|  |  |
| --- | --- |
| **Date** | 3 August 2025 |
| **Team ID** | LTVIP2025TMID30830 |
| **Project name** | Lease management |
| **Maximum Marks** |  |

**INTRODUCTION:**

The development of a Lease Management System requires a well-defined set of functional and non-functional requirements to ensure that the final solution effectively addresses the current challenges faced in property and lease administration.

**📌 Project Name:**

**Lease Management System**

**🧩 Team ID:**

**LTVIP2025TMID30830**

**📅 Date:**

**3 August 2025**

**🔍 1. Solution Requirements**

**✅ Functional Requirements**

1. **Lease Tracking**
   * Record start and end dates
   * Auto-reminders before lease expiration
2. **Tenant Management**
   * Store tenant details, contact info, lease status
   * Prevent assigning multiple tenants to the same property
3. **Property Management**
   * Store property metadata (name, type, address, area)
4. **Payment Processing**
   * Record rent amounts, payment dates, and status
   * Mark as Paid/Not Paid
5. **Automated Communication**
   * Monthly rent reminders
   * Email alerts for approval, rejection, and payment confirmation
6. **Approval Workflows**
   * Approve or reject tenant requests for lease creation or ending
7. **Validation Rules**
   * Lease end date must be after start date
   * Mandatory fields check
8. **Role-Based Access**
   * Admin, Manager, and (future) Tenant user roles

**🧪 Non-Functional Requirements**

* Response Time: Page load <1 sec
* Trigger execution: ~0.5 sec
* Scalability: Support more than 1000 tenants
* Security: Role-based access and data sharing rules
* Availability: 99.9% uptime through Salesforce

**🏗️ 2. Proposed Solution Architecture**

| **Layer** | **Component** | **Purpose** |
| --- | --- | --- |
| Data Layer | Salesforce Custom Objects | Store and link Property, Lease, Tenant, Payment |
| Logic Layer | Apex Triggers & Classes | Validation, email logic, relationship enforcement |
| Workflow Layer | Approval Process & Flows | Automate actions based on conditions |
| Scheduler Layer | Apex Scheduler | Send periodic email reminders |
| UI Layer | Lightning App | User interface for Admins/Managers |
| Notification Layer | Email Templates | Notify users of approvals, payments, alerts |

**🧰 3. Salesforce Implementation Plan**

**🔹 Custom Objects**

* **Property**
* **Tenant**
* **Lease**
* **Payment for Tenant**

**🔹 Fields & Relationships**

* Text, Picklist, Email, Phone, Long Text
* Master-Detail and Lookup relationships between entities

**🔹 Tabs & Lightning App**

* Custom Tabs for each object
* Lease Management App with personalized branding

**🔹 Automation Components**

* **Apex Trigger:** Prevent duplicate property assignments
* **Validation Rules:** Ensure data correctness
* **Approval Process:** Automate lease workflows
* **Scheduled Apex Job:** Email reminder on 1st of every month

**✉️ 4. Email Templates**

| **Template Name** | **Use Case** |
| --- | --- |
| Tenant Leaving | Request leave approval |
| Leave Approved | Notify tenant of approval |
| Leave Rejected | Deny tenant request with reason |
| Monthly Reminder | Rent payment reminder |
| Payment Confirmation | Confirm payment received |

**📈 5. Expected Outcomes**

**✅ Short-Term Benefits**

* Centralized data access
* Reduced manual work
* Auto-alerts and reminders
* Faster lease processing

**📊 Long-Term Impact**

* Dashboards for insights
* Future integration with payment gateways
* Mobile app readiness
* AI-powered rent forecasting

**🏁 6. Conclusion**

This Salesforce-based solution will replace manual processes with an automated, accurate, and scalable lease management system. It enhances operational efficiency, ensures compliance, improves tenant satisfaction, and sets the foundation for future growth.

Bottom of Form