
Project Documentation: Residential Maintenance Request Portal

- **Author:** Alla Vijaya Krishna
- **System:** Salesforce Lightning Platform

1. Executive Summary


This document outlines the design and functionality of the Residential Maintenance Request Portal, a custom application built on the Salesforce platform.

- **1.1. The Problem:** The original process for managing residential maintenance was manual, leading to delays, a lack of visibility for property managers, and an inefficient workflow for technicians.
 - **1.2. The Solution:** A centralized, custom Salesforce application was developed to automate the entire lifecycle of a maintenance request, from submission to resolution and reporting.
 - **1.3. Business Value:** This solution provides faster response times, ensures data integrity, automates key communications, and delivers powerful analytics for data-driven decision-making.
-

2. Key Stakeholders & Roles

The system is designed for two primary user groups with specific roles and permissions.

- **Property Manager (Profile: Property Manager):** Oversees all maintenance requests, approves high-priority issues, and views dashboards for performance tracking.
- **Maintenance Technician (Profile: Maintenance Technician):** Views and works on assigned maintenance requests, updating the status as work is completed.



SETUP

Profiles

Profile

Property Manager

[Help for this Page](#)

Users with this profile have the permissions and page layouts listed below. Administrators can change a user's profile by editing that user's personal information.

If your organization uses Record Types, use the Edit links in the Record Type Settings section below to make one or more record types available to users with this profile.

[Login IP Ranges \[0\]](#) |
 [Enabled Apex Class Access \[0\]](#) |
 [Enabled Visualforce Page Access \[0\]](#) |
 [Enabled External Data Source Access \[0\]](#) |
 [Enabled Named Credential Access \[0\]](#) |
 [Enabled External Credential Principal Access \[0\]](#) |
 [Enabled Custom Metadata Type Access \[0\]](#) |
 [Enabled Custom Setting Definitions Access \[0\]](#) |
 [Enabled Flow Access \[0\]](#) |
 [Enabled Service Presence Status Access \[0\]](#) |
 [Enabled Custom Permissions \[0\]](#)

Profile Detail

Edit

Clone

Delete

View Users

Name	Property Manager		
User License	Salesforce	Custom Profile	<input checked="" type="checkbox"/>
Description			
Created By	ALLA VIJAYA KRISHNA, 9/23/2025, 9:33 PM	Modified By	ALLA VIJAYA KRISHNA, 9/24/2025, 3:04 AM

Page Layouts

Standard Object Layouts	
Global	<div>Global Layout</div> <div>[View Assignment]</div>
Email Application	<div>Not Assigned</div> <div>[View Assignment]</div>
Home Page Layout	<div>Home Page Default</div> <div>[View Assignment]</div>
Account	<div>Account Layout</div>

Location Group Assignment	<div>Location Group Assignment Layout</div> <div>[View Assignment]</div>
Macro	<div>Macro Layout</div> <div>[View Assignment]</div>
Object Milestone	<div>Object Milestone Layout</div> <div>[View Assignment]</div>
Operating Hours	<div>Operating Hours Layout</div>

3. Data Model

The application is built on a custom data model with two core objects.

- **Property Unit (Property_Unit__c):** Represents a physical residential unit (e.g., Apartment A-101). This object stores static information about the property.
 - Key Fields: Unit Number (Name)
- **Maintenance Request (Maintenance_Request__c):** Represents a single work order or issue related to a Property Unit.
 - Key Fields: Request ID (Name), Status (Picklist), Priority (Picklist), Issue Description (Long Text).
- **Relationship:** There is a **Master-Detail** relationship between the objects. One **Property Unit** can have many **Maintenance Requests**.

SETUP > OBJECT MANAGER

Property Unit

Details

Fields & Relationships

Page Layouts

Lightning Record Pages

Buttons, Links, and Actions

Compact Layouts

Field Sets

Object Limits

Record Types

Related Lookup Filters

Restriction Rules

Fields & Relationships

5 Items, Sorted by Field Label

Q Quick Find

New

Deleted Fields

Field Dependencies

Set History Tracking

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED	
Contact res	Contact__c	Lookup(Contact)		✓	▼
Created By	CreatedById	Lookup(User)			
Last Modified By	LastModifiedById	Lookup(User)			
Owner	OwnerId	Lookup(User,Group)		✓	
Property Unit Name	Name	Text(80)		✓	▼

Maintenance Request Object

SETUP > OBJECT MANAGER

Maintenance Request

Details

Fields & Relationships

Page Layouts

Lightning Record Pages

Buttons, Links, and Actions

Compact Layouts

Field Sets

Object Limits

Record Types

Related Lookup Filters

Restriction Rules

Scoping Rules

Fields & Relationships

8 Items, Sorted by Field Label

Q Quick Find

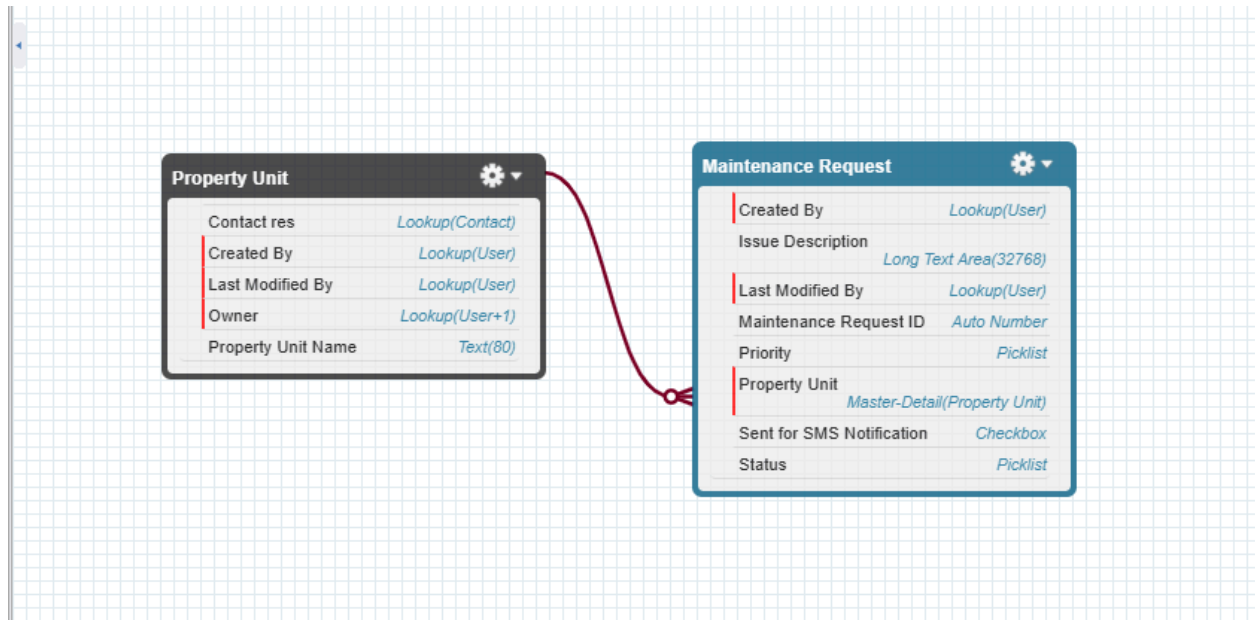
New

Deleted Fields

Field Dependencies

Set History Tracking

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED	
Created By	CreatedById	Lookup(User)			
Issue Description	Issue_Description__c	Long Text Area(32768)			▼
Last Modified By	LastModifiedById	Lookup(User)			
Maintenance Request ID	Name	Auto Number		✓	▼
Priority	Priority__c	Picklist			▼
Property Unit	Property_Unit__c	Master-Detail(Property Unit)		✓	▼
Sent for SMS Notification	Sent_for_SMS_Notification__c	Checkbox			▼
Status	Status__c	Picklist			▼

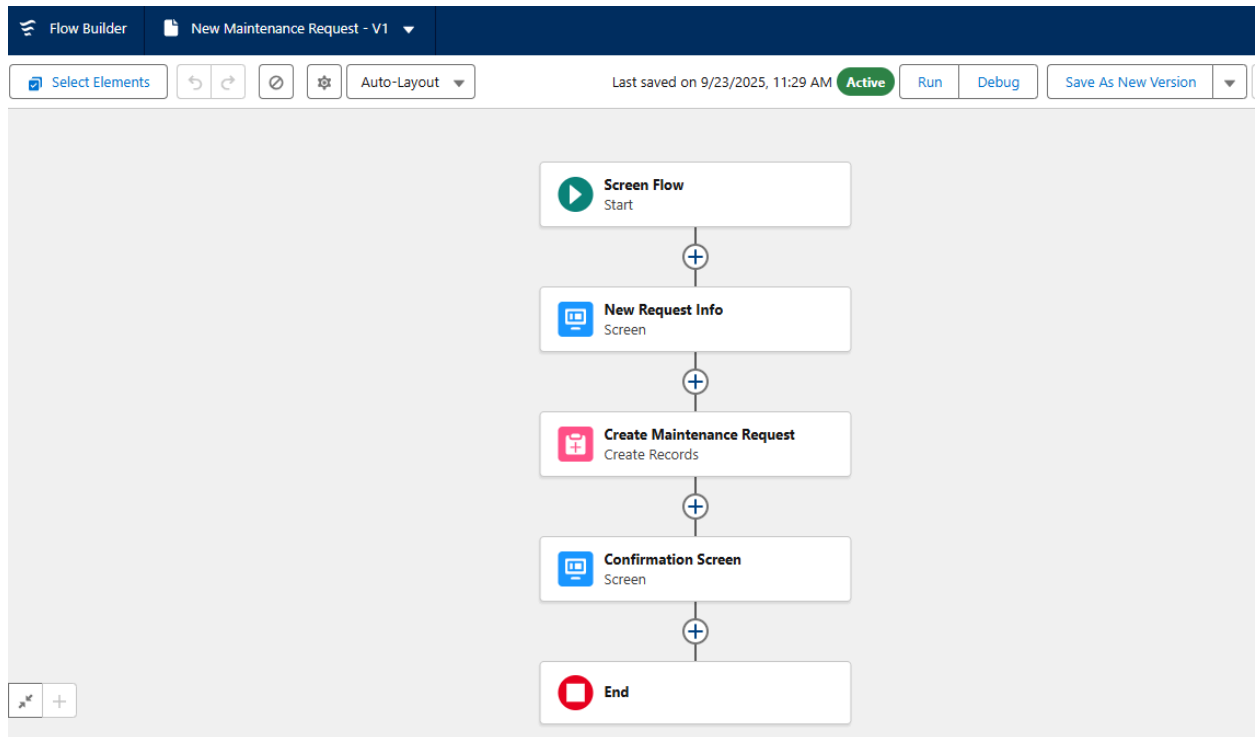


4. How The Project Works: Automation & Processes

The application's logic is driven by a series of automated processes that guide a maintenance request from creation to completion.

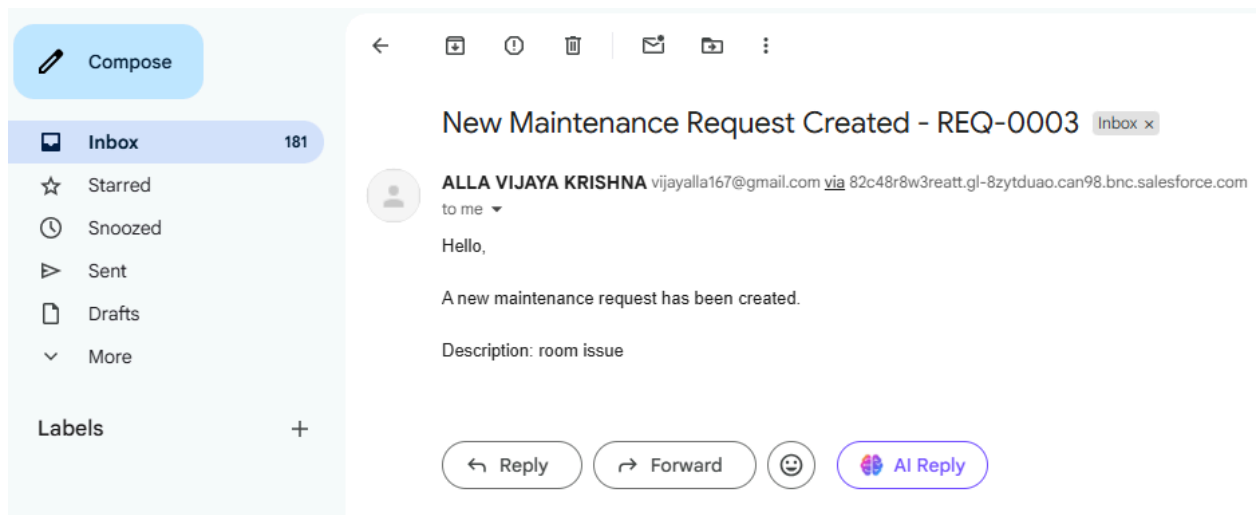
4.1. New Request Creation (Screen Flow)

A **Screen Flow** named "New Maintenance Request" is placed on the Home Page. This provides a simple, guided wizard for users to quickly create new requests by selecting the property, describing the issue, and setting the priority.



4.2. Manager Notification (Record-Triggered Flow)

A **Record-Triggered Flow** runs automatically in the background. When any new `Maintenance_Request__c` record is created, this flow instantly sends an email alert to the Property Manager, ensuring immediate awareness of new issues.



4.3. High-Priority Approval (Approval Process)

A formal **Approval Process** is in place for high-priority issues.

- **Criteria:** It can be initiated for any request where the Priority is "High".
- **Process:** A user clicks the "Submit for Approval" button, which locks the record and assigns an approval request to the Property Manager. The manager sees this request in the "Items to Approve" component on their Home Page and can formally Approve or Reject it.

The screenshot shows a web interface for a Maintenance Request. At the top, there's a header bar with a green wrench icon, the text 'Maintenance Request REQ-0004', and a 'Submit for Approval' button. Below this is a tabbed interface with 'Related' and 'Details' tabs. The 'Details' tab is active, showing a form with the following fields: 'Maintenance Request ID' (REQ-0004), 'Property Unit' (A-101), 'Issue Description' (room issue), 'Priority' (High), 'Status' (Completed), and 'Sent for SMS Notification' (checked). At the bottom, it shows 'Created By' (ALLA VIJAYA KRISHNA, 9/23/2025, 8:56 AM) and 'Last Modified By' (ALLA VIJAYA KRISHNA, 9/24/2025, 3:10 AM).

Field	Value	Action
Maintenance Request ID	REQ-0004	
Property Unit	A-101	
Issue Description	room issue	
Priority	High	
Status	Completed	
Sent for SMS Notification	<input checked="" type="checkbox"/>	
Created By	ALLA VIJAYA KRISHNA, 9/23/2025, 8:56 AM	
Last Modified By	ALLA VIJAYA KRISHNA, 9/24/2025, 3:10 AM	

4.4. Duplicate Prevention (Apex Trigger)

An **Apex Trigger** runs silently in the background to enforce data integrity. It prevents any user from creating a new maintenance request for a property that already has an open (not "Completed") request. This avoids duplicate work.

5. User Interface (UI) Configuration

The user experience has been customized for efficiency.

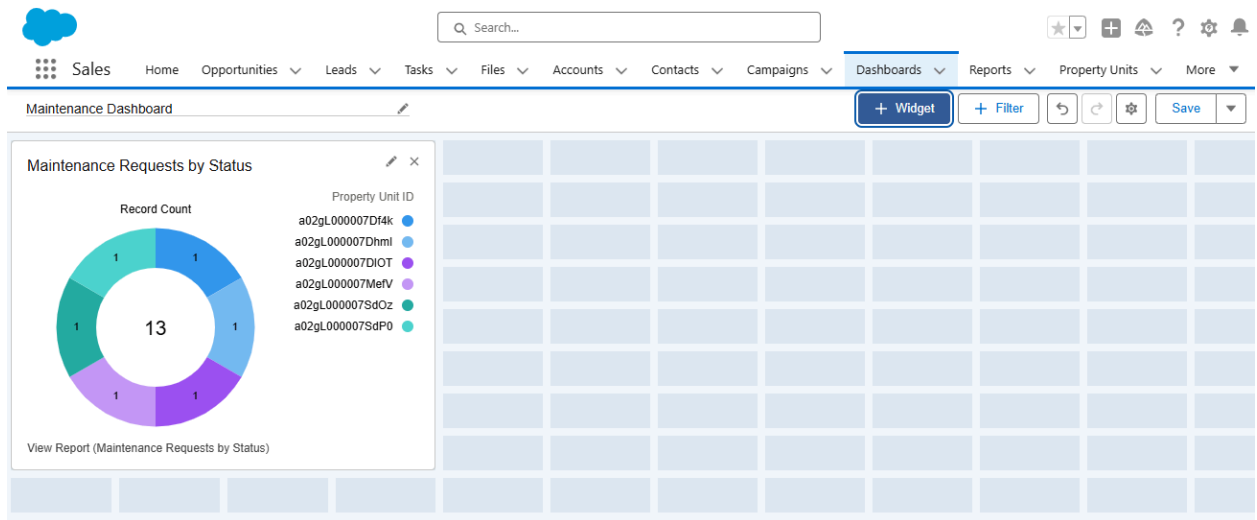
- **Custom Lightning App:** A dedicated "**Residential Maintenance**" app was created, showing only the relevant tabs: Home, Property Units, and Maintenance Requests.
- **Customized Record Page:** The Maintenance Request page layout was modified. The **Highlights Panel** at the top prominently displays the related **Property Unit**, giving technicians instant context.
- **Customized Home Page:** A unique Home Page was designed for Technicians and Managers. It features a **List View** component showing "My Open Requests," which acts as an immediate to-do list for technicians.

The screenshot shows the Salesforce Chatter interface for creating a new maintenance request. The top navigation bar includes the Salesforce logo, a search bar, and tabs for Home, Chatter, People, Groups, Files, Students, Property Units, and Maintenance Requests. The main content area is titled "New Maintenance Request" and contains three input fields: "Which property unit has the issue?" with a search bar, "Describe the issue" with a text area, and "What is the priority?" with a dropdown menu showing "--None--". A "Next" button is located at the bottom right of the form. On the right side, there is an "Assistant" panel with a chat icon and the text "Nothing nee".

6. Reporting & Dashboards

To provide business insights, a custom report and dashboard were created.

- **Report:** A "Maintenance Requests by Status" report was built to group all requests and show how many are New, In Progress, etc.
- **Dashboard:** A "Maintenance Dashboard" was created with a Donut Chart component to visually represent the data from the report.



7. Appendix: Summary of Custom Components

- **Custom Objects:** Property_Unit__c, Maintenance_Request__c
- **Custom Profiles:** Property Manager, Maintenance Technician
- **Flows:** 1 Screen Flow ("New Maintenance Request"), 1 Record-Triggered Flow ("Notify Manager on New Maintenance Request")
- **Apex:** 1 Apex Trigger (MaintenanceRequestTrigger), 1 Test Class (MaintenanceRequestTriggerTest)
- **Approval Process:** 1 process ("High Priority Request Approval")
- **Reports/Dashboards:** 1 Custom Report Type, 1 Report, 1 Dashboard