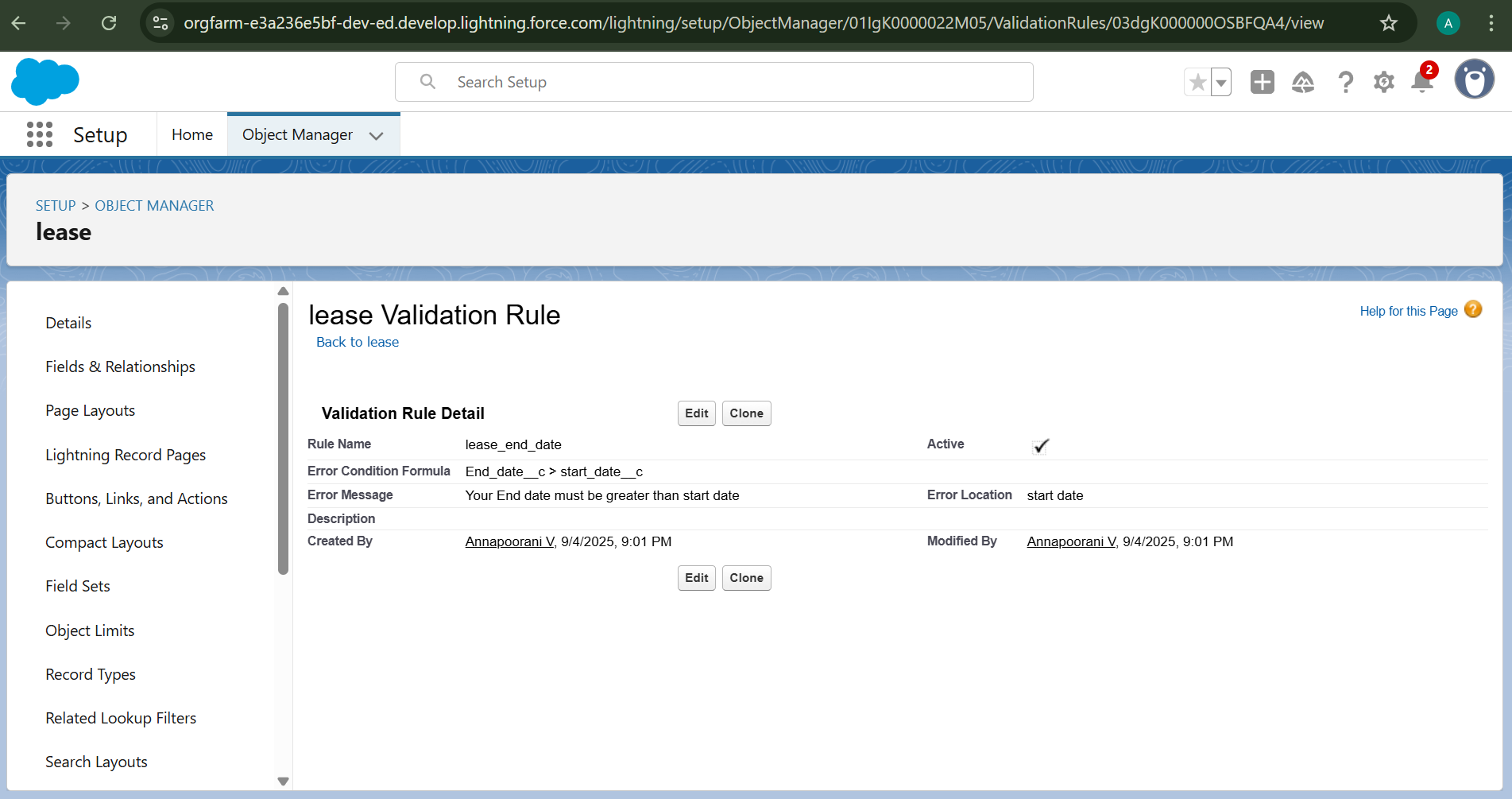




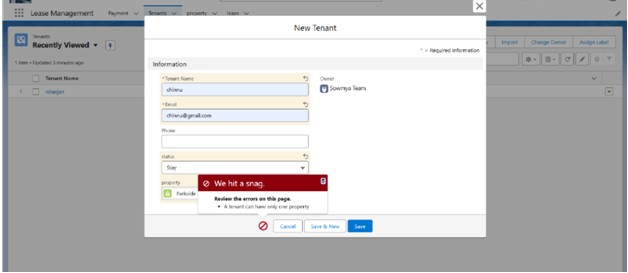


* + - ImplementedFlowsformonthlyrentandpaymentsuccess
    - TocreateavalidationruletoaLeaseObject

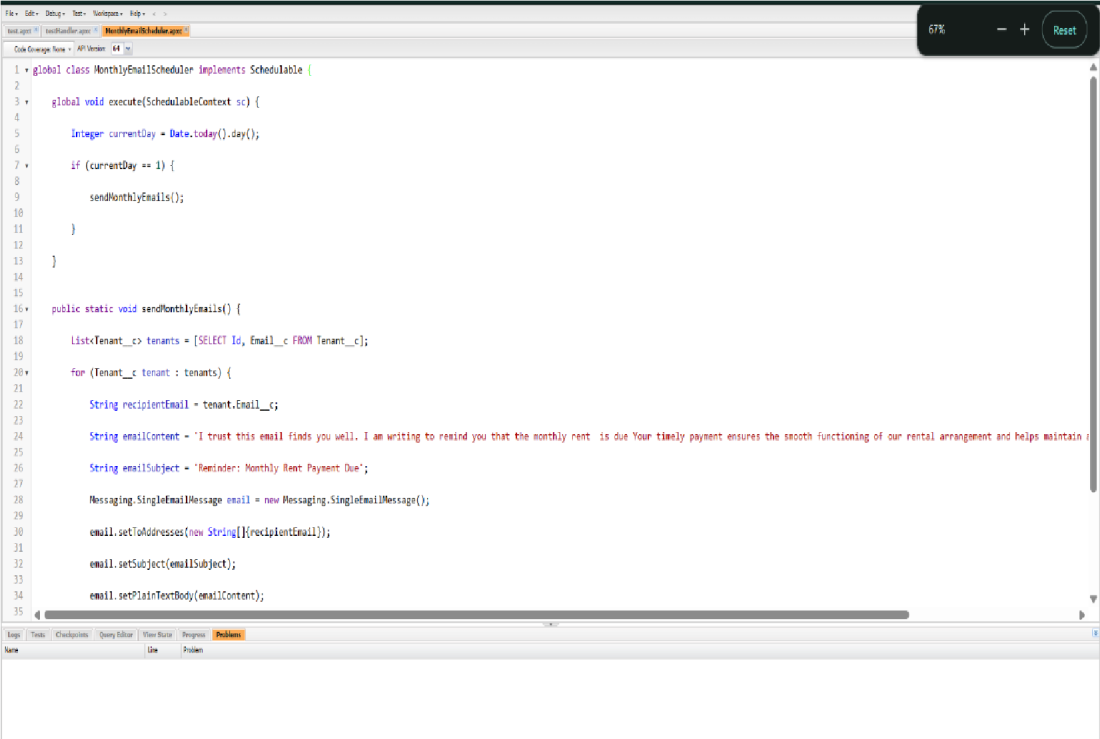




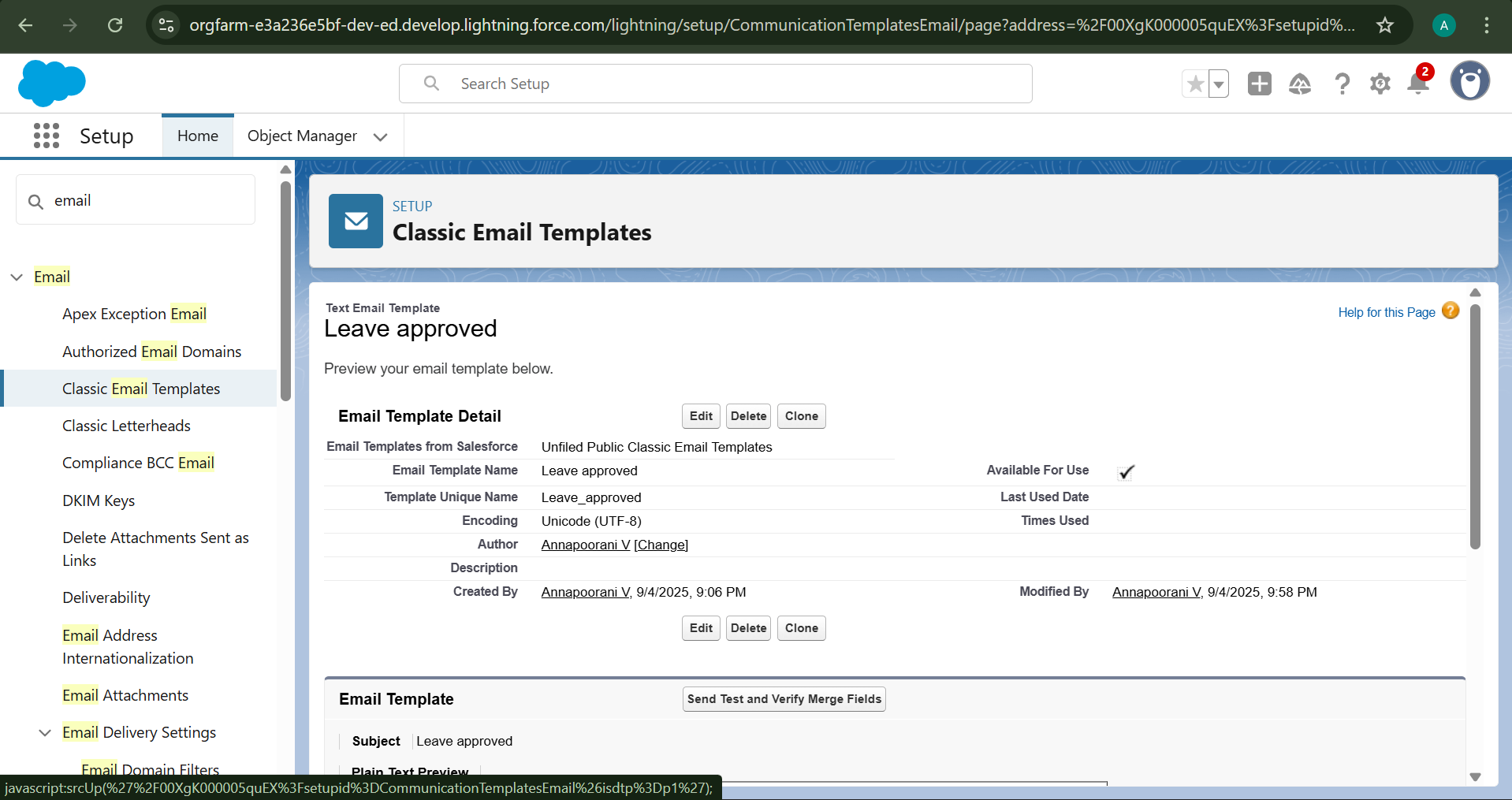
AddedApextrigger torestrictmultiple tenantsperproperty

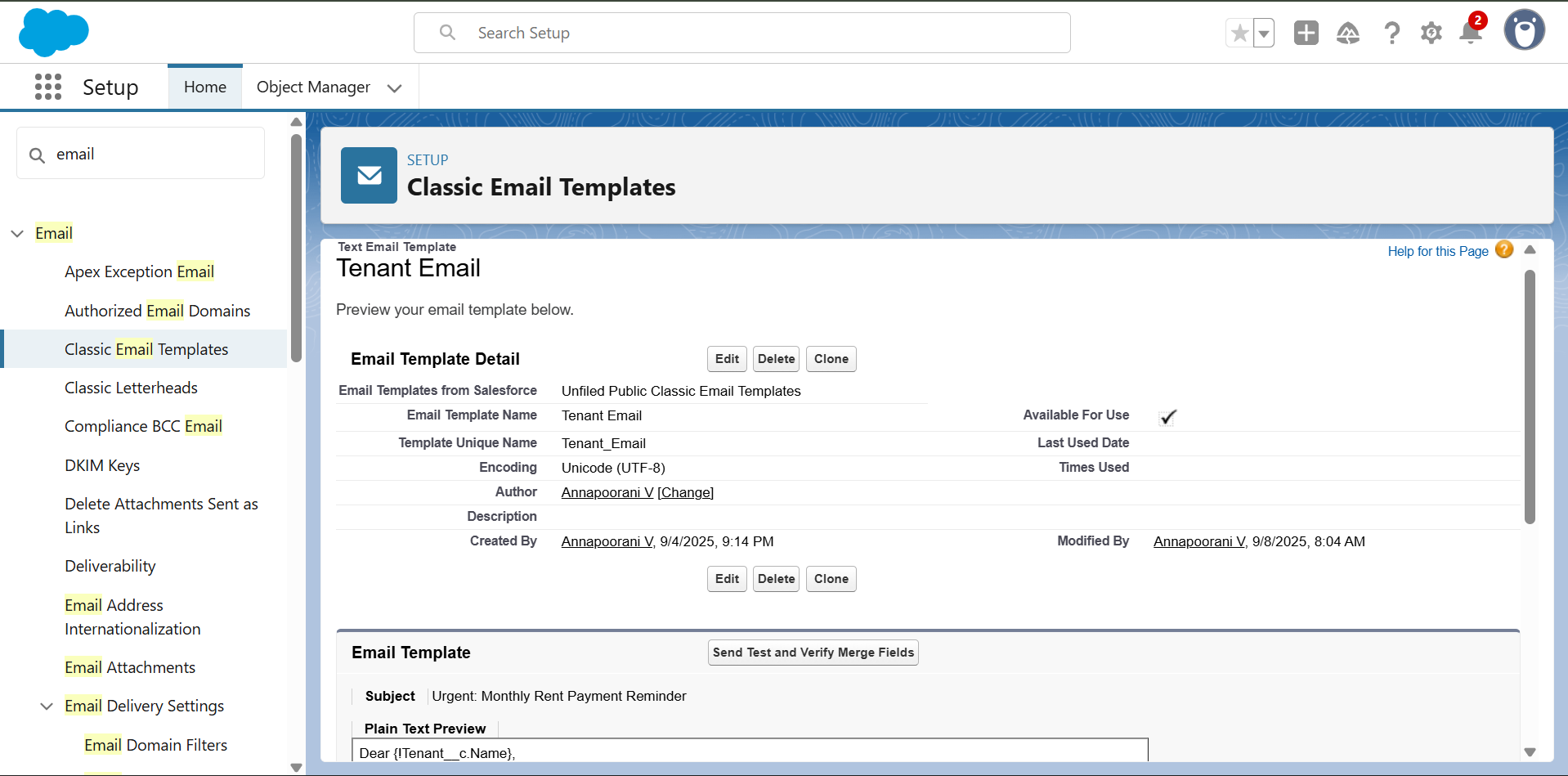


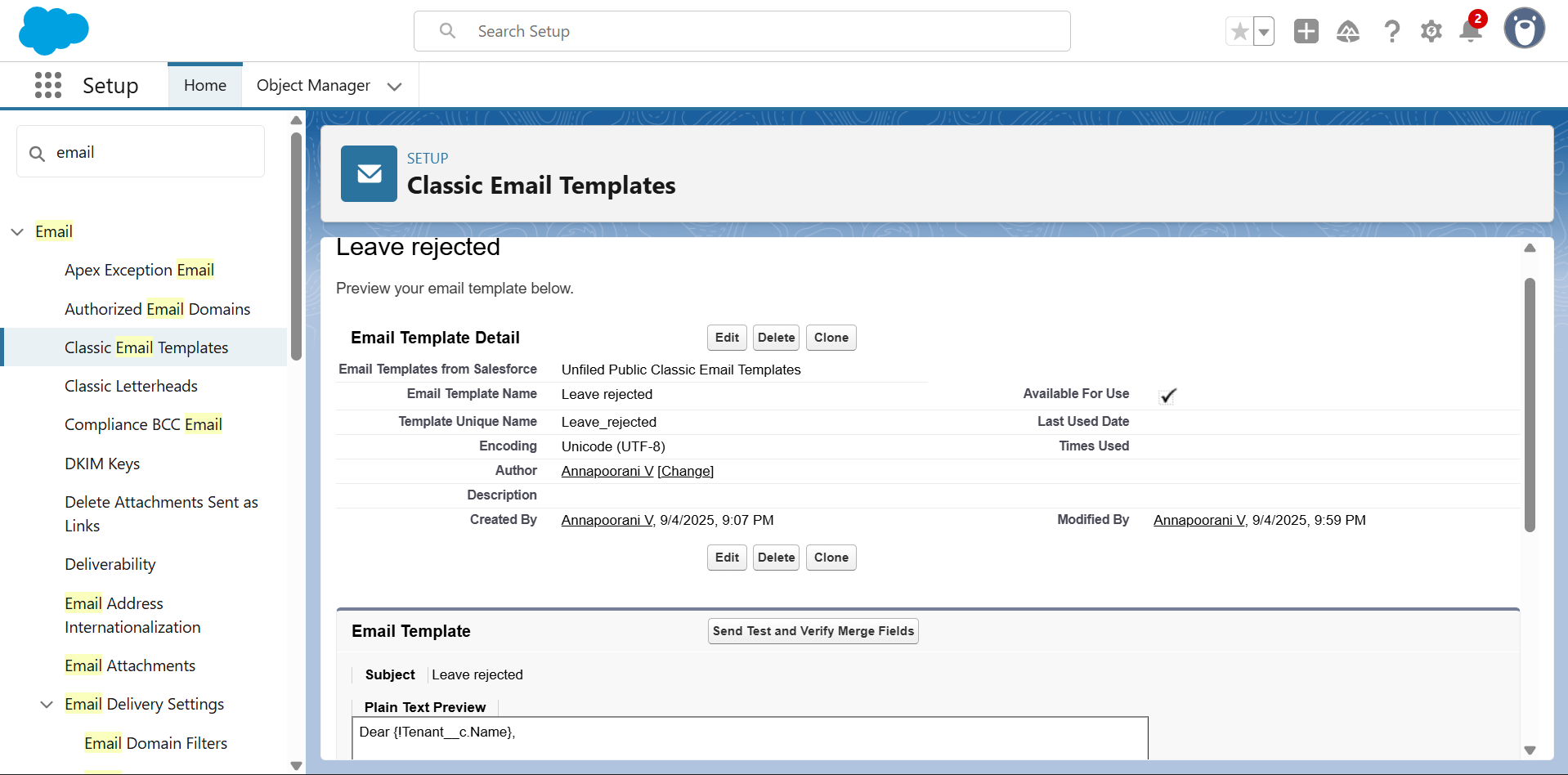
* + - ScheduledmonthlyreminderemailsusingApex class

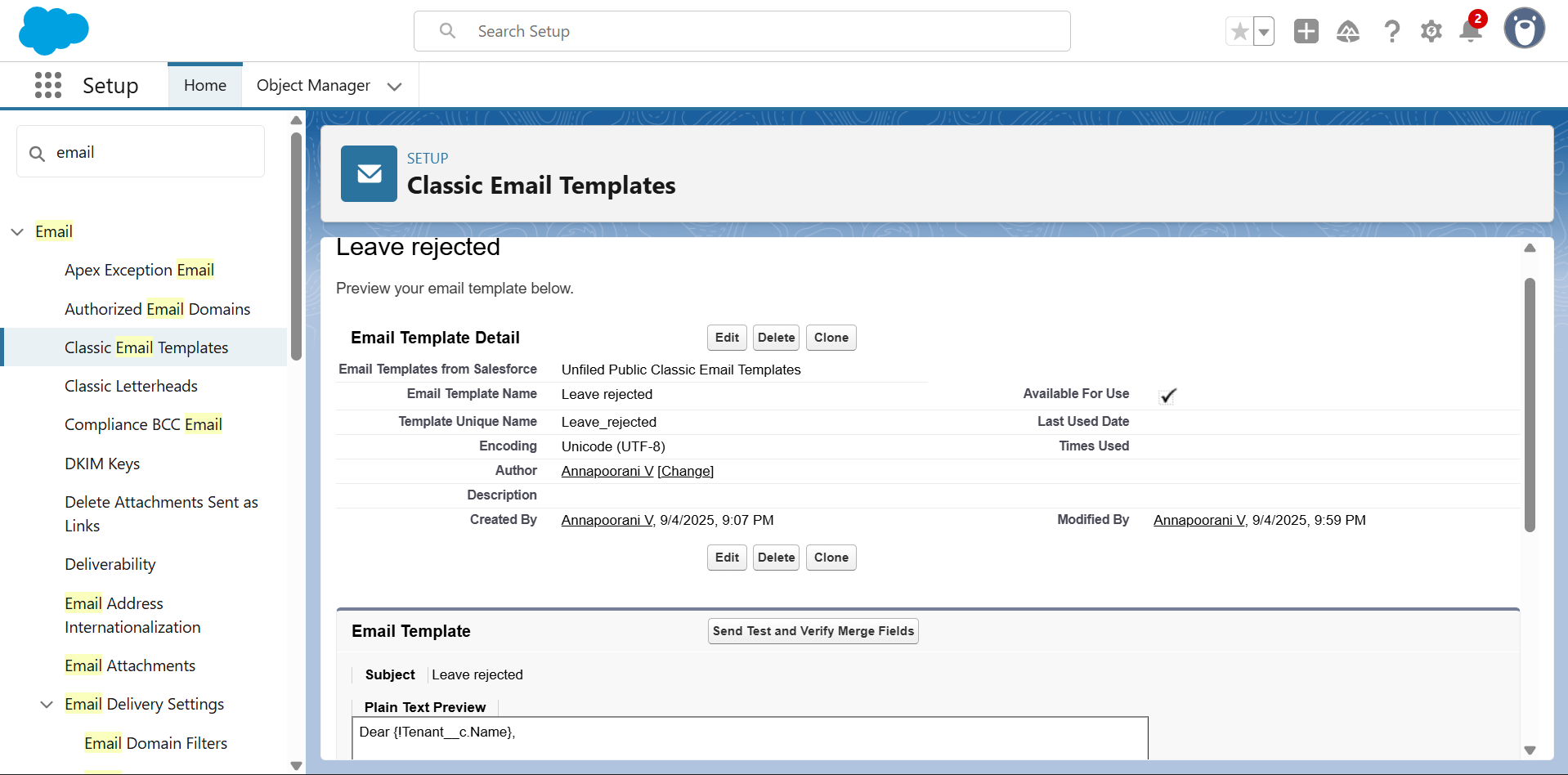


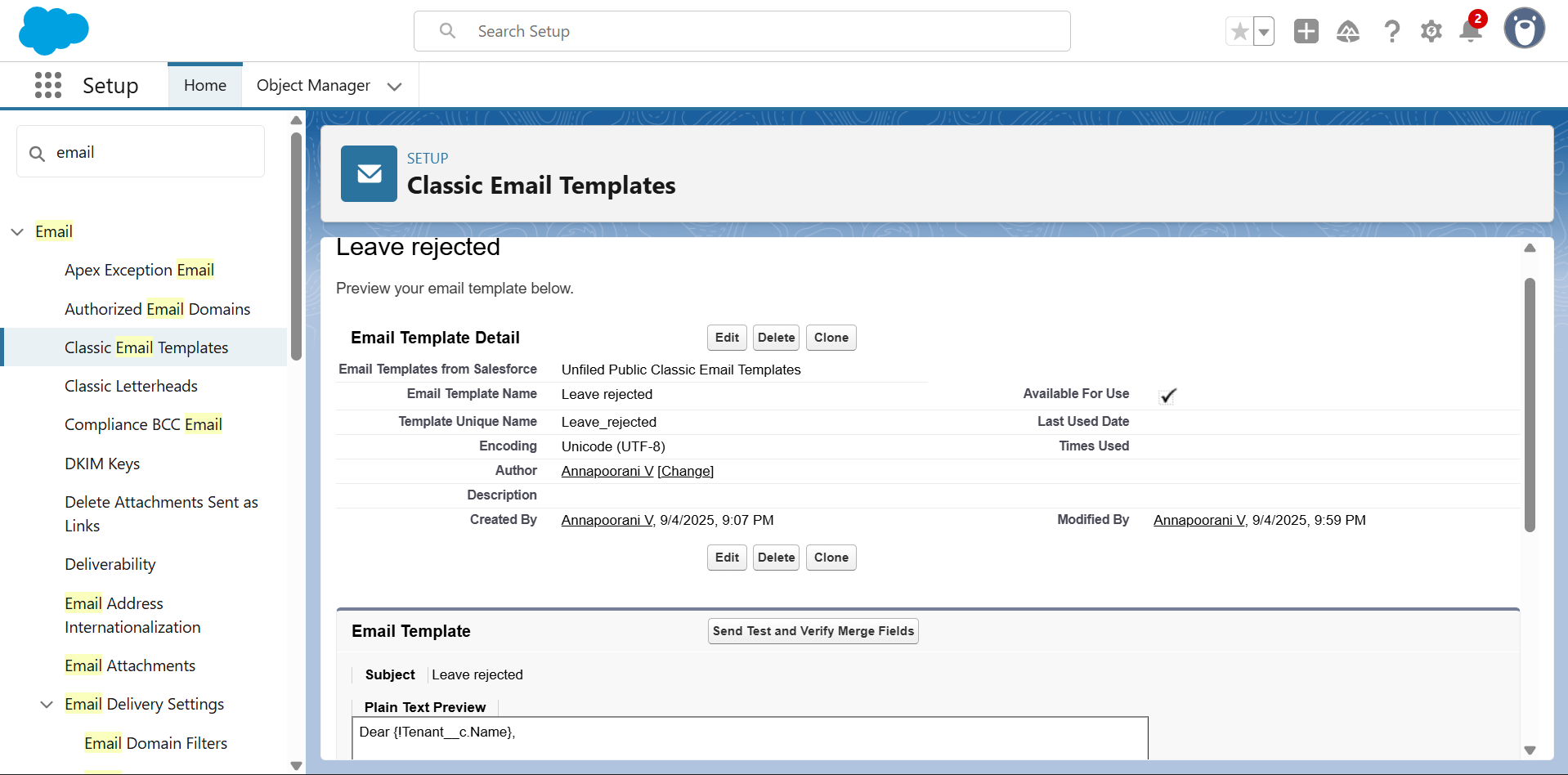
* + - Builtandtestedemailtemplatesforleaverequest,approval,rejection,payment,and reminders





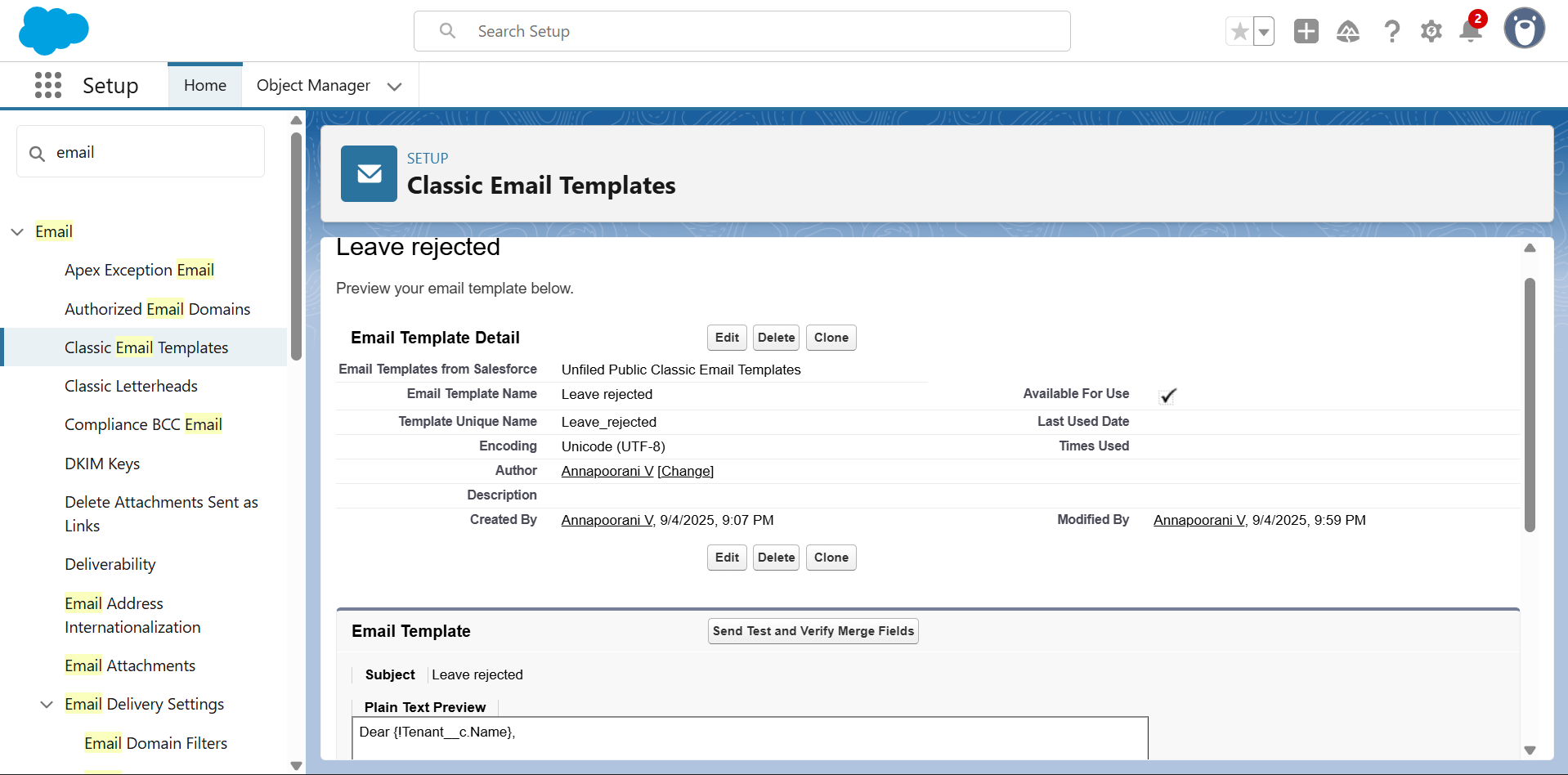




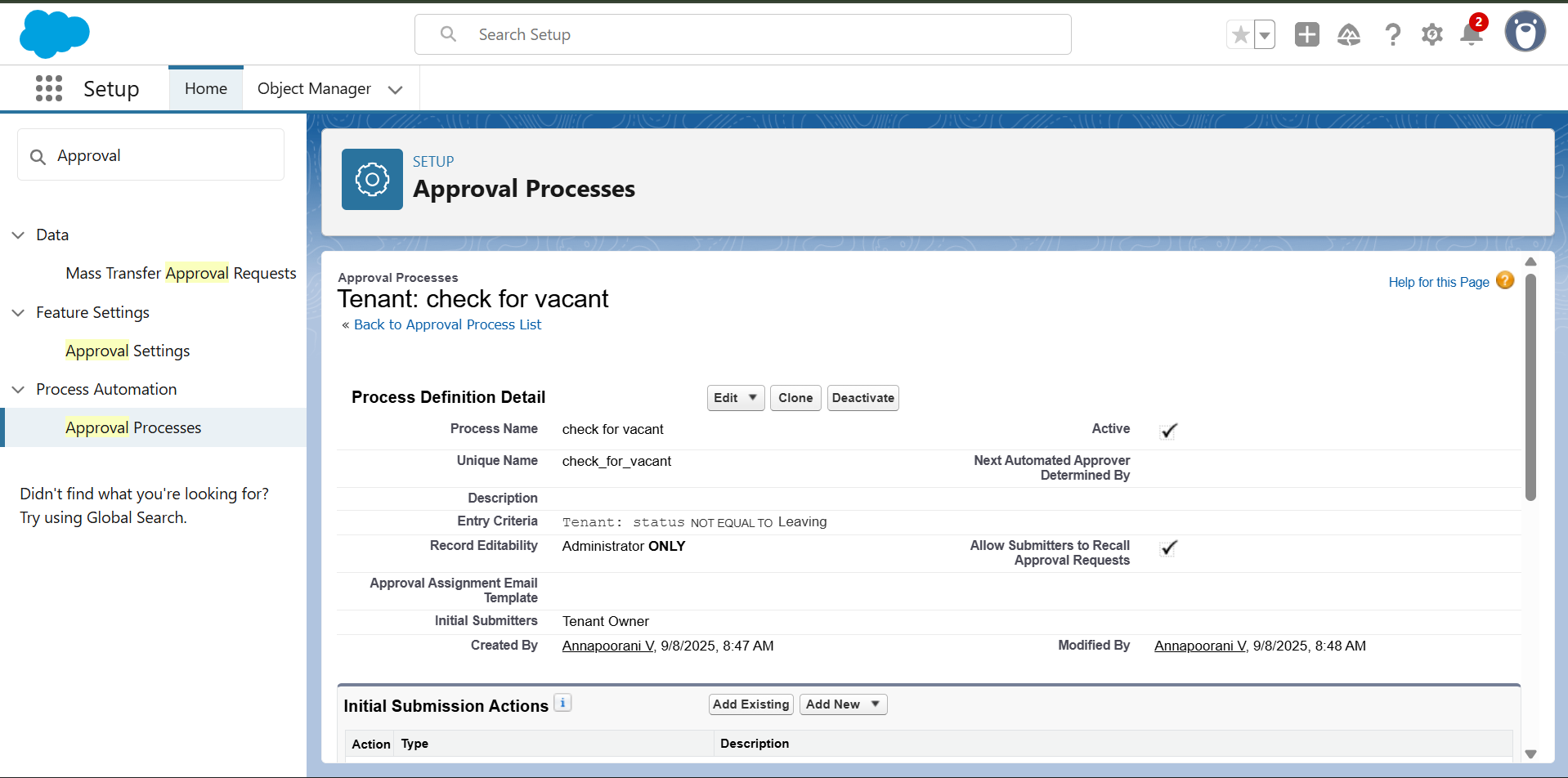


* + - ApprovalProcesscreation

ForTenantLeaving:

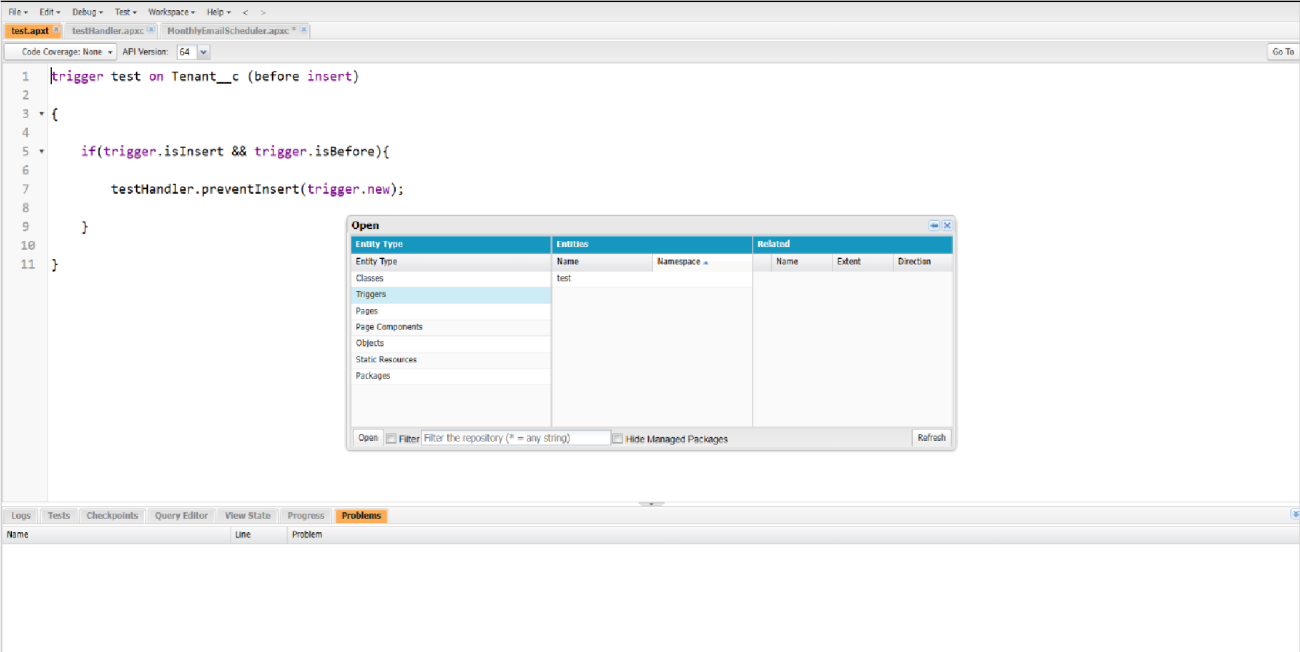


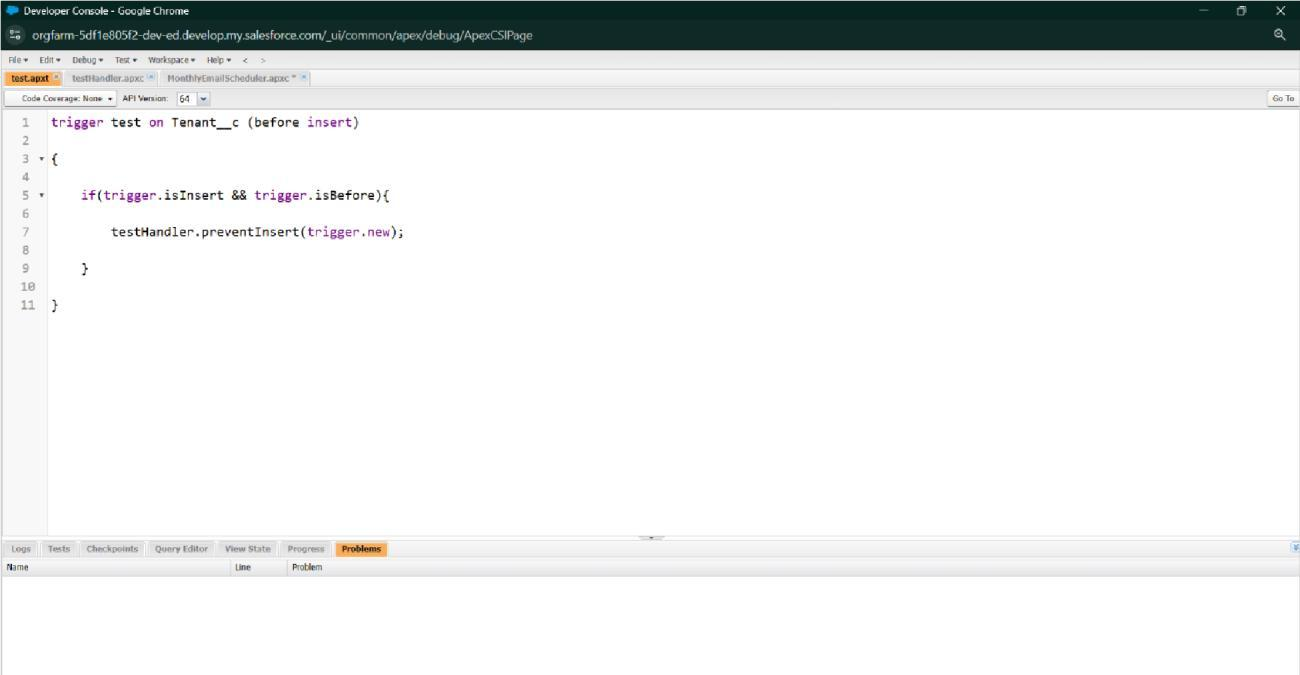
ForCheckforVacant:



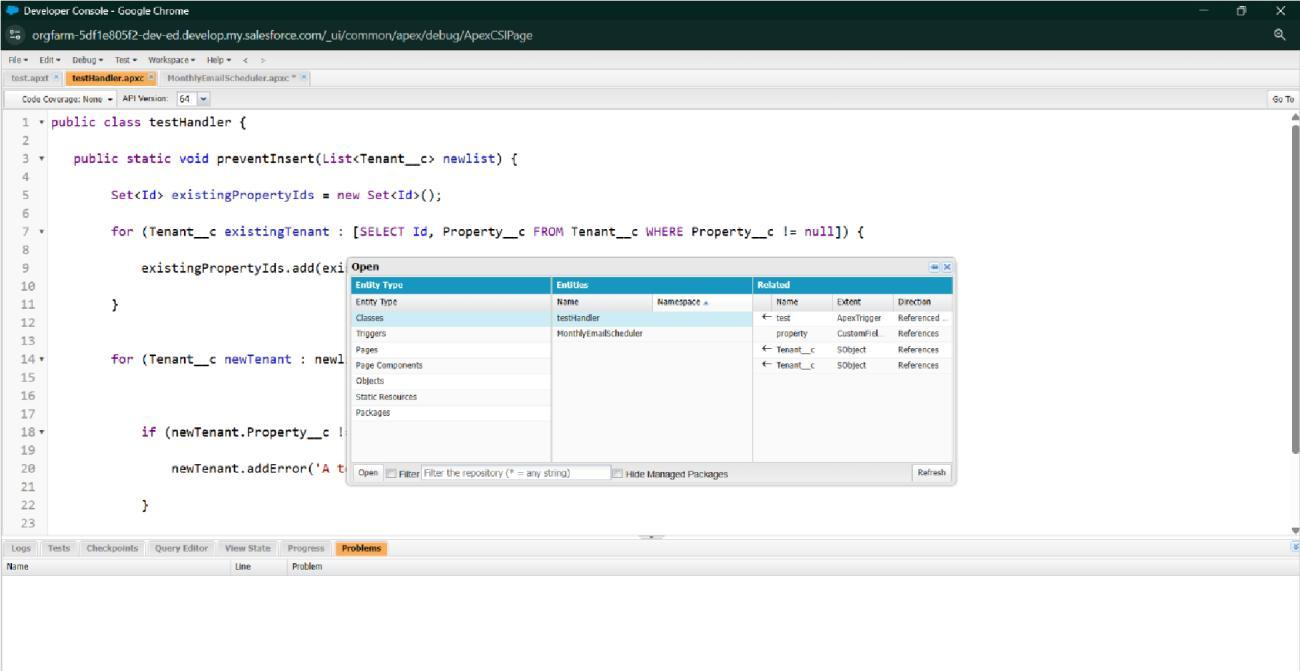
* + - ApexTrigger

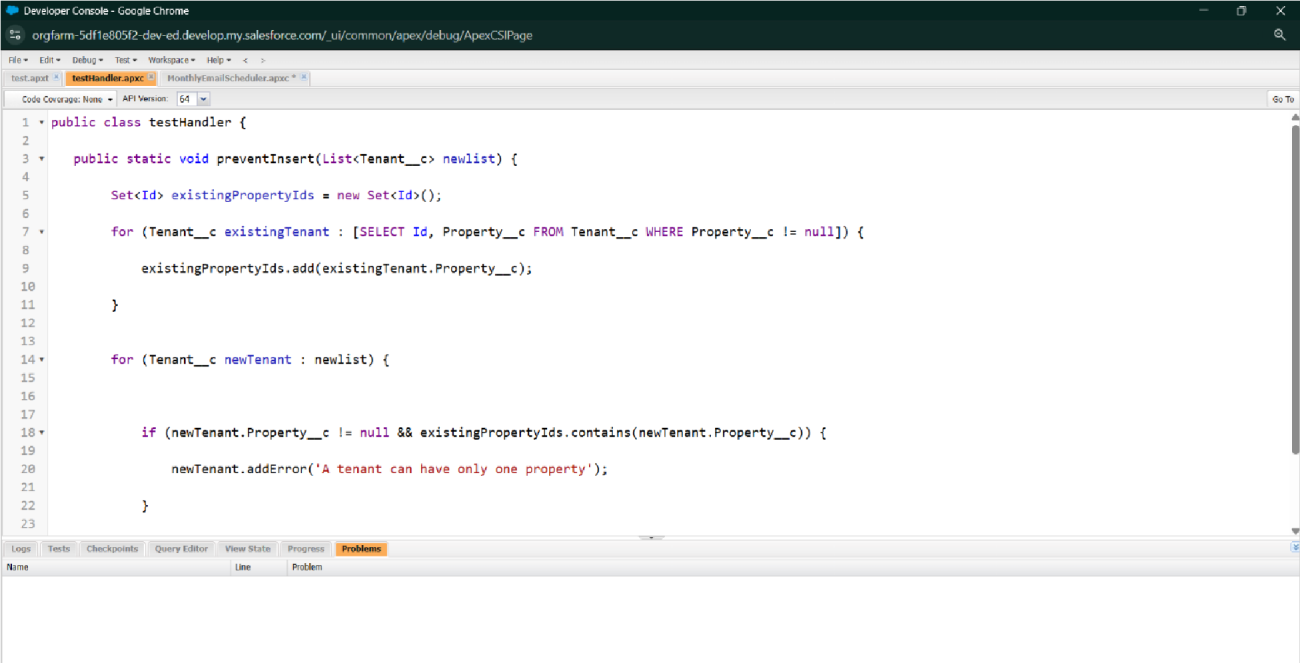
CreateanApexTrigger



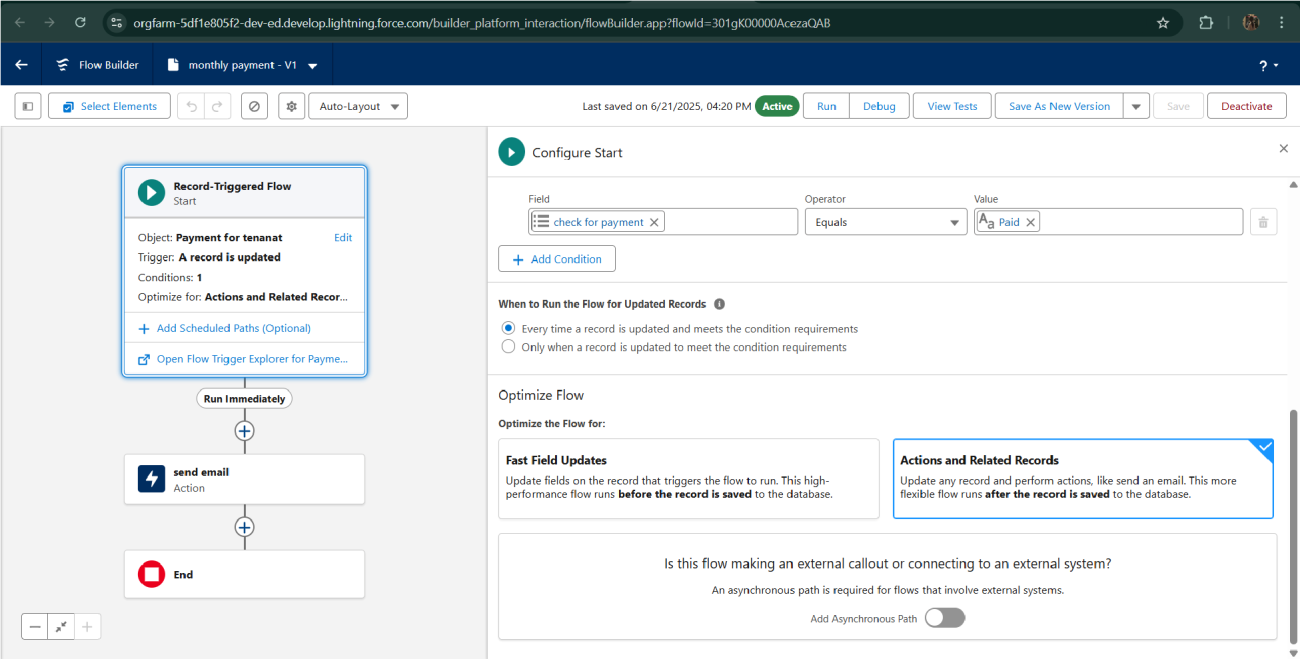


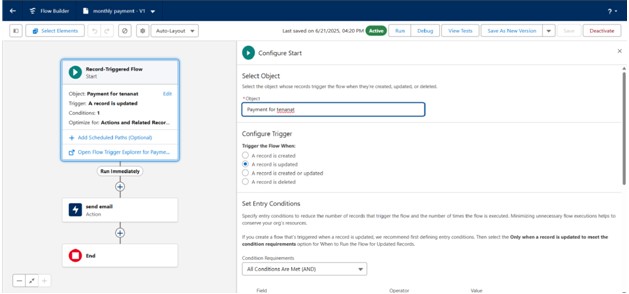
CreateanApex Handlerclass



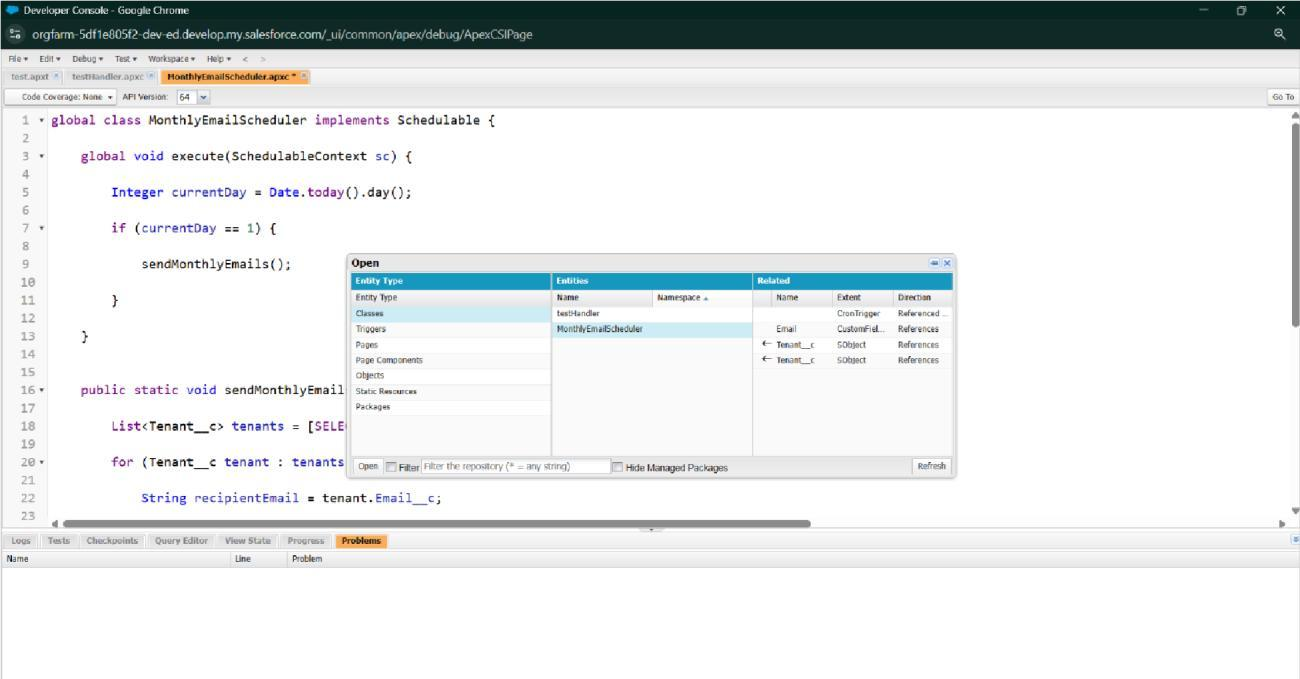


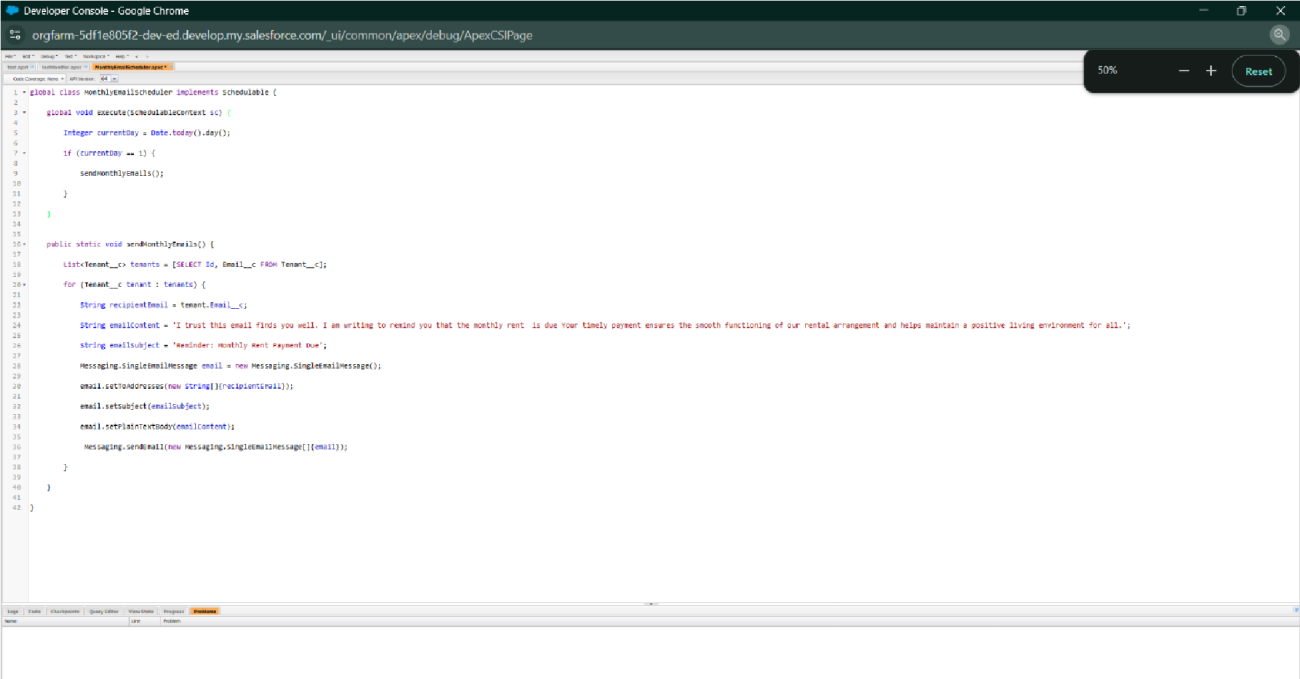
* + - FLOWS



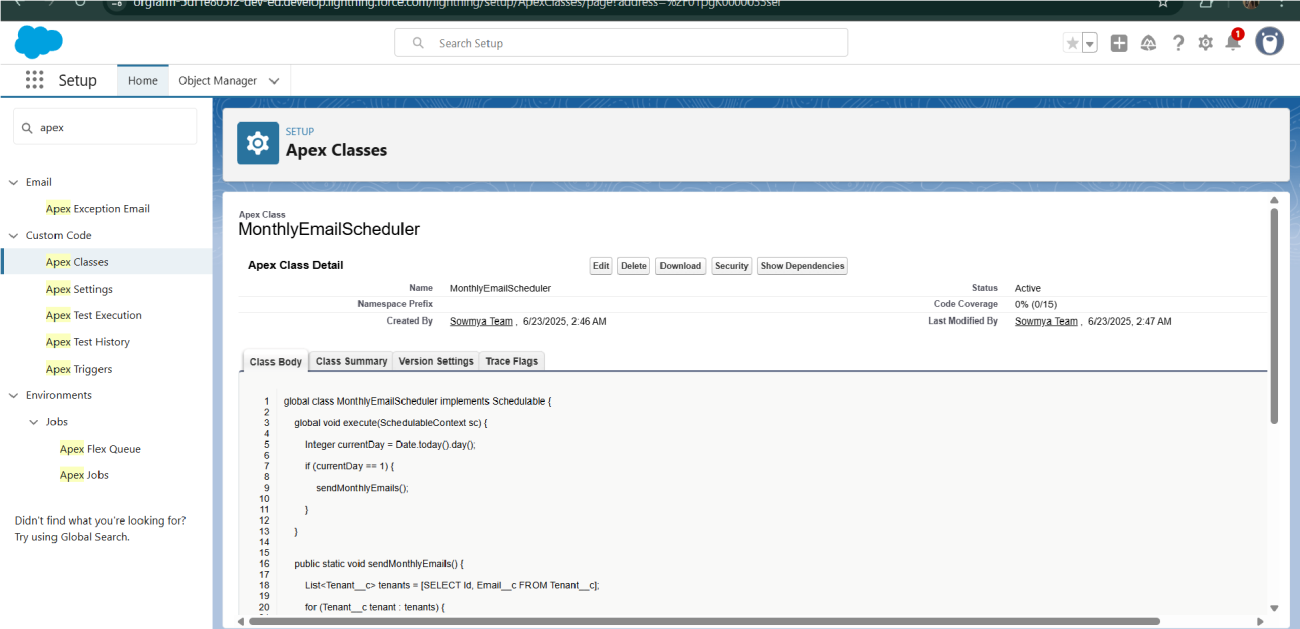


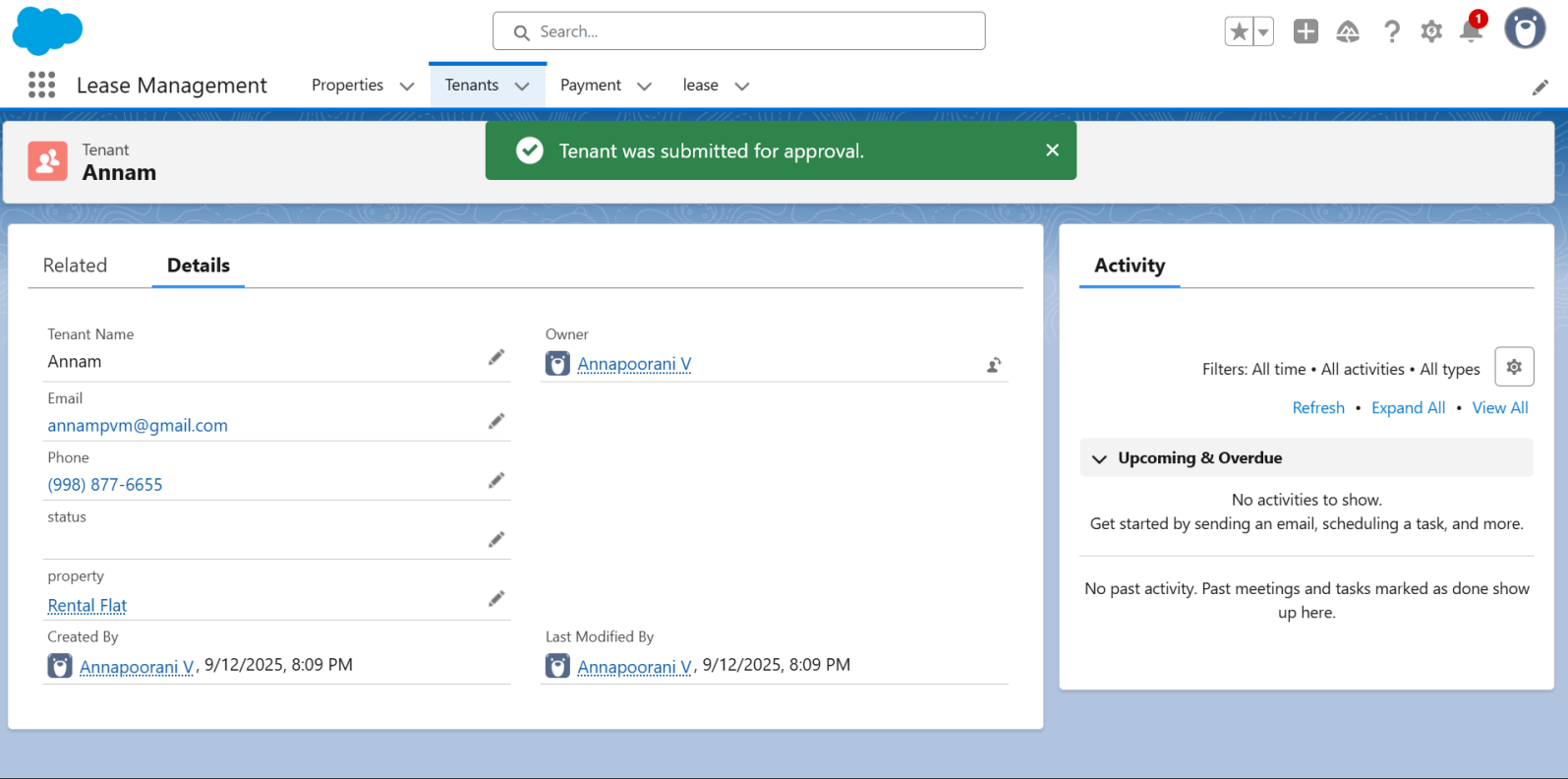
* + - Schedule class: CreateanApexClass

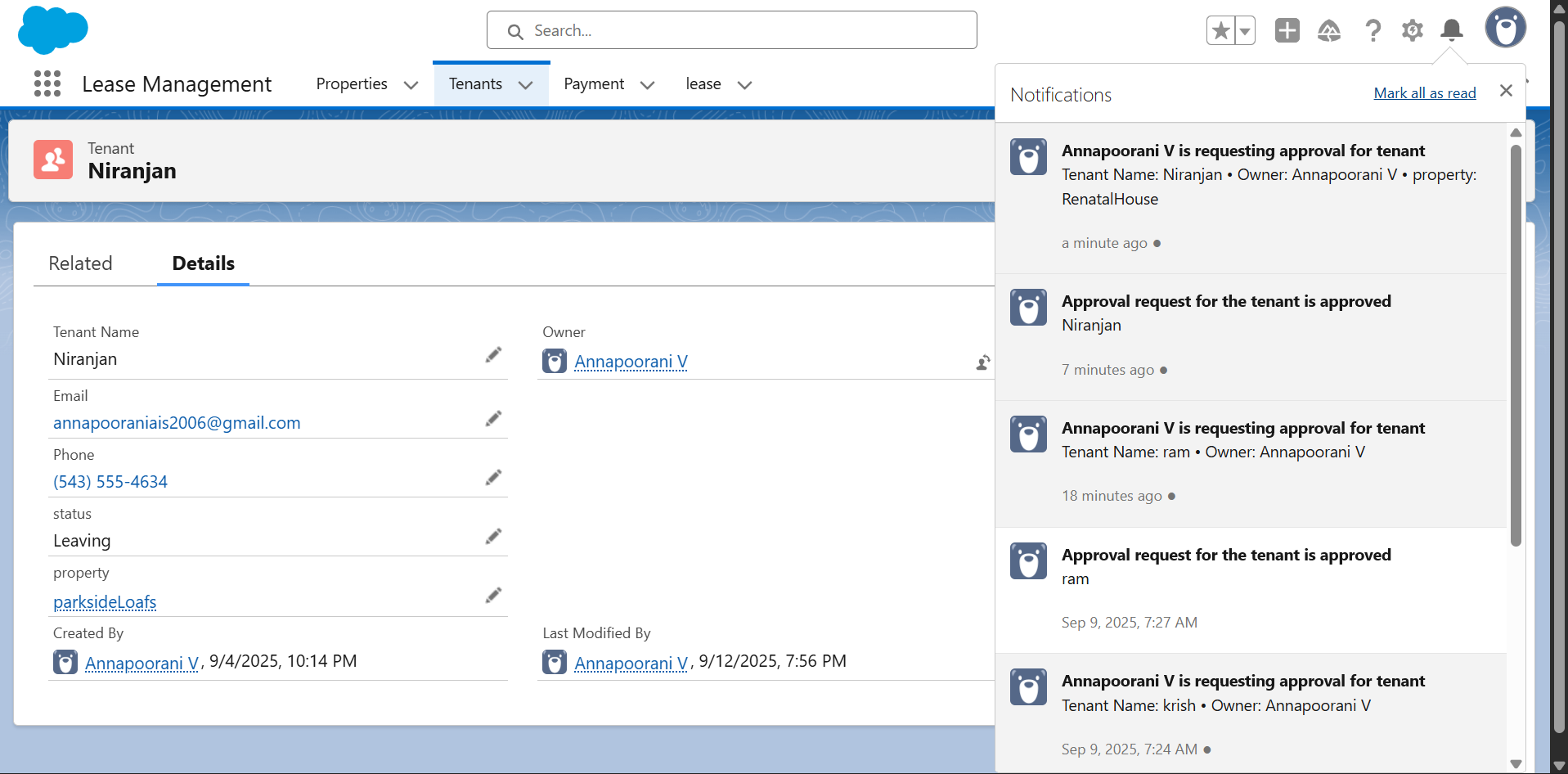




ScheduleApex class



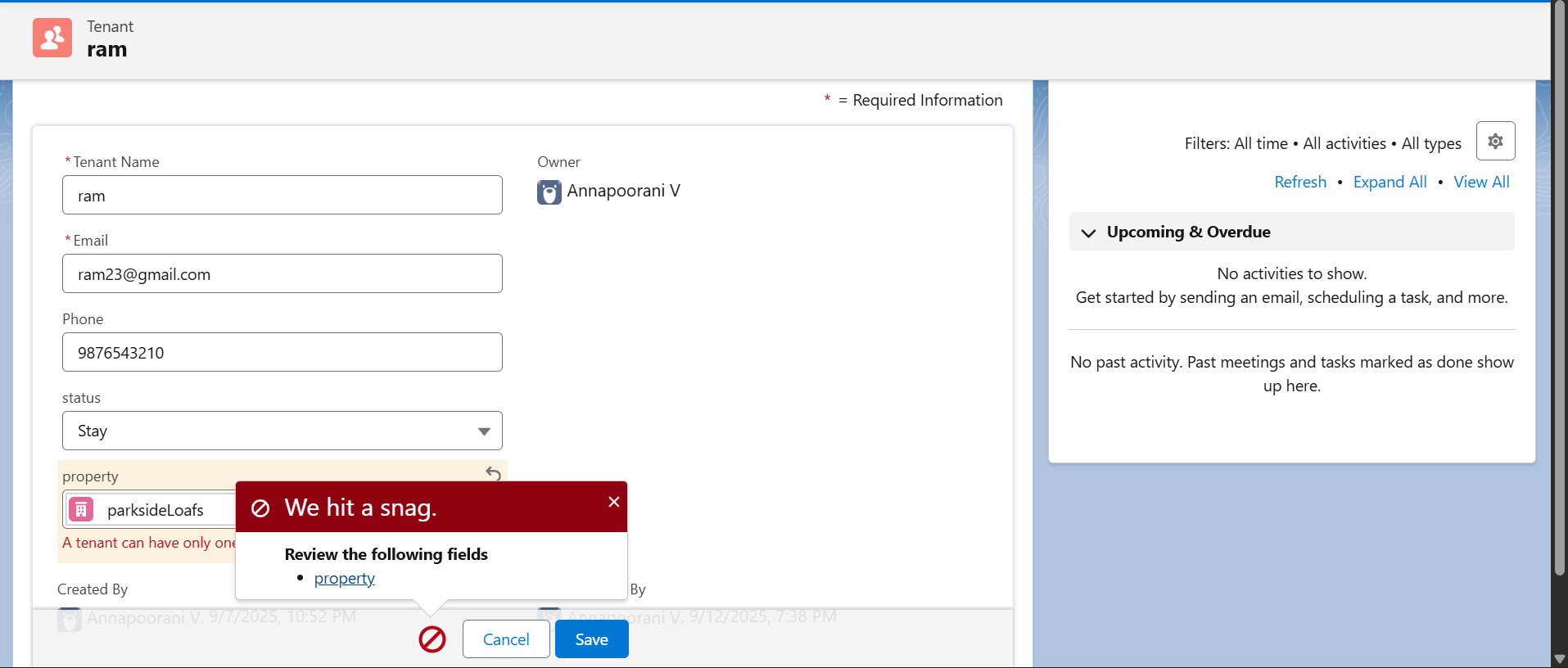




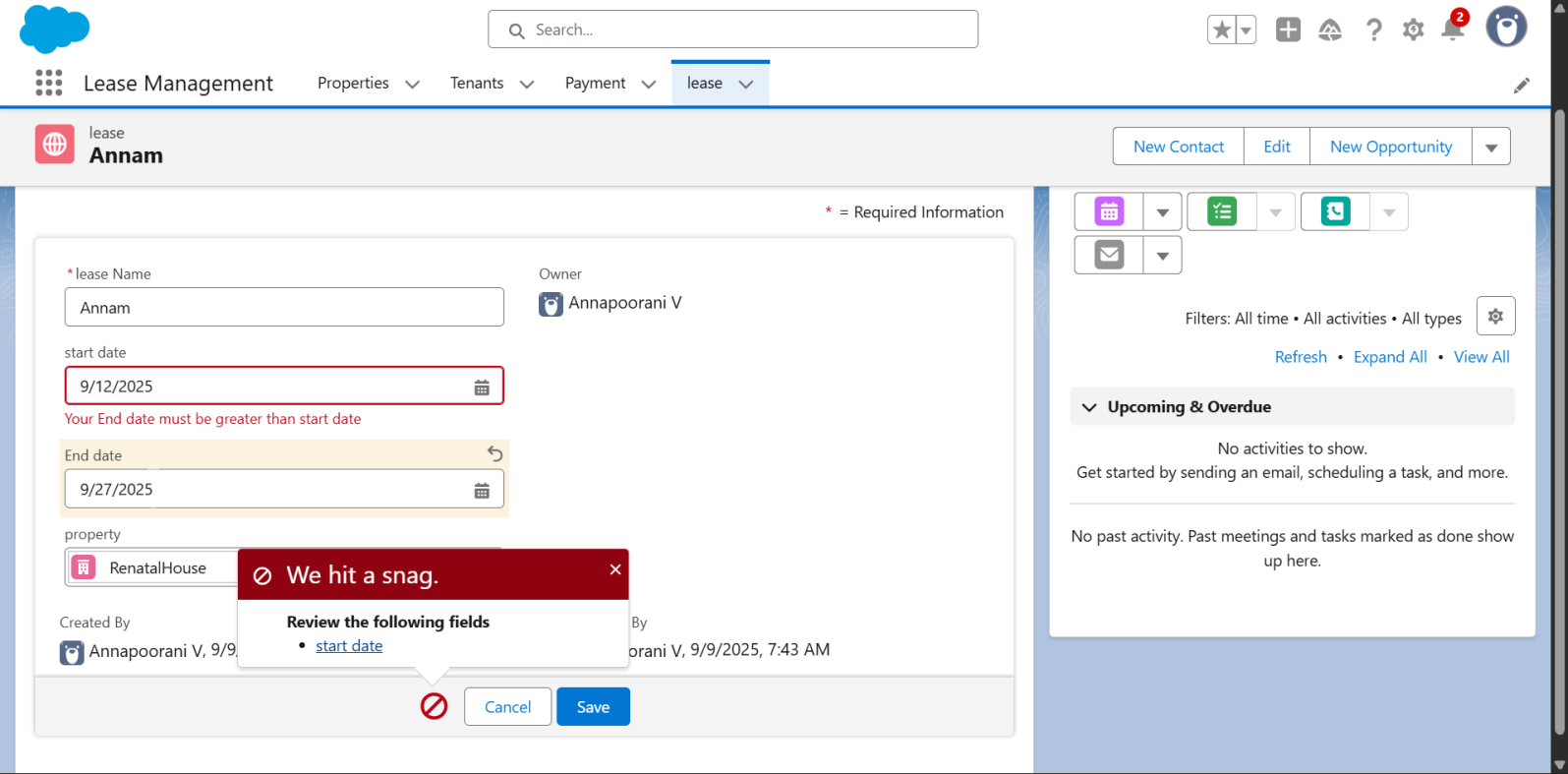
# FUNCTIONALANDPERFORMANCE TESTING

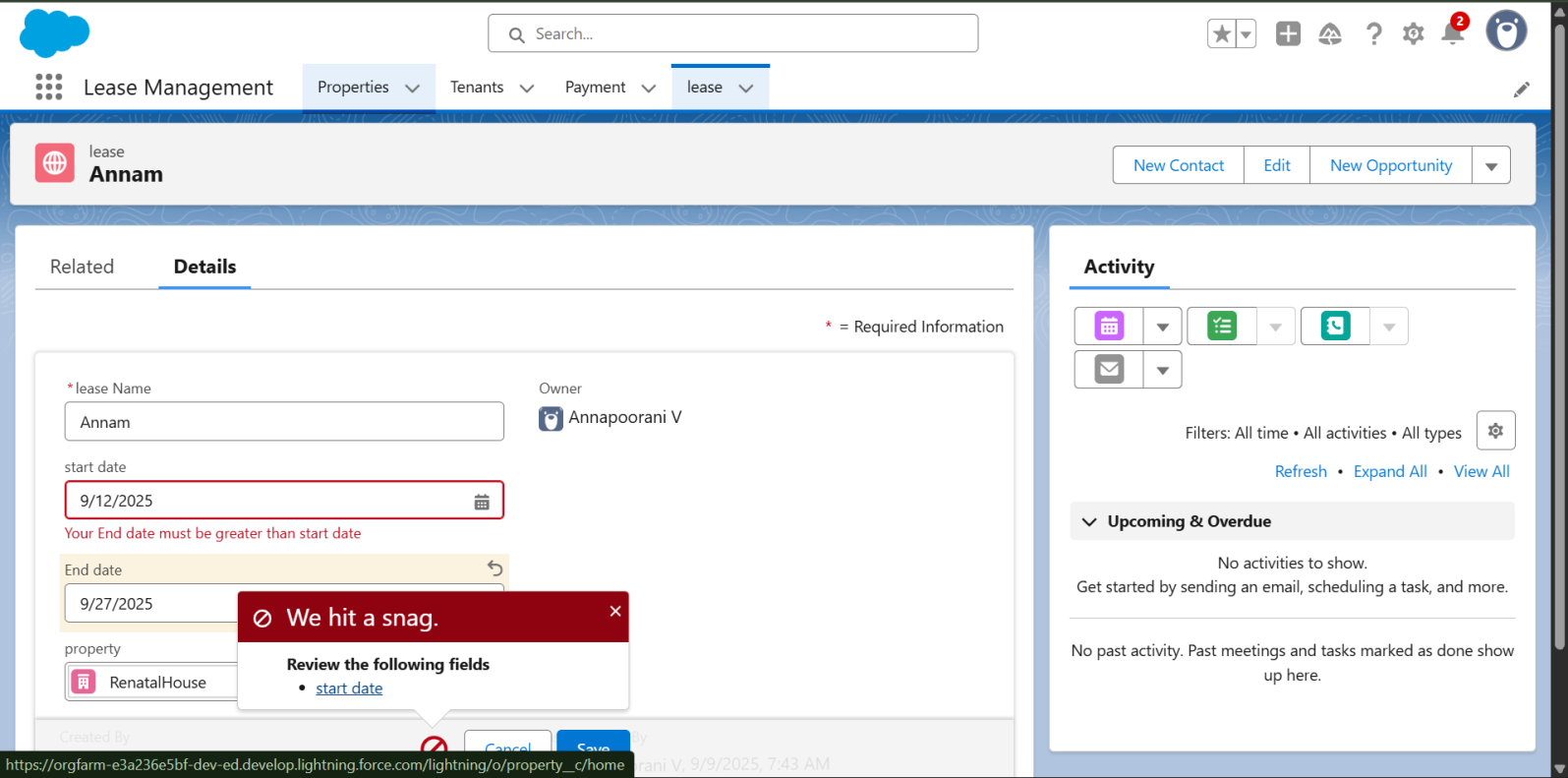
## PerformanceTesting

* + - Triggervalidationbyenteringduplicatetenant-property records

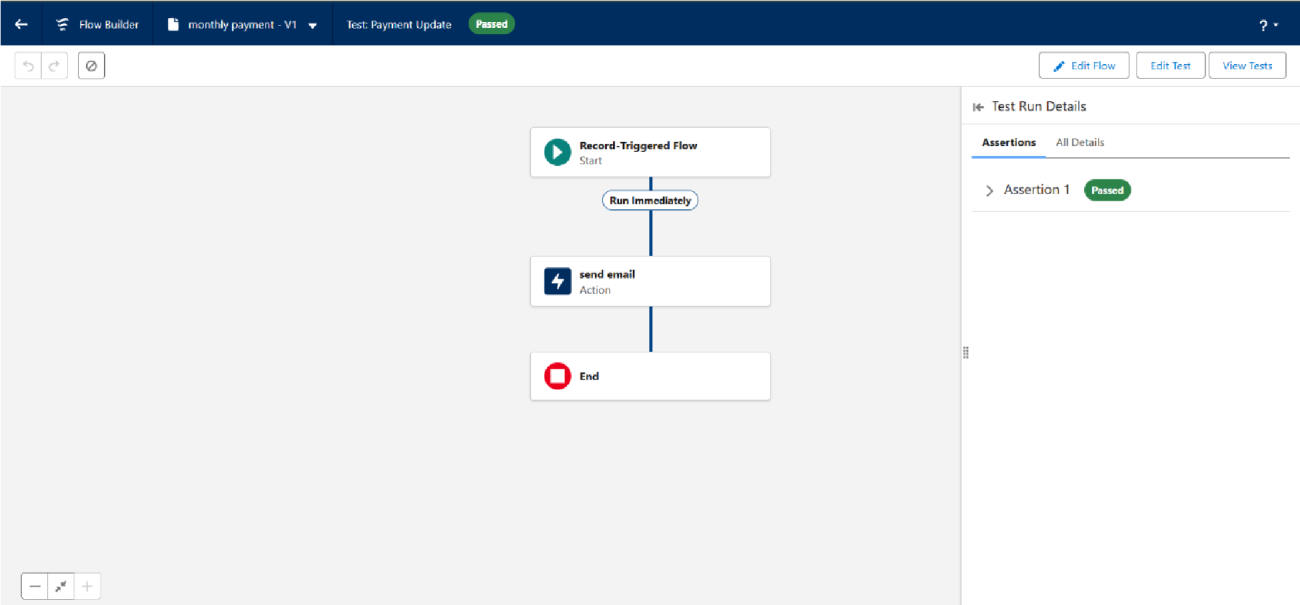


* + - ValidationRulechecking

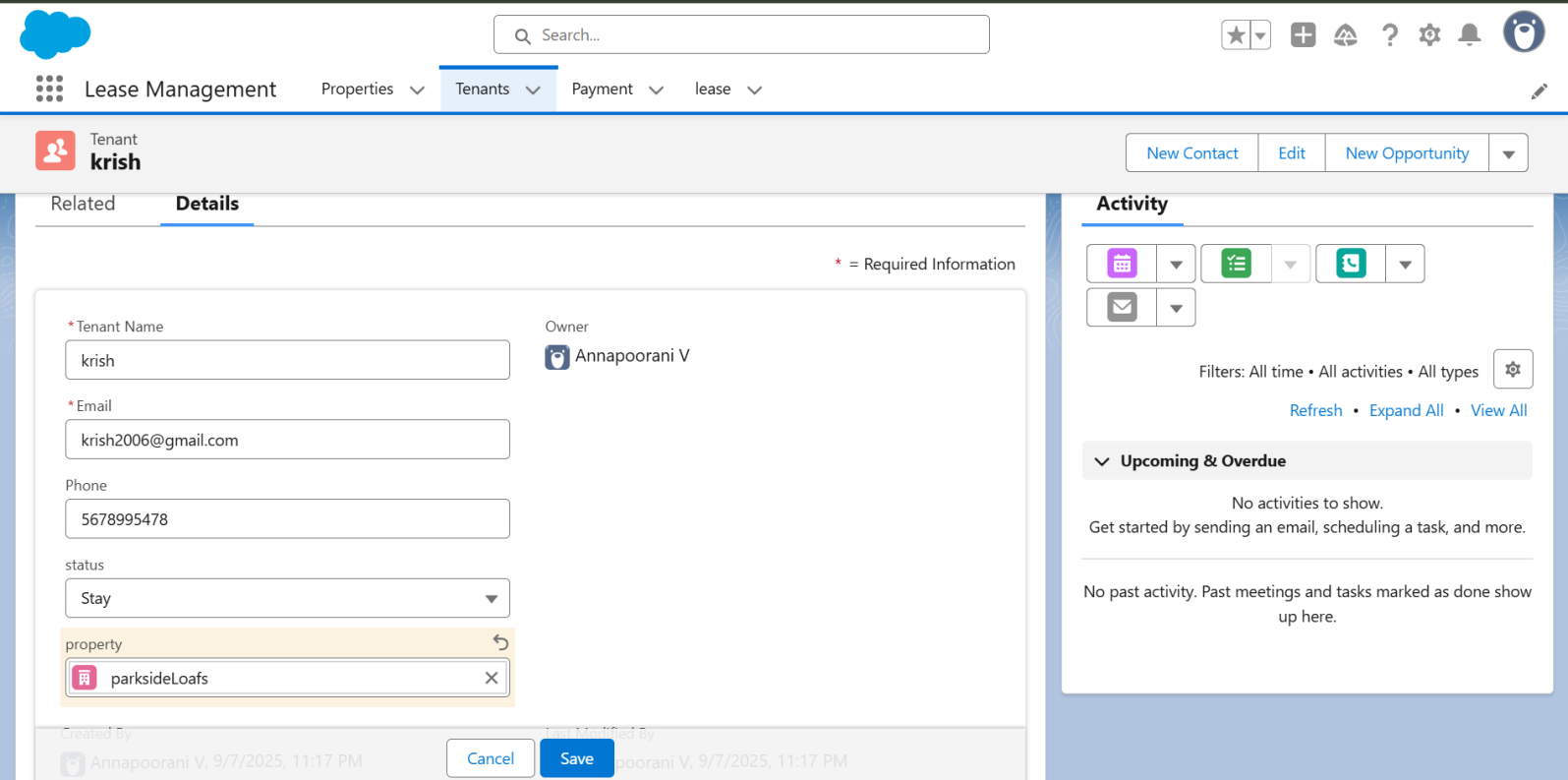


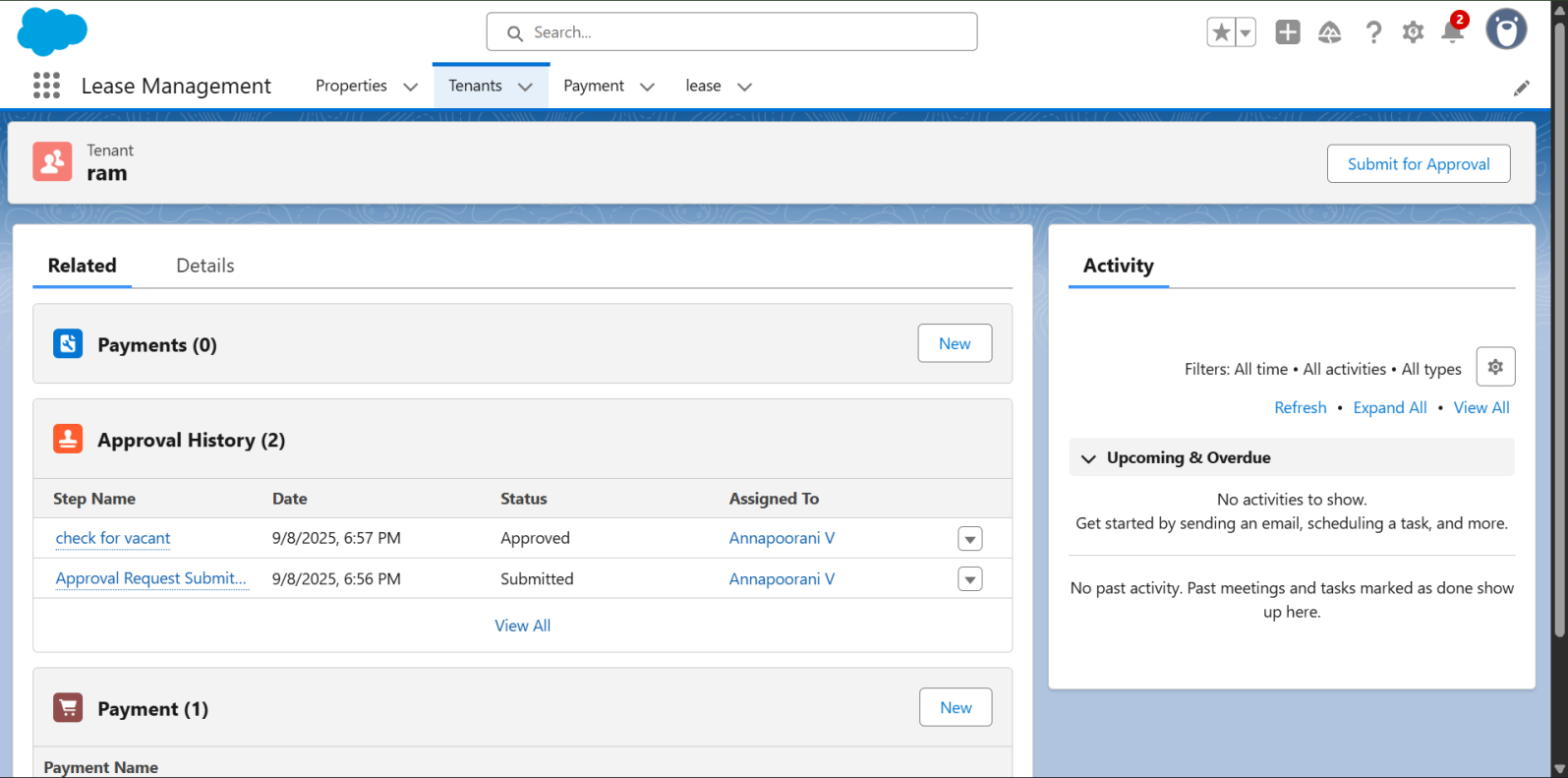


* + - Testflowsonpaymentupdate



* + - Approvalprocessvalidatedthroughemailalertsandstatusupdates

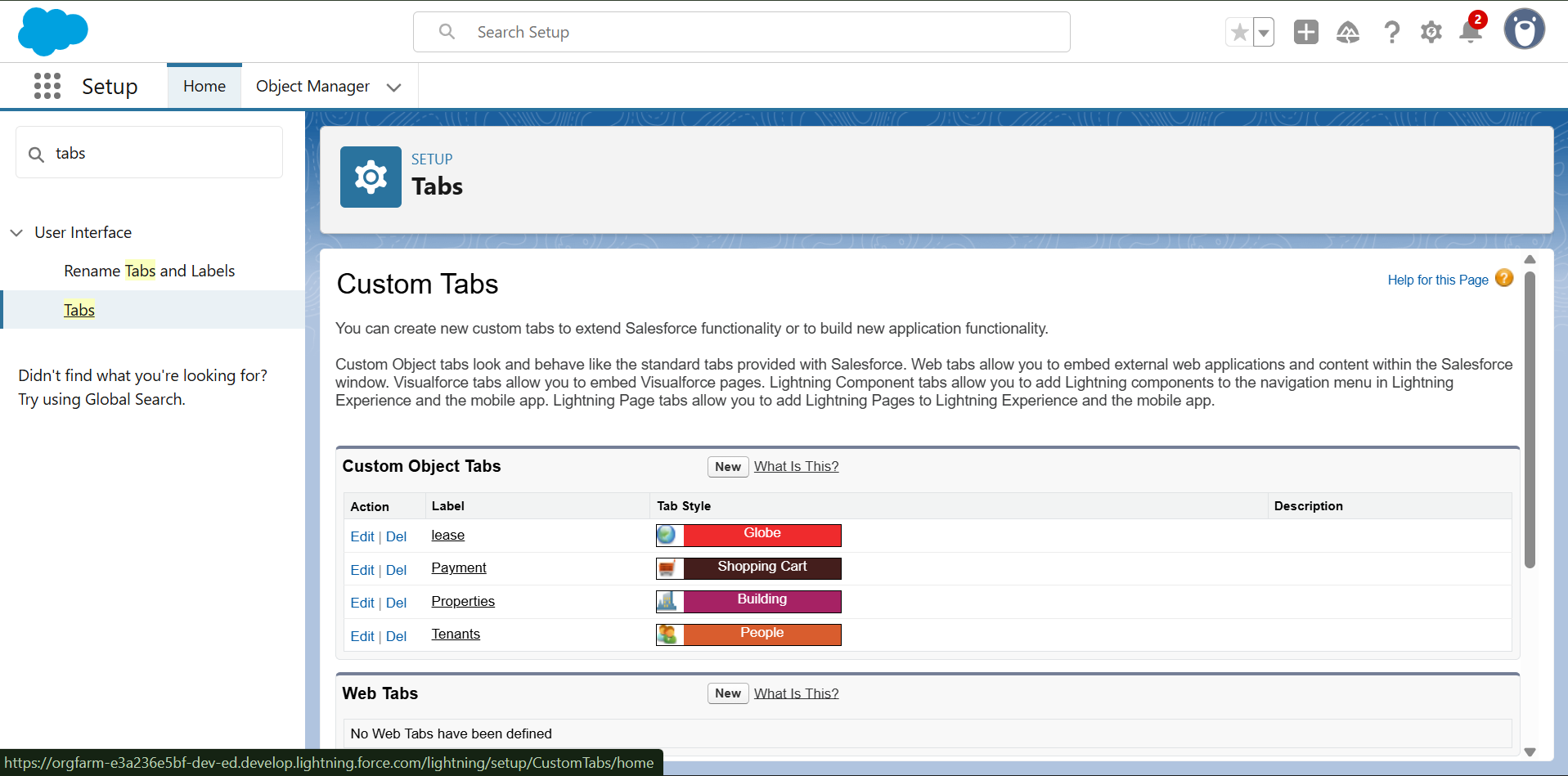




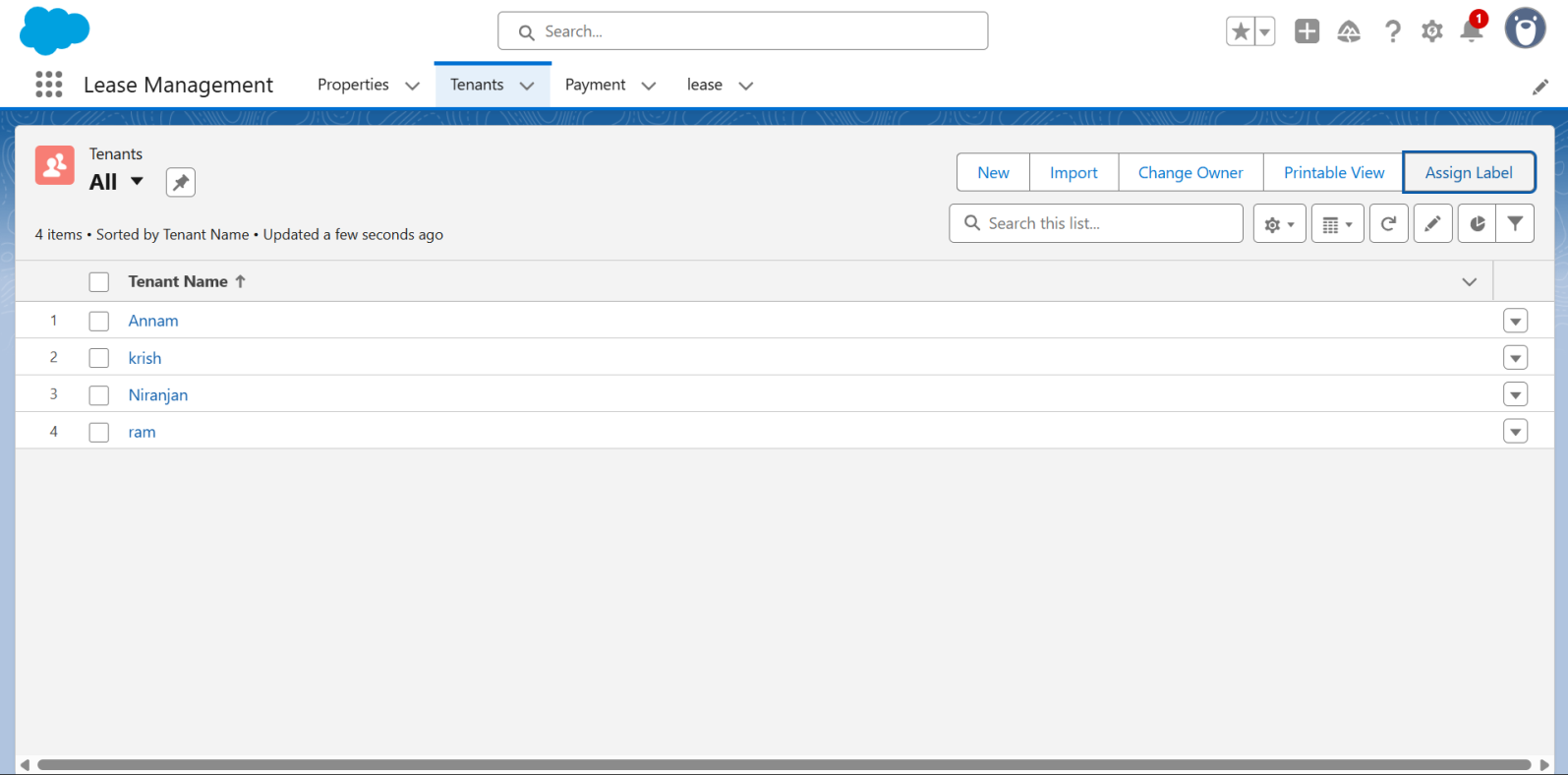
# RESULTS

## OutputScreenshots

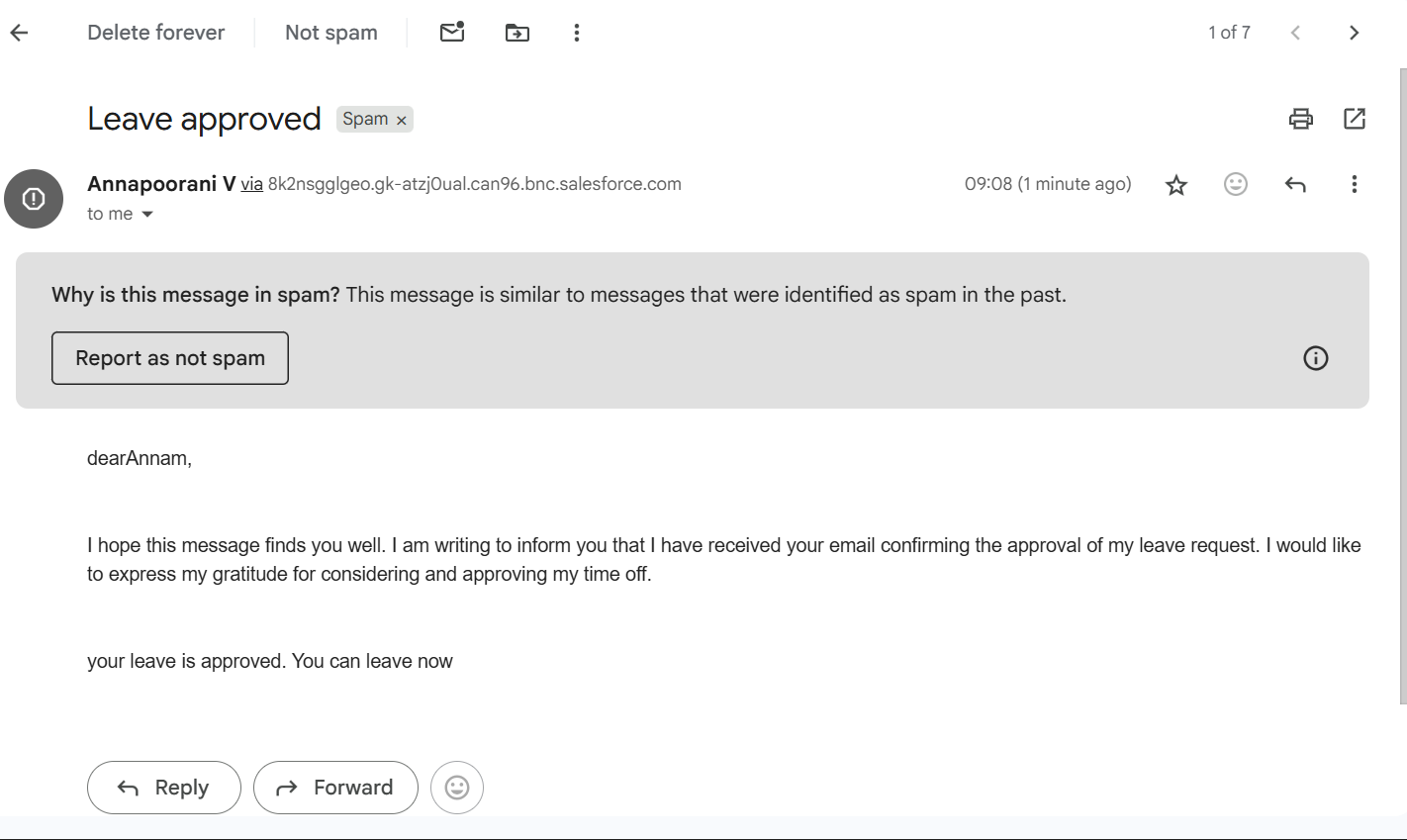
* TabsforProperty,Tenant,Lease,Payment



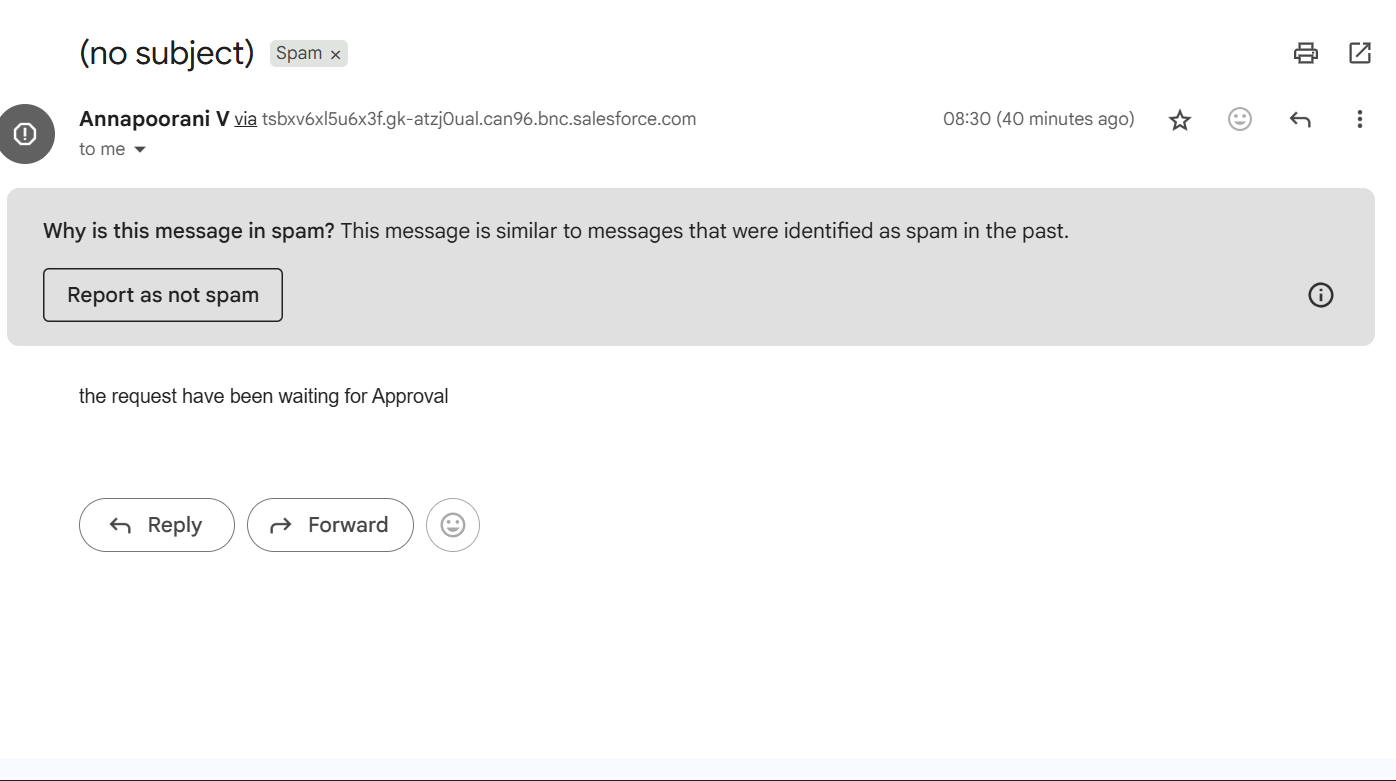
* Emailalerts



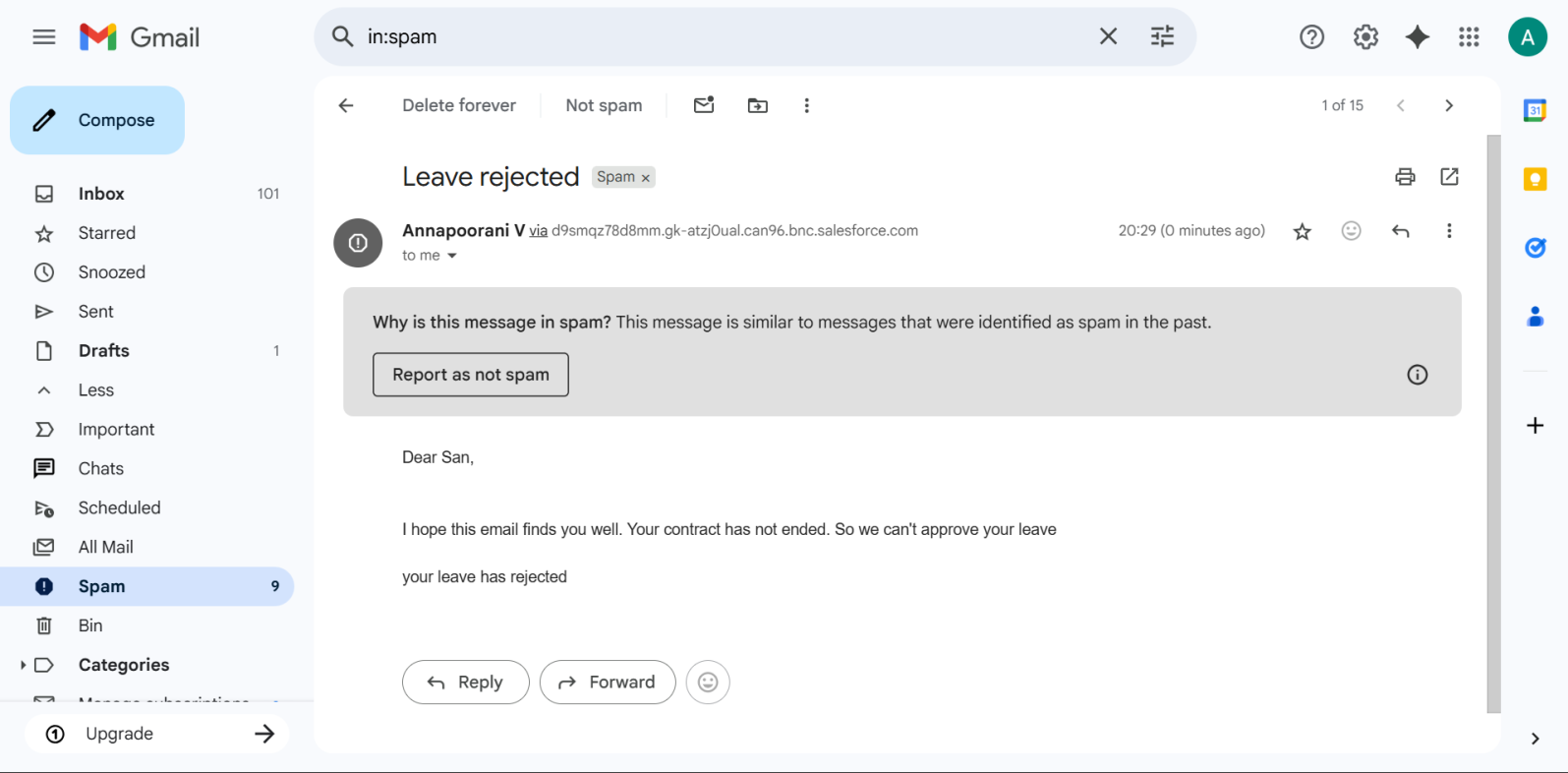
* Requestforapprovethe leave



* Waiting For Approval



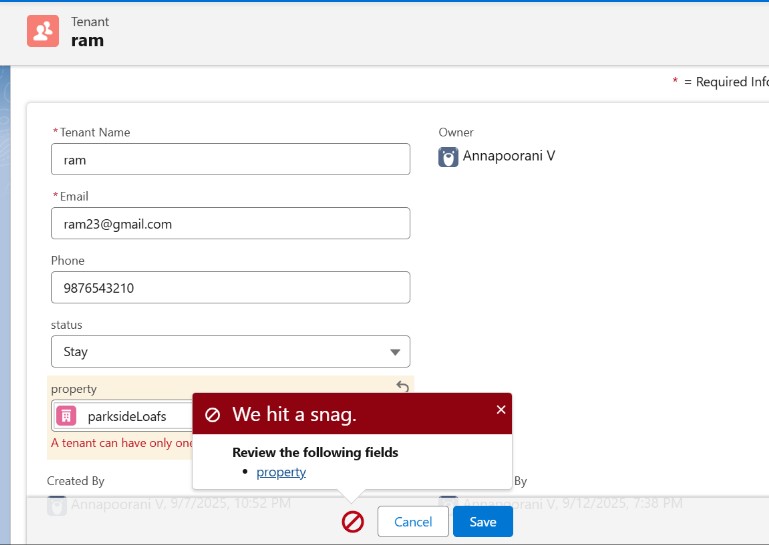
* Leave rejected



* Flowruns



* Triggererrormessages



* Approvalprocess notifications



# ADVANTAGES&DISADVANTAGES

.

# CONCLUSION

The Lease Management System successfully streamlines the operations of leasing through a structured,automatedSalesforceapplication.Itimprovesefficiency,communication,anddata accuracy for both admins and tenants.

## APPENDIX

* + **SourceCode:**ProvidedinApexClassesandTriggers

### Test.apxt:

triggertestonTenantc(beforeinsert){if (trigger.isInsert && trigger.isBefore){ testHandler.preventInsert(trigger.new);

}}

### testHandler.apxc:

public class testHandler { public static void preventInsert(List<

Tenantc>newlist)

{ Set<Id>

existingPropertyIds

=newSet<Id>()

for(TenantcexistingTenant:[SELECTId,PropertycFROMTenantc WHERE Propertyc != null]) {

existingPropertyIds.add(existingTenant.Propertyc;

}for(TenantcnewTenant:

newlist) {

if (newTenant.Propertyc != null && existingPropertyIds.contains(newTenant.Propertyc)){newTenant.addError('A

tenantcanhaveonlyoneproperty');

}

}

}

}

### MothlyEmailScheduler.apxc:

globalclassMonthlyEmailSchedulerimplementsSchedulable{ global void execute(SchedulableContext sc) { Integer currentDay = Date.today().day(); if (currentDay == 1) { sendMonthlyEmails();

}

}public static void sendMonthlyEmails(){List<Tenantc>

tenants=[SELECTId,EmailcFROM Tenantc]; for (Tenantc tenant :

tenants){

StringrecipientEmail=tenant.Emailc;

StringemailContent='Itrustthisemailfindsyouwell.Iamwritingtoremindyou that the monthly rentis dueYour timely payment ensures the smooth functioning of our rental arrangement and helps maintain a positive living environment for all.';

StringemailSubject='Reminder:MonthlyRentPaymentDue';

Messaging.SingleEmailMessage email = new Messaging.SingleEmailMessage(); email.setToAddresses(new String[]{recipientEmail}); email.setSubject(emailSubject); email.setPlainTextBody(emailContent);

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

}

}

}