

PROJECT TITLE : LEASE MANAGEMENT

CollegeName: Sri Vasavi College Self Finance Wing, Erode

CollegeCode: bru17

TEAMID: NM2025TMID2335

TEAMMEMBERS:

TeamLeaderName: BALAJI J

Email: balaji2005it@gmail.com

TeamMember1:JEEVA P

Email:jeevapalanismy8@gmail.com

TeamMember: SRIDHAR

Email: Sridharsrihdra2322@gmail.com

TeamMember: HEMALATHA T

Email: thiruarasan0221@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

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



1.2 Purpose

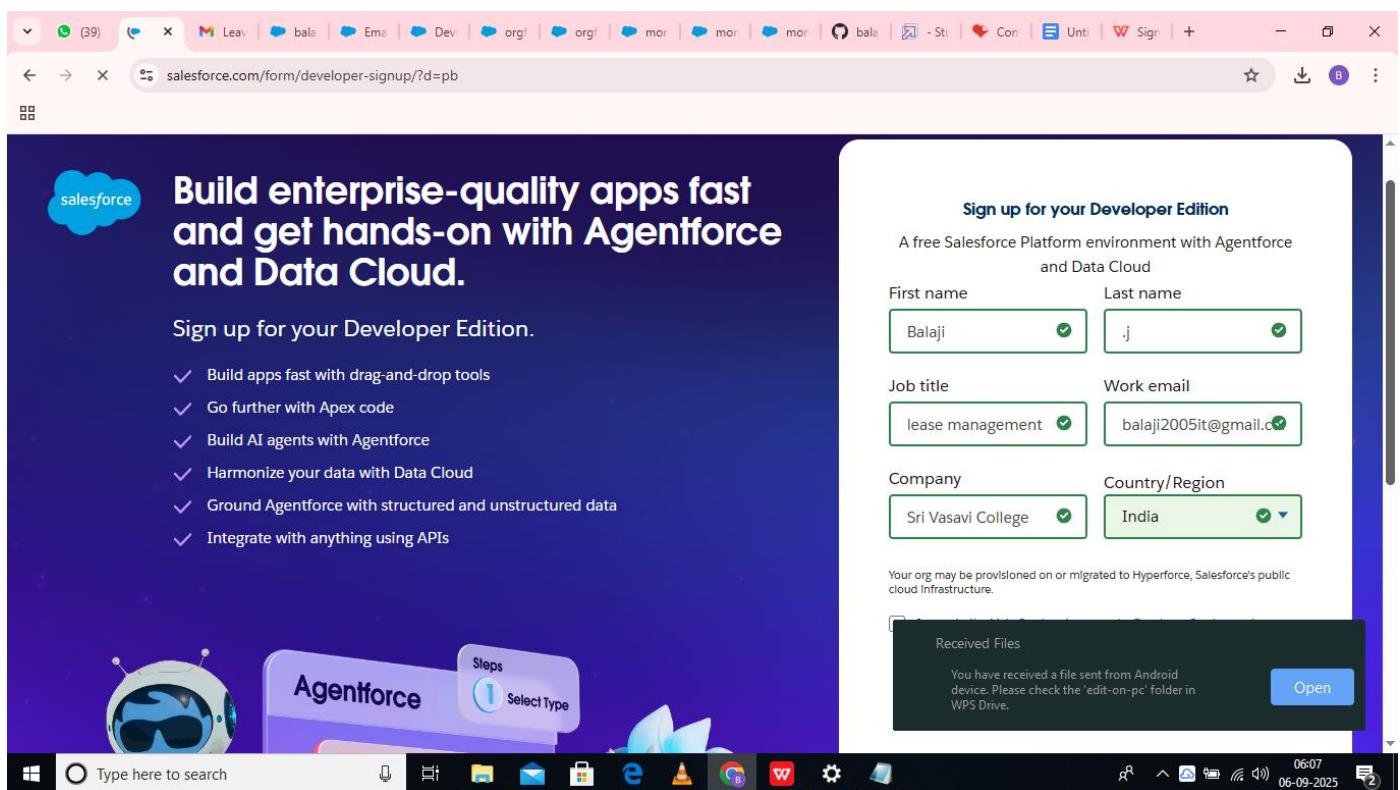
The main objective of the project is to enable organizations to efficiently manage properties, tenants, and

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

The screenshot shows the Salesforce Object Manager interface. The top navigation bar includes tabs like Setup, Home, and Object Manager. The main title is "SETUP > OBJECT MANAGER" followed by "Property". The left sidebar lists various configuration options: Fields & Relationships, Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, Restriction Rules, and Scoping Rules. The right panel displays the "Details" section for the "Property" object. It includes fields for API Name (Property__c), Singular Label (Property), Plural Label (Properties), and several checkboxes for reports and activities. The status is set to "Deployed" and "Help Settings" point to the standard salesforce.com Help Window.

The screenshot shows the Salesforce Object Manager interface. The top navigation bar includes tabs like Setup, Home, and Object Manager. The main title is "SETUP > OBJECT MANAGER" followed by "Tenant". The left sidebar lists various configuration options: Fields & Relationships, Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, Search Layouts, and List View Button Layout. The right panel displays the "Details" section for the "Tenant" object. It includes fields for API Name (Tenant__c), Singular Label (Tenant), Plural Label (Tenants), and several checkboxes for reports and activities. The status is set to "Deployed" and "Help Settings" point to the standard salesforce.com Help Window.

The screenshot shows the Salesforce Object Manager interface for the 'lease' object. The left sidebar lists various configuration tabs: Details, Fields & Relationships, Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, Restriction Rules, and Scoping Rules. The main content area displays the 'Details' tab for the 'lease' object. It includes fields for Description, API Name (lease__c), Custom (✓), Singular Label (lease), Plural Label (lease), and several checkboxes for Reports, Activities, and Field History. Deployment status is set to 'Deployed'. The URL in the browser bar is <https://orgfarm-d1005d6939-dev-ed.develop.lightning.force.com/lightning/setup/ObjectManager/01lgL0000020ekH/Details/view>. The system status bar at the bottom shows the date as 06-09-2025 and time as 06:17.

The screenshot shows the Salesforce Object Manager interface for the 'Payment for tenant' object. The left sidebar lists the same configuration tabs as the previous screenshot. The main content area displays the 'Details' tab for the 'Payment for tenant' object. It includes fields for Description, API Name (Payment_for_tenantat__c), Custom (✓), Singular Label (Payment for tenantat), Plural Label (Payment), and several checkboxes for Reports, Activities, and Field History. Deployment status is set to 'Deployed'. The URL in the browser bar is <https://orgfarm-d1005d6939-dev-ed.develop.lightning.force.com/lightning/setup/ObjectManager/01lgL0000020efR/Details/view>. The system status bar at the bottom shows the date as 06-09-2025 and time as 06:18.

Fields & Relationships
9 items, Sorted by Field Label

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Address	Address__c	Long Text Area(32768)		
Created By	CreatedById	Lookup(User)		
Last Modified By	LastModifiedById	Lookup(User)		
Name	Name__c	Text(25)		
Owner	OwnerId	Lookup(User,Group)		✓
property	property__c	Lookup(property)		✓
property Name	Name	Text(30)		✓
sfqt	sfqt__c	Text(18)		
Type	Type__c	Picklist		

- Configured fields and relationships

Fields & Relationships
7 items, Sorted by Field Label

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)		✓

The screenshot shows the Salesforce Object Manager interface. The top navigation bar includes a cloud icon, 'Setup', 'Home', and 'Object Manager'. A search bar says 'Search Setup' and a toolbar has icons for 'New', 'Deleted Fields', 'Field Dependencies', and 'Set History Tracking'. The main area shows the 'lease' object details. On the left, a sidebar lists various configuration options under 'Fields & Relationships'. The right side displays a table titled 'Fields & Relationships' with 7 items, sorted by Field Label.

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		

Fields & Relationships
7 Items. Sorted by Field Label

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Created By	CreatedById	Lookup(User)		
Email	Email_c	Email		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		✓
Phone	Phone_c	Phone		
status	status_c	Picklist		
Tenant Name	Name	Text(80)		✓

- Developed Lightning App with relevant tabs

App Details & Branding

Give your Lightning app a name and description. Upload an image and choose the highlight color for its navigation bar.

App Details

* App Name

* Developer Name

Description

App Branding

Image Primary Color Hex Value #0070D2

Org Theme Options

App Launcher Preview

LM Lease Management

The screenshot shows the Lightning App Builder interface for configuring a navigation bar. The left sidebar has 'App Settings' selected. Under 'Navigation Items', 'User Profiles' is selected. The main area is titled 'Navigation Items' and contains a description: '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.' Below this is a 'Available Items' list and a 'Selected Items' list.

Available Items

- Accounts
- Activation Targets
- Activations
- All Sites
- Alternative Payment Methods
- Analytics
- App Launcher
- Appointment Categories
- Appointment Invitations

Selected Items

- Properties
- Tenants
- lease
- Home
- Payment

At the bottom of the page, there is a search bar with 'Type here to search' and a taskbar with various icons.

The screenshot shows the Lightning App Builder interface for configuring user profiles. The left sidebar has 'User Profiles' selected. The main area is titled 'User Profiles' and contains a description: 'Choose the user profiles that can access this app.' Below this is a 'Available Profiles' list and a 'Selected Profiles' list.

Available Profiles

- 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
- Customer Support Profile

Selected Profiles

- System Administrator

At the bottom of the page, there is a search bar with 'Type here to search' and a taskbar with various icons.

Payment
Recently Viewed

5 items • Updated a few seconds ago

	Payment Name	
1	Rahul	
2	Jack	
3	Raj	
4	Sam	
5	Lahari	

Flow Builder monthly payment - V4

Last saved on 9/5/2025, 11:31 PM Active Run Debug View Tests Save As New Version Save Deactivate

```

graph TD
    Start((Record-Triggered Flow)) --> Run[Run Immediately]
    Run --> Action[send email]
    Action --> End([End])
    
```

ImplementedFlowsformonthlyrentandpaymentsuccess

- To create a validation rule to a Lease Object

The screenshot shows the Salesforce Setup interface with the URL <https://orgfarm-5d1fe805f2-dev-ed.develop.lightning.force.com/lightning/setup/ObjectManager/01IgK000000zTuj/ValidationRules/page?address=%2F03dgK00000CRqrQAG%2Fe%3FreURL%3D%2...>. The page title is "SETUP > OBJECT MANAGER lease".

Validation Rule Edit

- Rule Name:** lease_end_date
- Active:**
- Description:** (empty)
- Error Condition Formula:**

Example: `Discount_Percent__c>30` More Examples...

If this formula expression is true, display the text defined in the Error Message area.

`End_date_c <= start_date_c`
- Functions:**
 - All Function Categories
 - ABS
 - ACOS
 - ADDMONTHS
 - AND
 - ASCII
 - ASIN
 - Insert Selected Function
 - ABS(number)
Returns the absolute value of a number, a number without its sign
 - Help on this function

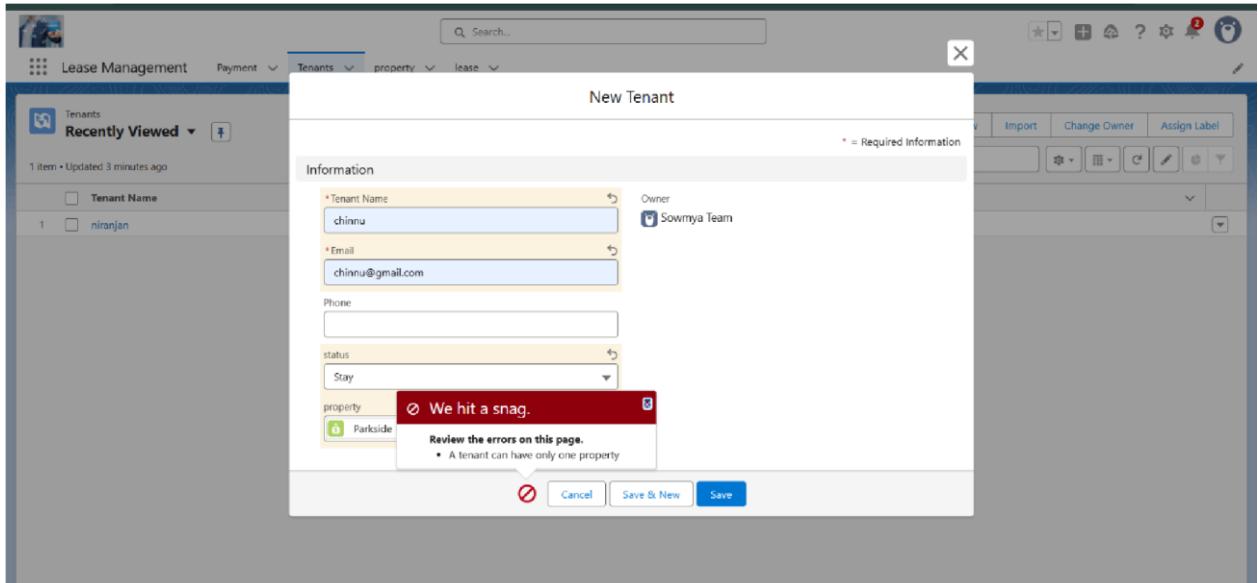
The screenshot shows the Salesforce Setup interface with the URL <https://orgfarm-5d1fe805f2-dev-ed.develop.lightning.force.com/lightning/setup/ObjectManager/01IgK000000zTuj/ValidationRules/03dgK00000CRqrQAG/view>. The page title is "SETUP > OBJECT MANAGER lease".

lease Validation Rule

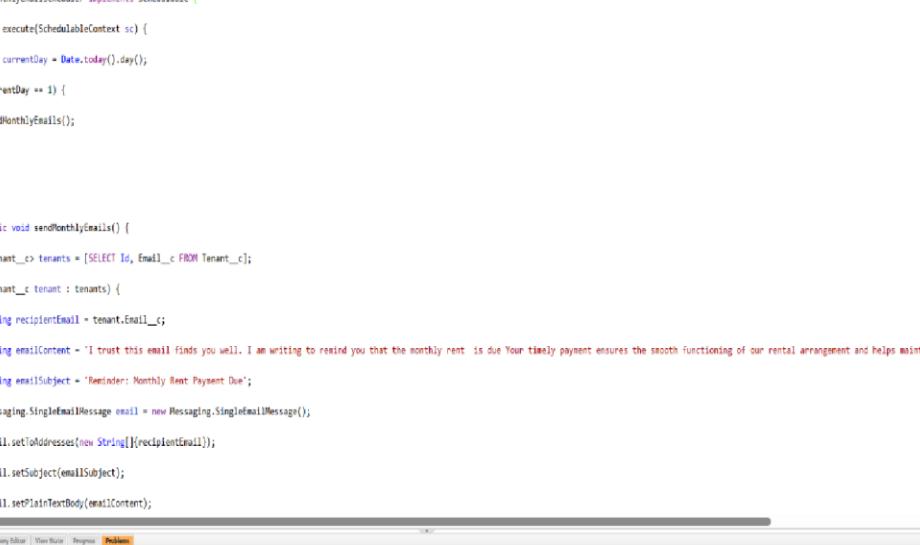
Validation Rule Detail

Rule Name	lease_end_date	Active	<input checked="" type="checkbox"/>
Error Condition Formula	<code>End_date_c <= start_date_c</code>	Error Message	Your End date must be greater than start date
Description		Error Location	start date
Created By	Sowmya Team, 6/19/2025, 5:37 AM	Modified By	Sowmya Team, 6/26/2025, 7:47 AM

- Added Apex trigger to restrict multiple tenants per property



- Scheduled monthly reminder emails using Apex classes



```
1 * global class MonthlyEmailScheduler implements Schedulable {
2
3     global void execute(SchedulableContext sc) {
4
5         Integer currentDay = Date.today().day();
6
7         if (currentDay == 1) {
8
9             sendMonthlyEmails();
10        }
11    }
12
13 }
14
15
16 public static void sendMonthlyEmails() {
17
18     List<Tenant__c> tenants = [SELECT Id, Email__c FROM Tenant__c];
19
20     for (Tenant__c tenant : tenants) {
21
22         String recipientEmail = tenant.Email__c;
23
24         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';
25
26         String emailSubject = 'Reminder: Monthly Rent Payment Due';
27
28         Messaging.SingleEmailMessage email = new Messaging.SingleEmailMessage();
29
30         email.setToAddresses(new String[]{recipientEmail});
31
32         email.setSubject(emailSubject);
33
34         email.setPlainTextBody(emailContent);
35 }
```

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

The screenshot shows the Salesforce Setup interface with the search bar set to 'email template'. Under the 'Email' category, 'Classic Email Templates' is selected. A specific template named 'Leave approved' is displayed in the center. The 'Email Template Detail' section shows the following information:

- Email Template Name:** Leave approved
- Template Unique Name:** Leave_approved
- Encoding:** Unicode (UTF-8)
- Author:** Sowmya Team [Changes]
- Description:** Created By Sowmya Team, 6/20/2025, 1:06 AM
- Modified By:** Sowmya Team, 6/20/2025, 1:06 AM
- Available For Use:** checked
- Last Used Date:** Not specified
- Times Used:** Not specified

The 'Email Template' preview section contains the following content:

```

Subject: Leave approved
Main Text Preview:
dear{Tenant__c.Name},  

I hope this message finds you well. I am writing to inform you that I have received your email confirming the approval of my leave request. I would like to express my gratitude for considering and approving my time off.  

your leave is approved. You can leave now.
  
```

The screenshot shows the Salesforce Setup interface with the search bar set to 'email template'. Under the 'Email' category, 'Classic Email Templates' is selected. A specific template named 'tenant leaving' is displayed in the center. The 'Email Template Detail' section shows the following information:

- Email Template Name:** tenant leaving
- Template Unique Name:** tenant_leaving
- Encoding:** Unicode (UTF-8)
- Author:** Sowmya Team [Changes]
- Description:** Created By Sowmya Team, 6/20/2025, 1:06 AM
- Modified By:** Sowmya Team, 6/20/2025, 1:06 AM
- Available For Use:** checked
- Last Used Date:** Not specified
- Times Used:** Not specified

The 'Email Template' preview section contains the following content:

```

Subject: request for approve the leave
Main Text Preview:
Dear {Tenant__c.CreatedBy},  

Please approve my leave
  
```

Cloud icon

Setup Home Object Manager

Search bar: Search Setup

Classic Email Templates

Email Template Detail:

Email Templates from Salesforce	Unfiled Public: Classic Email Templates
Email Template Name	Leave rejected
Template Unique Name	Leave_rejected
Encoding	Unicode (UTF-8)
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

Available For Use: ✓

Last Used Date: Times Used:

Email Template:

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

Send Test and Verify Merge Fields

Help for this Page

Cloud icon

Setup Home Object Manager

Search bar: Search Setup

Classic Email Templates

Email Template Detail:

Email Templates from Salesforce	Unfiled Public: Classic Email Templates
Email Template Name	Tenant Email
Template Unique Name	Tenant_Email
Encoding	Unicode (UTF-8)
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

Available For Use: ✓

Last Used Date: Times Used:

Email Template:

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.

Send Test and Verify Merge Fields

Help for this Page

Email Template Detail

- Email Template Name: tenant payment
- Template Unique Name: tenant_payment
- Encoding: Unicode (UTF-8)
- 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

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.

● ApprovalProcesscreation

For Tenant Leaving:

Process Definition Detail

- Process Name: TenantApproval
- Unique Name: TenantApproval
- Description: Tenant__c.status equals Stay
- Entry Criteria: Tenant__c.status equals Stay
- Record Editability: Administrator ONLY
- Active: ✓
- Next Automated Approver Determined By:
- Allow Submitters to Recall Approval Requests:

Initial Submission Actions

Action Type	Description
Record Lock	Lock the record from being edited

Approval Steps

Action	Step Number	Name	Description	Criteria	Assigned Approver	Reject Behavior
Show Actions Edit	1	Step 1			User: Sowmya Team	Final Rejection

For Check for Vacant:

The screenshot shows the Salesforce Setup interface under the Approval Processes section. A process named "Tenant: check for vacant" is displayed. The process definition details include:

- Process Name:** check for vacant
- Unique Name:** check_for_vacant
- Description:** Tenant: status EQUALS Leaving
- Entry Criteria:** Administrator ONLY
- Record Editability:** Next Automated Approver Determined By
- Approval Assignment Email Template:** leave_approved
- Initial Submitters:** Tenant Owner
- Created By:** Sowmya_Team
- Modified By:** Sowmya_Team

Initial Submission Actions:

Action	Type	Description
Record Lock	Record Lock	Lock the record from being edited
Email Alert	Email Alert	please approve my leave

Approval Steps:

Action	Step Number	Name	Description	Criteria	Assigned Approver	Reject Behavior
Show Actions	Edit	step1			User:Sowmya_Team	Final Rejection

● ApexTrigger

CreateanApexTrigger

The screenshot shows the Salesforce Developer Console with an Apex trigger named "test.apql" open. The trigger code is:

```

trigger test on Tenant_c (before insert)
{
    if(trigger.isInsert && trigger.isBefore){
        testHandler.preventInsert(trigger.new);
    }
}

```

A modal window titled "Open" is displayed, showing a list of entities in the repository. The "Triggers" option is selected, and the result "test" is listed.

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

File • Edit • Debug • Test • Workspace • Help • < >

test.apex [testHandler.apxc] MonthlyEmailScheduler.apxc []

Code Coverage: None API Version: 64 Go To

```
trigger test on Tenant__c (before insert)
{
    if(trigger.isInsert && trigger.isBefore){
        testHandler.preventInsert(trigger.new);
    }
}
```

Logs Tests Checkpoints Query Editor View State Progress Problems

Name Line Problem

CreateanApex Handlerclass

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

File • Edit • Debug • Test • Workspace • Help • < >

test.apex [testHandler.apxc] MonthlyEmailScheduler.apxc []

Code Coverage: None API Version: 64 Go To

```
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.Id);
        }
        for (Tenant__c newTenant : newList) {
            if (newTenant.Property__c != null) {
                newTenantaddError('A');
            }
        }
    }
}
```

Entity Type Entity Name Namespace Related

Entity Type	Entity Name	Namespace	Related
Classes	testHandler	MonthlyEmailScheduler	↳ test ApexTrigger Referenced
Triggers			↳ property CustomField References
Pages			↳ Tenant__c Object References
Page Components			↳ Tenant__c Object References
Objects			
Static Resources			
Packages			

Open Filter Hide Managed Packages Refresh

Logs Tests Checkpoints Query Editor View State Progress Problems

Name Line Problem

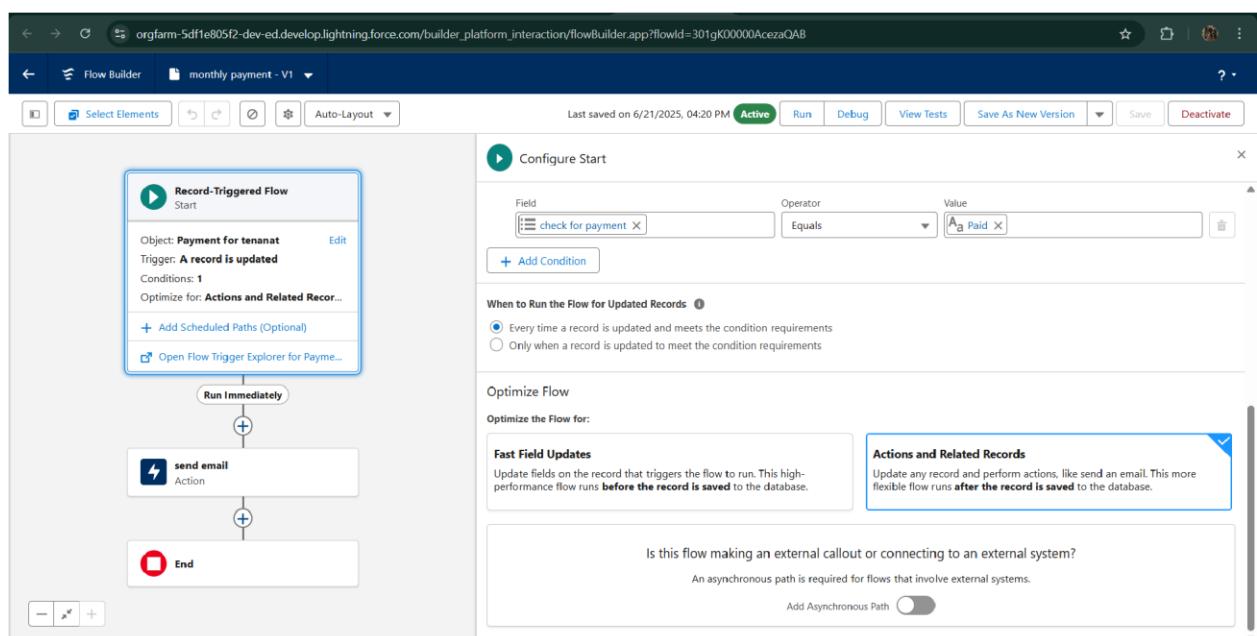
Developer Console - Google Chrome
 orgfarm-5df1e805f2-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSPage

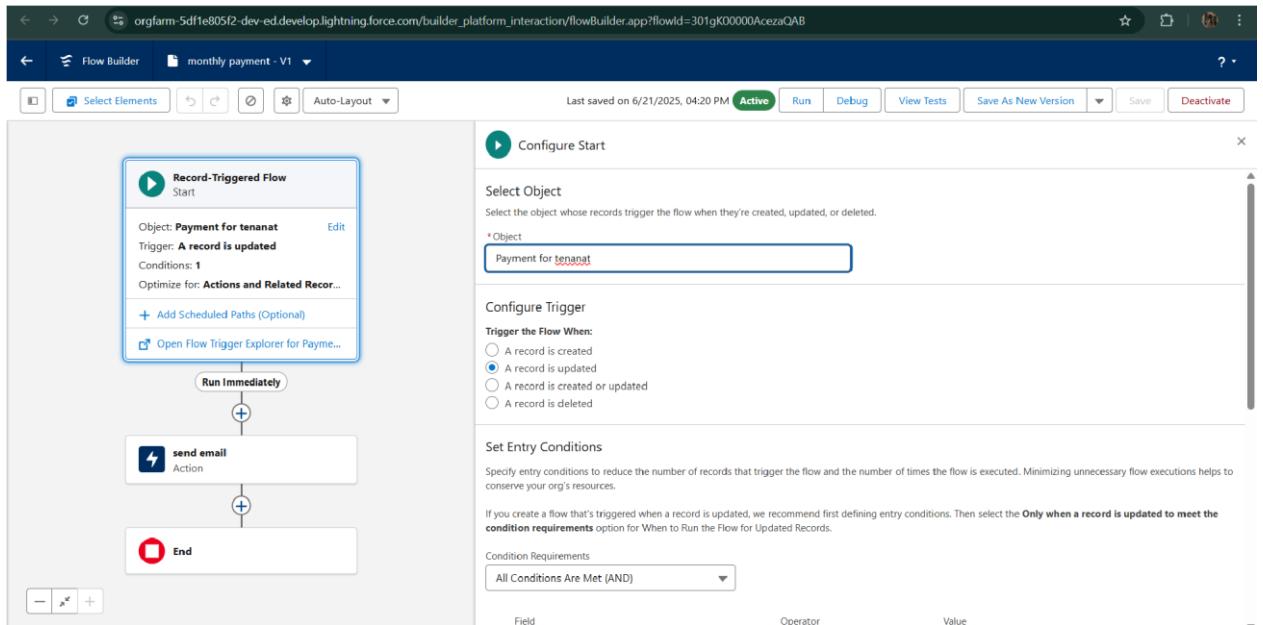
```

  File • Edit • Debug • Test • Workspace • Help • < >
  test.apfc testHandler.apfc MonthlyEmailScheduler.apfc
  Code Coverage: None • API Version: 64 Go 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             existingPropertyIds.add(existingTenant.Property__c);
  9
 10        }
 11
 12
 13        for (Tenant__c newTenant : newList) {
 14
 15
 16            if (newTenant.Property__c != null && existingPropertyIds.contains(newTenant.Property__c)) {
 17
 18                newTenantaddError('A tenant can have only one property');
 19
 20            }
 21
 22        }
 23
  
```

Logs Tests Checkpoints Query Editor View State Progress Problems

● FLOWS





- Schedule class:
CreateanApexClass

```

1 * global class MonthlyEmailScheduler implements Schedulable {
2
3     global void execute(SchedulableContext sc) {
4
5         Integer currentDay = Date.today().day();
6
7         if (currentDay == 1) {
8
9             sendMonthlyEmails();
10
11         }
12
13     }
14
15
16     public static void sendMonthlyEmail {
17
18         List<Tenant__c> tenants = [SELECT
19
20             for (Tenant__c tenant : tenants
21
22                 String recipientEmail = tenant.Email__c;

```

Developer Console - Google Chrome

org:form-5df1e805f2-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage

```

1 global class MonthlyEmailScheduler implements Schedulable {
2
3     global void execute(SchedulableContext sc) {
4
5         Integer currentDay = Date.today().day();
6
7         if (currentDay == 1) {
8
9             sendMonthlyEmails();
10        }
11    }
12
13
14
15    public static void sendMonthlyEmails() {
16
17        List<Tenant__c> tenants = [SELECT Id, Email__c FROM Tenant__c];
18
19        for (Tenant__c tenant : tenants) {
20
21            String recipientEmail = tenant.Email__c;
22
23            String emailContent = "I trust this small 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!";
24
25            String emailSubject = "Reminder: Monthly Rent Payment Due";
26
27            Messaging.SingleEmailMessage email = new Messaging.SingleEmailMessage();
28
29            email.setToAddresses(new String[] {recipientEmail});
30
31            email.setSubject(emailSubject);
32
33            email.setPlainTextBody(emailContent);
34
35            messaging.sendEmail(new Messaging.SingleEmailMessage[] {email});
36
37        }
38    }
39
40
41
42
}

```

ScheduleApexclass

Setup | Home | Object Manager | Search Setup

Q apex

Apex Classes

Apex Class MonthlyEmailScheduler

Apex Class Detail	
Name	MonthlyEmailScheduler
Namespace Prefix	
Created By	Somya Team, 6/23/2025, 2:46 AM
Status	Active
Code Coverage	0% (0/15)
Last Modified By	Somya Team, 6/23/2025, 2:47 AM
Edit Delete Download Security Show Dependencies	
Class Body Class Summary Version Settings Trace Flags	
<pre> 1 global class MonthlyEmailScheduler implements Schedulable { 2 3 global void execute(SchedulableContext sc) { 4 5 Integer currentDay = Date.today().day(); 6 7 if (currentDay == 1) { 8 9 sendMonthlyEmails(); 10 } 11 } 12 13 14 15 public static void sendMonthlyEmails() { 16 17 List<Tenant__c> tenants = [SELECT Id, Email__c FROM Tenant__c]; 18 19 for (Tenant__c tenant : tenants) { 20 21 String recipientEmail = tenant.Email__c; 22 23 String emailContent = "I trust this small 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!"; 24 25 String emailSubject = "Reminder: Monthly Rent Payment Due"; 26 27 Messaging.SingleEmailMessage email = new Messaging.SingleEmailMessage(); 28 29 email.setToAddresses(new String[] {recipientEmail}); 30 31 email.setSubject(emailSubject); 32 33 email.setPlainTextBody(emailContent); 34 35 messaging.sendEmail(new Messaging.SingleEmailMessage[] {email}); 36 37 } 38 } 39 40 41 42 } </pre>	

Didn't find what you're looking for?
Try using Global Search.

Lease Management Payment Tenants property lease

Tenant Aswini

Related Details

* = Required Information

Tenant Name: Aswini
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

New Contact Edit New Opportunity

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.

Submit for Approval

Edit Labels

Lease Management Payment Tenants property lease

Tenant Aswini

Related Details

* = Required Information

Tenant Name: Aswini
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

New Contact Edit New Opportunity

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.

Cloudy 30°C 12:46 27-06-2025

Lease Management Payment Tenants property lease Asvini Approval

Process Instance Step Tenant Approval Approved

Submitter Sowmya Team Date Submitted Jun 27, 2025 Actual Approver Sowmya Team Assigned To Sowmya Team

Details

Approval Details

Tenant Name Aswini Owner Sowmya Team

property Imran

Notifications

Approval request for the tenant is approved Asvini a few seconds ago

Approval request for the tenant is rejected Asvini an hour ago

Approval request for the tenant is approved Asvini an hour ago

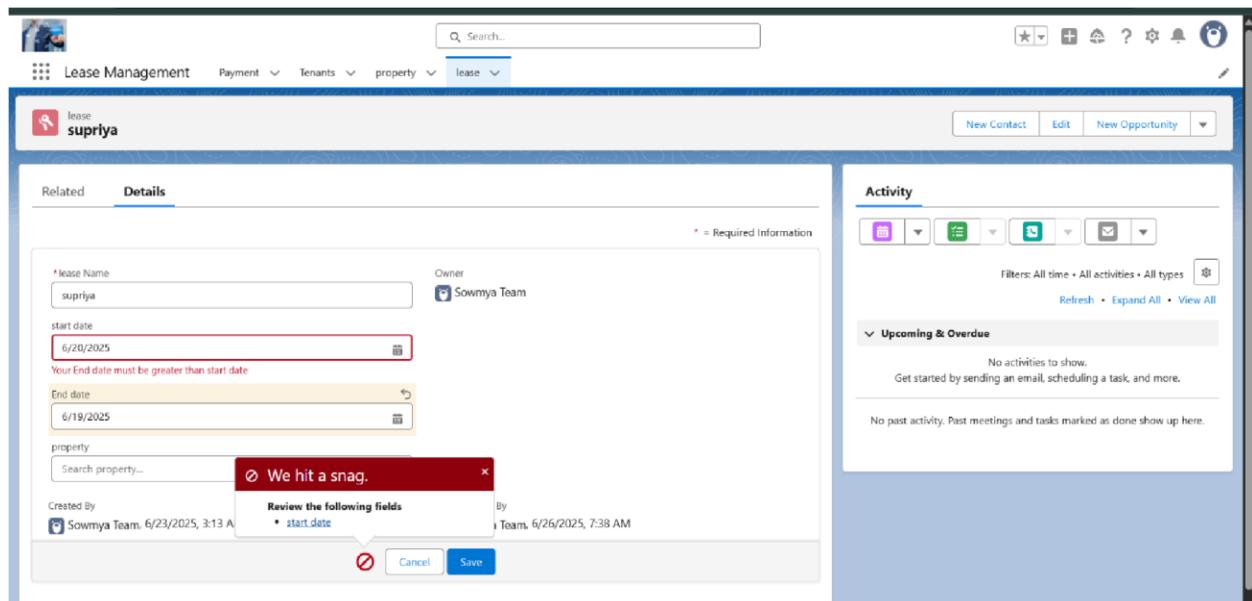
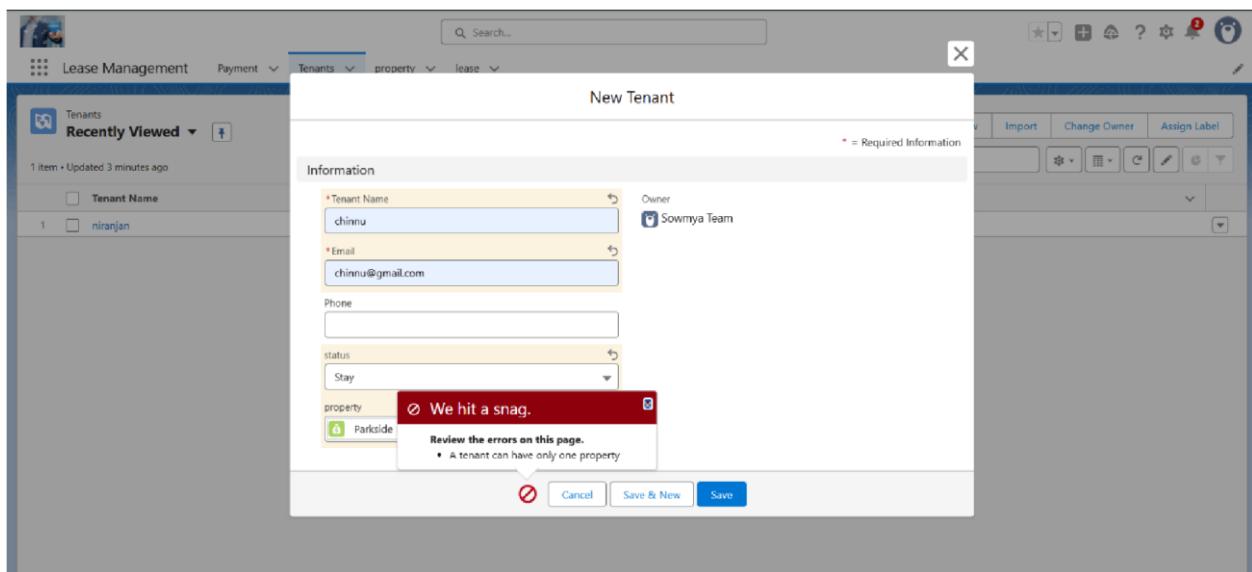
Approval request for the tenant is approved Asvini an hour ago

Approval request for the tenant is approved Kiran 19 hours ago

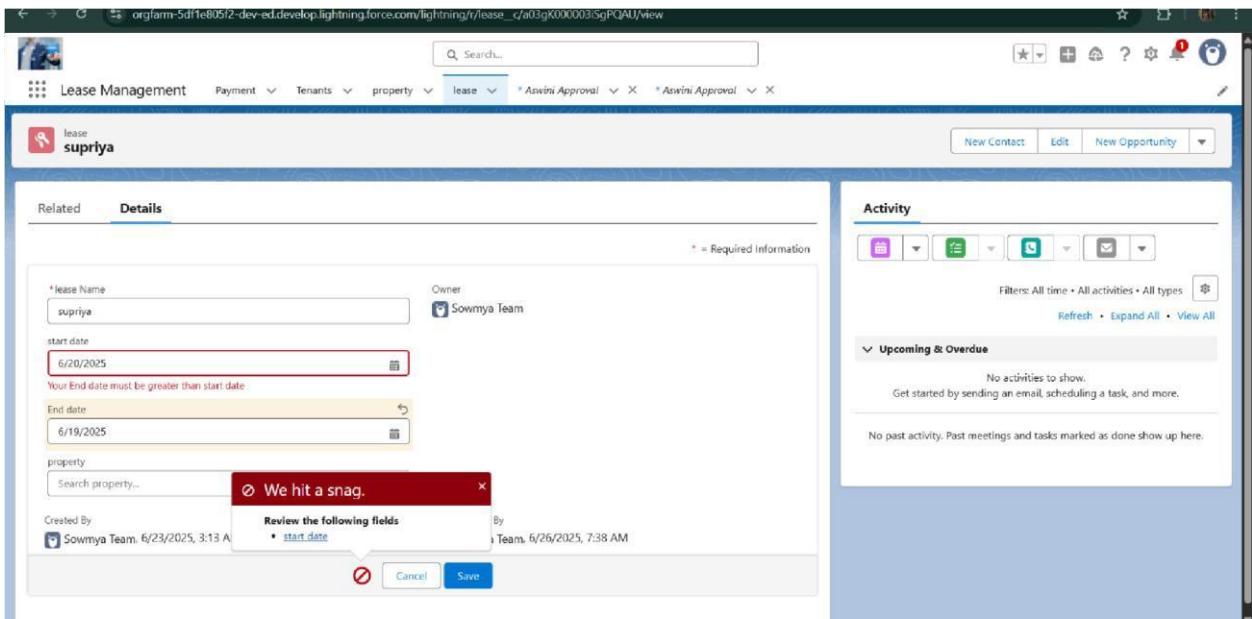
FUNCTIONAL AND PERFORMANCE TESTING

Performance Testing

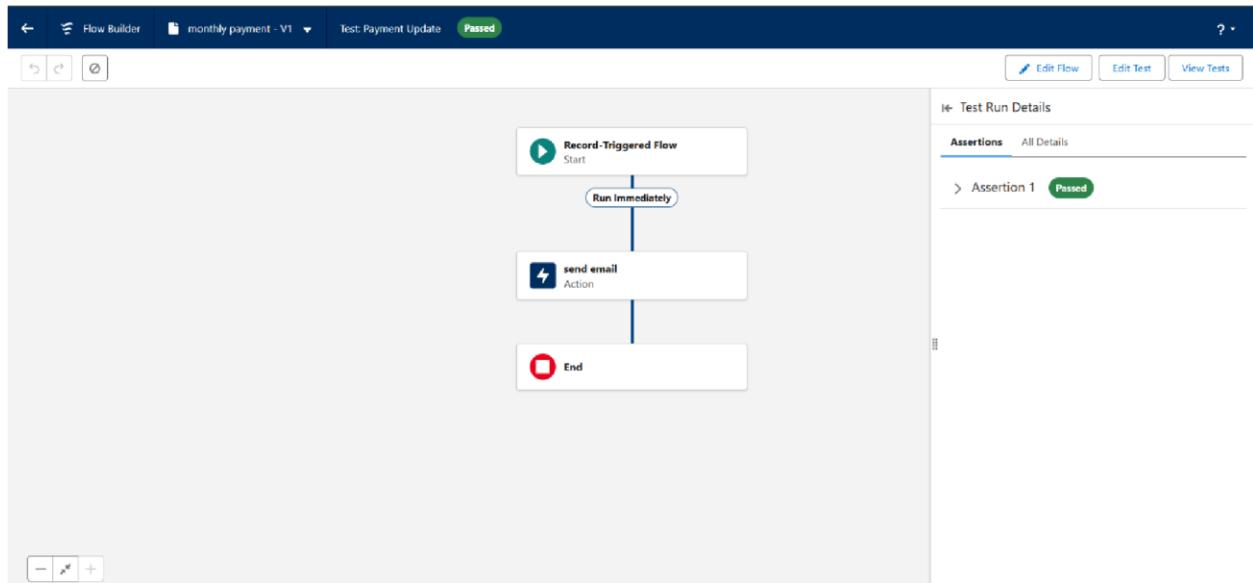
- Trigger validation by entering duplicate tenant-property records



● Validation Rule Checking



- Testflows on payment update
- Approval process validated through email alerts and status updates



The screenshot shows a tenant detail page for 'niranjan'. The 'Details' tab is selected. The page includes fields for Tenant Name (niranjan), Email (niranjan1506@gmail.com), Phone, Status (Stay), and Property (Parkside Lofts). It also shows the owner as 'Sowmya Team'. The bottom right corner of the main window has a small blue badge with the number 'A'.

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** Define Your Sales Process (Jun 20, 2025, 1:28 PM)

The screenshot shows the same tenant detail page for 'niranjan'. The 'Approval History' section displays six steps: Step 1 (Approved), Approval Request Submitted (Submitted), Step 1 (Rejected), Approval Request Submitted (Submitted), Step 1 (Approved), and Approval Request Submitted (Submitted). The 'Payment' section shows two entries: 'Jack' and 'Rahul'.

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

Payment (2)

Payment Name
Jack
Rahul

RESULTS

Output Screenshots

- Tabs for Property, Tenant, Lease, Payment

The screenshot shows the Salesforce Setup interface under the 'Tabs' section. It displays four custom object tabs: 'Lease' (Tab Style: Keys), 'Payment' (Tab Style: Credit card), 'property' (Tab Style: Sack), and 'Tenants' (Tab Style: Map). Below these are sections for 'Web Tabs' and 'Visualforce Tabs', both of which currently have no tabs defined.

- Email alerts

The screenshot shows the 'Approval History' section of the Lease Management application. It lists eight approval steps for a leave request, each with details like date, status, assignee, and comments. The steps are: Step 1 (Approved, Sowmya Team, approved), Approval Request Submitted (Submitted, Sowmya Team, leaving), Step 1 (Rejected, Sowmya Team, Rejected), Approval Request Submitted (Submitted, Sowmya Team, Leaving), Step 1 (Approved, Sowmya Team, Approved), Approval Request Submitted (Submitted, Sowmya Team, leaving), Step 1 (Approved, Sowmya Team, Approval Approved), and Approval Request Submitted (Submitted, Sowmya Team, Leaving).

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

- Request for approvethelave

- Leaveapproved

The screenshot shows a Gmail inbox with a search bar containing "in:spam". A single email message is visible, titled "Leave approved". The message is from "Balaji.j via x1giowt98tb3cx.gl-aacsvuaf.can98.bnc.salesforce.com" and was sent "to me" at 12:37 AM (6 hours ago). The message content is as follows:

Why is this message in spam? This message is similar to messages that were identified as spam in the past.

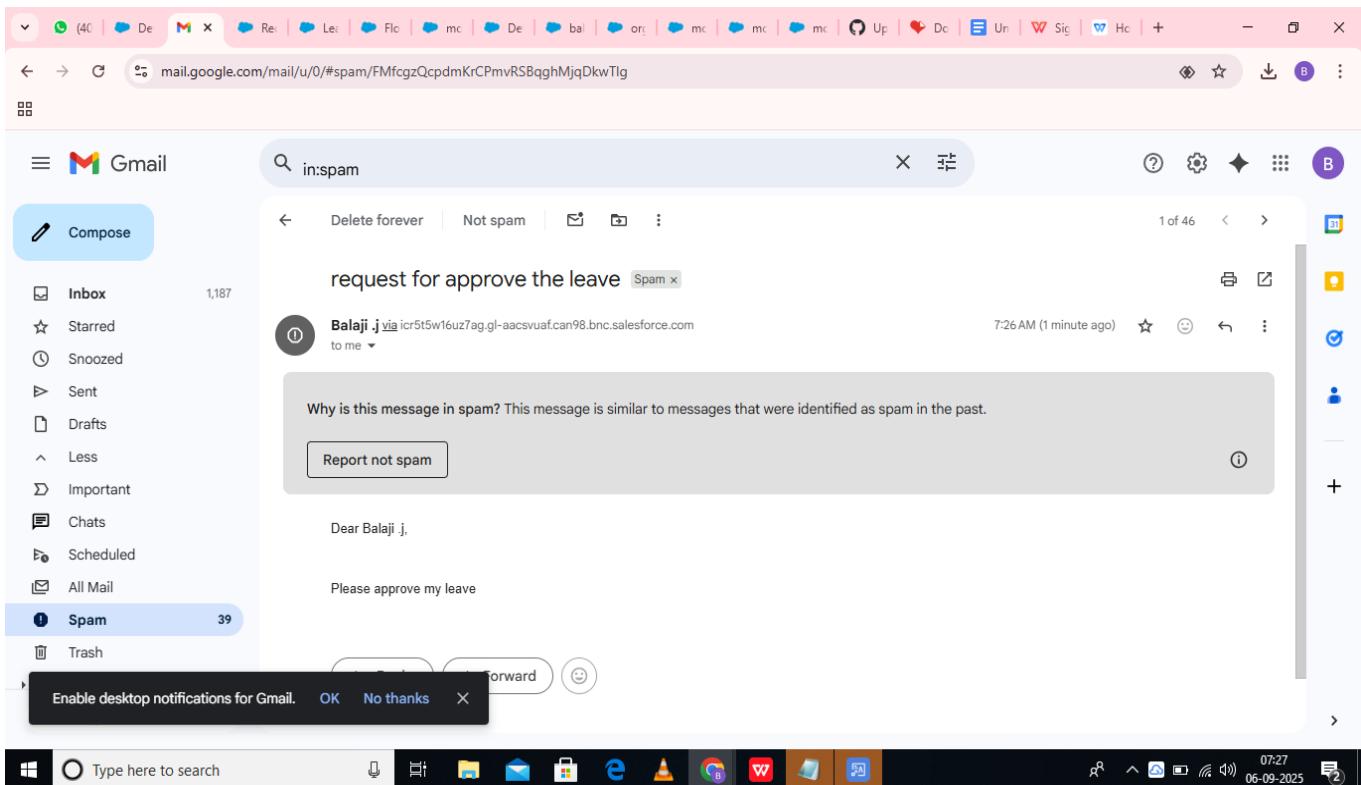
Report not spam

dearbalaji,

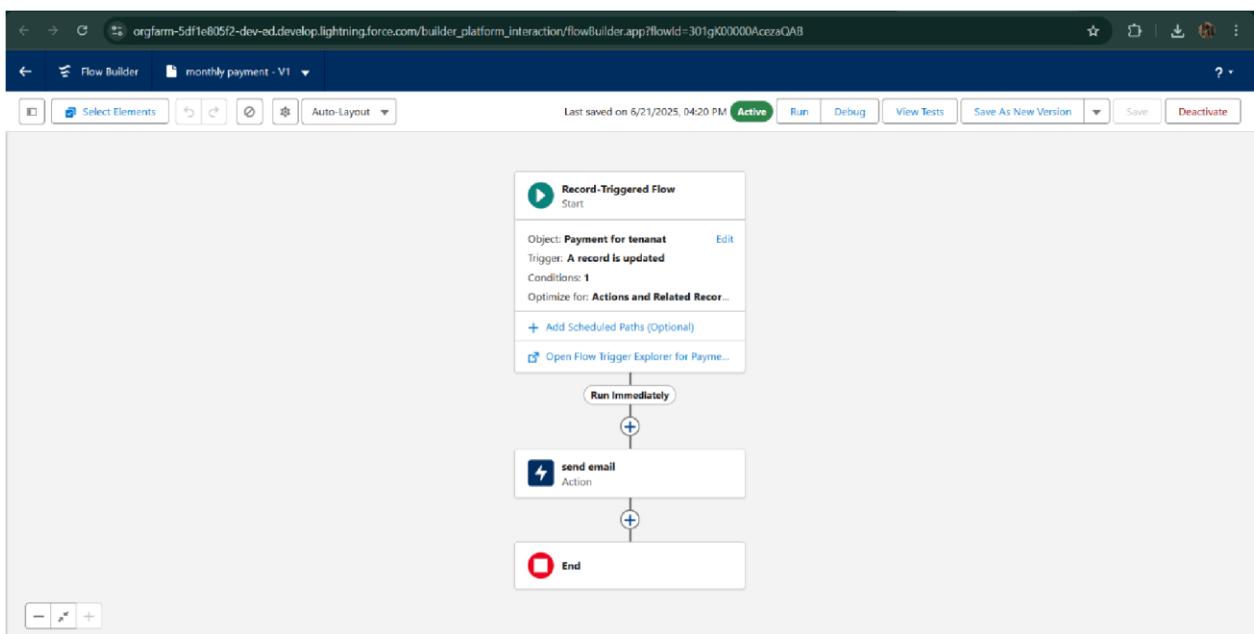
I hope this message finds you well. I am writing to inform you that I have received your email confirming the approval of my leave request. I would like to express my gratitude for considering and approving my time off.

your leave is approved. You can leave now

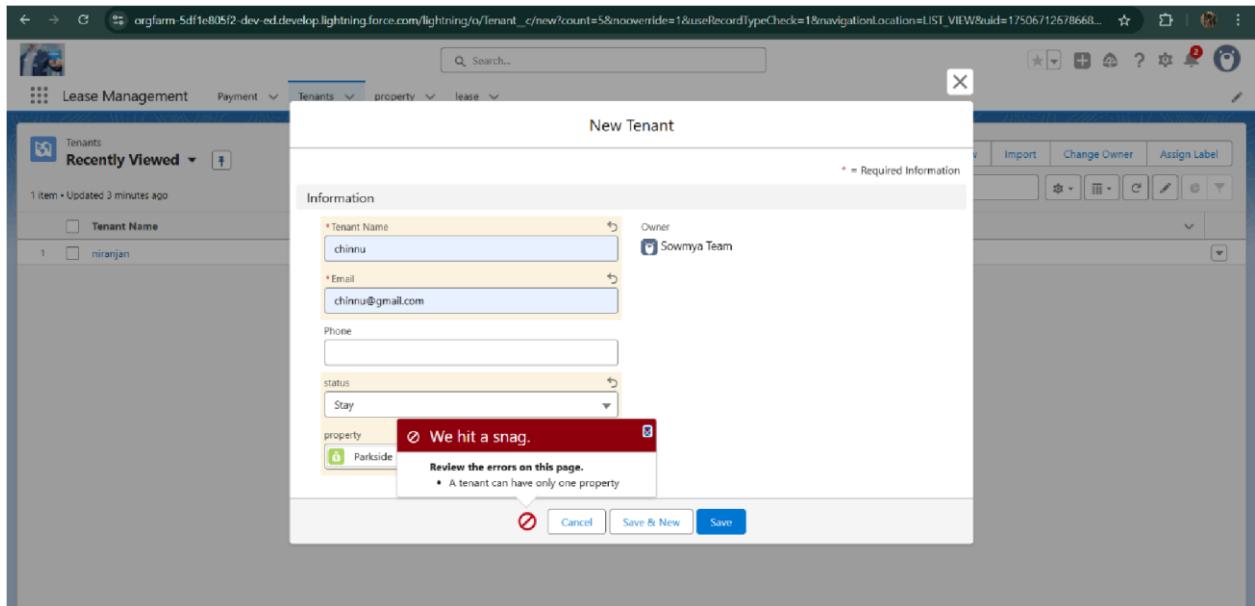
The taskbar at the bottom of the screen includes icons for File Explorer, Mail, Edge, and other system icons, along with the date and time (06-09-2025, 07:21).



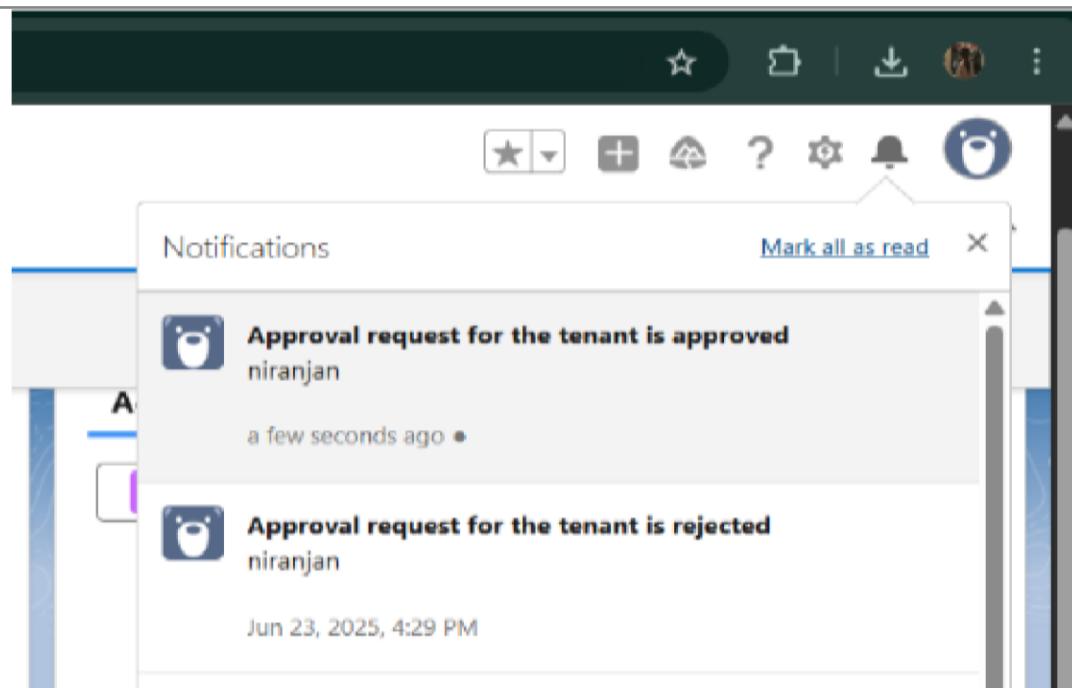
● Flowruns



● Triggererrormessages



- Approvalprocessnotifications



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

- **SourceCode:** Provided in Apex Classes and Triggers

Test.apxt:

```
trigger test on Tenantc(before insert){ if (trigger.isInsert &&
trigger.isBefore){ testHandler.preventInsert(trigger.new);
} }
```

testHandler.apxc:

```
public class
testHandler {
public static void
preventInsert(List<
Tenantc> newList)
{
    Set<Id>
existingPropertyIds
=newSet<Id>()
for(Tenantc existingTenant:[SELECT Id, Propertyc FROM Tenantc WHERE
Propertyc != null])
{
    existingPropertyIds.add(existingTenant.Propertyc;
```

```

}for(TenantcnewTenant:

newlist) {

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

tenantcanhaveonlyoneproperty');

}

}

}

```

MothlyEmailScheduler.apxc:

```

globalclassMonthlyEmailSchedulerimplementsSchedulable{ global

void execute(SchedulableContext sc) { Integer currentDay =


Date.today().day(); if (currentDay == 1) {

sendMonthlyEmails();

}

}

public static void

sendMonthlyEmails(){List<Tenantc>

tenants=[SELECTId,EmailcFROM

Tenantc]; for (Tenantc tenant :


tenants){

StringrecipientEmail=tenant.Emailc;

StringemailContent='Itrustthisemailfindsyouwell.Iamwritingtoremindyou that the

monthly rentis dueYour timely payment ensures the smooth functioning of our rental

arrangement and helps maintain a positive living environment for all.';

StringemailSubject='Reminder:MonthlyRentPaymentDue';


```

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