

# **PROJECT TITLE**

**College Name:** **Government arts  
college**

**College Code:** **Udhagamandalam  
Brull**

## **TEAM ID:**

## **TEAM MEMBERS:**

**Team LeaderName:** **Divya sree p**

**Email:** **Divyasreep05@gmail.  
Com**

**Team Member1:** **Krishnaraj v**

**Email:** **vengadachalam@gmail  
.com**

**Team Member:** **Kevin Abishek L**

**Email:** **Kevinabi3546@gmail  
.com**

**Team Member:** **Kesavamoorthy S**

**Kesavamoorthy  
kesavamoorthy  
220@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>



# Build enterprise-quality apps fast and get hands-on with Agentforce and Data Cloud.

Sign up for your Developer Edition.

- ✓ 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
- ✓ Integrate with anything using APIs



## Sign up for your Developer Edition

A free Salesforce Platform environment with Agentforce and Data Cloud

First name Last name

Divya Sree

P

Job title

Software

Work email

divyasreep05@gmail.com

Company

Government arts

Country/Region

India

Your org may be provisioned on or migrated to Hyperforce, Salesforce's public cloud infrastructure.

I agree to the Main Services Agreement – Developer Services and Salesforce Program Agreement. I acknowledge, as described in the Developer Documentation: (1) the Developer Edition includes autonomous and other generative AI features; and (2) Salesforce may limit use of those features and the org, and may terminate any org that has been inactive for 45 days.

We value your privacy. To learn more, visit our [Privacy Statement](#).

I'm not a robot



reCAPTCHA  
Privacy - Terms

[Sign Me Up](#)



Worldwide ▾

[Legal](#) [Terms of Service](#) [Privacy Information](#) [Responsible Disclosure](#) [Trust](#) [Contact](#)  
[Cookie Preferences](#) [Your Privacy Choices](#)

© Copyright 2025 Salesforce, Inc. All rights reserved. Various trademarks held by their respective owners. Salesforce, Inc. Salesforce Tower, 415 Mission Street, 3rd Floor, San Francisco, CA 94105, United States

- Created objects: Property, Tenant, Lease, Payment



Setup | Home | Object Manager | Tenant

SETUP > OBJECT MANAGER

## Tenant

Details

Description

All Name: Tenant\_2  
Custom:  
 Singular Label: Tenant  
 Plural Label: Tenants

Module Reports  
 Task Activities  
 Task Histories  
 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  
Sharing Rules

Details

Setup | Home | Object Manager | lease

SETUP > OBJECT MANAGER

## lease

Details

Description

All Name: Lease\_2  
Custom:  
 Singular Label: Lease  
 Plural Label: Leases

Module Reports  
 Task Activities  
 Task Histories  
 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

Details

The screenshot shows the Salesforce Setup interface with the 'Object Manager' selected. A sidebar on the left lists various configuration options like Details, Fields & Relationships, Page Layouts, etc. The main area displays the 'Payment for tenant' object details. The 'Details' tab is active, showing fields such as API Name (Payment\_for\_tenant\_\_c), Custom Label (Payment for tenant), Portal Label (Payment), and several checkboxes for integrations like Enable Reports, Track Activities, and Track Field History. Buttons for Edit and Cancel are at the top right.

- Configured fields and relationships

The screenshot shows the Salesforce Setup interface with the 'Object Manager' selected. A sidebar on the left lists various configuration options like Details, Fields & Relationships, Page Layouts, etc. The main area displays the 'property' object details. The 'Fields & Relationships' tab is active, showing a table of fields with columns for FIELD LABEL, FIELD NAME, DATA TYPE, CONTROLLING FIELD, and INDEXED status. Fields listed include Address, Created By, Last Modified By, Name, Owner, property, property.Name, type, and type\_c. Buttons for Quick Find, New, Edited Fields, Field Dependencies, and Set History Tracking are at the top right.

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Address	Address_c	Long Text Area(52798)		
Created By	CreatedById	Lookup(Id)		
Last Modified By	LastModifiedById	Lookup(Id)		
Name	Name_c	Text(25)		
Owner	Owner	Lookup(Users Group)		✓
property	property_c	Lookup(property)		✓
property.Name	Name	Text(30)		✓
type	type_c	Text(10)		
Type	Type_c	Picklist		

Setup > Object Manager

### Payment for tenantat

Fields & Relationships					
	FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Page Layouts	Amount	Amount_c	Number(18, 0)		
Lightning Record Pages	check for payment	check_for_payment_c	Text		
Buttons, Links, and Actions	Created By	CreatedById	Lookup(User)		
Compact Layouts	Last Modified By	LastModifiedById	Lookup(User)		
Field Sets	Owner	OwnerId	Lookup(User, Group)		<input checked="" type="checkbox"/>
Object Limits	Payment date	Payment_date_c	Date		
Record Types	Payment Name	Name	Text(50)		<input checked="" type="checkbox"/>
Related Lookup Filters					
Search Layouts					
Grid View Button Layout					
Restriction Rules					

Setup > Object Manager

### lease

Fields & Relationships					
	FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Page Layouts	Created By	CreatedById	Lookup(User)		
Lightning Record Pages	End date	End_date_c	Date		
Buttons, Links, and Actions	Last Modified By	LastModifiedById	Lookup(User)		
Compact Layouts	Lease Name	Name	Text(50)		<input checked="" type="checkbox"/>
Field Sets	Owner	OwnerId	Lookup(User, Group)		<input checked="" type="checkbox"/>
Object Limits	property	property_c	Lookup(Property)		<input checked="" type="checkbox"/>
Record Types	start date	start_date_c	Date		
Related Lookup Filters					
Search Layouts					
Grid View Button Layout					
Restriction Rules					
Scoping Rules					

Setup Home Object Manager

SETUP > OBJECT MANAGER Tenant

**Fields & Relationships**  
7 items, Sorted by Field Label

	FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Page Layouts	Created By	CreatedById	Lookup(User)		
Lightning Record Pages	Email	Email_c	Email		
Buttons, Links, and Actions	Last Modified By	LastModifiedById	Lookup(User)		
Compact Layouts	Owner	OwnerId	Lookup(User/Group)		
Field Sets	Phone	Phone_c	Phone		
Object Limits	Status	status_c	Picklist		
Record Types	Tenant Name	Name	Text(80)		

- Developed Lightning App with relevant tabs

Lightning App Builder App Settings

App Details & Branding

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

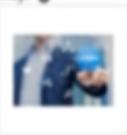
**App Details**

\*App Name: Lease Management

\*Developer Name: Lease Management

Description: Application to efficiently handle the processes related to leasing real estate properties.

**App Branding**

Image:  Primary Color Hex Value: #0070C0

Dog Theme Options:  Use the app's image and color instead of the org's custom theme

**App Launcher Preview**



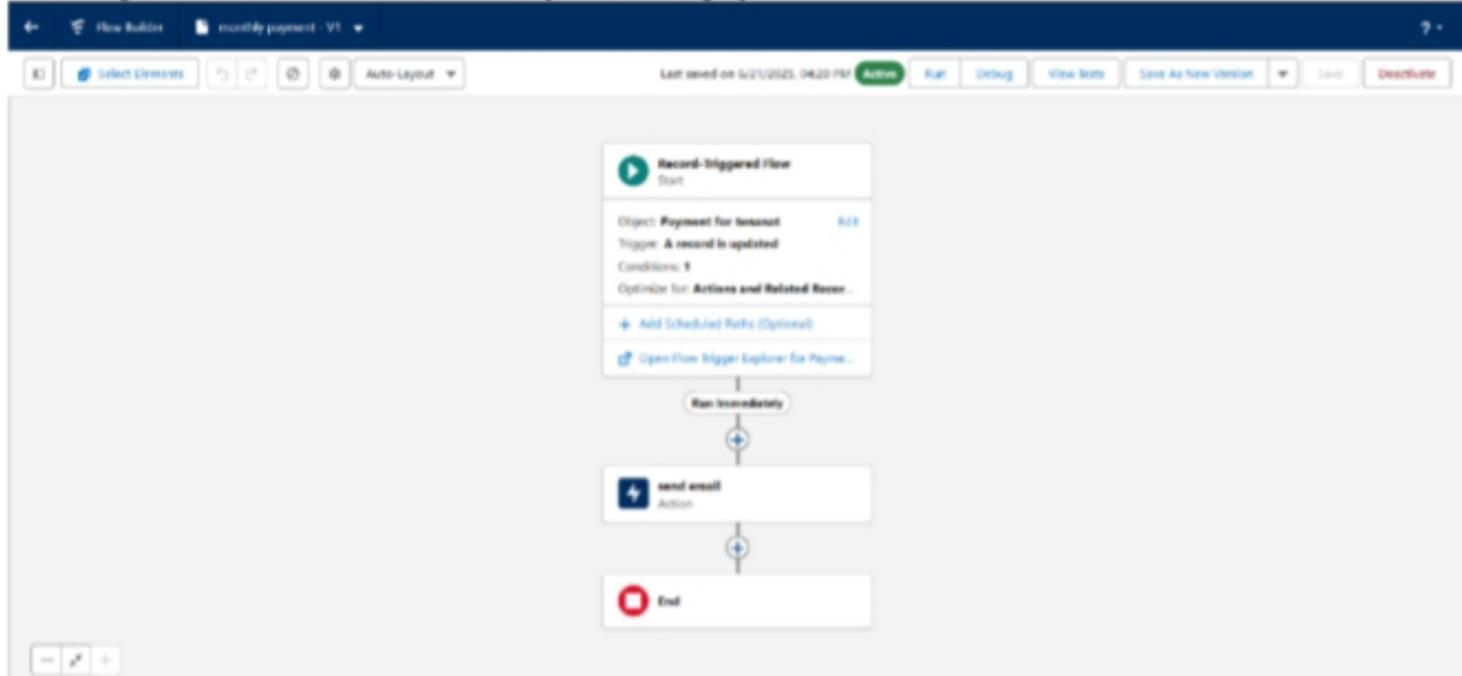
The screenshot shows the 'Navigation Items' configuration screen in the Lightning App Builder. On the left, a sidebar lists 'App Settings' (selected), 'App Details & Branding', 'App Options', and 'Utility Items (Desktop Only)'. Below that is a section titled 'Navigation Items' with a sub-section 'User Profiles'. The main area has a title 'Navigation Items' and a descriptive text about managing navigation items. It features two columns: 'Available Items' on the left and 'Selected Items' on the right. The 'Available Items' column contains a list of items with icons: Accounts (blue square), Activation Targets (purple square), Activations (light blue square), All Skins (orange square), Alternative Payment Methods (blue square), Analytics (blue square), App Launcher (green square with gear), Appointment Categories (blue square), Appointment Invitations (blue square), Approval Requests (orange square), Case (blue square), Contact (blue square), Lead (blue square), Opportunity (blue square), and User (blue square). The 'Selected Items' column contains four items with green checkmarks: 'Payment' (green square), 'Tenants' (blue square), 'property' (green square), and 'lease' (red square). Navigation arrows between the columns allow items to be moved.

**User Profiles**

Choose the user profiles that can access this app.

Available Profiles	Selected Profiles
<input type="text" value="Type to filter list."/> <ul style="list-style-type: none"> <li>Analytics Cloud Integration User</li> <li>Analytics Cloud Security User</li> <li>Anypoint Integration</li> <li>Authenticated Website</li> <li>Authenticated Website</li> <li>B2B Recording Portal Buyer Profile</li> <li>Contract Manager</li> <li>Custom Marketing Profile</li> <li>Custom Sales Profile</li> <li>Custom Support Profile</li> <li>Customer Community Login User</li> </ul>	<ul style="list-style-type: none"> <li>System Administrator</li> </ul>

- Implemented Flows for monthly rent and payment success



- To create a validation rule to a Lease Object

The screenshot shows a software application window with a dark blue header bar. In the top left, there's a cloud icon followed by a search bar with placeholder text "Search...". To the right of the search bar are several small icons: a star, a plus sign, a question mark, a gear, a bell, and a user profile. The main menu bar at the top includes "Lease Management", "Payment", and "Tenants".

The main content area has a sidebar on the left titled "Tenants" with a "Recently Viewed" section. It lists one item: "1 item · Updated a few seconds ago" followed by a table with a single row containing a checkbox, the label "Tenant Name", and the value "1" next to it, with "divyasree" listed below.

A large central panel displays two notifications:

- New Guidance Center learning resource available**  
Define Your Sales Process  
Learn how to guide reps through the sales process.  
Sep 11, 2025, 11:15 AM \*
- New Guidance Center learning resource available**  
Set Up Accounts & Contacts  
Start storing information about your customers with accounts and contacts.  
Sep 11, 2025, 11:15 AM \*

To the right of the notifications is a vertical sidebar with a title "Assign Label" and three small icons: a pencil, a circular arrow, and a magnifying glass.



Search...

Lease Management

Payment

Tenants

property

lease



Tenants

Recently Viewed



New

Import

Change Owner

Assign Label

Search this list...



1 item • Updated a few seconds ago

Tenant Name

1  divyasree



Search...

## Lease Management

Payment ▾ Tenants ▾ property ▾ lease ▾

Tenant  
divyasree

New Contact

Edit

New Opportunity ▾

## Related

## Details

Tenant Name

divyasree

Owner

divyasree.p



Email

divyasreep05@gmail.com



Phone

(987) 654-3210



Status

Stay



Created By

 divyasree.p, 9/11/2025,  
10:15 PM

Last Modified By

 divyasree.p, 9/11/2025,  
10:15 PM

## Activity



Filters: All time • All activities • All types



Refresh • Expand All • View All

## Upcoming &amp; 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.

 Search...

## Lease Management

Properties

Tenancies

Leasees

Home

Payment for tenants



Properties

Recently Viewed



New

Import

Change Owner

Assign Label

1 item • Updated a few seconds ago

Search this list...

 Property Name1  Hyd

4:03

New tenant | Salesforce × Developer Console × - Student × | +

orgfarm-2e2ee3d57c-dev-ed.lightning.force.com/lightning/o/tenant\_\_c/new?count=4&nooverride

Lease management

Search... ★ X + ? ⚙ ⚡ 2 🐻

Information

\* = Required Information

Owner: Manjubhashini k

tenant Name: Manju

Status: Stay

Email: pradheepaamsavali2006@gmail.com

Phone: 1234567890

property: Hyd

We hit a snag.

Review the errors on this page.

\* A tenant can have only one property

Cancel Save & New Save

This screenshot shows a Salesforce Lightning component for creating a new tenant record. The form includes fields for tenant name, status, email, phone number, and property location. An error message box is displayed, stating 'We hit a snag.' and 'A tenant can have only one property'. The browser toolbar at the bottom includes icons for back, forward, search, and various extensions.

 Search...

Lease Management

Properties

Tenancies

Leasees

Home

Payment for tenants



Tenancies

Recently Viewed



New

Import

Change Owner

Assign Label

2 items • Updated a few seconds ago

 Search this list... Tenant Name1  pradheepa2  pradheepa

Setup > Object Manager

### lease

**Validation Rule Edit**

Role Name: lease\_end\_date

Active:

Description:

Error Condition Formula:

Example: `Discount_Percent >= 30` [More Examples...](#)

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

Insert Field: `End Date` <`c>` Insert Operator: `>` `Start Date` <`c`

Functions:

- All Function Categories -
- ABS
- ACOS
- ACOSMONTHS
- AND
- ASCII
- ASN

Insert Selected Functions

ABSVolmber: Returns the absolute value of a number, a number without its sign.

Help on this formula

Check Syntax

Save Save & New Cancel

Quick Tips

- Commons & Functions

Exploit Information

Setup > Object Manager

### lease

**lease Validation Rule**

Back to lease

Key for this Page 

Validation Rule Detail	
Role Name:	lease_end_date
Error Condition Formula:	<code>End Date &lt;c&gt; Start Date &lt;c&gt;</code>
Error Message:	Your End date must be greater than start date.
Description:	Created By: Sumeet.Bhatt 4/19/2025, 9:37 AM
Modified By:	Sumeet.Bhatt 4/19/2025, 7:47 AM

Details

Fields & Relationships

Page Layouts

Lightning Record Pages

Buttons, Links, and Actions

Compact Layouts

Field Sets

Object Limits

Record Types

Related Lookup Filters

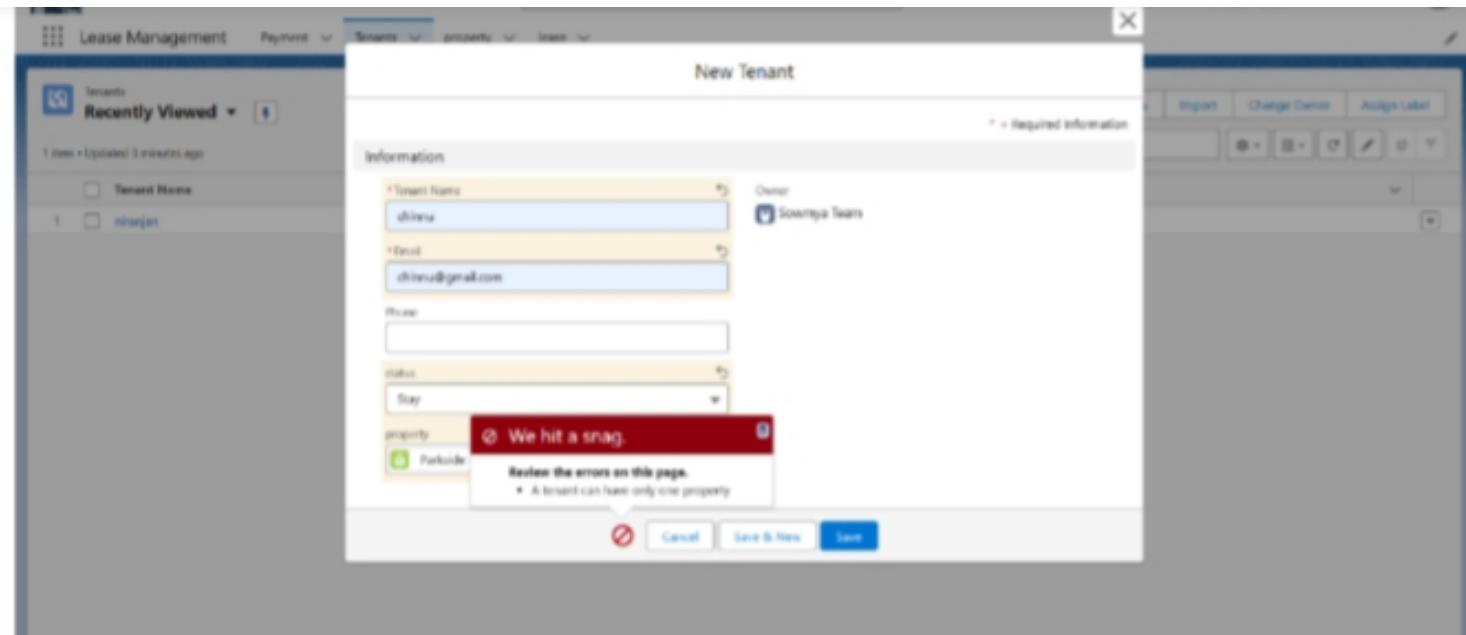
Search Layouts

List View Button Layout

Restriction Rules

Scoping Rules

- Added Apex trigger to restrict multiple tenants per property



- Scheduled monthly reminder emails using Apex class

```
1 * global class MonthlyEmailScheduler implements Scheduleable {
2 *
3 *     global void execute(SchedulableContext sc) {
4 *
5 *         Integer currentDay = Date.today().day();
6 *
7 *         if (currentDay == 10) {
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 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';
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 *     }
36 * }
```

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

The screenshot shows the Salesforce Setup interface with the 'Email' tab selected. Under 'Classic Email Templates', there is a card for the 'Leave approved' template. The card displays the template's details, including its name ('Leave approved'), subject ('Leave approved'), and previewed content.

**Email Template Detail:**

- Email Template Name: Leave approved
- Template Unique Name: Leave\_approved
- Created By: SURESHKUMAR, 09/09/2019, 1:06 AM
- Modified By: SURESHKUMAR, 09/09/2019, 1:06 AM

**Email Template Preview:**

**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.  
  
Thank you again for your prompt response.
```

The screenshot shows the Salesforce Setup interface with the 'Email' tab selected. Under 'Classic Email Templates', there is a card for the 'tenant leaving' template. The card displays the template's details, including its name ('tenant leaving'), subject ('Request for approve the leave'), and previewed content.

**Email Template Detail:**

- Email Template Name: tenant\_leaving
- Template Unique Name: tenant\_leaving
- Created By: SURESHKUMAR, 09/09/2019, 1:06 AM
- Modified By: SURESHKUMAR, 09/09/2019, 1:06 AM

**Email Template Preview:**

**Subject:** Request for approve the leave

**Main Text Preview:**

```
Dear [Tenant__c.Creativity],  
  
Please approve my leave.
```

Setup Home Object Manager

Q\_email template

v Email

Classic Email Templates

Lightning Email Templates

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

Search Setup

SETUP Classic Email Templates

Leave rejected

Preview your email template below:

Email Template Detail

Email Templates from Salesforce	United Public: Classic Email Templates
Email Template Name	Leave_rejected
Template Unique Name	Leave_rejected
Encoding	Unicode (UTF-8)
Author	Sonoma Team (23886)
Description	
Created By	Sonoma Team (400000000000000000, 400000000000000000)
Modified By	Sonoma Team (400000000000000000, 400000000000000000)

Available For Use

Last Used Date

Times Used

Help for this Page

Email Template

Send Test and Verify Merge Fields

Subject: Leave rejected

Please Next Previous

Dear {Recipient\_\_c.Name},

I hope this email finds you well. Your contract has not ended. No we can't approve your leave.  
Your leave has rejected.

Setup Home Object Manager

Q\_email template

v Email

Classic Email Templates

Lightning Email Templates

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

Search Setup

SETUP Classic Email Templates

Tenant Email

Preview your email template below:

Email Template Detail

Email Templates from Salesforce	United Public: Classic Email Templates
Email Template Name	Tenant_Email
Template Unique Name	Tenant_Email
Encoding	UTF-8 (UTF-8)
Author	Sonoma Team (23886)
Description	
Created By	Sonoma Team (400000000000000000, 400000000000000000)
Modified By	Sonoma Team (400000000000000000, 400000000000000000)

Available For Use

Last Used Date

Times Used

Help for this Page

Email Template

Send Test and Verify Merge Fields

Subject: Urgent: Monthly Rent Payment Reminder

Please Next Previous

Dear {Recipient\_\_c.Name},

I trust this email finds you well. We appreciate your continued tenancy of our property and I hope you have been comfortable in your residence.

The screenshot shows the Salesforce Setup interface with the 'Email' section selected. Under 'Email', 'Classic Email Templates' is chosen. A specific template named 'tenant payment' is highlighted. The 'Email Template Detail' section shows the following information:

- Email Template Issue Reference: United Public Classic Email Templates
- Email Template Name: tenant payment
- Template Unique Name: tenant\_payment
- Encoding: Unicode (UTF-8)
- Author: Sumeeta Srivastava (C4M000000000000000)
- Description: Created By: Sumeeta Srivastava, Modified By: Sumeeta Srivastava

The 'Email Template' section shows the template content:

```

Subject: Confirmation of successful monthly payment

Dear {Email__c.Email__r}:

We hope this email finds you well. We are writing to inform you that we have successfully received your monthly payment. Thank you for your prompt and diligent payment.

```

- Approval Process creation

For Tenant Leaving:

The screenshot shows the Salesforce Setup interface with the 'Process Automation' section selected. Under 'Approval Processes', a process named 'Tenant Approval' is selected. The 'Process Definition Detail' section shows the following information:

- Process Name: TenantApproval
- Unique Name: TenantApproval
- Description: Tenant or Vacant resource Step
- Entry Criteria: Administrator ONLY
- Record Editability: Allow Submitters to Recall Approval Requests
- Approval Assignment Email Template: Initial Submitter
- Created By: Sumeeta Srivastava, Modified By: Sumeeta Srivastava

The 'Initial Submission Actions' section shows:

Action Type	Description
Record Lock	Lock the record from being edited

The 'Approval Steps' section shows:

Action	Step Number	Name	Description	Criteria	Assigned Approver	Reject Behavior
Show Actions	1	Step 1			User:Sumeeta.Srivastava	Final Rejection

For Check for Vacant:

Salesforce Home Object Manager

Q: approval

v Data

- Main Tab: Approval Requests
- Feature Settings
- Approval Settings
- Process Automation
- Approval Processes**

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

Help for this Page

## Approval Processes

Approval Process: Tenant: check for vacant

Back to Approval Process List

**Process Definition Detail**

Process Name	check for vacant	Action	<input checked="" type="checkbox"/>
Unique Name	check_for_vacant	Next Automated Approval Determined By	
Description			
Entry Criteria	Tenant <vacant result> Leasing		
Review Eligibility	Administrator ONLY	Allow Submitters to Draft Approval Requests	<input type="checkbox"/>
Approval Assignment Email Template	Lease_Jerome	Modified By	Soumava.Jaiswal, 04/05/2015, 11:10 AM
Initial Submitters	Tenant Owner		
Created By	Soumava.Jaiswal, 04/05/2015, 11:10 AM		

**Initial Submission Actions**

Action	Type	Description
Record Lock	Lock	Lock the record from being edited
Roll / Reverse	Urgent Alert	05330_00000102.05.0000

**Approval Steps**

Action	Step Number	Name	Description	Criteria	Assigned Approver	Reject Behavior
Show Actions	1	step1			User: Soumava.Jaiswal	Final Rejection

## ● Apex Trigger

### Create an Apex Trigger

File Edit Debug Test Workspaces Help

Test Handler.apex

Code Coverage: None API Version: 44.0 Go To

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

Open

Entity Type	Category	Deleted
Classes	test	
Triggers		
Project		
Page Components		
Objects		
Static Resources		
Packages		

Open Filter: Filter the repository (\* = any string) Hide Unmanaged Packages Refresh

Logs Tests Checkpoints Query Editor View Notes Progress Problems

Name:  Last: Problem

```
File Edit Debug Test Workspace Help
Test Apex TestHandler.apex - MostRecentSchedule.apex
Code Coverage None API Version: (41 |)
trigger test on Tenant__c (before insert)
{
    if(trigger.isInsert && trigger.isBefore){
        testHandler.preventInsert(trigger.new);
    }
}

```

Logs Tests Checkpoints Query Editor View Status Progress Problems

## Create an Apex Handler class

```
Developer Console - Google Chrome
orgfarm-5d1e8052-dev-ed.development.salesforce.com/_ui/common/apex/debug/ApexCSPage
File Edit Debug Test Workspace Help
Test Apex TestHandler.apex - MostRecentSchedule.apex
Code Coverage None API Version: (41 |)
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) {
                newTenant.addError('A t');
            }
        }
    }
}
```

Open

Entity Type	Definition	Relationship
Custom Object	testHandler	testHandler
Trigger		
Page		
Page Component		
Object		
User Resource		
Profile		

Open Filter: Filter the repository (\* = any string) Hide Managed Packages Refresh

Logs Tests Checkpoints Query Editor View Status Progress Problems

Developer Console - Google Chrome

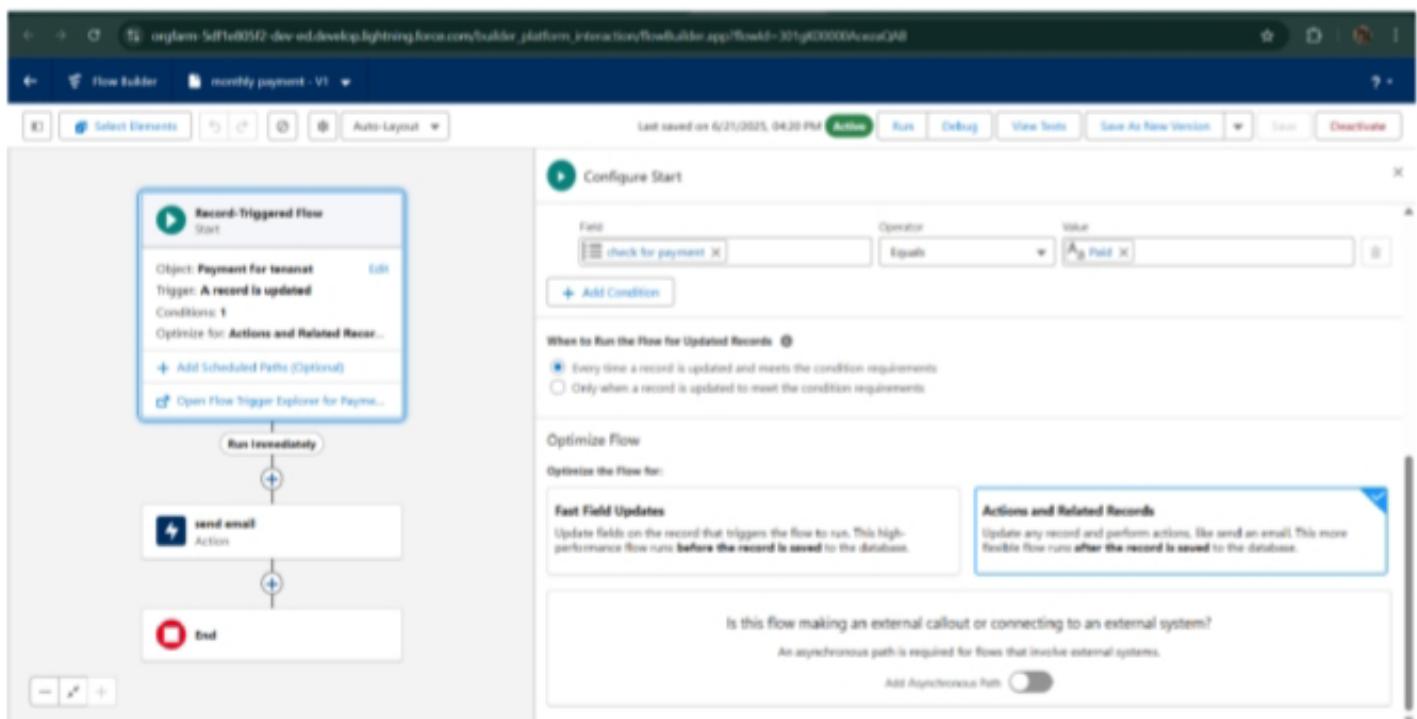
org:arm-5dfe05f2-dev-ed.develop.lightning.force.com/sf/common/specific/debug/SpecSOPage

```

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
14         for (Tenant__c newTenant : newList) {
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     }
26
27 }
```

Logs Tests Checkpoints Query Editor View Status Progress Problems

## ● FLOWS



The screenshot shows the Salesforce Flow Builder interface with a flow titled "Record-Triggered Flow" starting with "Start". The flow triggers on "Payment for tenant" when "A record is updated". It has one action step, "send email", which is triggered immediately after the start. The flow ends with a red circular "End" button.

```
graph TD; Start((Start)) --> Run[Run Immediately]; Run --> SendEmail[send email]; SendEmail --> End((End));
```

- Schedule class:  
Create an Apex Class

Developer Console - Google Chrome

cegharm-5dfe0502-dev-ed develop.my.salesforce.com/\_ui/common/apex/debug/ApexCSPage

File Edit Debug Test Workspaces Help

Toolbox: [ ] Inspector: [ ] Run: [ ] Run Related to Schedule Apex

Code Coverage: None API Version: 64

```
1 * global class MonthlyEmailScheduler implements Schedulable {
2 *
3 *     global void execute(SchedulableContext sc) {
4 *
5 *         Integer currentDay = Data.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 = [SEL
19
20     for (Tenant__c tenant : tenants
21
22
23         String recipientEmail = tenant.Email__c;
```

Open

Entity Type	Name	Namespace	Name	Class	Object
Entity Type	None		Class	Class	Referenced
Classes	testScheduler		CustomField	CustomField	Referenced
Triggers	MonthlyEmailScheduler		Object	Object	Referenced
Pages			Schema	Schema	Referenced
Page Components					
Objects					
User Resources					
Packages					

Open  Filter: Filter the repository (\* - any string)  Hide Managed Packages Refresh

Logs Tools Checkpoints Query Editor View Status Progress Problems

None Line Problem

```
12
13
14
15  public static void sendEmails() {
16
17      List<Lead__c> leads = [SELECT Id, Email__c FROM Lead__c];
18
19      for (Lead__c lead : leads) {
20
21          String recipientEmail = lead.Email__c;
22
23          String emailContent = "Hi there! 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.";
24
25          String subjectLine = "Monthly monthly rent request mail";
26
27          Messaging.SingleEmailMessage mail = new Messaging.SingleEmailMessage();
28
29          mail.setToAddresses(new String[]{recipientEmail});
30
31          mail.setSubject(subjectLine);
32
33          mail.setPlainText("Hello");
34
35          Messaging.sendEmail(true, mail);
36
37      }
38
39  }
```

## Schedule Apex class

The screenshot shows the Salesforce Setup interface with the 'Apex Classes' section selected. The 'Apex Class Detail' for 'MonthlyEmailScheduler' is displayed. The code implements the `Schedulable` interface and contains logic to check if the current day is the 1st, and if so, to call the `sendMonthlyEmails` method. The code also defines a static method `sendMonthlyEmails` which queries for leads and loops through them to send emails.

```
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              sendMonthlyEmails();
9          }
10     }
11
12 }
13
14
15
16  public static void sendMonthlyEmails() {
17
18      List<Lead__c> leads = [SELECT Id, Email__c FROM Lead__c];
19
20      for (Lead__c lead : leads) {
```

The screenshot shows a tenant detail page in a CRM system. The main area displays the following information:

- Tenant Name: Aswini
- Email: aswiniamaraad13@gmail.com
- Phone: (080) 223-5567
- Status: Leasing
- Property: 1man

Owner: Sowmya Team

Created By: Sowmya Team, 6/26/2025, 6:05 AM

Last Modified By: Sowmya Team, 6/26/2025, 11:06 PM

Activity sidebar:

- Activity section with icons for New Case, New Lead, Create, Close, Change Owner, Possible View, and Submit for Approval.
- Upcoming & Ongoing section: No activities to show.
- No past activity. Past meetings and tasks marked as done show up here.

The screenshot shows the same tenant detail page after the "Submit for Approval" button was clicked. A green success message at the top indicates that the tenant has been submitted for approval.

The main area displays the same information as the first screenshot, including:

- Tenant Name: Aswini
- Email: aswiniamaraad13@gmail.com
- Phone: (080) 223-5567
- Status: Leasing
- Property: 1man

Owner: Sowmya Team

Created By: Sowmya Team, 6/26/2025, 6:05 AM

Last Modified By: Sowmya Team, 6/26/2025, 11:06 PM

Activity sidebar:

- Activity section with icons for New Case, New Lead, Create, Close, Change Owner, Possible View, and Submit for Approval.
- Upcoming & Ongoing section: No activities to show.
- No past activity. Past meetings and tasks marked as done show up here.

Lease Management Payment Tenants property Lease Assets Approval

Process Instance Step Tenant Approval Approved

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

**Details**

Approval Details

Tenant Name	Owner
Aywin	Sowmya Team

No C

Approval request for the tenant is approved (Aywin) a few seconds ago

Approval request for the tenant is rejected (Aywin) an hour ago

Approval request for the tenant is approved (Aywin) an hour ago

Approval request for the tenant is approved (Aywin) 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

New Tenant

Recently Viewed ▾

1 item • Updated 3 minutes ago

Tenant Name: niranjan

Information

\* Tenant Name: chinu

\* Email: chinu@gmail.com

Phone:

Status: Stay

Property: Parkside

**We hit a snag.**

Review the errors on this page.

\* A tenant can have only one property.

- Validation Rule checking

Lease Management

Payment ▾      Tenants ▾      property ▾      Lease ▾

User: supriya

Related Details

\* Lease Name: supriya

Owner: Sowmya Team

Start Date: 6/18/2025

End Date: 6/18/2025

No! End date must be greater than start date.

Property: Search property...

**We hit a snag.**

Review the following fields:

- \* start\_date

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

Activity

Filters: All time • All activities • All types

Recent All • Export All • View All

Upcoming & Overview

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.

The screenshot shows a Salesforce lease record detail page. The 'Details' tab is selected. A validation error message 'We hit a snag.' is displayed over the start date field, stating: 'Review the following fields: start\_date'. The start date is set to 6/26/2025 and the end date is 6/18/2025. The end date is highlighted in yellow with a warning icon. The message also says 'By j Tane, 6/26/2025, 7:38 AM'. Below the message are 'Cancel' and 'Save' buttons.

- Test flows on payment update

The screenshot shows a flow editor interface. At the top, it says 'New Builder' and 'monthly payment - V1'. The flow is titled 'Test: Payment Update' and is in 'Paused' status. On the left, there are buttons for creating new steps: 'Record Triggered Flow', 'Run Immediately', 'Send Email', and 'End'. The main area displays a flow diagram with four steps: 'Record Triggered Flow Start', 'Run Immediately', 'Send Email Action', and 'End'. To the right, there's a sidebar titled 'Test Run Details' with 'Assertion: All Details' and 'Assertion 1' which is marked as 'Passed'.

- Approval process validated through email alerts and status updates

Lease Management Payment ▾ Tenants ▾ property ▾ lease ▾ \*eSignature Approval ▾ X

**Tenant** niranjan

**Related** Details

\* Tenant Name: niranjan Owner: Sowmya Team

\* Email: niranjan123@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

**Save**

Notifications

A Approval request for the tenant is approved niranjan a few seconds ago

B Approval request for the tenant is rejected niranjan Jun 23, 2025, 4:29 PM

C Approval request for the tenant is approved niranjan Jun 23, 2025, 4:29 PM

D Approval request for the tenant is approved niranjan Jun 23, 2025, 4:34 PM

E New Guidance Center learning resource available Define Your Sales Process Learn how to guide reps through the sales process. Jun 20, 2025, 1:20 PM

Lease Management Payment ▾ Tenants ▾ property ▾ lease ▾

**Tenant** niranjan

**Approval History (5+)**

Step Name	Date	Status	Assigned To
Step 1	6/23/2025, 5:39 AM	Approved	Sowmya Team
Approval Request Submitted	6/23/2025, 3:29 AM	Submitted	Sowmya Team
Step 1	6/23/2025, 3:58 AM	Rejected	Sowmya Team
Approval Request Submitted	6/23/2025, 3:58 AM	Submitted	Sowmya Team
Step 1	6/23/2025, 3:59 AM	Approved	Sowmya Team
Approval Request Submitted	6/23/2025, 3:59 AM	Submitted	Sowmya Team

**Payment (2)**

Payment Name
Jack
Rahul

New Contact Edit New Opportunity

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

# RESULTS

## Output Screenshots

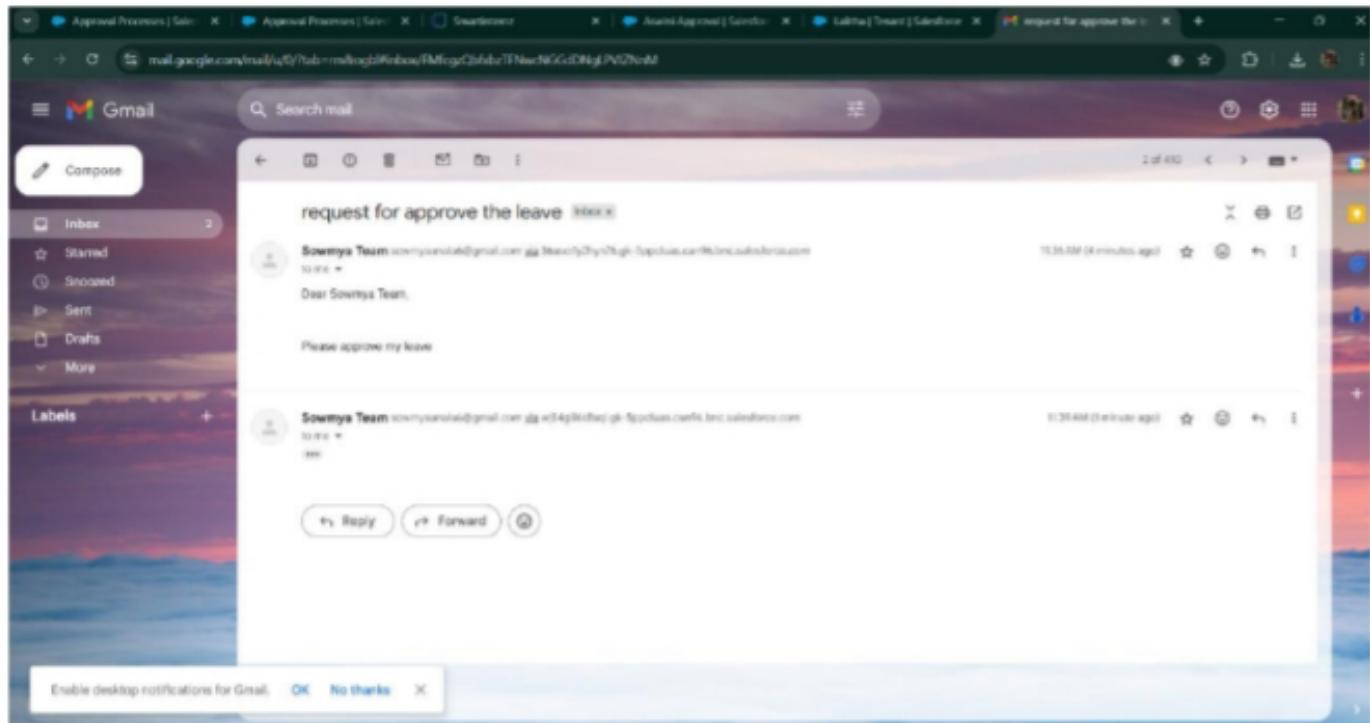
- Tabs for Property, Tenant, Lease, Payment

The screenshot shows the Salesforce Setup interface with the 'Custom Tabs' page selected. The page title is 'Custom Tabs'. It includes a sidebar with 'User Interface' and 'Rename Tabs and Labels' sections, and a search bar at the top. The main content area displays three sections: 'Custom Object Tabs', 'Web Tabs', and 'Visualforce Tabs'. The 'Custom Object Tabs' section lists four tabs: 'Basic', 'Element', 'Advanced', and 'Details', each with a 'Style' dropdown menu. The 'Web Tabs' and 'Visualforce Tabs' sections both state 'No tabs have been defined'.

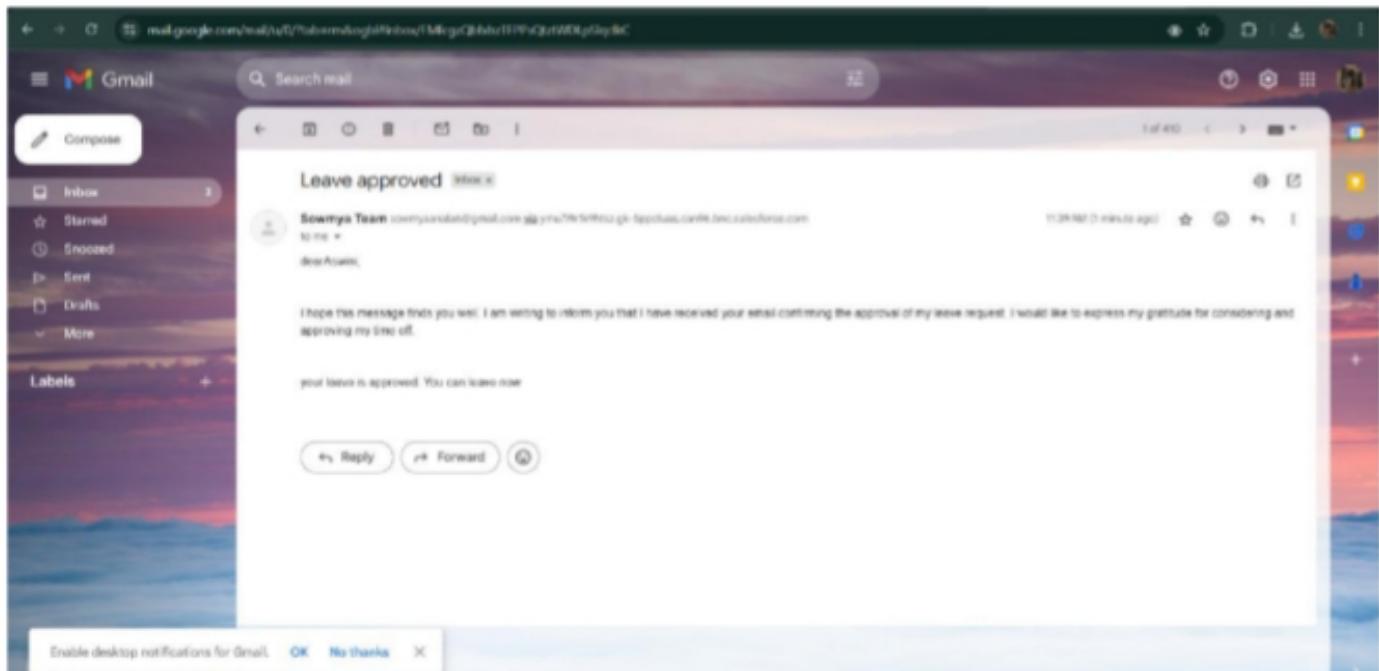
- Email alerts

The screenshot shows the Salesforce Lease Management interface with the 'Approval History' list view. The title bar includes 'Lease Management', 'Payment', 'Tenants', 'property', and 'Lease' buttons. The list view has columns for 'Step Name', 'Date', 'Status', 'Assigned To', 'Actual Approver', and 'Comments'. The data shows several approval steps, some pending and some completed with status like 'Approved', 'Rejected', or 'Leaving'. The last few rows show successful approvals with comments such as 'Approved', 'Leaving', and 'Approval Approved'.

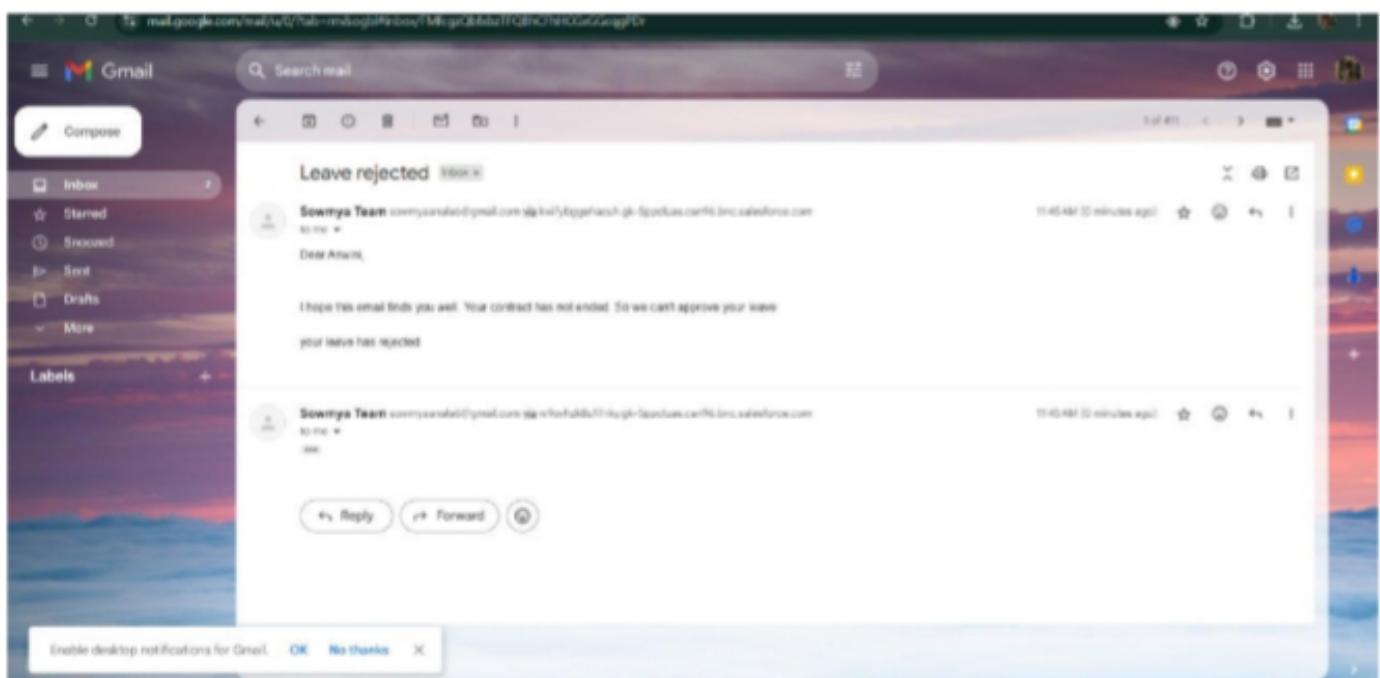
- Request for approve the leave



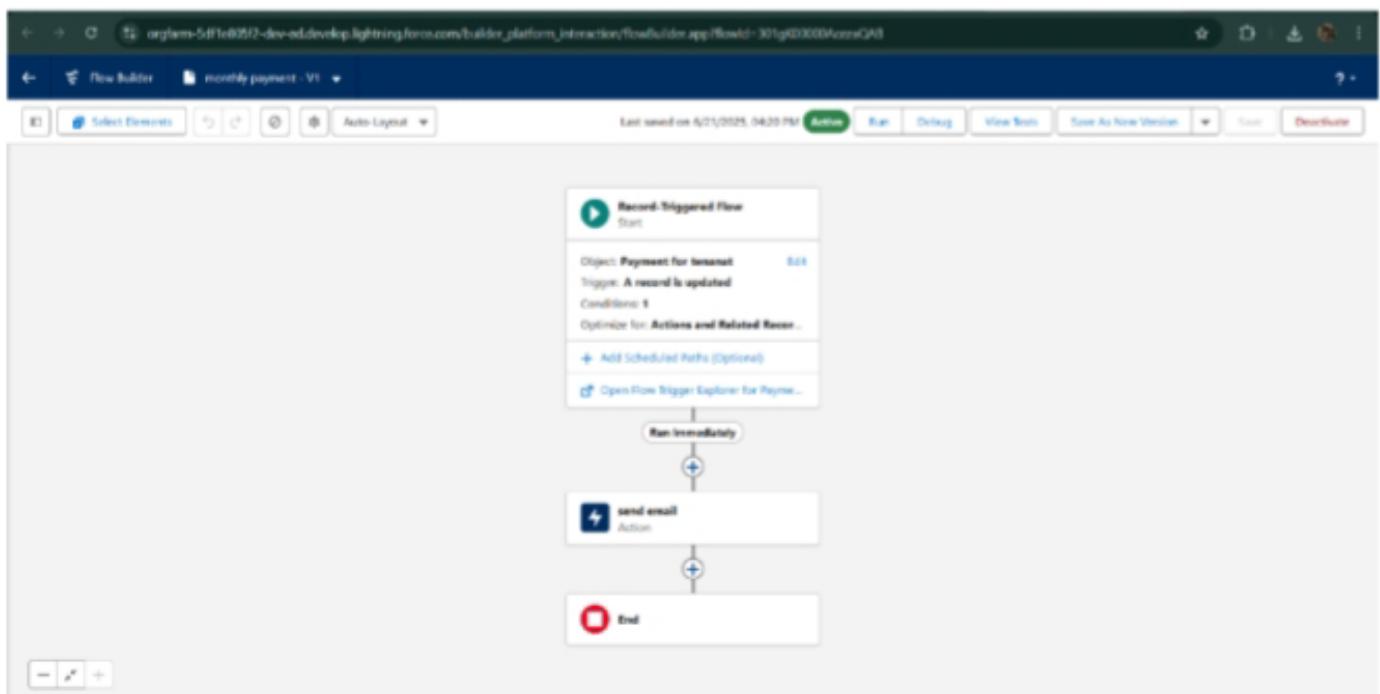
- Leave approved



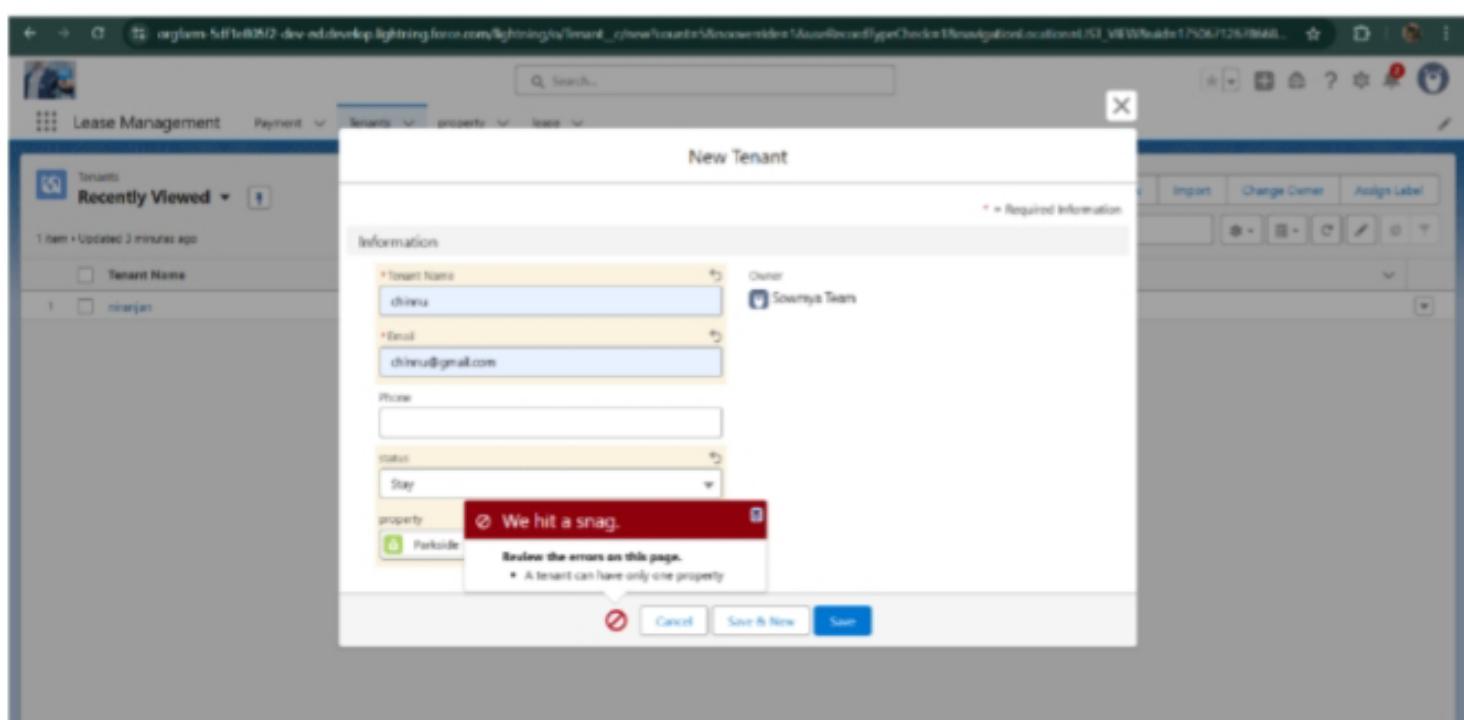
- Leave rejected



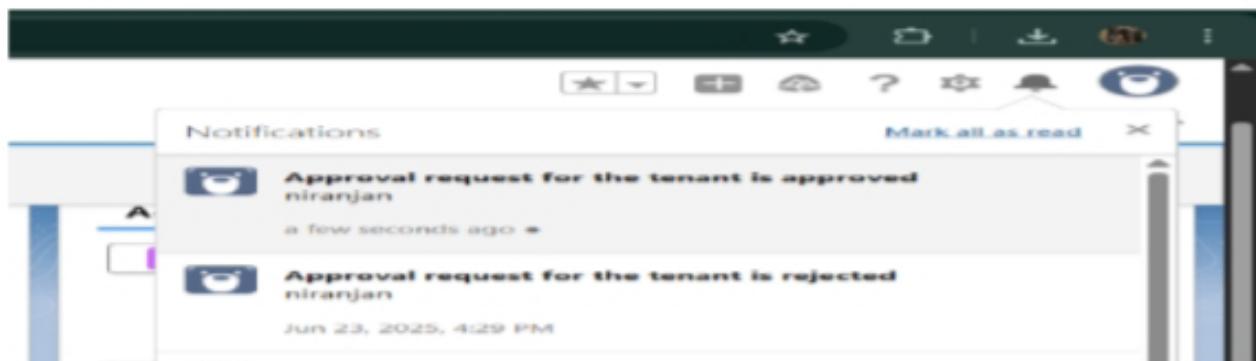
- Flow runs



- Trigger error messages



- Approval process notifications



## ADVANTAGES & DISADVANTAGES

# 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');  
            }  
        }  
    }  
}
```

#### **MothlyEmailScheduler.apxc:**

```
global class MonthlyEmailScheduler implements Schedulable {  
  
    global void execute(SchedulableContext sc) { Integer  
        currentDay = Date.today().day(); if (currentDay == 1) {  
            sendMonthlyEmails();  
        }  
    } public static void  
sendMonthlyEmails() { List<Tenant__c>  
tenants = [SELECT Id, Email__c FROM  
Tenant__c]; for (Tenant__c tenant :  
tenants) {  
            String recipientEmail = tenant.Email__c;  
            String emailContent = 'I trust this email finds you well. I am writing to remind you  
that the monthly rent is due Your timely payment ensures the smooth functioning of our  
rental arrangement and helps maintain a positive living environment for all.';  
            String emailSubject = 'Reminder: Monthly Rent Payment Due';
```

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