

# **LEASE MANAGEMENT**

**College Name:**[Michael Job College of Arts And Science For Women](#)

**College Code:**[bru4z](#)

**TEAM ID : NM2025TMID26951**

**Team Members:**

**Team LeaderName:** [PREETHI.A](#)

Email:[preethinavamani8@gmail.com](mailto:preethinavamani8@gmail.com)

**Team Member1:**[Varshinipriya.V](#)

Email:[varshinipriyavarshinipriya6@gmail.com](mailto:varshinipriyavarshinipriya6@gmail.com)

**Team Member2:**[Sathiya Sree.S](#)

Email:[ssathyasree951@gmail.com](mailto:ssathyasree951@gmail.com)

**Team Member3:**[Mathimitha.R](#)

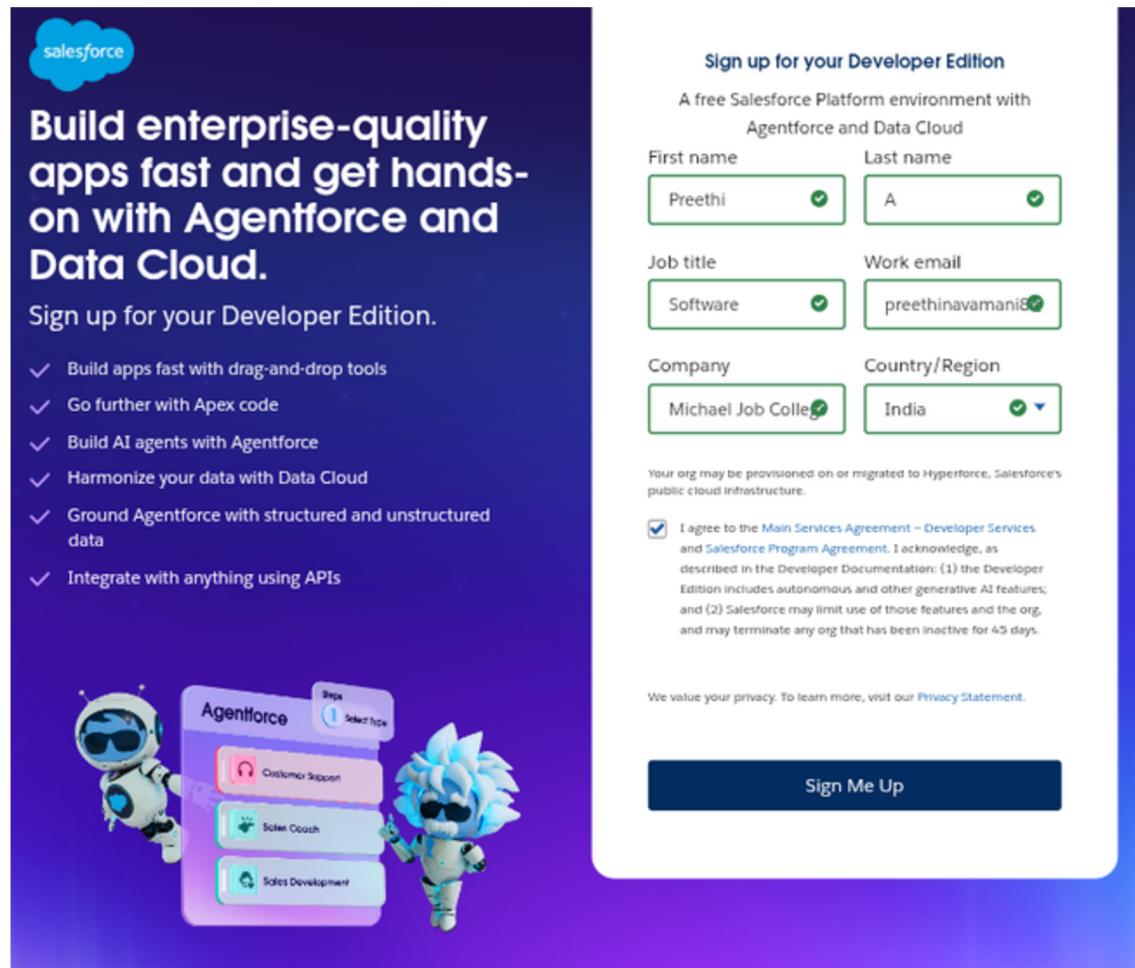
Email:[mathimitha2005@gmail.com](mailto:mathimitha2005@gmail.com)

# Introduction

- 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.

# Salesforce Overview

- - Cloud-based CRM platform
- - Key features used in this project
- - Benefits of Salesforce for lease management



# Project Modules

- - Objects
- - Tabs
- - Lightning App
- - Fields
- - Validation Rules
- - Email Templates
- - Approval Process
- - Apex Trigger
- - Flows
- - Schedule Class

# Objects

- - Custom Object:  
Lease  
Agreement,  
Property, Tenant
- - Relationships  
between objects
- - Example:  
Tenant ↔ Lease  
Agreement ↔  
Property

The screenshot shows the Salesforce Object Manager interface. The top navigation bar includes 'Setup', 'Home', and 'Object Manager'. The main title is 'SETUP > OBJECT MANAGER property'. On the left, a sidebar lists various configuration tabs: 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, Object Access, Triggers, Flow Triggers, Validation Rules, and Conditional Field Formatting. The right panel displays the 'Details' section for the 'property' object. It includes fields for Description, API Name (set to 'property\_\_c'), Singular Label (set to 'property'), Plural Label (set to 'property'), and several checkboxes for reporting and tracking features like 'Enable Reports', 'Track Activities', 'Track Field History', and 'Deployment Status' (which is set to 'Deployed'). There are also links for 'Help Settings' and 'Standard salesforce.com Help Window'. At the bottom right of the panel are 'Edit' and 'Delete' buttons.

Setup Home Object Manager

SETUP > OBJECT MANAGER

## lease

| Details                      | Details  |
|------------------------------|--|
| Fields & Relationships       | Description  |
| Page Layouts                 |  |
| Lightning Record Pages       | API Name<br>lease_c  |
| Buttons, Links, and Actions  | Custom<br><input checked="" type="checkbox"/>              |
| Compact Layouts              | Singular Label<br>lease                                    |
| Field Sets                   | Plural Label<br>lease                                      |
| Object Limits                | Enable Reports<br><input checked="" type="checkbox"/>      |
| Record Types                 | Track Activities<br><input checked="" type="checkbox"/>    |
| Related Lookup Filters       | Track Field History<br><input checked="" type="checkbox"/> |
| Search Layouts               | Deployment Status<br>Deployed                              |
| List View Button Layout      | Help Settings<br>Standard salesforce.com Help Window       |
| Restriction Rules            |  |
| Scoping Rules                |  |
| Object Access                |  |
| Triggers                     |  |
| Flow Triggers                |  |
| Validation Rules             |  |
| Conditional Field Formatting |  |

Edit Delete



Search Setup

Setup

Home

Object Manager

SETUP &gt; OBJECT MANAGER

## Tenant

### Details

Fields &amp; 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

Object Access

Triggers

Flow Triggers

Validation Rules

Conditional Field Formatting

### Details

[Edit](#)[Delete](#)

Description

API Name

Tenant\_c

Custom

✓

Singular Label

Tenant

Plural Label

Tenant

Enable Reports

✓

Track Activities

✓

Track Field History

✓

Deployment Status

Deployed

Help Settings

Standard salesforce.com Help Window



SETUP > OBJECT MANAGER

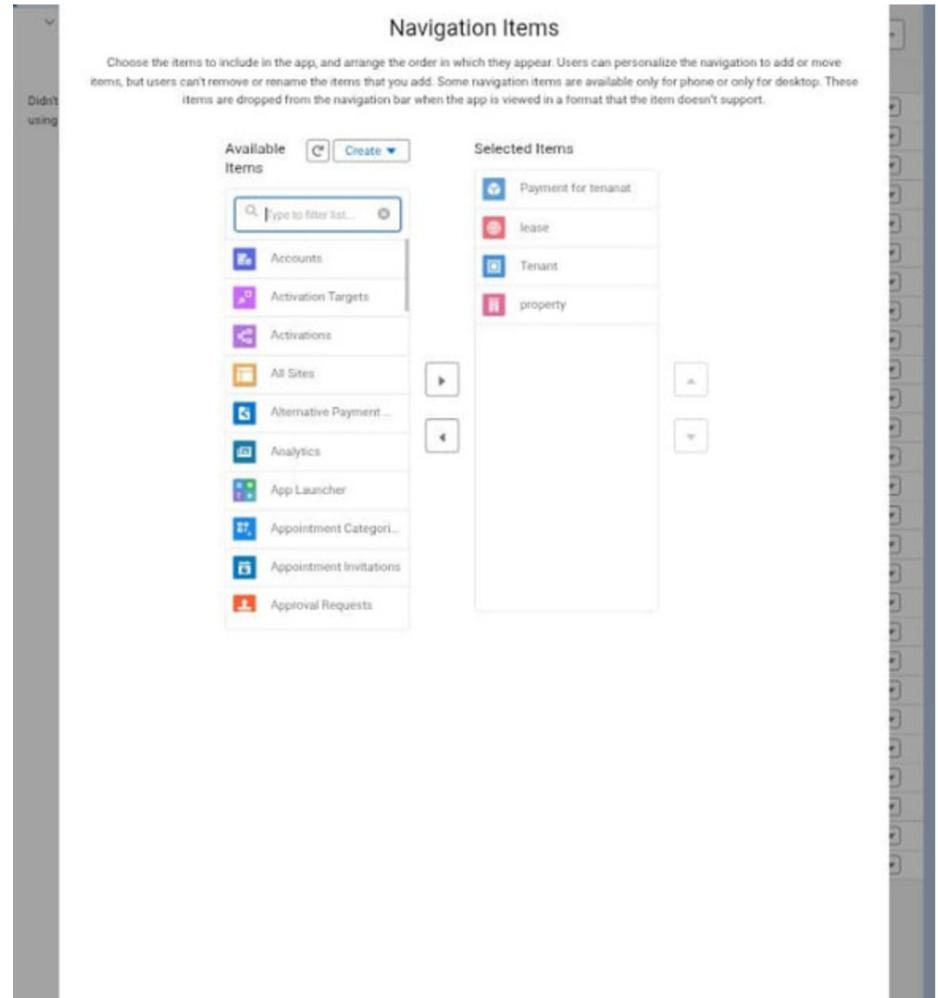
## Payment for tenant

| Details                      | Details                |                                     |
|------------------------------|------------------------|-------------------------------------|
| Fields & Relationships       | Description            | <input type="checkbox"/>            |
| Page Layouts                 |                        | <input checked="" type="checkbox"/> |
| Lightning Record Pages       | API Name               | <input checked="" type="checkbox"/> |
| Buttons, Links, and Actions  | Payment_for_tenantat_c | <input checked="" type="checkbox"/> |
| Compact Layouts              | Custom                 | <input checked="" type="checkbox"/> |
| Field Sets                   | Singular Label         | <input checked="" type="checkbox"/> |
| Object Limits                | Payment for tenantat   | <input checked="" type="checkbox"/> |
| Record Types                 | Plural Label           | <input checked="" type="checkbox"/> |
| Related Lookup Filters       | Payment for tenantat   | <input checked="" type="checkbox"/> |
| Search Layouts               |                        | <input checked="" type="checkbox"/> |
| List View Button Layout      |                        | <input checked="" type="checkbox"/> |
| Restriction Rules            |                        | <input checked="" type="checkbox"/> |
| Scoping Rules                |                        | <input checked="" type="checkbox"/> |
| Object Access                |                        | <input checked="" type="checkbox"/> |
| Triggers                     |                        | <input checked="" type="checkbox"/> |
| Flow Triggers                |                        | <input checked="" type="checkbox"/> |
| Validation Rules             |                        | <input checked="" type="checkbox"/> |
| Conditional Field Formatting |                        | <input checked="" type="checkbox"/> |

[Edit](#) [Delete](#)

# Tabs & Lightning App

- - Created custom tabs for navigation
- - Lightning App for user-friendly interface
- - Screenshots of navigation



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

- Header:** Includes a blue cloud icon, a search bar with the placeholder "Search Setup", and various navigation icons.
- Top Navigation:** Shows "Setup" as the active tab, followed by "Home" and "Object Manager".
- Left Sidebar:** Under "User Interface", it has links for "Rename Tabs and Labels" and "Tabs", with "Tabs" being the active link. It also includes a "Global Search" section.
- Main Content Area:**
  - Section Header:** "Custom Tabs" with a "Help for this Page" link.
  - Text:** "You can create new custom tabs to extend Salesforce functionality or to build new application functionality."
  - Description:** "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."
- Table Section:** "Custom Object Tabs" table with columns: Action, Label, Tab Style, and Description.

| Action     | Label              | Tab Style      | Description |
|------------|--------------------|----------------|-------------|
| Edit   Del | lease              | Globe          |             |
| Edit   Del | Payment for tenant | Building Block |             |
| Edit   Del | property           | Building       |             |
| Edit   Del | Tenant             | Postage        |             |
- Table Section:** "Web Tabs" table with columns: Action, Label, Tab Style, and Description.

| Action                        | Label | Tab Style | Description |
|-------------------------------|-------|-----------|-------------|
| No Web Tabs have been defined |       |           |             |

# Fields & Validation Rules

- - Important custom fields (Start Date, End Date, Rent, Status)
- - Validation Rule Example: End Date > Start Date

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

**Header:** Search Setup, Home, Object Manager

**Breadcrumbs:** SETUP > OBJECT MANAGER > lease

**Left Sidebar (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
- Object Access
- Triggers
- Flow Triggers
- Validation Rules** (highlighted in blue)
- Conditional Field Formatting

**Right Panel (lease Validation Rule):**

**Validation Rule Detail:**

| Rule Name               | lease_end_date                                | Active                                   |
|-------------------------|---|--|
| Error Condition Formula | End_date__c > start_date__c                   | <input checked="" type="checkbox"/>      |
| Error Message           | Your End date must be greater than start date | Error Location: start date               |
| Description             | Preeti A 9/3/2025, 12:36 AM                   | Modified By: Preeti A 9/3/2025, 12:36 AM |

**Buttons:** Edit, Clone

# Email Templates

- - Automated communication with tenants
- - Example: Rent Due Reminder Email

sending mass email.

Available Merge Fields

Select Field Type

Contact Fields

Select Field

Copy Merge Field Value

Copy and paste the merge field value into your template below.

## Step 2. Text Email Template: New Template

Step 2 of 2

Previous Save Cancel

### Email Template Information

Required Information

Folder: Unfiled Public Classic Email Templates

Available For Use:

Email Template Name: Leave approved

Template Unique Name: Leave\_approved

Encoding: Unicode (UTF-8)

Description:

Subject: Leave approved

Email Body: dear {!Tenant\_\_c.Name}

Previous Save Cancel

## Classic Email Templates

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

Text Email Template  
Tenant EmailHelp for this Page 

Preview your email template below.

## Email Template Detail

[Edit](#) [Delete](#) [Clone](#)

|                                 |  |
|---------------------------------|--|
| Email Templates from Salesforce | Unfiled Public Classic Email Templates         |
| Email Template Name             | Tenant Email                                   |
| Template Unique Name            | Tenant_Email                                   |
| Encoding                        | Unicode (UTF-8)                                |
| Author                          | <a href="#">Preethi A [Change]</a>             |
| Description                     |  |
| Created By                      | <a href="#">Preethi A</a> , 9/3/2025, 12:53 AM |
| Modified By                     | <a href="#">Preethi A</a> , 9/3/2025, 4:48 AM  |

[Edit](#) [Delete](#) [Clone](#)

## Email Template

[Send Test and Verify Merge Fields](#)

Subject : \*Urgent: Monthly Rent Payment Reminder\*

## Plain Text Preview

Dear {!Tenant\_\_c.Name}

Attachments

[Attach File](#)

# Approval Process

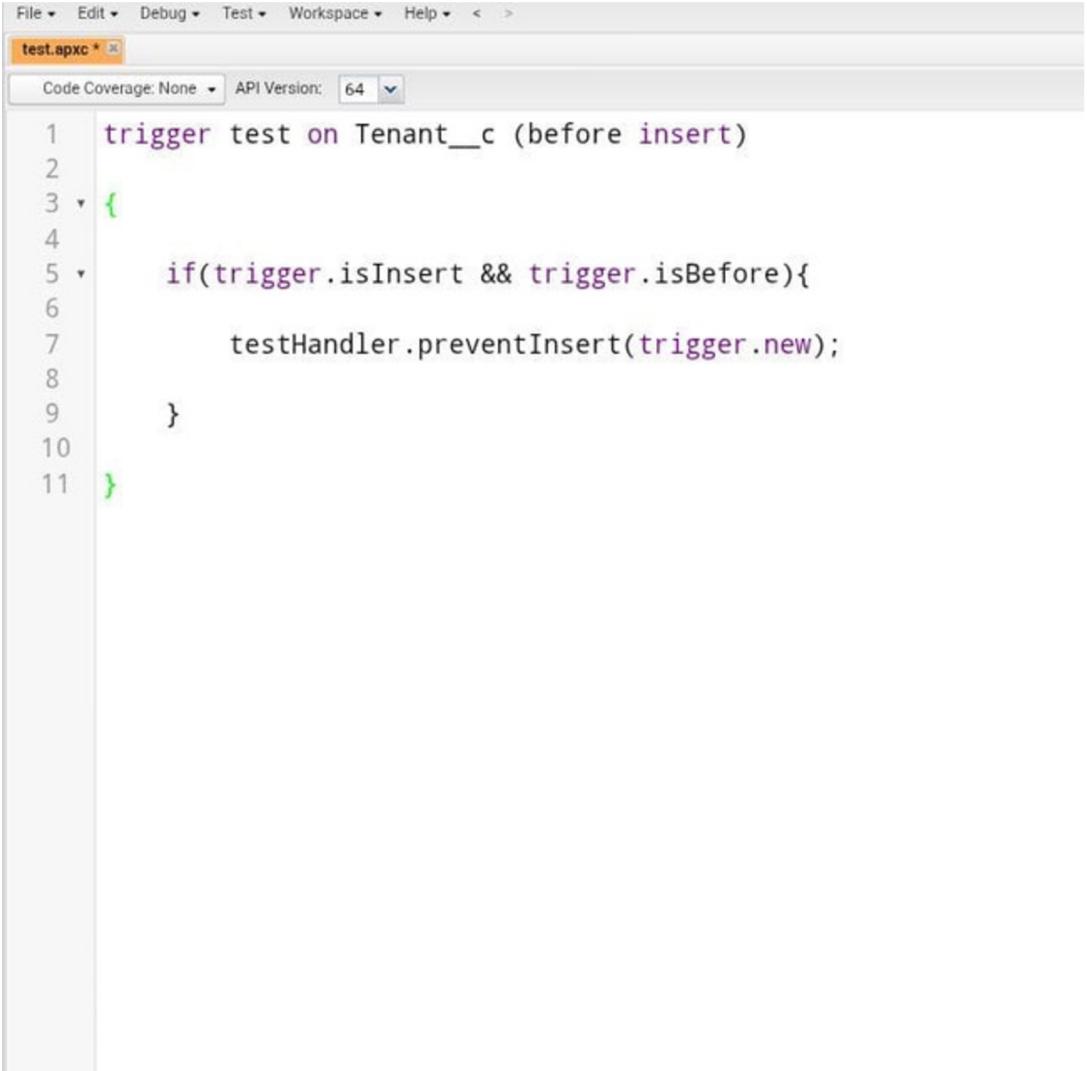
- Approval workflow for Lease Agreement
- Example: Manager approval for new lease

The screenshot shows the Salesforce Setup interface with the 'Approval Processes' page selected. The page title is 'Approval Processes' under the 'SETUP' tab. A sub-header reads 'Tenant: check for vacant'. The 'Process Definition Detail' section contains the following information:

| Process Name                       | check for vacant                    | Active                                       |
|------------------------------------|-------------------------------------|--|
| Unique Name                        | check_for_vacant                    | Next Automated Approver Determined By        |
| Description                        |                                     |  |
| Entry Criteria                     | Tenant: status NOT EQUAL TO Leaving |  |
| Record Editability                 | Administrator ONLY                  | Allow Submitters to Recall Approval Requests |
| Approval Assignment Email Template |                                     |  |
| Initial Submitters                 | Tenant Owner                        |  |
| Created By                         | Preethi A 9/3/2025, 3:59 AM         | Modified By Preethi A 9/8/2025, 12:30 AM     |

# Apex Trigger

- - Used to automate backend logic
- - Example:  
Auto-update Lease Status when End Date passes



The screenshot shows the Salesforce IDE interface with a code editor window titled "test.apxc". The API Version is set to 64. The code in the editor is an Apex trigger:

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

Optimizer

Sales Cloud Everywhere

## ADMINISTRATION

&gt; Users

&gt; Data

&gt; Email

## PLATFORM TOOLS

&gt; Subscription Management

&gt; Apps

&gt; Feature Settings

&gt; Slack

&gt; Data Cloud

&gt; Heroku

&gt; MuleSoft

&gt; Einstein

&gt; Objects and Fields

&gt; Events

&gt; Process Automation

&gt; User Interface

## Custom Code

## Apex Classes

Apex Settings

Apex Test Execution

Apex Test History

Apex Triggers

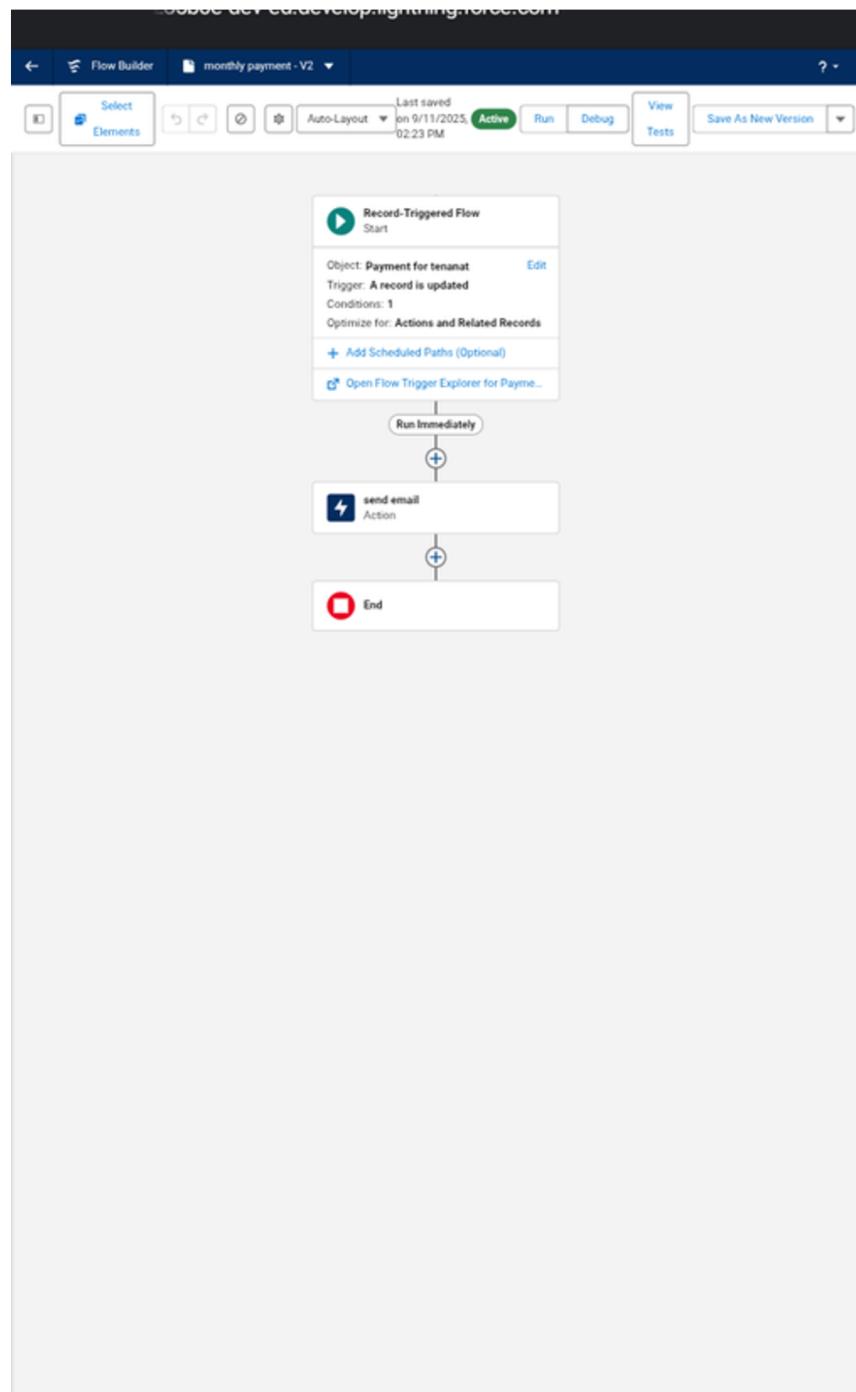
Canvas App Previewer

```
1 public class testHandler {
2
3     public static void preventInsert(List<Tenant__c> newList) {
4
5         Set<Id> existingPropertyIds = new Set<Id>();
6
7         for (Tenant__c existingTenant : [SELECT Id, Property__c FROM Tenant__c WHERE Property__c != null]) {
8
9             existingPropertyIds.add(existingTenant.Property__c);
10
11        }
12
13        for (Tenant__c newTenant : newList) {
14
15            if (newTenant.Property__c != null && existingPropertyIds.contains(newTenant.Property__c)) {
16
17                newTenantaddError('A tenant can have only one property');
18
19            }
20
21        }
22    }
23
24 }
25
26 }
27
28 }
```

[Edit](#) [Delete](#) [Download](#) [Security](#) [Show Dependencies](#)

# Flows

- -  
Automation  
for lease  
renewal
- - Collecting  
tenant data  
using  
screen  
flows



# Schedule Class

- - Scheduled jobs for monthly rent reminders
- - Example: Cron job for sending emails every 1st of the month

The screenshot shows a software interface for 'Lease Management'. At the top, there's a navigation bar with icons for search, refresh, and help, followed by a 'Search...' input field. Below the bar, the title 'Lease Management' is displayed, along with dropdown menus for 'Home', 'property', 'lease', 'Payment for tenant', and 'Tenant'. A blue ribbon bar highlights the 'Tenant' tab. The main content area is titled 'niranjan' under a 'Tenant' icon. It contains two tabs: 'Related' and 'Details'. The 'Details' tab is active, showing the following fields:

- Tenant Name: niranjan
- Owner: Preethi A.
- Phone: (876) 543-2105
- Status: Stay
- Property: hyd
- Created By: Preethi A., 9/11/2025, 1:56 AM
- Last Modified By: Preethi A., 9/11/2025, 2:05 AM

On the right side, there's an 'Activity' section with a header 'Activity' and a sub-section 'Upcoming & Overdue'. It displays a message: 'No activities to show. Get started by sending an email, scheduling a task, and...'. Below this, it says 'No past activity. Past meetings and tasks marked as done sh...'. There are also icons for creating new contacts, editing, and viewing opportunities.

# Benefits of the System

- - Faster lease processing
- - Reduced manual work
- - Improved communication  
with tenants
- - Automatic reminders &  
approvals

# 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.