LEASE MANAGEMENT

COLLEGENAME:KGCollege ofArtsAndScience
TEAM ID:NM2025TMID23624
TEAM SIZE:4

TEAM MEMBERS:

Team Leader Name:Narmathashree T

Email:2326kb29@kgcas.com

Team Member:Nishanth C

Email:2326kb30@kgcas,com

Team Member:Nithish S

Email:2326kb31@kgcas,com

Team Member:Nithyashree S

Email:2326kb32@kgcas,com

1. INTRODUCTION

1.1 Project Overview

TheLease Management System is aSalesforce-based application designed to streamline the processes associated with leasing real estate properties. It handles tenant management, lease contracts, payments, and communication with automation features such as flows, approval processes, and email alerts.

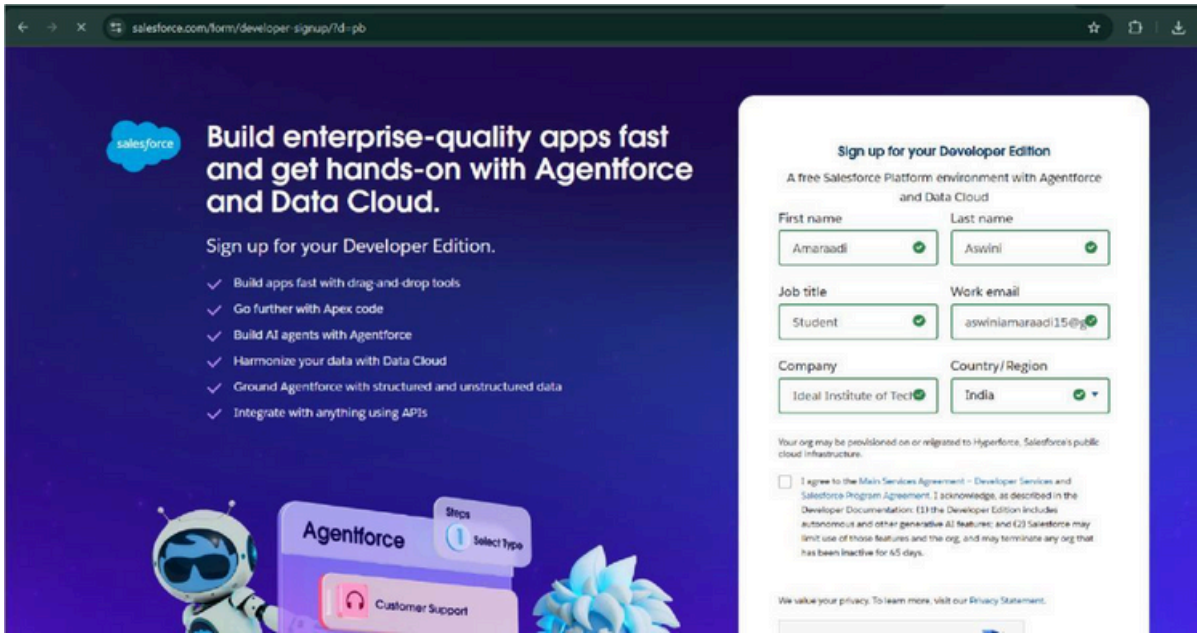
1.2 Purpose

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

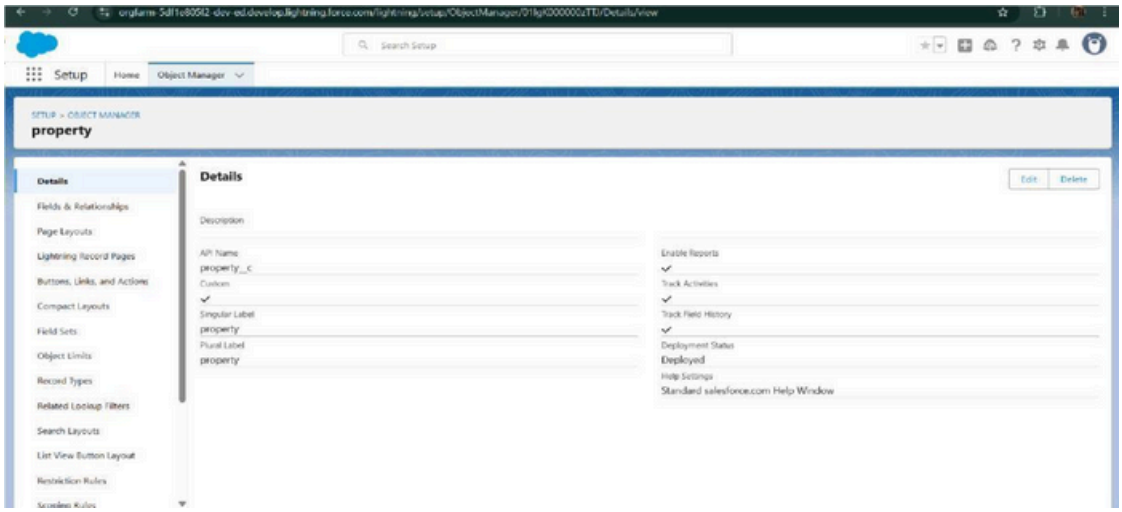
DEVELOPMENT PHASE

Creating Developer Account:

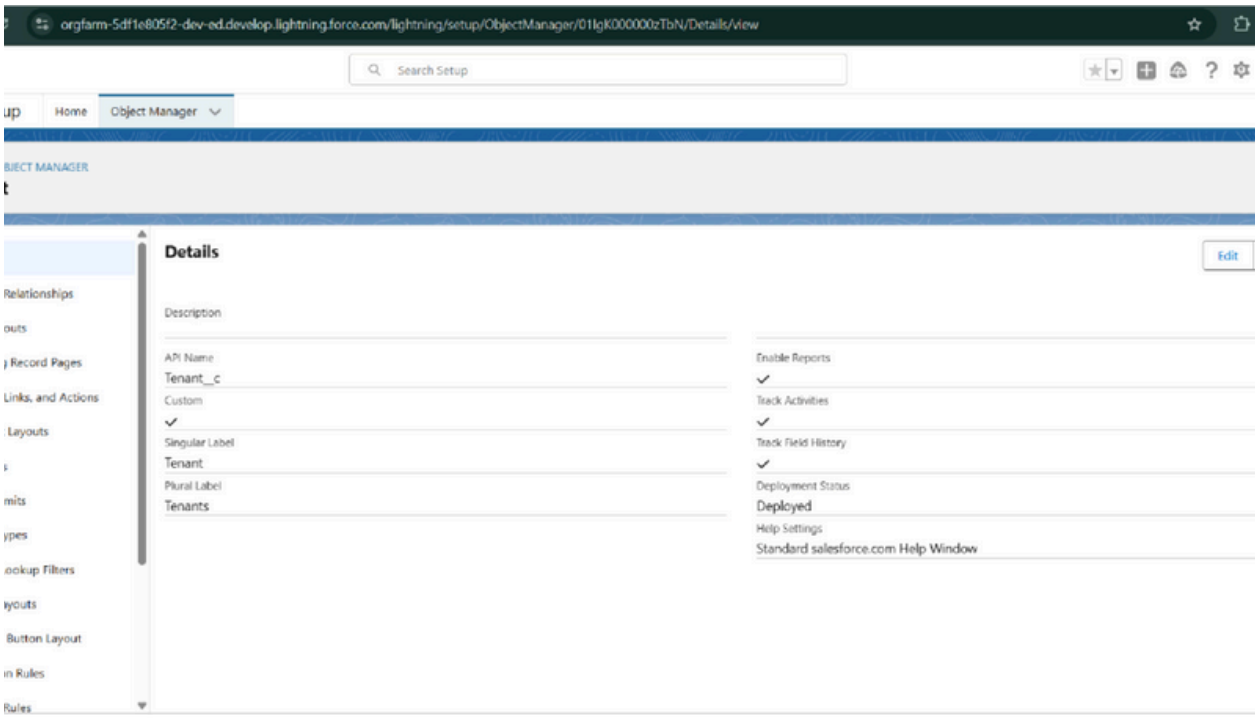
By using this URL - https://www.salesforce.com/form/developer-signup/?d=pb

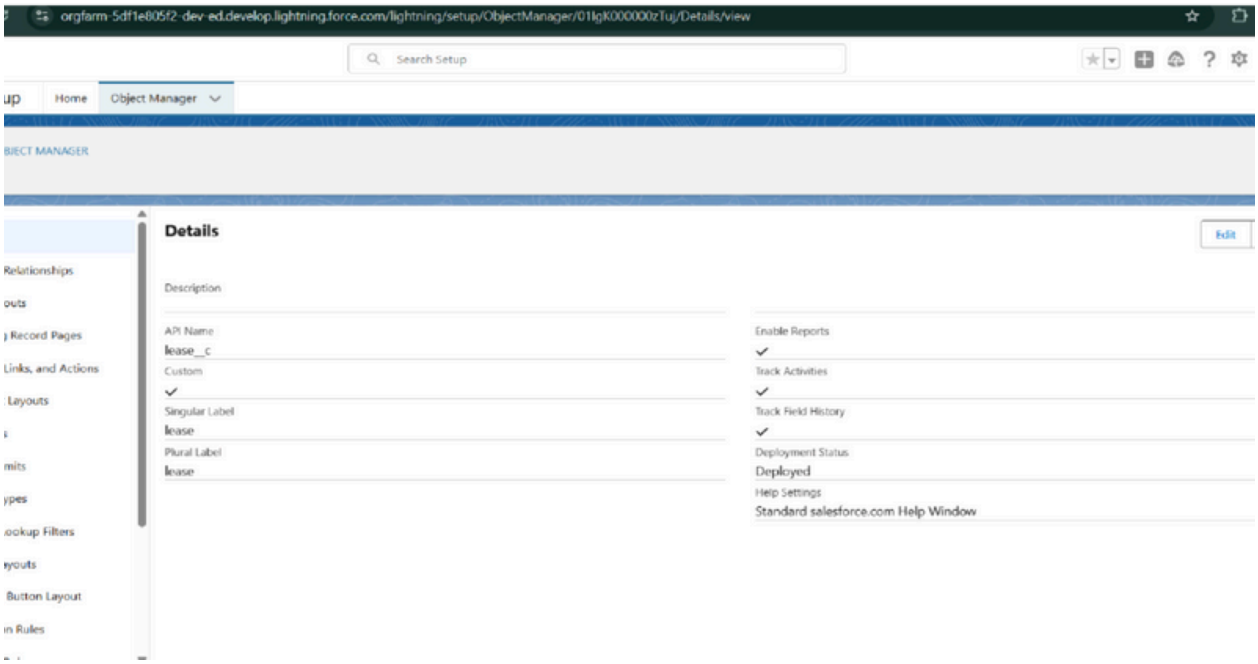


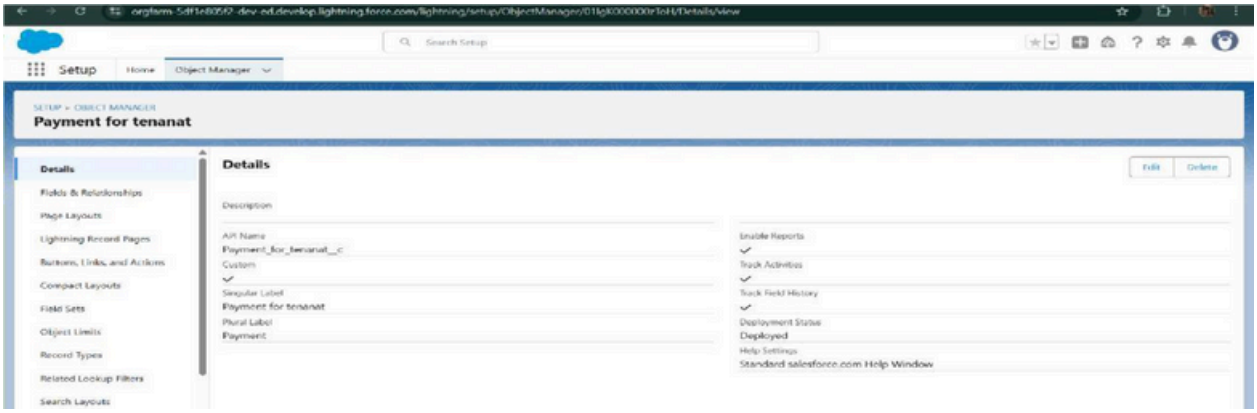
- Created objects: Property, Tenant, Lease, Payment



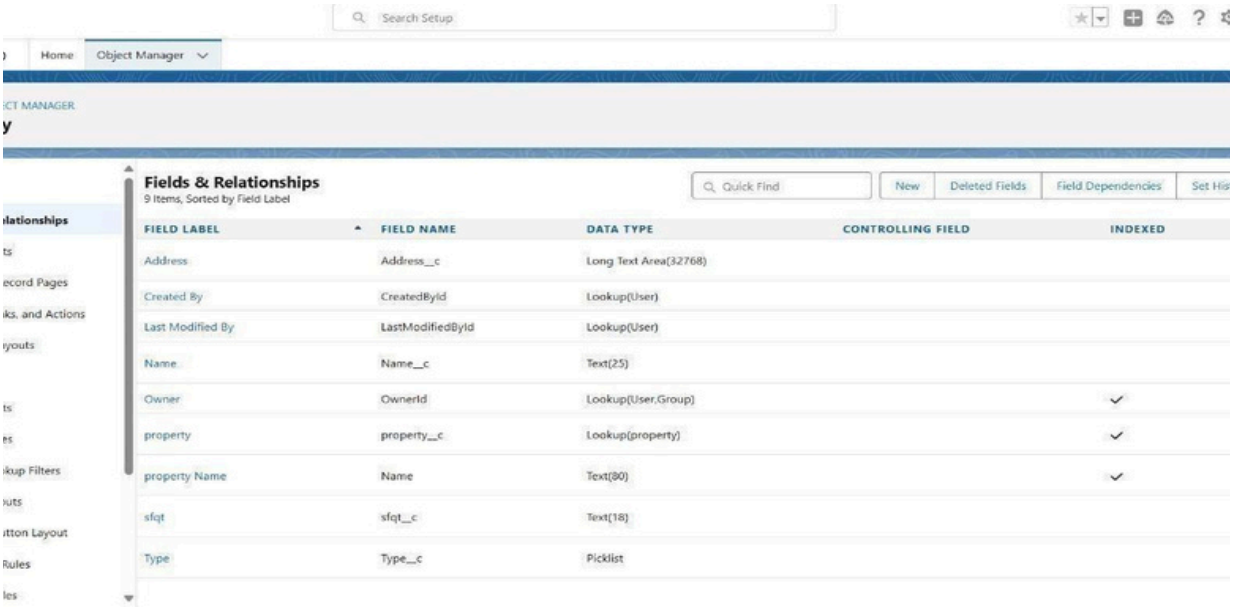
- Here are 8 key points from the screenshot:
- Object Name: property
- API Name: property__c
- Custom Object: Yes (✓)
- Singular Label: property
- Plural Label: property
- Enable Reports: Checked (✓)
- Track Activities and Field History: Enabled (✓)
- Deployment Status: Deployed

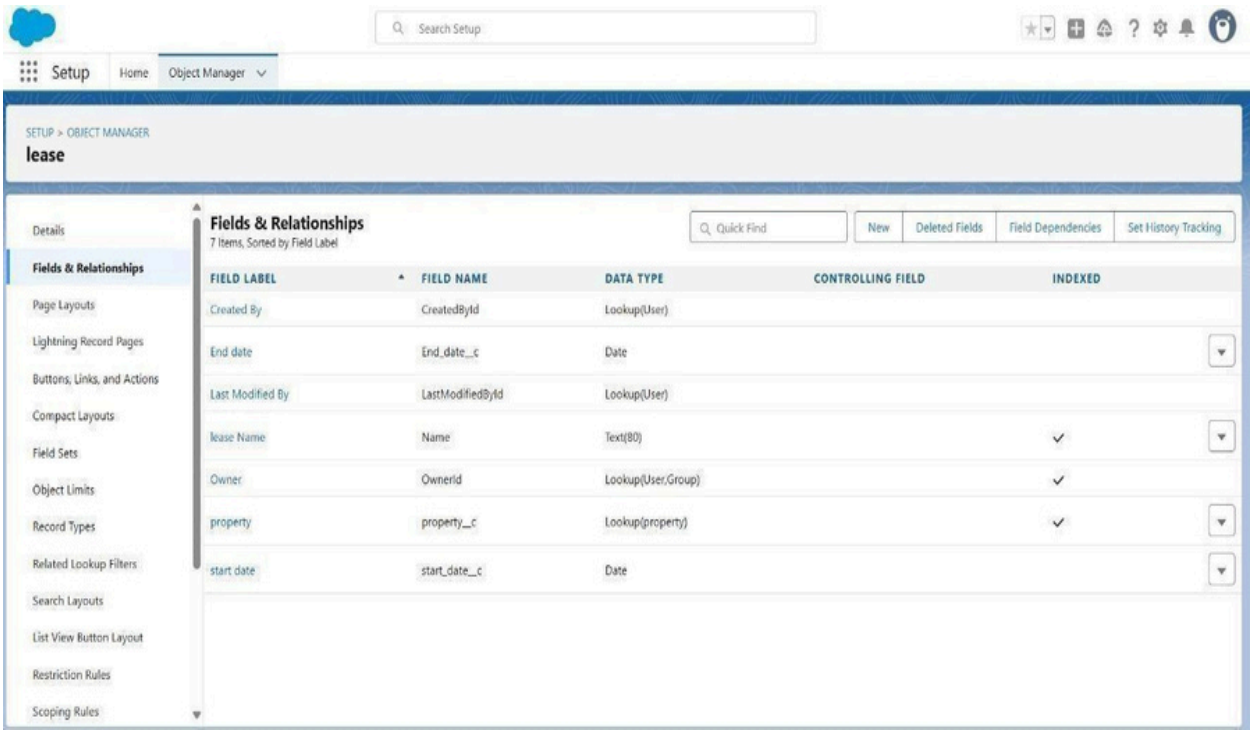

orgfarm-5df1e805f2-dev-ed.develop.lightning.force.com/lightning/setup/ObjectManager/01IgK000000zTbN/Details/view
Search Setup
Home
Object Manager
Details
Edit
Description
API Name
Tenant__c
Custom
Singular Label
Tenant
Plural Label
Tenants
Enable Reports
Track Activities
Track Field History
Deployment Status
Deployed
Help Settings
Standard salesforce.com Help Window

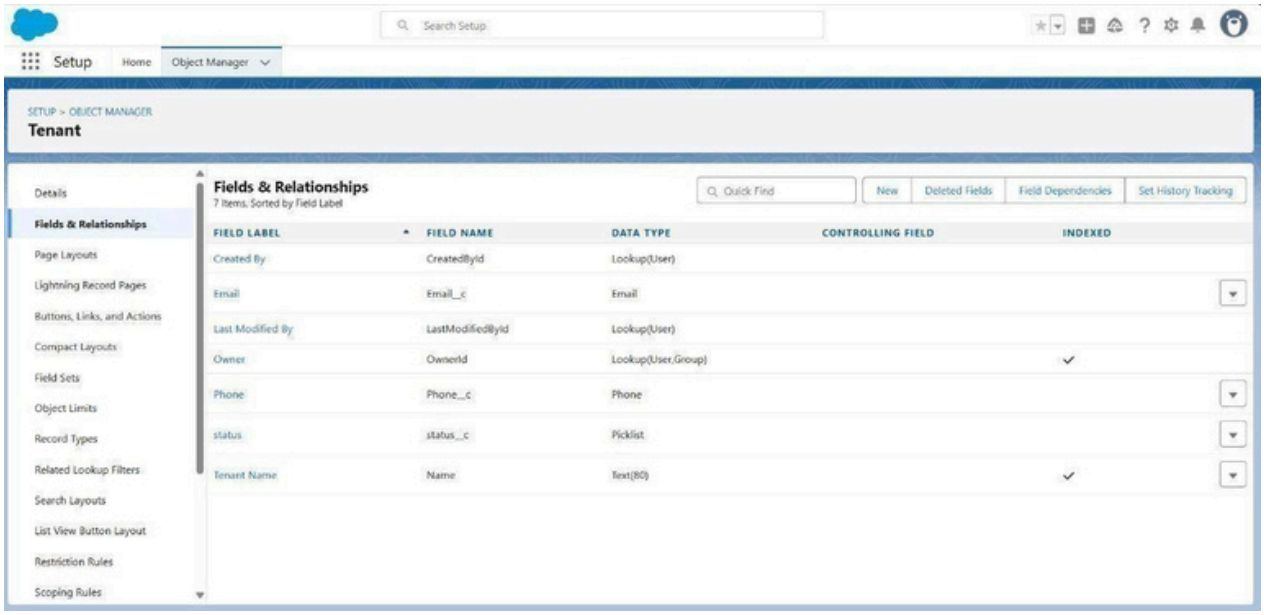

orgfarm-5df1e805f2-dev-ed.develop.lightning.force.com/lightning/setup/ObjectManager/01IgK000000zTuj/Details/view
Search Setup
Home
Object Manager
Details
Edit
Description
API Name
lease__c
Custom
Singular Label
lease
Plural Label
lease
Enable Reports
Track Activities
Track Field History
Deployment Status
Deployed
Help Settings
Standard salesforce.com Help Window



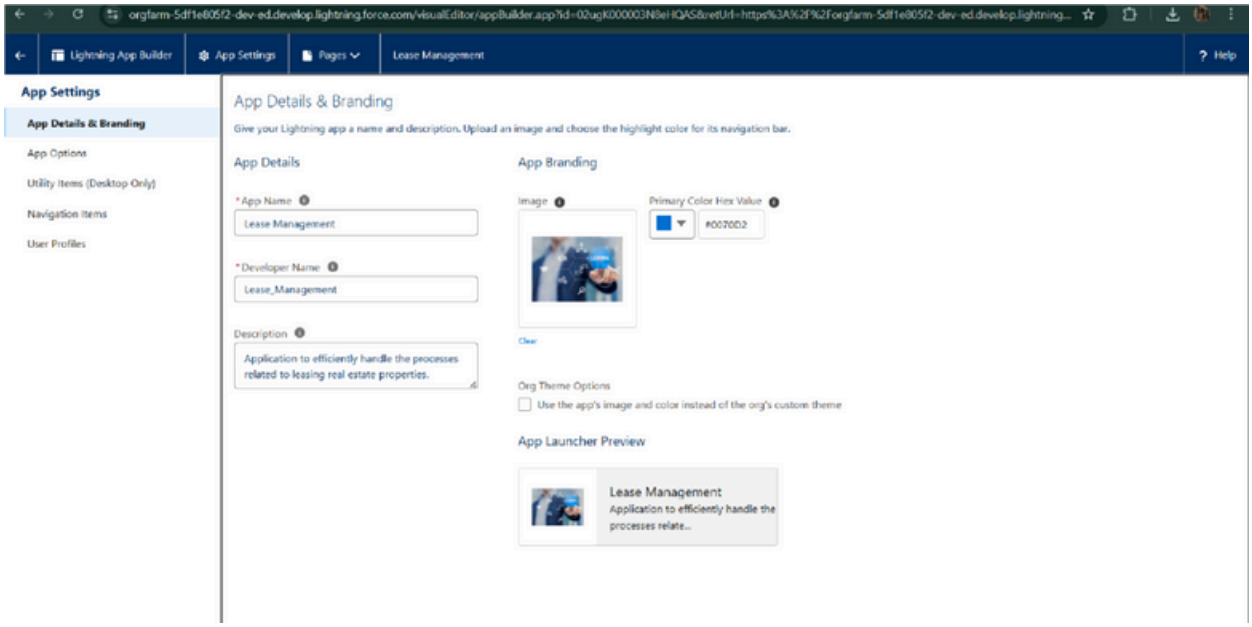
- Configured fields and relationships



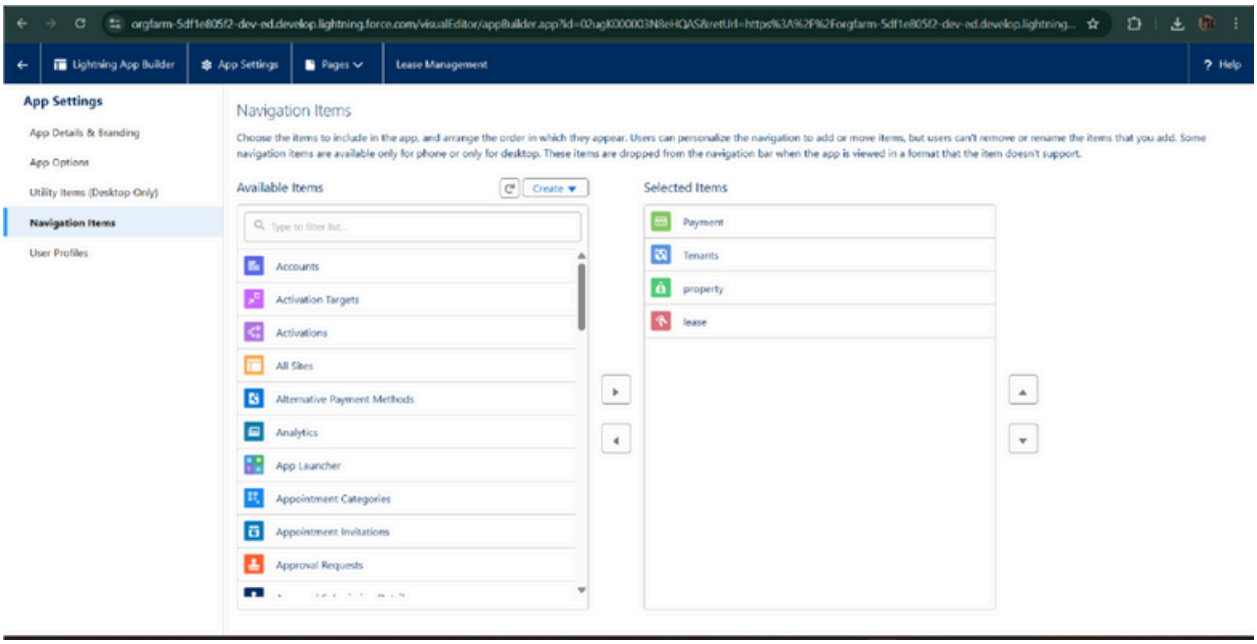


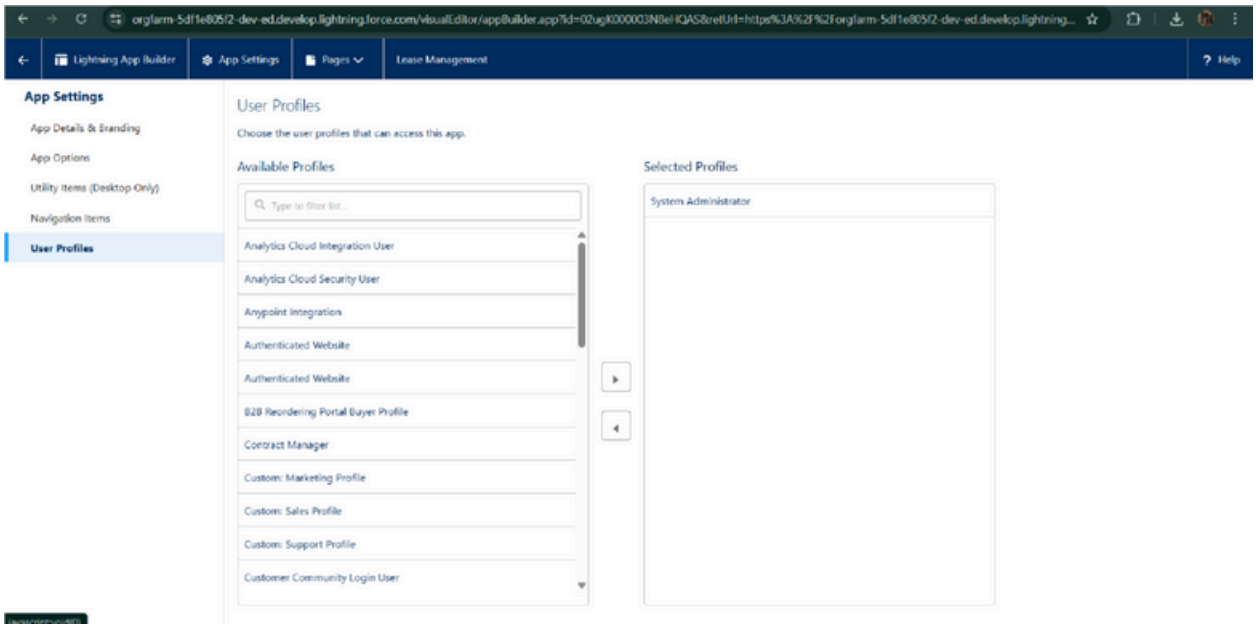


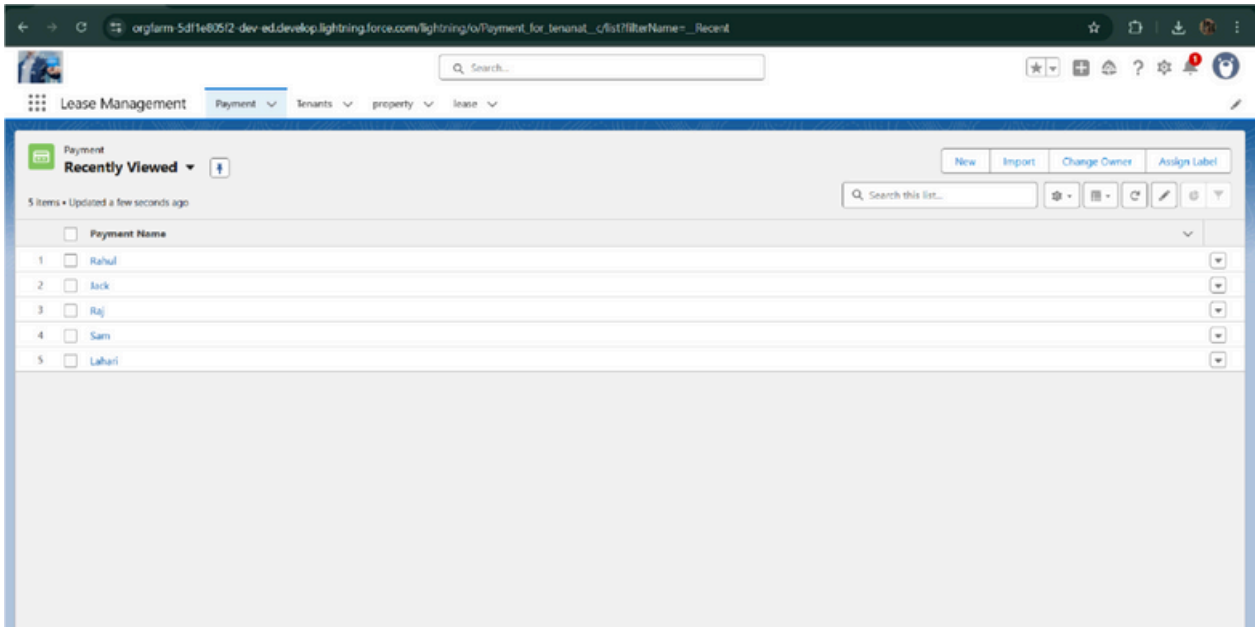
- Developed Lightning App with relevant tabs

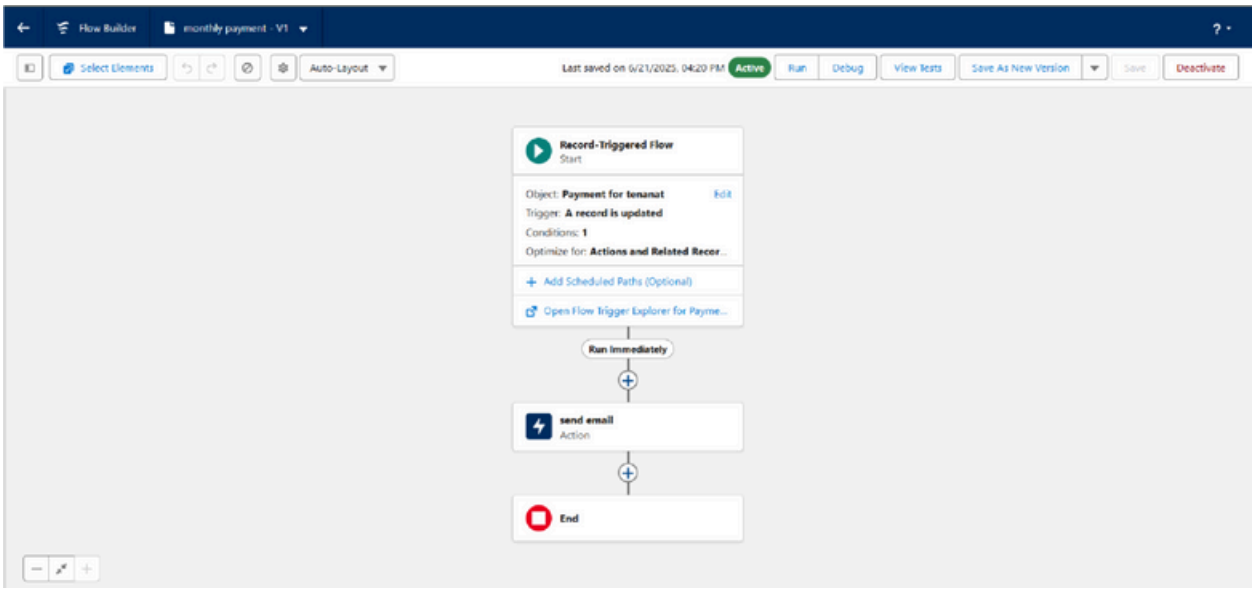


- App Name is “Lease Management”.
- Developer Name is “Lease_Management”.
- Description explains handling leasing real estate processes.
- Primary Color Hex Value is “#0070D2”.
- App Launcher Preview displays app name, image, and description.

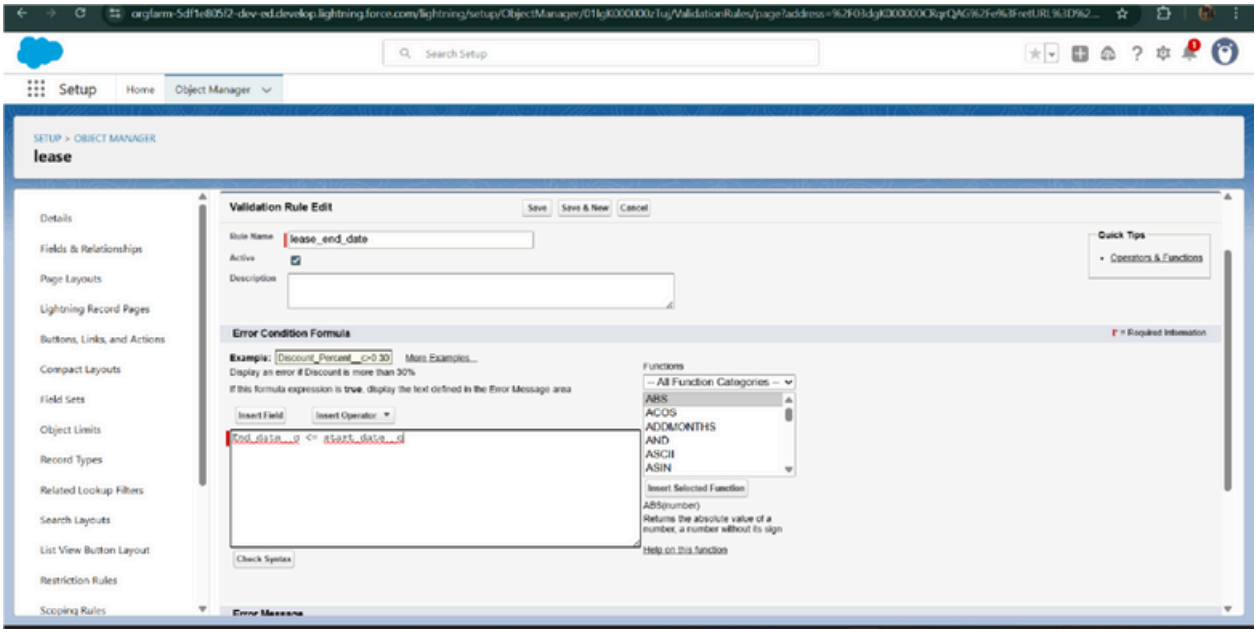



Lightning App Builder
App Settings
Pages
Lease Management
Help
App Settings
App Details & Branding
App Options
Utility Items (Desktop Only)
Navigation Items
User Profiles
User Profiles
Choose the user profiles that can access this app.
Available Profiles
Selected Profiles
System Administrator
Analytics Cloud Integration User
Analytics Cloud Security User
Anypoint Integration
Authenticated Website
Authenticated Website
B2B Reordering Portal Buyer Profile
Contract Manager
Custom: Marketing Profile
Custom: Sales Profile
Custom: Support Profile
Customer Community Login User

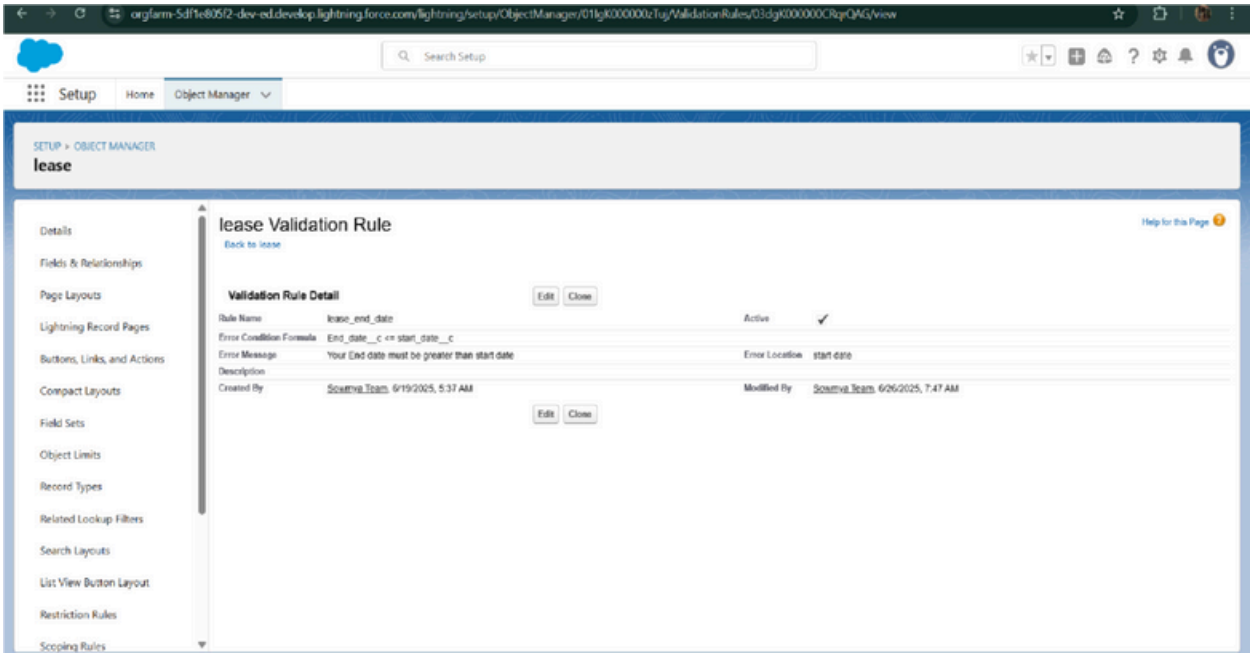

Lease Management
Payment
Tenants
property
lease
Payment
Recently Viewed
5 items • Updated a few seconds ago
New
Import
Change Owner
Assign Label
Payment Name
1 Rahul
2 Jack
3 Raj
4 Sam
5 Lahari



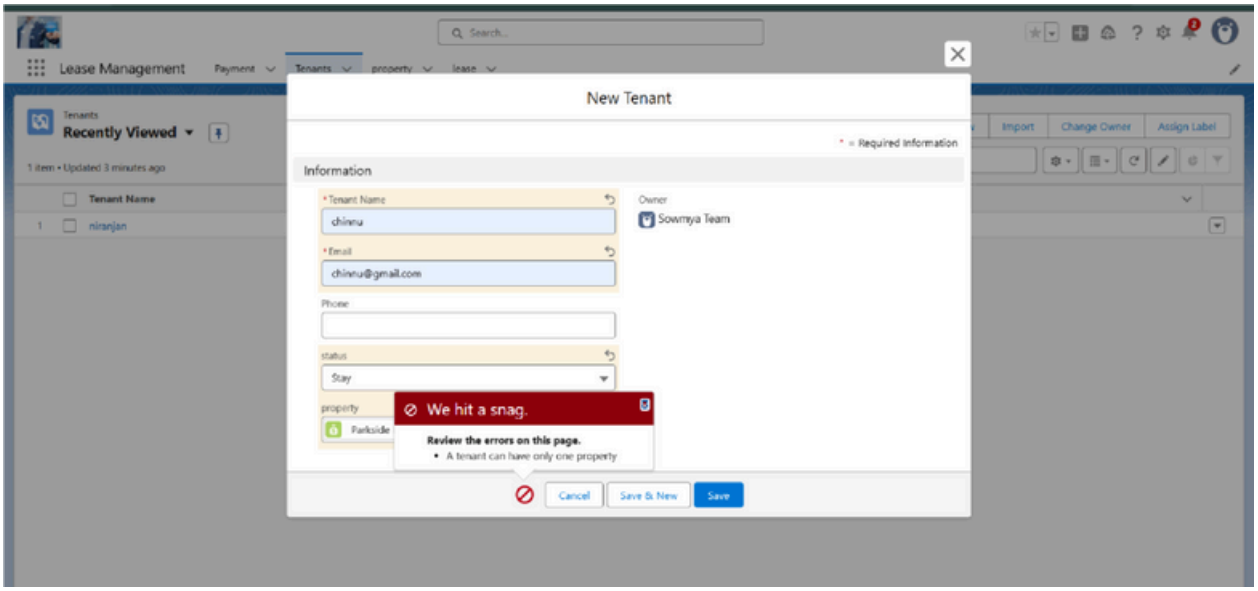
- Implemented Flows for monthly rent and payment success



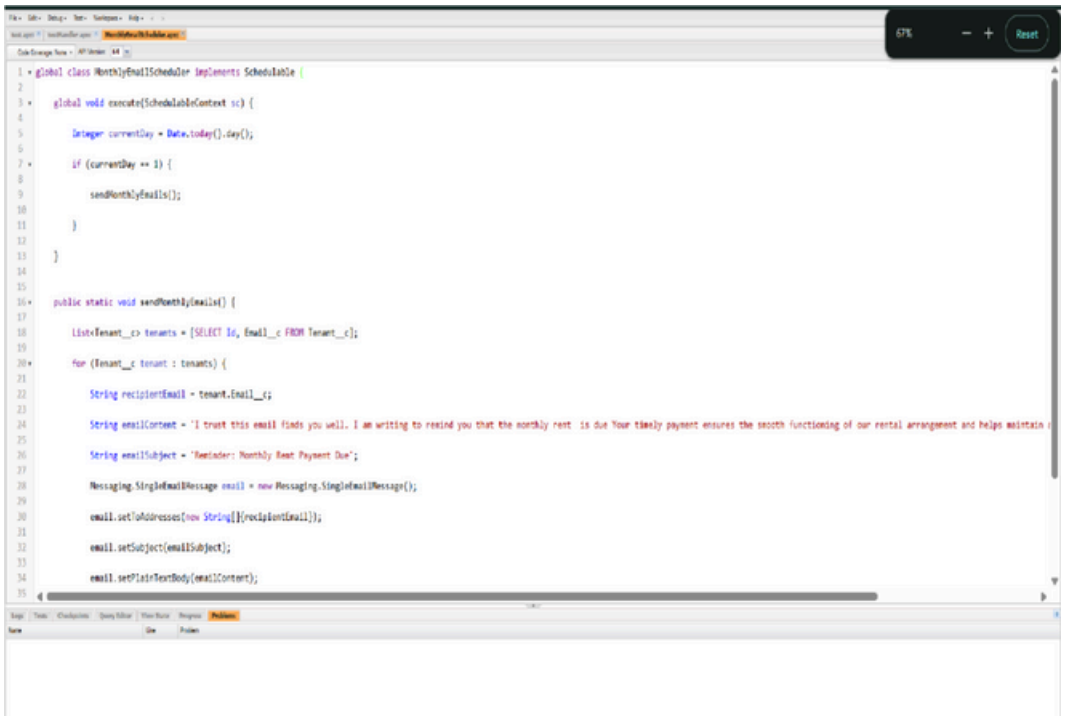
- To create a validation rule to a Lease Object



- Added Apex trigger to restrict multiple tenants per property

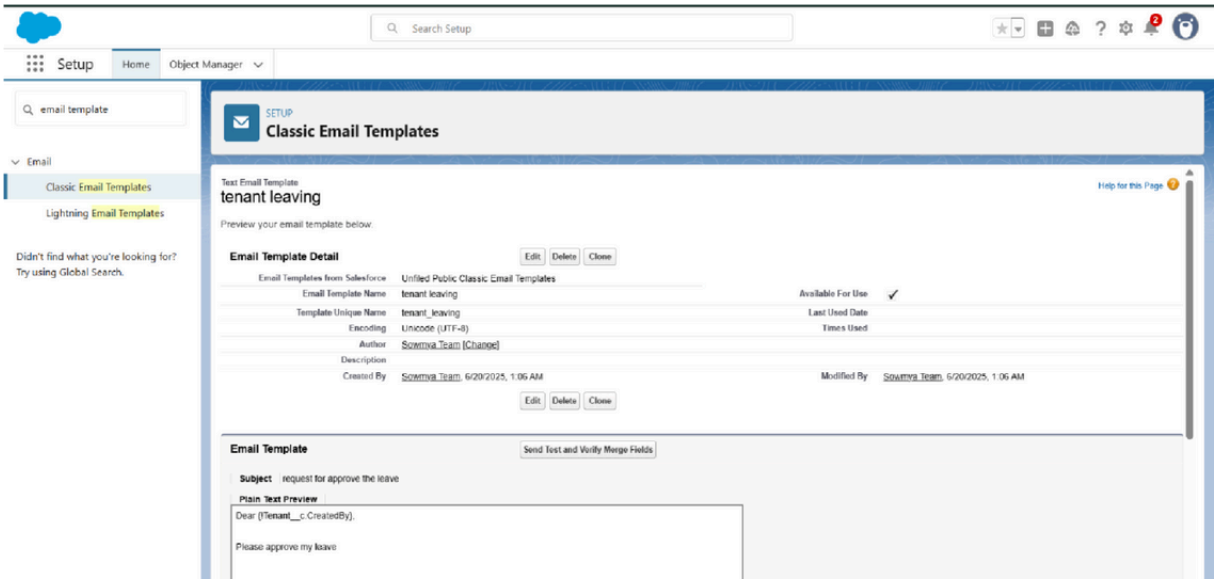


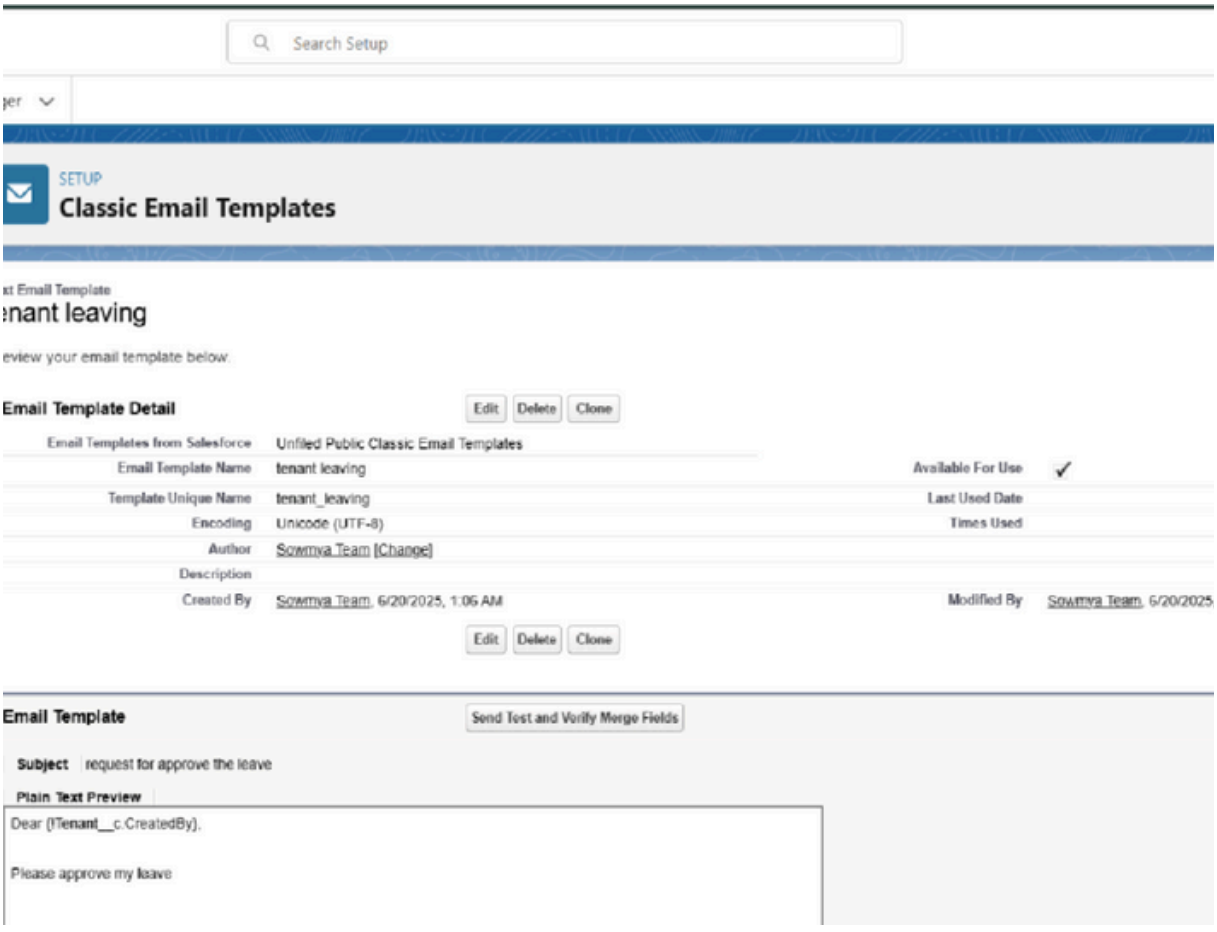
- Scheduled monthly reminder emails using Apex class

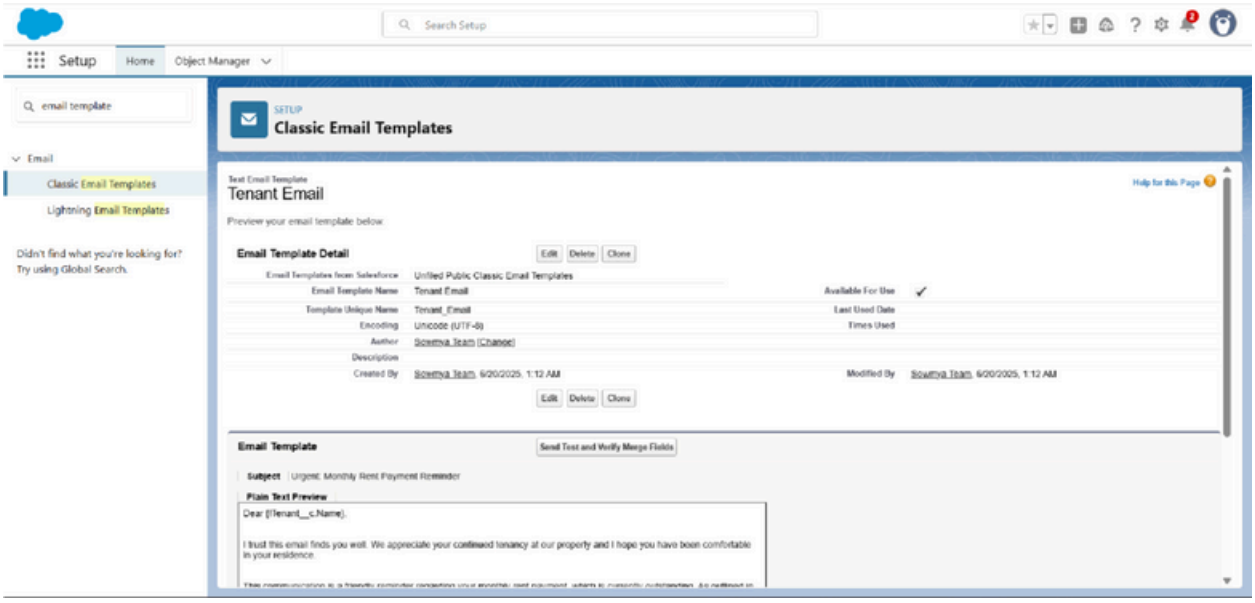


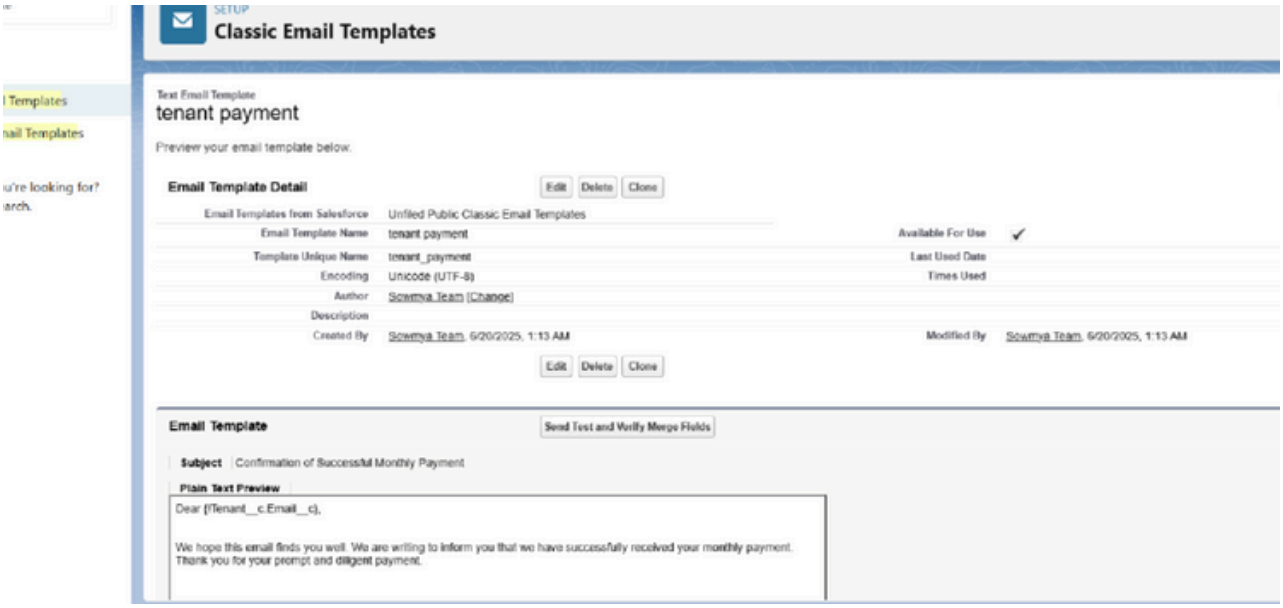
```
global class MonthlyEmailScheduler implements Schedulable {

    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

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

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

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

            email.setSubject(emailSubject);

            email.setPlainTextBody(emailContent);
```

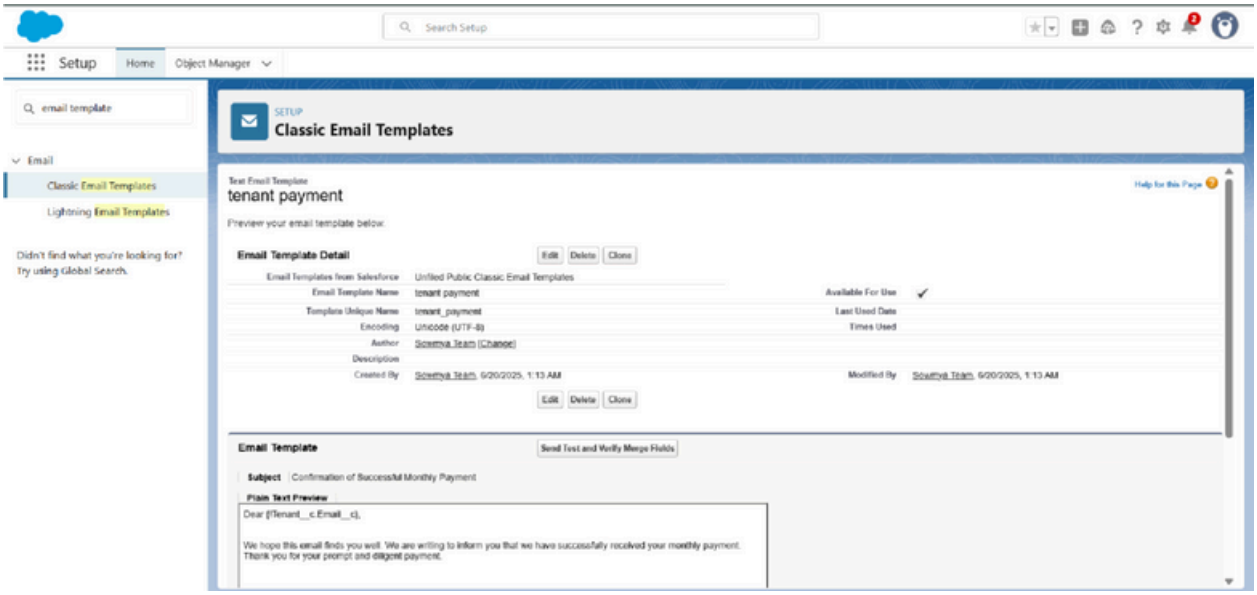
- Built and tested email templates for leave request, approval, rejection, payment, and reminders
- Here are 5 key points (in 2 lines each) from the code in the screenshot:
- Apex class MonthlyEmailScheduler implements Schedulable.
- It runs scheduled jobs to send monthly reminder emails.
- execute method checks if the day is the 1st.
- If true, it calls the sendMonthlyEmails() method.
- sendMonthlyEmails() queries Tenant records.
- It fetches Id and Email__c fields from Tenant__c object.
- For each tenant, an email is prepared.
- Email includes subject "Reminder: Monthly Rent Payment Due".
- Uses Messaging.SingleEmailMessage to send mail.
- Sets recipient, subject, and email content dynamically.


Search Setup
Setup
Home
Object Manager
email template
Email
Classic Email Templates
Lightning Email Templates
Didn't find what you're looking for?
Try using Global Search.
SETUP
Classic Email Templates
Text Email Template
tenant leaving
Help for this Page
Preview your email template below.
Email Template Detail
Edit
Delete
Clone
Email Templates from Salesforce
Unfiled Public Classic Email Templates
Email Template Name
tenant leaving
Available For Use
Template Unique Name
tenant_leaving
Last Used Date
Encoding
Unicode (UTF-8)
Times Used
Author
Sowmya Team [Change]
Description
Created By
Sowmya Team, 6/20/2025, 1:06 AM
Modified By
Sowmya Team, 6/20/2025, 1:06 AM
Email Template
Send Test and Verify Merge Fields
Subject
request for approve the leave
Plain Text Preview
Dear {!Tenant__c.CreatedBy},
Please approve my leave


Search Setup
SETUP
Classic Email Templates
enant leaving
eview your email template below.
Email Template Detail
Edit
Delete
Clone
Email Templates from Salesforce
Unfiled Public Classic Email Templates
Email Template Name
tenant leaving
Available For Use
Template Unique Name
tenant_leaving
Last Used Date
Encoding
Unicode (UTF-8)
Times Used
Author
Sowmya Team [Change]
Description
Created By
Sowmya Team, 6/20/2025, 1:06 AM
Modified By
Sowmya Team, 6/20/2025
Email Template
Send Test and Verify Merge Fields
Subject
request for approve the leave
Plain Text Preview
Dear {!Tenant__c.CreatedBy},
Please approve my leave

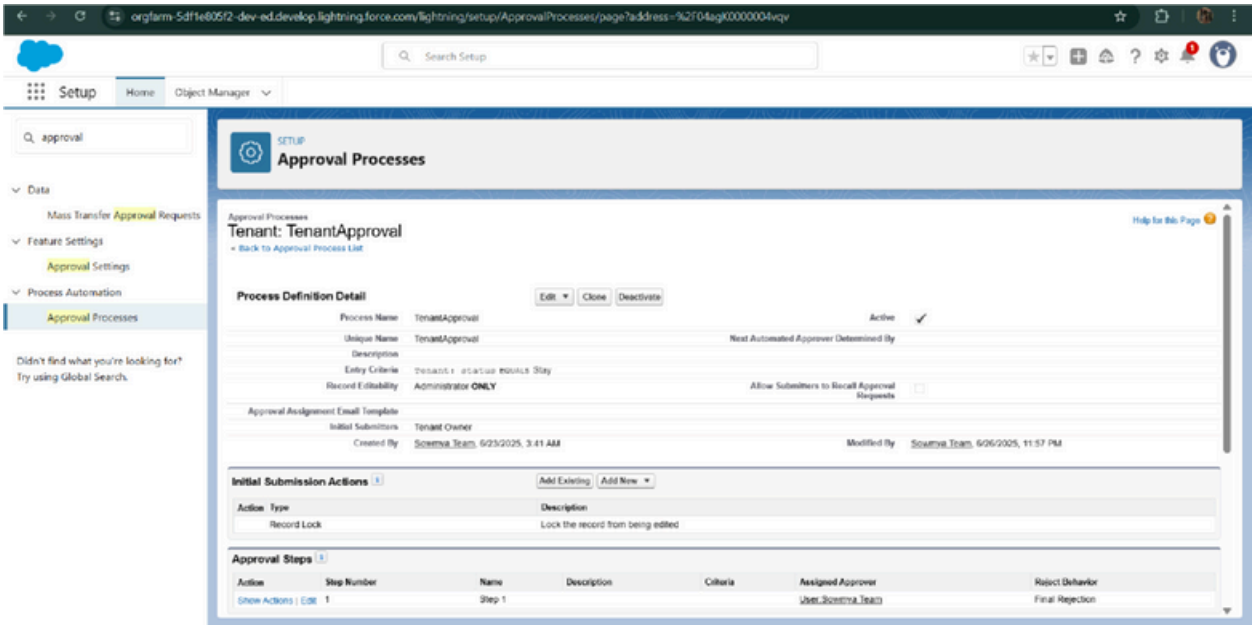

Search Setup
Setup
Home
Object Manager
email template
Email
Classic Email Templates
Lightning Email Templates
Didn't find what you're looking for?
Try using Global Search.
SETUP
Classic Email Templates
Text Email Template
Tenant Email
Help for this Page
Preview your email template below.
Email Template Detail
Edit
Delete
Clone
Email Templates from Salesforce
Unfiled Public Classic Email Templates
Email Template Name
Tenant Email
Available For Use
Template Unique Name
Tenant_Email
Last Used Date
Encoding
Unicode (UTF-8)
Times Used
Author
Sowmya Team [Change]
Description
Created By
Sowmya Team, 6/20/2025, 1:12 AM
Modified By
Sowmya Team, 6/20/2025, 1:12 AM
Email Template
Send Test and Verify Merge Fields
Subject
Urgent: Monthly Rent Payment Reminder
Plain Text Preview
Dear {!Tenant__c.Name},
I trust this email finds you well. We appreciate your continued tenancy at our property and I hope you have been comfortable in your residence.


SETUP
Classic Email Templates
Text Email Template
tenant payment
Preview your email template below.
Email Template Detail
Edit
Delete
Clone
Email Templates from Salesforce
Unfiled Public Classic Email Templates
Email Template Name
tenant payment
Available For Use
Template Unique Name
tenant_payment
Last Used Date
Encoding
Unicode (UTF-8)
Times Used
Author
Sowmya Team [Change]
Description
Created By
Sowmya Team, 6/20/2025, 1:13 AM
Modified By
Sowmya Team, 6/20/2025, 1:13 AM
Email Template
Send Test and Verify Merge Fields
Subject
Confirmation of Successful Monthly Payment
Plain Text Preview
Dear {!Tenant__c.Email__c},
We hope this email finds you well. We are writing to inform you that we have successfully received your monthly payment.
Thank you for your prompt and diligent payment.

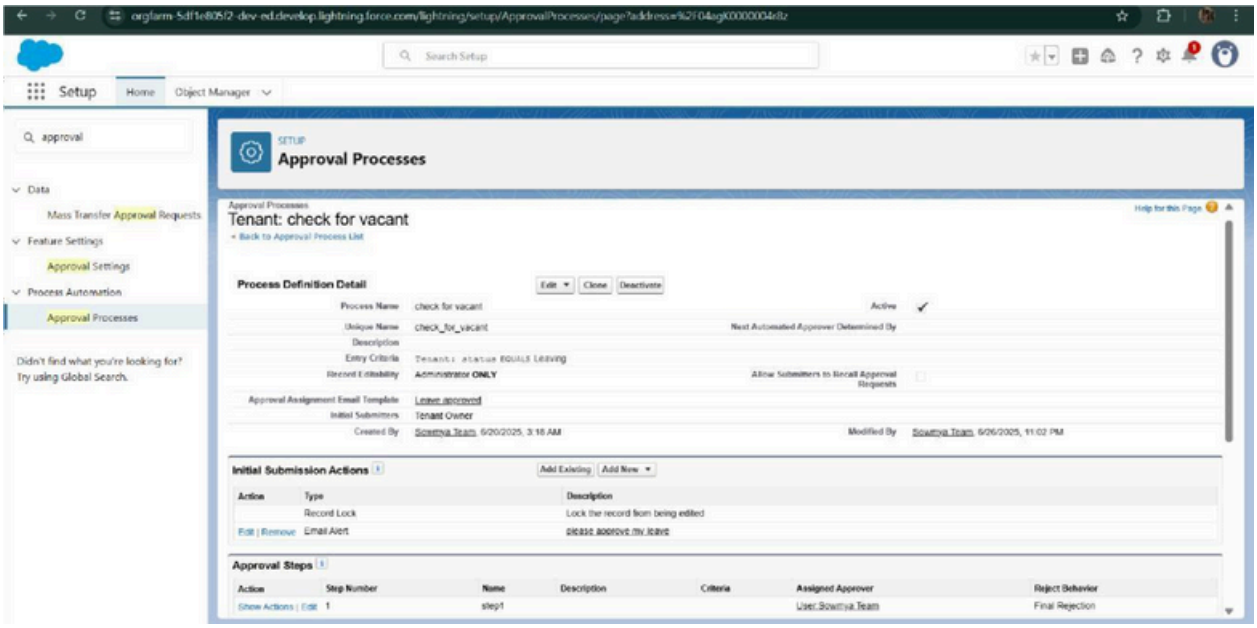


● Approval Process creation

For Tenant Leaving:

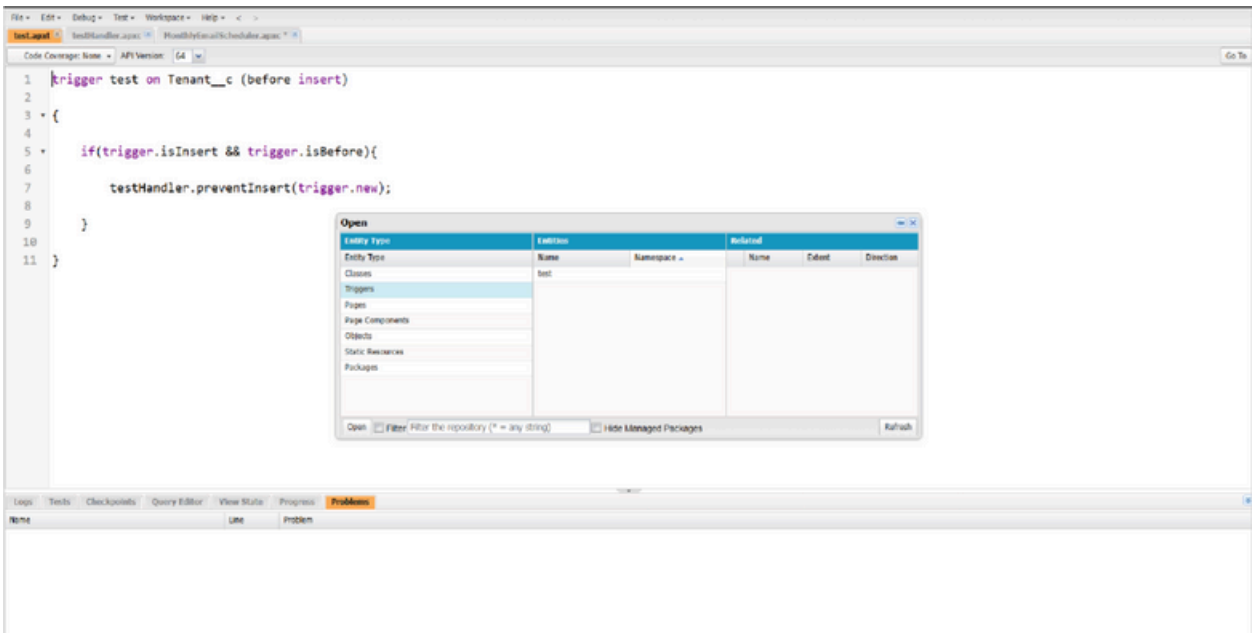


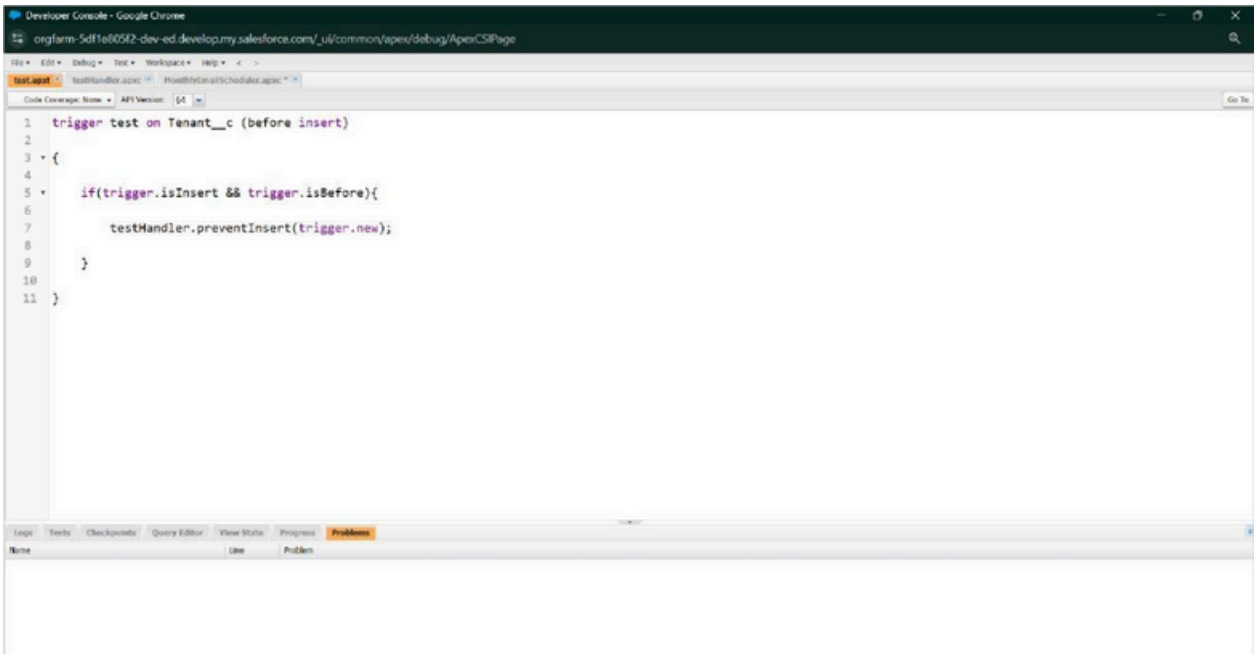
For Check for Vacant:



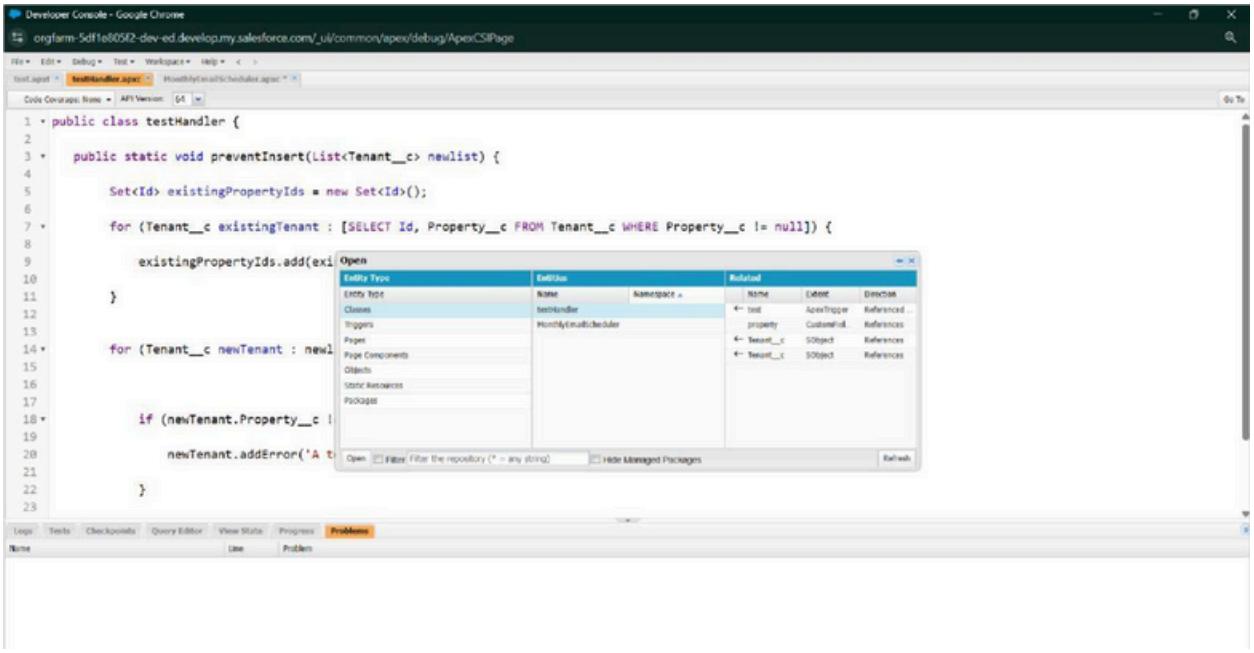
● Apex Trigger

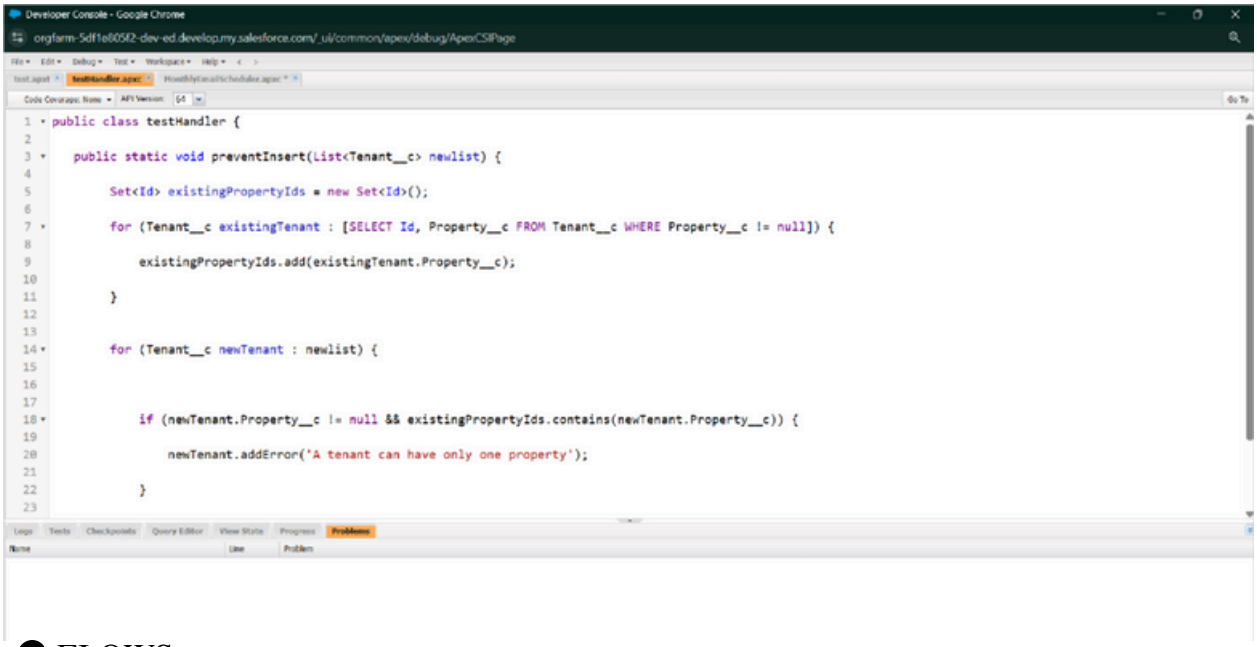
Create an Apex Trigger



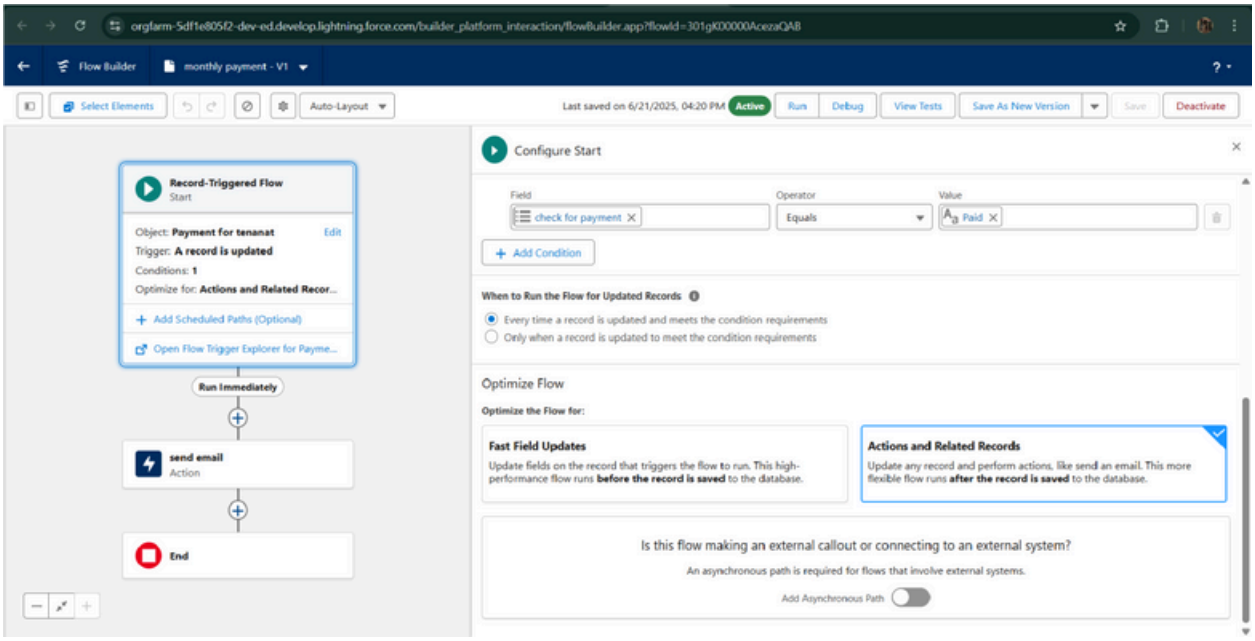


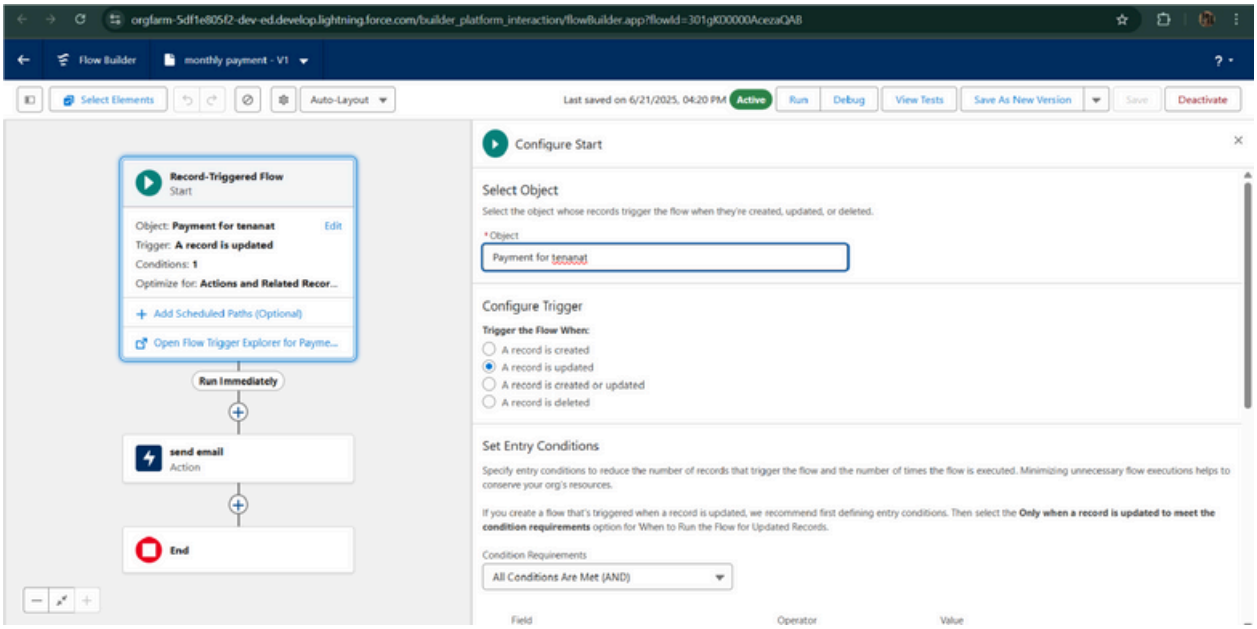
Create an Apex Handler class



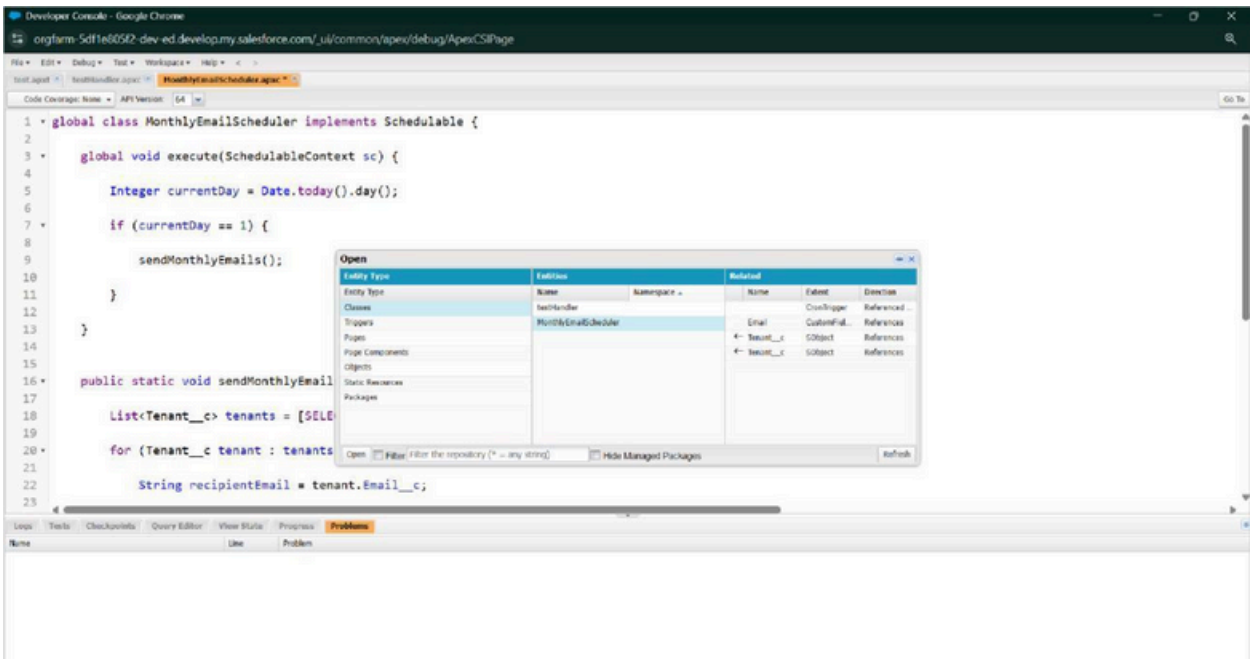

Developer Console - Google Chrome
orgfarm-5df1e805f2-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage
public class testHandler {
public static void preventInsert(List<Tenant__c> newlist) {
Set<Id> existingPropertyIds = new Set<Id>();
for (Tenant__c existingTenant : [SELECT Id, Property__c 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');
}

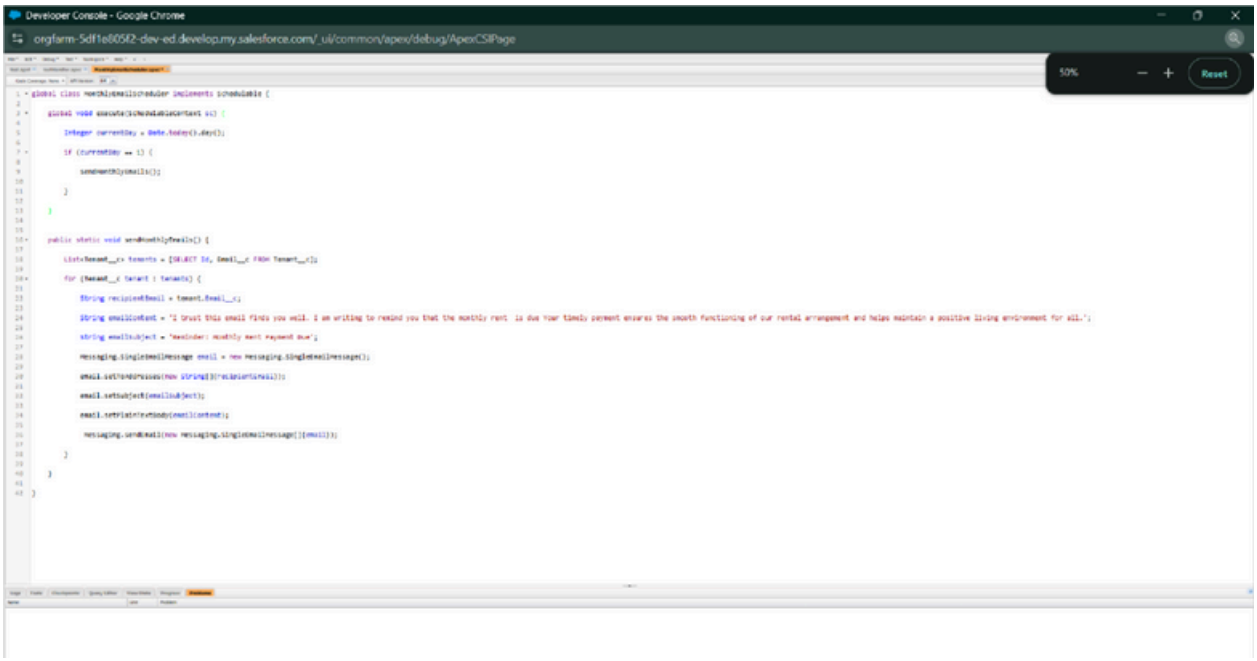
● FLOWS


orgfarm-5df1e805f2-dev-ed.develop.lightning.force.com/builder_platform_interaction/flowBuilder.app?flowId=301gK00000AcezaQAB
Flow Builder
monthly payment - V1
Select Elements
Auto-Layout
Last saved on 6/21/2025, 04:20 PM
Active
Run
Debug
View Tests
Save As New Version
Save
Deactivate
Record-Triggered Flow
Start
Object: Payment for tenanat
Edit
Trigger: A record is updated
Conditions: 1
Optimize for: Actions and Related Recor...
Add Scheduled Paths (Optional)
Open Flow Trigger Explorer for Payme...
Run Immediately
send email
Action
End
Configure Start
Field
Operator
Value
check for payment
Equals
Paid
Add Condition
When to Run the Flow for Updated Records
Every time a record is updated and meets the condition requirements
Only when a record is updated to meet the condition requirements
Optimize Flow
Optimize the Flow for:
Fast Field Updates
Update fields on the record that triggers the flow to run. This high-performance flow runs before the record is saved to the database.
Actions and Related Records
Update any record and perform actions, like send an email. This more flexible flow runs after the record is saved to the database.
Is this flow making an external callout or connecting to an external system?
An asynchronous path is required for flows that involve external systems.
Add Asynchronous Path

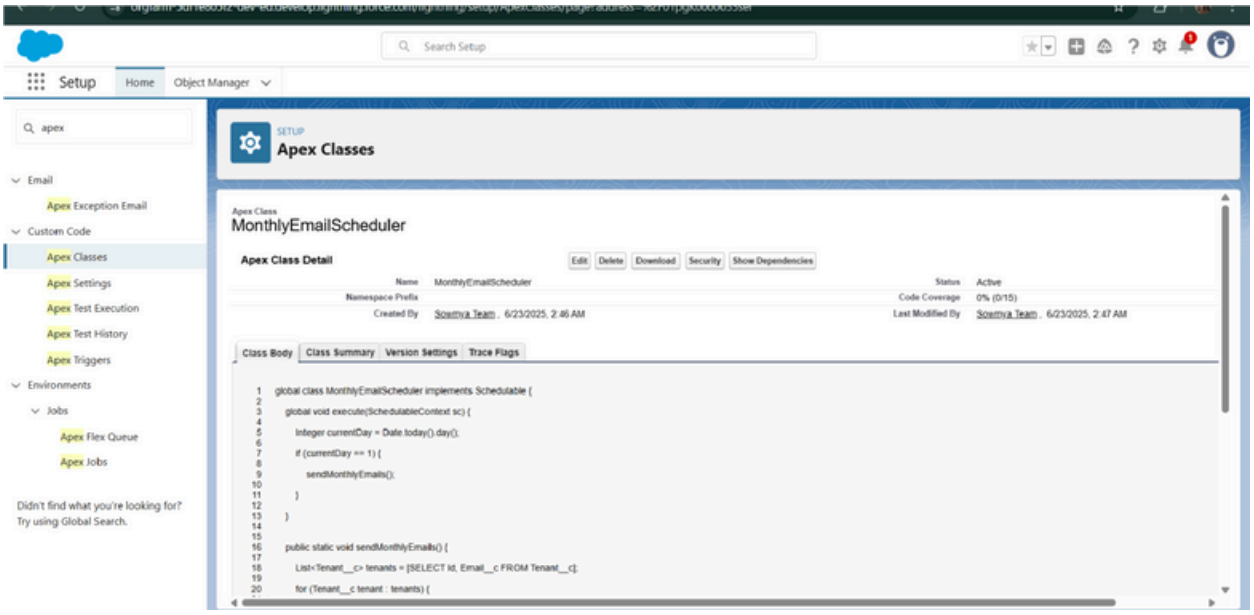


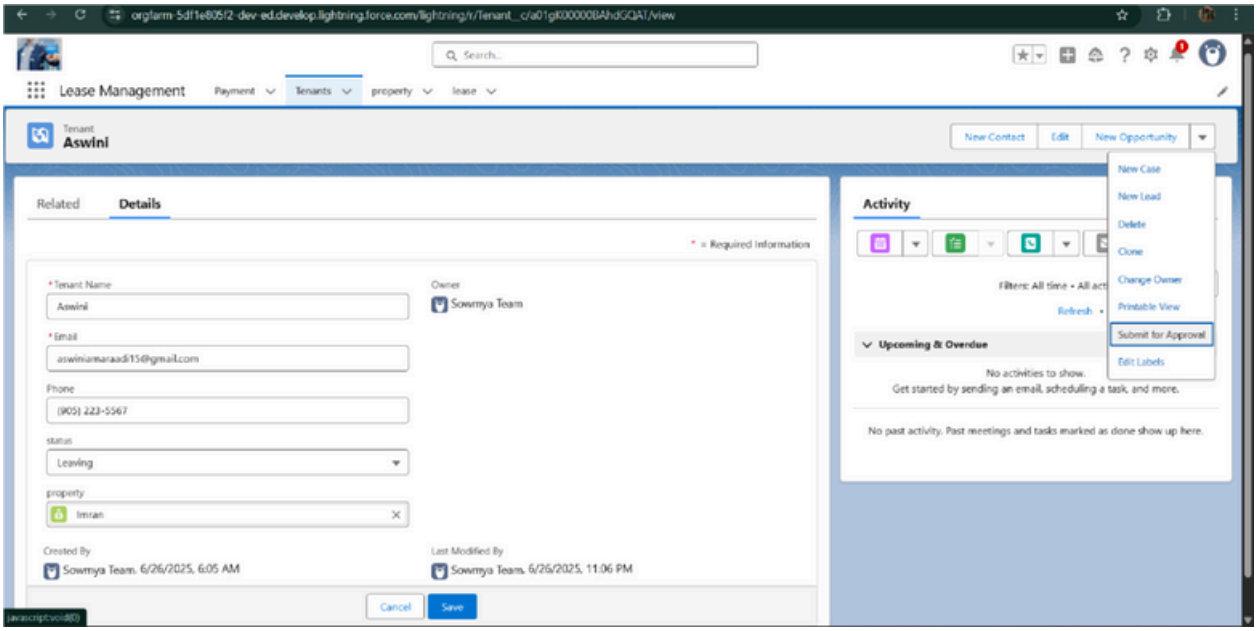
- Schedule class:
  Create an Apex
  Class

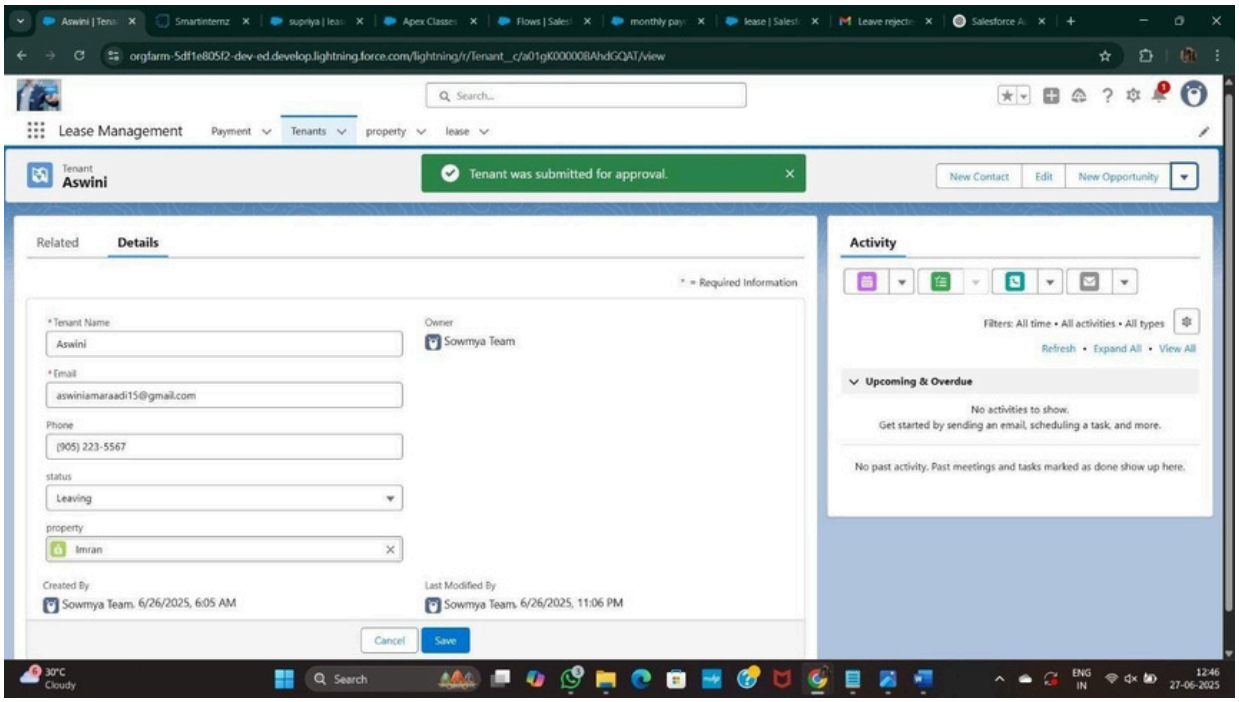


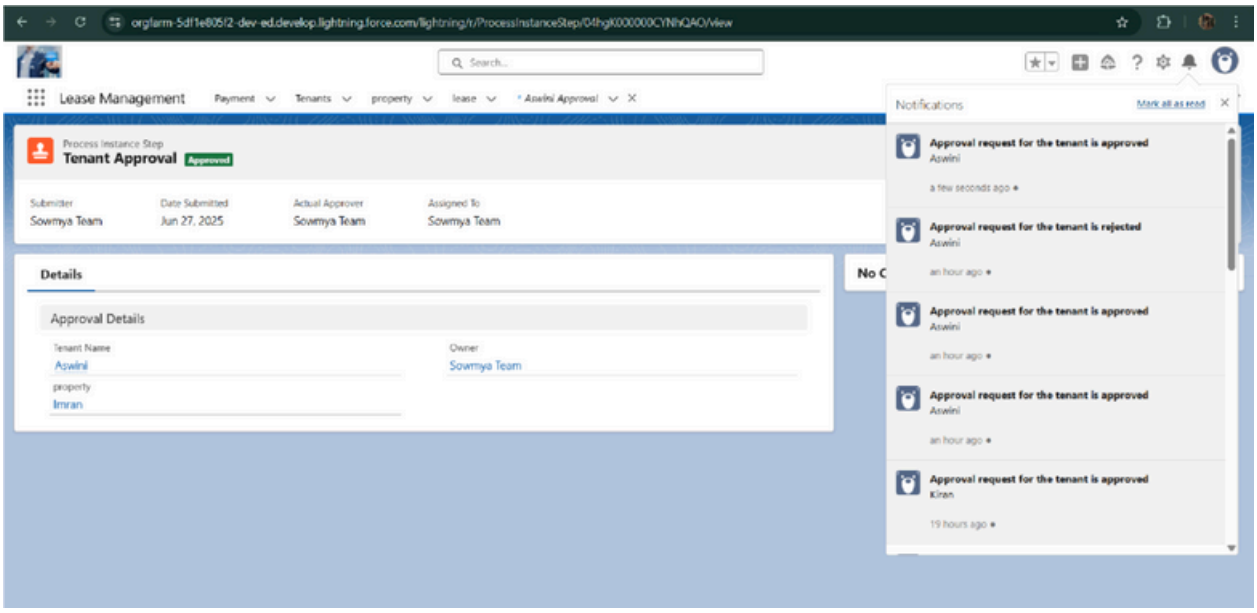


Schedule Apex class




Lease Management
Payment
Tenants
property
lease
Tenant
Aswini
New Contact
Edit
New Opportunity
New Case
New Lead
Delete
Clone
Change Owner
Printable View
Submit for Approval
Edit Labels
Related
Details
* = Required Information
*Tenant Name
Aswini
Owner
Sowmya Team
*Email
aswiniamaraadi15@gmail.com
Phone
(905) 223-5567
status
Leaving
property
Imran
Created By
Sowmya Team, 6/26/2025, 6:05 AM
Last Modified By
Sowmya Team, 6/26/2025, 11:06 PM
Cancel
Save
Activity
Upcoming & Overdue
No activities to show.
Get started by sending an email, scheduling a task, and more.
No past activity. Past meetings and tasks marked as done show up here.

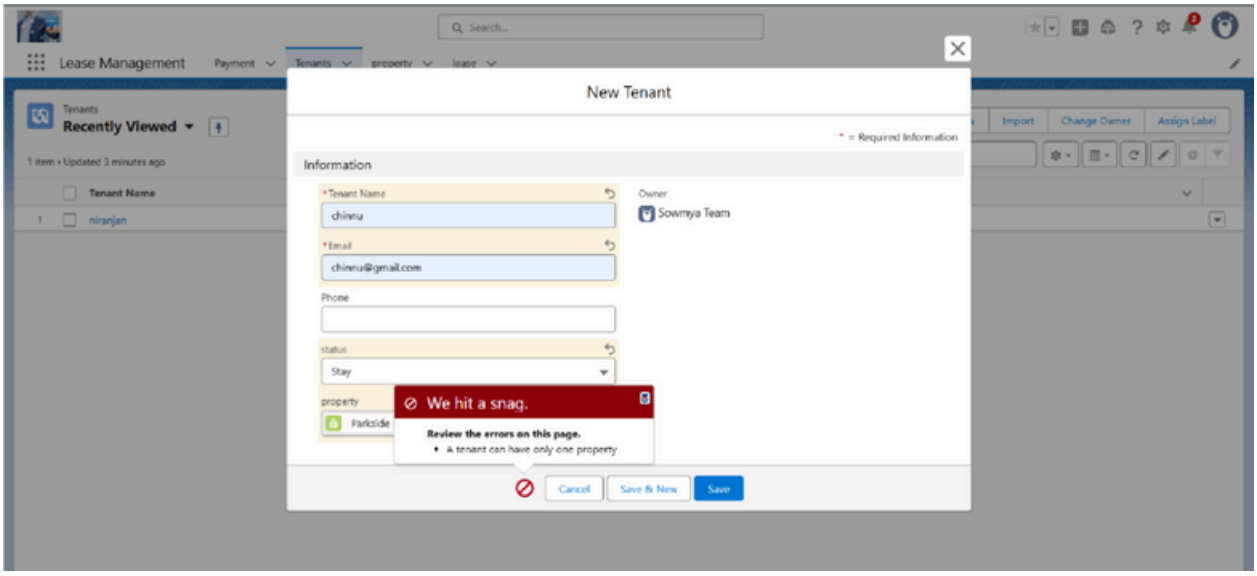

Lease Management
Payment
Tenants
property
lease
Tenant
Aswini
Tenant was submitted for approval.
New Contact
Edit
New Opportunity
Related
Details
* = Required Information
*Tenant Name
Aswini
Owner
Sowmya Team
*Email
aswiniamaraadi15@gmail.com
Phone
(905) 223-5567
status
Leaving
property
Imran
Created By
Sowmya Team, 6/26/2025, 6:05 AM
Last Modified By
Sowmya Team, 6/26/2025, 11:06 PM
Cancel
Save
Activity
Filters: All time • All activities • All types
Refresh • Expand All • View All
Upcoming & Overdue
No activities to show.
Get started by sending an email, scheduling a task, and more.
No past activity. Past meetings and tasks marked as done show up here.
30°C
Cloudy
Search
ENG
IN
12:46
27-06-2025



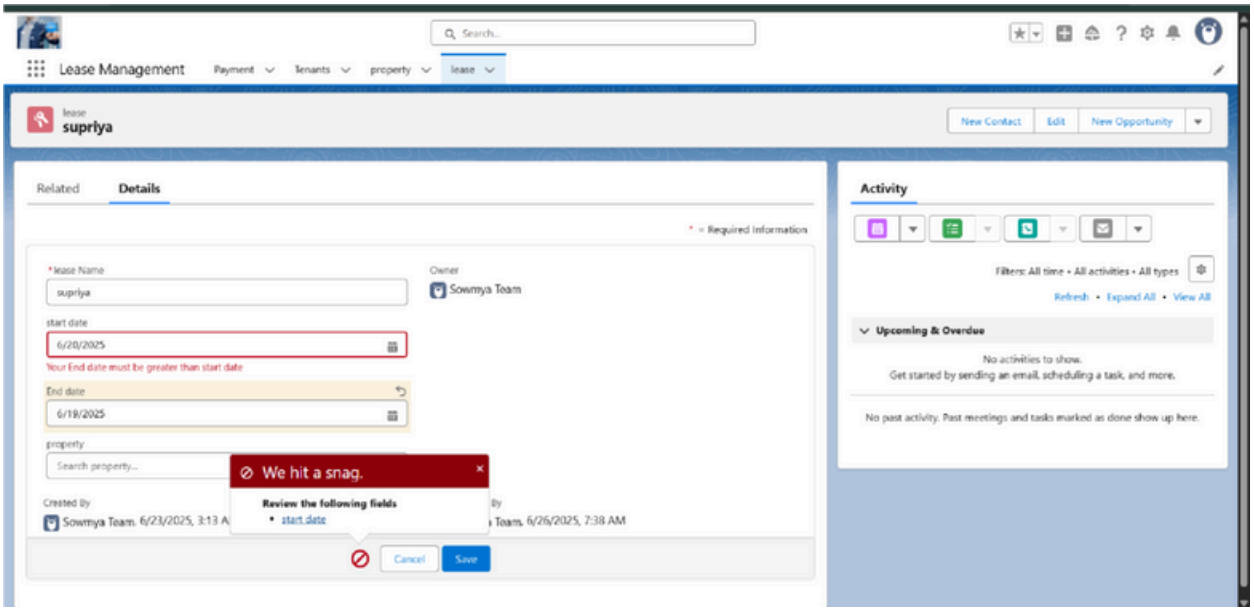
FUNCTIONAL AND PERFORMANCE TESTING

Performance Testing

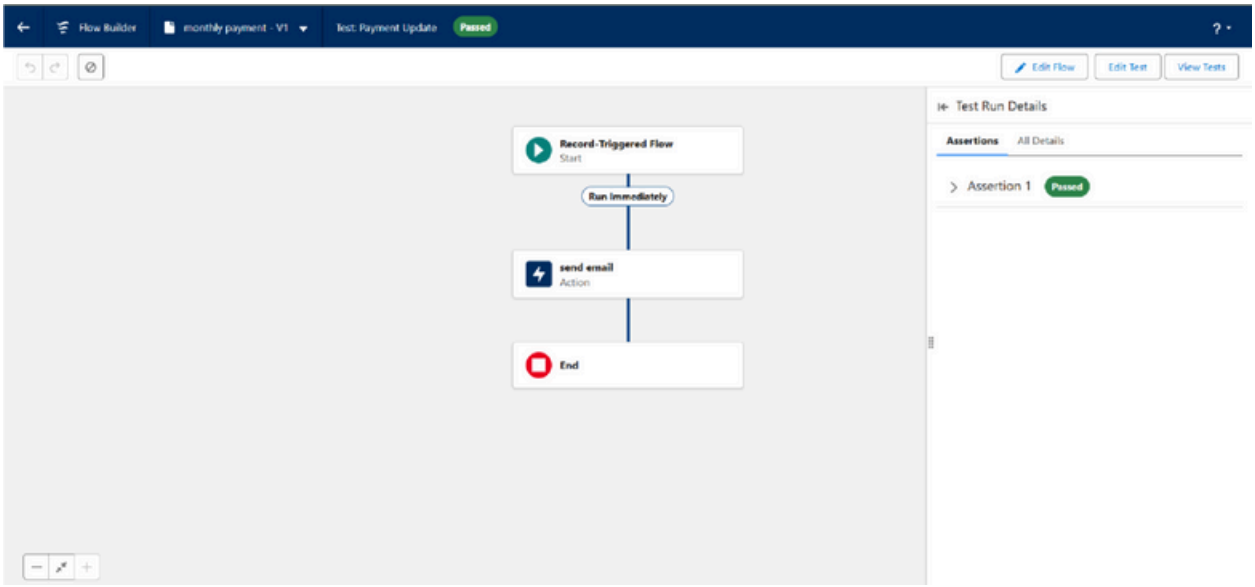
- Triggervalidationbyenteringduplicate tenant-property records



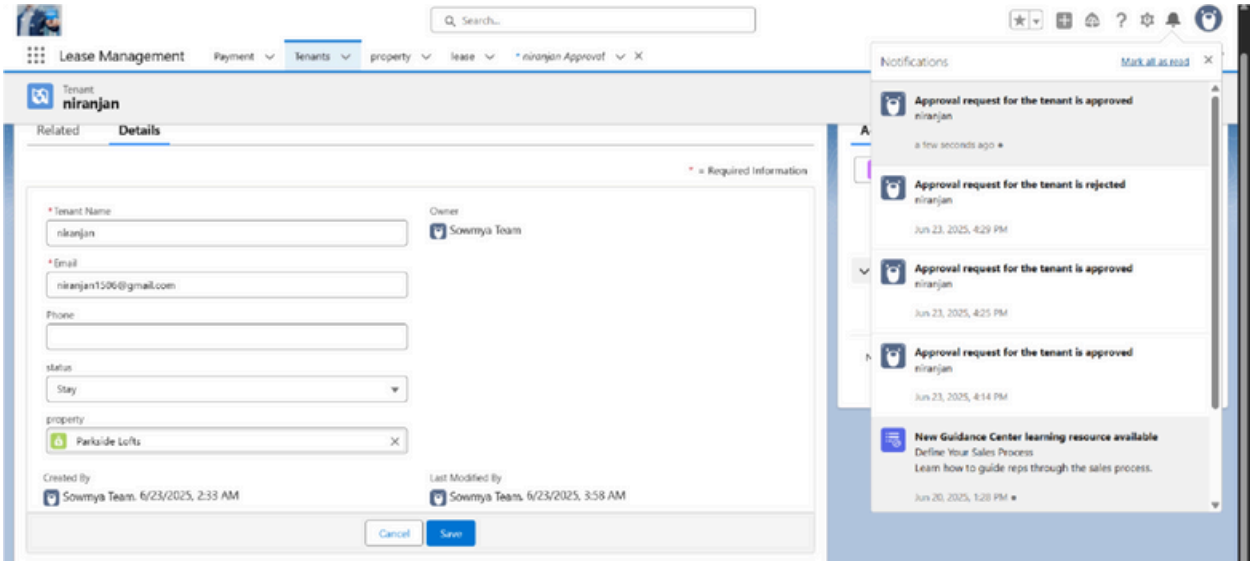
● Validation Rule checking

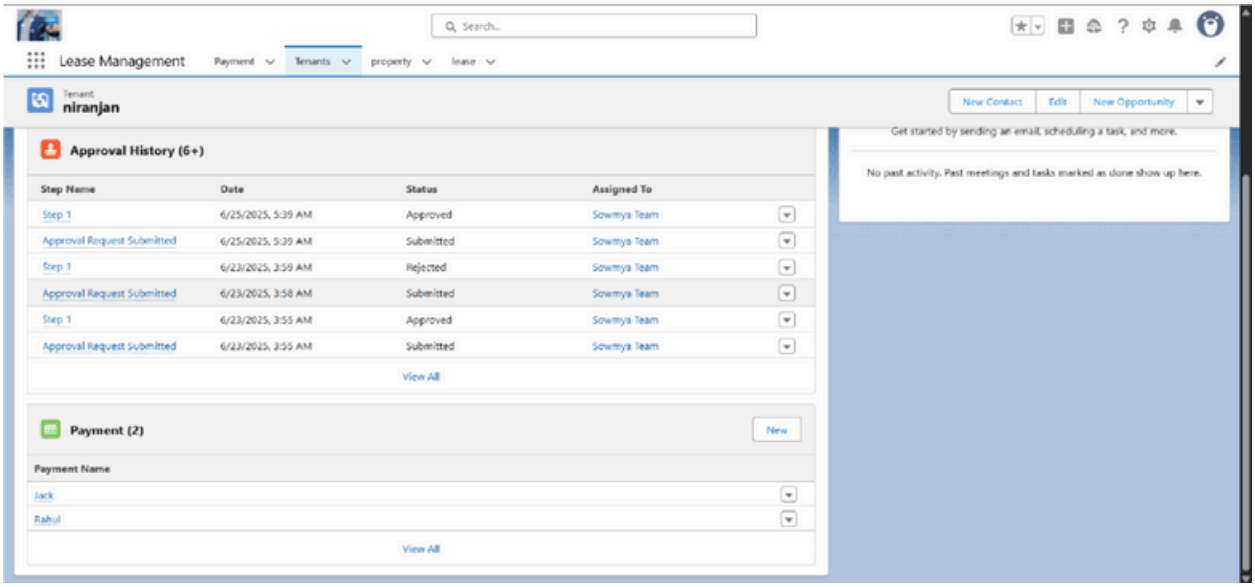


● Test flows on payment update



● Approval process validated through email alerts and status updates

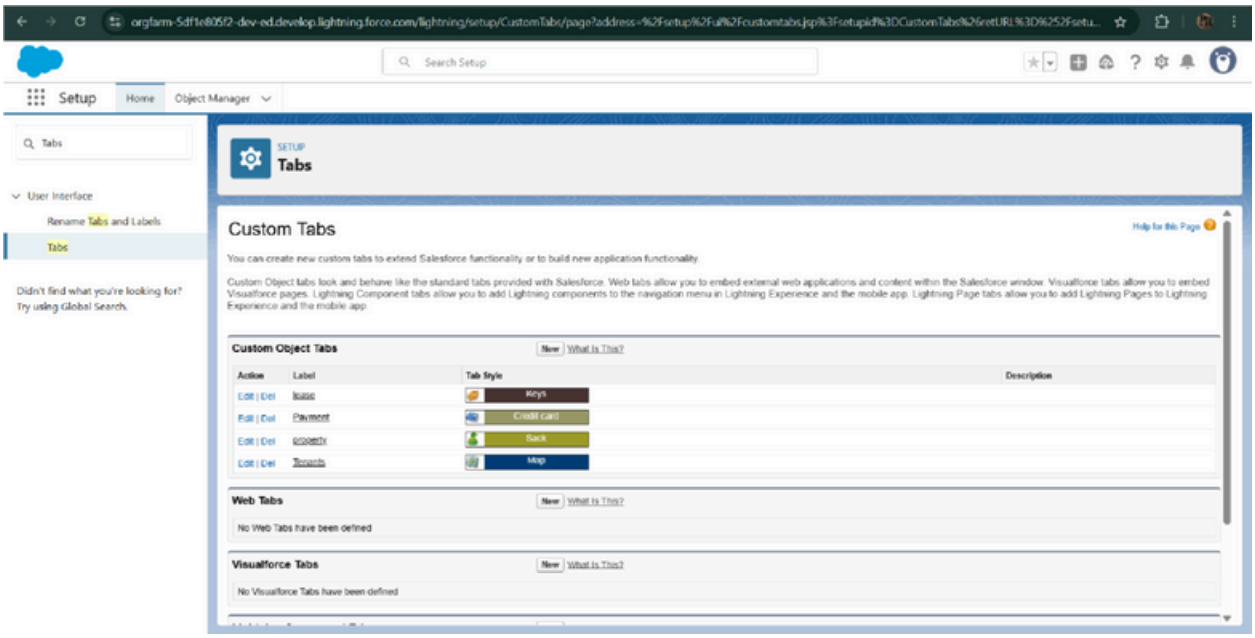

Search...
Lease Management
Payment
Tenants
property
lease
*niranjan Approval
Tenant
niranjan
Related
Details
* = Required Information
*Tenant Name
niranjan
Owner
Sowmya Team
*Email
niranjan1506@gmail.com
Phone
status
Stay
property
Parkside Lofts
Created By
Sowmya Team, 6/23/2025, 2:33 AM
Last Modified By
Sowmya Team, 6/23/2025, 3:58 AM
Cancel
Save
Notifications
Mark all as read
Approval request for the tenant is approved
niranjan
a few seconds ago
Approval request for the tenant is rejected
niranjan
Jun 23, 2025, 4:29 PM
Approval request for the tenant is approved
niranjan
Jun 23, 2025, 4:25 PM
Approval request for the tenant is approved
niranjan
Jun 23, 2025, 4:14 PM
New Guidance Center learning resource available
Define Your Sales Process
Learn how to guide reps through the sales process.
Jun 20, 2025, 1:28 PM


Search...
Lease Management
Payment
Tenants
property
lease
Tenant
niranjan
New Contact
Edit
New Opportunity
Approval History (6+)
Step Name
Date
Status
Assigned To
Step 1
6/25/2025, 5:39 AM
Approved
Sowmya Team
Approval Request Submitted
6/25/2025, 5:39 AM
Submitted
Sowmya Team
Step 1
6/23/2025, 3:59 AM
Rejected
Sowmya Team
Approval Request Submitted
6/23/2025, 3:58 AM
Submitted
Sowmya Team
Step 1
6/23/2025, 3:55 AM
Approved
Sowmya Team
Approval Request Submitted
6/23/2025, 3:55 AM
Submitted
Sowmya Team
View All
Payment (2)
New
Payment Name
Jack
Rahul
View All
Get started by sending an email, scheduling a task, and more.
No past activity. Past meetings and tasks marked as done show up here.

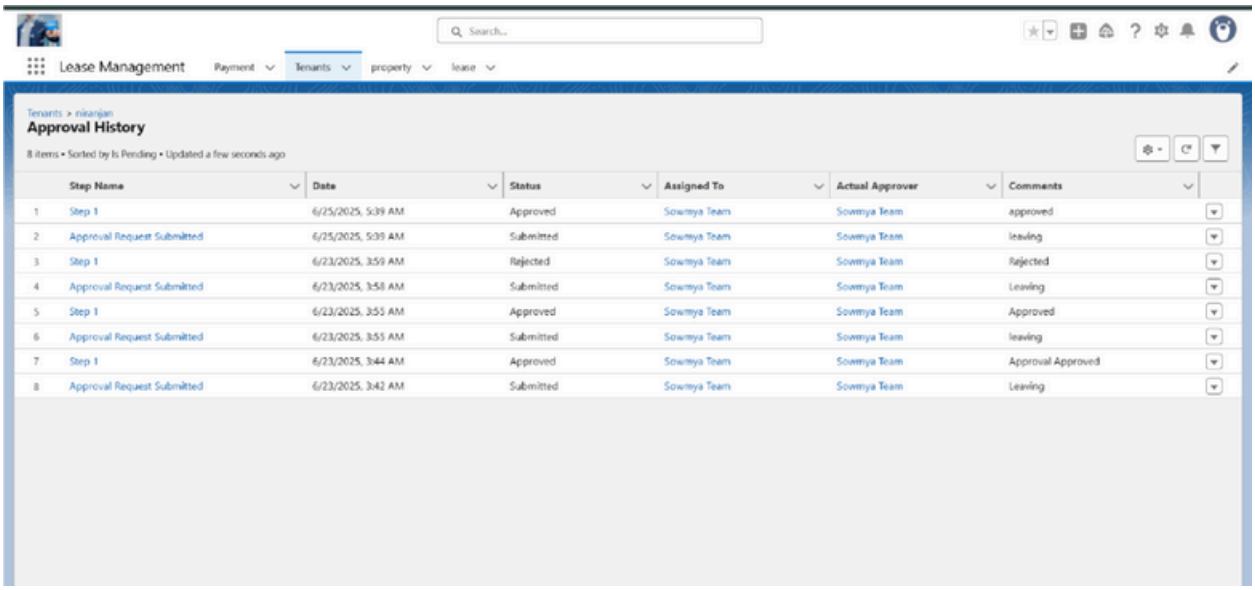
RESULTS

OutputScreenshots

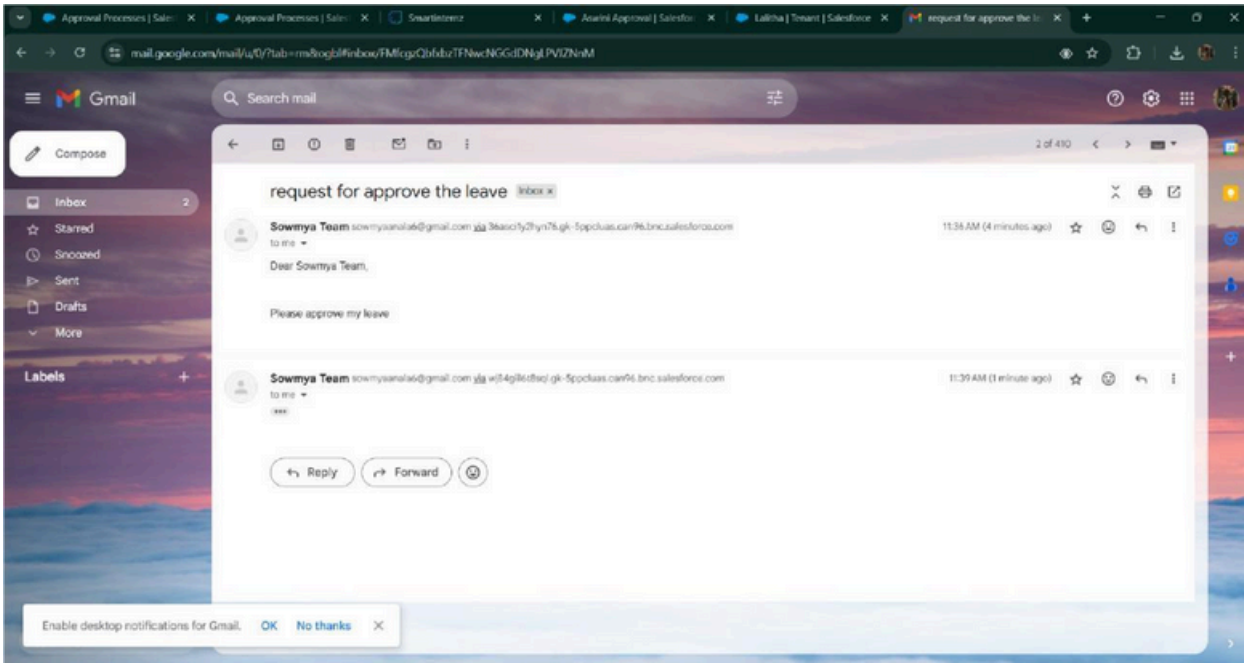
- Tabs for Property, Tenant, Lease, Payment



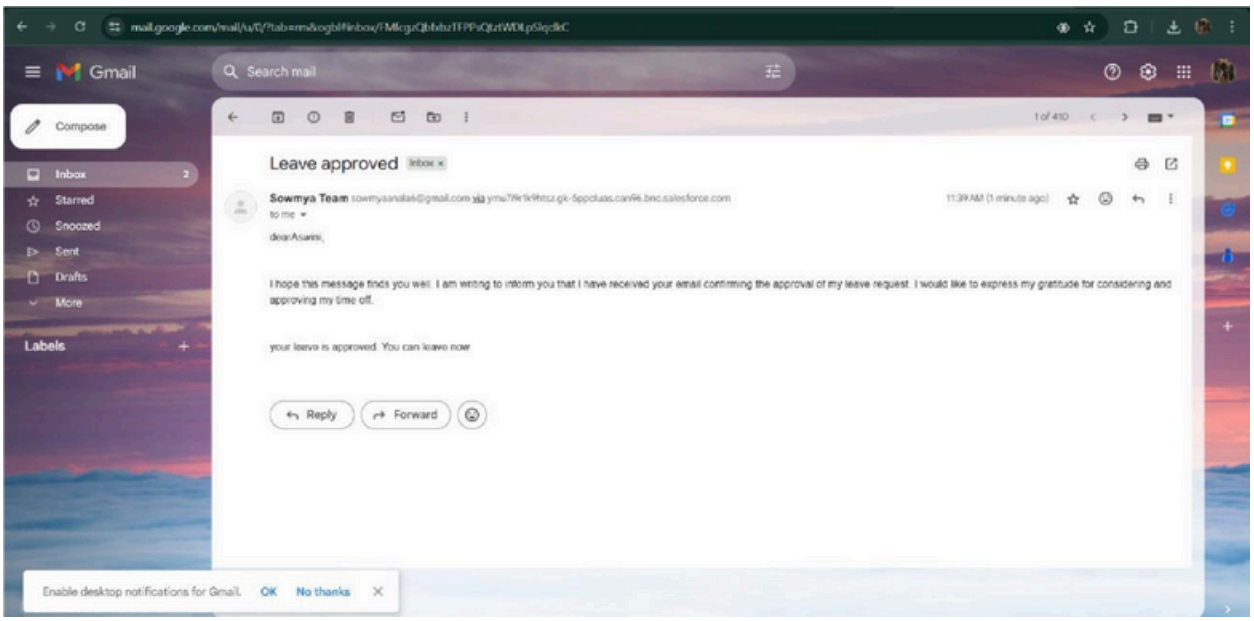
- Email alerts



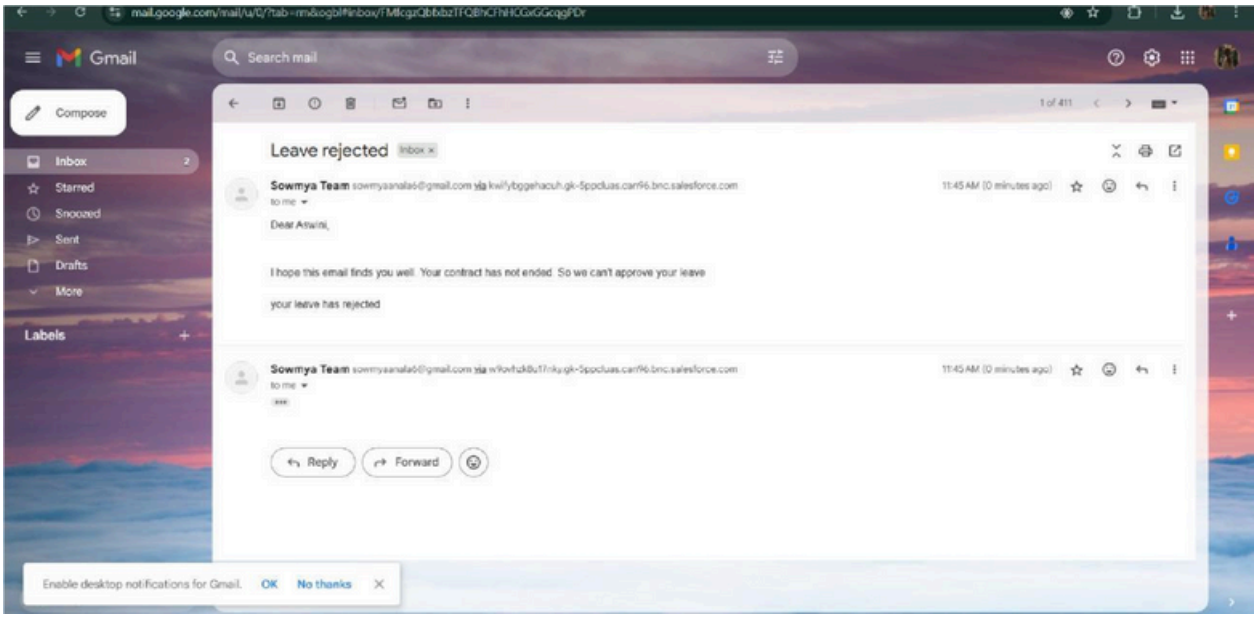
- Request for approve the leave



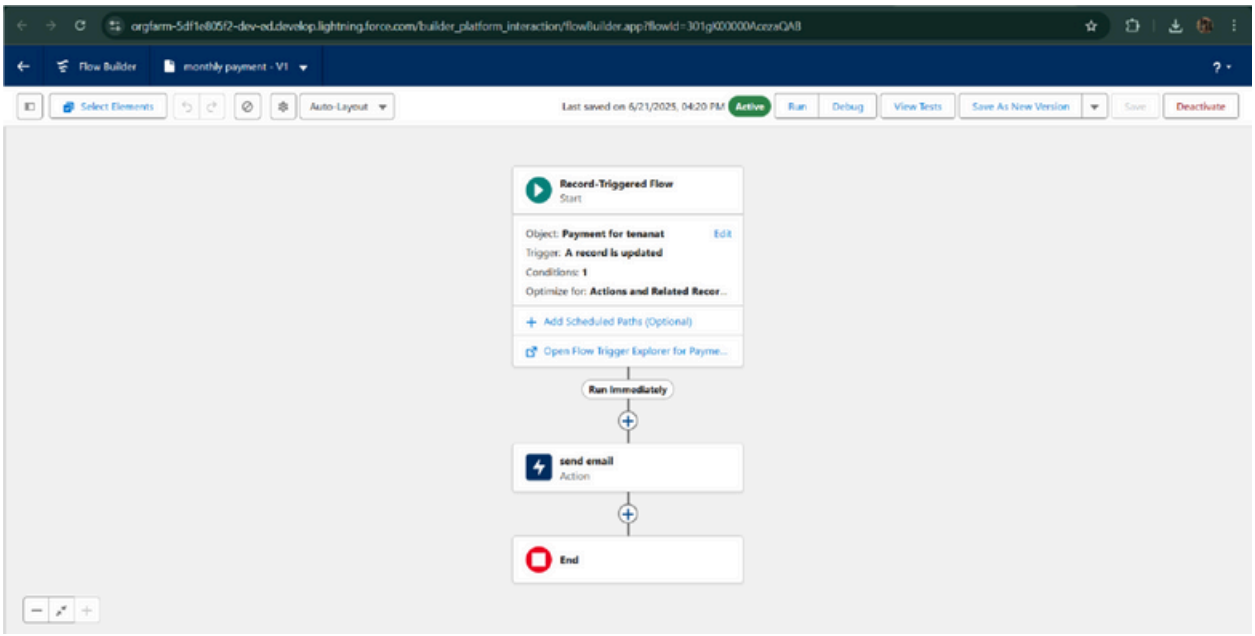
- Leave approved



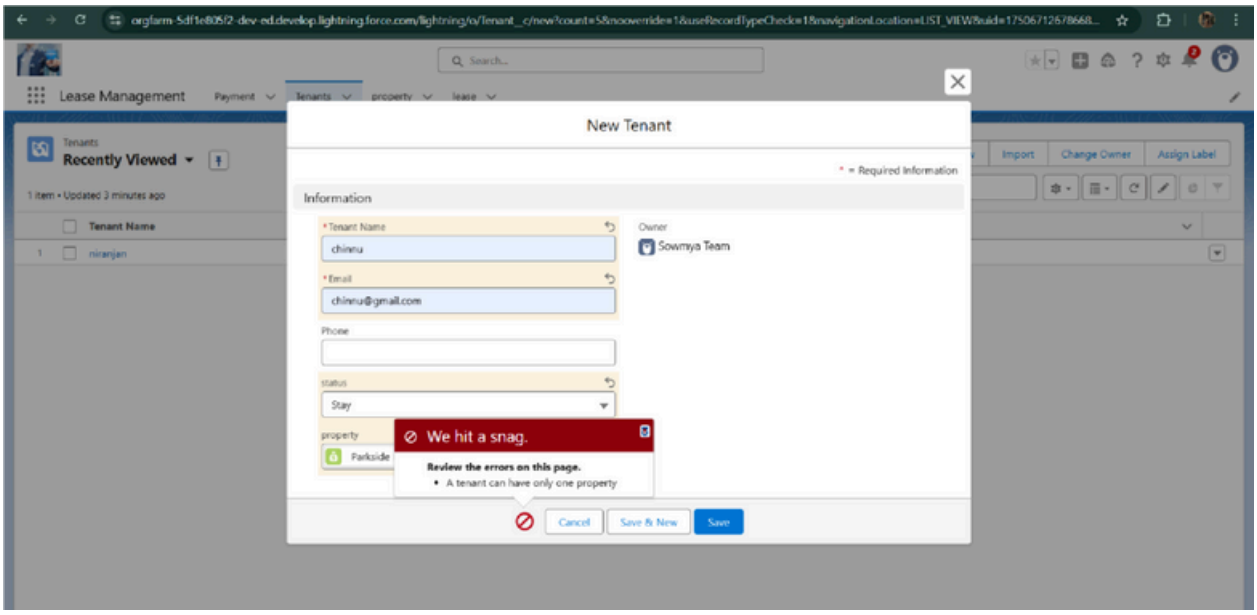
- Leave rejected



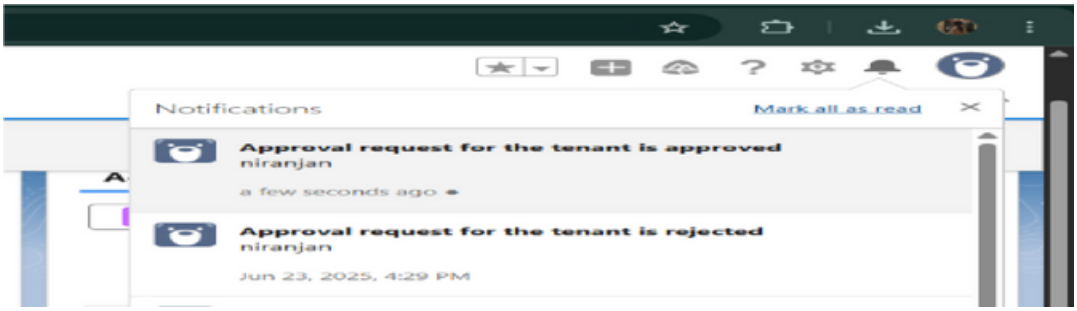
● Flow runs



● Trigger error messages



● Approval process notifications



ADVANTAGES & DISADVANTAGES

Advantages

1. Cost Predictability
   - Lease agreements typically involve fixed payments, making budgeting and forecasting easier.
2. Access to High-Value Assets
   - Allows businesses to use expensive assets (e.g., real estate, machinery, vehicles) without large upfront capital investment.

3. Flexibility

- Leasing can offer flexibility in contract length, upgrades, and expansion options.

4. Tax Benefits

- Lease payments can often be deducted as business expenses, reducing taxable income.

5. Reduced Maintenance Responsibility

- Depending on the lease type, maintenance and repairs may be the responsibility of the lessor (e.g., in triple net or full-service leases).

Disadvantages

1. Long-Term Cost

- Over time, leasing can be more expensive than purchasing the asset outright.

2. Limited Control

- Leased assets are not owned, so modifications or usage may be restricted by lease terms.

3. Contractual Obligations

- Lease agreements may lock businesses into long-term commitments, limiting flexibility if business needs change.

4. Complexity

- Managing multiple lease agreements across locations or asset types can be administratively without the right system.

5. Penalties

- Early termination, overuse, or violations of lease terms can result in penalties or legal disputes.

CONCLUSION

TheLease Management System successfully streamlines the operations of leasing through a structured, automatedSalesforce application. It improves efficiency, communication, and data accuracy for both admins and tenants.

APPENDIX

Source Code: Provided in Apex Classes and Triggers

Test.apxt:

```
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});
            }
        }
    }
```