

# PROJECT DESIGN PHASE

## PROBLEM-SOLUTION FIT TEMPLATE

Date: 06 NOVEMBER 2025

Team ID: NM2025TMID07231

Maximum Marks: 4 Marks

PROJECT NAME LEASE MANAGEMENT SYSTEM

### Problem — Solution Fit Template

The Problem—Solution Fit ensures that the identified problem faced by users Or administrators in the leasing process is effectively addressed by your proposed system. It helps align your digital solution with user needs and operational workflows to ensure efficiency, transparency, and ease Of management.

#### Purpose:

- Z Simplify and automate the lease tracking and management process.
- ✓ Enhance efficiency by eliminating manual errors and delays in lease renewals or payments.

1. Requirement Gathering Understand issues like missed payments, expired leases, and manual tracking difficulties	2. Plan Development Design a cloud-based Salesforce application to automate lease creation, payment tracking, and renewal alerts
--	---

<b>4.</b> Tree-inmanagemen Develop the Lease Object, add custom fields. and automate processes using Flows and Validation Rules	<b>5.</b> Testing & Validation Test lease creation, renewal alerts, and payment updates to ensure relliability and reporting efficiency
--	--

- ✓ Improve transparency for both tenants and property managers.
- Z Provide timely alerts, payment tracking. and renewal reninders.
- ✓ Support decision—making through reai-time reports and dashboards.

### Prevent Lease Mismanagement or Delays

<b>DATE</b>	11 November 2025
<b>TEAM ID</b>	NM2025TMID04855
<b>PROJECT NAME</b>	Lease Management System
<b>MAXIMUM MARKS</b>	4 Marks

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	In real estate and asset leasing, managing leases manually or across disparate systems leads to inefficiencies, data errors, compliance risks, and difficulty tracking lease agreements.

2.	Idea / Solution Description	A Salesforce-based lease management system that automates tracking of properties, tenants, payments, and leases. It leverages custom objects, validation rules, flows, and approval processes to streamline lease workflows and ensure data integrity.
3.	Novelty / Uniqueness	Integrates leasing processes into a single platform with native Salesforce tools, no external plugins, customizable for different asset types, and supports automated alerts, approvals, and reporting.
4.	Social Impact / Customer Satisfaction	Promotes accountability and transparency among property managers, tenants, and finance teams, reducing lease violations, missed payments, and errors that impact customer satisfaction.
5.	Business Model (Revenue Model)	While not directly revenue-generating, the system saves administrative costs, reduces lease disputes, and enhances property utilization efficiency, delivering financial benefits indirectly.

6. Scalability of the Solution Can be extended to manage various asset classes beyond real estate, integrate role-based access controls for teams, support multinational lease regulations, and expand reporting capabilities.

## Conclusion

The "Lease Management" solution addresses critical gaps in lease and regulations, management by centralizing leases, payments, and tenant on one Automated form. This reduces errors, enhances compliance and streamline roles communication between stakeholders. workflows and efficiency data rules ensure data accuracy and lease lifecycle diverse management, fostering better operational and tenant satisfaction. a scalable solution, it can adapt to organizational needs and all portfolios.

## Project Design Phase-II

### Data Flow Diagram & User Stories

#### Data Flow Diagrams

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It can be manual, automated or a combination of both. It shows how information enters and leaves the system, what changes the information, and where data is stored.

For the project "Lease Management System", Data Flow Diagrams (DFDs) illustrate how user deletion requests are processed within the system. The DFD shows the interaction between the admin, the system, and the incident database to validate whether a user is assigned to any active matters. If a user is linked to an incident, the system blocks deletion and sends a notification explaining why the user cannot be deleted.

#### Example Data Flow Diagram

#### Context Diagram (Level 0 DFD)

The context diagram shows the Lease Management System as a single process with external entities:

External Entities:

- Admin/Manager
- Tenant
- Property Owner
- Payment Gateway Data Flows:
- Property & Tenant Information →+ System
- Lease Contracts & Approvals →+ System
- Payment Records →+ System
- Notifications & Reports ←- System

## Level 1 DFD - Lease Management System

### Main Processes:

1. Manage Properties: Admin creates and updates property records (property details, address, type, sqft)
2. Manage Tenants: Admin adds, edits tenant information (name, email, phone, status)
3. Create Lease Contracts: Manager initiates lease agreements linking tenants to properties
4. Process Payments: System records payment data and validates payment status
5. Send Notifications: Automated email alerts for approvals, payments, and lease expiry
6. Generate Reports: Dashboard and analytics for occupancy, revenue, and payment tracking

### Data Stores:

- Property Database
- Tenant Database
- Lease Database
- Payment Database Data

### Flow Example:

- Admin Property Info Manage Properties Property Database
- Tenant —+ Payment Info Process Payments Payment Database Email Notification —Y Tenant
- Manager Lease Request Create Lease Lease Database Approval Notification Manager

## Level 2 DFD - Payment Processing Module Sub-processes:

1. Validate Payment Data: Check tenant ID, property ID, amount, and payment date
  2. Record Payment: Store payment record in Payment Database
  3. Update Payment Status: Mark as "Paid" or "Not Paid"
  4. Trigger Notification: Send confirmation email to tenant if payment successful
  5. Generate Payment Report: Create payment history for tenant
- Data Flows:
- Tenant submits payment Validate Payment Data —+ Check Tenant & Property Database
  - Valid payment Record Payment Payment Database
  - Payment successful Update Status —+ Trigger Email Notification Tenant receives confirmation

## User Stories

User stories define what different users need from the system in simple, goal-focused language. In this project, they help ensure the system blocks unnecessary operations (like user deletion when a user is assigned to an incident) and fulfills stakeholder needs

## Detailed User Stories

USN-I: Prevent User Deletion if Assigned to an Incident

As an admin, I want to create a user from the system Task

Description:

An administrator attempts to delete a user account. The system checks if the user is currently assigned to any active property, tenant record, or lease agreement. If the user is linked to any active record, the system prevents deletion and displays an error message: "User cannot be deleted as they are assigned to active records. "

Acceptance Criteria:

- The system validates user assignment status before deletion
- If assigned, deletion is blocked and an appropriate message is shown
- If not assigned, user is successfully deleted

Priority: High

Sprint: Sprint-I

## USN-2: Lease Contract Creation with Validation

As a property manager, I want to create lease contracts by assigning a tenant to a property Task

Description:

A property manager initiates a new lease agreement. The system validates that the property does not already have an active lease assigned. If validation fails, the system blocks lease creation and notifies the manager.

Acceptance Criteria:

- Only one active lease per property is allowed
- Validation occurs before lease is saved
- Error message displayed if property already has active lease

Priority: High

Sprint: Sprint-I

## USN-3: Tenant Alert Notifications

As a tenant, I want to receive an alert when my lease is close to expiry or when payment is due

Task Description:

The system automatically sends email notifications to tenants 30 days before lease expiration and on the 1 st of every month for payment reminders.

Acceptance Criteria:

- Email sent automatically based on scheduled triggers
- Email contains lease/payment details
- Tenant receives notification at registered email address

Priority: Medium

Sprint: Sprint-2

## Benefits of DFD and User Stories

Data Flow Diagrams:

- Provide a clear visual representation of system processes

- Identify data inputs, outputs, and storage requirements
- Help stakeholders understand system workflows User Stories:
- Focus on user needs and business value
- Written in plain language for easy understanding
- Serve as basis for development and testing

#### References

- <https://www.visual-paradigm.com/tutorials/data-flow-diagram-example-video-rentalsystem.isp>
- <https://www.atlassian.com/agile/project-management/user-stories>
- [https://www.cs.uct.ac.za/mit notes/software/pdfs/ChpQ6.pdf](https://www.cs.uct.ac.za/mit%20notes/software/pdfs/ChpQ6.pdf)
- <https://depexotechnologies.com/blog/lease-management-software-development-a-complete-guide/>

## Project Planning Phase

(Product Backlog, Sprint Planning, Stories, Story points)

<b>DATE</b>	11 November 2025
<b>TEAM ID</b>	NM2025TMID04855
<b>PROJECT NAME</b>	Lease Management System
<b>MAXIMUM MARKS</b>	4 Marks

### 1. Project Overview

The Lease Management System streamlines real estate operations by automating property, tenant, lease, and payment workflows for property managers. The system offers features such as property/tenant management, lease contract creation and approval, payment recording, validation, and notifications. Agile methodology with sprints is used for iterative delivery and clear tracking of project goals.

### 2. Product Backlog, Sprint Schedule, and Estimation

### 3. User Stories (Examples)

- USN-1: As an admin, I can add, edit, or delete property and tenant records so I can maintain accurate data.
- USN-2: As a tenant, I can view my payment history and see upcoming payment due dates.
- USN-3: As a manager, I can initiate, approve, and terminate lease agreements to streamline the contract process.
- USN-4: As a user, I am prevented from assigning more than one active lease per property, ensuring business rules compliance.
- USN-5: As a manager, I receive timely notifications about due payments and expiring leases for better decisionmaking.
- USN-6: As a developer, I document all modules and workflows for future maintenance and handover.

- Average velocity: (16 completed SP) / (9 days) = 1.78 points/day

#### Burndown Chart:

A burndown chart displays story points remaining over project days, providing a visual indicator of progress and helping teams stay on schedule.

### 5. Major Features & Modules

- Property Management: Add, edit, view properties, addresses, types.
- Tenant Management: Manage tenants, statuses, contact info.
- Lease Contracts: Initiate, approve, terminate leases; validation rules.
- Payment Life Cycle: Record, track payments; send payment reminders & confirmations.
- Notifications: Email alerts for approvals, rejections, payment due/successful, and expiring contracts.
- Security & Validation: Custom validation, permissions, approval process logic.
- Reporting: Dashboards for property status, payment history, vacancy rate, revenue per property.
- Documentation: User and technical documentation for all functional modules.

### 6. Technology Stack

- Platform: Salesforce (Lightning, Apex, Flows, Approval Process, Email Templates)
- Database: Salesforce Custom/Standard Objects
- Frontend: Salesforce Lightning UI
- Automation: Flows, Apex Scheduler, Validation Rules
- Reporting: Salesforce Reports and Dashboards

### 7. Stakeholders

- Admin/Management: Oversees data and approvals
- Coordinator/Clerk: Manages tenant and contract data
- Tenants: End-users accessing payment/status info

### 8. References

- <https://www.smartsheet.com/content/sprint-planning-templates>
- <https://www.atlassian.com/agile/project-management/templates>
- <https://tangoanalytics.com/blog/lease-management/>
- <https://1000projects.org/lease-management-system-salesforce-project.html>



# Project Design Phase-II

## Data Flow Diagram & User Stories

<b>DATE</b>	11 November 2025
<b>TEAM ID</b>	NM2025TMID04855
<b>PROJECT NAME</b>	Lease Management System
<b>MAXIMUM MARKS</b>	4 Marks

### Data Flow Diagrams

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It can be manual, automated or a combination of both. It shows how information enters and leaves the system, what changes the information, and where data is stored.

For the project "Lease Management System", Data Flow Diagrams (DFDs) illustrate how user deletion requests are processed within the system. The DFD shows the interaction between the admin, the system, and the incident database to validate whether a user is assigned to any active matters. If a user is linked to an incident, the system blocks deletion and sends a notification explaining why the user cannot be deleted.

### Example Data Flow Diagram

#### Context Diagram (Level 0 DFD)

The context diagram shows the Lease Management System as a single process with external entities:

External Entities:

- Admin/Manager
- Tenant
- Property Owner
- Payment Gateway Data

Flows:

- Property & Tenant Information —+ System
- Lease Contracts & Approvals System • Payment Records System
- Notifications & Reports 4— System

#### Level 1 DFD - Lease Management System

Main Processes:

1. Manage Properties: Admin creates and updates property records (property details, address, type, sqft)
2. Manage Tenants: Admin adds, edits tenant information (name, email, phone, status)
3. Create Lease Contracts: Manager initiates lease agreements linking tenants to properties

4. Process Payments: System records payment data and validates payment status
  5. Send Notifications: Automated email alerts for approvals, payments, and lease expiry
  6. Generate Reports: Dashboard and analytics for occupancy, revenue, and payment tracking
- Data Stores:
- Property Database
  - Tenant Database
  - Lease Database
  - Payment Database
- Data Flow Example:
- Admin —+ Property Info —+ Manage Properties Property Database
  - Tenant —+ Payment Info Process Payments Payment Database —+ Email Notification —+ Tenant
  - Manager Lease Request Create Lease Lease Database Approval Notification Manager

## Level 2 DFD - Payment Processing Module Sub-processes:

1. Validate Payment Data: Check tenant ID, property ID, amount, and payment date
2. Record Payment: Store payment record in Payment Database
3. Update Payment Status: Mark as 'Paid' or 'Not Paid'
4. Trigger Notification: Send confirmation email to tenant if payment successful
5. Generate Payment Report: Create payment history for tenant

Data Flows:

- Tenant submits payment —+ Validate Payment Data Check Tenant & Property Database
- Valid payment —+ Record Payment —+ Payment Database
- Payment successful —+ Update Status Trigger Email Notification Tenant receives confirmation

## User Stories

User stories define what different users need from the system in simple, goal-focused language. In this project, they help ensure the system blocks unnecessary operations (like user deletion when a user is assigned to an incident) and fulfills stakeholder needs effectively.

## Detailed User Stories

### USN-I: Prevent User Deletion if Assigned to an Incident

As an admin, I want to create a user from the system Task

Description:

An administrator attempts to delete a user account. The system checks if the user is currently assigned to any active property, tenant record, or lease agreement. If the user is linked to any active record, the system prevents deletion and displays an error message: "User cannot be deleted as they are assigned to active records. "

Acceptance Criteria:

- The system validates user assignment status before deletion
- If assigned, deletion is blocked and an appropriate message is shown
- If not assigned, user is successfully deleted

Priority: High

Sprint: Sprint-I

### USN-2: Lease Contract Creation with Validation

As a property manager, I want to create lease contracts by assigning a tenant to a property Task

Description:

A property manager initiates a new lease agreement. The system validates that the property does not already have an active lease assigned. If validation fails, the system blocks lease creation and notifies the manager.

Acceptance Criteria:

- Only one active lease per property is allowed
- Validation occurs before lease is saved
- Error message displayed if property already has active lease

Priority: High

Sprint: Sprint-I

### USN-3: Tenant Alert Notifications

As a tenant, I want to receive an alert when my lease is close to expiry or when payment is due

Task Description:

The system automatically sends email notifications to tenants 30 days before lease expiration and on the 1st of every month for payment reminders.

Acceptance Criteria:

- Email sent automatically based on scheduled triggers
- Email contains lease/payment details
- Tenant receives notification at registered email address

Priority: Medium

Sprint: Sprint-2

### Benefits of DFD and User Stories

Data Flow Diagrams:

- Provide a clear visual representation of system processes
  - Identify data inputs, outputs, and storage requirements
  - Help stakeholders understand system workflows
- User Stories:
- Focus on user needs and business value
  - Written in plain language for easy understanding
  - Serve as basis for development and testing

## References

- <https://www.visual-paradigm.com/tutorials/data-flow-diagram-example-video-rental-system.jsp>
- <https://www.atlassian.com/agile/project-management/user-stories>
- [https://www.cs.uct.ac.za/mit notes/software/pdfs/Chp06.pdf](https://www.cs.uct.ac.za/mit%20notes/software/pdfs/Chp06.pdf)