NAAN MUDHALVAN-SALESFORCE REPORT

LEASE MANAGEMENT

PROJECT CREATED BY

BE-V SEMESTER

SAI SARAN.V - 422422104015

PAVIYA DHARSHINI.V - 422422104016

ANITHA.M - 422422104017

ATCHAYA.S - 422422104018

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
UNIVERSITY COLLEGE OF ENGINEERING TINDIVANAM

ANNA UNIVERSITY ,CHENNAI -60025

Contents

1.Project overview

A lease management project involves creating a system or application to efficiently handle the processes related to leasing real estate properties, equipment, or other assets. The goal is to streamline and automate various tasks associated with lease agreements, ensuring accurate recordkeeping, compliance with regulations, and effective communication between parties involved.

2. Project Objectives (Points Only)

- **Automate Lease Tracking**

Set up a system to automatically track key lease details like start/ dates, payments, and renewals.

- **Streamline Approvals**

Implement a quick, easy approval process to reduce delays and manual work.

- **Improve Data Accuracy**

Enforce rules to ensure lease data is entered correctly and consistently.

- **Enable Timely Notifications**

Automatically send reminders for important lease dates, like renewals and payments.

- **Enhance Reporting and Visibility**

Provide clear reports and visibility into all lease data to help with decision-making.

Figure 1:

3. Salesforce

Salesforce is your customer success platform, designed to help you sell, service, market, analyze, and connect with your customers. Salesforce has everything you need to run your business from anywhere. Using standard products and features, you can manage relationships with prospects and customers, collaborate and engage with employees and partners, and store your data securely in the cloud. So what does that really mean? Well, before Salesforce, your contacts, emails, followup tasks, and prospective deals might have been organized something.

like this:

https://youtu.be/r9EX3lGde5k

4.Detailed Steps to Solution Design

Develop thorough documentation of the design, encompassing data models, user interface designs, and business logic. Ensure that all elements are accompanied by relevant screenshots.

➢ Creating Developer Account:

Creating a developer org in salesforce.

1. Go to https://developer.salesforce.com/signup
2. On the sign up form, enter the following details :

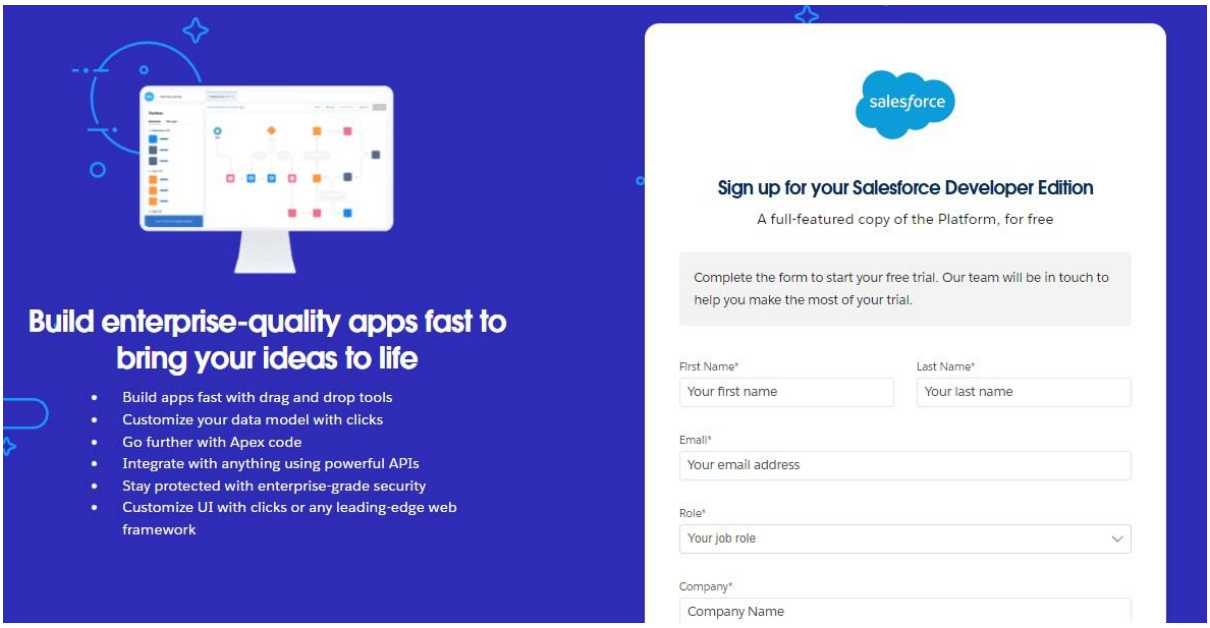

Figure 2

- First name & Last name
- Email
- Role : Developer
- Company : College Name
- County : India
- Postal Code : pin code
- Username : should be a combination of your name and company

This need not be an actual email id, you can give anything in the format

: username@organization.com

➢ Account Activation:

Go to the inbox of the email that you used while signing up. Click on the verify account to activate your account. The email may take 5-10mins.

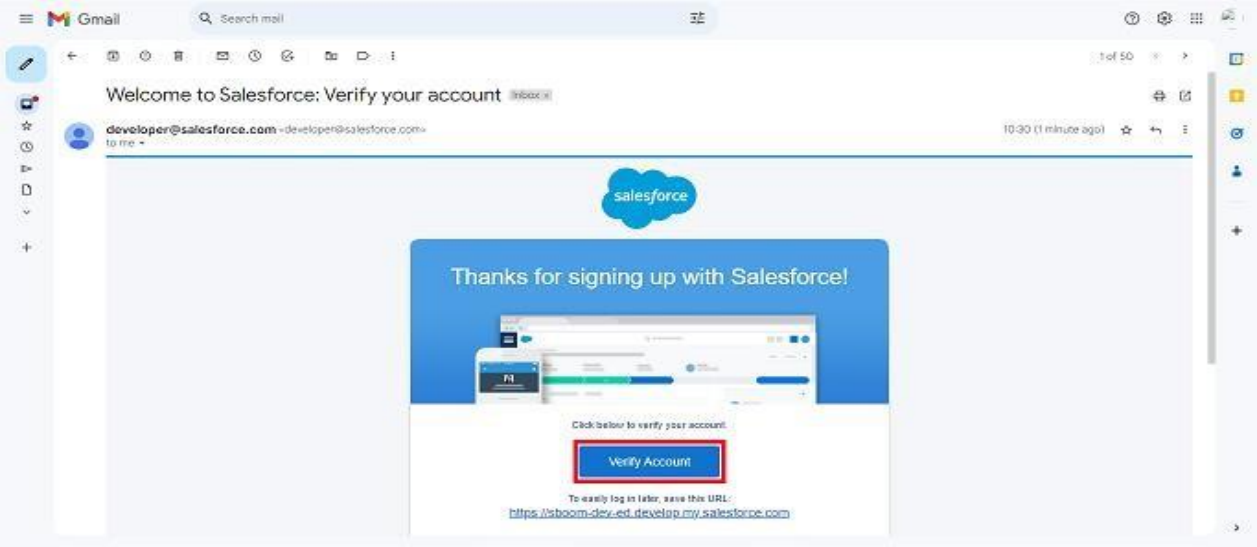

Figure 3

1.Click on Verify Account

2.Give a password and answer a security question and click on change password.

3.Give a password and answer a security question and click on change password.

4.Then you will redirect to your salesforce setup page.

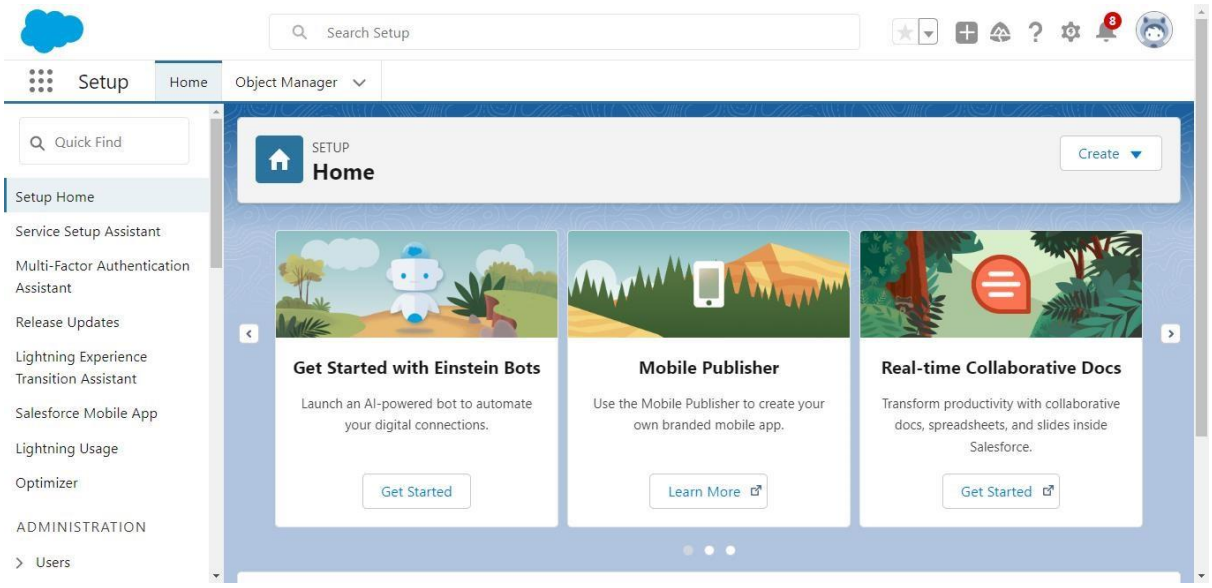

Figure 4

➢ Object

Salesforce objects are database tables that permit you to store data that is specific to an organization.

To create an object:

1.From the setup page ? Click on Object Manager ? Click on Create ? Click on Custom Object.

2.On Custom object defining page:

3.Enter the label name, plural label name, click on Allow reports, Allow search.

4.Click on Save.

Create Property Object:

To create an object:

1.From the setup page >> Click on Object Manager >> Click on Create >> Click on Custom Object.

1.Enter the label name>> property

2.Plural label name>> property

3.Enter Record Name Label and Format

- Record Name >>property Name
- Data Type >> Text

2.Click on Allow reports and Track Field History, Allow Activities

3.Allow search >> Save.

Create Tenant Object:

To create an object:

1.From the setup page >> Click on Object Manager >>Click on Create >> Click on Custom Object.

1.Enter the label name>> Tenant

2.Plural label name>> Tenants

3.Enter Record Name Label and Format

- Record Name >> Tenant Name
- Data Type >> Text

2.Click on Allow reports and Track Field History, Allow Activities

3.Allow search >> Save.

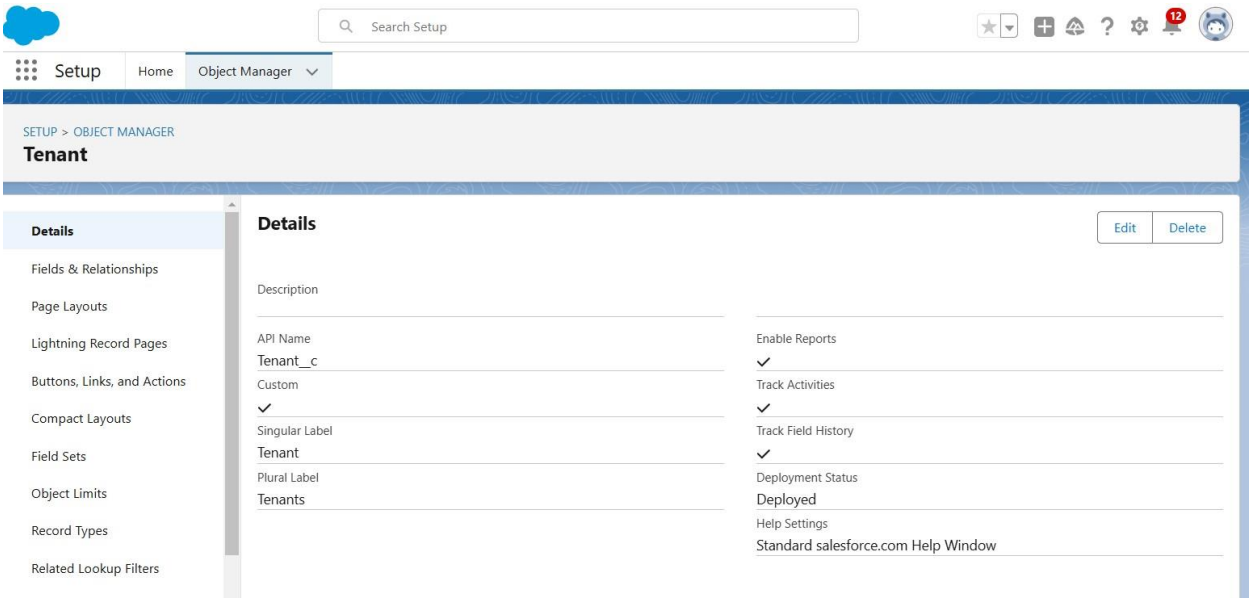

Figure 5

Create Payment Object

To create an object:

1.From the setup page >> Click on Object Manager >> Click on Create >> Click on Custom Object.

1.Enter the label name>> Payment for tenanat

2.Plural label name>> Payment

3.Enter Record Name Label and Format

- Record Name >> Payment Name
- Data Type >> Text

2. Click on Allow reports and Track Field History, Allow Activities Allow search >> Save.

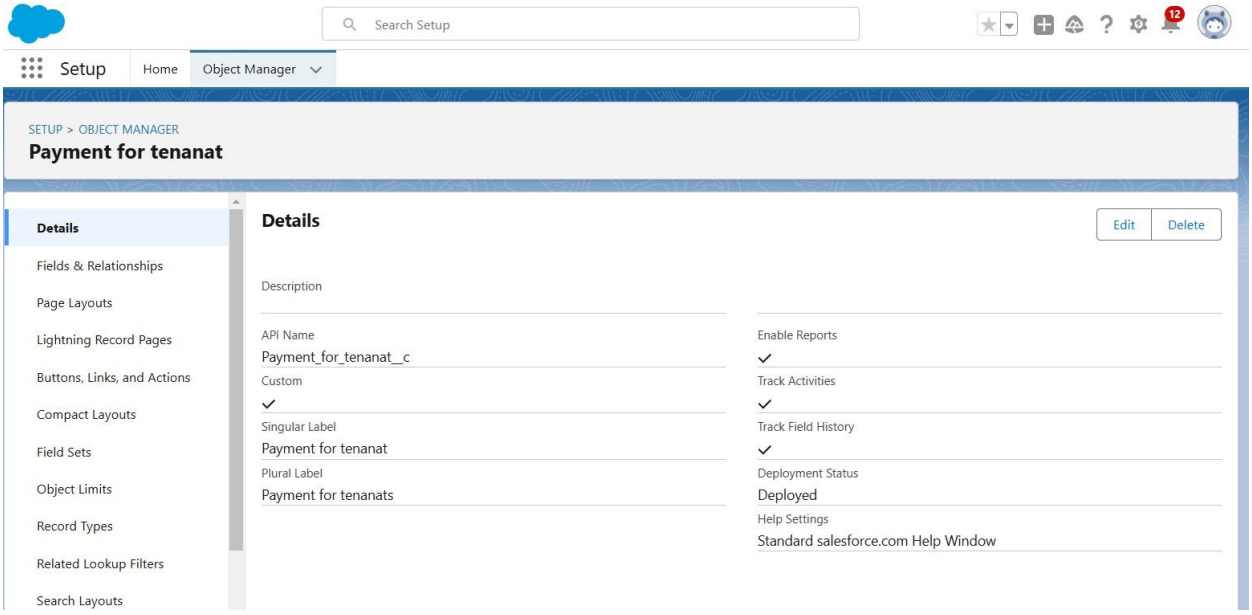

Figure 6

Create **Lease Object**

To create an object:

1.From the setup page >> Click on Object Manager >> Click on Create >> Click on Custom Object.

1.Enter the label name>> lease

2.Plural label name>> lease

3.Enter Record Name Label and Format

- Record Name >> lease Name
- Data Type >> Text

2.Click on Allow reports and Track Field History, Allow Activities

3.Allow search >> Save.

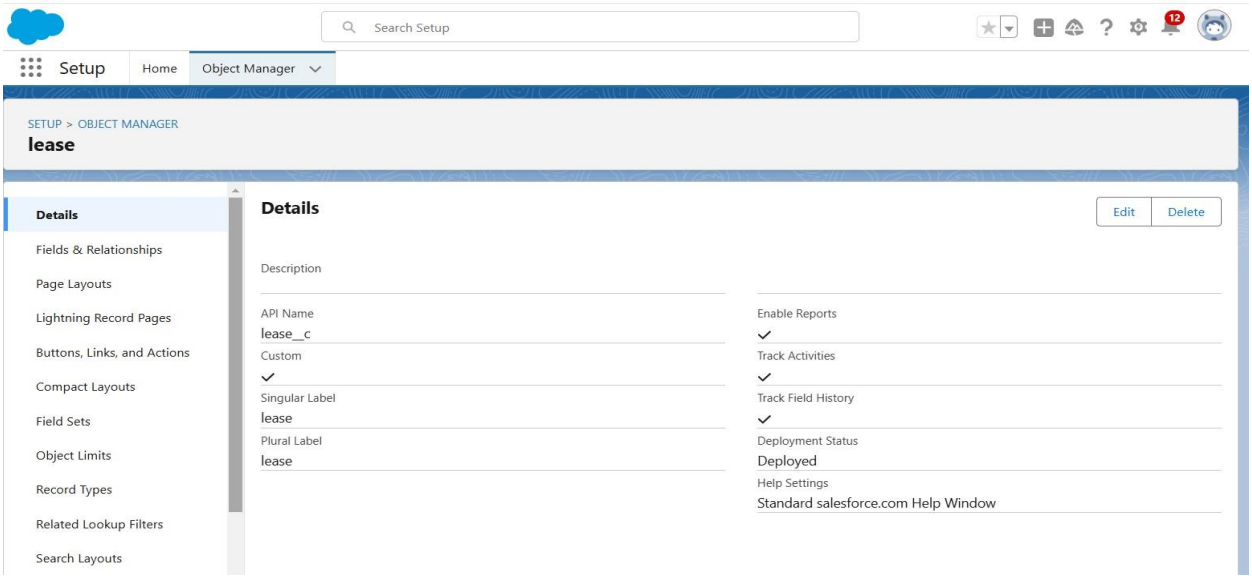

Figure 7

➢ Tabs

A tab is like a user interface that is used to build records for objects and to view the records in the objects.

Creating a Custom Tab

To create a Tab:(Property)

1. Select Object(property) >> Select the tab style >> Next (Add to profiles page) keep it as default >> Next (Add to Custom App) uncheck the include tab .
2. Make sure that the Append tab to users' existing personal customizations is checked.
3. Click save

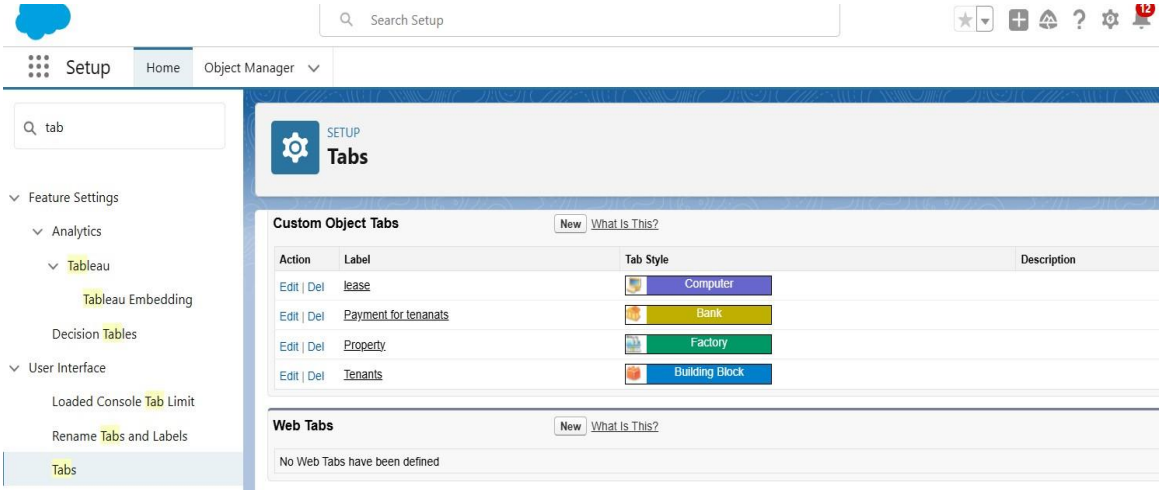

Figure 8

➢ The Lightning App

An app is a collection of items that work together to serve a particular function. In Lightning Experience, Lightning apps gives users access to sets of objects, tabs, and other items all in one convenient bundle in the navigation bar. Lightning apps let you brand your apps with a custom color and logo. You can even include a utility bar and Lightning page tabs in your Lightning app. Members of your org can work more efficiently by easily switching between apps.

Create a Lightning App

To create a lightning app page:

1.Go to setup page >> search “app manager” in quick find >> select “app manager” >> click on New lightning App.

2.Fill the app name in app details and branding as follow

- App Name : Lease Management
- Developer Name : This will auto populated

- Image : optional (if you want to give any image you can otherwise not mandatory) Primary colour hex value : keep this default.

3.Then click Next >> (App option page)Set Navigation Style as Standard Navigation >> Next.

(Utility Items) keep it as default >> Next.

4.To Add Navigation Items:

Search for the item in the (Payment for tenant, Tenants, property, lease) from the search bar and move it using the arrow button ? Next? Next.

5.To Add User Profile

Search profiles (System administrator) in the search bar >>click on the arrow button >> save & finish.

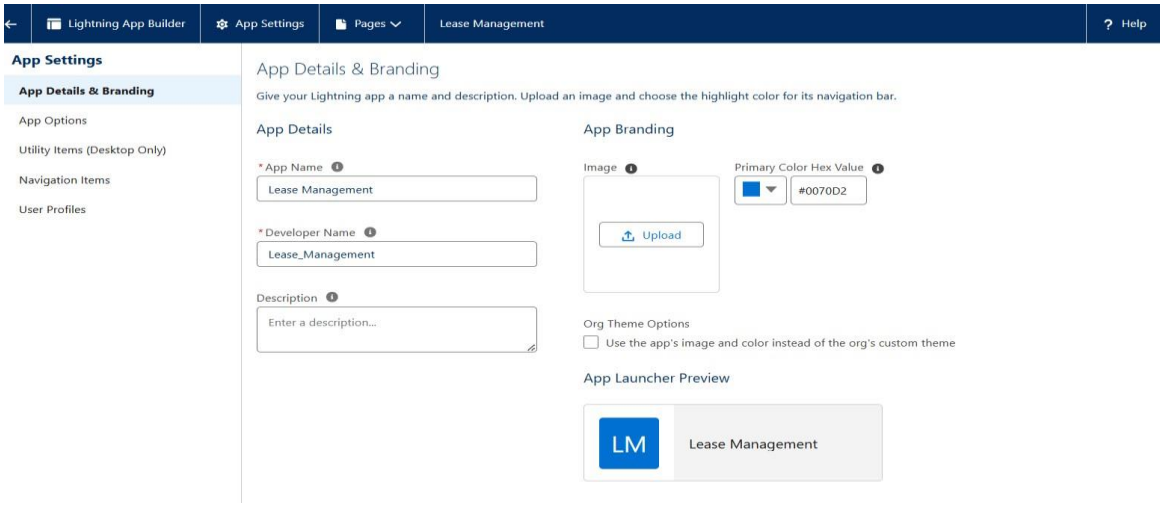

Figure 9

➢ Fields

When we talk about Salesforce, Fields represent the data stored in the columns of a relational database. It can also hold any valuable information that you require for a specific object. Hence, the overall searching, deletion, and editing of the records become simpler and quicker.

Creation of fields for the property object

To create fields in an object:
1.Go to setup >> click on Object Manager >> type object name(property) in search bar >>click on the object.

2.Now click on “Fields & Relationships” >> New

3.Select Data Type as “Text”

4.Click on next

5. Fill the Above as following

- Field Label: Name
- Field Name : gets auto generated
- Length : 25
- Required :check box
- Click on Next >> Next >> Save and new.

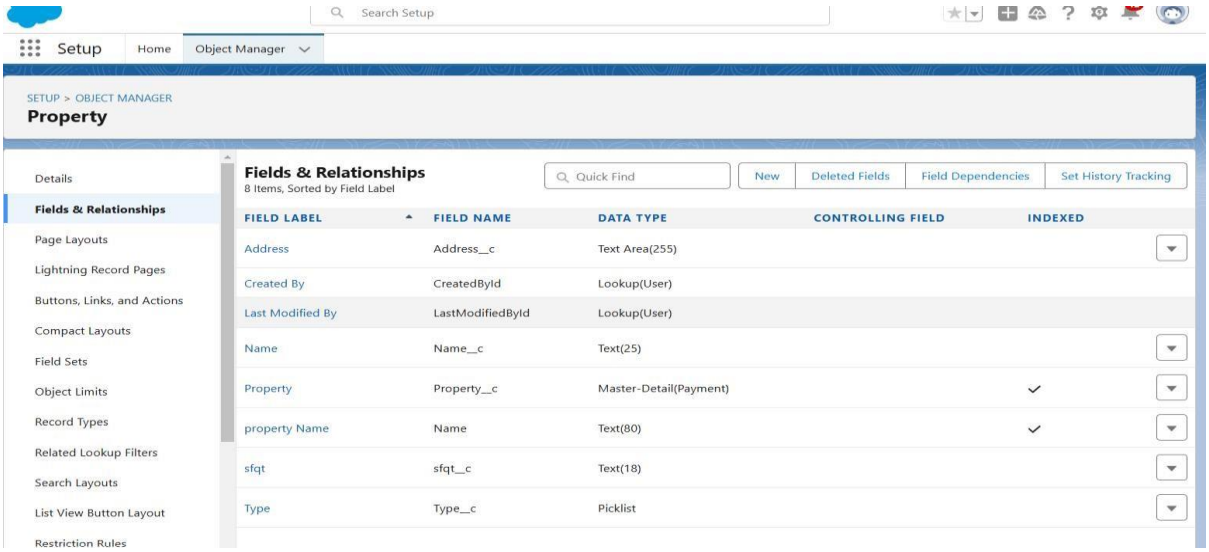

Figure 10

Creation of fields for the Tenant object

1. Go to setup >> click on Object Manager >> type object name(Tenant) in search bar >> click on the object.
2. Now click on “Fields & Relationships” >> New
3. Select Data type as a “Email” and Click on Next
4. Fill the Above as following:
    - Field Label : Email
    - Field Name : gets auto generated
    - Click on required check box
        - Click on Next >> Next >> Save and new.

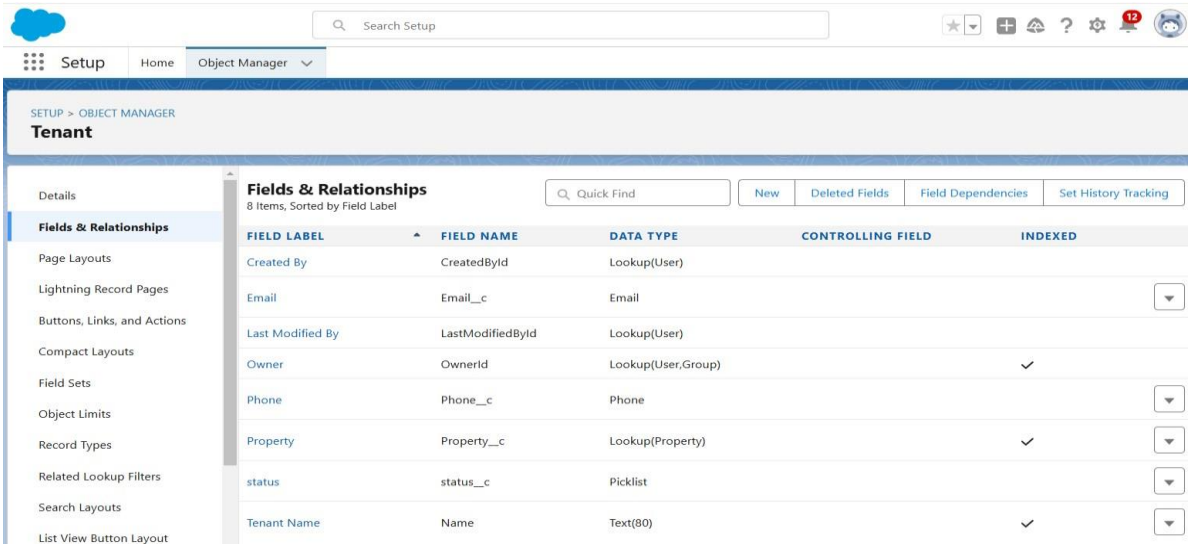

Figure 11

Creation of fields for the Lease object

1. Go to setup >> click on Object Manager >> type object name(Lease) in search bar >> click on the object.
2. Now click on “Fields & Relationships” >> New 3. Select Data type as a “Date” and Click on Next

4. Fill the Above as following:

- Field Label : start date
- Field Name : gets auto generated

5. Click on Next >> Next >> Save and new.

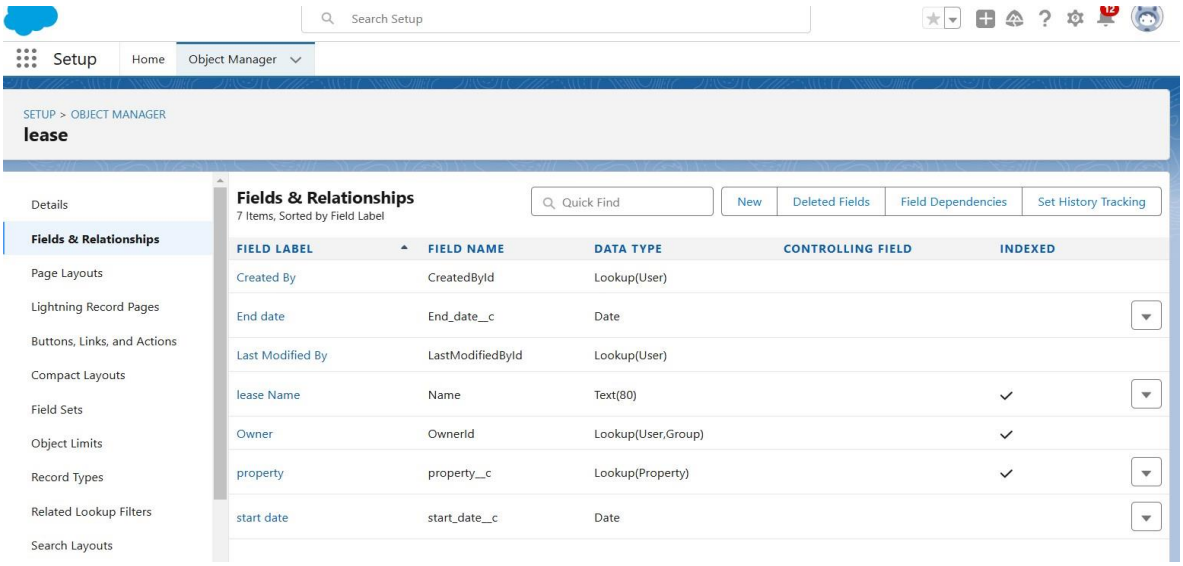

Figure 12

5.Validation rule

Validation rules are applied when a user tries to save a record and are used to check if the data meets specified criteria. If the criteria are not met, the validation rule triggers an error message and prevents the user from saving the record until the issues are resolved.

To create a validation rule to an Lease Object

1. Go to the setup page >> click on object manager >> From drop down click edit for Lease object.
2. Click on the validation rule >> click New.

3.Enter the Rule name as " lease_end_date".

4.Insert the Error Condition Formula as : End_date__c > start_date__c

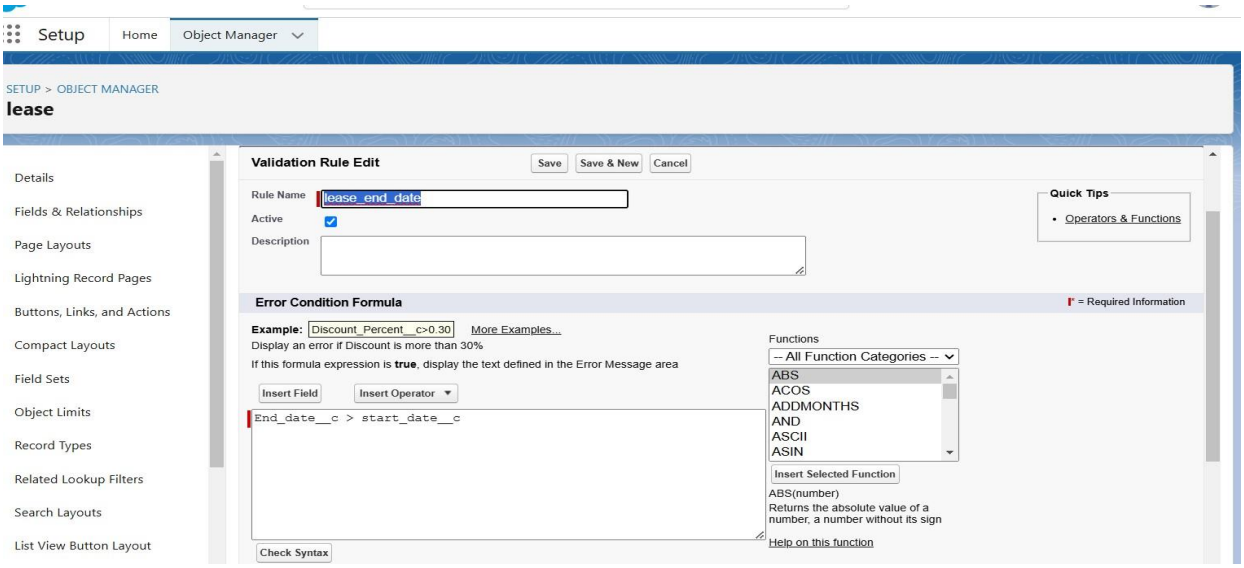

Figure 13

➢ Approval Process

The Approval Process is an automated process that an org uses to approve records in Salesforce. For example, When In the organization, someone is not able to decide a particular thing then he can ask someone else for approval. So, for such frequent cases or situations, one can define the approval process. So, Users can take benefit of such an approval process needed.

Create Approval Process For check for vacant

To create fields in an object:

1. Go to setup >> Approval Processes in quick find bar>>click on it.
2. Manage Approval Process For >> “Tenant” from the drop down.
3. Click on “Create New Approval Process” >> Use standard setup wizard.

4. Process Name “check for vacant” >> Click Next.
5. Field “Tenant: status” >> Operator : Not equals , Value >> Click on the lookup filter icon and select “Leaving”.
6. Click insert field, then click Next.

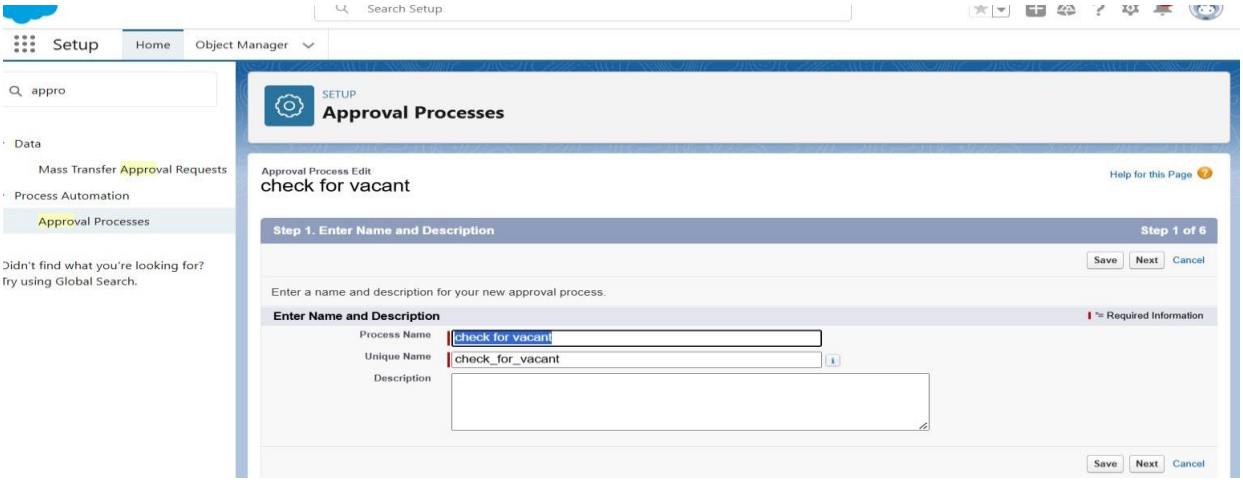

Figure 14

➢ Apex Trigger

The tenant and property are in a master-detail relationship, wherein each tenant is associated with only one property. When a tenant attempts to create a new record with an existing property, an error should be displayed, indicating that a tenant can have only one property.

Create an Apex Trigger

1. To create a new Apex Class follow the below steps:
2. Click on the file >> New ? Apex Class.

3. Give the Apex Trigger name as “test”, and select “Tenant__c” from the dropdown for sObject.
4. Click Submit.

Trigger Code:

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

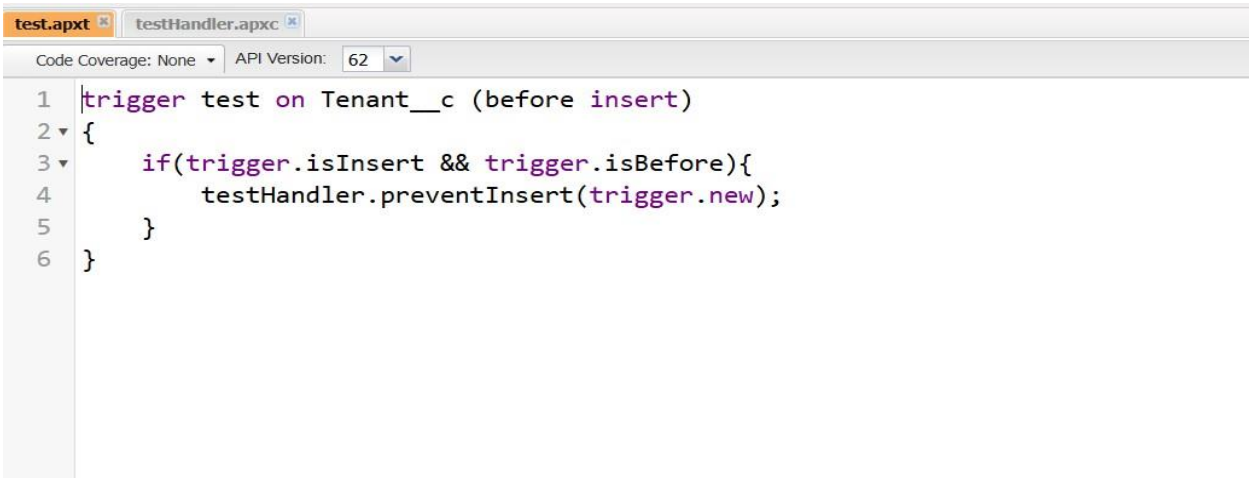

Figure 15

Create an Apex Handler class

To create a new Apex Class follow the below steps:

```
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');
        }
    }
  }
}
```

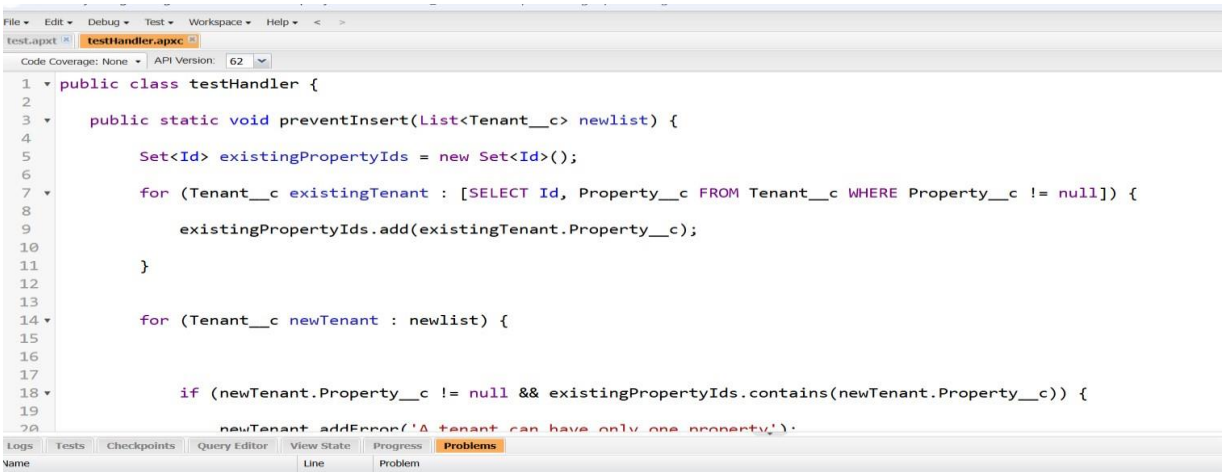

Figure 16

➢ FLOWS

In Salesforce, a flow is a tool that automates complex business processes. Simply put, it collects data and then does something with that data. Flow Builder is the declarative interface used to build individual flows

Create Flow for monthly payment

1. Go to setup >> type Flow in quick find box >> Click on the Flow and Select the New Flow.
2. Select the record Triggered flow. Click on create.
3. Under Object select ”Payment for tenant”. Click on A record is updated.
4. Set Entry Conditions
5. Under Condition Requirements
6. All Conditions are met

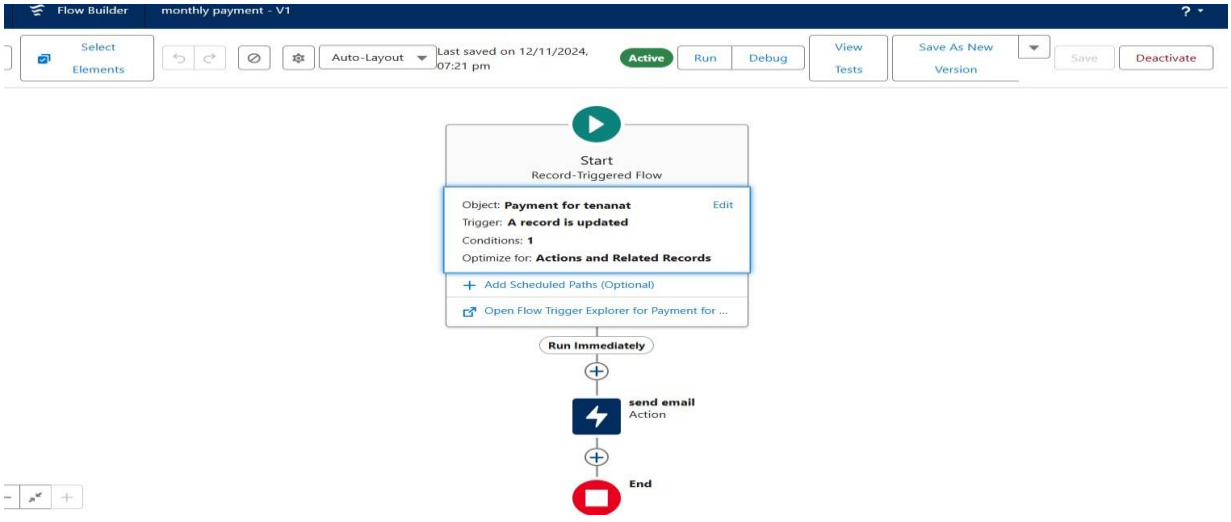

Figure 17

- **Schedule class**

Create an Apex Class

To create a new Apex Class follow the below steps:

```
  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});
```

```
    }
  }
}
```

Save the code.

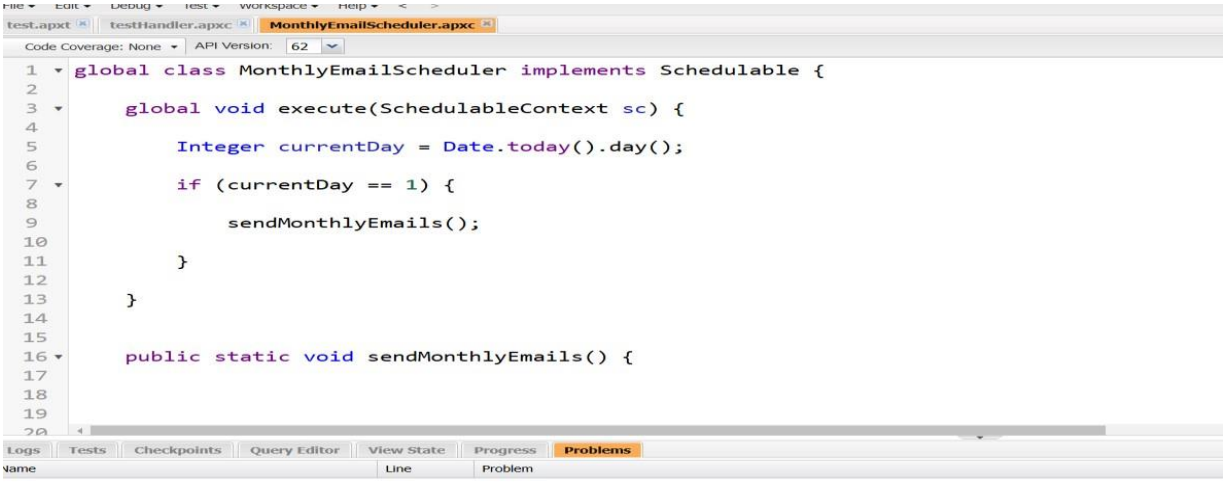

Figure 18

Schedule Apex class

1. Enter Apex class in quick find box
2. Select schedule Apex

    - Enter job Name : MonthlyEmailScheduler
    - Apex class : MonthlyEmailScheduler
    - Frequency : Monthly===>select on day 1
    - Start date : 04/12/2023
    - End date : 04/01/2024
    - Preferred start time : 09:00 am

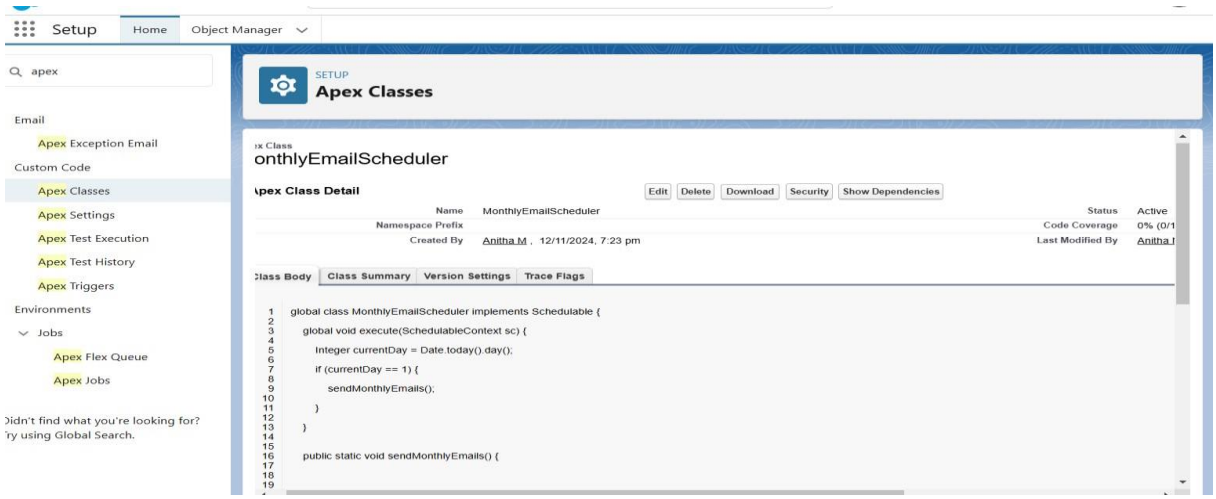

Figure 19

6.Key Scenarios Addressed by Salesforce

1. Automated Lease Tracking

Salesforce allows for tracking all lease agreements with details like start and end dates, payment schedules, and renewal terms. This ensures all data is accessible, up-to-date, and easy to manage.

2. Streamlined Approval Workflows

Using Salesforce's approval processes, multi-step approvals can be automated, making it easy for relevant stakeholders to review and approve leases without manual follow-ups. This reduces approval time and ensures that all leases go through proper authorization.

3.Data Validation and Consistency

Salesforce provides validation rules that enforce data accuracy, ensuring that key fields are correctly populated. This reduces data entry errors, maintains consistency, and improves the reliability of lease information.

4. Automated Notifications and Reminders

Salesforce Flows and email templates enable automated reminders for lease renewals, payment due dates, and expirations, helping the team stay proactive and avoid missed deadlines.

7.Conclusion

The Lease Management System has successfully automated and streamlined key leasing processes. It has reduced administrative work, improved data accuracy, and allowed for better visibility of lease agreements. Future enhancements could include billing integrations and advanced analytics.
