

# LEASE MANAGEMENT

College Name: Kathir College of Arts and Science

College Code: bruah

TEAM ID: NM2025TMID2441B

TEAM MEMBERS:5

Team Leader Name: Naveen Raj R

Email: dxvxdpaul@gmail.com

Team Member1: Deepak S

Email: deepakdeepud131@gmail.com

Team Member 2: Ganapathi B

Email: kingofganapathy989565@gmail.com

Team Member 3: Karthick Raja R

Email: karthickraja32006106@agentforce.com

Team Member 4: Sanjay V

Email: sanjayswathini156@gmail.com

# ABSTARCT

A Lease Management Project focuses on developing a system or application designed to efficiently manage the processes related to leasing real estate properties, equipment, or other valuable assets. The primary objective of the system is to streamline and automate various tasks associated with lease agreements, such as property listing, tenant registration, contract generation, payment tracking, and renewal management. By integrating these features into a single platform, the system minimizes manual work, reduces the chances of human error, and ensures smooth handling of leasing operations.

Furthermore, the project emphasizes accurate record-keeping, regulatory compliance, and effective communication between all parties involved, including landlords, tenants, and administrators. By adopting this digital solution, users can monitor payments, track lease expirations, generate invoices, and receive automated notifications, improving transparency and operational efficiency. Such a system not only simplifies the overall leasing process but also enhances decision-making for property managers and owners by providing real-time insights and reports. Ultimately, this project aims to deliver a reliable, user-friendly, and scalable platform that improves the management of lease agreements and supports efficient property and asset management.

As a team We've learned,

- ☐ Salesforce
- ☐ Object
- ☐ Tabs
- ☐ The Lightning App
- ☐ Fields
- ☐ Validation Rule
- ☐ Email Templates
- ☐ Approval Process
- ☐ Apex Trigger
- ☐ FLOWS
- ☐ Schedule Class

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



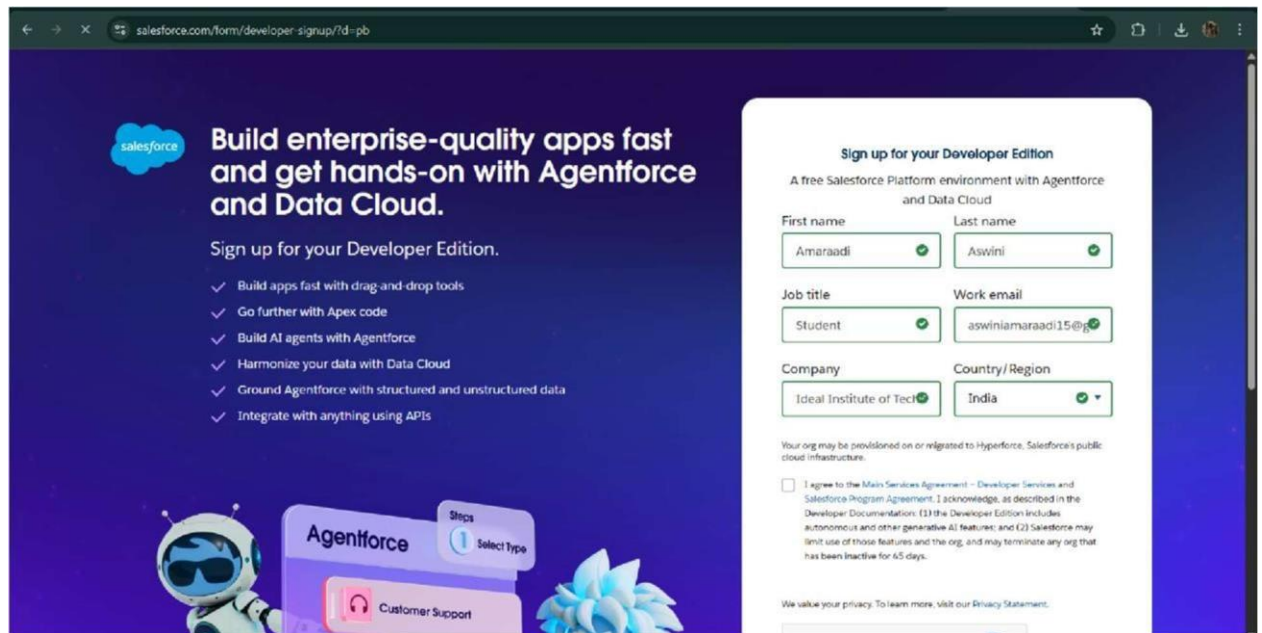
## 1.2 Purpose

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

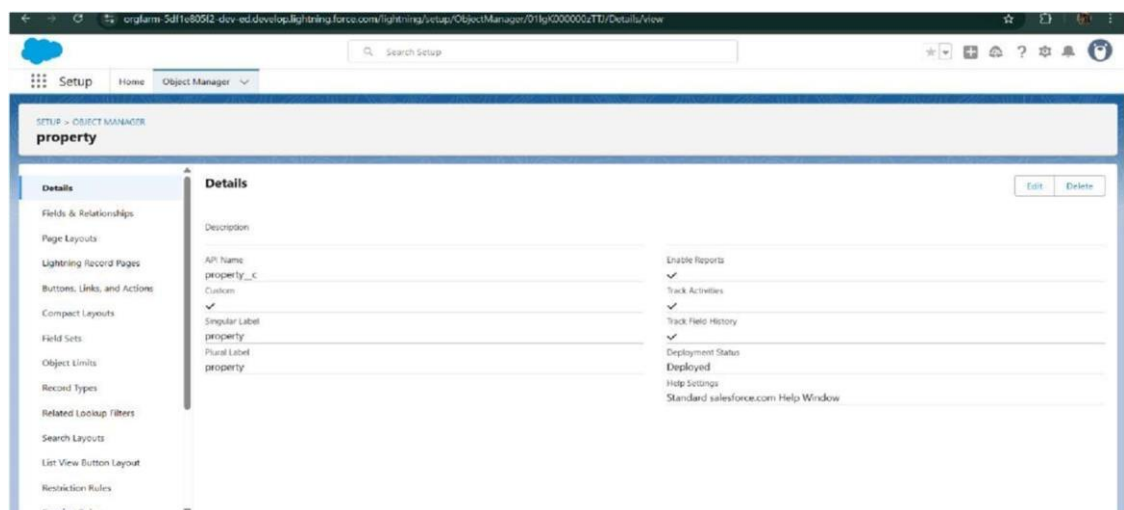
Creating Developer Account:

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



The screenshot shows the Salesforce Developer Edition sign-up page. The left side features a blue background with the Salesforce logo and text: "Build enterprise-quality apps fast and get hands-on with Agentforce and Data Cloud. Sign up for your Developer Edition." Below this are five bullet points: "Build apps fast with drag-and-drop tools", "Go further with Apex code", "Build AI agents with Agentforce", "Harmonize your data with Data Cloud", "Ground Agentforce with structured and unstructured data", and "Integrate with anything using APIs". The right side is a white form titled "Sign up for your Developer Edition" with the subtitle "A free Salesforce Platform environment with Agentforce and Data Cloud". The form contains fields for First name (Amaradi), Last name (Aswini), Job title (Student), Work email (aswiniamaraadi15@gmail.com), Company (Ideal Institute of Tech), and Country/Region (India). A checkbox for the Main Services Agreement is present, and a link to the Privacy Statement is at the bottom.

- Created objects: Property, Tenant, Lease, Payment



The screenshot shows the Salesforce Setup page for the 'property' object. The left sidebar contains a navigation menu with options: 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, and Roles. The main content area is titled 'property' and includes a 'Details' section with fields for Description, API Name (property\_\_c), Custom (checked), Singular Label (property), and Plural Label (properties). On the right, there are checkboxes for 'Enable Reports' (checked), 'Track Activities' (checked), 'Track Field History' (checked), and 'Deployment Status' (Deployed). A link to the 'Standard Salesforce.com Help Window' is also present.

orgfarm-Sdf1e805f2-dev-ed.develop.lightning.force.com/lightning/setup/ObjectManager/01lgK000000z7bN/Details/view

Setup Home Object Manager

SETUP > OBJECT MANAGER

### Tenant

Details Edit Delete

Description	
API Name	Tenant_c
Custom	<input checked="" type="checkbox"/>
Singular Label	Tenant
Plural Label	Tenants
Enable Reports	<input checked="" type="checkbox"/>
Track Activities	<input checked="" type="checkbox"/>
Track Field History	<input checked="" type="checkbox"/>
Deployment Status	Deployed
Help Settings	Standard salesforce.com Help Window

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

orgfarm-Sdf1e805f2-dev-ed.develop.lightning.force.com/lightning/setup/ObjectManager/01lgK000000z7uj/Details/view

Setup Home Object Manager

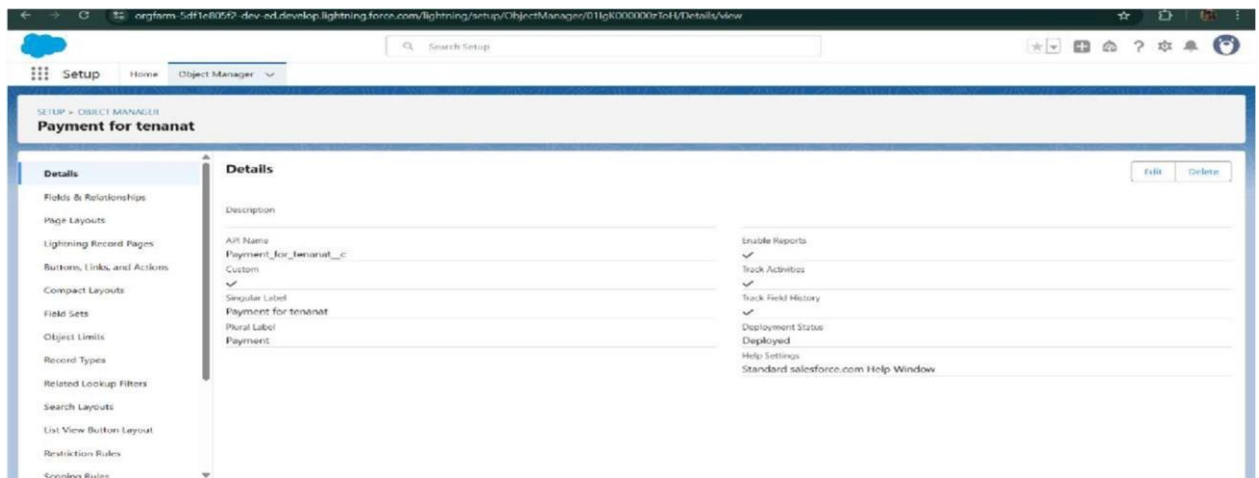
SETUP > OBJECT MANAGER

### lease

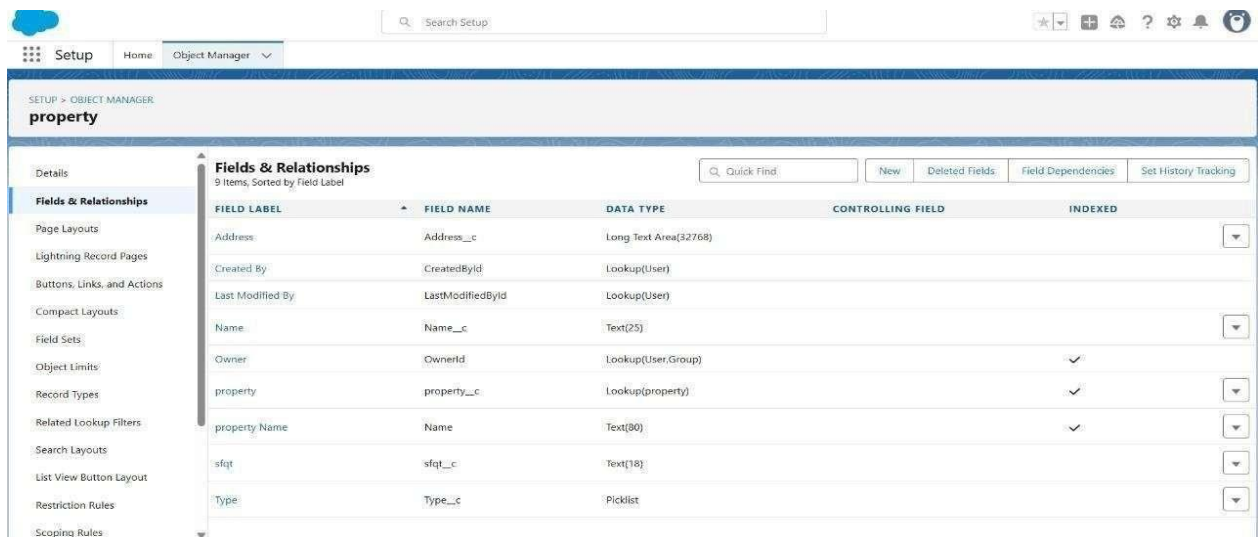
Details Edit Delete

Description	
API Name	lease_c
Custom	<input checked="" type="checkbox"/>
Singular Label	lease
Plural Label	leases
Enable Reports	<input checked="" type="checkbox"/>
Track Activities	<input checked="" type="checkbox"/>
Track Field History	<input checked="" type="checkbox"/>
Deployment Status	Deployed
Help Settings	Standard salesforce.com Help Window

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



- Configured fields and relationships



SetupHomeObject Manager

Search Setup

Setup > OBJECT MANAGER

Payment for tenanat

Details

Fields & Relationships

7 Items, Sorted by Field Label

Quick FindNewDeleted FieldsField DependenciesSet 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)		

SetupHomeObject Manager

Search Setup

Setup > OBJECT MANAGER

lease

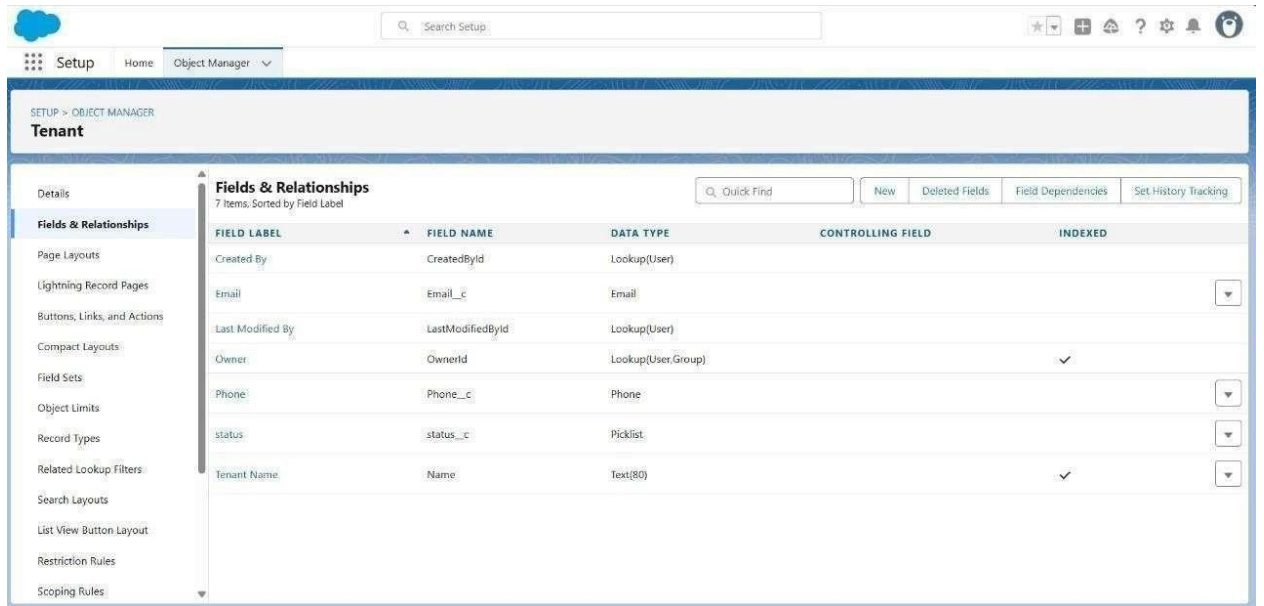
Details

Fields & Relationships

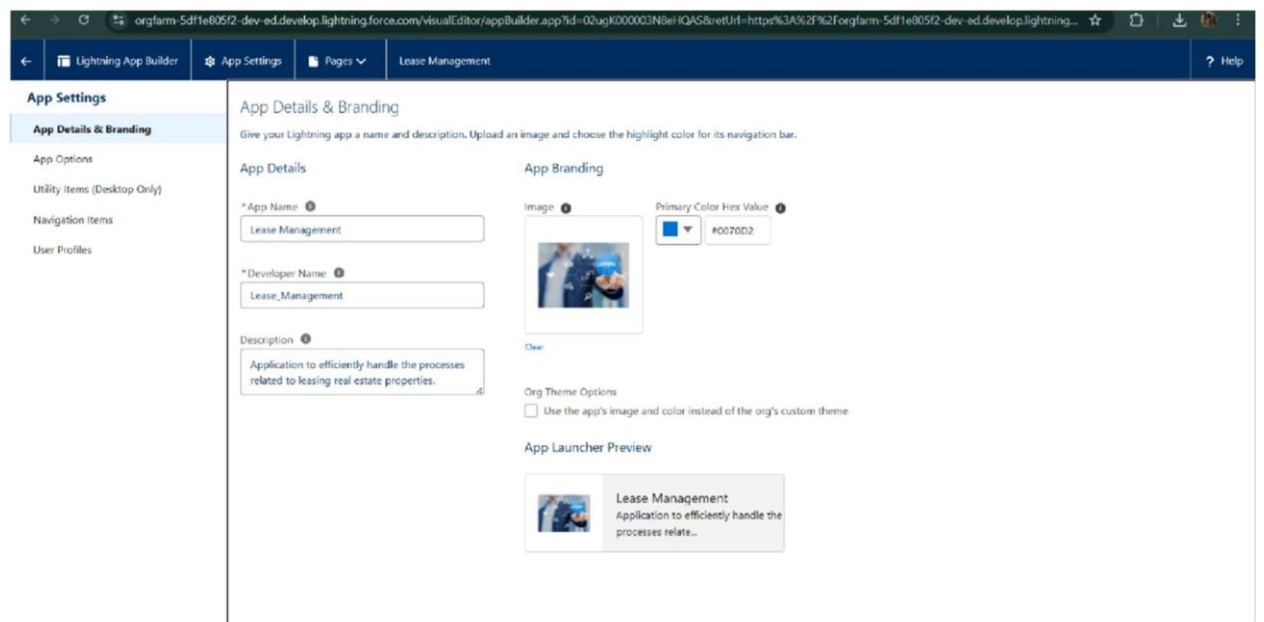
7 Items, Sorted by Field Label

Quick FindNewDeleted FieldsField DependenciesSet 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		



- Developed Lightning App with relevant tabs





←

→

↺

orgfarm-5df1e805f2-dev-ed.develop.lightning.force.com/visualEditor/appBuilder.app?Id=02ugK000003N8eHQAS&retUrl=https%3A%2F%2Forgfarm-5df1e805f2-dev-ed.develop.lightning...

☆

📄

📄

⋮

← 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

🔍 Type to filter list...

Accounts

Activation Targets

Activations

All Sites

Alternative Payment Methods

Analytics

App Launcher

Appointment Categories

Appointment Invitations

Approval Requests

Create ▾

➡

⬅

Selected Items

Payment

Tenants

property

lease

⬆

⬇

←

→

↺

orgfarm-5df1e805f2-dev-ed.develop.lightning.force.com/visualEditor/appBuilder.app?Id=02ugK000003N8eHQAS&retUrl=https%3A%2F%2Forgfarm-5df1e805f2-dev-ed.develop.lightning...

☆

📄

📄

⋮

← 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

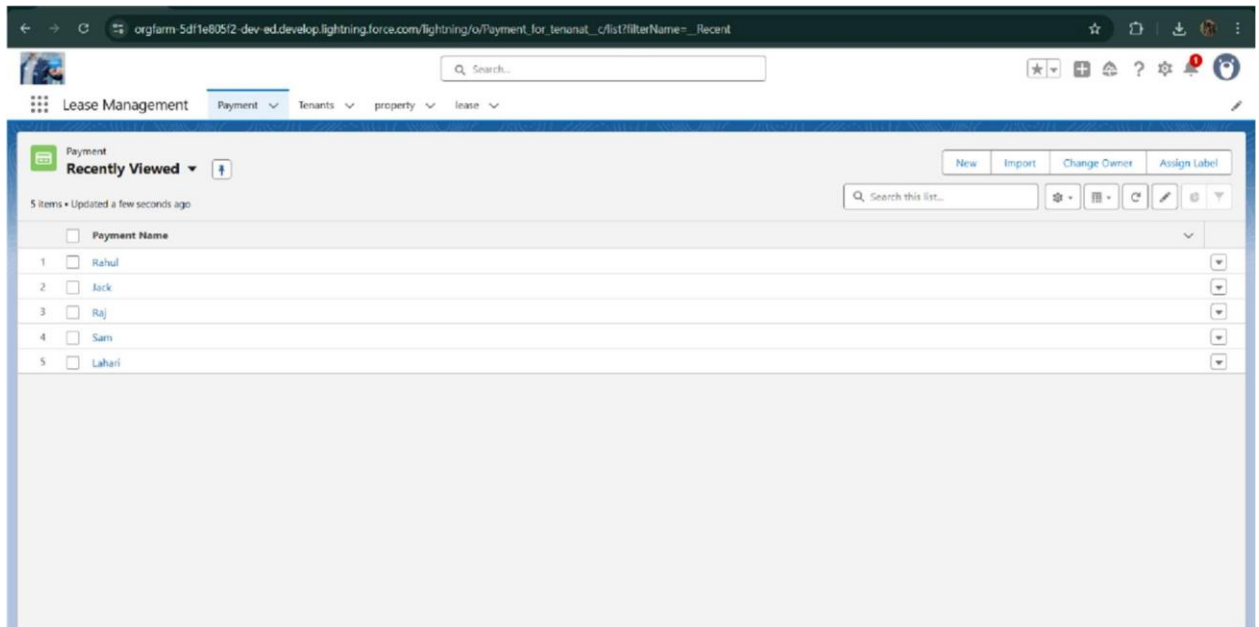
Customer Community Login User

➡

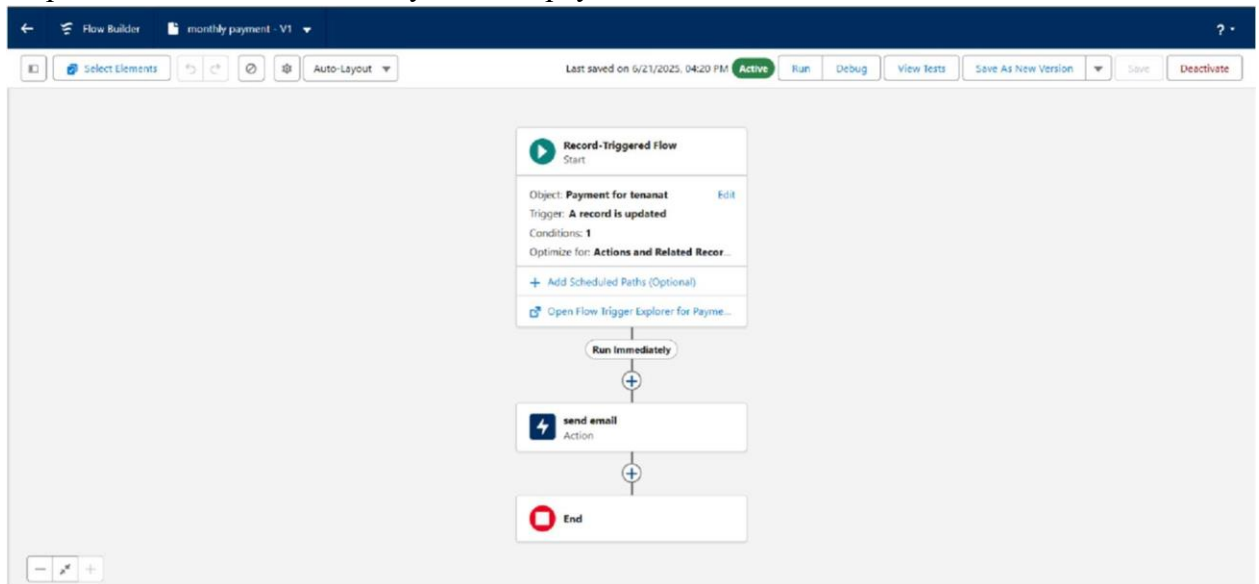
⬅

Selected Profiles

System Administrator



- Implemented Flows for monthly rent and payment success



- To create a validation rule to a Lease Object

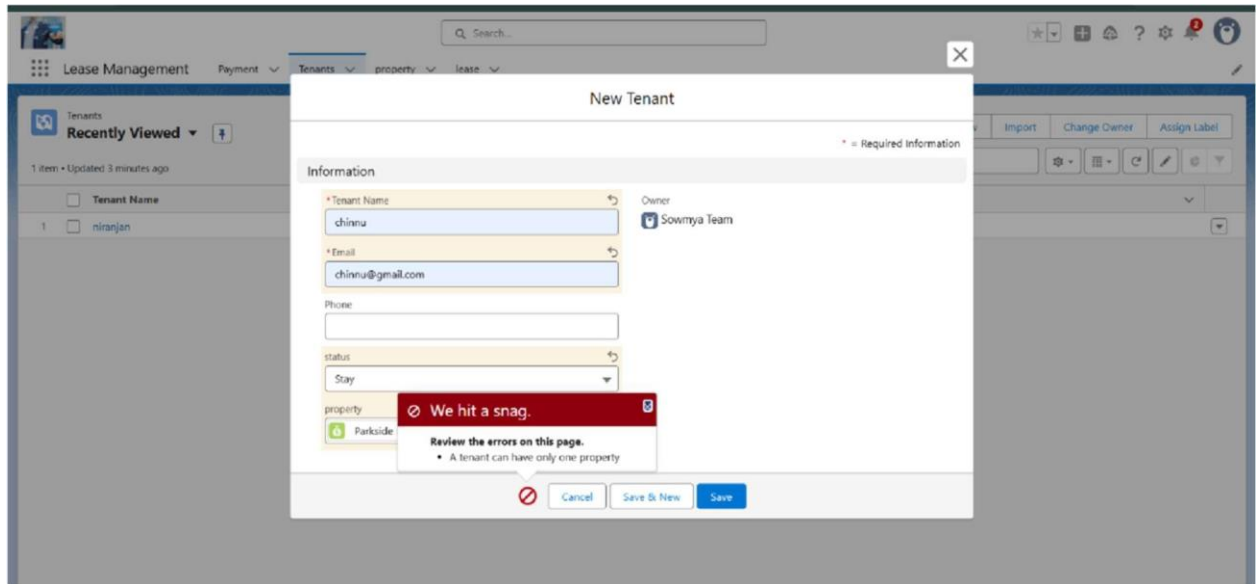
The screenshot shows the 'Validation Rule Edit' page in Salesforce Setup. The left sidebar contains a navigation menu with options like Details, Fields & Relationships, Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, Search Layouts, List View Button Layout, Restriction Rules, and Scoping Rules. The main content area is titled 'Validation Rule Edit' and includes a 'Rule Name' field with the value 'lease\_end\_date', an 'Active' checkbox, and a 'Description' field. Below these is the 'Error Condition Formula' section, which contains an example formula: `End_date__c <= start_date__c`. A 'Functions' dropdown menu is open, showing a list of functions including ABS, ACOS, ADDMONTHS, AND, ASCII, and ASIN. The 'Error Message' field is visible at the bottom.

The screenshot shows the 'lease Validation Rule' page in Salesforce Setup. The left sidebar is the same as the previous screenshot. The main content area is titled 'lease Validation Rule' and includes a 'Back to lease' link. Below this is the 'Validation Rule Detail' section, which displays the following information:

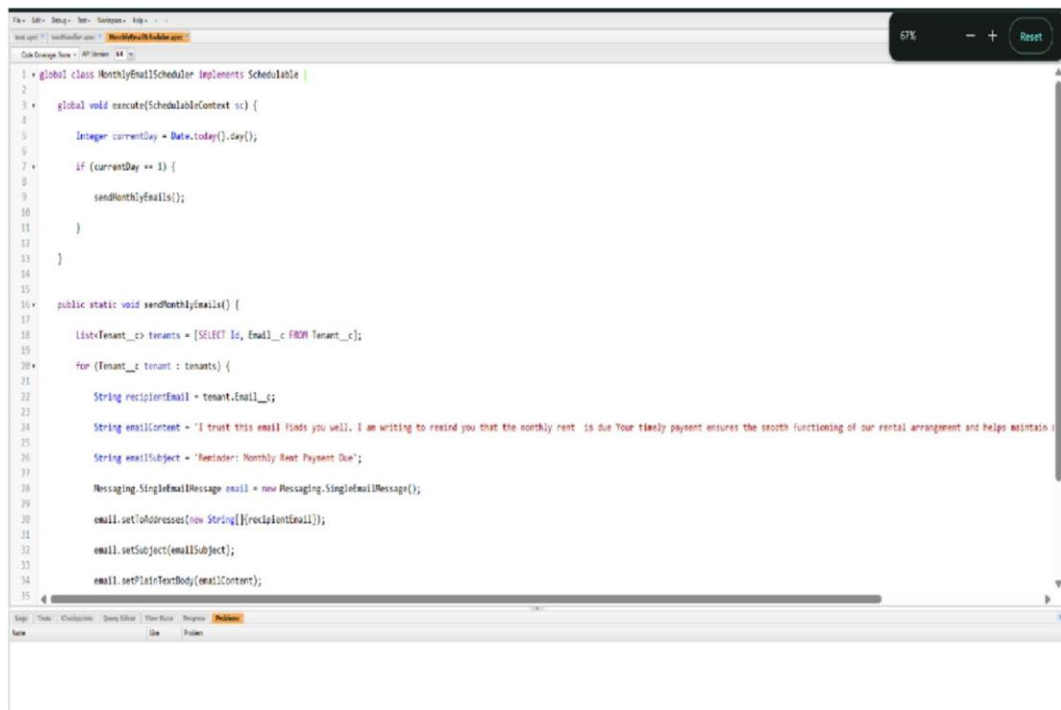
Validation Rule Detail		Edit		Close	
Rule Name	lease_end_date	Active	<input checked="" type="checkbox"/>		
Error Condition Formula	End_date__c <= start_date__c	Error Location	start date		
Error Message	Your End date must be greater than start date				
Description					
Created By	Soumya Team 6/19/2025, 5:37 AM	Modified By	Soumya Team 6/26/2025, 7:47 AM		

At the bottom of the 'Validation Rule Detail' section, there are 'Edit' and 'Close' buttons.

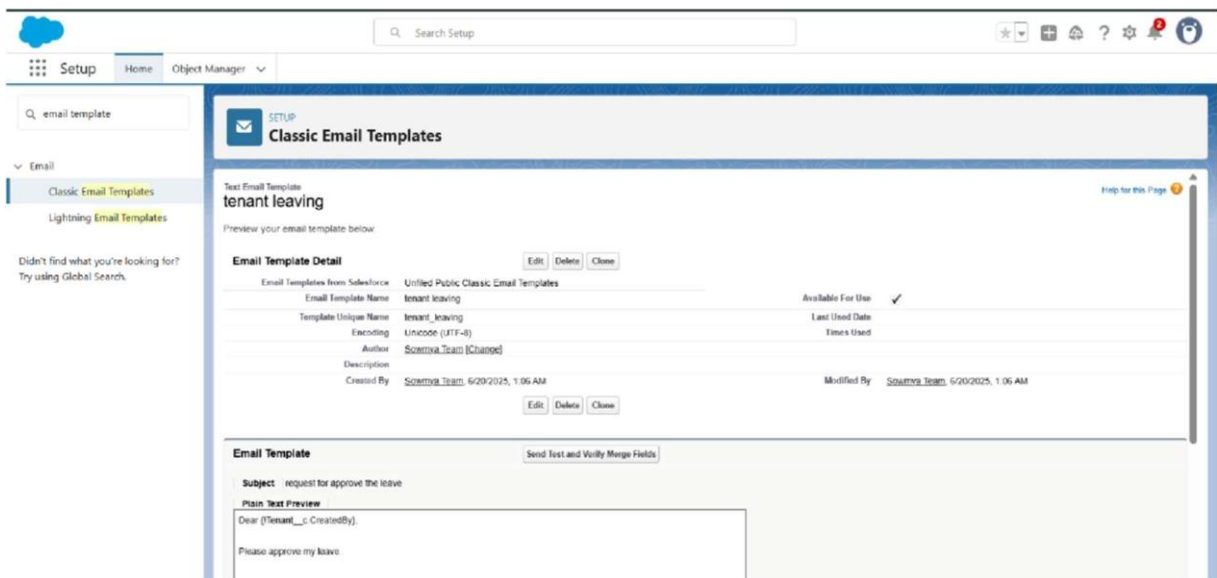
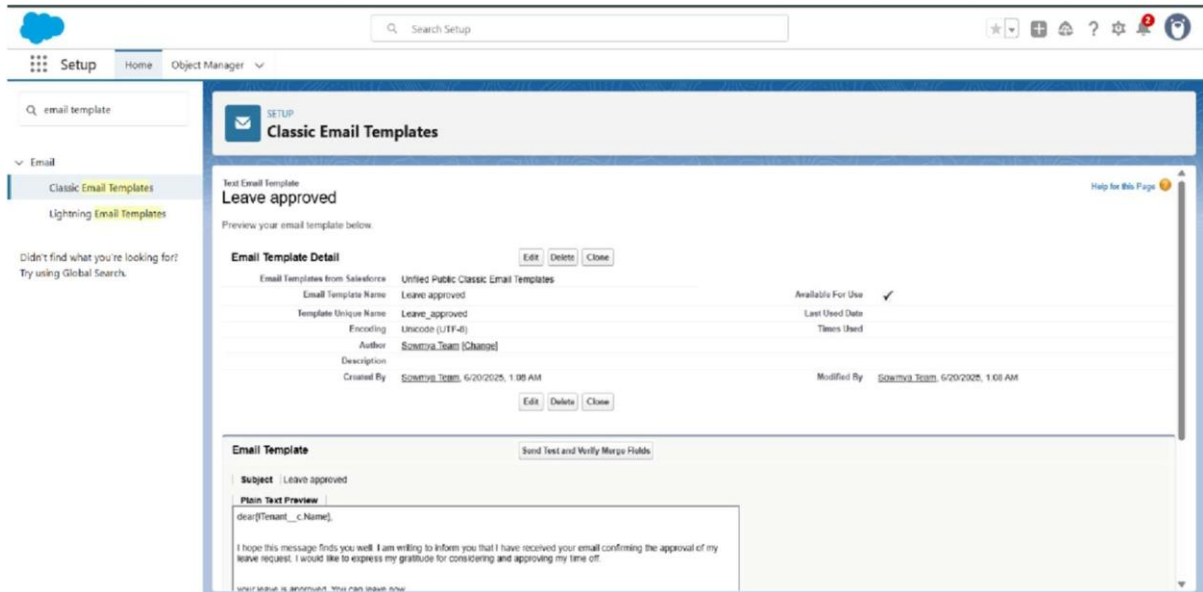
- Added Apex trigger to restrict multiple tenants per property



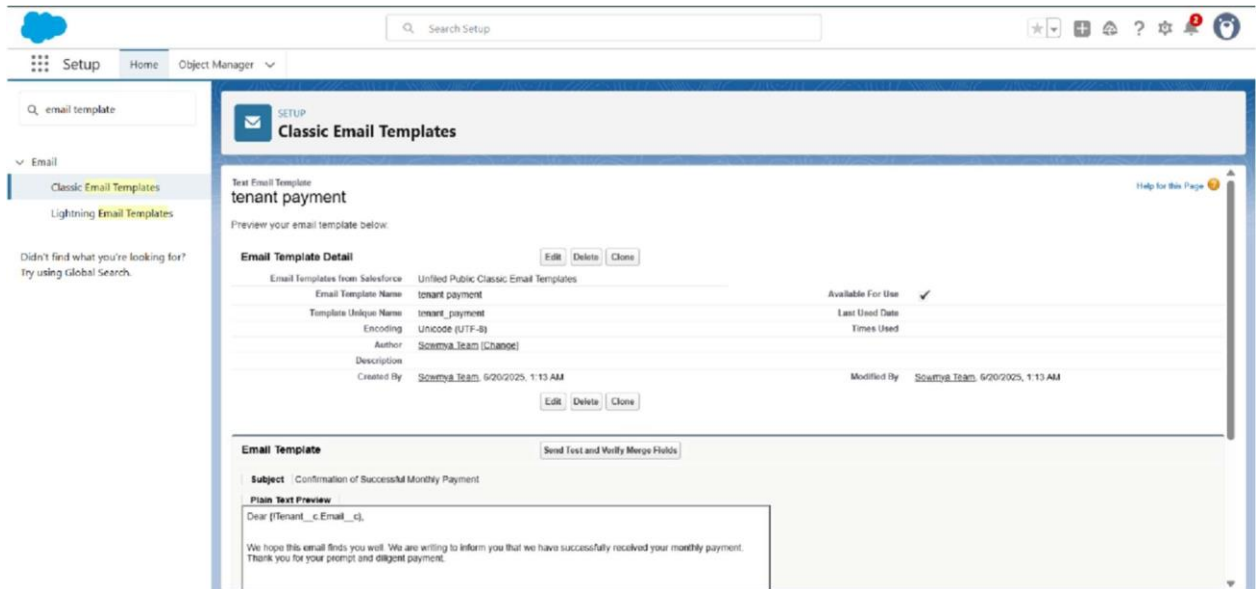
- Scheduled monthly reminder emails using Apex class



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

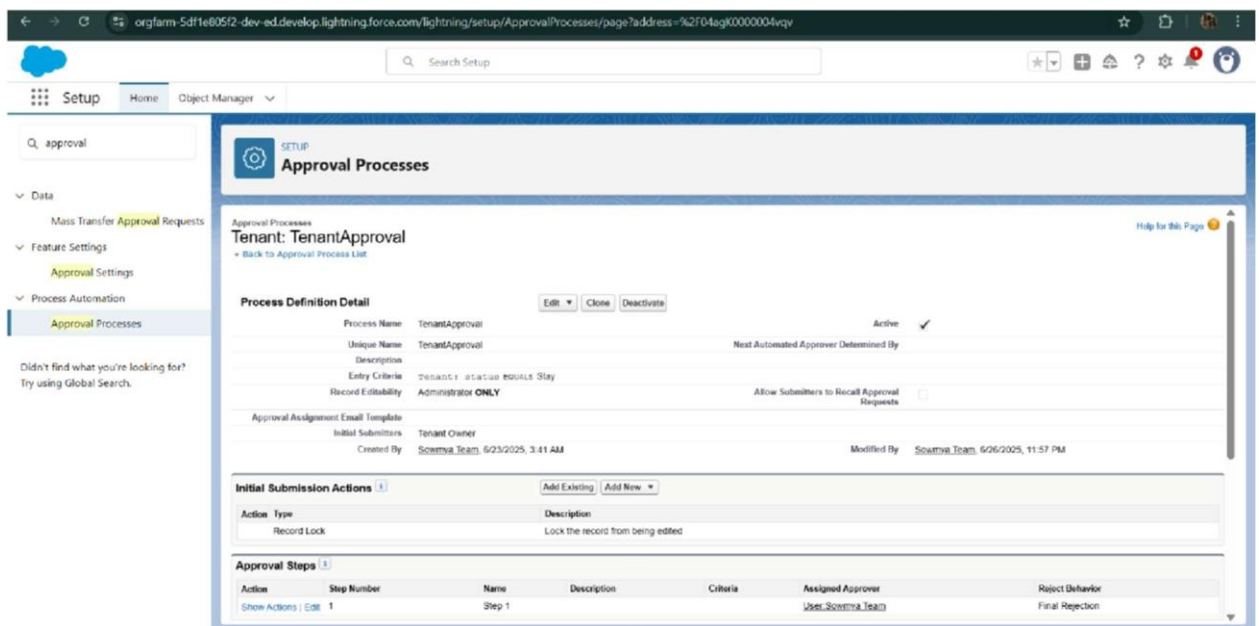






- Approval Process creation

For Tenant Leaving:



For Check for Vacant:

The screenshot shows the Salesforce Setup page for Approval Processes. The left sidebar contains a search bar and a navigation menu with options like Data, Feature Settings, Approval Settings, and Process Automation. The main content area displays the configuration for the 'Tenant: check for vacant' process. The 'Process Definition Detail' section includes fields for Process Name, Unique Name, Description, Entry Criteria, Record Eligibility, Approval Assignment Email Template, Initial Submitters, and Created By. The 'Initial Submission Actions' section shows a table with actions like Record Lock and Email Alert. The 'Approval Steps' section shows a table with steps, including step1 with a description 'Lock the record from being edited' and a reject behavior of 'Final Rejection'.

Action	Type	Description
Record Lock	Record Lock	Lock the record from being edited
Email Alert	Email Alert	DREASE_APPROVAL_REQUEST

Action	Step Number	Name	Description	Criteria	Assigned Approver	Reject Behavior
1	1	step1	Lock the record from being edited		User:Soumya.Team	Final Rejection

- Apex Trigger

Create an Apex Trigger

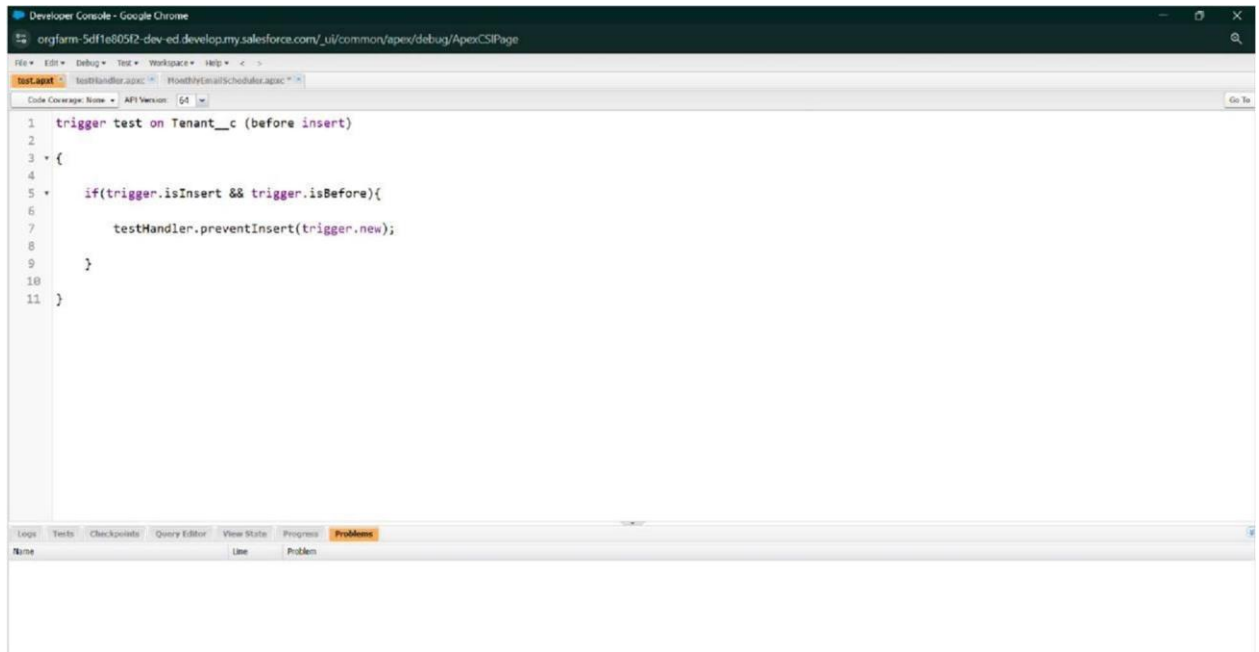
The screenshot shows the Salesforce IDE with an Apex trigger test. The code is as follows:

```
1 trigger test on Tenant__c (before insert)
2
3 {
4
5     if(trigger.isInsert && trigger.isBefore){
6
7         testHandler.preventInsert(trigger.new);
8
9     }
10
11 }
```

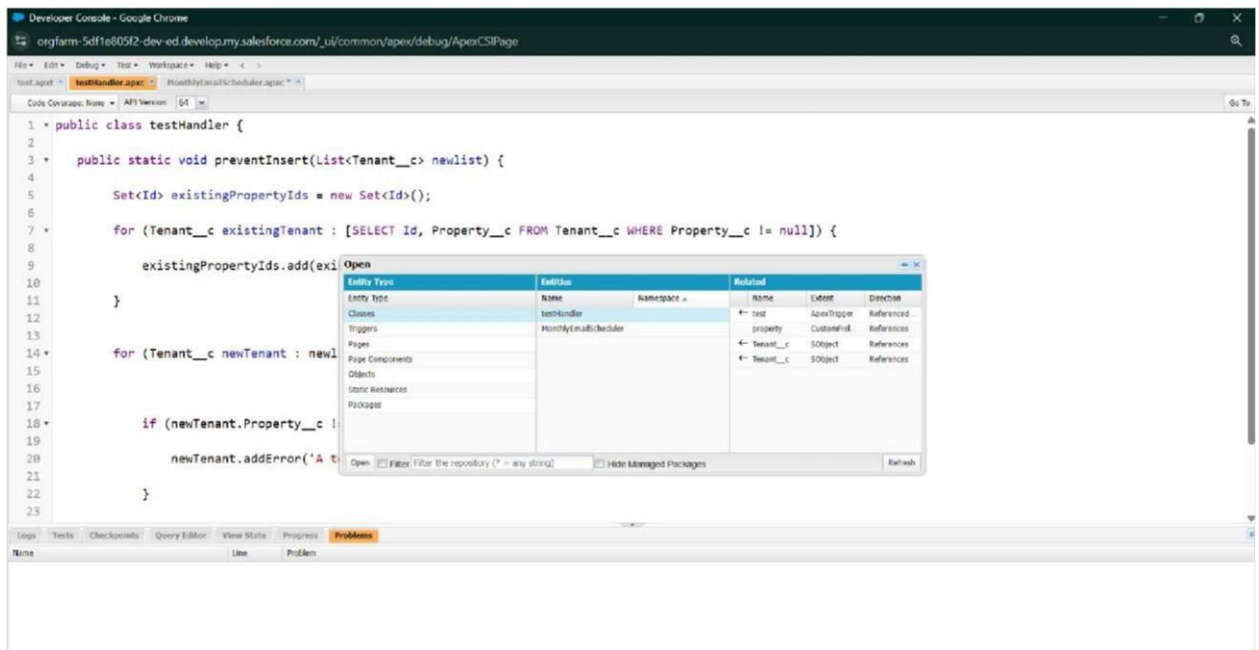
A search window is open, showing a table with columns: Entity Type, Name, Namespace, Name, Extent, and Direction. The table contains one row with the name 'test'.

Entity Type	Name	Namespace	Name	Extent	Direction
test					





Create an Apex Handler class

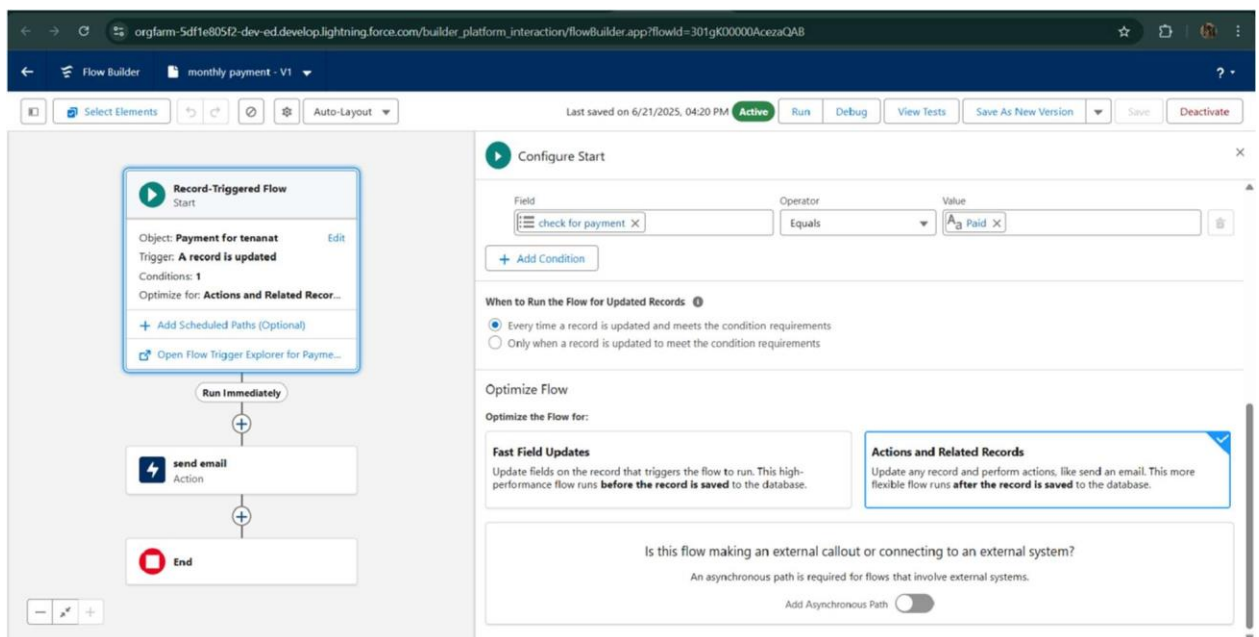


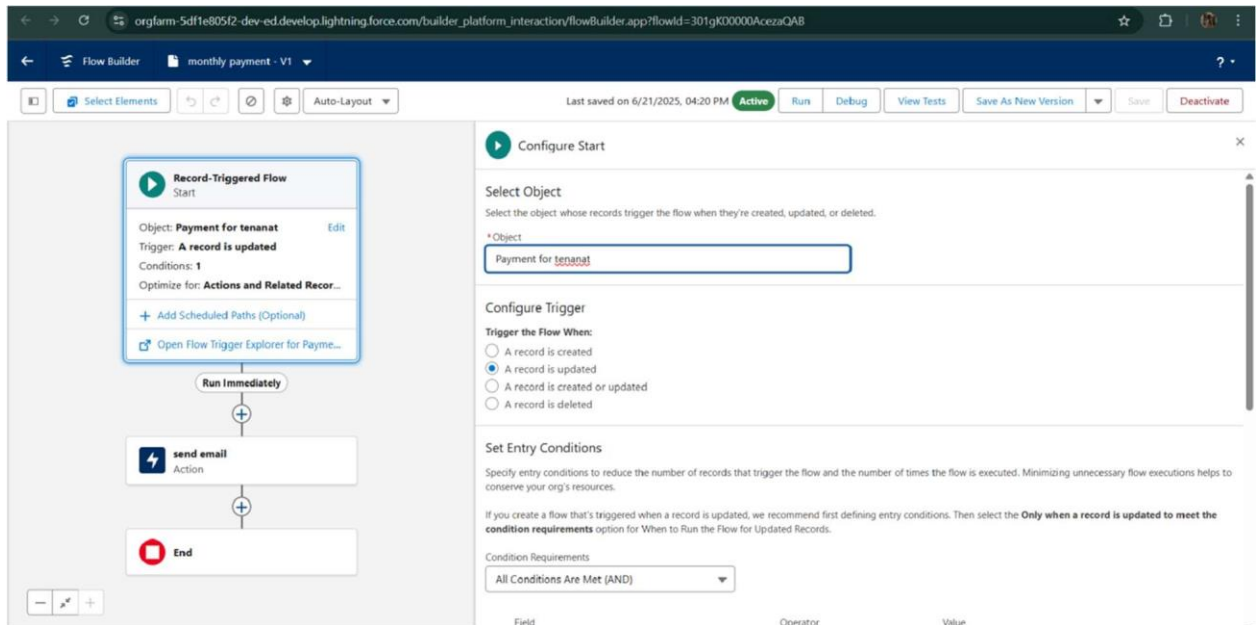
```
Developer Console - Google Chrome
orgfarm:5df1e805f2-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage

test.apex | testHandler.apex | Household__c.apex | testHandler.apex
Code Coverage: None | API Version: 64
do To

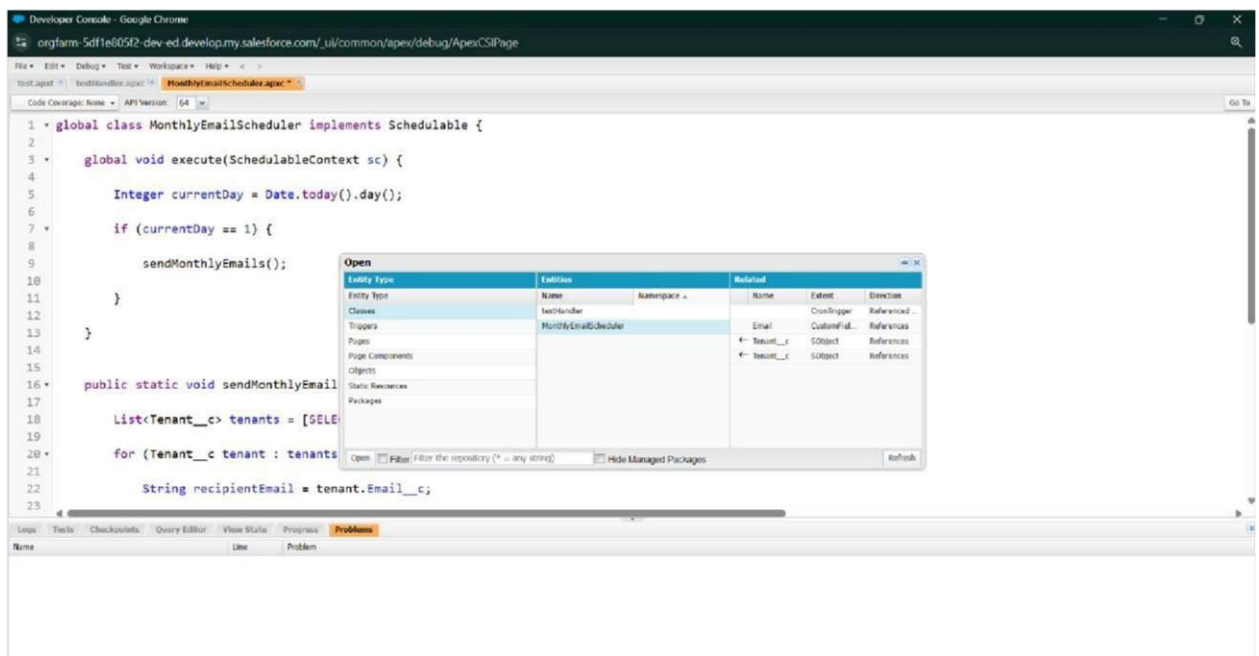
1 public class testHandler {
2
3     public static void preventInsert(List<Tenant__c> newList) {
4
5         Set<Id> existingPropertyIds = new Set<Id>();
6
7         for (Tenant__c existingTenant : [SELECT Id, Property__c FROM Tenant__c WHERE Property__c != null]) {
8
9             existingPropertyIds.add(existingTenant.Property__c);
10
11         }
12
13         for (Tenant__c newTenant : newList) {
14
15
16
17             if (newTenant.Property__c != null && existingPropertyIds.contains(newTenant.Property__c)) {
18
19                 newTenant.addError('A tenant can have only one property');
20
21             }
22
23         }
24     }
25 }
```

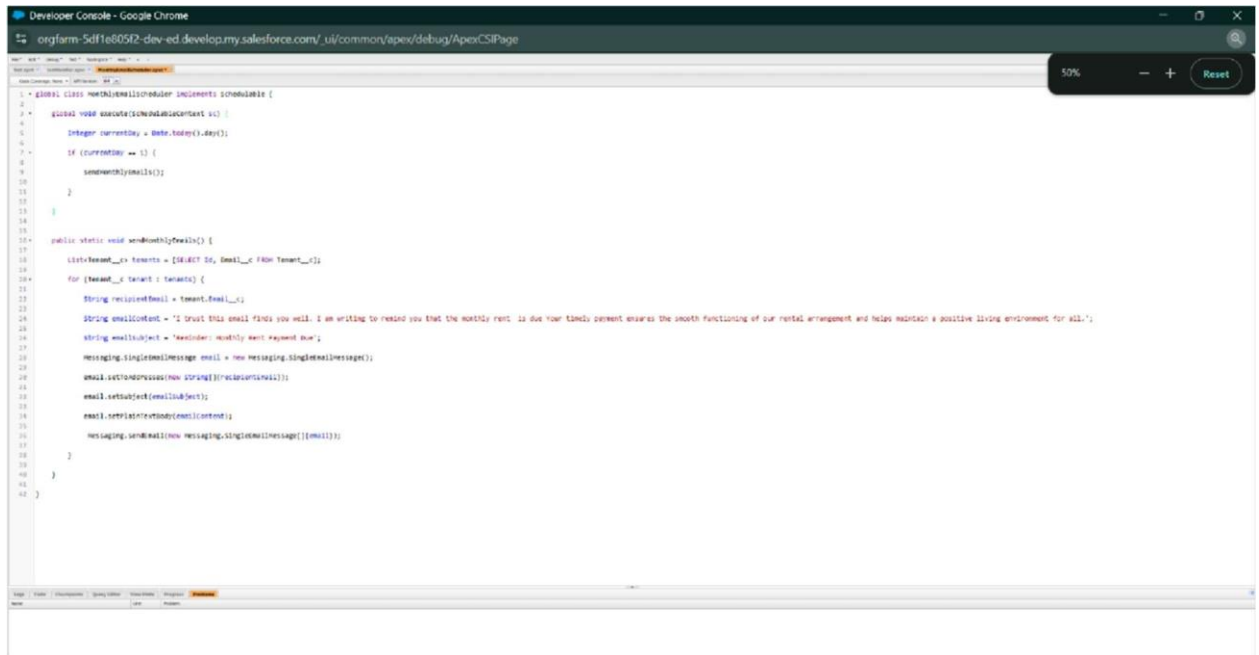
- FLOWS





- Schedule class:  
Create an Apex Class

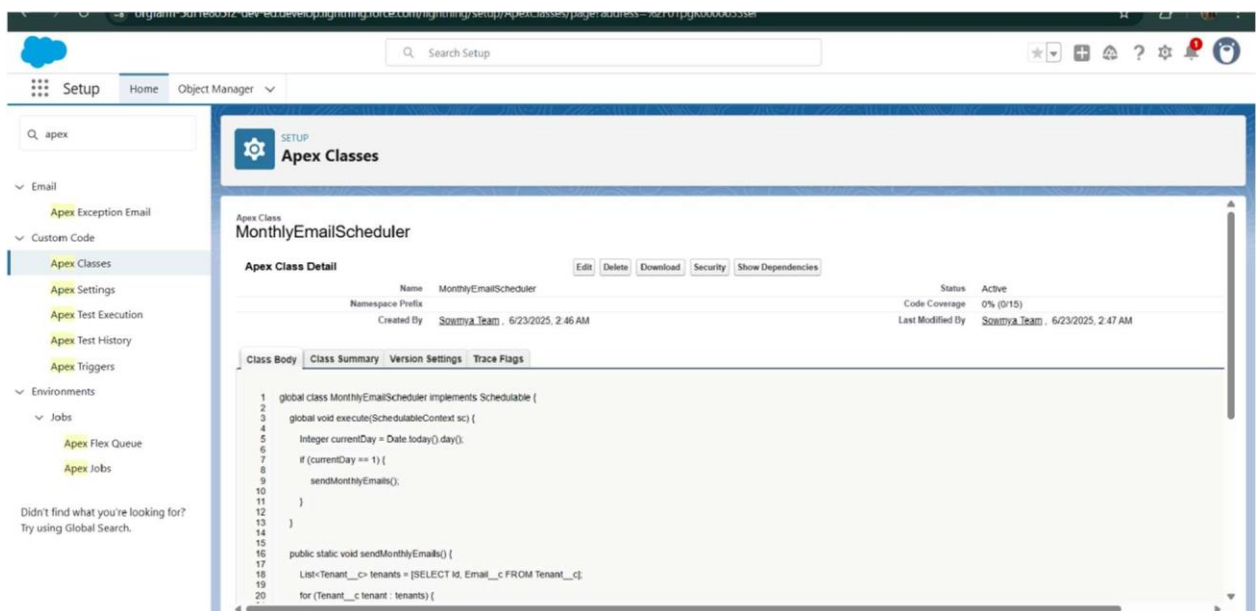




The screenshot shows the Google Chrome Developer Console with the URL `orgfarm-5d1fe605f2-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage`. The console displays the source code for the `MonthlyEmailScheduler` class, which implements the `Schedulable` interface. The code includes a `execute` method that checks the current day and sends an email if it's the first day of the month. The email content is a reminder about the monthly rent. The code is as follows:

```
1 global class MonthlyEmailScheduler implements Schedulable {
2
3     global void execute(SchedulableContext sc) {
4         Integer currentDay = Date.today().day();
5         if (currentDay == 1) {
6             sendMonthlyEmails();
7         }
8     }
9
10
11
12
13
14
15
16 public static void sendMonthlyEmails() {
17     List<Tenant__c> tenants = [SELECT Id, Email__c FROM Tenant__c];
18     for (Tenant__c tenant : tenants) {
19         String recipientEmail = tenant.Email__c;
20         String emailContent = "I trust this email finds you well. I am writing to remind you that the monthly rent is due your time; payment ensures the smooth functioning of our rental arrangement and helps maintain a positive living environment for all.";
21         String emailSubj = "Reminder: Monthly rent payment due";
22         Messaging.SingleEmailMessage email = new Messaging.SingleEmailMessage();
23         email.setBodyPlainText(emailContent);
24         email.setSubject(emailSubj);
25         email.setToAddresses(new String[] {recipientEmail});
26         email.setFromAddress(emailContent);
27         Messaging.sendEmail(new Messaging.SingleEmailMessage(email));
28     }
29 }
30 }
```

## Schedule Apex class



The screenshot shows the Salesforce Setup page for Apex Classes. The left sidebar contains a navigation menu with the following items:

- Setup
- Home
- Object Manager
- Apex
- Apex Exception Email
- Custom Code
- Apex Classes
- Apex Settings
- Apex Test Execution
- Apex Test History
- Apex Triggers
- Environments
- Jobs
- Apex Flex Queue
- Apex Jobs

The main content area displays the details for the `MonthlyEmailScheduler` class. The class is active and has a code coverage of 0% (0/15). The class was created by `Sowmya Team` on 6/23/2025 at 2:46 AM and last modified by `Sowmya Team` on 6/23/2025 at 2:47 AM. The class body is shown in the `Class Body` tab, which contains the following code:

```
1 global class MonthlyEmailScheduler implements Schedulable {
2
3     global void execute(SchedulableContext sc) {
4         Integer currentDay = Date.today().day();
5         if (currentDay == 1) {
6             sendMonthlyEmails();
7         }
8     }
9
10
11
12
13
14
15
16 public static void sendMonthlyEmails() {
17     List<Tenant__c> tenants = [SELECT Id, Email__c FROM Tenant__c];
18     for (Tenant__c tenant : tenants) {
19         String recipientEmail = tenant.Email__c;
20         String emailContent = "I trust this email finds you well. I am writing to remind you that the monthly rent is due your time; payment ensures the smooth functioning of our rental arrangement and helps maintain a positive living environment for all.";
21         String emailSubj = "Reminder: Monthly rent payment due";
22         Messaging.SingleEmailMessage email = new Messaging.SingleEmailMessage();
23         email.setBodyPlainText(emailContent);
24         email.setSubject(emailSubj);
25         email.setToAddresses(new String[] {recipientEmail});
26         email.setFromAddress(emailContent);
27         Messaging.sendEmail(new Messaging.SingleEmailMessage(email));
28     }
29 }
30 }
```

orgfarm-5df1e805f2-dev-ed.develop.lightning.force.com/lightning/r/Tenant\_\_c/a01gK00000BAhdGQAT/view

Lease Management Payment Tenants property lease

Tenant Aswini

New Contact Edit New Opportunity

Related Details

\* Required Information

\* Tenant Name: Aswini Owner: Sowmya Team

\* Email: aswinimaraadi15@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 act

Refresh

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.

New Case New Lead Delete Clone Change Owner Printable View Submit for Approval Edit Labels

orgfarm-5df1e805f2-dev-ed.develop.lightning.force.com/lightning/r/Tenant\_\_c/a01gK00000BAhdGQAT/view

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: aswinimaraadi15@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

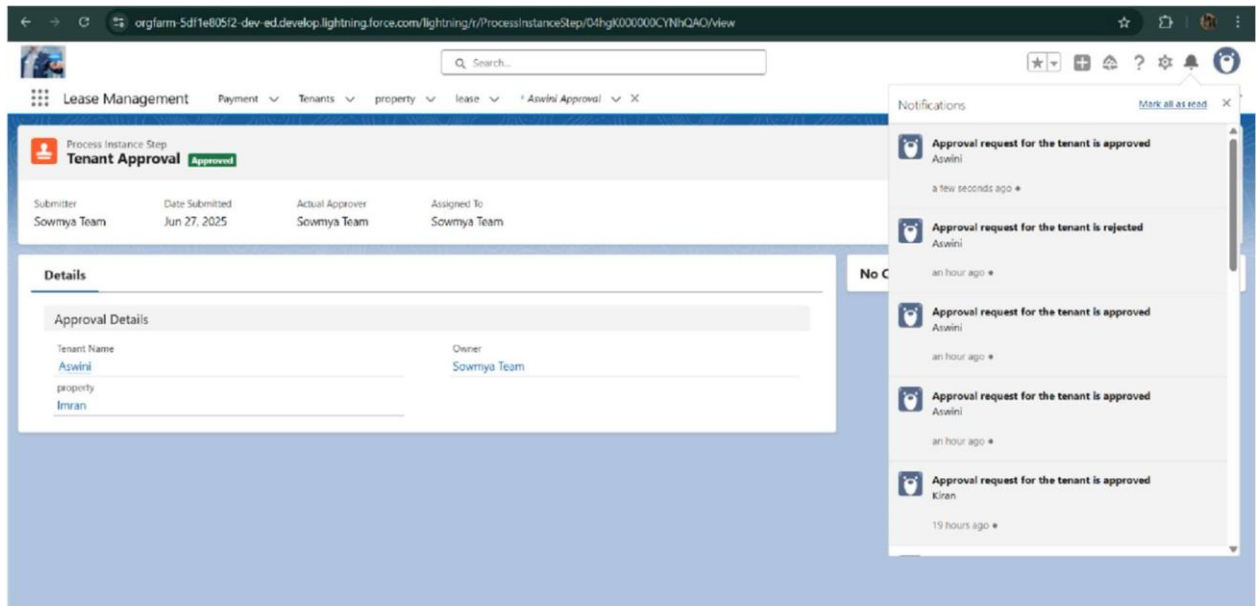
Filters: All time • All activities • All types

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.



# FUNCTIONAL AND PERFORMANCE TESTING

## Performance Testing

- Trigger validation by entering duplicate tenant-property records

**New Tenant**

\* = Required Information

Information

\* Tenant Name: chinnu

Owner: Sowmya Team

\* Email: chinnu@gmail.com

Phone:

Status: Stay

Property: Parkside

**We hit a snag.**

Review the errors on this page.

- A tenant can have only one property

Cancel Save & New Save

- Validation Rule checking

**New Lease**

\* = Required Information

Lease Name: supriya

Owner: Sowmya Team

Start Date: 6/20/2025

Your End date must be greater than start date

End Date: 6/18/2025

Property: Search property...

**We hit a snag.**

Review the following fields

- start date

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.

org/.../lease\_.../view

Lease Management | Payment | Tenants | property | lease | \* Anvini Approval | \* Anvini Approval

lease supriya

New Contact | Edit | New Opportunity

Related Details

\* Required Information

\* Lease Name: supriya

Owner: Sowmya Team

start date: 6/20/2025

Your End date must be greater than start date

End date: 6/19/2025

property: Search property...

Created By: Sowmya Team, 6/23/2025, 3:13 AM

By: Teams, 6/26/2025, 7:38 AM

We hit a snag.

Review the following fields:

- start date

Cancel Save

Activity

Filters: All time • All activities • All types

Refresh • Expand All • View All

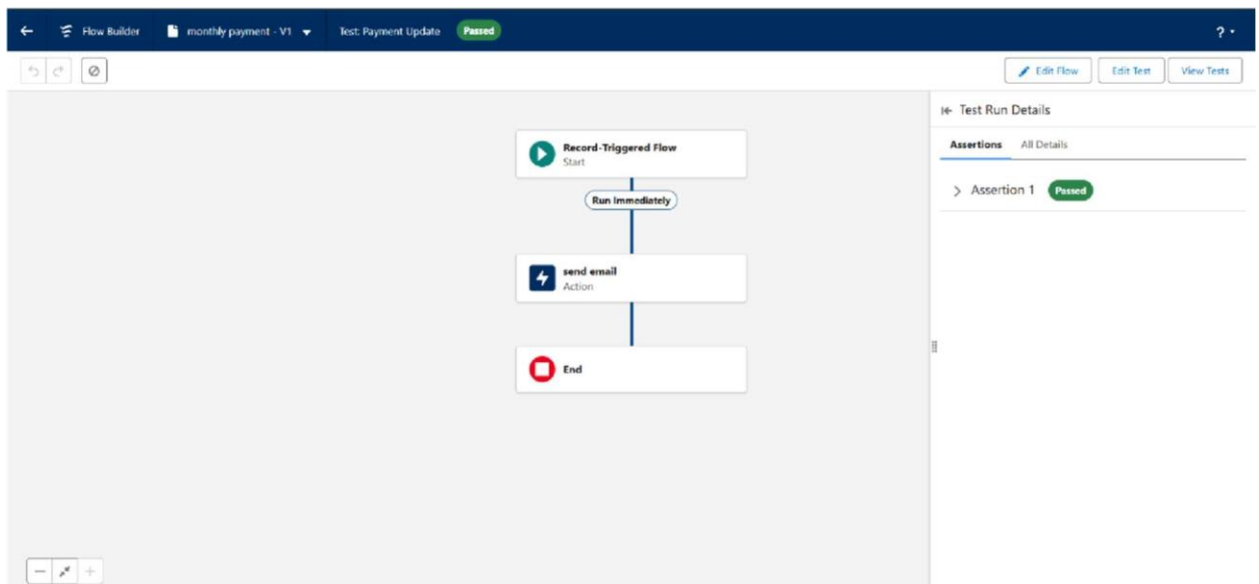
Upcoming & Overdue

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.

- Test flows on payment update





- Approval process validated through email alerts and status updates

**Tenant 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

Buttons: Cancel, Save

**Notifications**

- 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 (Jun 20, 2025, 1:28 PM)

**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)**

Payment Name

Jack

Bahul

[View All](#)

Buttons: New Contact, Edit, New Opportunity

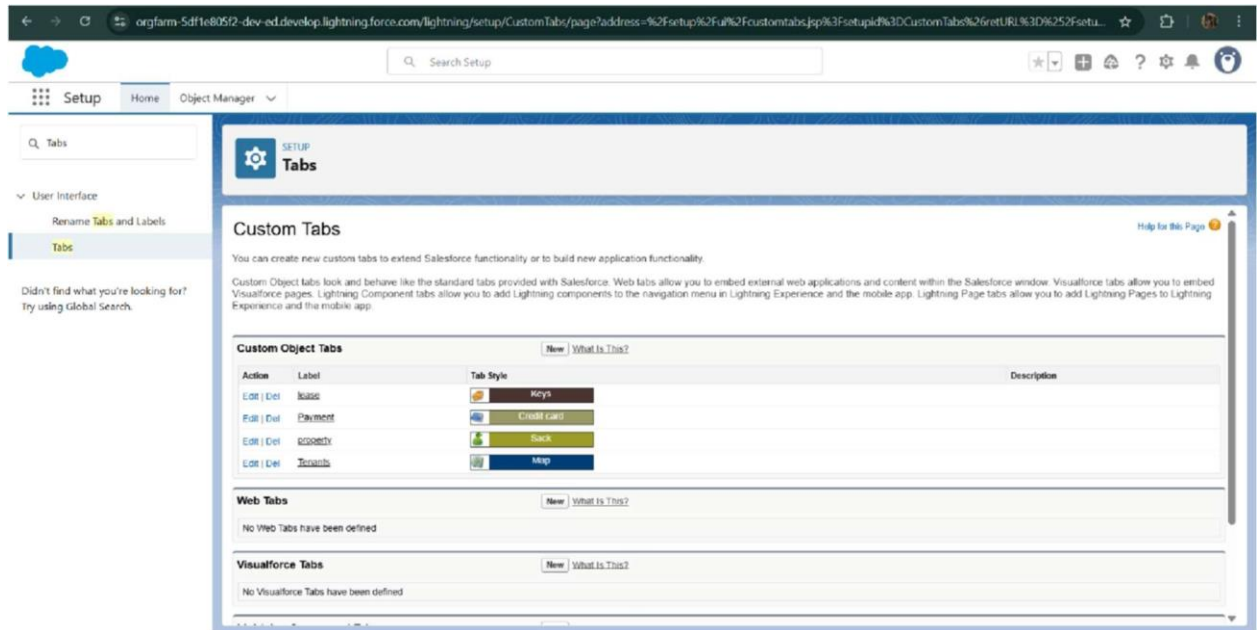
Get started by sending an email, scheduling a task, and more.

No past activity. Past meetings and tasks marked as done show up here.

# RESULTS

## Output Screenshots

- Tabs for Property, Tenant, Lease, Payment

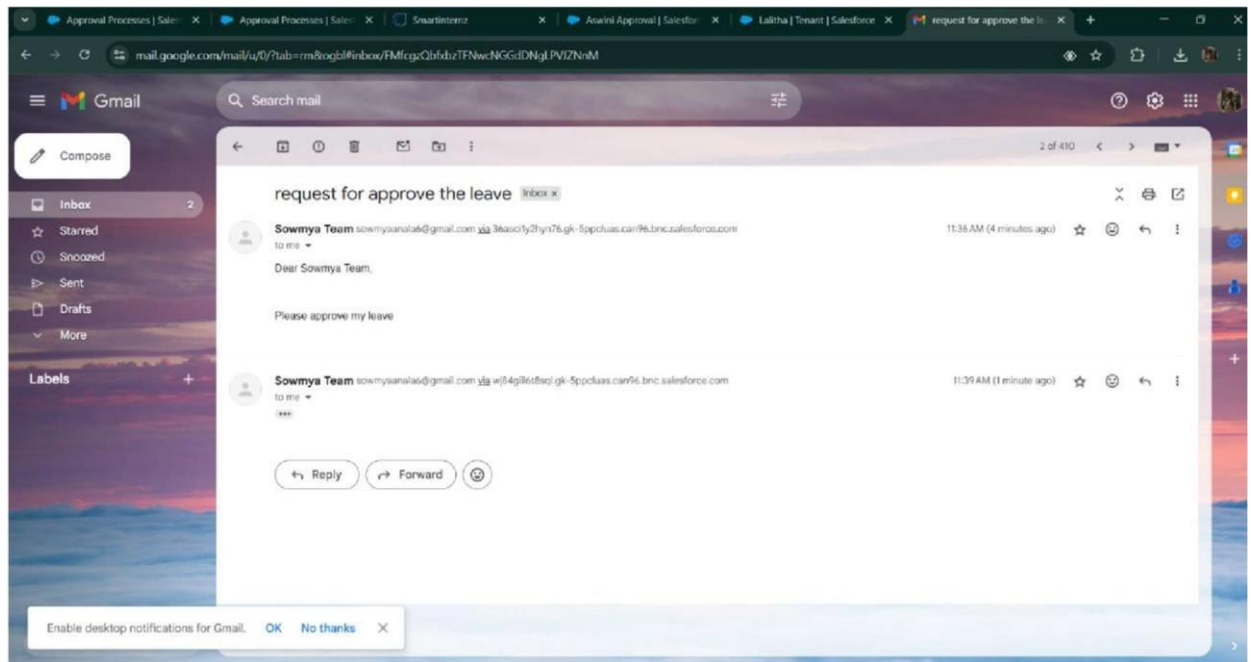


- Email alerts

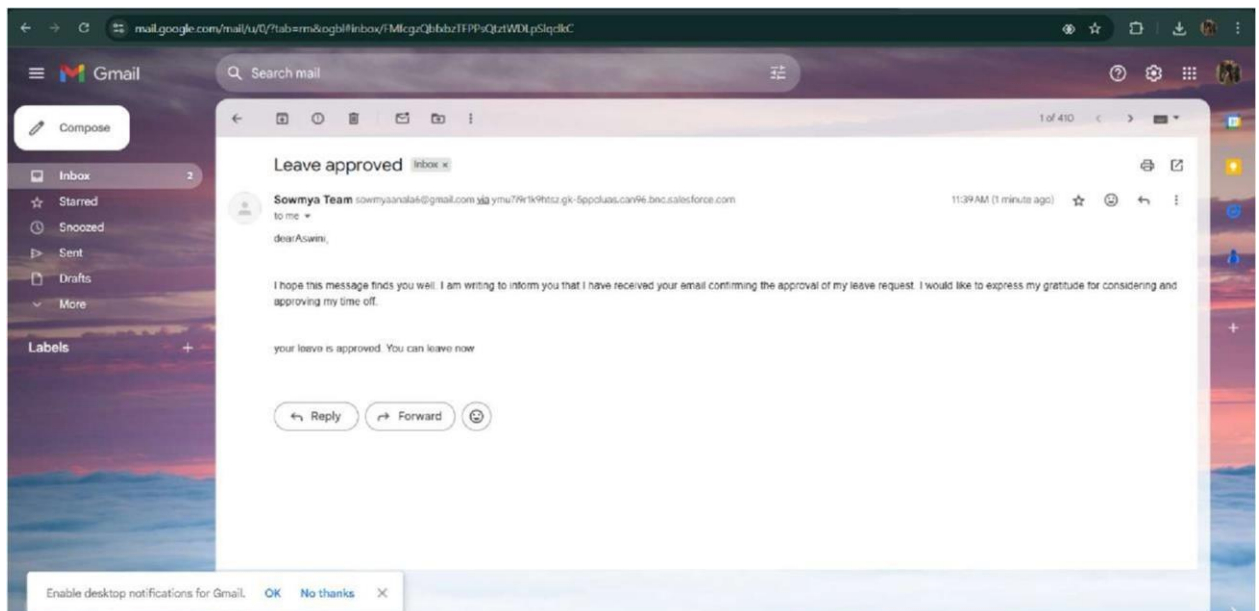
The screenshot shows the Salesforce Lease Management page. The top navigation bar includes 'Lease Management', 'Payment', 'Tenants', 'property', and 'lease'. The main content area is titled 'Approval History' and shows a table with 8 items. The table has columns for Step Name, Date, Status, Assigned To, Actual Approver, and Comments.

Step Name	Date	Status	Assigned To	Actual Approver	Comments
1 Step 1	6/25/2025, 5:39 AM	Approved	Sowmya Team	Sowmya Team	approved
2 Approval Request Submitted	6/25/2025, 5:39 AM	Submitted	Sowmya Team	Sowmya Team	leaving
3 Step 1	6/23/2025, 3:59 AM	Rejected	Sowmya Team	Sowmya Team	Rejected
4 Approval Request Submitted	6/23/2025, 3:58 AM	Submitted	Sowmya Team	Sowmya Team	Leaving
5 Step 1	6/23/2025, 3:55 AM	Approved	Sowmya Team	Sowmya Team	Approved
6 Approval Request Submitted	6/23/2025, 3:55 AM	Submitted	Sowmya Team	Sowmya Team	leaving
7 Step 1	6/23/2025, 3:44 AM	Approved	Sowmya Team	Sowmya Team	Approval Approved
8 Approval Request Submitted	6/23/2025, 3:42 AM	Submitted	Sowmya Team	Sowmya Team	Leaving

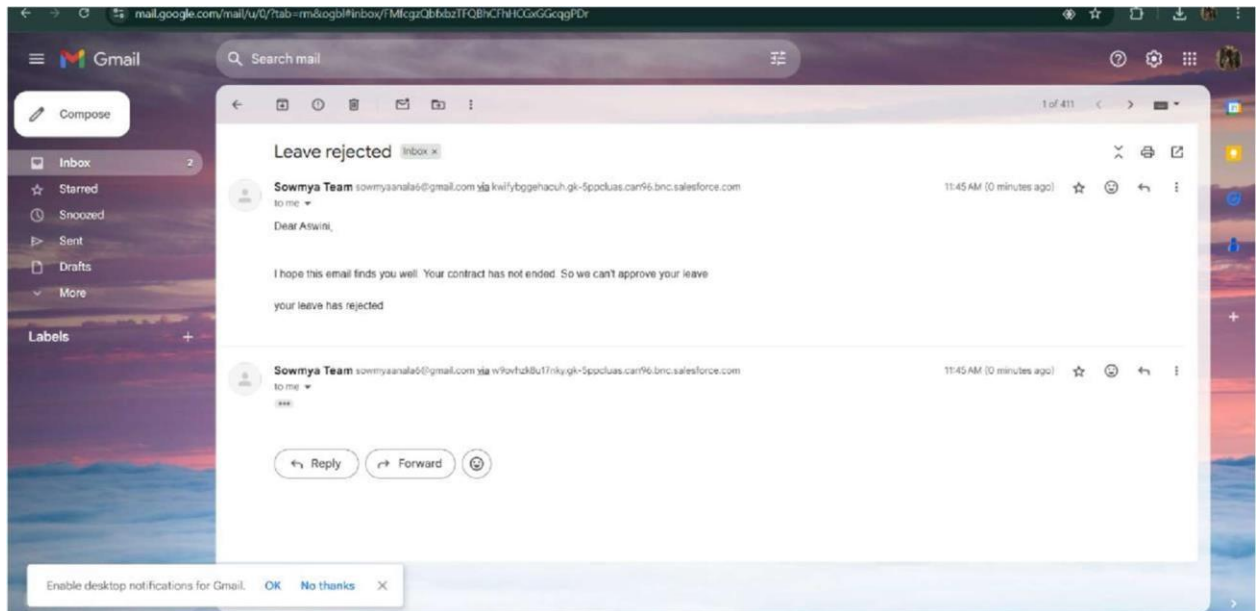
- Request for approve the leave



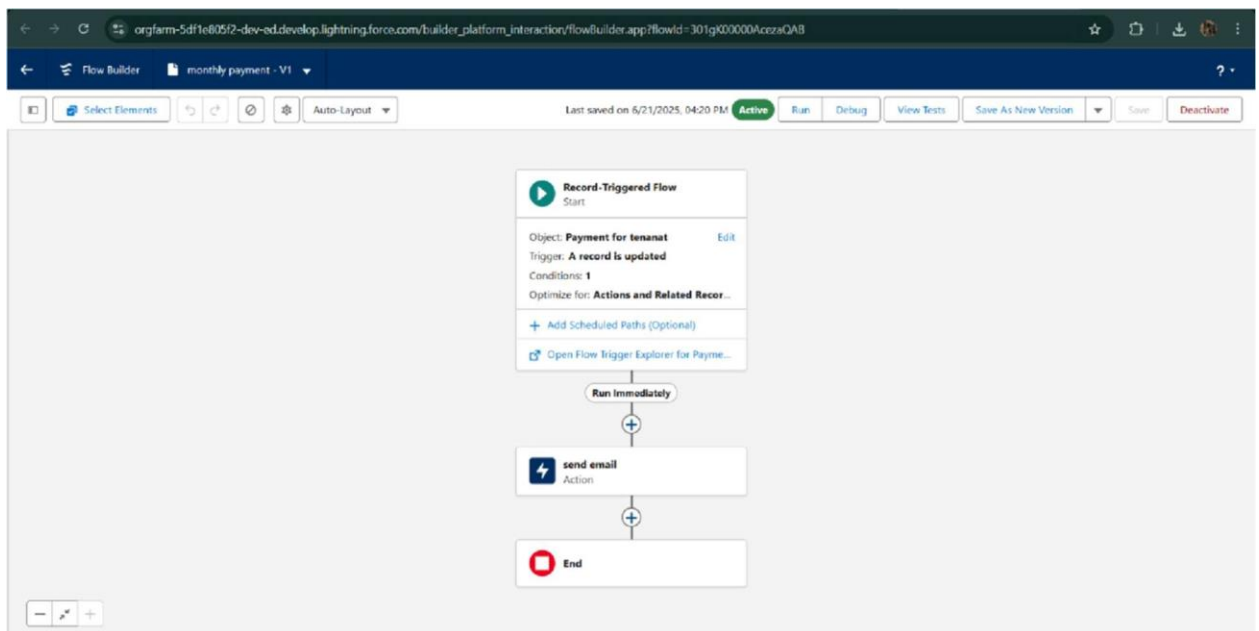
- Leave approved



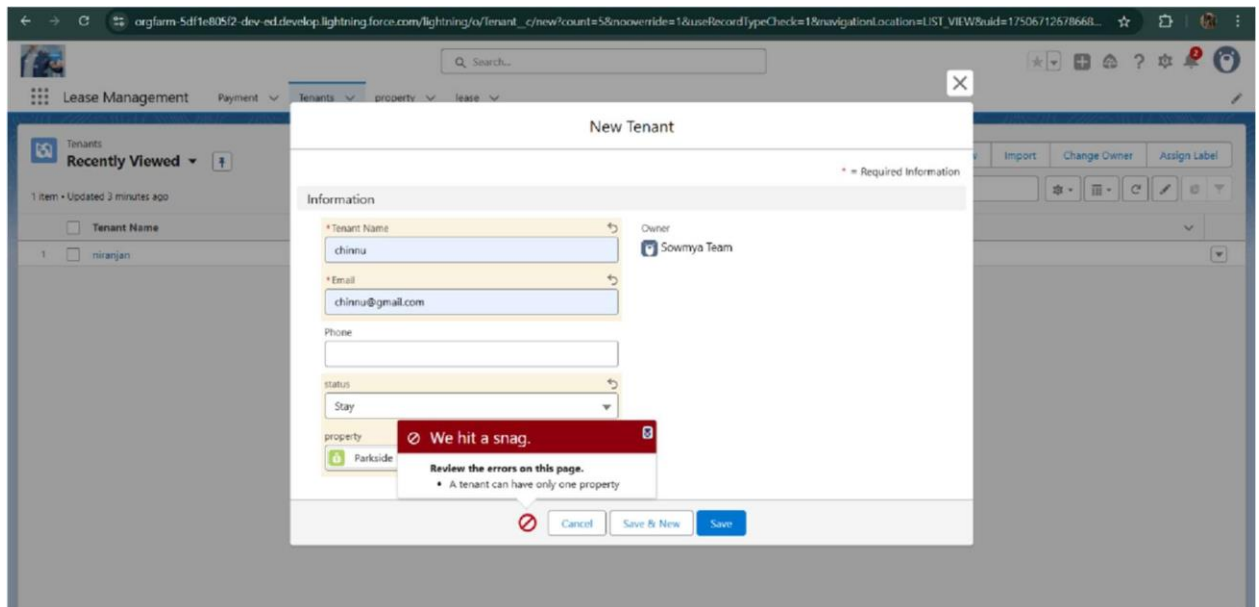
Leave rejected



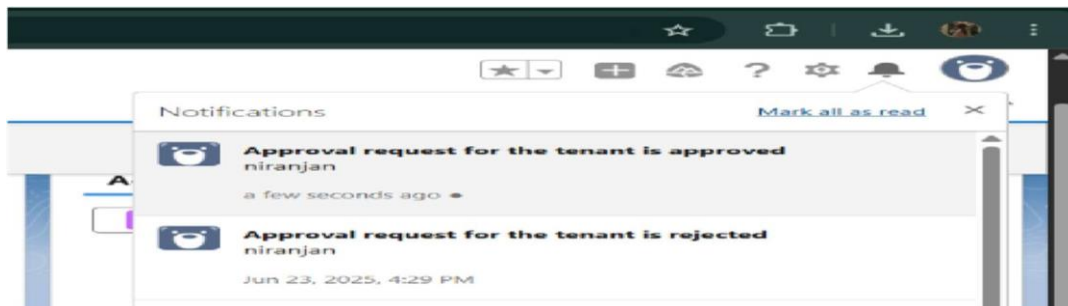
Flow runs



Trigger error messages



- Approval process notifications



# 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;

```

    } for (Tenant__c newTenant :
newlist) {

        if (newTenant.Property__c != null &&
existingPropertyIds.contains(newTenant.Property__c)) { newTenant.addError('A
        tenant can have only one property');

        }

    }

}

```

#### MonthlyEmailScheduler.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});
    }
}
}
```