

**PROJECT TITLE**

**A CRM APPICATION TO MANAGE**

**THE MALL**

College Name: **MAILAM ENGINEERING COLLEGE**

Code: **4216**

**Department of Computer Science and Engineering**

**Team Leader:**

Name: Mohamed Faiyas M Reg No: 421622104084

**Team Members:**

Name: Jeeva S Reg No: 421622104058

Name: Nandha Kumar G Reg No: 421622104092

Name: Manikandan R Reg No: 421622104078

Registered Email Id: [mohamedfaiyas11@gmail.com](mailto:mohamedfaiyas11@gmail.com)

TrailheadUrl: [https://mailamengineeringcolleg-5d5-dev-](https://mailamengineeringcolleg-5d5-dev-ed.develop.lightning.force.com/lightning/setup/SetupOneHome/home) [ed.develop.lightning.force.com/lightning/setup/SetupOneHome/home](https://mailamengineeringcolleg-5d5-dev-ed.develop.lightning.force.com/lightning/setup/SetupOneHome/home)



# Project Overview

The Management App is a comprehensive software solution designed to enhance the management of commercial malls. Built on the Salesforce platform, this innovative app focuses on optimizing the lease tracking process, improving tenant interactions, and providing actionable insights for informed decision-making.

The solution addresses the challenges of traditional mall management by automating essential processes, such as tracking lease agreements, managing tenant data, and ensuring timely rent collections. This automation significantly reduces manual tasks, improves data accuracy, and boosts operational efficiency.

Key features include real-time tracking of lease statuses, user-friendly interfaces for managing tenant information, and enhanced reporting capabilities. With robust data security and role-based access controls, the app ensures that sensitive information is protected, while supporting seamless collaboration among mall management teams.

Customizable dashboards and reports provide mall managers with valuable insights into tenant performance, lease utilization, and financial metrics. This facilitates more informed decision-making, improved tenant relationships, and enhanced resource allocation.

In summary, the Management App is an innovative solution that modernizes mall management, helping businesses streamline operations, enhance tenant satisfaction, and drive overall success in the commercial real estate market.

Contents

[1. Project Overview 2](#_Toc184154142)

[2. Objectives 4](#_Toc184154143)

[3. Salesforce Key Features and Concepts Utilized 5](#_Toc184154144)

[ Custom Objects and Fields: 5](#_Toc184154145)

[ Process Automation: 5](#_Toc184154146)

[ Validation Rules: 5](#_Toc184154147)

[ Reports and Dashboards: 5](#_Toc184154148)

[ Lightning Components: 6](#_Toc184154149)

[ Data Security and Sharing: 6](#_Toc184154150)

[ Approval Processes: 6](#_Toc184154151)

[ Chatter Collaboration: 6](#_Toc184154152)

[ Mobile Accessibility: 6](#_Toc184154153)

[ Integration with Third-Party Tools: 6](#_Toc184154154)

[4. Detailed Steps to Solution Design 7](#_Toc184154155)

[Step 1: Creating Developer Account and Activation 7](#_Toc184154156)

[Step 2: Create Custom Objects 8](#_Toc184154157)

[Step 3: Create Custom Tabs 9](#_Toc184154158)

[Step 4: Create Fields 10](#_Toc184154159)

[Step 5: Create a Lightning App 11](#_Toc184154160)

[Step 6: Record Insertion 13](#_Toc184154161)

[Step 7: Create a Record-Triggered Flow on Tenant Object 15](#_Toc184154162)

[Step 8: Create a Schedule-Triggered Flow on Lease Management Object 16](#_Toc184154163)

[Step 9: Apex Triggers 18](#_Toc184154164)

[Step 10: Schedule Apex for Deleting Tenant Records Monthly 21](#_Toc184154165)

[5. Testing and Validation for Reports and Dashboards 23](#_Toc184154166)

[1. Testing and Validation for Lease Management Report 23](#_Toc184154167)

[2. Testing and Validation for Tenant Issue Report 23](#_Toc184154168)

[3. Testing and Validation for Tenant Details Report 24](#_Toc184154169)

[4. Testing and Validation for Dashboard Creation 24](#_Toc184154170)







# Objectives

The primary objective of the CRM Application to Manage the Mall project is to develop a comprehensive system that enhances the management of commercial malls, focusing on lease tracking, tenant management, and overall operational efficiency. The key goals include:

* **Streamlining Lease Management**:

Centralizes the management of lease agreements, tenant details, and rental payments to ensure smooth and efficient operations within the mall.

* **Automating Tenant Interactions**:

Reduces manual tasks by automating key processes such as tenant communication, lease renewal reminders, and maintenance requests, improving operational workflow.

* **Improving Resource Allocation**:

Provides real-time insights into mall occupancy rates, tenant needs, and resource availability, ensuring the optimal allocation of space and resources.

* **Tracking Performance Metrics:**

Monitors tenant performance, including rental payments, lease duration, and occupancy rates, enabling informed decisions for better management and planning.

* **Enhancing Data Security and Integrity:**

Implements role-based access controls and secure data storage to ensure that sensitive tenant and lease information is protected and managed accurately.

* **Delivering Actionable Insights:**

Provides comprehensive reporting and analytics features to track financial performance, tenant satisfaction, and operational efficiency, supporting data-driven decision-making.

* **Increasing Efficiency and Tenant Satisfaction:**

Improves tenant relationships by ensuring timely communication and a transparent leasing process, leading to higher tenant satisfaction, retention, and operational efficiency.







# Salesforce Key Features and Concepts Utilized

## Custom Objects and Fields:

* + Created custom objects to manage tenant data, lease agreements, and mall resources.
  + Defined fields to store relevant details such as lease terms, tenant information, payment schedules, and maintenance requests.

## Process Automation:

* + Used Flow Builder to automate workflows like lease renewal reminders, maintenance request tracking, and payment processing.
  + Developed Apex Triggers to handle complex processes such as automated tenant notifications, lease status updates, and occupancy reports.

## Validation Rules:

* + Applied validation rules to ensure accurate and consistent data entry for lease agreements, tenant details, and payment schedules.

## Reports and Dashboards:

* + Designed custom dashboards to visualize tenant occupancy, lease expiration dates, and financial performance.
* Created detailed reports for tracking rental payments, lease durations, and mall resource utilization, supporting better decision-making.

A blue text on a black background

Description automatically generated



A black and blue text

Description automatically generated



## Lightning Components:

* + Utilized Salesforce Lightning for a dynamic and responsive user interface.
* Integrated drag-and-drop functionality and interactive tools to enhance user experience and streamline lease management processes.

## Data Security and Sharing:

* + Configured role-based access controls to safeguard sensitive tenant and lease information, ensuring secure data sharing across teams.

## Approval Processes:

* + Established workflows for approvals, such as lease agreements, tenant requests, and payment authorizations, to enhance operational efficiency.

## Chatter Collaboration:

* + Enabled real-time collaboration among mall management teams using Salesforce Chatter for project updates, tenant communications, and discussions.

## Mobile Accessibility:

* + Configured the Salesforce Mobile App to provide on-the-go access to tenant data, lease details, and mall performance metrics for management teams.

## Integration with Third-Party Tools:

* + I Integrated with external systems for accounting, payment processing, and facility management, creating a cohesive operational ecosystem that simplifies mall management.

A blue text on a black background

Description automatically generated

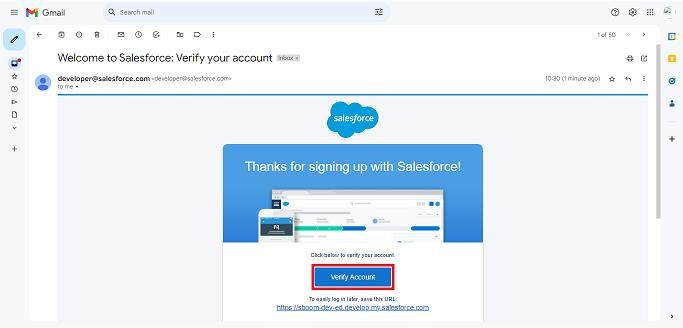
A black and blue text

Description automatically generated

# Detailed Steps to Solution Design

## Step 1: Creating Developer Account and Activation

* Creating a developer org in salesforce.
  + Activate the account by clicking on the verification link.
  + Set a password and answer a security question, then click Change Password.
  + After setting the password, you will be redirected to the Salesforce setup page.









## Step 2: Create Custom Objects

* + Navigate to **Setup > Object Manager > Create > Custom Object**.
  + Create the following objects:

1. **Tenant Object**
2. **Lease Tracking Object**
   * **Tenant Issues Object**Define labels, plural labels, record name format, and data type for each object.









## Step 3: Create Custom Tabs

* + Tabs allow users to view and create records for your objects.
  + Select the object from the list (e.g., Employee Object, Project Object, ProjectTask, Asset, or Asset Service).
  + Select an object and choose a tab style.

A screenshot of a computer

Description automatically generated

* + Ensure "Append tab to user’s existing personal customizations" is checked.
  + Save your changes.





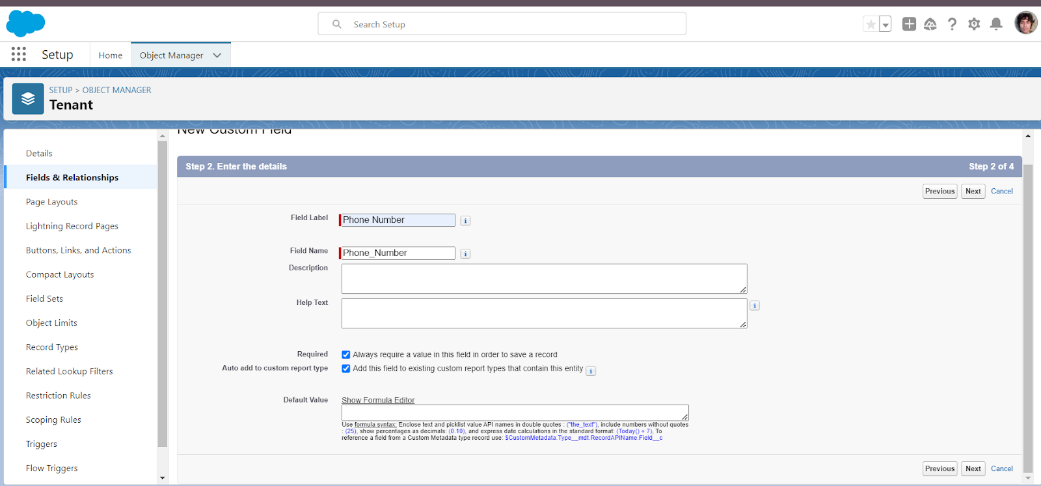
A black and blue text

Description automatically generated

## Step 4: Create Fields

* + Add custom fields for each object.
  + Example fields for the **Tenant Object:**

1. **PAN Card**
   * **Field Name**: PAN\_Card\_\_c
   * **Data Type**: Text (Length: 18)



1. **Possession Due Date**

* **Field Name:** Possession\_Due\_Date\_\_c
* **Data Type:** Formula (Return Type: Date)

A screenshot of a computer

Description automatically generated

* Repeat for other objects.

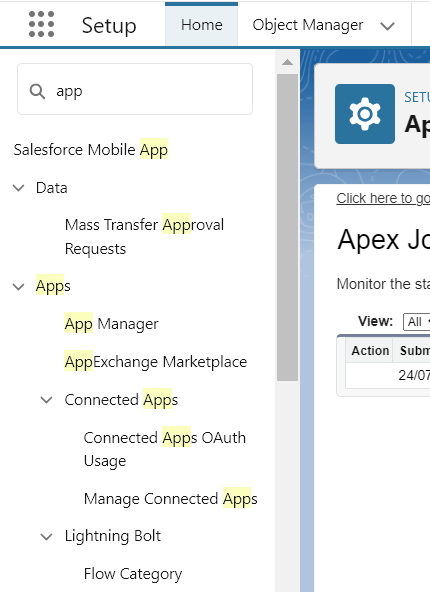






## Step 5: Create a Lightning App

* + **Go to Setup** > search **“App Manager”** in Quick Find > select **App Manager** > click on **New Lightning App**.



* + Fill in the app details:
* **App Name**: SmartMall
* **Developer Name**: Auto-populated
* **Description**: Provide a meaningful description
* **Image**: Optional
  + Click Next, leave **App Options** and **Utility Items** as default, then click **Next**.
  + Add Navigation Items:
* Search and add **Tenant**, **Lease Tracking**, and **Tenant Issues** to the selected items from the available list.



A blue text on a black background

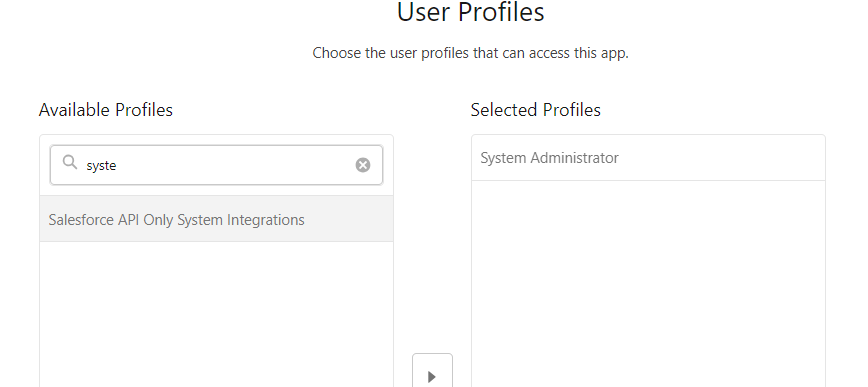
Description automatically generated

A black and blue text

Description automatically generated

* **Add User Profiles**:

* Search for the **System Administrator** profile, add it to the selected profiles list, and click **Save & Finish**.





A blue text on a black background

Description automatically generated

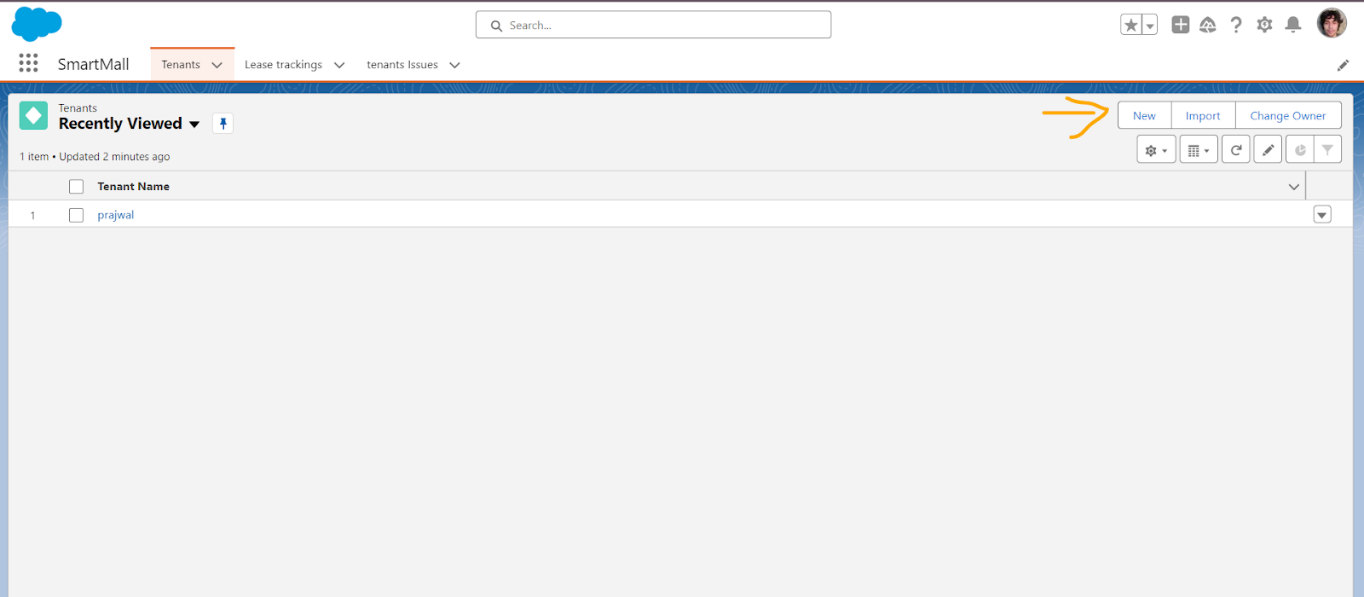
A black and blue text

Description automatically generated

## Step 6: Record Insertion

Creating records in Salesforce is a key activity to ensure proper data management and smooth operations. Below are the steps to insert records into the Tenant Object, Lease Tracking Object, and Tenant Issue Object:

* + **Access the Tenant Object:**
* Click on the **App Launcher**, search for **Tenant**, and open the object.
* Click **New** in the right corner to create a record.



* + **Enter Valid Data:**
* Fill in all required fields with valid information. Ensure that the data adheres to the validation rules set for fields like **Phone Number** and **Date of Reg**.
  + **Handle Validation Errors:**
* If a phone number is more or less than 10 digits, an error message will appear: *“Enter Valid 10 digit Phone Number.”*
* If **Date of Reg** is a past date, an error message will appear: *“Enter Valid Date.”*

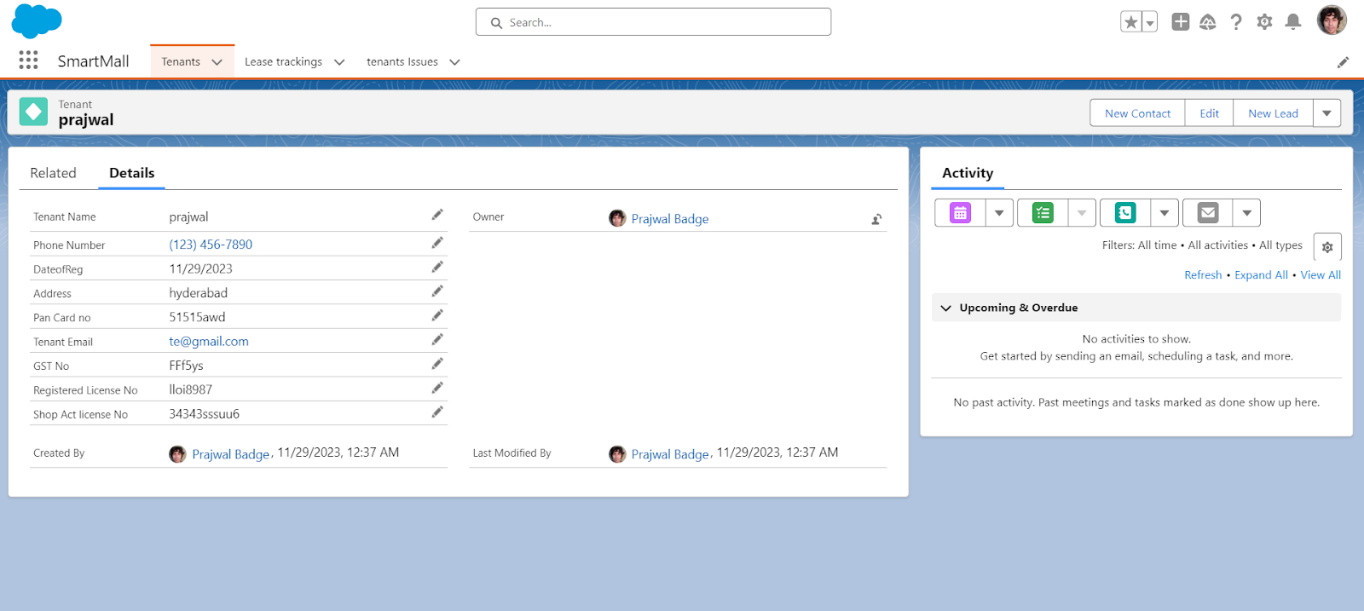
A blue text on a black background

Description automatically generated

A black and blue text

Description automatically generated

* + **Save the Record:**
* Once all fields are valid, click **Save** to create the record. Repeat the process to add at least **10 records** to the **Tenant Object**.



* + **Repeat for Other Objects:**
* Follow similar steps to create at least **10 records each** for the **Lease Tracking Object** and **Tenant Issue Object**.

A blue text on a black background

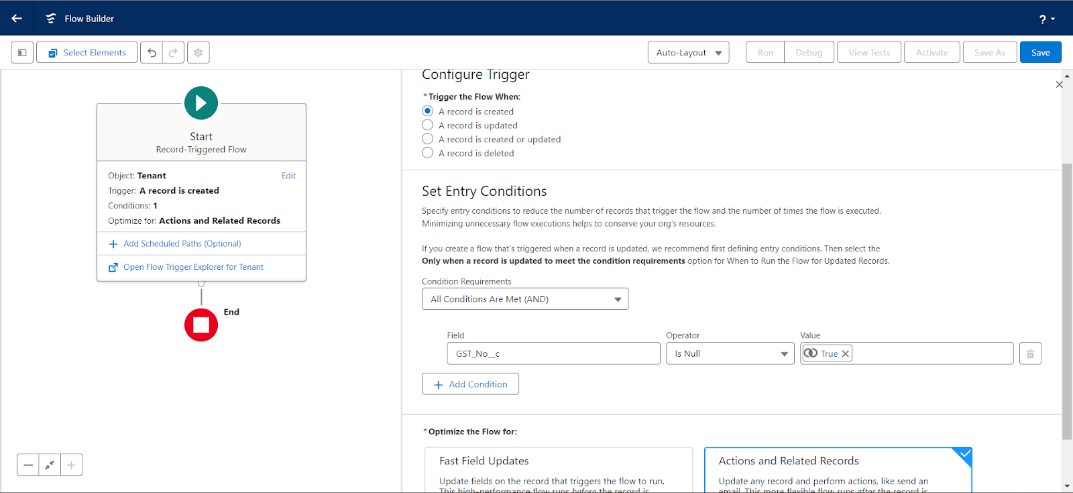
Description automatically generated

A black and blue text

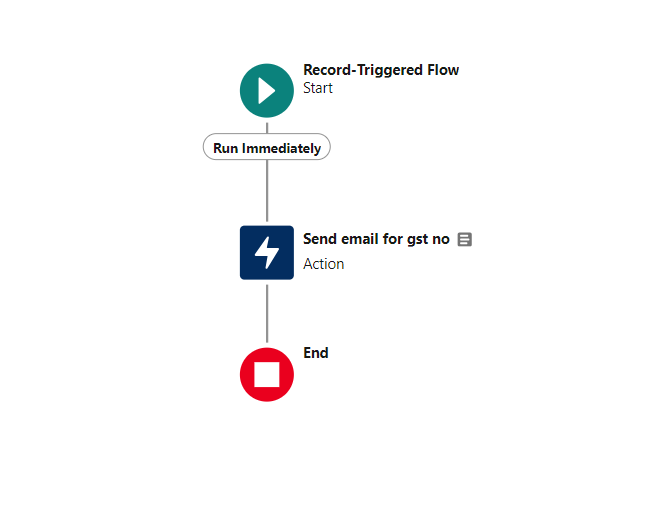
Description automatically generated

## Step 7: Create a Record-Triggered Flow on Tenant Object

* Go to **Setup** > search **Flow** in Quick Find > click **New Flow** > select **Record-Triggered Flow**.
* Set the trigger to fire **When a Record is Created** and add a condition:
* Field: **GST\_No\_\_c**, Operator: **Is Null**, Value: **True**.



* Optimize the flow for **Actions and Related Records** and add an **Action** element.
* Configure the action:
* **Label Name**: "Send email for GST No."
* Include a **Text Template** for the email body, the **Tenant Email ID** as the recipient, and set the subject to "Regarding your GST Details."
* Save the flow with the name **Email Flow for Tenant**, activate it, and ensure it triggers correctly



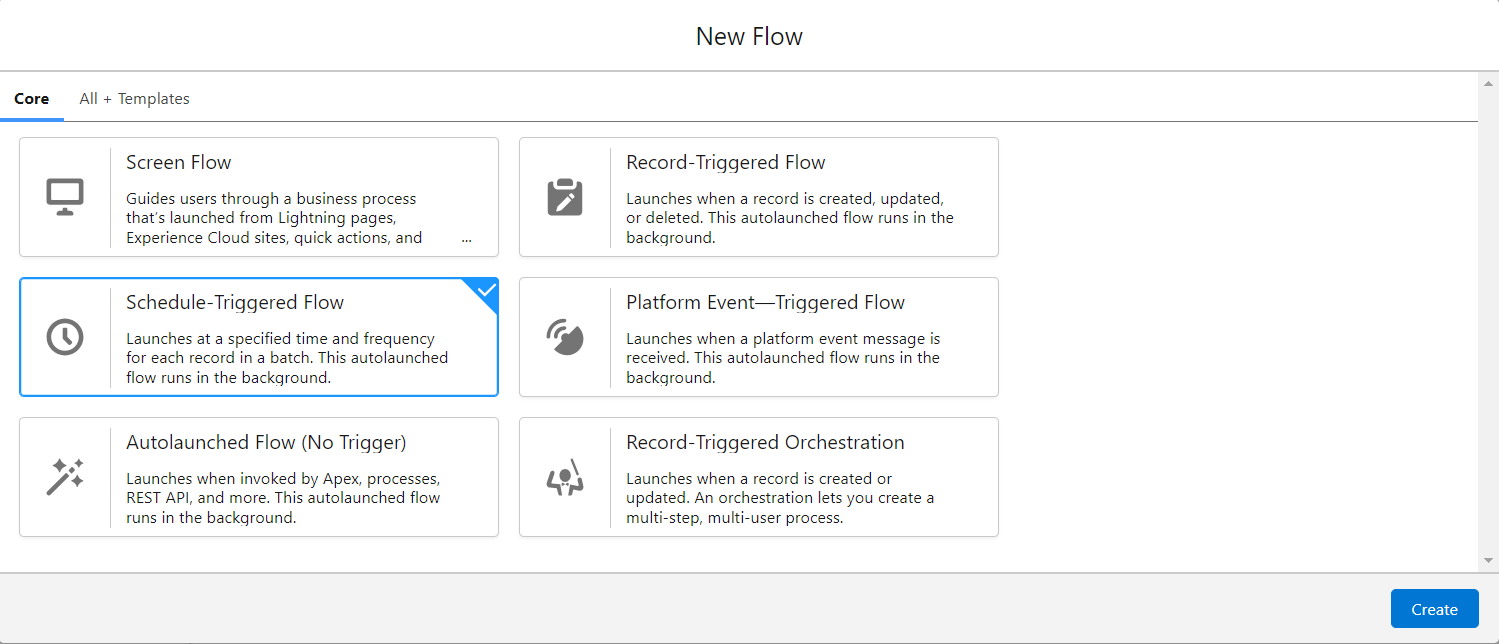






## Step 8: Create a Schedule-Triggered Flow on Lease Management Object

* + Go to **Setup** > search **Flow** in Quick Find > click **New Flow** > select **Schedule-Triggered Flow**.



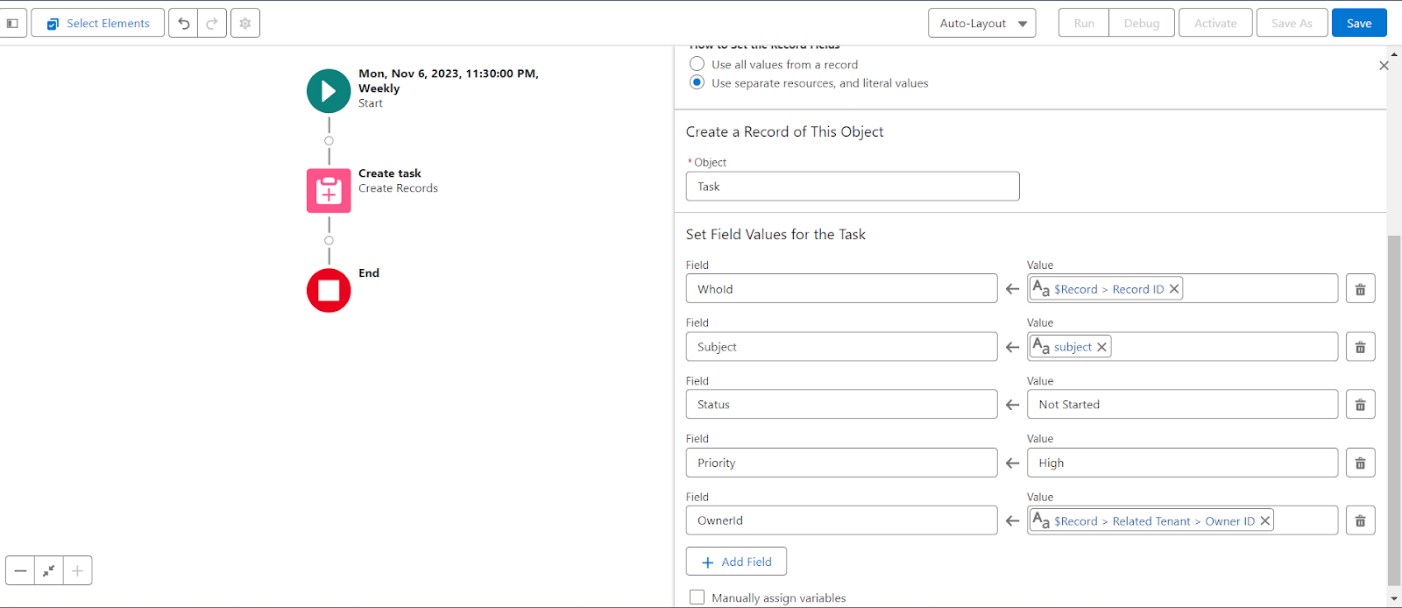
* + Set the schedule:
* Date: Any upcoming **Monday**
* Time: **11:30 AM**
* Frequency: **Weekly**.
  + Choose the **Lease Tracking** object and set the condition:
* Field: **End\_Date\_of\_Possession\_\_c**, Operator: **Less Than**, Value: **1/1/2024**.
  + Add a **Create Record** element:
* Label: "Create Task"
* Object: **Task**
* Subject: "Your possession is going to end soon. Please contact the Manager to renew your possession or end the contract."

EA blue text on a black background

Description automatically generated

A black and blue text

Description automatically generated



* + Save the flow with the name **Schedule Flow on Lease Tracking**, activate it, and confirm the task is created weekly as scheduled.

A blue text on a black background

Description automatically generated

A black and blue text

Description automatically generated

## Step 9: Apex Triggers

**Overview of Apex Triggers**

Apex triggers are used to automate tasks before or after DML (Data Manipulation Language) events such as insert, update, or delete. Triggers enable complex automation, such as validations or custom workflows, beyond what the Salesforce UI provides.

**Trigger Types:**

* **Before Triggers**: Execute before database changes; typically used for validation.
* **After Triggers**: Execute after database changes; used for modifying or interacting with other records.

**Apex Trigger for Lease Tracking**

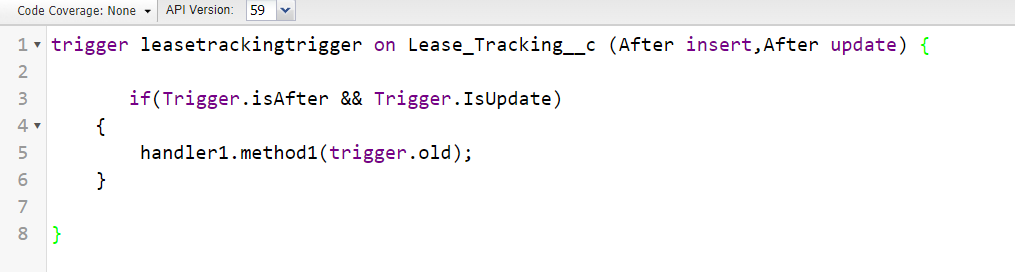
**Objective:**

Send an email notification if the tenant has not paid 50% of the total yearly rent.

**Steps to Implement:**

1. Open the **Developer Console** from the gear icon.
2. Create a new trigger: **Name:** leasetrackingtrigger, **Object:** Lease\_Tracking\_\_c.
3. Use events: **After Insert** and **After Update**.

**Trigger Code:**

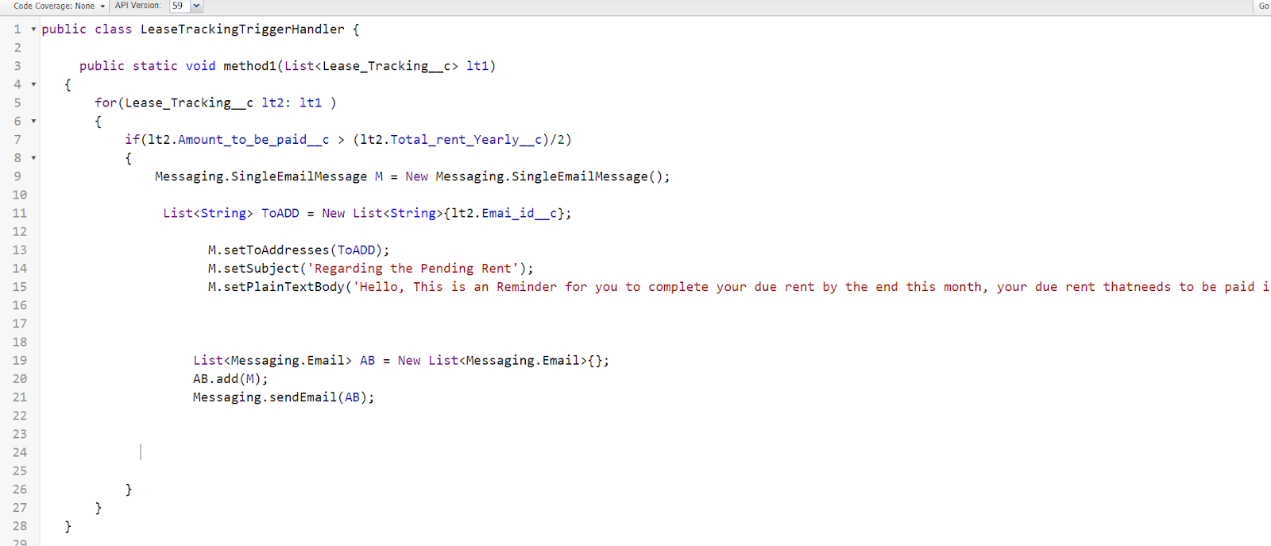








**Handler Class Code:**



**Apex Trigger for Tenant Object**

**Objective:**

Show an error if the PAN card number entered is invalid (length exceeds 10 characters).

**Steps to Implement:**

1. Open the **Developer Console** from the gear icon.
2. Create a new trigger: **Name:** TenantTrigger, **Object:** Tenant\_\_c.
3. Use event: **Before Insert.**

A blue text on a black background

Description automatically generated



A black and blue text

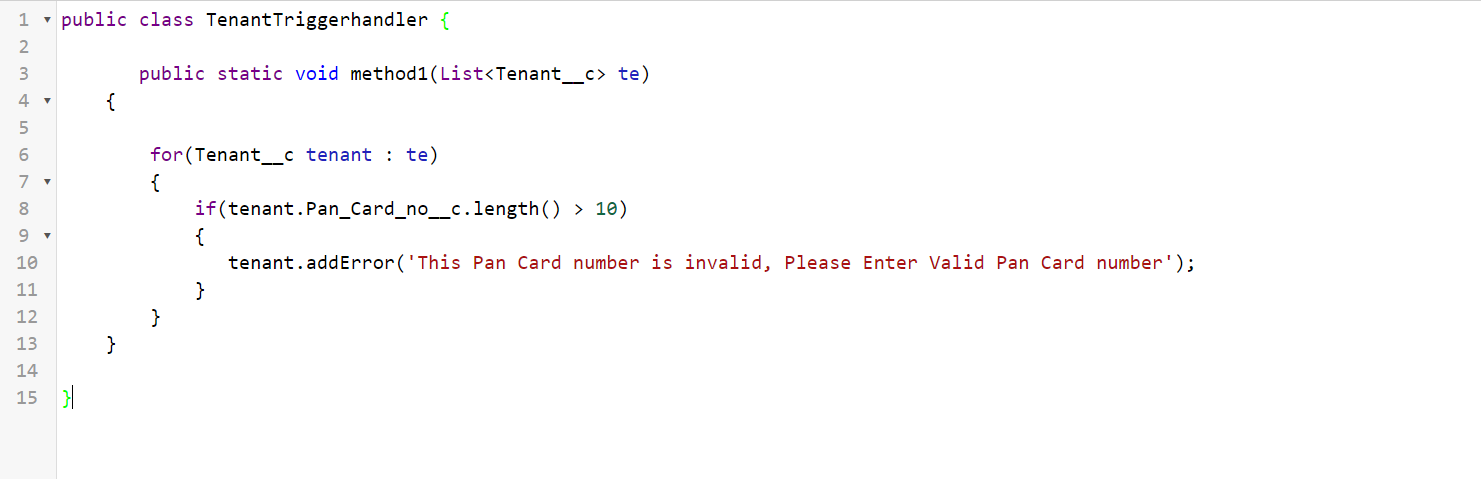
Description automatically generated

**Trigger Code:**

**A screenshot of a computer program

Description automatically generated**

**Handler Class Code:**



A blue text on a black background

Description automatically generated

A black and blue text

Description automatically generated

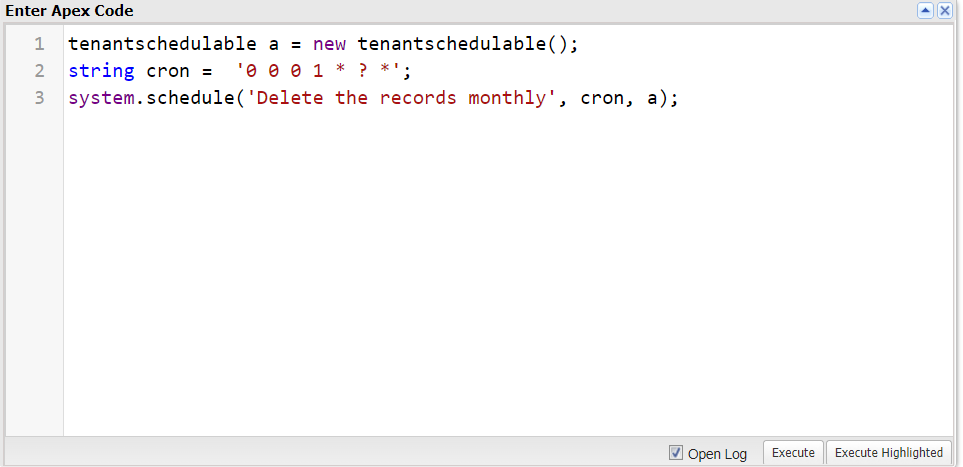
## Step 10: Schedule Apex for Deleting Tenant Records Monthly

Objective

Schedule an Apex class to automatically delete tenant records on a monthly basis where the Status of Possession is marked as "Closed." This helps maintain data hygiene by removing outdated records.

Steps to Implement

* + **Create the Apex Class:**
* Name the class **tenantschedulable** and implement the **Schedulable** interface.
  + **Write the Apex Code:**
* The class will query the **Tenant\_\_c** records, identify those with **Status\_of\_Possession\_\_c = 'Closed'**, and then delete those records.
  + **Open the Anonymous Window:**
* In the **Developer Console**, open the **Anonymous Window** to schedule the class.
  + **Schedule the Class:**
* Use the following code in the **Anonymous Window** to schedule the class to run monthly:



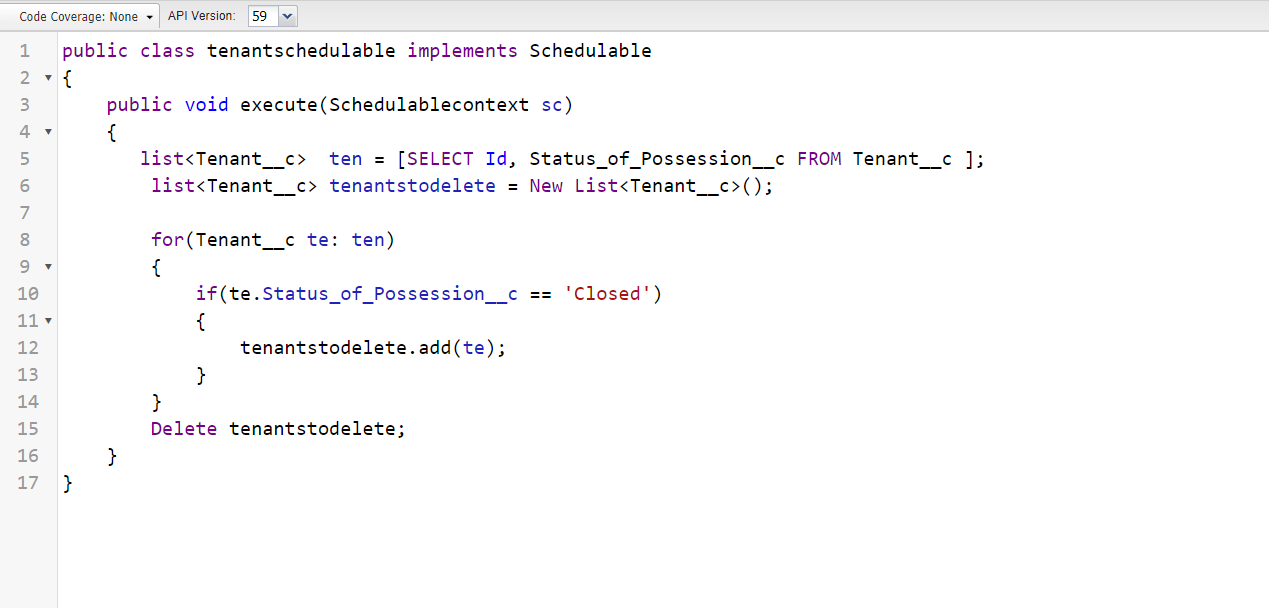
A blue text on a black background

Description automatically generated

A black and blue text

Description automatically generated

**Code Snippet for the Apex Class:**

****







# Testing and Validation for Reports and Dashboards

Effective testing ensures the functionality, performance, and reliability of the Salesforce solution. Here's a concise approach:

## Testing and Validation for Lease Management Report

* Test the Report Creation Process:
  + Ensure that the **MallReports** folder is created successfully and the **Lease Tracking** report is generated.
  + Validate that the **Amount to be paid** column correctly categorizes remaining amounts using the bucket list:
    - Amounts greater than **1,000,000** should appear in **red**.
    - Amounts between **500,000** and **1,000,000** should appear in **blue**.
    - Amounts less than or equal to **500,000** should appear in **yellow**.
  + Verify that the report groups data correctly by **Date of Registration.**
  + Ensure the report is saved under the **MallReports** folder as expected.

## Testing and Validation for Tenant Issue Report

* **Test the Report Creation Process:**
  + Validate the **Tenant Issue** report is generated based on the specified criteria (issues not contacted or open, high priority, encountered by phone and mail, and date of issue within the last 7 days).
  + Check that the report is saved under the **Mall Reports** folder.
  + Confirm the report's **Name** is set to **Issue Report**.
  + Ensure the filter for high-priority issues and the date range is applied correctly.

A blue text on a black background

Description automatically generated



A black and blue text

Description automatically generated

## Testing and Validation for Tenant Details Report

* **Test the Report Creation Process:**
  + Verify that the **Tenant Details** report is created and displays all **pending possessions** with the **PAN Card no** and **GST No.** fields.
  + Ensure the report is grouped by **Date of Registration** and **Status of Possession.**
  + Validate that the report is saved under the **Mall Reports** folder and the name is set to **Tenant Details.**

## Testing and Validation for Dashboard Creation

* **Test the Dashboard Creation Process:**
  + Ensure the **Mall Dashboard** folder is created successfully.
  + Validate that the **December Dashboard** is created and saved in the **Mall Dashboard** folder.
  + Check the components added:
    - **Tenant Details** should display as a **Horizontal Bar Chart** with dimensions **10x12**.
    - **Issue Report** should display as a **Lightning Table** with dimensions **8x6**.
    - **Lease Report** should display as a **Lightning Table** with dimensions **8x6**.
  + Confirm that all components reflect the correct data from their respective reports.
  + Verify that the dashboard is saved and the **Done** button functions as expected.