

LEASE MANAGEMENT

College Name: KPR College of Arts Science and Research

College Code: bruaz

TEAM ID: NM2025TMID21467

TEAM MEMBERS:

Team Leader Name: RAGUL R

Email: 23bai040@kprcas.ac.in

Team Member 1: JAIGUKANESH J

Email: 23bai015@kprcas.ac.in

Team Member 2: RANJITH V

Email: 23bai042@kprcas.ac.in

Team Member 3: SUHITHA S

Email: 23bai053@kprcas.ac.in

1.INTRODUCTION

1.1 Project Overview

The Lease Management System is a Salesforce-based application designed to streamline the processes associated with leasing real estate properties. It handles tenant management, lease contracts, payments, and communication with automation features such as flows, approval processes, and email alerts.



1.2 Purpose

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

DEVELOPMENT PHASE

Creating Developer Account:

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

The screenshot shows a web browser window with the URL `salesforce.com/form/developer-signup/?d=pb`. The page has a dark blue header with the Salesforce logo and the text "Build enterprise-quality apps fast and get hands-on with Agentforce and Data Cloud." Below this, it says "Sign up for your Developer Edition." and lists several benefits: "Build apps fast with drag-and-drop tools", "Go further with Apex code", "Build AI agents with Agentforce", "Harmonize your data with Data Cloud", "Ground Agentforce with structured and unstructured data", and "Integrate with anything using APIs".

The main content area is a white form titled "Sign up for your Developer Edition". It includes a sub-header "A free Salesforce Platform environment with Agentforce and Data Cloud". The form fields are: "First name" (Suhitha), "Last name" (Subramaniam), "Job title" (DEVELOPER), "Work email" (23bai053@kprcas.ac), "Company" (KPRCAS), and "Country/Region" (India). Below these fields, there is a checkbox for "I agree to the Main Services Agreement - Developer Services and Salesforce Program Agreement" which is checked. A note states: "Your org may be provisioned on or migrated to Hyperforce, Salesforce's public cloud infrastructure." Below the checkbox, there is a link to the "Privacy Statement". At the bottom of the form, there is a checkbox for "I'm not a robot" and a reCAPTCHA logo.

The browser's taskbar at the bottom shows the Windows logo, a search bar, and various application icons. The system tray on the right shows the date and time as "10:15 18-09-2025".

➤ Created objects: Property, Tenant, Lease, Payment

Student x Recently Viewed | Tenants | School x Payment for tenant | Salesforce x

orgflam-26892a0870-dev-ed.develop.lightning.force.com/lightning/setup/ObjectManager/011g.000002531y/Details/view

Setup Home Object Manager

SETUP > OBJECT MANAGER

Payment for tenant

Details Edit Delete

Fields & Relationships	Description	Enable Reports
Page Layouts	API Name: Payment_for_tenant__c	<input checked="" type="checkbox"/>
Lightning Record Pages	Custom	<input checked="" type="checkbox"/>
Buttons, Links, and Actions	Custom	<input checked="" type="checkbox"/>
Compact Layouts	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Field Sets	Singular Label: Payment for tenant	<input checked="" type="checkbox"/>
Object Limits	Plural Label: Payment	<input checked="" type="checkbox"/>
Record Types	Deployment Status: Deployed	<input checked="" type="checkbox"/>
Related Lookup Filters	Help Settings: Standard salesforce.com Help Window	<input checked="" type="checkbox"/>
Search Layouts		
List View Buttons Layout		
Restriction Rules		
Scoping Rules		
Object Access		
Triggers		

Student x Recently Viewed | Tenants | School x property | Salesforce x

orgflam-26892a0870-dev-ed.develop.lightning.force.com/lightning/setup/ObjectManager/011g.000002531y/Details/view

Setup Home Object Manager

SETUP > OBJECT MANAGER

property

Details Edit Delete

Fields & Relationships	Description	Enable Reports
Page Layouts	API Name: property__c	<input checked="" type="checkbox"/>
Lightning Record Pages	Custom	<input checked="" type="checkbox"/>
Buttons, Links, and Actions	Custom	<input checked="" type="checkbox"/>
Compact Layouts	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Field Sets	Singular Label: property	<input checked="" type="checkbox"/>
Object Limits	Plural Label: property	<input checked="" type="checkbox"/>
Record Types	Deployment Status: Deployed	<input checked="" type="checkbox"/>
Related Lookup Filters	Help Settings: Standard salesforce.com Help Window	<input checked="" type="checkbox"/>

Student x Recently Viewed | Tenants | School x lease | Salesforce x

orgflam-26892a0870-dev-ed.develop.lightning.force.com/lightning/setup/ObjectManager/011g.000002531y/Details/view

Setup Home Object Manager

SETUP > OBJECT MANAGER

lease

Details Edit Delete

Fields & Relationships	Description	Enable Reports
Page Layouts	API Name: lease__c	<input checked="" type="checkbox"/>
Lightning Record Pages	Custom	<input checked="" type="checkbox"/>
Buttons, Links, and Actions	Custom	<input checked="" type="checkbox"/>
Compact Layouts	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Field Sets	Singular Label: lease	<input checked="" type="checkbox"/>
Object Limits	Plural Label: lease	<input checked="" type="checkbox"/>
Record Types	Deployment Status: Deployed	<input checked="" type="checkbox"/>
Related Lookup Filters	Help Settings: Standard salesforce.com Help Window	<input checked="" type="checkbox"/>

Student x Recently Viewed | Tenants | School x Tenant | Salesforce x

orgflam-26892a0870-dev-ed.develop.lightning.force.com/lightning/setup/ObjectManager/011g.0000025615/Details/view

Setup Home Object Manager

SETUP > OBJECT MANAGER

Tenant

Details Edit Delete

Fields & Relationships	Description	Enable Reports
Page Layouts	API Name: Tenant__c	<input checked="" type="checkbox"/>
Lightning Record Pages	Custom	<input checked="" type="checkbox"/>
Buttons, Links, and Actions	Custom	<input checked="" type="checkbox"/>
Compact Layouts	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Field Sets	Singular Label: Tenant	<input checked="" type="checkbox"/>
Object Limits	Plural Label: Tenants	<input checked="" type="checkbox"/>
Record Types	Deployment Status: Deployed	<input checked="" type="checkbox"/>
Related Lookup Filters	Help Settings: Standard salesforce.com Help Window	<input checked="" type="checkbox"/>

➤ Configured fields and relationships

Setup > OBJECT MANAGER

Payment for tenant

Details

Fields & Relationships

8 items, Sorted by Field Label

Quick Find New Deleted Fields Field Dependencies Set History Tracking

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Amount	Amount_c	Number(18, 0)		
check for payment	check_for_payment_c	Picklist		
Created By	CreatedById	Lookup(User)		
Last Modified By	LastModifiedById	Lookup(User)		
Payment date	Payment_date_c	Date		
Payment for tenant Name	Name	Text(80)		
property	property_c	Lookup(property)		
Tenant	Tenant_c	Lookup(Tenant)		

Setup > OBJECT MANAGER

property

Details

Fields & Relationships

8 items, Sorted by Field Label

Quick Find New Deleted Fields Field Dependencies Set History Tracking

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(255)		
Owner	OwnerId	Lookup(User Group)		
property Name	Name	Text(80)		
slug	slug_c	Text(11)		

Setup > OBJECT MANAGER

Tenant

Details

Fields & Relationships

8 items, Sorted by Field Label

Quick Find New Deleted Fields Field Dependencies Set History Tracking

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Created By	CreatedById	Lookup(User)		
Email	Email_c	Email		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User Group)		
Phone	Phone_c	Phone		
property	property_c	Lookup(property)		
status	status_c	Picklist		

Setup > OBJECT MANAGER

lease

Details

Fields & Relationships

11 items, Sorted by Field Label

Quick Find New Deleted Fields Field Dependencies Set History Tracking

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Amount	Amount_c	Number(18, 0)		
check for payment	check_for_payment_c	Picklist		
Created By	CreatedById	Lookup(User)		
End date	End_date_c	Date		
Last Modified By	LastModifiedById	Lookup(User)		
lease Name	Name	Text(80)		
Owner	OwnerId	Lookup(User Group)		

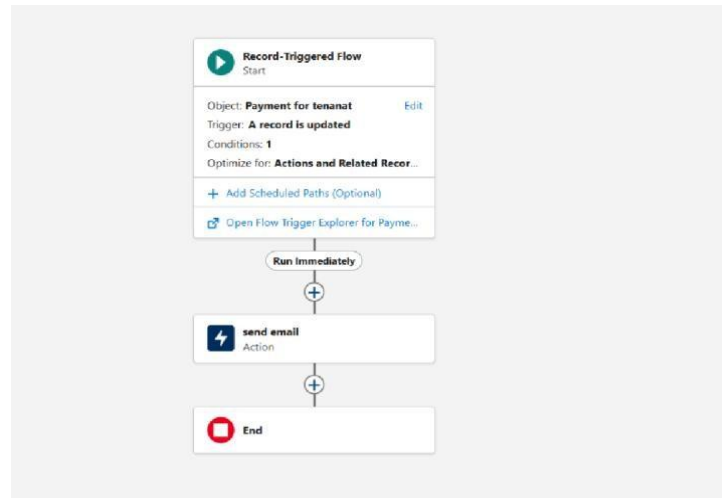
➤ Developed Lightning App with relevant tabs

The screenshot shows the 'App Details & Branding' configuration screen in the Lightning App Builder. The left sidebar lists 'App Settings' with sub-items: 'App Details & Branding' (selected), 'App Options', 'Utility Items (Desktop Only)', 'Navigation Items', and 'User Profiles'. The main content area is divided into two sections: 'App Details' and 'App Branding'. In 'App Details', the 'App Name' is 'Lease Management', the 'Developer Name' is 'Lease_Management', and the 'Description' is 'Application to efficiently handle the processes related to leasing real estate properties.'. In 'App Branding', an image is uploaded, and the 'Primary Color Hex Value' is '#0070D2'. Below this, 'Org Theme Options' are set to 'Use the app's image and color instead of the org's custom theme'. An 'App Launcher Preview' shows a card for 'Lease Management' with the description 'Application to efficiently handle the processes relate...'.

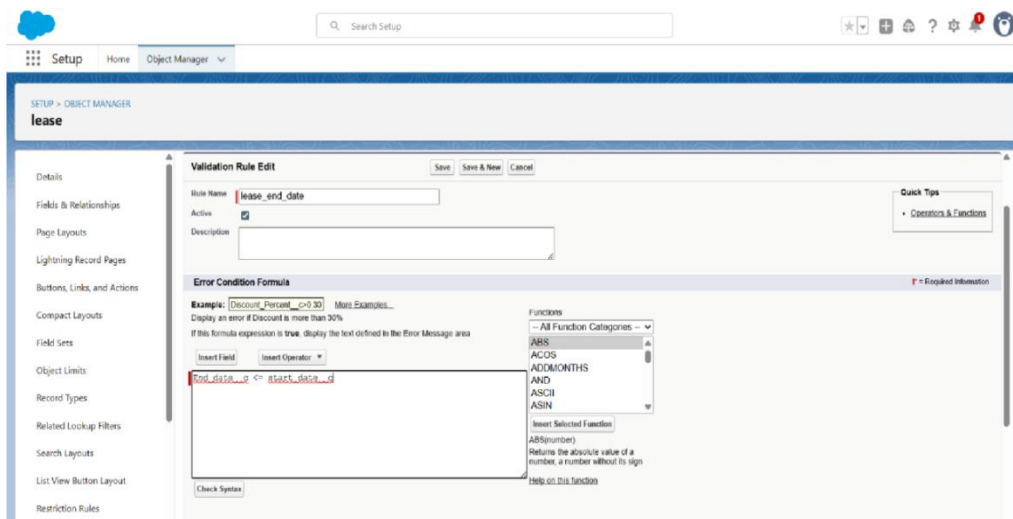
The screenshot shows the 'Navigation Items' configuration screen. The left sidebar is the same as the previous screen. The main content area is titled 'Navigation Items' and includes instructions: 'Choose the items to include in the app, and arrange the order in which they appear. Users can personalize 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.' There are two lists: 'Available Items' and 'Selected Items'. 'Available Items' includes: Accounts, Activation Targets, Activations, All Sites, Alternative Payment Methods, Analytics, App Launcher, Appointment Categories, Appointment Invitations, and Approval Requests. 'Selected Items' includes: Payment, Tenants, property, and lease. Arrows between the lists allow for moving items.

The screenshot shows the Lightning App running in a web browser. The browser address bar shows the URL: 'orgfarm-26892a0870-dev-ed.develop.lightning.force.com/lightning/o/property_c/list?filterName=__Recent'. The app's navigation bar at the top includes 'Lease Management', 'Tenants', 'property' (selected), 'Payment', and 'lease'. The main content area is titled 'property Recently Viewed' and shows a table with one item: 'property Name' with a value of 'saksha'. The table has columns for 'Name', 'Assign', 'Change Owner', and 'Assign Label'. A search bar is present above the table.

- Implemented Flows for monthly rent and payment success



- To create a validation rule to a Lease Object



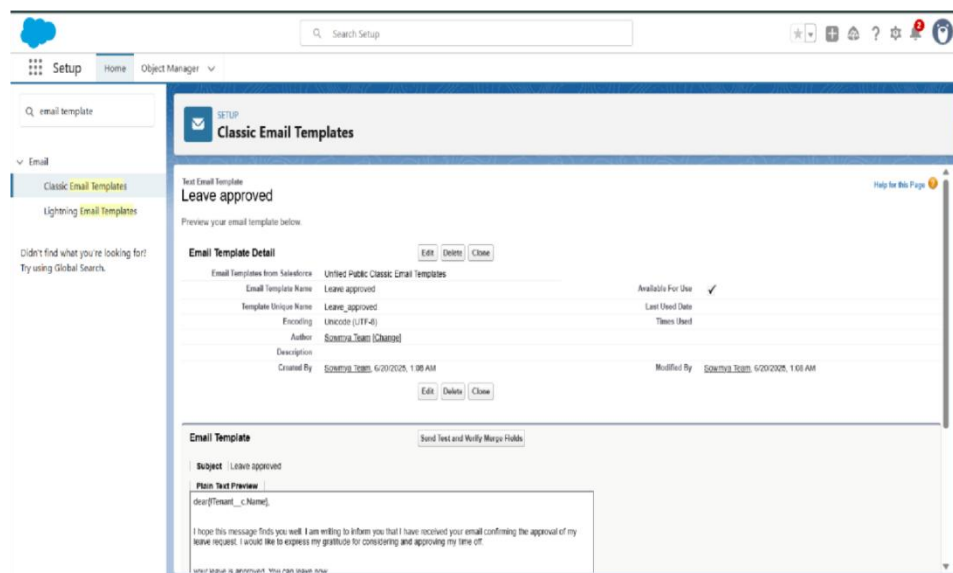
- Added Apex trigger to restrict multiple tenants per property
- Scheduled monthly reminder emails using Apex class

```

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             sendMonthlyEmails();
9         }
10    }
11
12
13 }
14
15
16 public static void sendMonthlyEmails() {
17     List<Tenant__c> tenants = [SELECT Id, Email_c FROM Tenant__c];
18     for (Tenant__c tenant : tenants) {
19         String recipientEmail = tenant.Email_c;
20         String emailContent = 'I trust this email finds you well. I am writing to remind you that the monthly rent is due. Your timely payment ensures the smooth functioning of our rental arrangement and helps maintain';
21         String emailSubject = 'Reminder: Monthly Rent Payment Due';
22         Messaging.SingleEmailMessage email = new Messaging.SingleEmailMessage();
23         email.setToAddresses(new String[]{recipientEmail});
24         email.setSubject(emailSubject);
25         email.setPlainTextBody(emailContent);
26     }
27 }

```

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



Setup

Home

Object Manager

Q email template

Email

Classic Email Templates

Lightning Email Templates

Didn't find what you're looking for?

Try using Global Search.

Search Setup

Help for this Page

Classic Email Templates

tenant leaving

Preview your email template below

Email Template Detail

Email Template Name

tenant_leaving

Available For Use

✓

Template Unique Name

tenant_leaving

Last Used Date

Encoding

Unicode (UTF-8)

Times Used

Author

System Team (Changed)

Description

Created By

System Team

6/29/2025, 1:06 AM

Modified By

System Team

6/29/2025, 1:06 AM

Email Template

Send Text and Verify Merge Fields

Subject

request for approval the leave

Plain Text Preview

Dear {Tenant__c.CreatedBy},
Please approve my leave

Setup

Home

Object Manager

Search Setup

email template

Classic Email Templates

Lightning Email Templates

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

SETUP

Classic Email Templates

Test Email Template

Leave rejected

Preview your email template below.

Email Template Detail

Email Templates from Salesforce

Email Template Name

Template Unique Name

Encoding

Author

Description

Created By

Unified Public Classic Email Templates

Leave_rejected

Unicode (UTF-8)

Scotlyn, Sean (Chatter)

Scotlyn, Sean 6/20/2025, 1:11 AM

Available For Use

Last Used Date

Times Used

Modified By

✓

Scotlyn, Sean 6/20/2025, 1:11 AM

Edit

Delete

Close

Email Template

Send Text and Merge Fields

Subject

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

Setup

Home

Object Manager

Search Setup

email template

Classic Email Templates

Lightning Email Templates

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

SETUP

Classic Email Templates

Test Email Template

Tenant Email

Preview your email template below.

Email Template Detail

Email Templates from Salesforce

Email Template Name

Template Unique Name

Encoding

Author

Description

Created By

Unified Public Classic Email Templates

Tenant_Email

Unicode (UTF-8)

Scotlyn, Sean (Chatter)

Scotlyn, Sean 6/20/2025, 1:12 AM

Available For Use

Last Used Date

Times Used

Modified By

✓

Scotlyn, Sean 6/20/2025, 1:12 AM

Edit

Delete

Close

Email Template

Send Text and Merge Fields

Subject

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.

Setup

Home

Object Manager

Search Setup

email template

Classic Email Templates

Lightning Email Templates

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

SETUP

Classic Email Templates

Test Email Template

tenant payment

Preview your email template below.

Email Template Detail

Email Templates from Salesforce

Email Template Name

Template Unique Name

Encoding

Author

Description

Created By

Unified Public Classic Email Templates

tenant_payment

Unicode (UTF-8)

Scotlyn, Sean (Chatter)

Scotlyn, Sean 6/20/2025, 1:13 AM

Available For Use

Last Used Date

Times Used

Modified By

✓

Scotlyn, Sean 6/20/2025, 1:13 AM

Edit

Delete

Close

Email Template

Send Text and Merge Fields

Subject

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.

Approval Processes

Tenant: TenantApproval

Process Definition Detail

Process Name: TenantApproval

Unique Name: TenantApproval

Description: Tenant Approval

Entry Criteria: Tenant1.Status EQUALS Stay

Record Editability: Administrator ONLY

Approval Assignment Email Template: Tenant Owner

Initial Submitters: Tenant Owner

Created By: Soumya Team, 6/25/2025, 3:41 AM

Modified By: Soumya Team, 6/26/2025, 11:57 PM

Initial Submission Actions

Action	Type	Description
Record Lock	Record Lock	Lock the record from being edited

Approval Steps

Action	Step Number	Name	Description	Criteria	Assigned Approver	Reject Behavior
Step 1	1	Step 1			User: Soumya Team	Final Rejection

Approval Process creation

For Check for Vacant:

Approval Processes

Tenant: check for vacant

Process Definition Detail

Process Name: check for vacant

Unique Name: check_for_vacant

Description: check for vacant

Entry Criteria: Tenant1.Status EQUALS LEAVING

Record Editability: Administrator ONLY

Approval Assignment Email Template: LEAVE_REQUESTED

Initial Submitters: Tenant Owner

Created By: Soumya Team, 6/26/2025, 3:16 AM

Modified By: Soumya Team, 6/26/2025, 11:02 PM

Initial Submission Actions

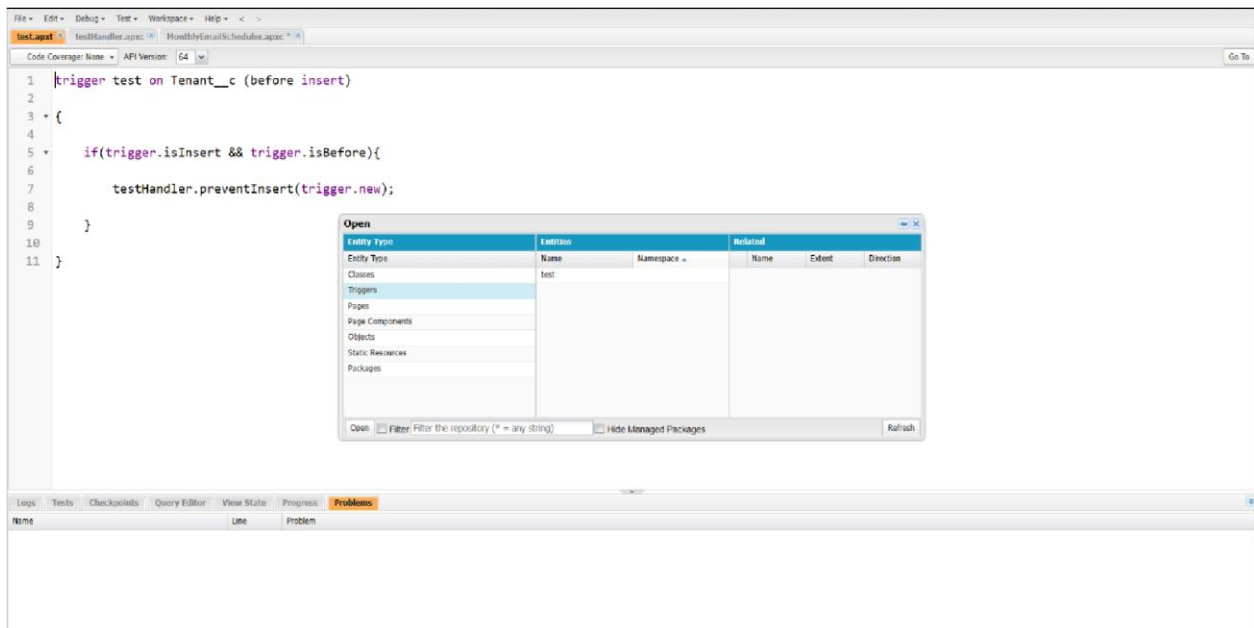
Action	Type	Description
Record Lock	Record Lock	Lock the record from being edited
Email Alert	Email Alert	SEND APPROVAL REQUEST

Approval Steps

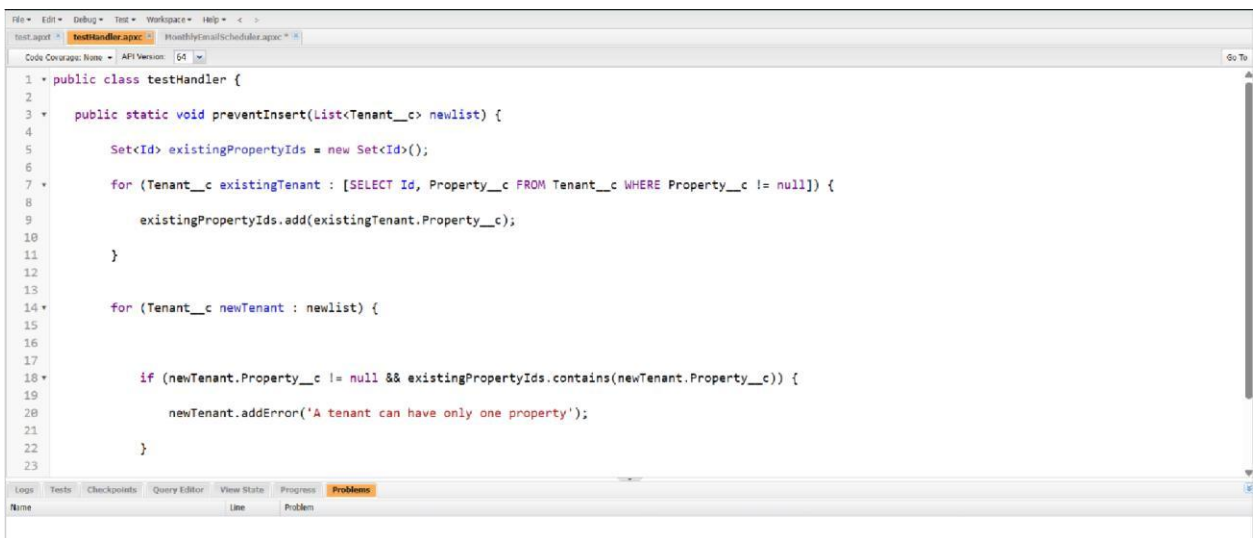
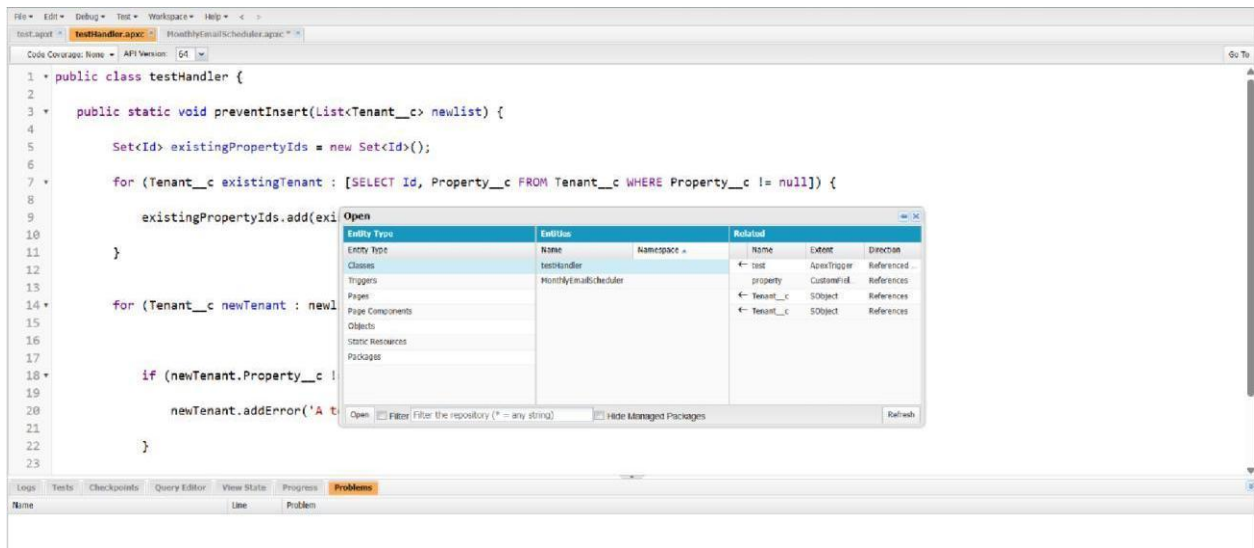
Action	Step Number	Name	Description	Criteria	Assigned Approver	Reject Behavior
step1	1	step1			User: Soumya Team	Final Rejection

Apex Trigger

Create an Apex Trigger



Create an Apex Handler class



FLAWS

Flow Builder | monthly payment - V1

Last saved on 6/21/2025, 04:20 PM **Active** Run Debug View Tests Save As New Version Save Deactivate

Select Elements Auto-Layout

Record-Triggered Flow
Start

Object: **Payment for tenant** Edit

Trigger: **A record is updated**

Conditions: 1

Optimize for: **Actions and Related Records**

+ Add Scheduled Paths (Optional)

Open Flow Trigger Explorer for Payme...

Run Immediately

send email Action

End

Configure Start

Field: check for payment X Operator: Equals Value: A3 Paid X

+ Add Condition

When to Run the Flow for Updated Records

☒ Every time a record is updated and meets the condition requirements

☐ Only when a record is updated to meet the condition requirements

Optimize Flow

Optimize the Flow for:

Fast Field Updates
Update fields on the record that triggers the flow to run. This high-performance flow runs **before the record is saved** to the database.

Actions and Related Records
Update any record and perform actions, like send an email. This more flexible flow runs **after the record is saved** to the database.

Is this flow making an external callout or connecting to an external system?
An asynchronous path is required for flows that involve external systems.

Add Asynchronous Path ☐

Flow Builder | monthly payment - V1

Last saved on 6/21/2025, 04:20 PM **Active** Run Debug View Tests Save As New Version Save Deactivate

Select Elements Auto-Layout

Record-Triggered Flow
Start

Object: **Payment for tenant** Edit

Trigger: **A record is updated**

Conditions: 1

Optimize for: **Actions and Related Records**

+ Add Scheduled Paths (Optional)

Open Flow Trigger Explorer for Payme...

Run Immediately

send email Action

End

Configure Start

Select Object
Select the object whose records trigger the flow when they're created, updated, or deleted.

* Object: Payment for tenant

Configure Trigger

Trigger the Flow When:

☐ A record is created

☒ A record is updated

☐ A record is created or updated

☐ A record is deleted

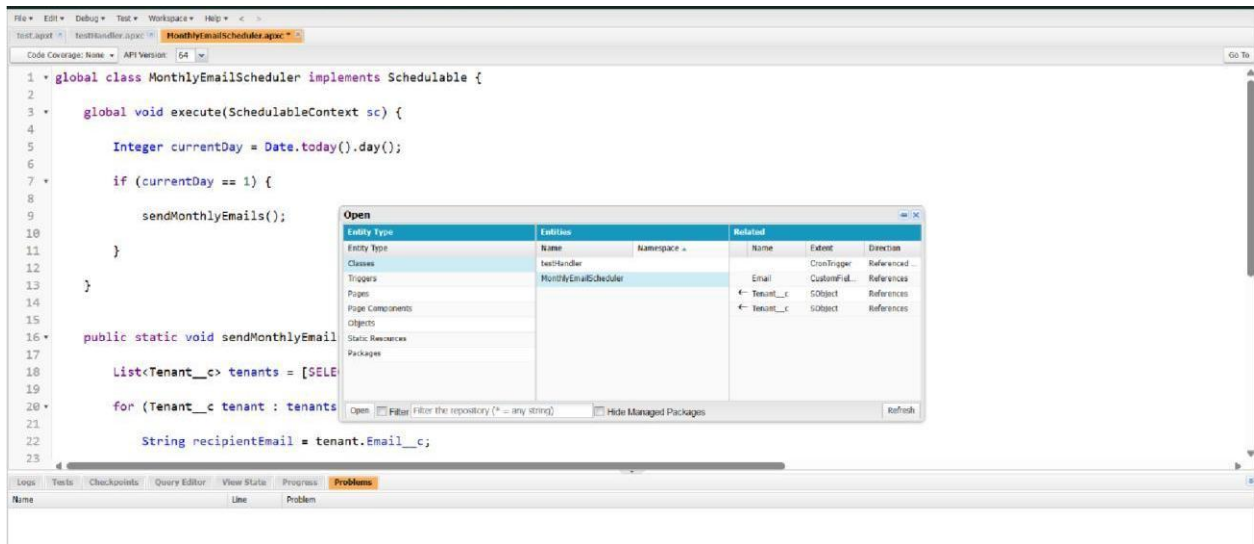
Set Entry Conditions
Specify entry conditions to reduce the number of records that trigger the flow and the number of times the flow is executed. Minimizing unnecessary flow executions helps to conserve your org's resources.

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.

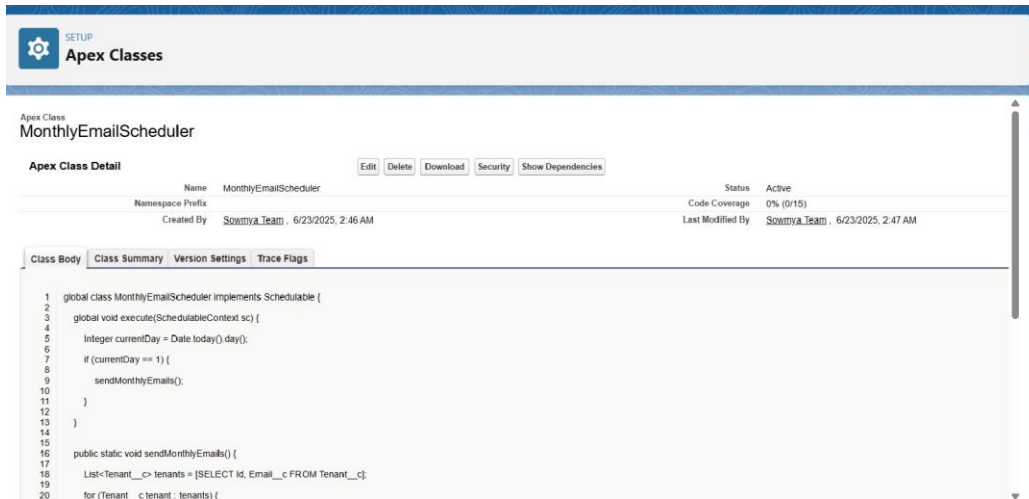
Condition Requirements: All Conditions Are Met (AND)

➤ Schedule class:

Create an Apex Class



Schedule Apex class



The screenshot displays the Salesforce 'Apex Classes' page. At the top, there's a 'SETUP' button and the title 'Apex Classes'. Below this, the class 'MonthlyEmailScheduler' is selected. The 'Apex Class Detail' section shows the class name, namespace prefix, and creation/modification details. The 'Class Body' tab is active, showing the following code:

```
1 global class MonthlyEmailScheduler implements Schedulable {
2
3     global void execute(SchedulableContext sc) {
4         Integer currentDay = Date.today().day();
5         if (currentDay == 1) {
6             sendMonthlyEmails();
7         }
8     }
9 }
10
11 public static void sendMonthlyEmails() {
12     List<Tenant__c> tenants = [SELECT Id, Email__c FROM Tenant__c];
13     for (Tenant__c tenant : tenants) {
```

The code is a global class implementing the Schedulable interface. It has a global void execute method that checks if the current day is 1. If so, it calls the sendMonthlyEmails method. The sendMonthlyEmails method is a public static void that queries for all Tenant__c records and iterates over them.

FUNCTIONAL AND PERFORMANCE TESTING

Performance Testing

- Trigger validation by entering duplicate tenant-property records
- Validation Rule checking

Lease Management

lease Properties Payment Tenants Home

NEW lease

New Contact Edit New Opportunity

Related Details

* = Required Information

*lease Name
NEW lease

Owner
Kaviya Shree V

start date
9/1/2024
Your End date must be greater than start date

End date
8/31/2025

Property
qwerty

We hit a snag.
Review the following fields
• start_date

Cancel Save

Activity

Filters: All time • All activities • All types

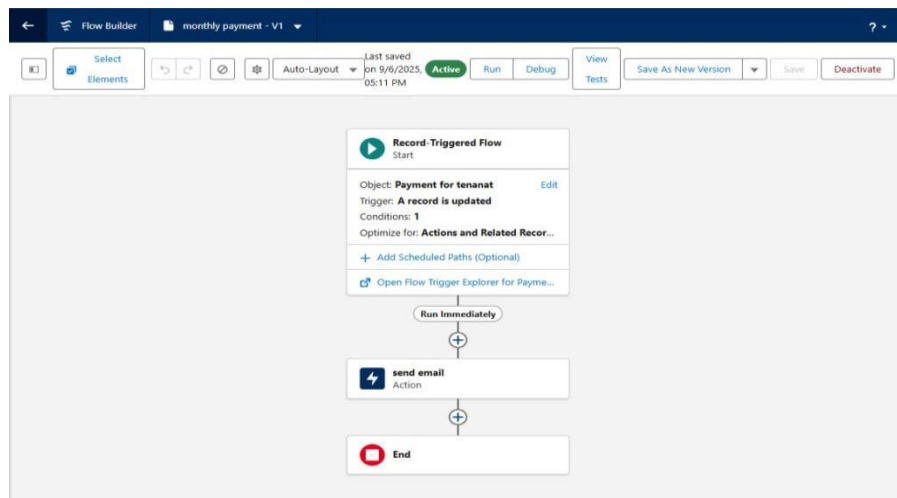
Refresh • Expand All • View All

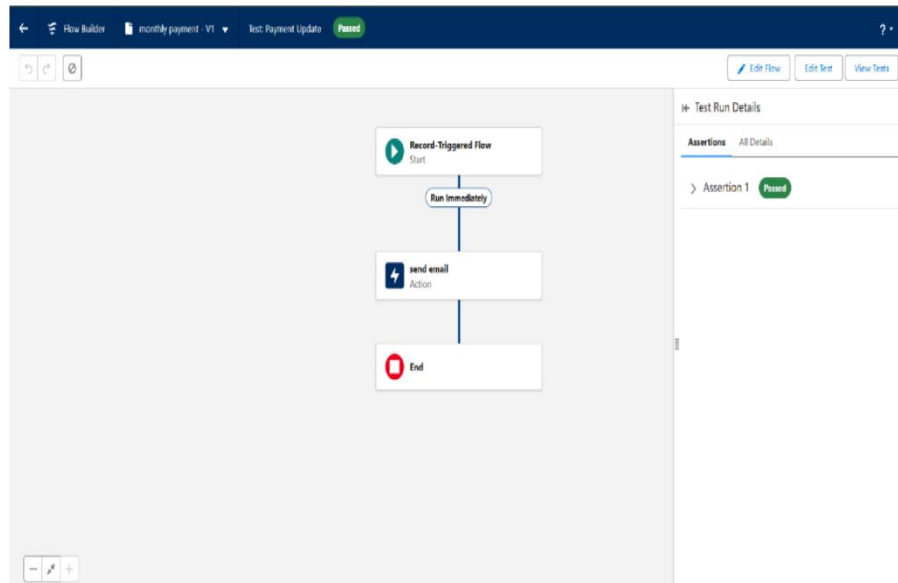
Upcoming & Overdue

No activities to show.
Get started by sending an email, scheduling a task, and more.

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

➤ Test flows on payment update





- Approval process validated through email alerts and status updates

RESULTS

Output Screenshots

- Tabs for Property, Tenant, Lease, Payment

Search Setup

Setup Home Object Manager

tabs

User Interface

Rename Tabs and Labels

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

Custom Tabs

You can create new custom tabs to extend Salesforce functionality or to build new application functionality.

Custom Object tabs look and behave like the standard tabs provided with Salesforce. Web tabs allow you to embed external web applications and content within the Salesforce window. Visualforce tabs allow you to embed Visualforce pages. Lightning Component tabs allow you to add Lightning components to the navigation menu in Lightning Experience and the mobile app. Lightning Page tabs allow you to add Lightning Pages to Lightning Experience and the mobile app.

Action	Label	Tab Style	Description
Edit Del	LEASE	Trigger	
Edit Del	Payment	Custom	
Edit Del	Properties	Homeowner	
Edit Del	Tenants	Graders	

Web Tabs

No Web Tabs have been defined

Visualforce Tabs

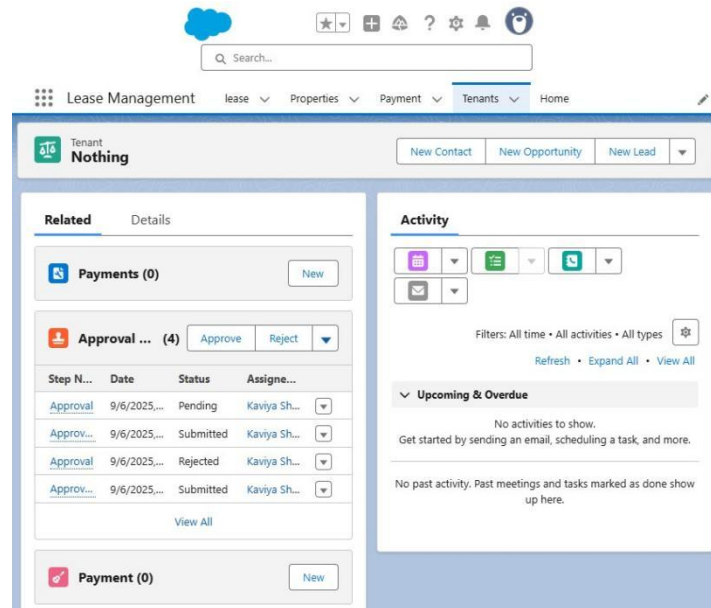
No Visualforce Tabs have been defined

Lightning Component Tabs

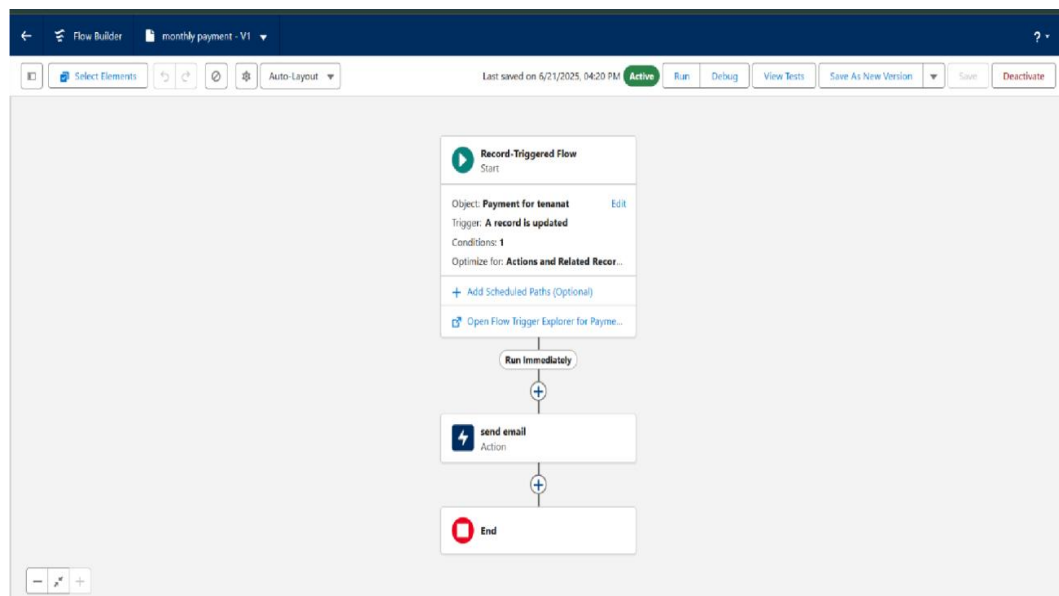
No Lightning component tabs have been defined

Lightning Page Tabs

➤ Email alerts



- Request for approve the leave
- Leave approved
- Leave rejected
 - Flow runs



- Trigger error messages

CONCLUSION

The Lease Management System successfully streamlines the operations of leasing through a structured, automated Salesforce application. It improves efficiency, communication, and data accuracy for both admins and tenants.

APPENDIX

- **Source Code:** Provided in Apex Classes and Triggers

Test.apxt: trigger test on Tenant__c

(before insert)

```
{ if (trigger.isInsert &&
trigger.isBefore)
{ testHandler.preventInsert(trigger.new);
}
}
```

testHandler.apxc:

```
public      class
```

```
testHandler { public
```

```
static      void
```

```
preventInsert(List<
```

```
Tenant__c> newList)
```

```
{
```

```
Set<Id> existingPropertyIds = new Set<Id>()
```

```
for (Tenant__c existingTenant : [SELECT Id, Property__c FROM Tenant__c WHERE Property__c
```

```

!= null])
{ existingPropertyIds.add(existingTenant.Property__c;
} for (Tenant__c newTenant :
newlist)

{

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

    }

}

}

}

}

```

MonthlyEmailScheduler.apxc: global class

MonthlyEmailScheduler implements Schedulable

```

{

global void execute(SchedulableContext sc)

{

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

{

sendMonthlyEmails();

        }

    }

public static void sendMonthlyEmails()

{

List<Tenant__c> tenants = [SELECT Id

```

,Email__c FROM

Tenant__c]; for

(Tenant__c tenant :

tenants)

{

String recipientEmail = tenant.Email__c;

String emailContent = 'I trust this email finds you well. I am writing to remind you that the monthly rent is due Your timely payment ensures the smooth functioning of our rental arrangement and helps maintain a positive living environment for all.';

String emailSubject = 'Reminder: Monthly Rent Payment Due';

Messaging.SingleEmailMessage email = new

Messaging.SingleEmailMessage(); email.setToAddresses(new String[] {recipientEmail});

email.setSubject(emailSubject); email.setPlainTextBody(emailContent);

Messaging.sendEmail(new Messaging.SingleEmailMessage[] {email});

}

}

}