PROJECT TITLE

College Name: LRG COLLEGE FOR WOMEN

College Code :bru07

TEAM ID: NM2025TMID28009

TEAM MEMBERS:

Team LeaderName: Sandhiya S

Email: sandhiyaspsm@gmail.com

Team Member1: ShreeDharshini S

Email: shreeshreedharshini@gmail.com

Team Member: Saranya K

Email: saranyakumar.sarojini@gmail.com

Team Member: Harini P

Email: harini8041@gmail.com

1.INTRODUCTION

1.1 Project Overview

The Lease Management System is a Salesforce-based application designed to streamline the processes associated with leasing real estate properties. It handles tenant management, lease

1

contracts, payments, and communication with automation features such as flows, approval processes, and email alerts.

1.2 Purpose

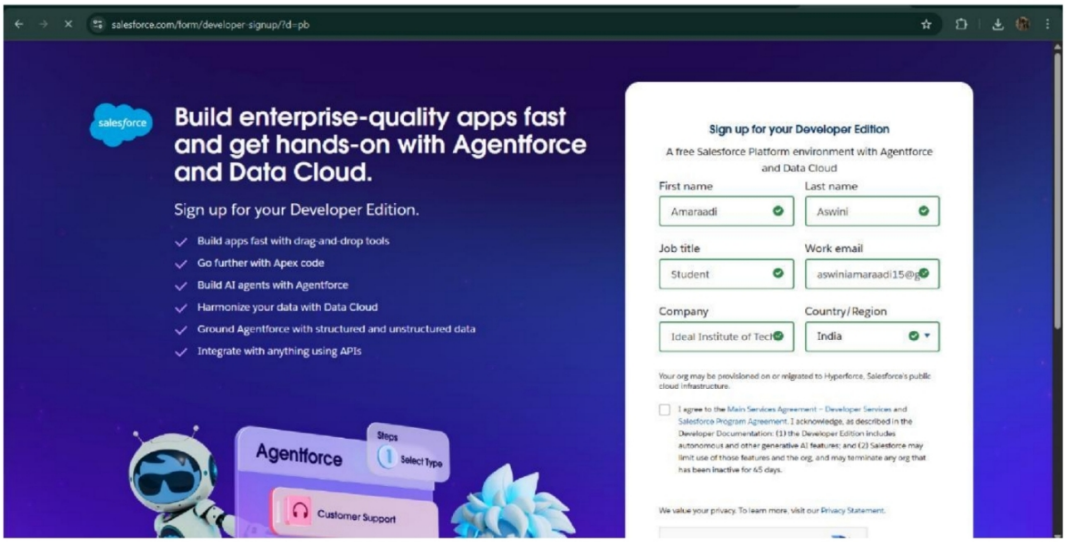
The main objective of the project is to enable organizations to efficiently manage properties, tenants, and lease-related activities. It reduces manual intervention, improves accuracy, and ensures better compliance and communication.

DEVELOPMENT PHASE

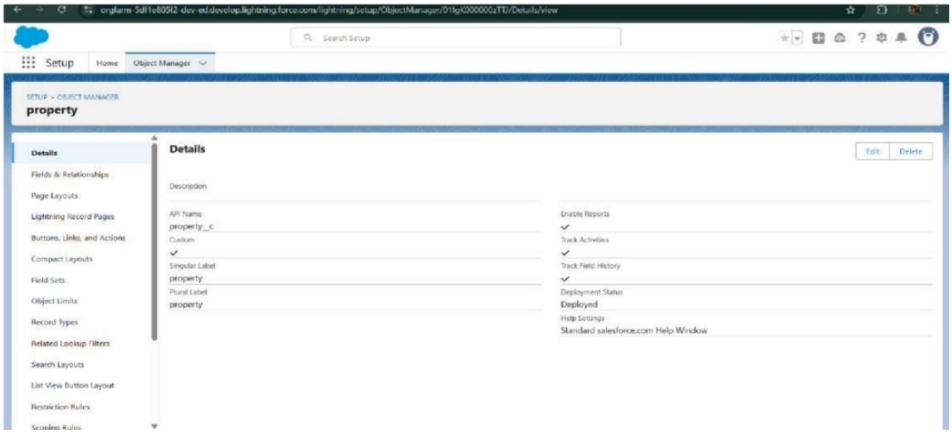
Creating Developer Account:

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

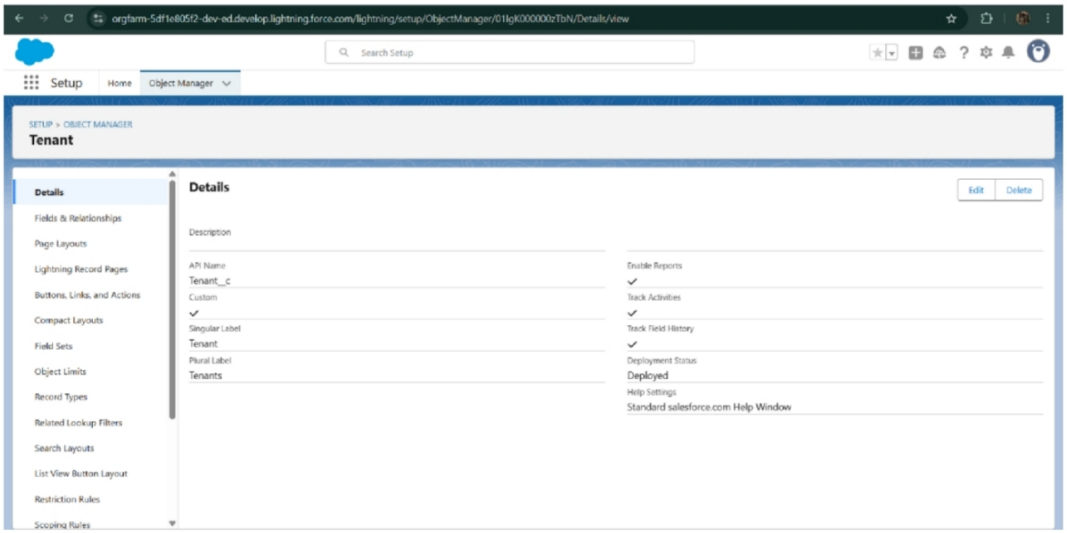
2

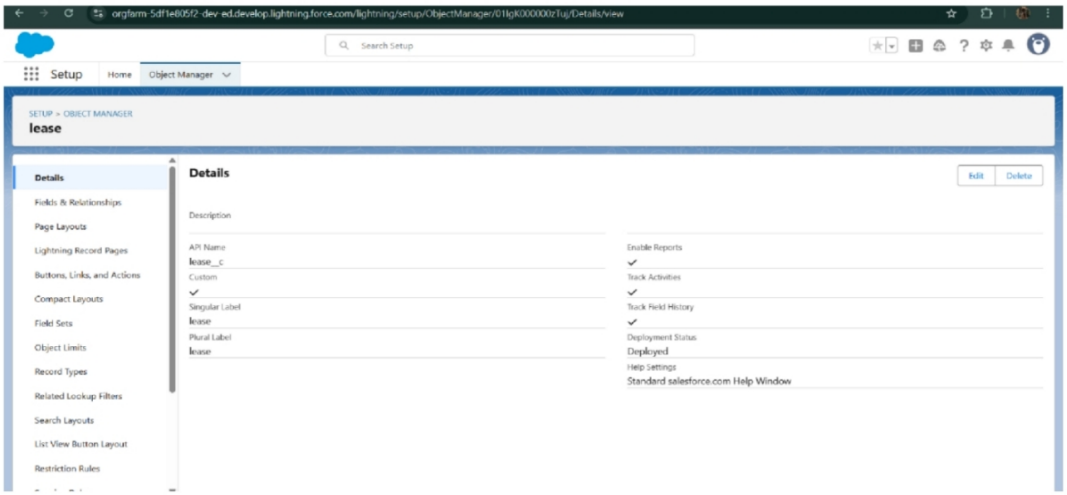


- Created objects: Property, Tenant, Lease, Payment

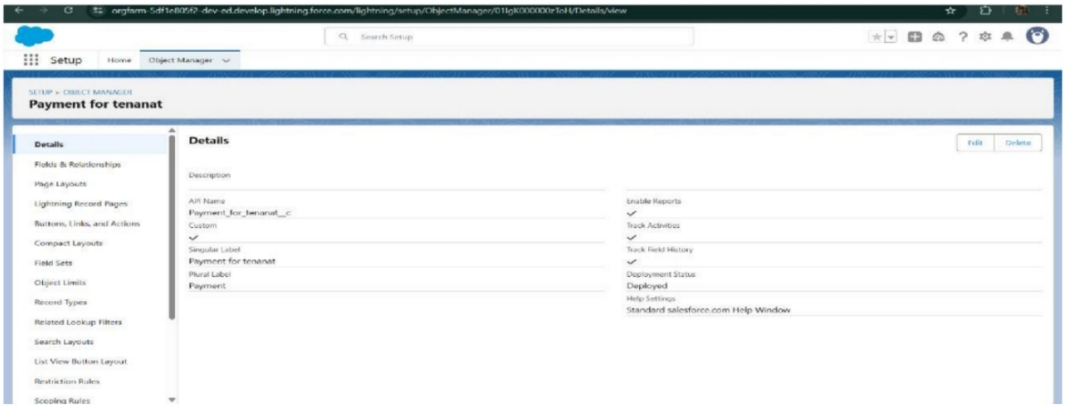


3

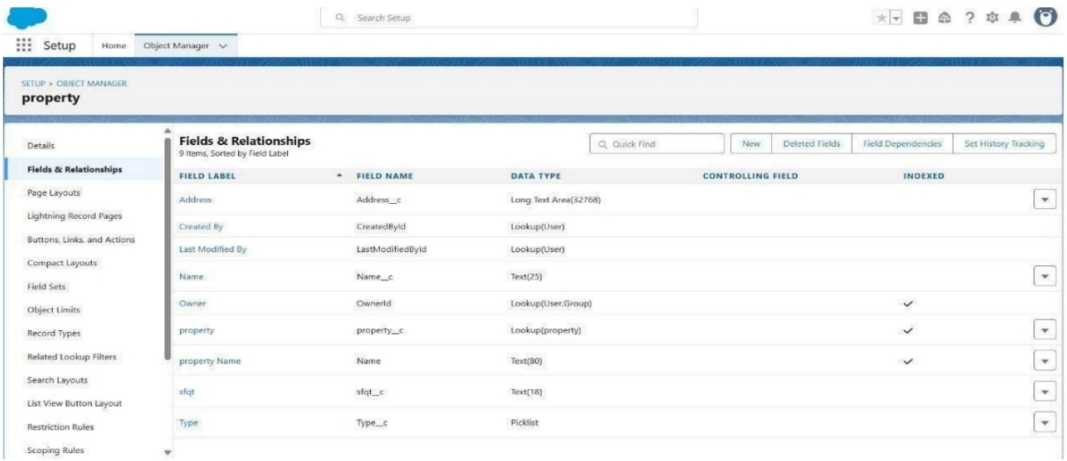

Setup
Home
Object Manager
SETUP > OBJECT MANAGER
Tenant
Details
Fields & Relationships
Page Layouts
Lightning Record Pages
Buttons, Links, and Actions
Compact Layouts
Field Sets
Object Limits
Record Types
Related Lookup Filters
Search Layouts
List View Button Layout
Restriction Rules
Scoping Rules
Edit
Delete
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


Setup
Home
Object Manager
SETUP > OBJECT MANAGER
lease
Details
Fields & Relationships
Page Layouts
Lightning Record Pages
Buttons, Links, and Actions
Compact Layouts
Field Sets
Object Limits
Record Types
Related Lookup Filters
Search Layouts
List View Button Layout
Restriction Rules
Edit
Delete
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

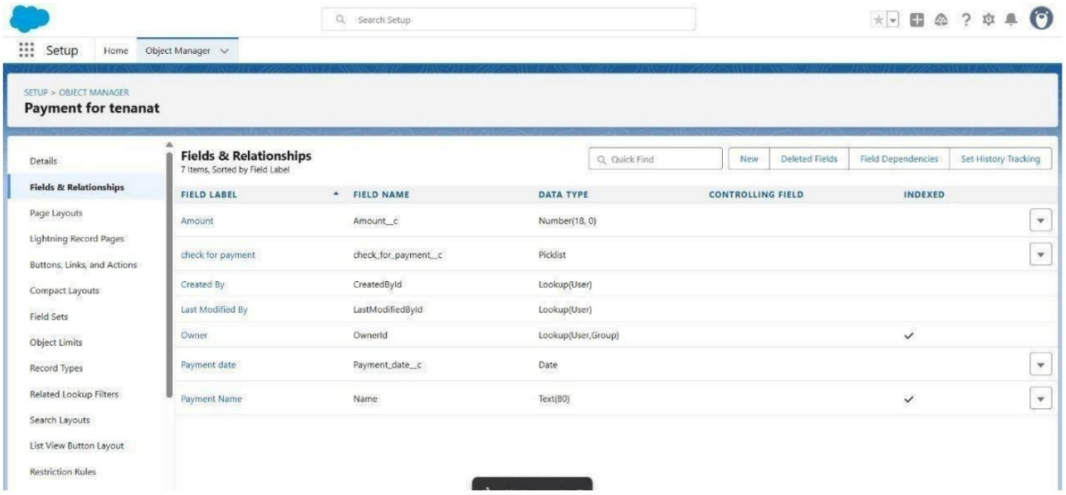
4

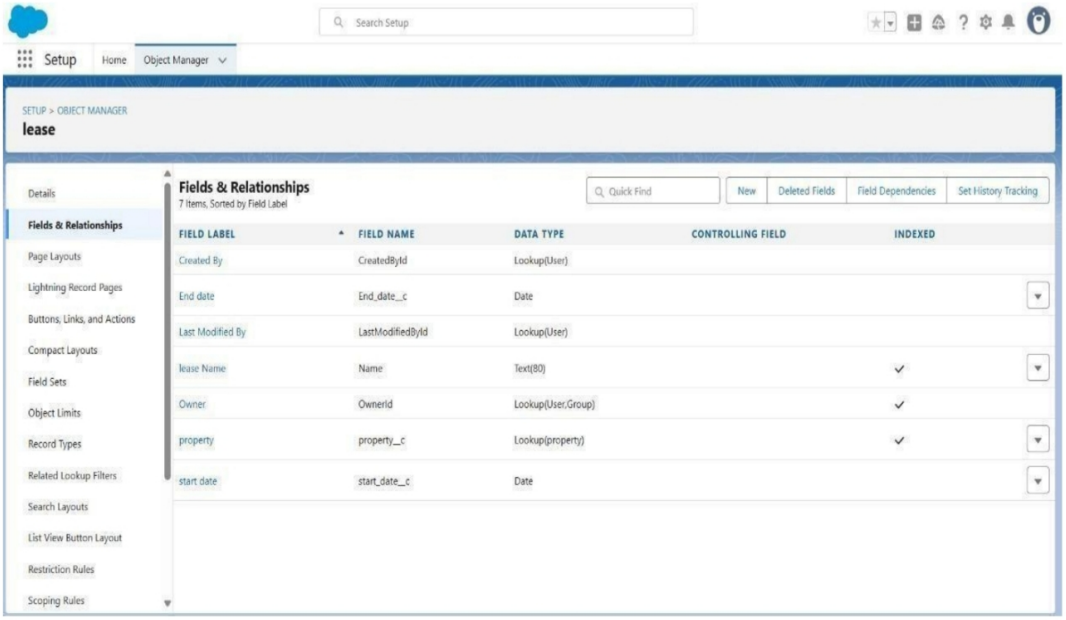


- Configured fields and relationships

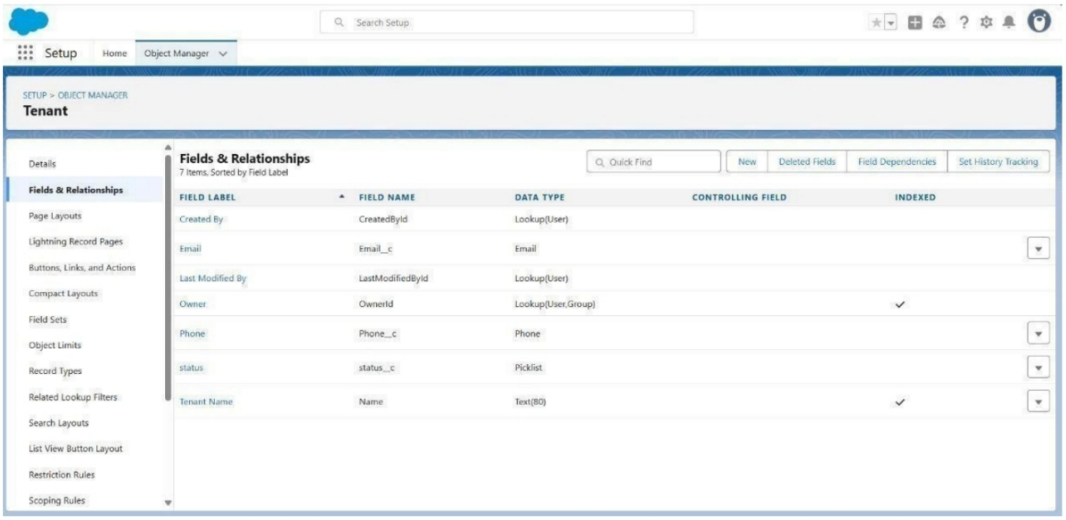


5

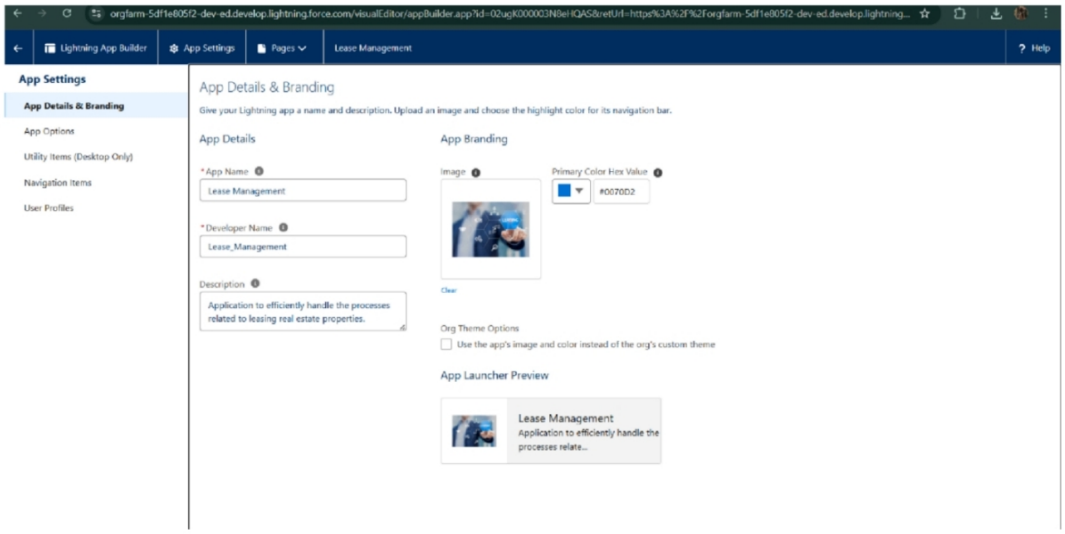

Search Setup
Setup
Home
Object Manager
SETUP > OBJECT MANAGER
Payment for tenanat
Details
Fields & Relationships
Page Layouts
Lightning Record Pages
Buttons, Links, and Actions
Compact Layouts
Field Sets
Object Limits
Record Types
Related Lookup Filters
Search Layouts
List View Button Layout
Restriction Rules
Fields & Relationships
7 Items, Sorted by Field Label
Quick Find
New
Deleted Fields
Field Dependencies
Set History Tracking
FIELD LABEL
FIELD NAME
DATA TYPE
CONTROLLING FIELD
INDEXED
Amount
Amount__c
Number(18, 0)
check for payment
check_for_payment__c
Picklist
Created By
CreatedById
Lookup(User)
Last Modified By
LastModifiedById
Lookup(User)
Owner
OwnerId
Lookup(User,Group)
Payment date
Payment_date__c
Date
Payment Name
Name
Text(80)


Search Setup
Setup
Home
Object Manager
SETUP > OBJECT MANAGER
lease
Details
Fields & Relationships
Page Layouts
Lightning Record Pages
Buttons, Links, and Actions
Compact Layouts
Field Sets
Object Limits
Record Types
Related Lookup Filters
Search Layouts
List View Button Layout
Restriction Rules
Scoping Rules
Fields & Relationships
7 Items, Sorted by Field Label
Quick Find
New
Deleted Fields
Field Dependencies
Set History Tracking
FIELD LABEL
FIELD NAME
DATA TYPE
CONTROLLING FIELD
INDEXED
Created By
CreatedById
Lookup(User)
End date
End_date__c
Date
Last Modified By
LastModifiedById
Lookup(User)
lease Name
Name
Text(80)
Owner
OwnerId
Lookup(User,Group)
property
property__c
Lookup(property)
start date
start_date__c
Date

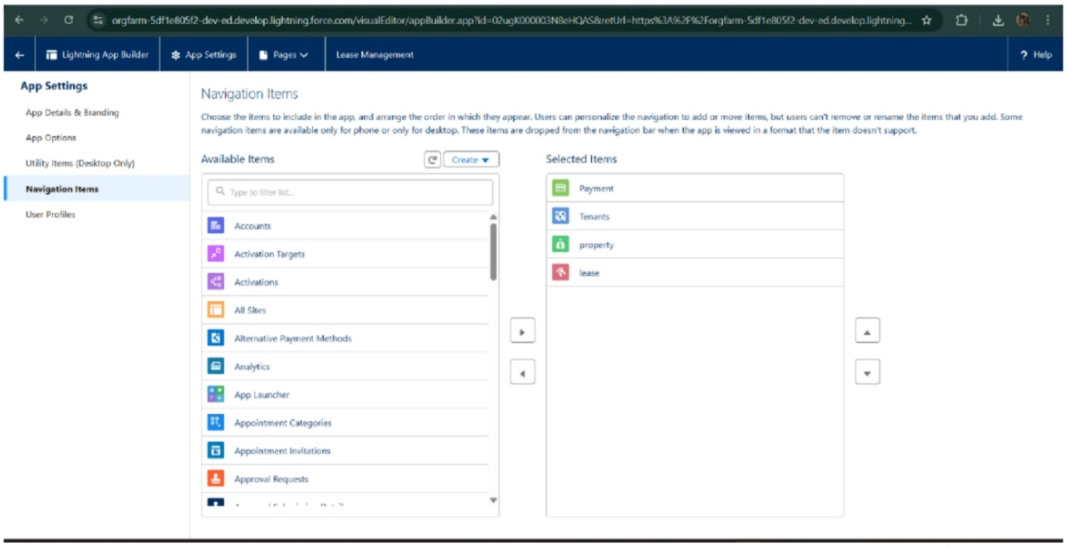
6

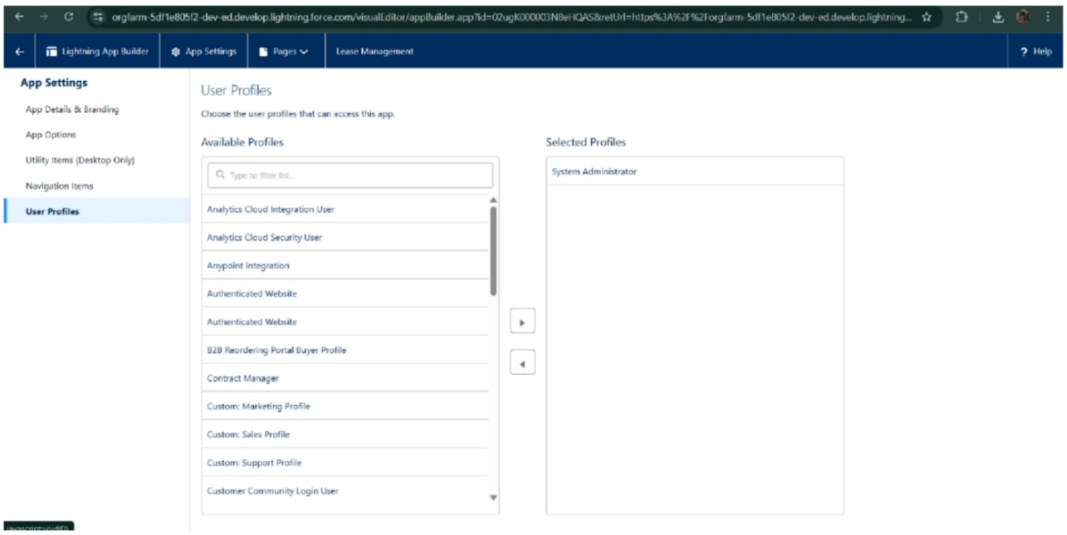


- Developed Lightning App with relevant tabs

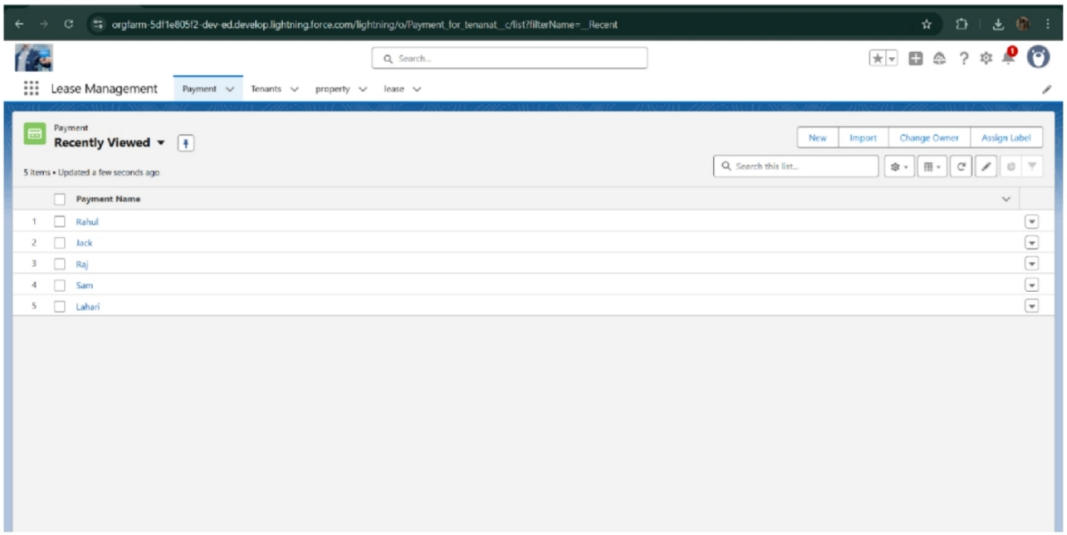


7

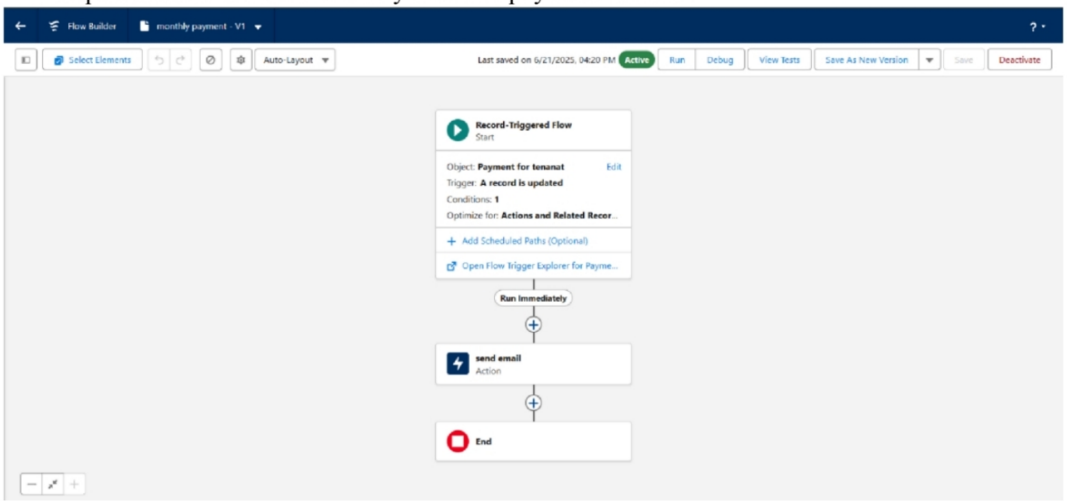

Lightning App Builder
App Settings
Pages
Lease Management
Help
App Settings
App Details & Branding
App Options
Utility Items (Desktop Only)
Navigation Items
User Profiles
Navigation Items
Choose the items to include in the app, and arrange the order in which they appear. Users can personalize the navigation to add or move items, but users can't remove or rename the items that you add. Some navigation items are available only for phone or only for desktop. These items are dropped from the navigation bar when the app is viewed in a format that the item doesn't support.
Available Items
Create
Type to filter list...
Accounts
Activation Targets
Activations
All Sites
Alternative Payment Methods
Analytics
App Launcher
Appointment Categories
Appointment Invitations
Approval Requests
Selected Items
Payment
Tenants
property
lease


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
Type to filter list...
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
Selected Profiles
System Administrator

8

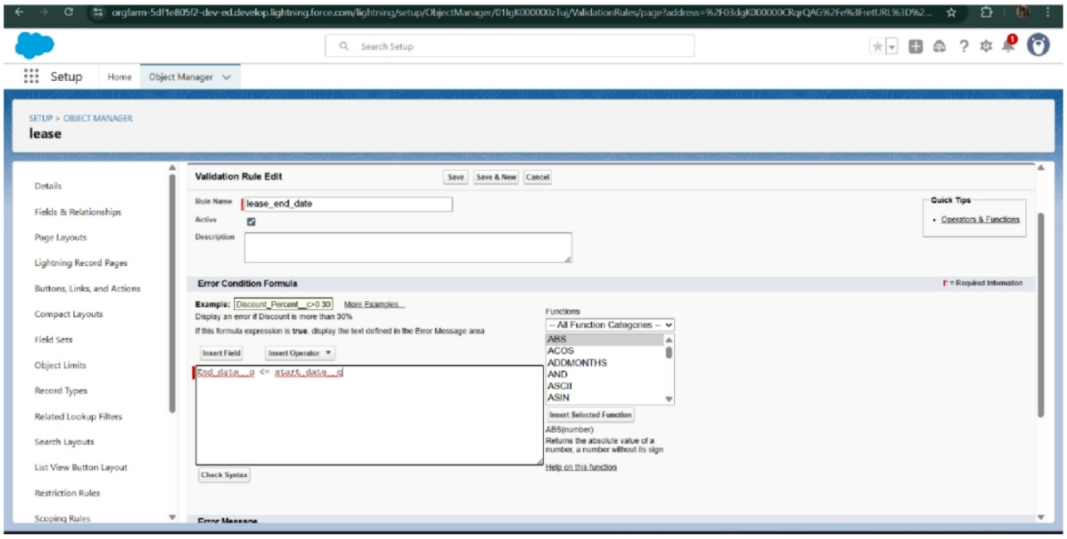


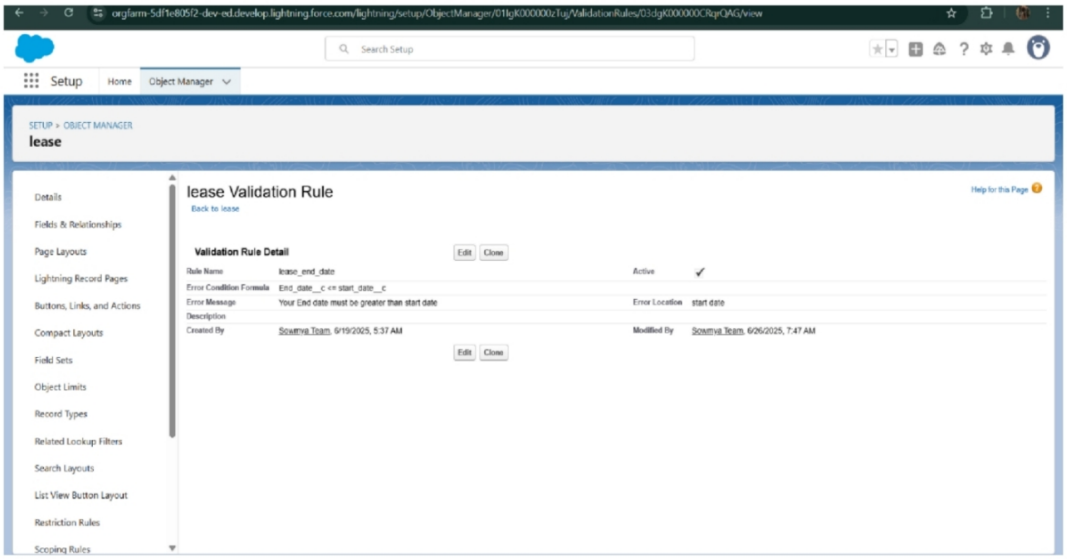
- Implemented Flows for monthly rent and payment success



- To create a validation rule to a Lease Object

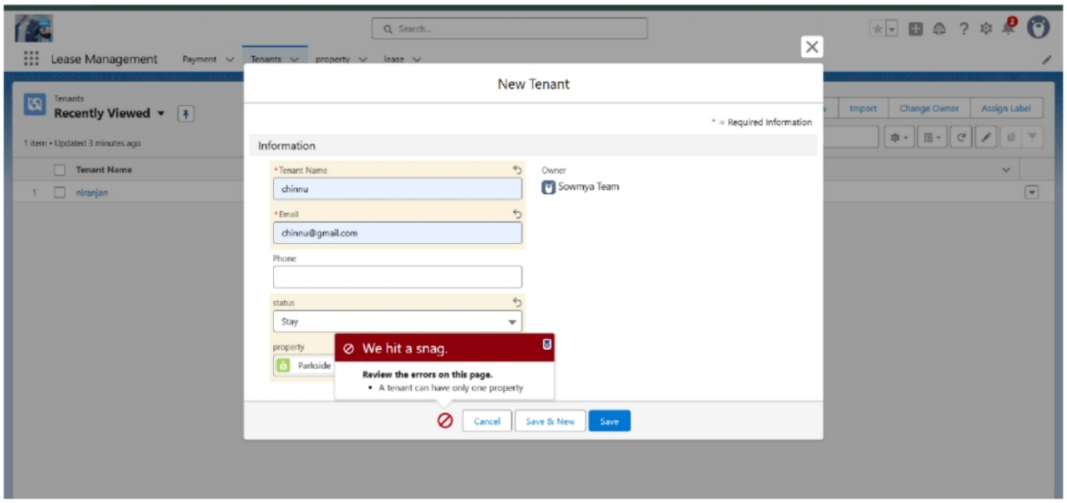
9



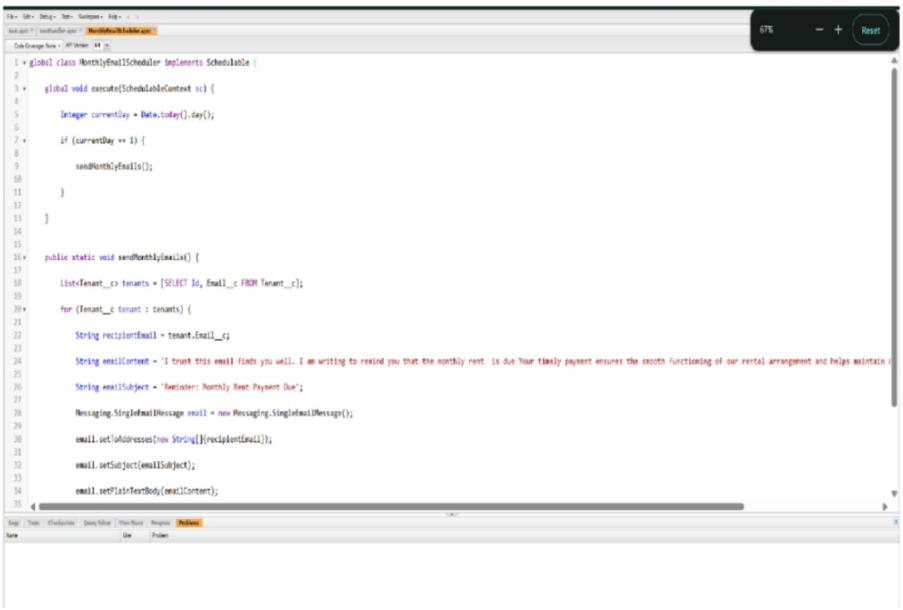


- Added Apex trigger to restrict multiple tenants per property

10

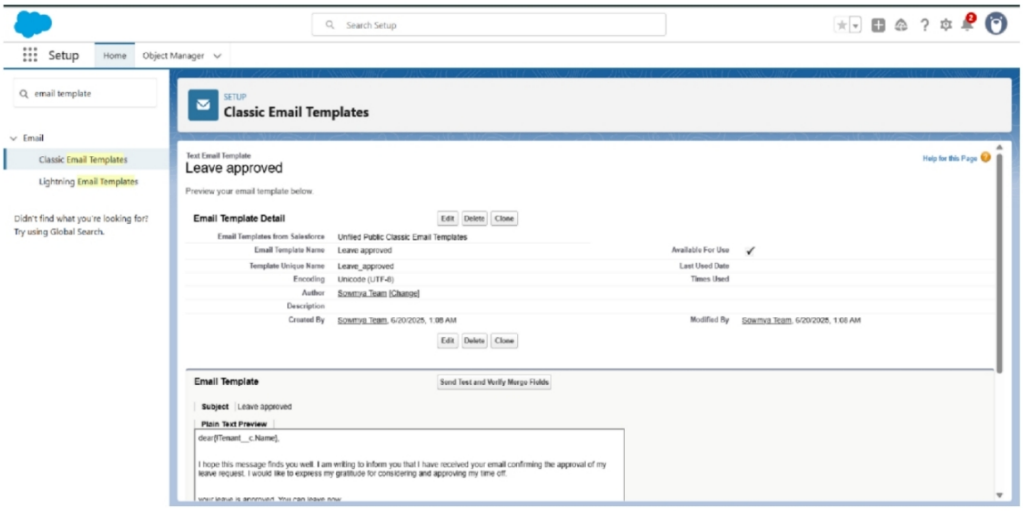


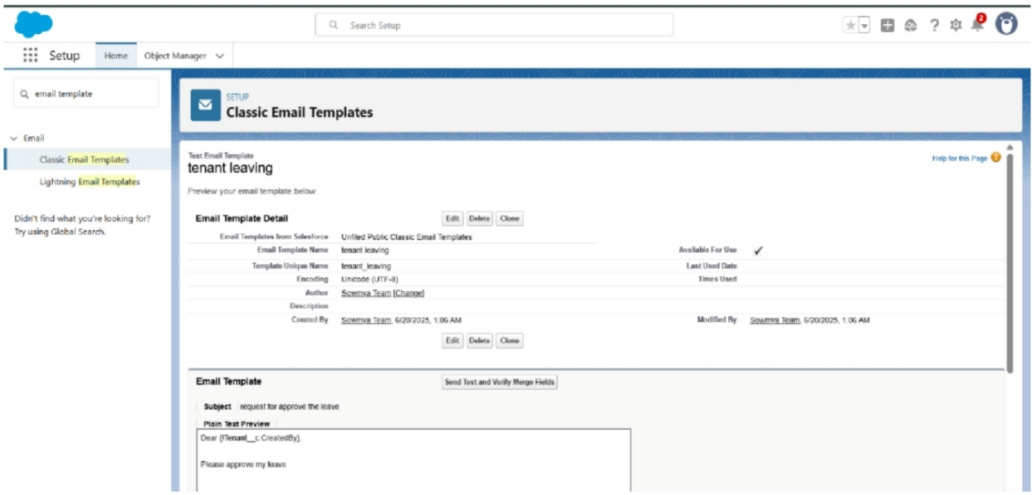
- Scheduled monthly reminder emails using Apex class



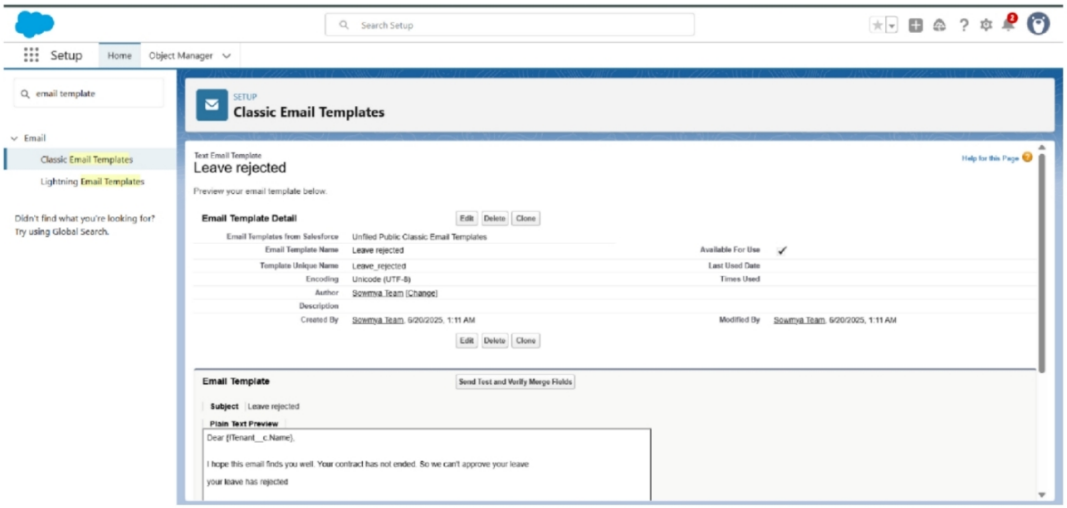
11

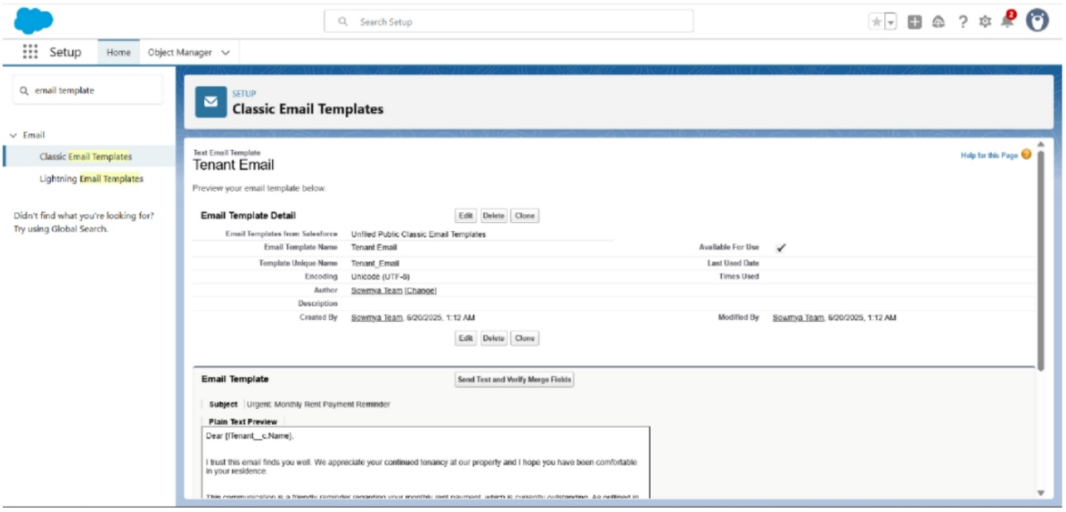
- Built and tested email templates for leave request, approval, rejection, payment, and reminders



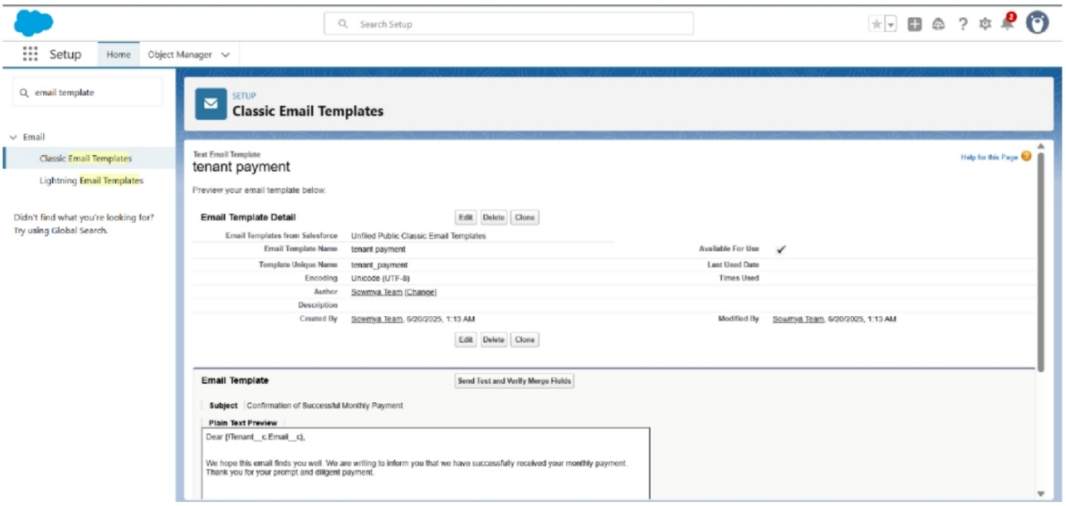


12


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
Leave rejected
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
Leave rejected
Available For Use
Template Unique Name
Leave_rejected
Last Used Date
Encoding
Unicode (UTF-8)
Times Used
Author
Sowmya Team [Change]
Description
Created By
Sowmya Team, 6/20/2025, 1:11 AM
Modified By
Sowmya Team, 6/20/2025, 1:11 AM
Email Template
Send Test and Verify Merge Fields
Subject
Leave rejected
Plain Text Preview
Dear {!Tenant__c.Name},
I hope this email finds you well. Your contract has not ended. So we can't approve your leave
your leave has rejected

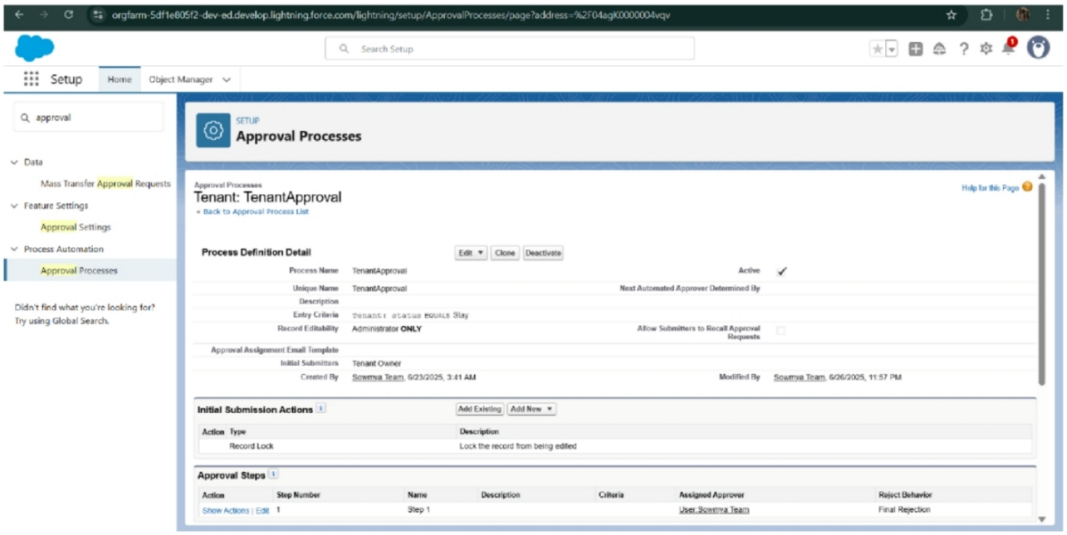

Search Setup
Setup
Home
Object Manager
email template
Email
Classic Email Templates
Lightning Email Templates
Didn't find what you're looking for?
Try using Global Search.
SETUP
Classic Email Templates
Text Email Template
Tenant Email
Help for this Page
Preview your email template below.
Email Template 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.

13



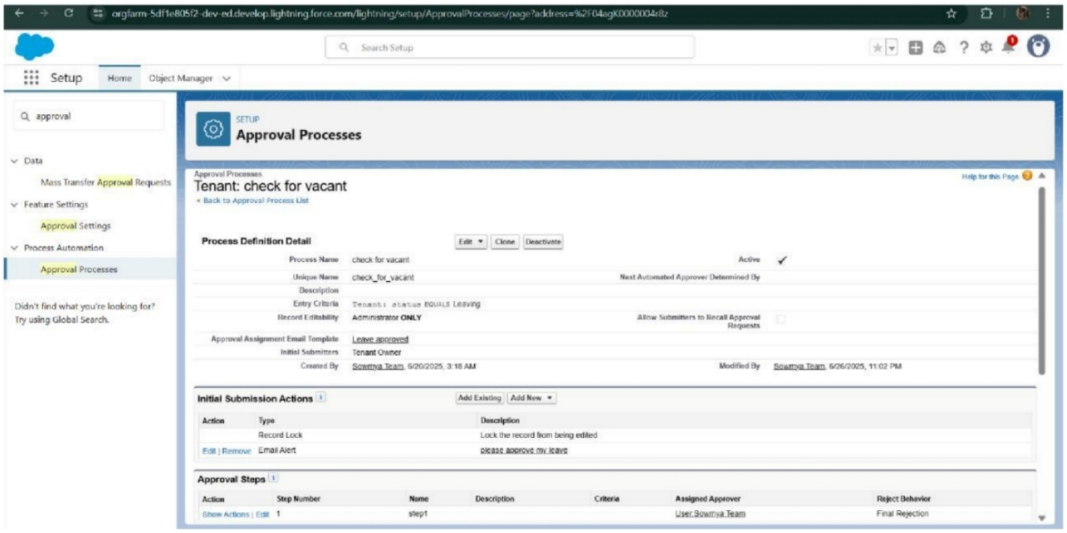
- Approval Process creation

For Tenant Leaving:



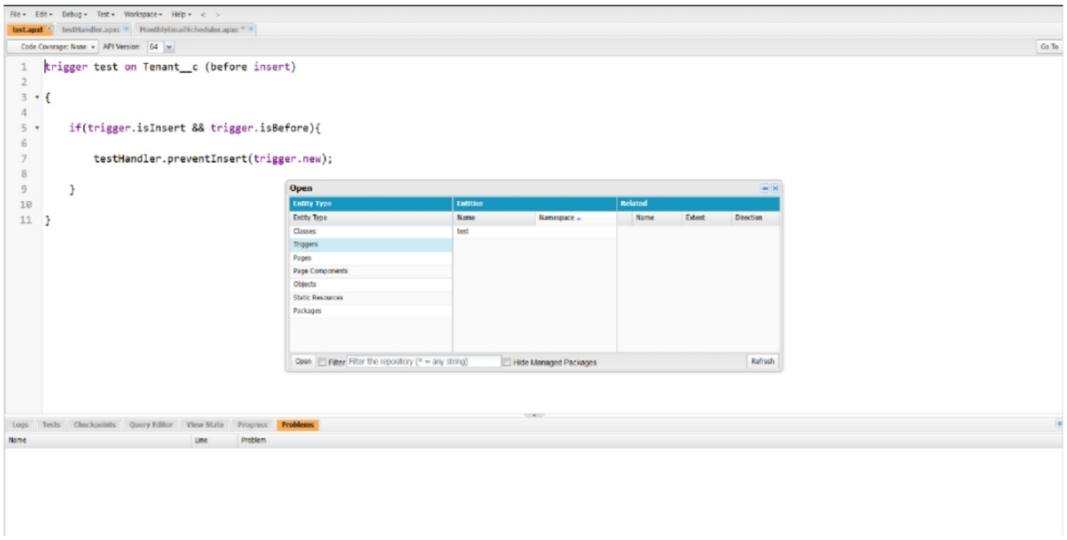
For Check for Vacant:

14

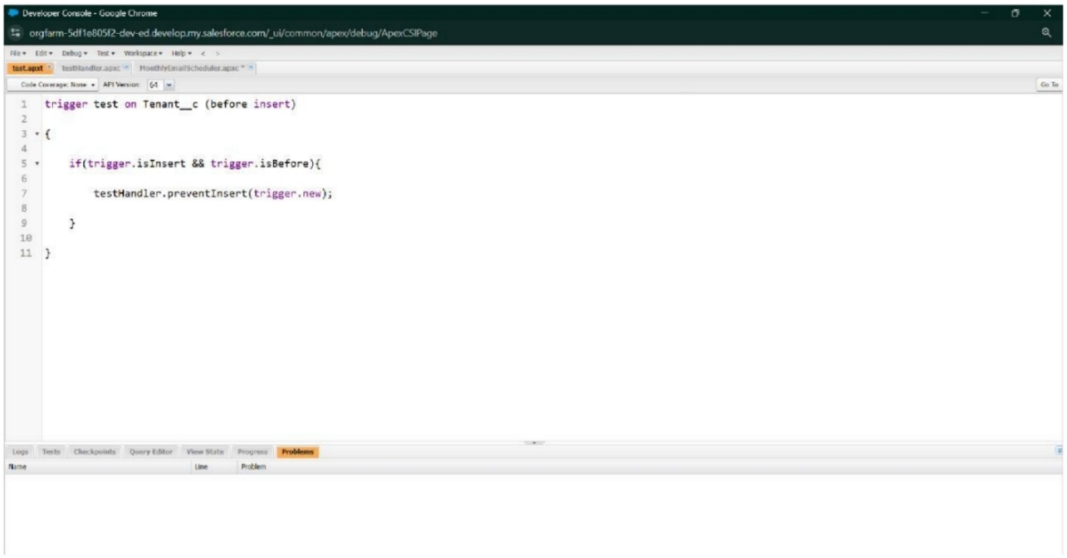


- Apex Trigger

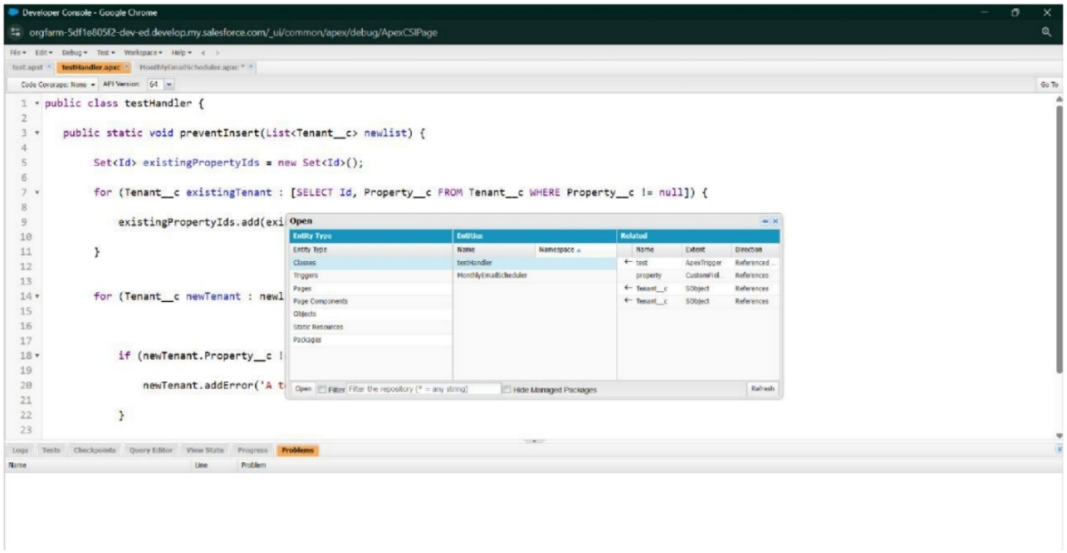
  Create an Apex Trigger



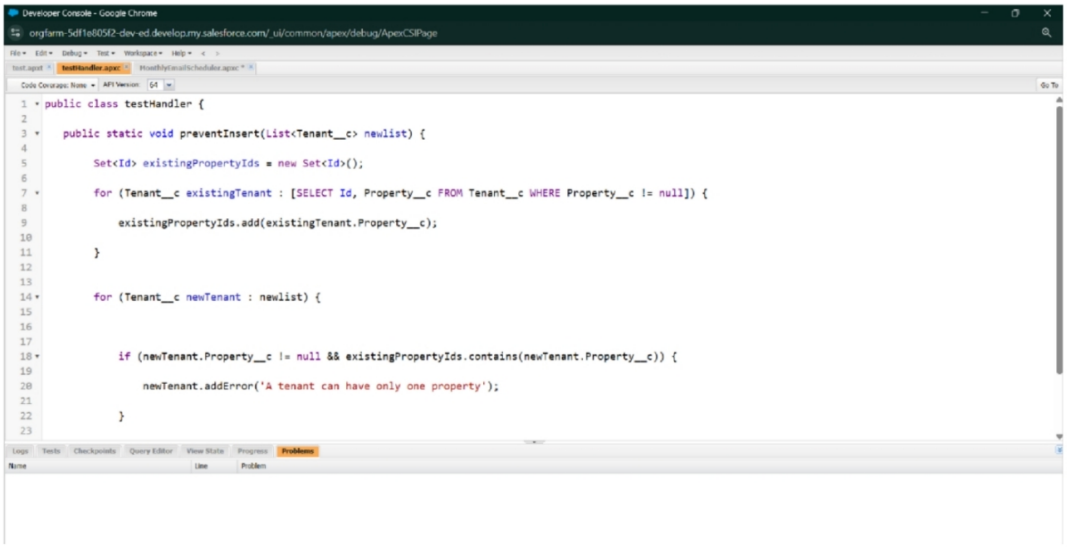
15



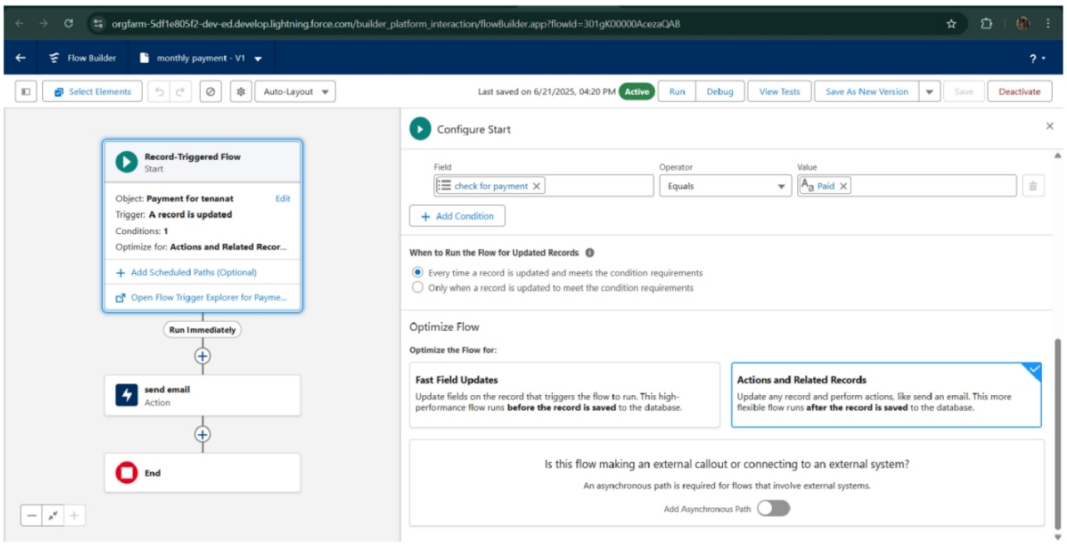
Create an Apex Handler class



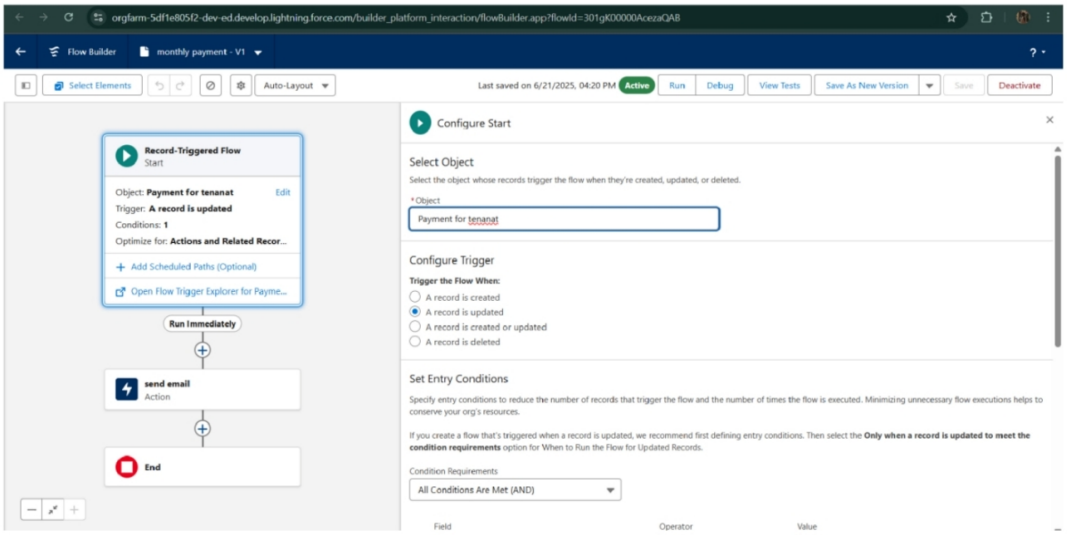
16


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');
}

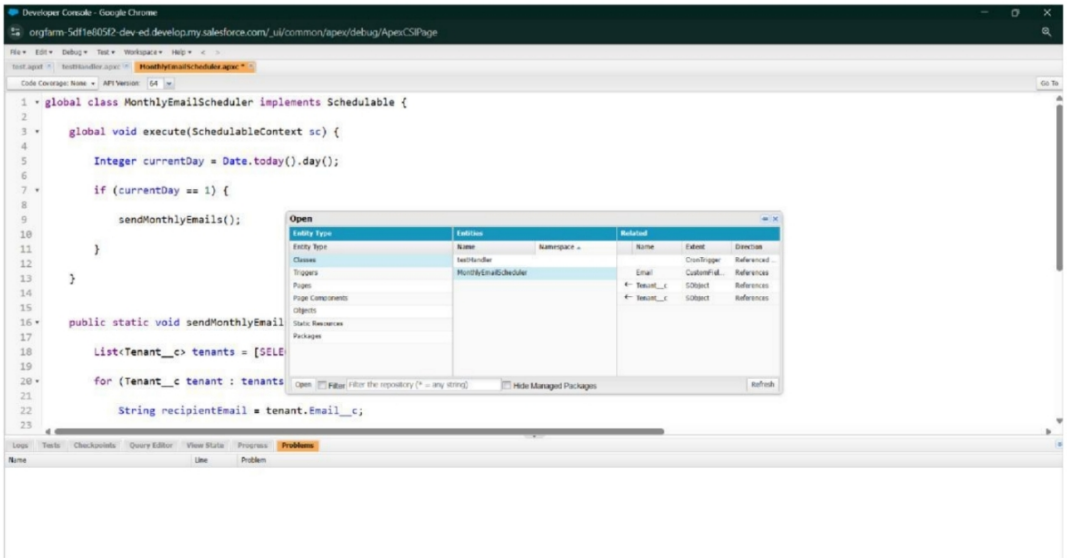
- FLOWS


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
check for payment
Operator
Equals
Value
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

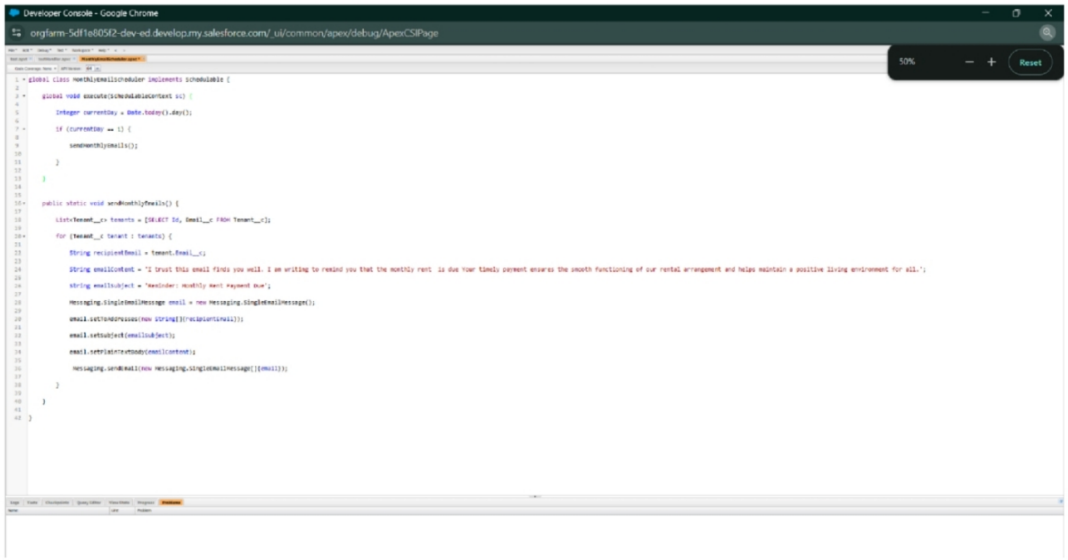
17



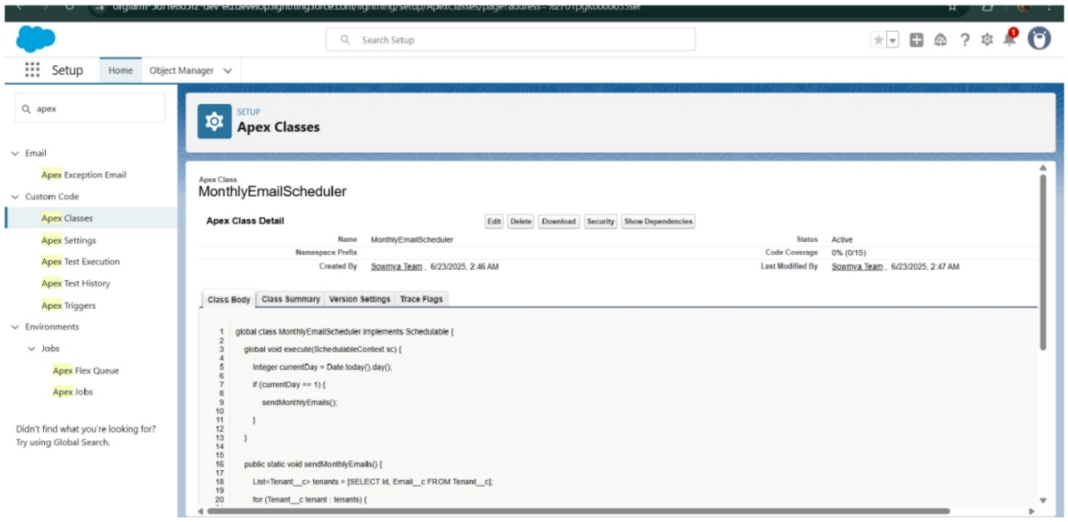
- Schedule class:
  Create an Apex Class



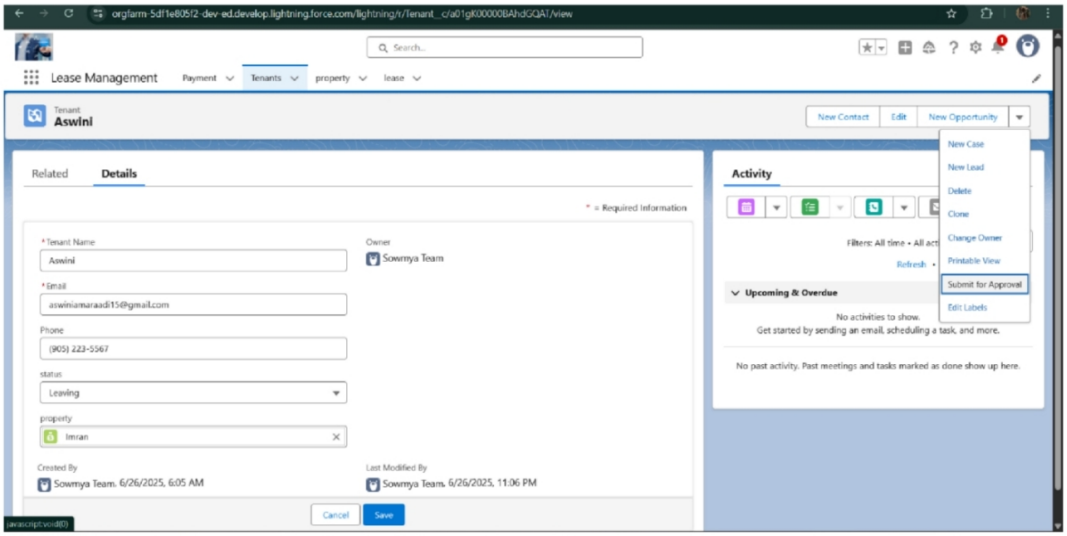
18

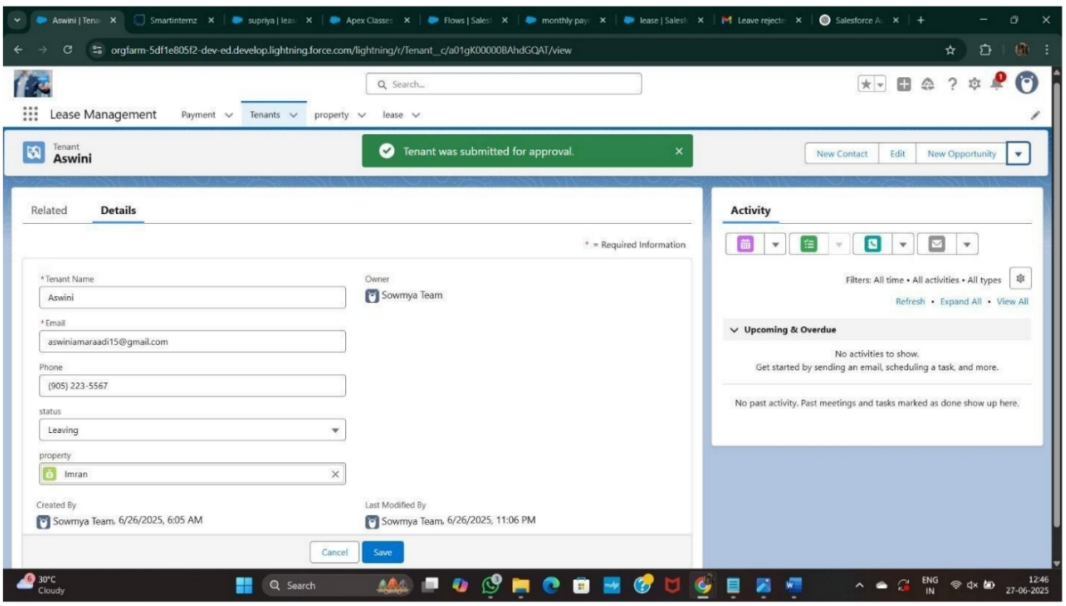


Schedule Apex class

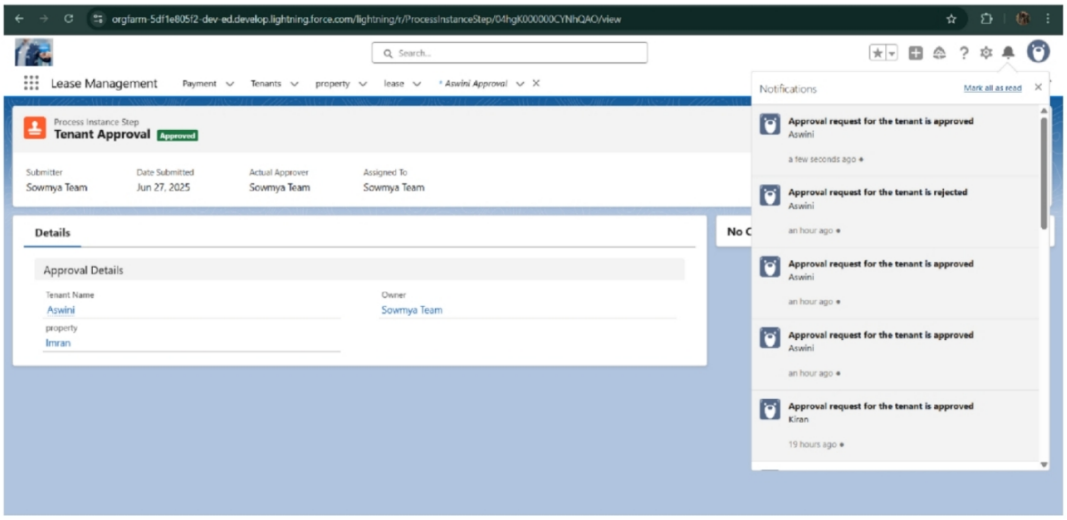


19


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
Activity
* = 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
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
Activity
* = 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
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.

20

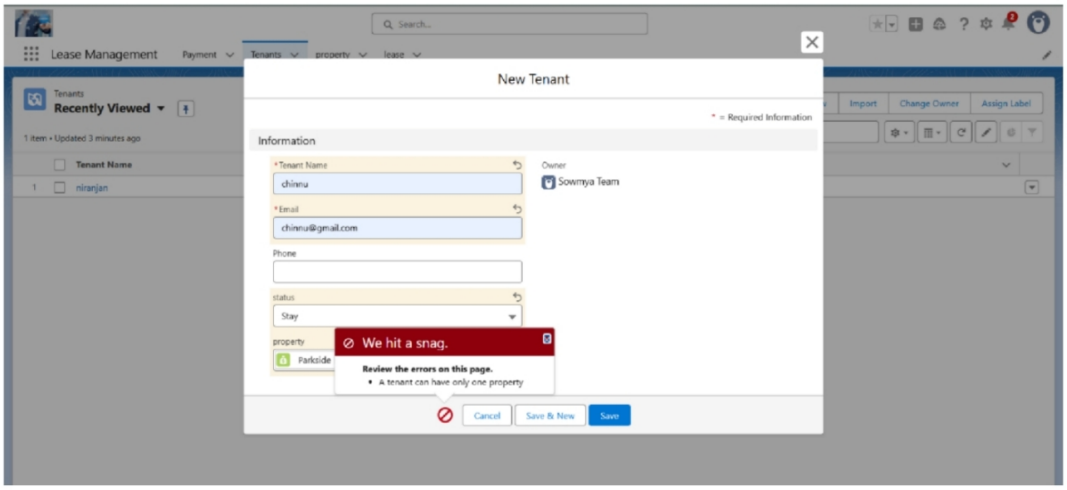


FUNCTIONAL AND PERFORMANCE TESTING

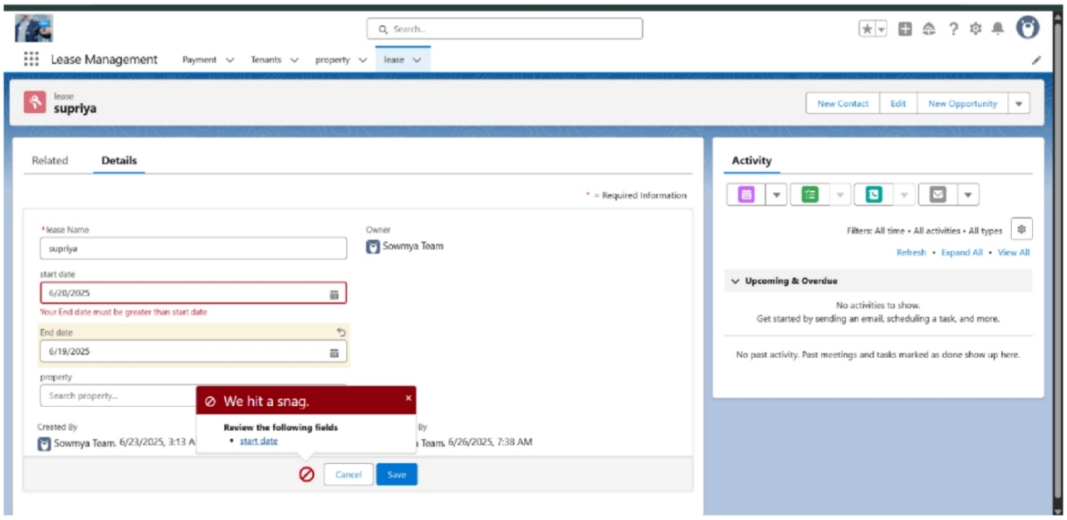
Performance Testing

- Trigger validation by entering duplicate tenant-property records

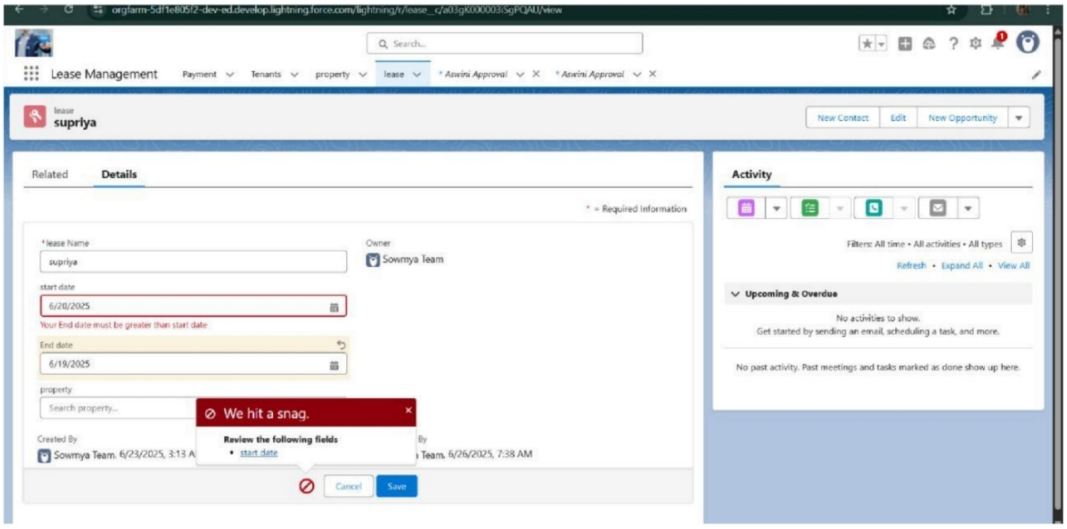
21



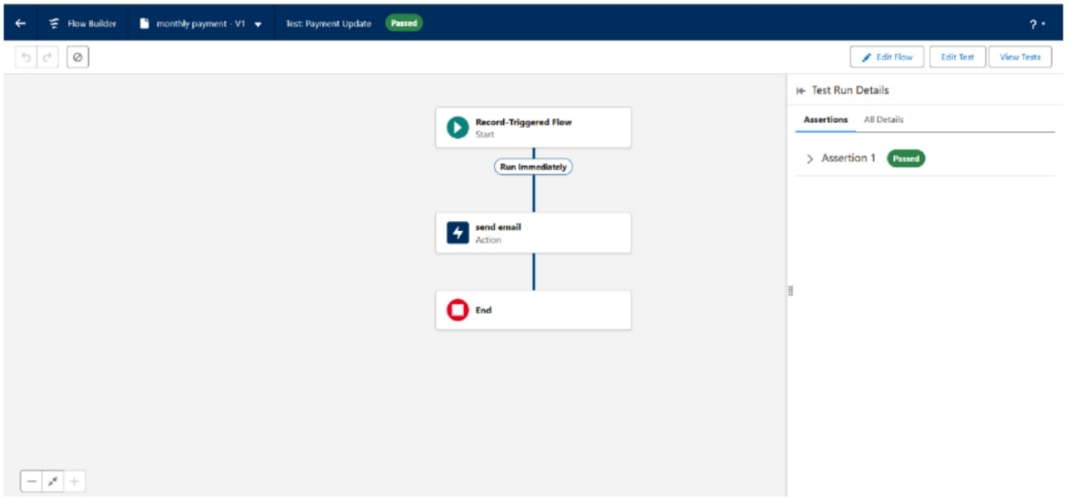
- Validation Rule checking



22

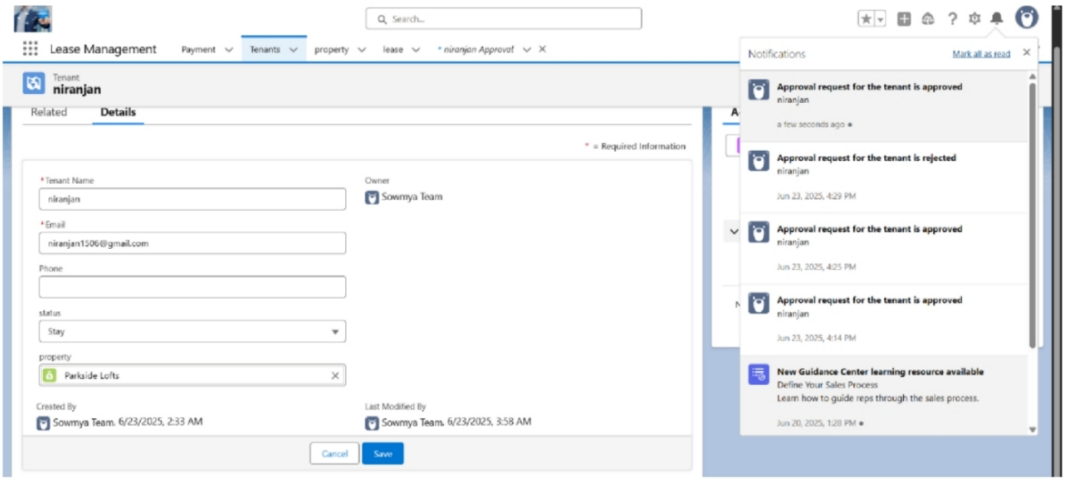


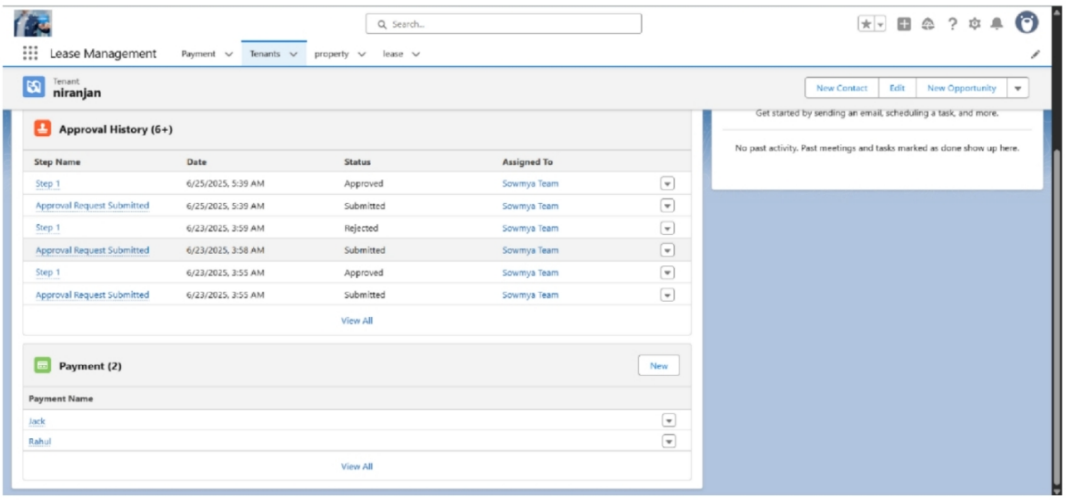
- Test flows on payment update



- Approval process validated through email alerts and status updates

23


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

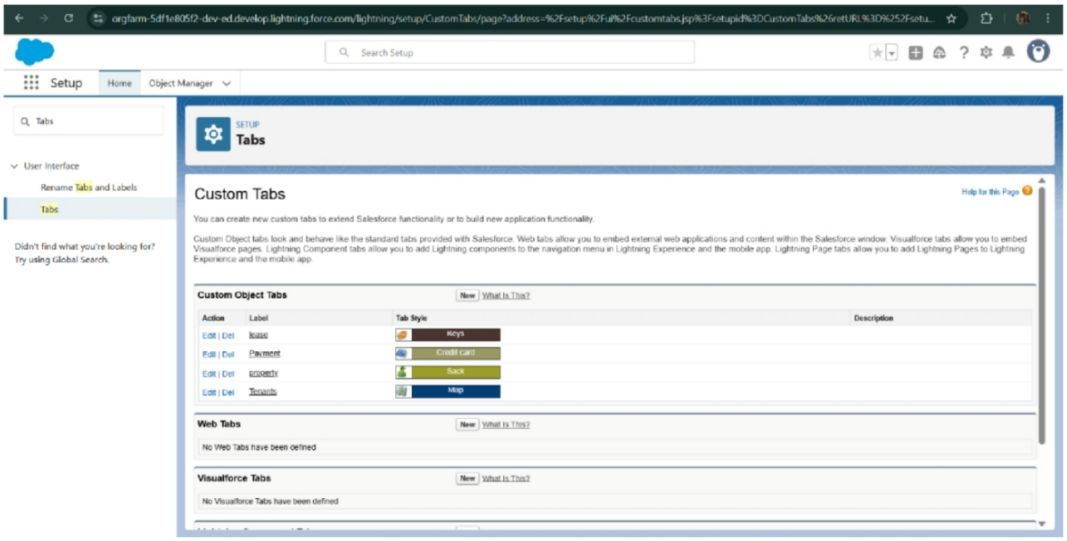

Search...
Lease Management
Payment
Tenants
property
lease
Tenant
niranjan
New Contact
Edit
New Opportunity
Get started by sending an email, scheduling a task, and more.
No past activity. Past meetings and tasks marked as done show up here.
Approval History (6+)
Step Name
Date
Status
Assigned To
Step 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

24

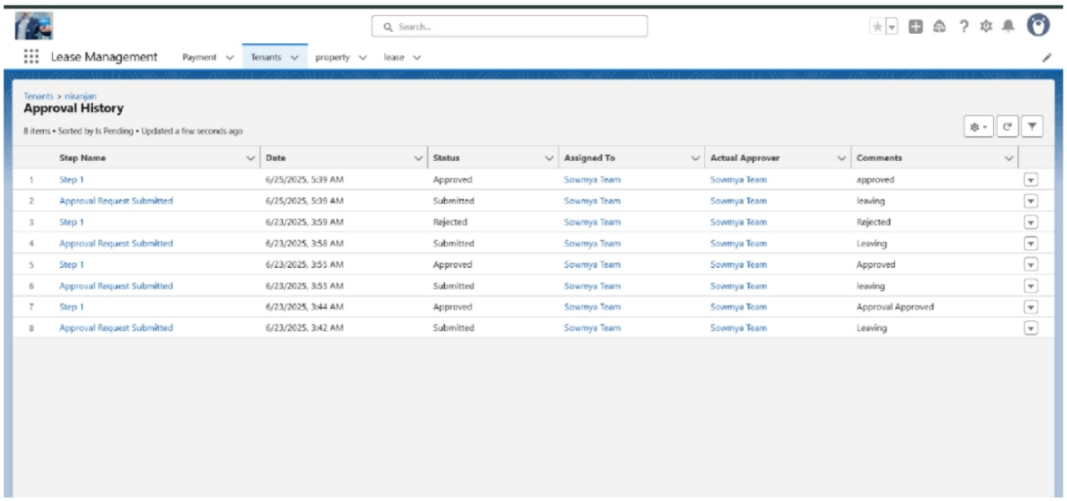
RESULTS

Output Screenshots

- Tabs for Property, Tenant, Lease, Payment

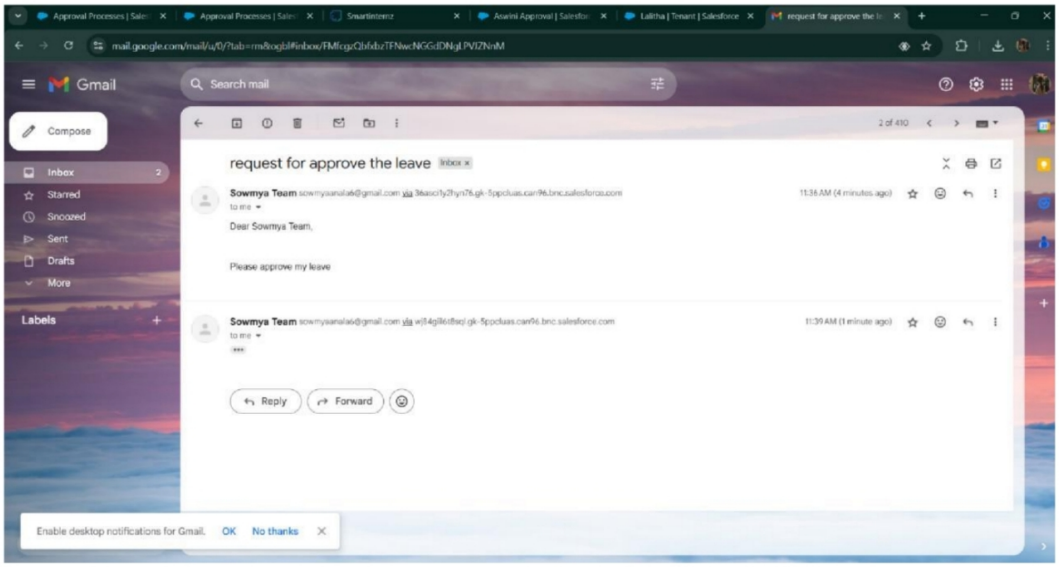


- Email alerts

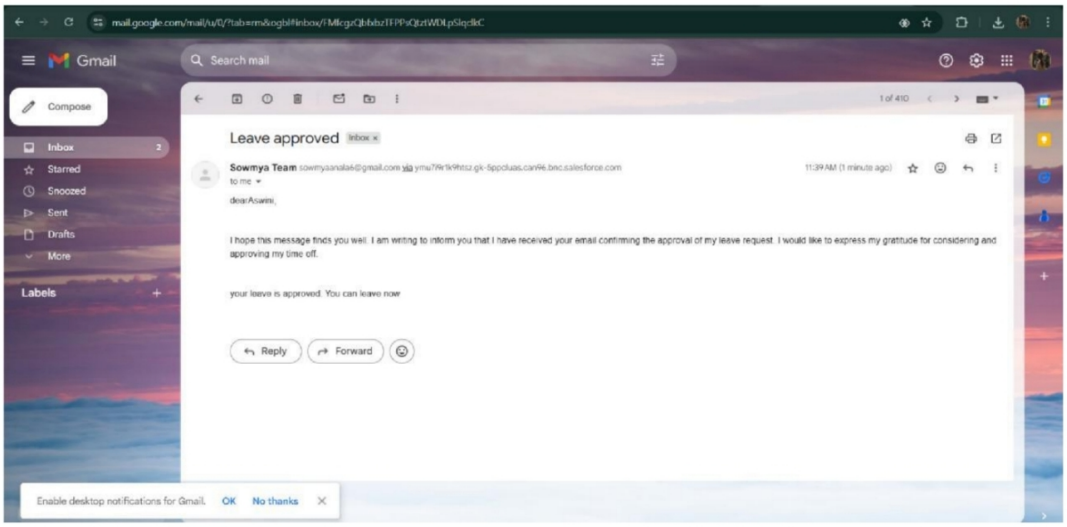


- Request for approve the leave

25

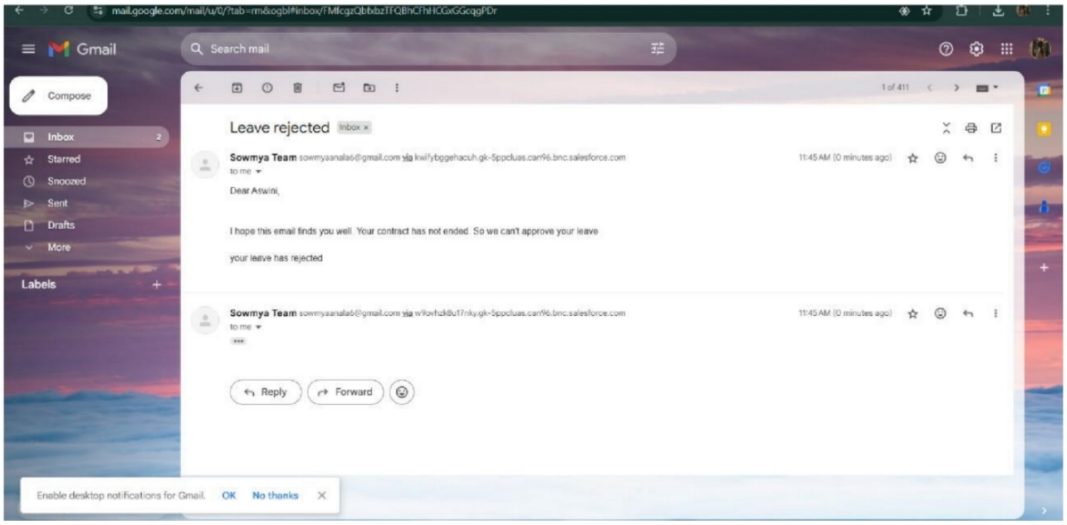


- Leave approved

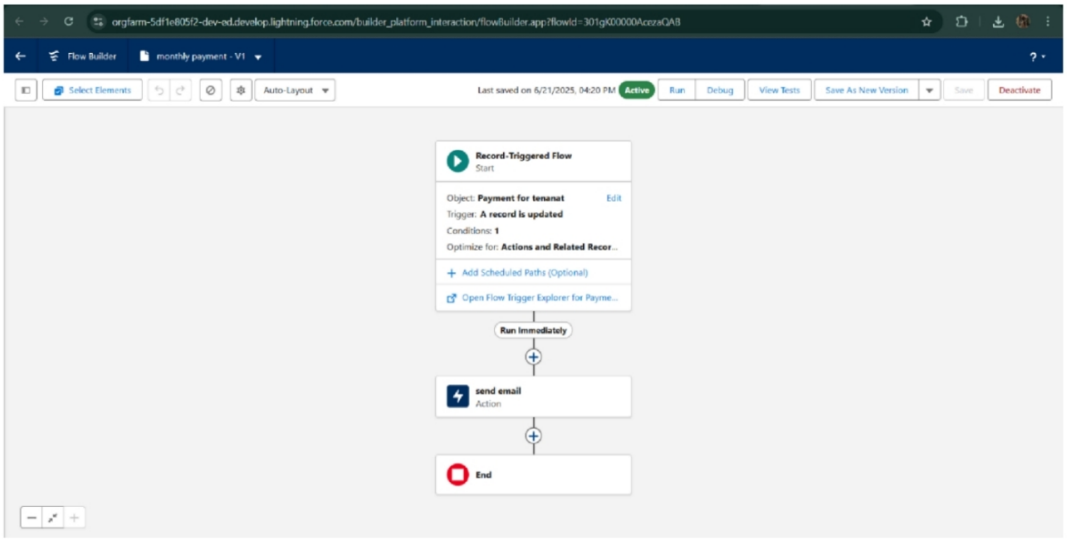


- Leave rejected

26

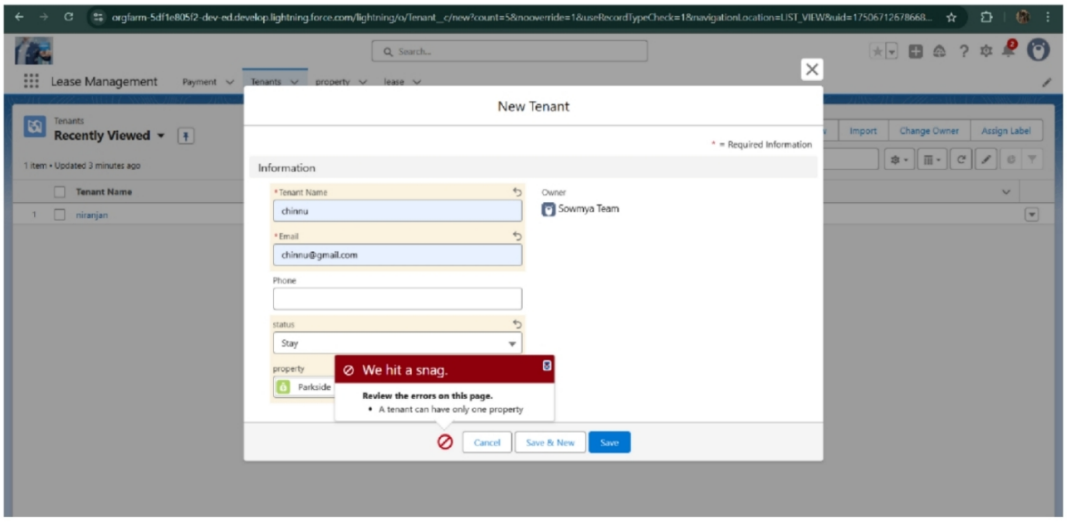


- Flow runs

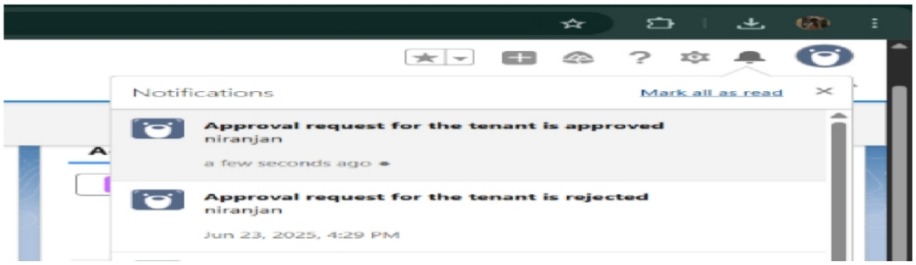


- Trigger error messages

27



- Approval process notifications



ADVANTAGES & DISADVANTAGES

.

28

CONCLUSION

The Lease Management System successfully streamlines the operations of leasing through a structured, automated Salesforce 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;
```


29

```
        } 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';
```


30

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


31