

Project Title: Lease Management System
Platform: Salesforce CRM

1. Introduction

The Lease Management System was developed to streamline the process of managing property and equipment leases. It enables organizations to maintain lease agreements, tenant details, payment schedules, renewals, and compliance tracking all in one platform.

Business Objectives:

Centralize tenant and property information.

Track lease agreements and contract details.

Automate rent invoicing and reminders.

Monitor payment status and overdue accounts.

Generate reports for financial and operational insights.

Tools & Features Used: Salesforce Custom Objects, Flows, Validation Rules, Approval Processes, Apex Triggers, Reports & Dashboards.

2. App Overview

A custom app named Lease Management was developed and added to the App Launcher.

Custom Objects Created

1. Property – Records property details (Name, Location, Type, Size).
2. Tenant – Stores tenant information (Name, Contact, Email, Address).
3. Lease Agreement – Captures agreement details (Start Date, End Date, Rent Amount, Security Deposit).

4. Payment – Tracks rent payments (Payment Date, Amount, Status, Linked Tenant).

5. Invoice – Auto-generated for rent charges.

Standard Objects Used

Users

Reports

Dashboards

3. User Interface Demonstration

Property Management: Create and track available/leased properties.

Tenant Records: Store tenant details and link them with lease agreements.

Validation Rules:

Lease End Date must be greater than Start Date.

Payment Amount must equal or exceed the rent due.

Dynamic Forms: Payment form shows only pending invoices for selection.

4. Business Process Automation

Flows:

Auto-generate invoices monthly based on active leases.

Auto-update payment status once recorded.

Workflow Rules:

Send reminders to tenants before due dates.

Approval Process:

Approval required for rent discounts or contract extensions.

Apex Triggers:

Auto-calculate late fees for overdue payments.

5. Reports & Dashboards

Reports:

Active and expired leases.

Rent collection by month.

Overdue payments and pending invoices.

Dashboards:

Financial overview (monthly revenue, overdue amounts).

Occupancy rates of properties.

6. User Management & Security

Profiles & Permission Sets:

Property Manager: Full access to leases and tenants.

Accountant: Payment and invoice access only.

Admin: Full system access.

Role Hierarchy: Maintains data security by role.

Field History Tracking: Enabled for Lease Agreement and Payment objects.

7. Error Handling & Debugging

Debug logs used to resolve automation issues.

Flow Debugger applied to check invoice automation.

Errors in late fee calculation fixed using test cases.

8. Highlights

Automated monthly invoice generation.

Late fee calculation for overdue payments.

Lease renewal tracking with reminders.

Centralized storage of tenant and property data.

9. Testing Approach

Functional Testing: Tested lease creation, invoice generation, and payment updates.

Integration Testing: Verified links between Lease, Tenant, and Payment objects.

User Testing: Confirmed functionality for Property Manager and Accountant roles.

10. Future Enhancements

Integration with online payment gateways for direct rent collection.

Mobile app for tenants to view invoices and pay rent.

AI-based forecasting of property utilization and rental trends.

Chatbot integration for tenant support.

11. Conclusion

The Lease Management System provides an efficient way to manage leases, track payments, and automate invoicing. It reduces manual workload, prevents delays, and enhances financial control. The project was fully implemented, tested, and documented, with opportunities for future upgrades.