

LEASE MANAGEMENT

College Name: Sree Narayana Guru College

College Code:Bru36

TEAM ID: NM2025TMID26439

TEAM MEMBERS: 4

Team LeaderName: Akhil M

Email: akhilakhil38044@gmail.com

Team Member: AnithJ

Email: anithani867@gmail.com

Team Member: AL Sabith Z

Email: alsabithhh@gmail.com

Team Member: Anush EK

Email: ekanush78@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.

2.DEVELOPMENT PHASE

Creating Developer Account:

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

- **Created objects:** Property, Tenant, Lease, Payment

Free Startup PPT Template | Desdemona - SlidesCarnival | WhatsApp | Developer Edition with AI

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: Akhil Last name: M

Job title: Software developer Work email: ikhil3804@gmail.co

Company: sngc 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

Sign Me Up

lease | Salesforce

orgfarm-2e0c8b3af9-dev-ed.develop.lightning.force.com/lightning/setup/ObjectManager/01lgK0000028TOf/Details/view

lease

Setup Home Object Manager

SEARCH Type here to search

DETAILS

Description

API Name: lease_c

Custom

Singular Label: lease

Plural Label: leases

Enable Reports: ✓

Track Activities: ✓

Track Field History: ✓

Deployment Status: Deployed

Help Settings: Standard salesforce.com Help Window

EDIT Delete

iValue Infosolutions s... 06:26 PM 15-09-2025 ENG

Payment for Tenant | Salesforce

orgfarm-2e0c8b3af9-dev-ed.develop.lightning.force.com/lightning/setup/ObjectManager/01lgK0000028RRh/Details/view

Cloud icon

Search Setup

Setup Home Object Manager

SETUP > OBJECT MANAGER
Payment for Tenant

Details

Description

API Name: Payment_for_Tenant_c

Custom:

Singular Label: Payment for Tenant

Plural Label: Payment

Enable Reports: ✓

Track Activities: ✓

Track Field History: ✓

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

Edit Delete

Type here to search

iValue Infosolutions s... 06:27 PM 15-09-2025

Tenant | Salesforce

orgfarm-2e0c8b3af9-dev-ed.develop.lightning.force.com/lightning/setup/ObjectManager/01lgK0000028RGP/Details/view

Cloud icon

Search Setup

Setup Home Object Manager

SETUP > OBJECT MANAGER
Tenant

Details

Description

API Name: Tenant_c

Custom:

Singular Label: Tenant

Plural Label: Tenants

Enable Reports: ✓

Track Activities: ✓

Track Field History: ✓

Deployment Status: Deployed

Help Settings: Standard salesforce.com Help Window

Fields & Relationships

Page Layouts

Lightning Record Pages

Buttons, Links, and Actions

Compact Layouts

Field Sets

Object Limits

Record Types

Related Lookup Filters

Search Layouts

Edit Delete

Type here to search

Sunset 06:28 PM 15-09-2025

The screenshot shows the Salesforce Object Manager interface for the 'property' 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, and Search Layouts. The main content area is titled 'Details' and shows the following fields:

Description	
API Name	property__c
Custom	✓
Singular Label	property
Plural Label	property
Enable Reports	✓
Track Activities	✓
Track Field History	✓
Deployment Status	Deployed
Help Settings	Standard salesforce.com Help Window

At the bottom right of the main area are 'Edit' and 'Delete' buttons. The system navigation bar at the top includes Home, Object Manager, and a search bar labeled 'Search Setup'. The taskbar at the bottom shows various application icons and the date/time: 07:44 PM 15-09-2025.

- **Configured fields and relationships**

The screenshot shows the 'Fields & Relationships' section of the Salesforce Object Manager for the 'property' object. The left sidebar lists the same configuration tabs as the previous screenshot. The main content area is titled 'Fields & Relationships' and displays the following fields:

Field Label	Name	Type
Address	Address__c	Long Text Area(32768)
Created By	CreatedBy	Lookup(User)
Last Modified By	LastModifiedBy	Lookup(User)
Name	Name__c	Text(25)
Owner	OwnerId	Lookup(User,Group)
property Name	Name	Text(80)
sfqt	sfqt__c	Text(18)
Type	Type__c	Picklist

At the top of the main area, there are buttons for 'Quick Find', 'New', 'Deleted Fields', 'Field Dependencies', and 'Set History Tracking'. The system navigation bar and taskbar are identical to the first screenshot.

Screenshot of the Salesforce Setup Object Manager for the 'Tenant' object.

The page title is "SETUP > OBJECT MANAGER Tenant".

The left sidebar shows the "Fields & Relationships" section selected under "FIELDS & RELATIONSHIPS".

The main content area displays the "Fields & Relationships" table:

	FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Page Layouts	Created By	CreatedBy	Lookup(User)		
Lightning Record Pages	Email	Email_c	Email		
Buttons, Links, and Actions	Last Modified By	LastModifiedBy	Lookup(User)		
Compact Layouts	Owner	OwnerId	Lookup(User,Group)		
Field Sets	Phone	Phone_c	Phone		
Object Limits	property	property_c	Lookup(property)		
Record Types	status	status_c	Picklist		
Related Lookup Filters	Tenant Name	Name	Text(80)		
Search Layouts					

At the bottom of the page, there is a Windows taskbar showing the date and time as 15-09-2025 at 08:27 PM.

Screenshot of the Salesforce Setup Object Manager for the 'lease' object.

The page title is "SETUP > OBJECT MANAGER lease".

The left sidebar shows the "Fields & Relationships" section selected under "FIELDS & RELATIONSHIPS".

The main content area displays the "Fields & Relationships" table:

	FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Page Layouts	Created By	CreatedBy	Lookup(User)		
Lightning Record Pages	End date	End_date_c	Date		
Buttons, Links, and Actions	Last Modified By	LastModifiedBy	Lookup(User)		
Compact Layouts	lease Name	Name	Text(80)		
Field Sets	Owner	OwnerId	Lookup(User,Group)		
Object Limits	property	property_c	Lookup(property)		
Record Types	start date	start_date_c	Date		
Related Lookup Filters					
Search Layouts					

At the bottom of the page, there is a Windows taskbar showing the date and time as 15-09-2025 at 08:28 PM.

The screenshot shows the Salesforce Setup interface with the Object Manager for the 'Payment for Tenant' object. The 'Fields & Relationships' tab is selected. The table lists the following fields:

	Field Name	Type
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)
Payment date	Payment_date_c	Date
Payment Name	Name	Text(80)
property	property_c	Master-Detail(property)
Tenant	Tenant_c	Lookup(Tenant)

- Developed Lightning App with relevant tabs

The screenshot shows the Lightning App Builder interface with the 'App Details & Branding' tab selected. The form includes the following fields:

App Details	App Branding
* App Name Lease Management	Image Primary Color Hex Value #0070D2
* Developer Name Lease_Management	
Description Enter a description...	Org Theme Options <input type="checkbox"/> Use the app's image and color instead of the org's custom theme

Below the form, there is a preview section titled 'App Launcher Preview' showing a blue icon with 'LM' and the text 'Lease Management'.

Screenshot of the Lightning App Builder interface showing the "Navigation Items" configuration screen.

App Settings

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

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

Selected Items

- Home
- leases
- Payment
- property
- Tenants

Lightning App Builder | App Settings | Pages | Lease Management | Help

Type here to search 24°C Light rain 09:23 PM 17-09-2025

Screenshot of the Lightning App Builder interface showing the "User Profiles" configuration screen.

App Settings

User Profiles

Choose the user profiles that can access this app.

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

Selected Profiles

- System Administrator

Lightning App Builder | App Settings | Pages | Lease Management | Help

Type here to search 24°C Light rain 09:23 PM 17-09-2025



Implemented Flows for monthly rent and payment success

The screenshot shows the Salesforce Flow Builder interface. A 'Record-Triggered Flow' is selected, triggered by 'A record is updated' of the 'Payment for Tenant' object. The flow starts with a 'Run Immediately' step, followed by a 'send email' action. The flow ends. The configuration pane on the right shows the 'Configure Start' settings for the 'Record-Triggered Flow'.

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

Configure Start

Select Object: Payment for Tenant

Trigger the Flow When:

- A record is created
- A record is updated
- A record is created or updated
- A record is deleted

Set Entry Conditions

The screenshot shows the Salesforce Flow Builder interface. A 'Record-Triggered Flow' is selected, triggered by 'A record is updated' of the 'Payment for Tenant' object. The flow starts with a 'Run Immediately' step, followed by a 'send email' action. The flow ends. The configuration pane on the right shows the 'Send Email' configuration for the 'send_email' action.

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

Send Email

* Label: send_email

* API Name: send_email

Description

Use values from earlier in the flow to set the inputs for the "Send Email" core action. To use its outputs later in the flow, store them in variables.

Set Input Values

Configure Recipient Details

Recipient Address List: ...ng_Payment_for_Tenant__c > Tenant > Email



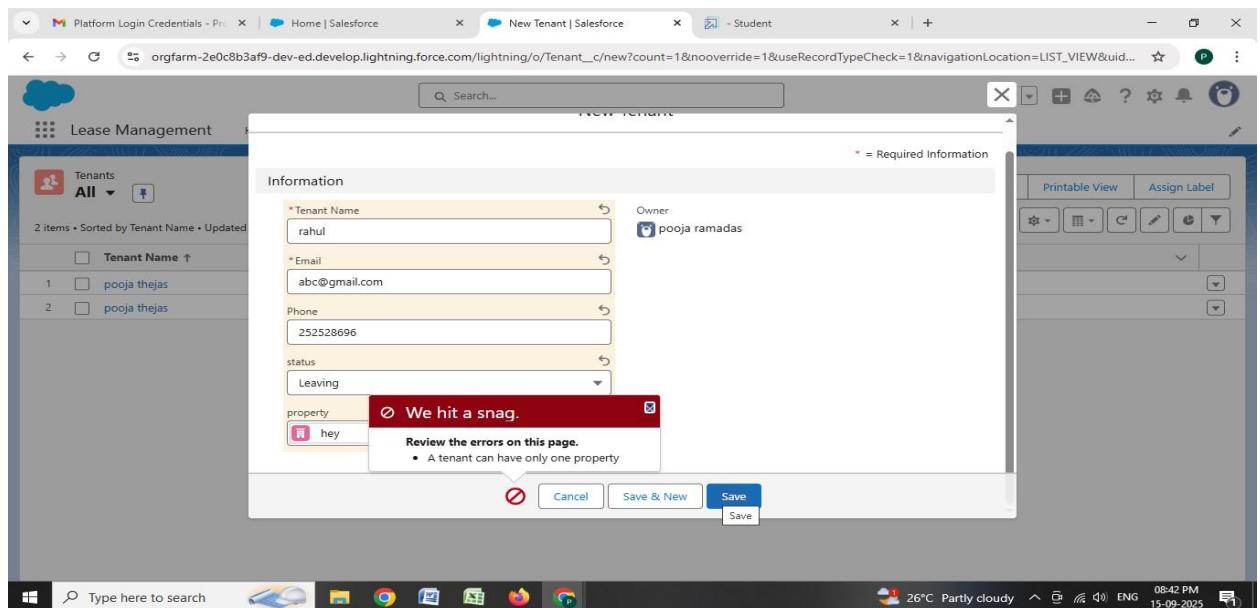
To create a validation rule to a Lease Object

The screenshot shows the 'Validation Rule Edit' screen for the 'lease' object in the Salesforce setup. The 'Rule Name' is set to 'lease_end_date'. The 'Active' checkbox is checked. The 'Error Condition Formula' is defined as 'End_date_c > start_date_c'. A tooltip for 'More Examples...' indicates that it displays an error if the discount is more than 30%. The 'Functions' sidebar lists various mathematical and string functions like ABS, ACOS, ADDMONTHS, AND, ASCII, ASIN, etc.

The screenshot shows the 'Validation Rule Detail' screen for the 'lease_end_date' rule. The rule is active and defines the formula 'End_date_c > start_date_c' with the error message 'Your End date must be greater than start date'. The rule was created by 'pooja ramadas' on 9/10/2025 at 6:48 AM and modified by 'pooja ramadas' on 9/17/2025 at 8:57 AM. The 'Error Location' is set to 'start date'.



Added Apex trigger to restrict multiple tenants per property



- Scheduled monthly reminder emails using Apex class

```
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
18     List<Tenant__c> tenants = [SELECT Id, Email__c FROM Tenant__c];
19
20
21
22
23
24     for (Tenant__c tenant : tenants) {
25
26         String recipientEmail = tenant.Email__c;
27
28         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.';
29
30         String emailSubject = 'Reminder: Monthly Rent Payment Due';
31
32
33         Messaging.SingleEmailMessage email = new Messaging.SingleEmailMessage();
34
35         email.setToAddresses(new String[]{recipientEmail});
36
37         email.setSubject(emailSubject);
38
39         email.setPlainTextBody(emailContent);
40
41
42
43 }
```



Built and tested email templates for tenant leaving, leave approval, rejected, monthly payment, success payment

The screenshot shows the Salesforce Setup interface with the following details:

Classic Email Templates Detail:

Email Templates from Salesforce	Unfiled Public Classic Email Templates
Email Template Name	tenant_leaving
Template Unique Name	tenant_leaving
Encoding	Unicode (UTF-8)
Author	pooja ramadas [Change]
Description	
Created By	pooja ramadas, 9/10/2025, 6:57 AM
Modified By	pooja ramadas, 9/10/2025, 6:57 AM

Email Template:

Subject: request for approve the leave

Plain Text Preview:

```
Dear {!Tenant__c.CreatedBy},  
Please approve my leave
```

System Navigation:

- Platform Login Credentials - Pr... (Minimized)
- Home | Salesforce
- Classic Email Templates | Salesf... (Active)
- Student

Setup Sidebar:

- Search bar: Q Search Setup
- Cloud icon
- Setup
- Home
- Object Manager

Left Sidebar:

- Email
 - Apex Exception Email
 - Authorized Email Domains
 - Classic Email Templates** (Selected)
 - Classic Letterheads
 - Compliance BCC Email
 - DKIM Keys
 - Delete Attachments Sent as Links
 - Deliverability
 - Email Address Internationalization
 - Email Attachments
 - Email Delivery Settings** (Selected)
 - Email Domain Filters
 - Email Relays

Taskbar:

- Windows Start button
- Type here to search
- Icons for File Explorer, Task View, Edge, Google Chrome, Microsoft Edge, Firefox, and Google Photos
- System tray icons: Rolling Stones, Billboard, Battery, Signal, ENG, 08:32 PM, 15-09-2025

Platform Login Credentials - Pr... X | Home | Salesforce X | Classic Email Templates | Salesfo... X | - Student X | +

orgfarm-2e0c8b3af9-dev-ed.develop.lightning.force.com/lightning/setup/CommunicationTemplatesEmail/page?address=%2F00XgK0000068AV0%3Fsetupid%3DCommunication&tab=

Search Setup

Setup Home Object Manager

email

v Email

- Apex Exception Email
- Authorized Email Domains
- Classic Email Templates
- Classic Letterheads
- Compliance BCC Email
- DKIM Keys
- Delete Attachments Sent as Links
- Deliverability
- Email Address
- Internationalization
- Email Attachments
- v Email Delivery Settings
- Email Domain Filters

SETUP Classic Email Templates

Email Template Detail

Email Templates from Salesforce	Unfiled Public Classic Email Templates
Email Template Name	Leave approved
Template Unique Name	Leave_approved
Encoding	Unicode (UTF-8)
Author	pooja ramadas [Change]
Description	
Created By	pooja ramadas, 9/10/2025, 6:59 AM
Modified By	pooja ramadas, 9/10/2025, 6:59 AM

Available For Use ✓

Last Used Date Times Used

Email Template

Send Test and Verify Merge Fields

Subject: Leave approved

Plain 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

Type here to search

Rolling Stones, Billbo... 08:32 PM 15-09-2025

Email Template Detail

Email Templates from Salesforce	Unfiled Public Classic Email Templates	Available For Use <input checked="" type="checkbox"/>	
Email Template Name	Leave rejected	Last Used Date	
Template Unique Name	Leave_rejected	Times Used	
Encoding	Unicode (UTF-8)		
Author	pooja ramadas (Change)		
Description			
Created By	pooja ramadas 9/10/2025, 7:00 AM	Modified By	pooja ramadas 9/10/2025, 7:00 AM

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

Email Template Detail

Email Templates from Salesforce	Unfiled Public Classic Email Templates	Available For Use <input checked="" type="checkbox"/>	
Email Template Name	Tenant Email	Last Used Date	
Template Unique Name	Tenant_Email	Times Used	
Encoding	Unicode (UTF-8)		
Author	pooja ramadas (Change)		
Description			
Created By	pooja ramadas 9/10/2025, 7:01 AM	Modified By	pooja ramadas 9/10/2025, 7:01 AM

Email Template

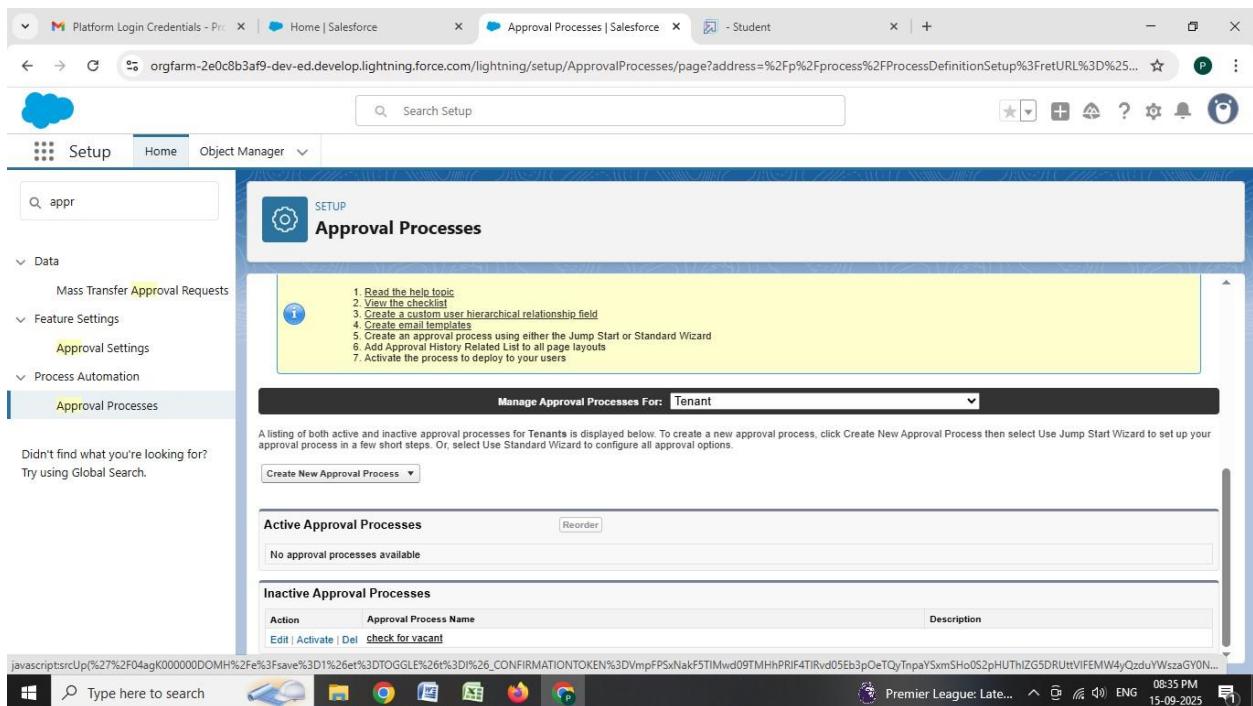
Subject: Urgent: Monthly Rent Payment Reminder

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.

This communication is a friendly reminder regarding your monthly rent payment, which is currently outstanding. As outlined in our rental agreement, the payment is due . To ensure the smooth operation of our property management and to avoid any inconvenience, we kindly request that you make arrangements to pay the outstanding amount as soon as possible. Your earliest convenience.

- For Check for Vacant:



- **Apex Trigger**

```

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

```

Create an Apex Handler class

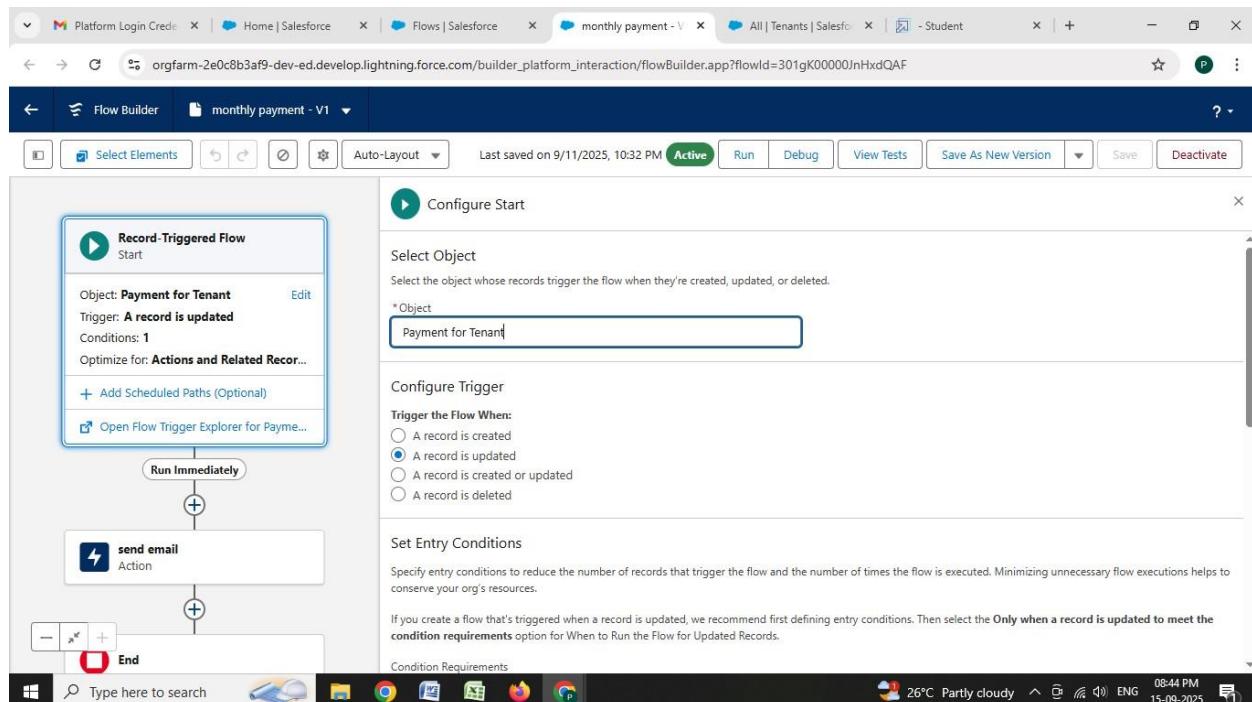
Developer Console - Google Chrome
 orgfarm-2e0c8b3af9-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage

```

  File • Edit • Debug • Test • Workspace • Help • 
  test.apex [ TestHandler.apex ]
  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 *
  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 *         newTenantaddError('A tenant can have only one property');
  20 *
  21     }
  22 *
  23   }
  24 *
  25 }
  26 *
  27 }
  28 }

  Logs Tests Checkpoints Query Editor View State Progress Problems
  User Application Operation Time Status Read Scan
  Filter Click here to filter the log list
  Type here to search 26°C Partly cloudy 08:41 PM 15-09-2025 ENG
  
```

• FLOWS



Flow Builder monthly payment - V1

Last saved on 9/11/2025, 10:32 PM Active Run Debug View Tests Save As New Version Save Deactivate

Record-Triggered Flow Start

Object: Payment for Tenant Edit
Trigger: A record is updated
Conditions: 1
Optimize for: Actions and Related Recor...

+ Add Scheduled Paths (Optional)
Open Flow Trigger Explorer for Payne...

Run Immediately

send email Action

End

Configure Start

If you create a flow that's triggered when a record is updated, we recommend first defining entry conditions. Then select the **Only when a record is updated to meet the condition requirements** option for When to Run the Flow for Updated Records.

Condition Requirements: All Conditions Are Met (AND)

Field	Operator	Value
check for payment	Equals	A_a Paid

+ Add Condition

When to Run the Flow for Updated Records:

- Every time a record is updated and meets the condition requirements
- Only when a record is updated to meet the condition requirements

Optimize Flow

Optimize the Flow for:

- Fast Field Updates
- Actions and Related Records

Flow Builder monthly payment - V1

Last saved on 9/11/2025, 10:32 PM Active Run Debug View Tests Save As New Version Save Deactivate

Record-Triggered Flow Start

Object: Payment for Tenant Edit
Trigger: A record is updated
Conditions: 1
Optimize for: Actions and Related Recor...

+ Add Scheduled Paths (Optional)
Open Flow Trigger Explorer for Payne...

Run Immediately

send email Action

End

Send Email

*Label: send email
*API Name: send_email

Description

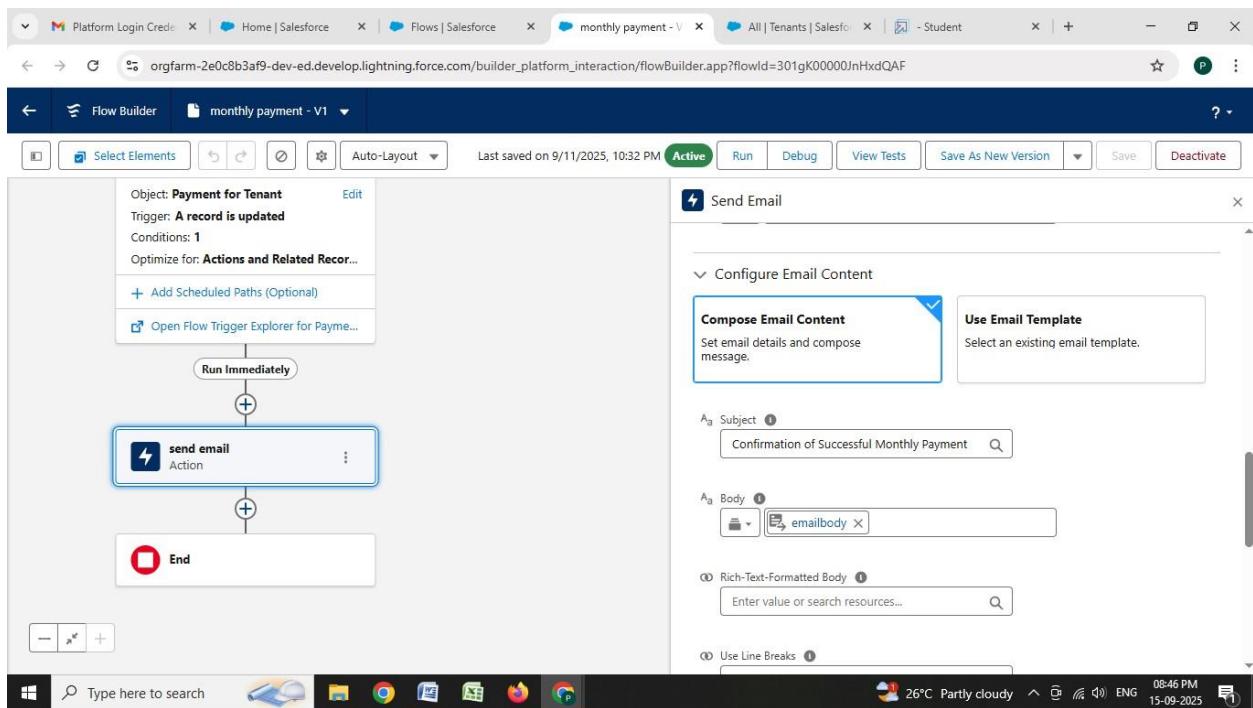
Use values from earlier in the flow to set the inputs for the "Send Email" core action. To use its outputs later in the flow, store them in variables.

Set Input Values

Configure Recipient Details

A_a Recipient Address List: A_a ...ng Payment_for_Tenant_c > Tenant > Email

A_a Recipient Address Collection:



Schedule class:

- Create an Apex Class

```

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 *
19 *         List<Tenant__c> tenants = [SELECT Id, Email__c FROM Tenant__c];
20 *
21 *
22 *         for (Tenant__c tenant : tenants) {
23 *
24 *             String recipientEmail = tenant.Email__c;
25 *
26 *             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.';
27 *
28 *             String emailSubject = 'Reminder: Monthly Rent Payment Due';
29 *
30 *
31 *             Messaging.SingleEmailMessage email = new Messaging.SingleEmailMessage();
32 *
33 *             email.setToAddresses(new String[]{recipientEmail});
34 *
35 *             email.setSubject(emailSubject);
36 *
37 *             email.setPlainTextBody(emailContent);
38 *
39 *         }
40 *
41 *     }
42 *
43 *
44 */

```

- Schedule Apex class

Screenshot of the Salesforce Setup Apex Classes page.

Apex Classes

Apex Code is an object oriented programming language that allows developers to develop on-demand business applications on the Lightning Platform.

Percent of Apex Used: 0.03%
You are currently using 1,656 characters of Apex Code (excluding comments and @isTest annotated classes) in your organization, out of an allowed limit of 6,000,000 characters. Note that the amount in use includes both Apex Classes and Triggers defined in your organization.

Estimate your organization's code coverage [\[i\]](#)
[Compile all classes](#) [\[i\]](#)

View: All [Create New View](#)

Action	Name	Namespace Prefix	Api Version	Status	Size Without Comments	Last Modified By	Has Trace Flags
Edit Del Security	MonthlyEmailScheduler		64.0	Active	1,125	pooja ramadas	9/11/2025, 10:04 AM
Edit Del Security	testHandler		64.0	Active	584	pooja ramadas	9/10/2025, 9:34 AM

Dynamic Apex Classes

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

https://orgfarm-2e0c8b3af9-dev-ed.develop.lightning.force.com/lightning/setup/ApexClasses/home

Screenshot of the Salesforce Lease Management Tenant details page.

Tenant **pooja thejas**

Related **Details**

Tenant Name: pooja thejas
Email: poojathejas05@gmail.com
Phone: (123) 456-7890
status: Stay
property: hey
Created By: [pooja ramadas](#), 9/11/2025, 9:18 AM

Owner: [pooja ramadas](#)

Last Modified By: [pooja ramadas](#), 9/11/2025, 9:53 AM

Activity

New Contact | New Opportunity | New Lead | **Submit for Approval**

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

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.

javascript:void(0)

Platform Login Credentials - X | Home | Salesforce X | Approval Processes | Salesforce X | pooja thejas | Tenant | Sales X | Student X | +

Lease Management Tenant pooja thejas

Submit for Approval

Comments
Leaving

Cancel Submit

Last Modified By: pooja ramadas, 9/11/2025, 9:53 AM

Activity

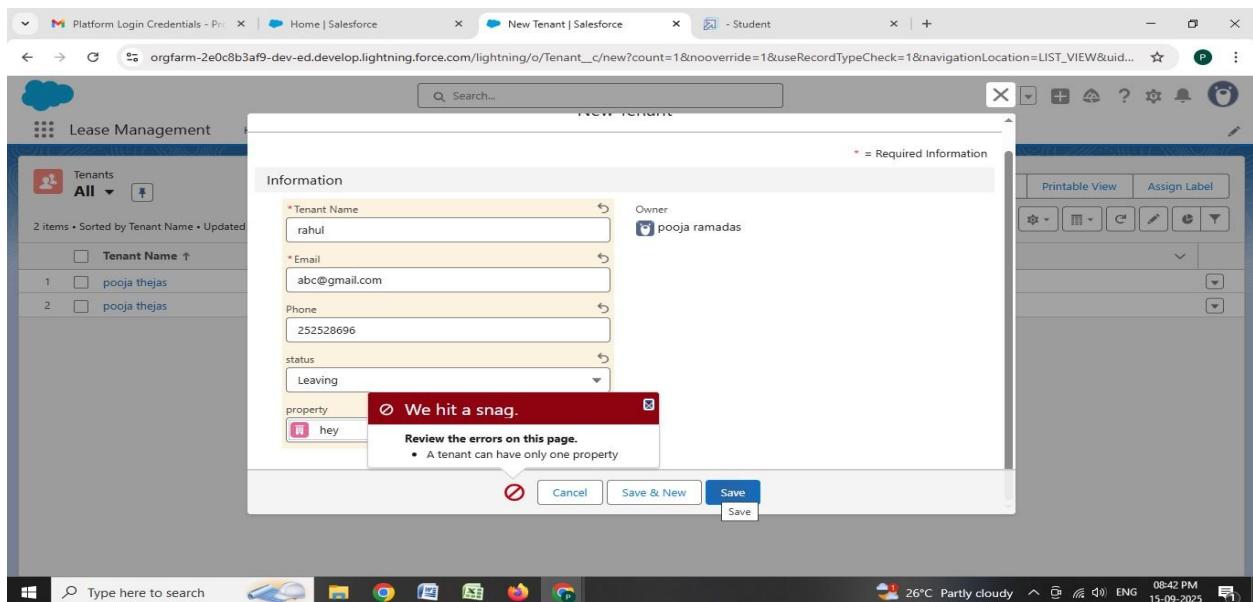
Tenant was submitted for approval.

Rain tomorrow 08:51 PM 15-09-2025

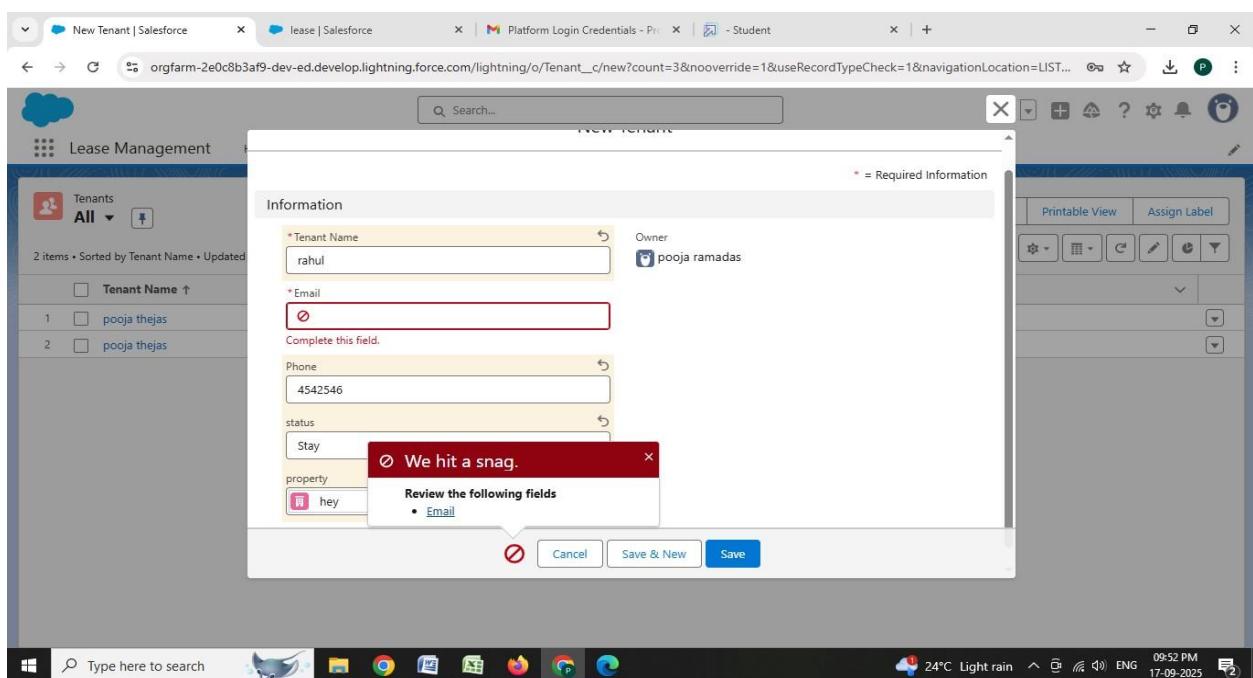
3. FUNCTIONAL AND PERFORMANCE TESTING

Performance Testing

- Trigger validation by entering duplicate tenant-property records



□ Validation Rule checking



4.RESULTS

Output Screenshots

- Tabs for Property, Tenant, Lease, Payment

The screenshot shows the Salesforce Setup interface with the 'Custom Tabs' page open. The top navigation bar includes tabs for 'Home', 'Object Manager', and 'Custom Tabs'. The left sidebar has sections for 'User Interface' and 'Rename Tabs and Labels'. The main content area is titled 'Custom Tabs' and contains a sub-section 'Custom Object Tabs' with a table showing four rows:

Action	Label	Tab Style	Description
Edit Del	Leases	Globe	
Edit Del	Payment	Shopping Cart	
Edit Del	Property	Building	
Edit Del	Tenants	People	

Below this is a section for 'Web Tabs' which states 'No Web Tabs have been defined'.

Request for approve the leave

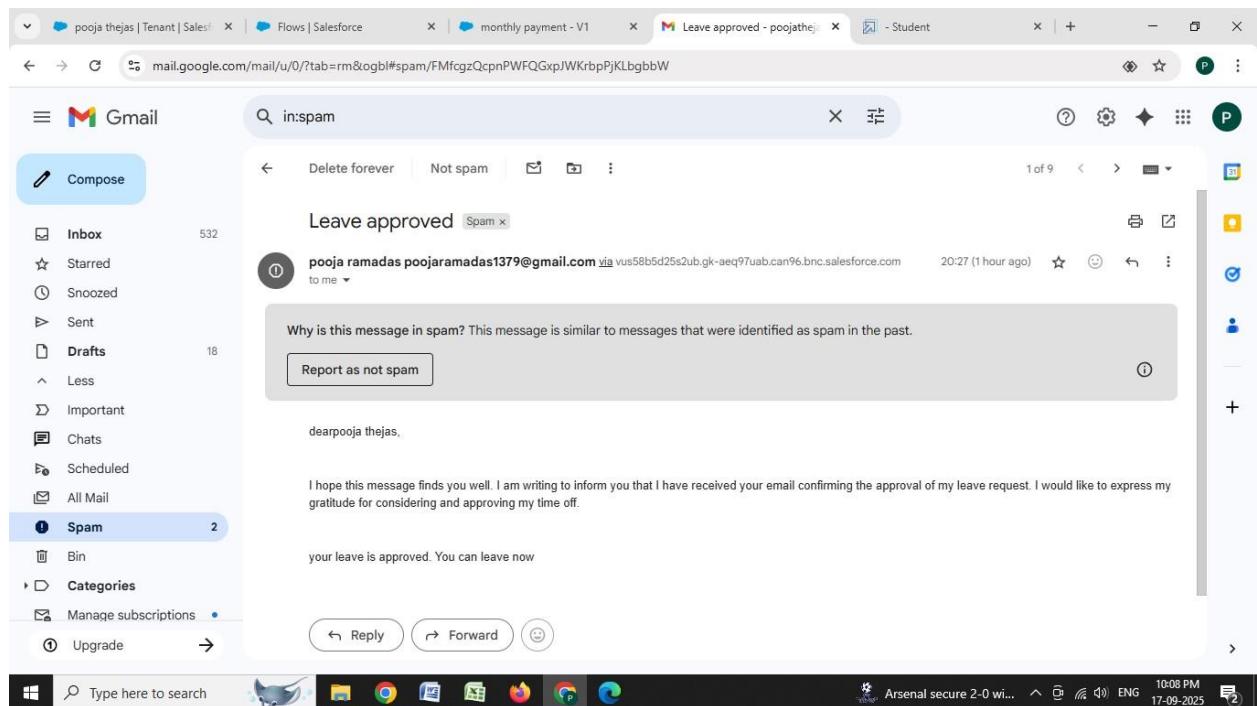
The screenshot shows a Gmail inbox with a single message from 'pooja ramadas' (poojaramadas1379@gmail.com) with the subject 'request for approve the leave'. The message is marked as spam. The message content is as follows:

Why is this message in spam? This message is similar to messages that were identified as spam in the past.
Report as not spam

Dear pooja ramadas,
Please approve my leave

At the bottom of the message are buttons for 'Reply', 'Forward', and a smiley face icon.

Leave approved



5. APPENDIX

• Source Code:

Provided in Apex Classes and Triggers

Test.apxt

Trigger Code:

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

testHandler.apxc

```
Apex logic: 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)) {
                newTenantaddError('A tenant can have only one property');
            }
        }
    }
}
```

MonthlyEmailScheduler.apxc

```
Apex logic: 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});
    }
}

```

6. 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.