

NAAN MUDHAVALVAN- SALESFORCE REPORT

A CRM APPLICATION TO MANAGE THE MALL

PROJECT CREATED BY

BE –V SEMESTER

Team Members:

Register Number:

ANANTHI D

421622104004

ARULSELVI S

421622104014

DIVYA K

421622104032

HEMALATHA P

421622104046

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

MAILAM ENGINEERING COLLEGE



1. Project Overview:

This project involves the development of a Customer Relationship Management (CRM) application tailored for mall management, leveraging the Salesforce platform for its robust features and scalability. The application is designed to manage tenants, leases, and issue tracking efficiently. It includes creating custom objects such as Tenant, Lease Tracking, and Tenant Issues, along with corresponding fields, relationships, and custom tabs.

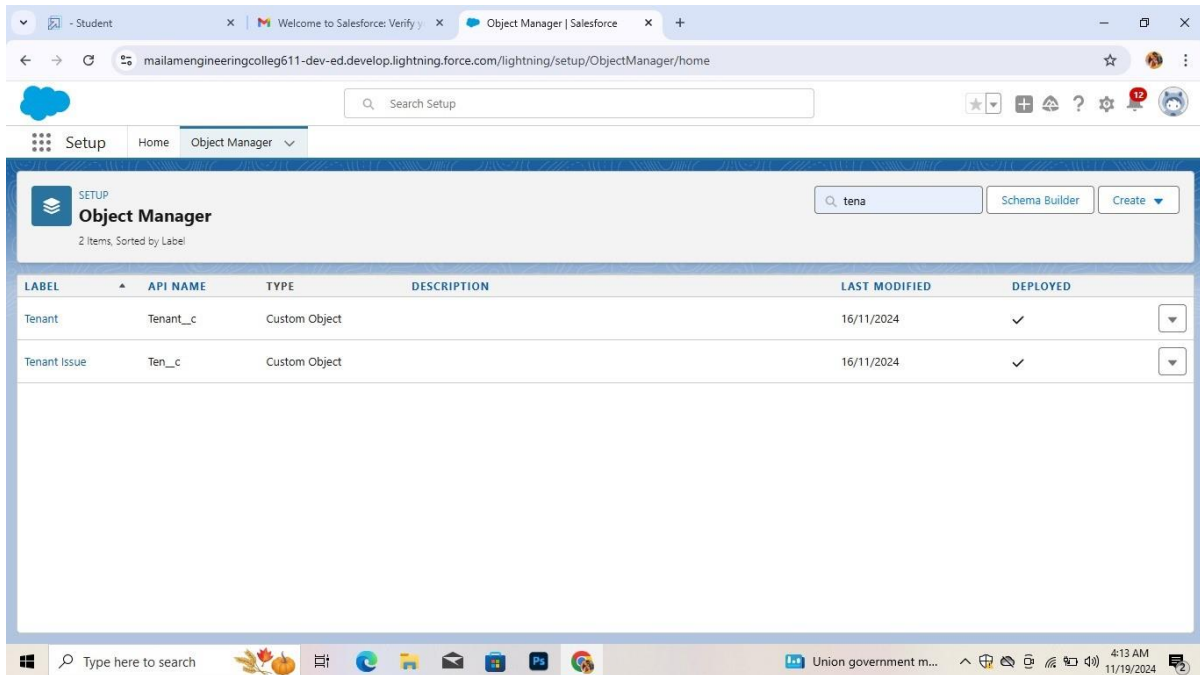
2. Objectives:

The primary objective of this project is to design and implement a robust CRM application using the Salesforce platform to streamline mall management operations. The system aims to efficiently manage tenant information, lease agreements, and tenant issue tracking while providing actionable insights through reports and dashboards. By creating custom objects like Tenant, Lease Tracking, and Tenant Issues, the application ensures that all critical data points are captured and organized.

3. Solution Design Steps for the Salesforce Mall Management CRM Application:

□ Create Custom Objects:

We should create a custom objects in salesforce, follow these details:

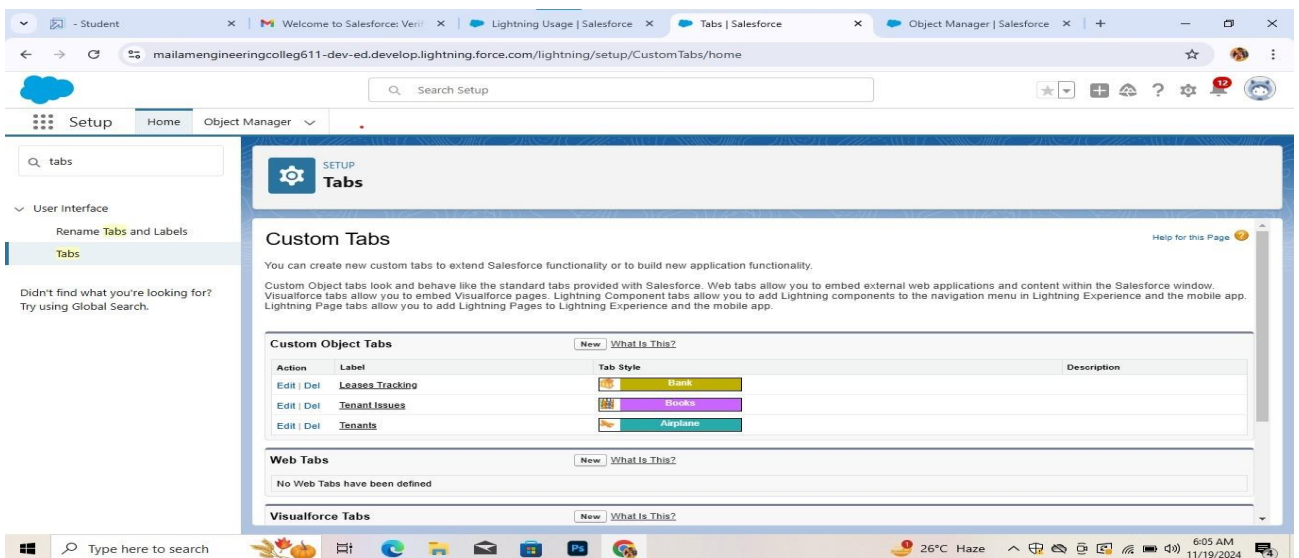


The screenshot shows the Salesforce Object Manager setup page. The browser address bar displays the URL: mailamengineeringcolleg611-dev-ed.develop.lightning.force.com/lightning/setup/ObjectManager/home. The page header includes the Salesforce logo and a search bar. The main content area is titled "Object Manager" and shows a list of custom objects. The table below contains the data:

LABEL	API NAME	TYPE	DESCRIPTION	LAST MODIFIED	DEPLOYED
Tenant	Tenant__c	Custom Object		16/11/2024	✓
Tenant Issue	Ten__c	Custom Object		16/11/2024	✓

□ Create a Tabs:

Overview of key metrics, such as footfall, sales data, and tenant performance.

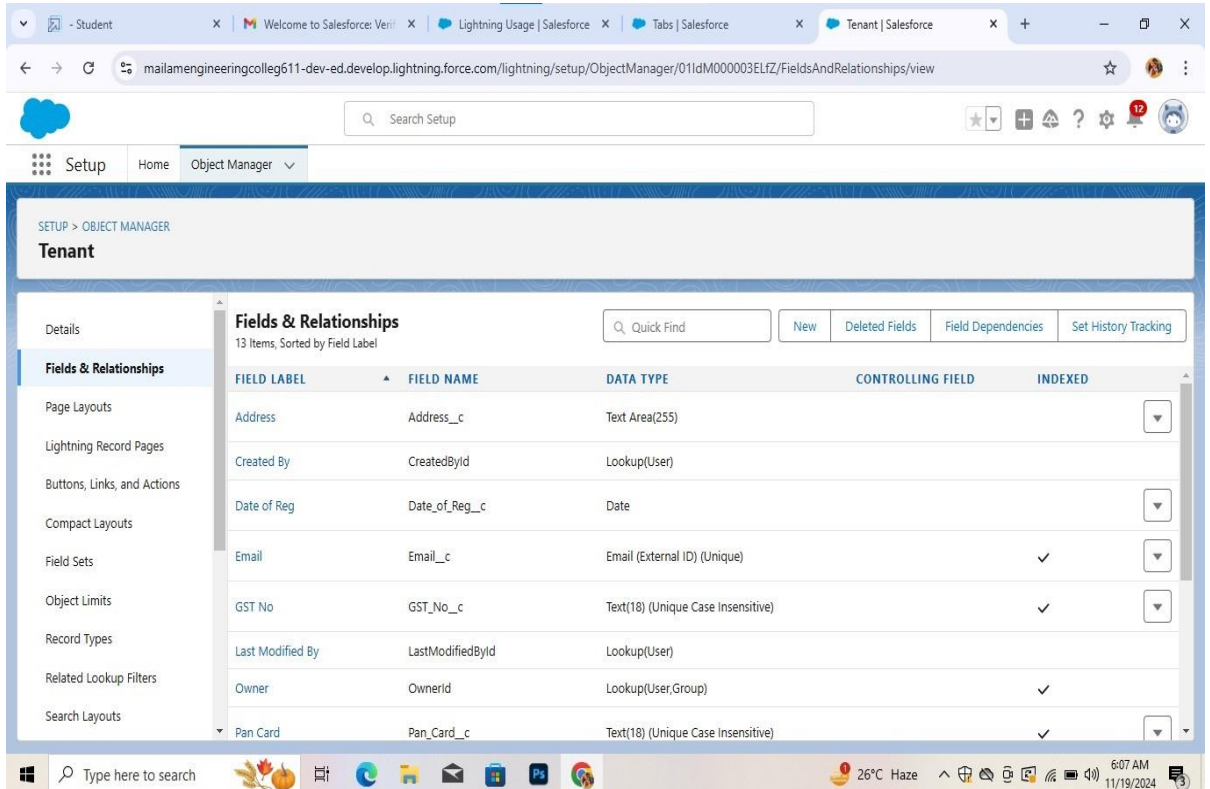


The screenshot shows the Salesforce Tabs setup page. The browser address bar displays the URL: mailamengineeringcolleg611-dev-ed.develop.lightning.force.com/lightning/setup/CustomTabs/home. The page header includes the Salesforce logo and a search bar. The main content area is titled "Custom Tabs" and provides instructions on how to create new custom tabs. The page is divided into three sections: Custom Object Tabs, Web Tabs, and Visualforce Tabs. The Custom Object Tabs section shows a table with the following data:

Action	Label	Tab Style	Description
Edit Del	Leases_Tracking	Bank	
Edit Del	Tenant_Issues	Books	
Edit Del	Tenants	Airplane	

The Web Tabs section shows a message: "No Web Tabs have been defined". The Visualforce Tabs section shows a message: "No Visualforce Tabs have been defined".

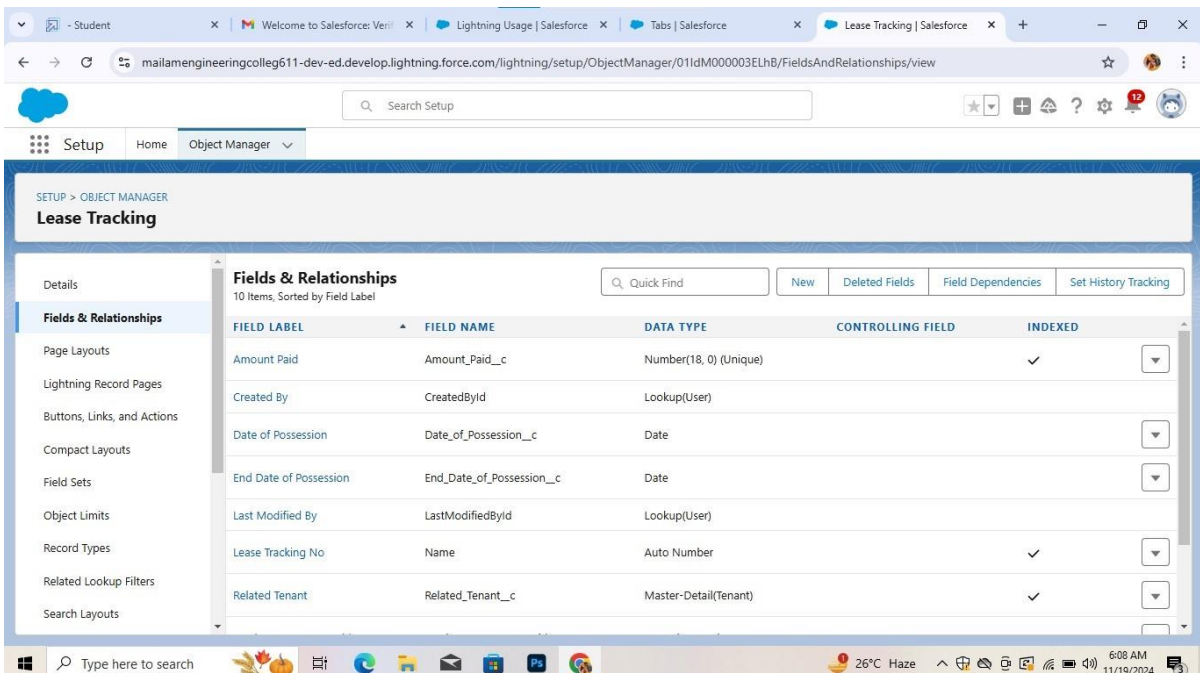
□ Create fields on tenant object:



The screenshot shows the Salesforce Setup interface for the 'Tenant' object. The 'Fields & Relationships' section is active, displaying a list of 13 fields. The fields are sorted by Field Label. The table below represents the data shown in the screenshot.

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Address	Address__c	Text Area(255)		
Created By	CreatedById	Lookup(User)		
Date of Reg	Date_of_Reg__c	Date		
Email	Email__c	Email (External ID) (Unique)		✓
GST No	GST_No__c	Text(18) (Unique Case Insensitive)		✓
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		✓
Pan Card	Pan_Card__c	Text(18) (Unique Case Insensitive)		✓

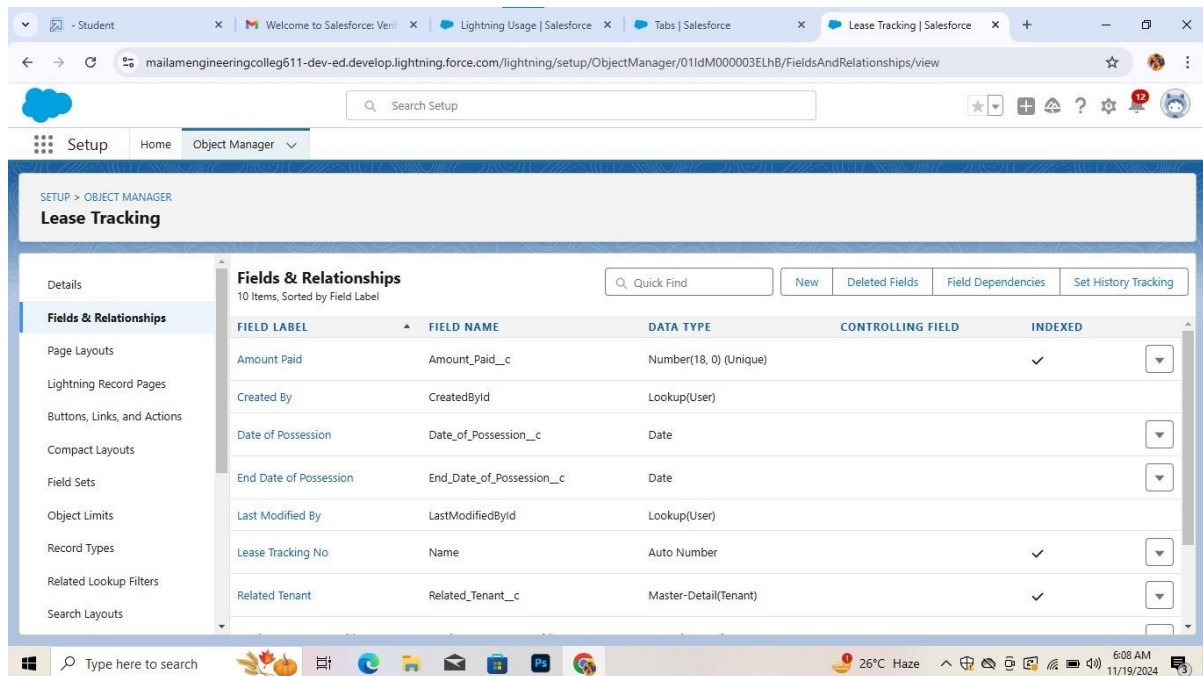
□ Create fields on Lease tracking object:



The screenshot shows the Salesforce Setup interface for the 'Lease Tracking' object. The 'Fields & Relationships' section is active, displaying a list of 10 fields. The fields are sorted by Field Label. The table below represents the data shown in the screenshot.

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Amount Paid	Amount_Paid__c	Number(18, 0) (Unique)		✓
Created By	CreatedById	Lookup(User)		
Date of Possession	Date_of_Possession__c	Date		
End Date of Possession	End_Date_of_Possession__c	Date		
Last Modified By	LastModifiedById	Lookup(User)		
Lease Tracking No	Name	Auto Number		✓
Related Tenant	Related_Tenant__c	Master-Detail(Tenant)		✓

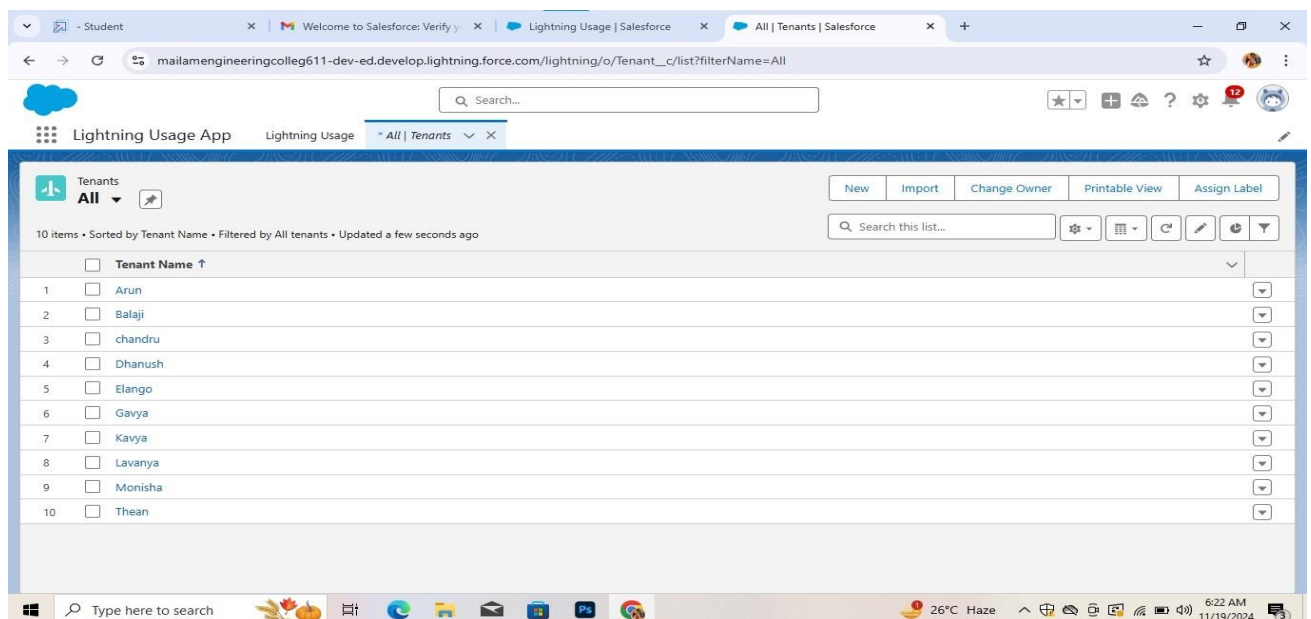
□ Create fields on tenant issues:



The screenshot shows the Salesforce Object Manager interface for the 'Lease Tracking' object. The 'Fields & Relationships' tab is selected, displaying a list of 10 fields. The fields are sorted by Field Label. The table below represents the data shown in the screenshot:

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Amount Paid	Amount_Paid__c	Number(18, 0) (Unique)		✓
Created By	CreatedById	Lookup(User)		
Date of Possession	Date_of_Possession__c	Date		
End Date of Possession	End_Date_of_Possession__c	Date		
Last Modified By	LastModifiedById	Lookup(User)		
Lease Tracking No	Name	Auto Number		✓
Related Tenant	Related_Tenant__c	Master-Detail(Tenant)		✓

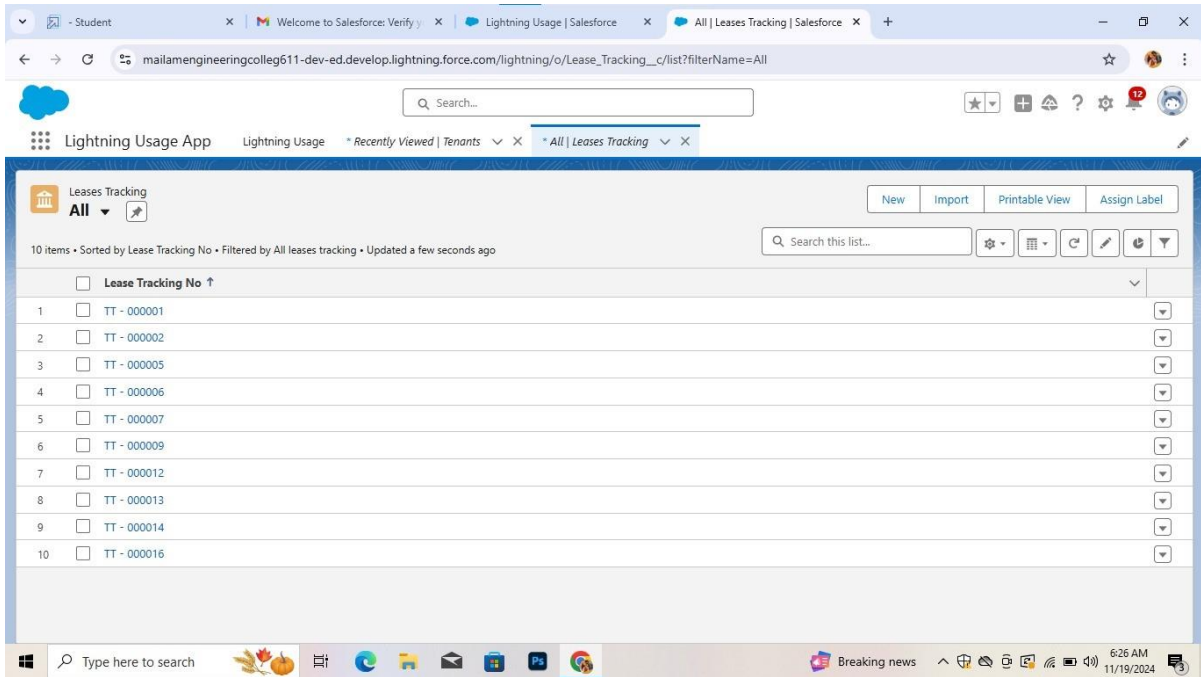
□ Inserting records in tenant object:



The screenshot shows the Salesforce Lightning Usage App interface for the 'Tenants' object. The 'All' filter is selected, displaying a list of 10 tenants. The table below represents the data shown in the screenshot:

Tenant Name
1. Arun
2. Balaji
3. Chandru
4. Dhanush
5. Elango
6. Gavya
7. Kavya
8. Lavanya
9. Monisha
10. Thean

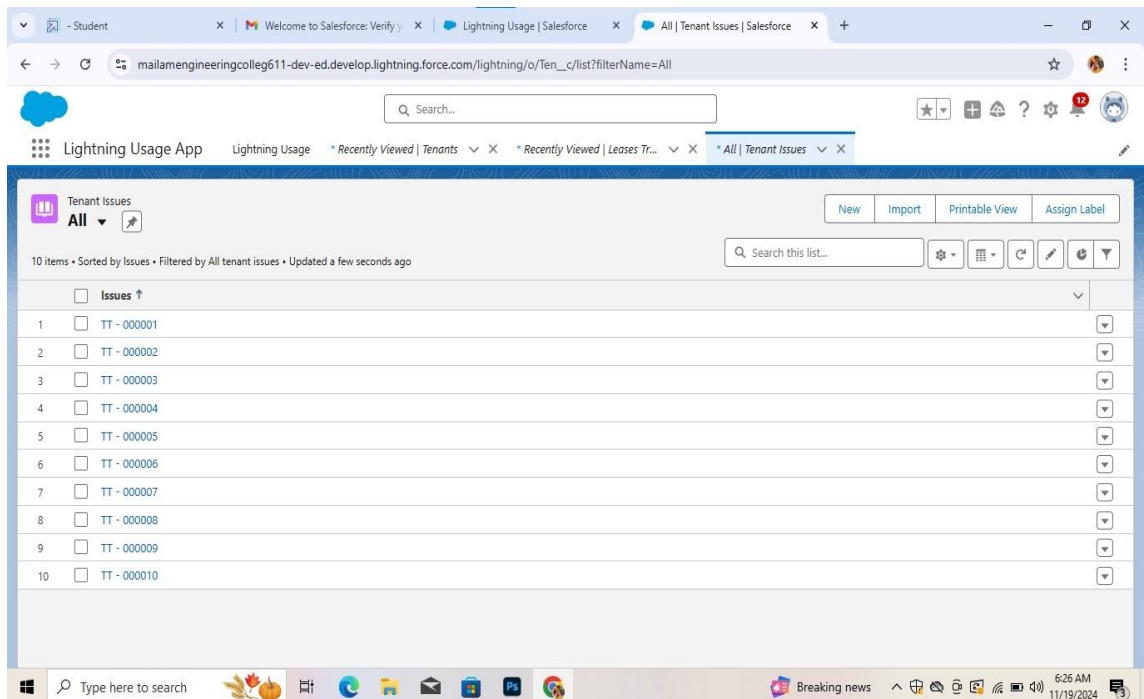
□ Inserting records in lease tracking object:



The screenshot shows the Salesforce Lightning Usage App interface. The browser address bar displays the URL: `mailamengineeringcolleg611-dev-ed.develop.lightning.force.com/lightning/o/Lease_Tracking__c/list?filterName=All`. The page title is "Lease Tracking" with a filter dropdown set to "All". The interface includes buttons for "New", "Import", "Printable View", and "Assign Label". A search bar is present with the text "Search this list...". Below the search bar, a table lists 10 items, sorted by "Lease Tracking No" in ascending order. The table columns are "Lease Tracking No" and a dropdown menu. The data rows are as follows:

Lease Tracking No	
1	TT - 000001
2	TT - 000002
3	TT - 000005
4	TT - 000006
5	TT - 000007
6	TT - 000009
7	TT - 000012
8	TT - 000013
9	TT - 000014
10	TT - 000016

□ Inserting records in tenant issues object:

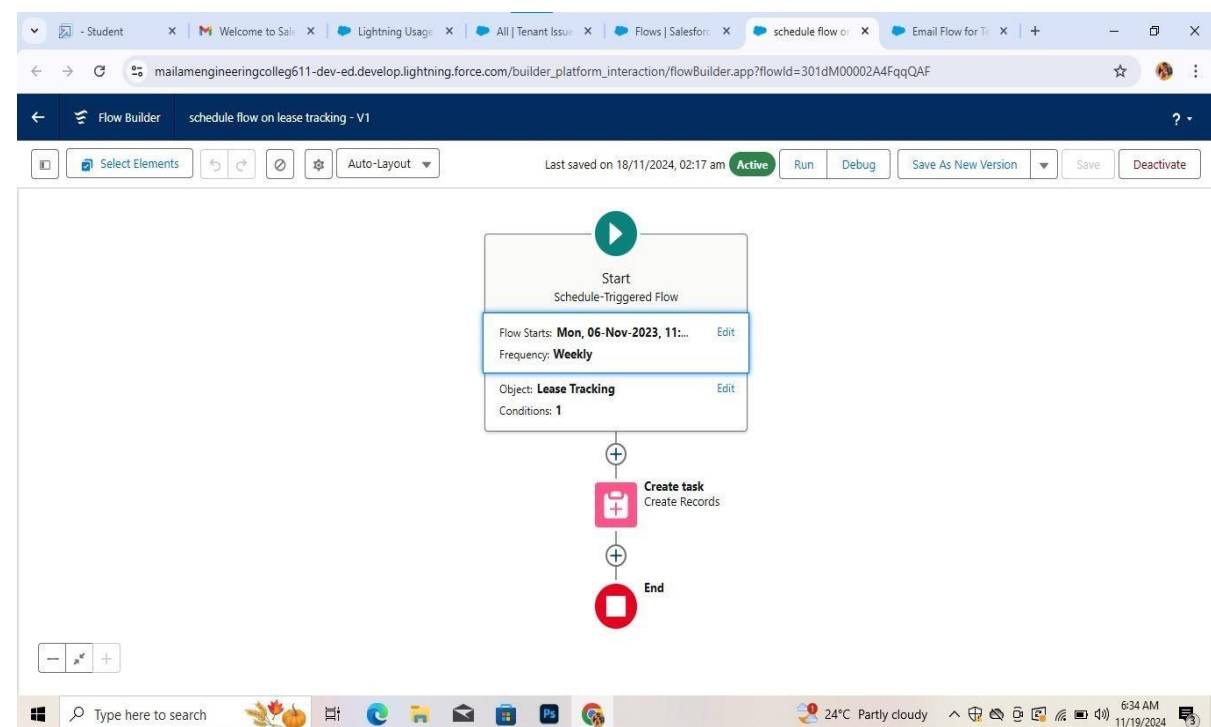
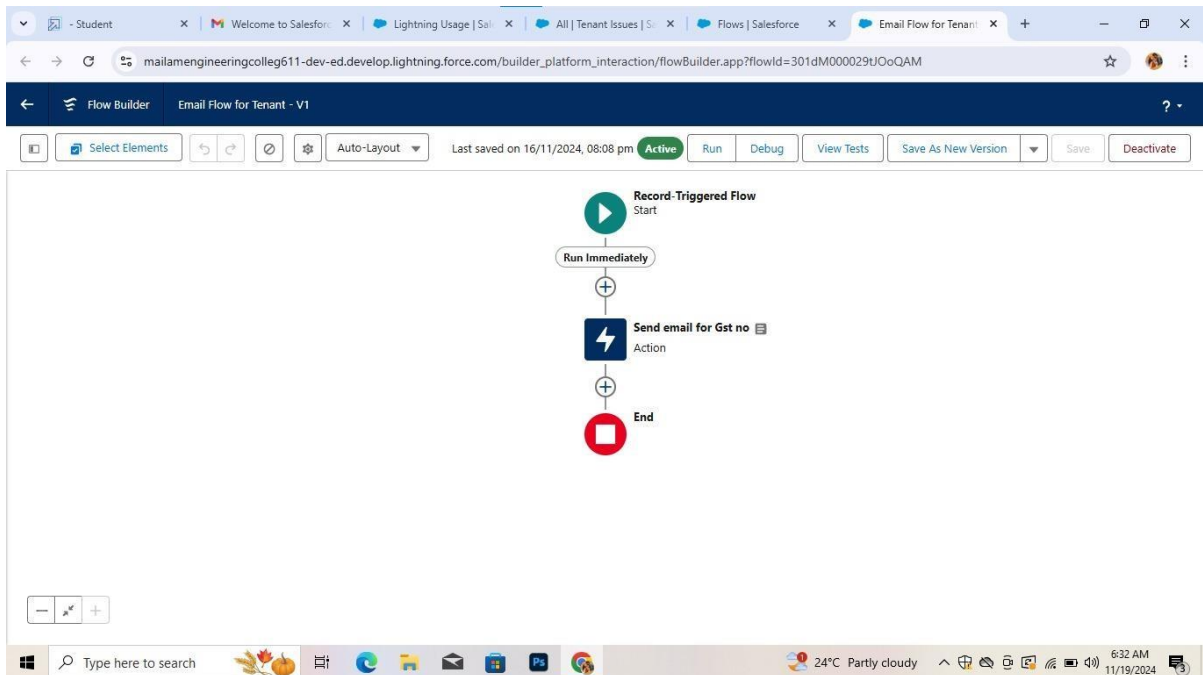


The screenshot shows the Salesforce Lightning Usage App interface for the "Tenant Issues" object. The browser address bar displays the URL: `mailamengineeringcolleg611-dev-ed.develop.lightning.force.com/lightning/o/Ten__c/list?filterName=All`. The page title is "Tenant Issues" with a filter dropdown set to "All". The interface includes buttons for "New", "Import", "Printable View", and "Assign Label". A search bar is present with the text "Search this list...". Below the search bar, a table lists 10 items, sorted by "Issues" in ascending order. The table columns are "Issues" and a dropdown menu. The data rows are as follows:

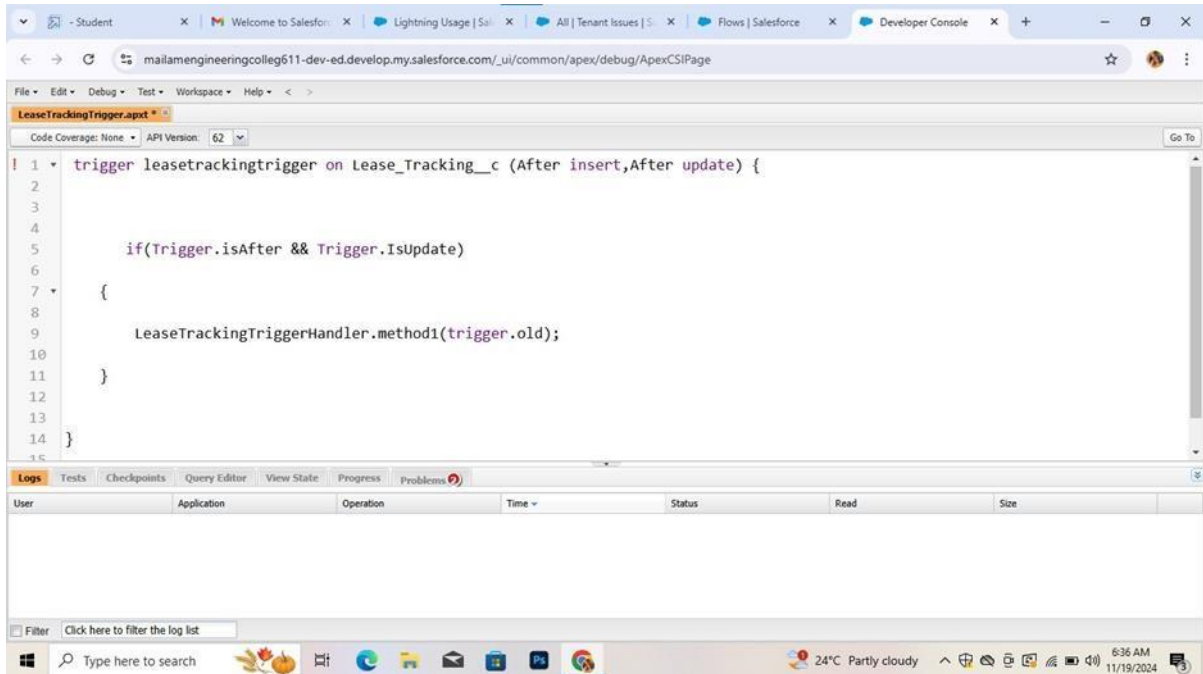
Issues	
1	TT - 000001
2	TT - 000002
3	TT - 000003
4	TT - 000004
5	TT - 000005
6	TT - 000006
7	TT - 000007
8	TT - 000008
9	TT - 000009
10	TT - 000010

□ Create Flows And Flows For Lease Tracking:

Below are detailed flows tailored to the key user groups: Mall Administrators, Store Managers, and Customers.



□ Lease tracking trigger:

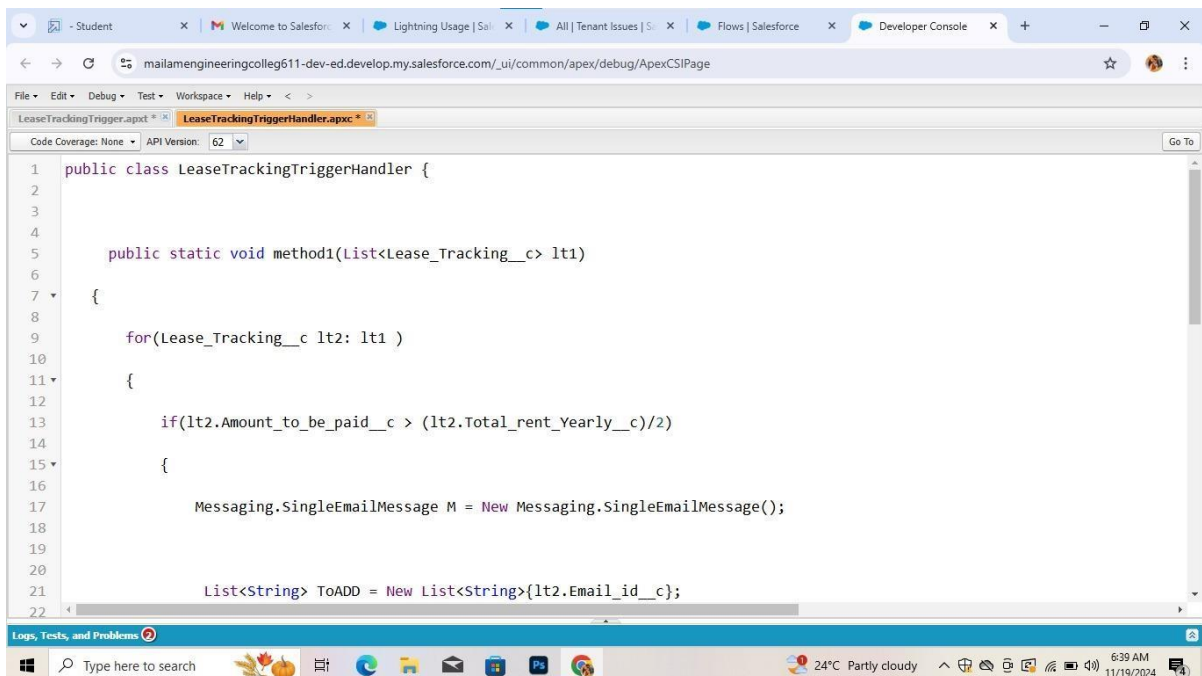


```

1 trigger leasetrackingtrigger on Lease_Tracking__c (After insert,After update) {
2
3
4
5     if(trigger.isAfter && Trigger.IsUpdate)
6
7     {
8
9         LeaseTrackingTriggerHandler.method1(trigger.old);
10
11     }
12
13
14 }
  
```

The screenshot shows the Salesforce Developer Console with the 'LeaseTrackingTrigger.apex' file open. The code is a trigger that fires after insert and after update on the Lease_Tracking__c object. It checks if the trigger is after and if it's an update, then it calls the method1 of the LeaseTrackingTriggerHandler class with the trigger.old record.

□ Lease tracking trigger handler:

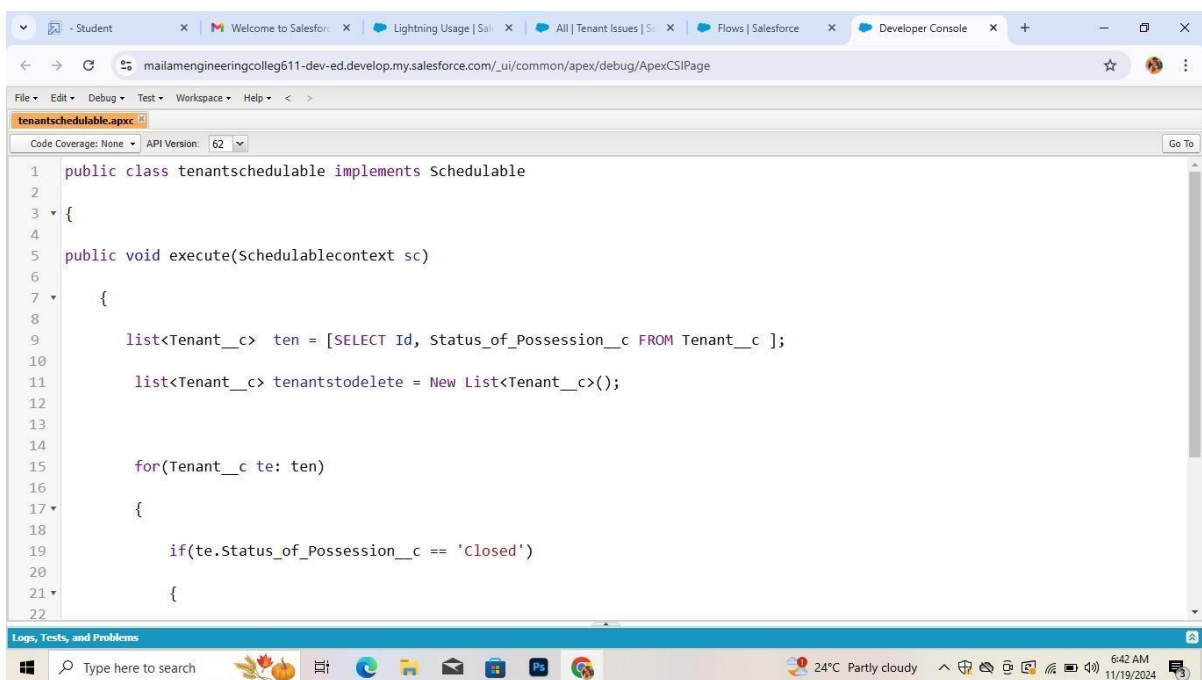
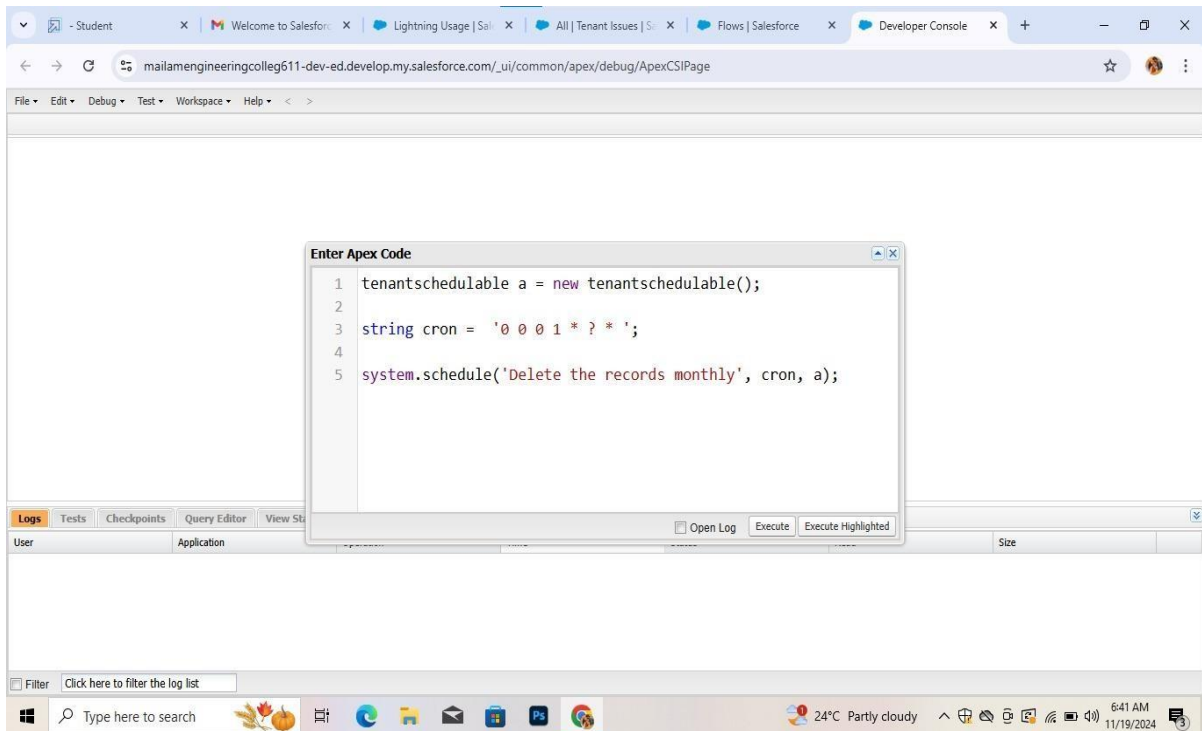


```

1 public class LeaseTrackingTriggerHandler {
2
3
4
5     public static void method1(List<Lease_Tracking__c> lt1)
6
7     {
8
9         for(Lease_Tracking__c lt2: lt1 )
10
11         {
12
13             if(lt2.Amount_to_be_paid__c > (lt2.Total_rent_Yearly__c)/2)
14
15             {
16
17                 Messaging.SingleEmailMessage M = New Messaging.SingleEmailMessage();
18
19
20
21                 List<String> ToADD = New List<String>{lt2.Email_id__c};
22
  
```

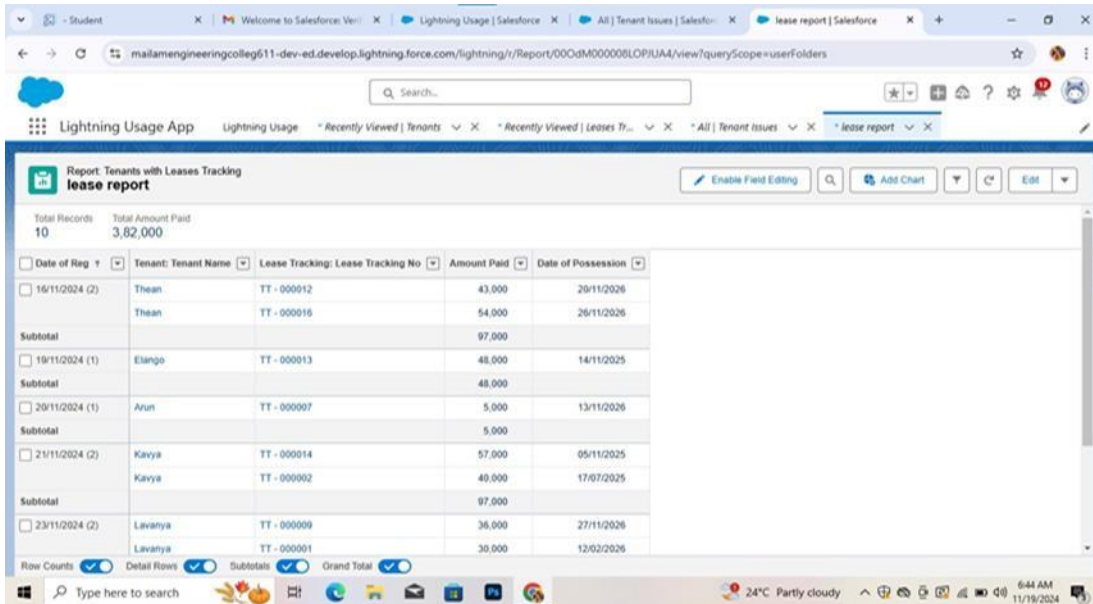
The screenshot shows the Salesforce Developer Console with the 'LeaseTrackingTriggerHandler.apex' file open. The code is a class with a static method method1 that takes a list of Lease_Tracking__c records. It iterates through the list and checks if the Amount_to_be_paid__c is greater than half of the Total_rent_Yearly__c. If so, it creates a Messaging.SingleEmailMessage object and sets the ToADD list with the Email_id__c of the current record.

□ Asynchronous apex:



□ Reports:

- Create A Report of lease Management Records:

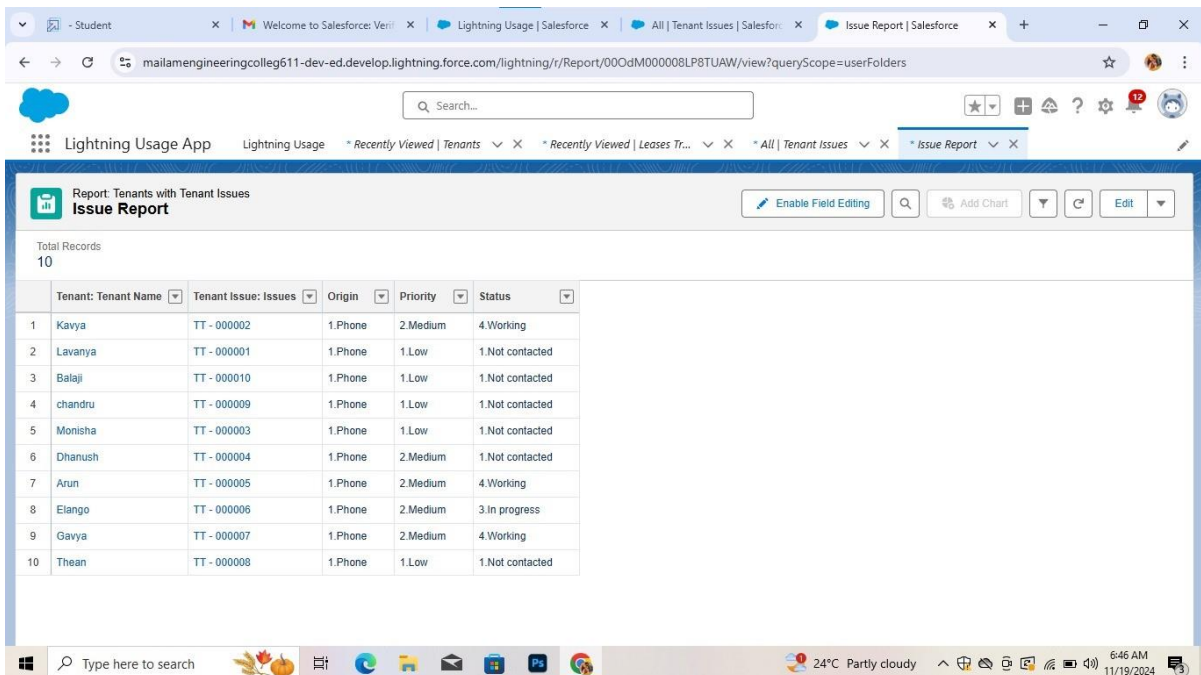


Report: Tenants with Leases Tracking
lease report

Total Records: 10
Total Amount Paid: 3,82,000

Date of Reg	Tenant: Tenant Name	Lease Tracking: Lease Tracking No	Amount Paid	Date of Possession
16/11/2024 (2)	Thean	TT - 000012	43,000	20/11/2026
	Thean	TT - 000015	54,000	26/11/2026
Subtotal			97,000	
19/11/2024 (1)	Elango	TT - 000013	48,000	14/11/2025
Subtotal			48,000	
20/11/2024 (1)	Arun	TT - 000007	5,000	13/11/2026
Subtotal			5,000	
21/11/2024 (2)	Kavya	TT - 000014	57,000	05/11/2025
	Kavya	TT - 000002	40,000	17/07/2025
Subtotal			97,000	
23/11/2024 (2)	Lavanya	TT - 000009	36,000	27/11/2026
	Lavanya	TT - 000001	30,000	12/02/2026
Grand Total				

- Create a report of Tenant issue:

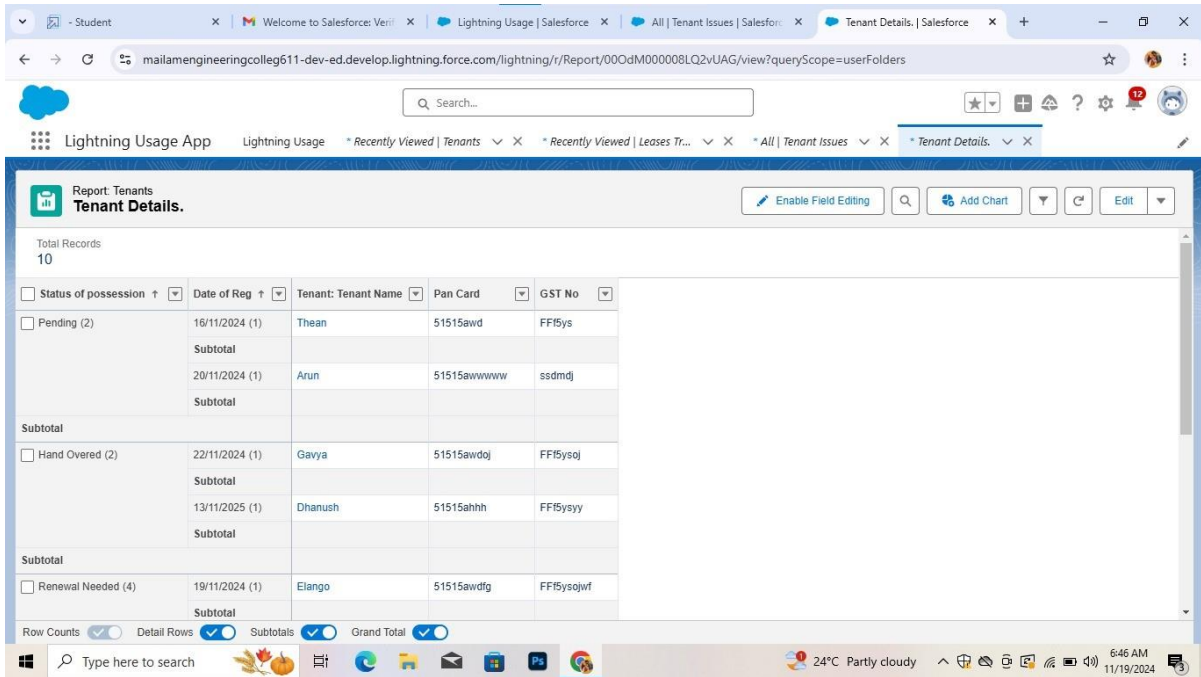


Report: Tenants with Tenant Issues
Issue Report

Total Records: 10

	Tenant: Tenant Name	Tenant Issue: Issues	Origin	Priority	Status
1	Kavya	TT - 000002	1.Phone	2.Medium	4.Working
2	Lavanya	TT - 000001	1.Phone	1.Low	1.Not contacted
3	Balaji	TT - 000010	1.Phone	1.Low	1.Not contacted
4	chandru	TT - 000009	1.Phone	1.Low	1.Not contacted
5	Monisha	TT - 000003	1.Phone	1.Low	1.Not contacted
6	Dhanush	TT - 000004	1.Phone	2.Medium	1.Not contacted
7	Arun	TT - 000005	1.Phone	2.Medium	4.Working
8	Elango	TT - 000006	1.Phone	2.Medium	3.In progress
9	Gavya	TT - 000007	1.Phone	2.Medium	4.Working
10	Thean	TT - 000008	1.Phone	1.Low	1.Not contacted

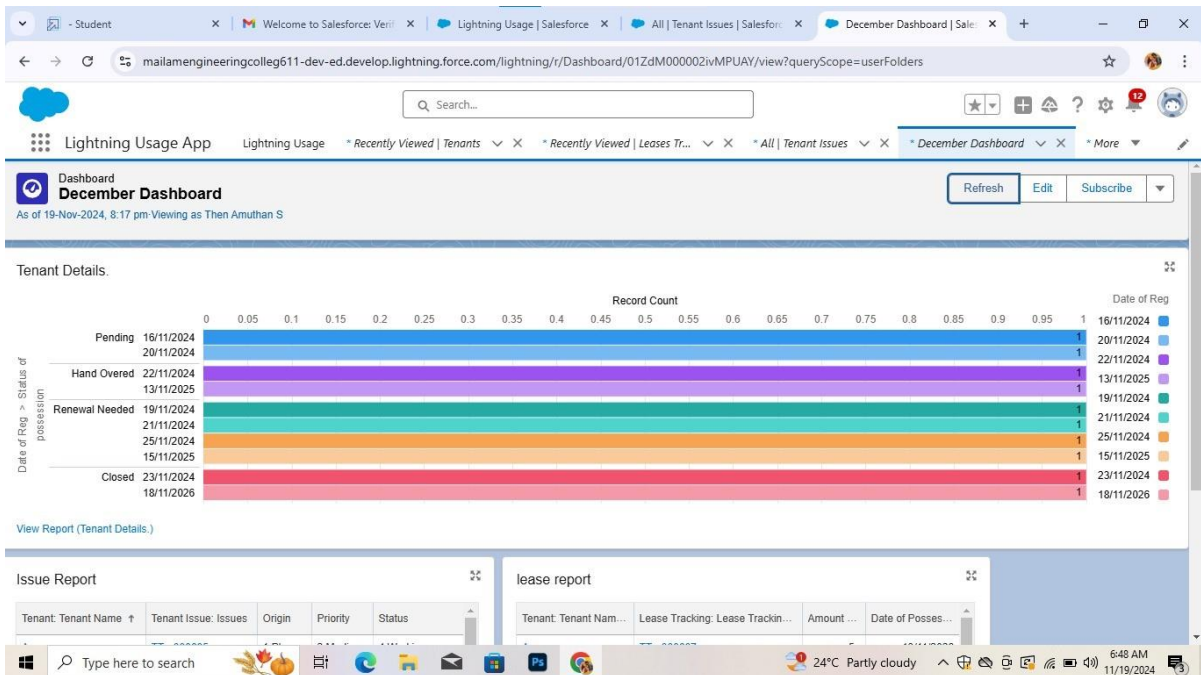
- **Create A Report on Tenant Records:**



The screenshot shows the Salesforce Lightning Usage App interface. The top navigation bar includes the Lightning Usage App and several tabs: Lightning Usage, Recently Viewed | Tenants, Recently Viewed | Leases Tr..., All | Tenant Issues, and Tenant Details. The main content area displays a report titled "Report: Tenants Tenant Details" with a total of 10 records. The report is organized into a table with columns for Status of possession, Date of Reg, Tenant: Tenant Name, Pan Card, and GST No. The data is grouped by status: Pending (2), Hand Overed (2), and Renewal Needed (4). Each group has a Subtotal row. The bottom of the report shows row counts and checkboxes for Detail Rows, Subtotals, and Grand Total.

Status of possession	Date of Reg	Tenant: Tenant Name	Pan Card	GST No
Pending (2)	16/11/2024 (1)	Thean	51515awd	FFf5ys
	Subtotal			
	20/11/2024 (1)	Arun	51515awwww	ssdmj
Hand Overed (2)	22/11/2024 (1)	Gavya	51515awdoj	FFf5ysoj
	Subtotal			
	13/11/2025 (1)	Dhanush	51515ahhh	FFf5ysy
Renewal Needed (4)	19/11/2024 (1)	Elango	51515awdfg	FFf5ysojwf
	Subtotal			

Dashboard:



The screenshot shows the Salesforce Lightning Usage App interface with a dashboard titled "December Dashboard". The dashboard displays a horizontal bar chart titled "Tenant Details" showing the record count for various statuses and dates. The x-axis represents the Record Count (0 to 1.0), and the y-axis represents the Date of Reg - Status of possession. The data is grouped by status: Pending, Hand Overed, Renewal Needed, and Closed. Each group has a bar for each date. The bottom of the dashboard shows a section for "Issue Report" and a "lease report" table.

Date of Reg - Status of possession	Record Count
Pending 16/11/2024	1
Pending 20/11/2024	1
Hand Overed 22/11/2024	1
Hand Overed 13/11/2025	1
Renewal Needed 19/11/2024	1
Renewal Needed 21/11/2024	1
Renewal Needed 25/11/2024	1
Renewal Needed 15/11/2025	1
Closed 23/11/2024	1
Closed 18/11/2026	1

4.Conclusion:

In conclusion, the CRM system allows for efficient management of customer relationships through personalized marketing, targeted promotions, loyalty programs, and responsive customer service.

Ultimately, the CRM application not only boosts customer satisfaction and loyalty but also contributes to the mall's overall growth and operational efficiency, making it an indispensable tool for modern retail management.