PROJECT DOCUMENT

* **Project Title**: LEASE MANAGAMENT
* College Name: Annamacharya Institute of Technology and Science

**TEAM**

* + **Team ID:** LTVIP2025TMID29022
  + **Team Size:** 4
  + **Team Leader:** K. Uday Kumar
  + **Team member:** J. Uday kiran
  + **Team member:** G. Vishnu Vardhan Reddy
  + **Team member:** Thappani Vinay Kumar

**MAIL IDs:**

Udaykumar215@gmail.com

Junnupaluudaykiran2002@gmail.Com

Velidhandla@gmail.Com

Vk641471@gmail.Com

**Project Overview:**

* Objective : Lease Managament

Lease management is a critical function in real estate, facilities, and asset management. Whether for commercial, industrial, or residential properties, effectively managing lease agreements plays a key role in maintaining financial stability, ensuring legal compliance, and supporting operational efficiency. With the advancement of digital tools like Salesforce, SAP, and other lease management platforms, organizations are now focusing on automating and optimizing the entire lease lifecycle. The following are the detailed objectives of a lease management system or project.

**IDEATION PHASE**

The Ideation Phase serves as the foundation of any successful project. It blends creativity, user empathy, and structured thinking to identify the core challenges, generate meaningful ideas, and prioritize solutions that bring tangible value to all users involved. This dynamic phase combines imagination with insight to ensure every feature addresses real-world needs.

As our project titled: **“LEASE MANAGAMENT”** The success of any software or system development project often depends on how well it begins. In the case of a **Lease Management project**, the **ideation phase** plays a vital role in setting a clear direction, aligning stakeholders, and shaping the foundation for effective system development. This phase is where problems are deeply explored, creative solutions are brainstormed, and strategic decisions are made about what features or processes the project should prioritize.

1. **Brainstorming & Idea Prioritization Template**

**Step 1: Team Gathering, Collaboration, and Selecting the**

**Problem Statement**

Developing a **Lease Management System** is a multidisciplinary process that requires effective teamwork, open collaboration, and a clear understanding of the problem to be solved. At the early stage of such a project, **gathering the right team**, encouraging **collaborative efforts**, and **selecting a well-defined problem statement** are foundational steps that significantly influence the project's success. This phase ensures that everyone involved is aligned with the project’s goals and understands the purpose and direction of the development process.

* **Lease Agreement Handling**
* Rent and Payment Tracking
* Lease Lifecycle Management
* Compliance and Legal Requirements
* Reporting and Analytics

After several discussions and stakeholder reviews, we clearly articulated the core issue:

**Problem Statement:**

**"The current lease management process is manual, error-prone, and lacks centralization, leading to missed deadlines, financial inaccuracies, and compliance risks. A modern, automated system is needed to streamline lease tracking, improve efficiency, and support business growth."**

**Step 2: Brainstorm, Idea Listing, and Grouping**

As part of our ideation phase for the Lease Management System, we conducted a collaborative brainstorming session using a digital whiteboard. Each team member contributed raw ideas based on market research, user requirements, and current inefficiencies in traditional lease handling systems.

We generated around 25–30 initial ideas. To bring structure and clarity, we grouped similar ideas into major functional themes relevant to lease management:

**Booking & Scheduling**: · Centralized calendar to manage lease start and end dates across multiple properties, with real-time conflict detection for overlapping leases and automated alerts for upcoming renewals or terminations.

* **Customer Relationship Management**: Detailed profiles for tenants and landlords, including lease history, contact details, preferences, payment records, and notes for personalized communication and improved engagement.
* **Communication & Notifications**: Automated communication via email, SMS, and WhatsApp for sending rent reminders, renewal alerts, maintenance schedules, and lease status confirmations.
* **Analytics & Reporting**:Interactive dashboards showing rental income trends, occupancy/vacancy rates, upcoming lease expirations, and financial performance metrics across properties.
* **Access Control & Security**: Role-based access system for property managers, tenants, landlords, accountants, and maintenance teams, ensuring secure and appropriate access to lease data and operations.

From a pool of 25–30 ideas, we clustered and shortlisted those that directly improved efficiency, reduced manual work, and elevated the customer experience. These formed the foundation for our product roadmap.

**Step 3: Idea Prioritization**

Each clustered idea was carefully evaluated against three key criteria:

After brainstorming and grouping the ideas into key functional areas, we proceeded with a structured **idea prioritization process** to determine which features should be developed first. Our goal was to identify ideas that deliver the highest impact with the least complexity and resource investment.

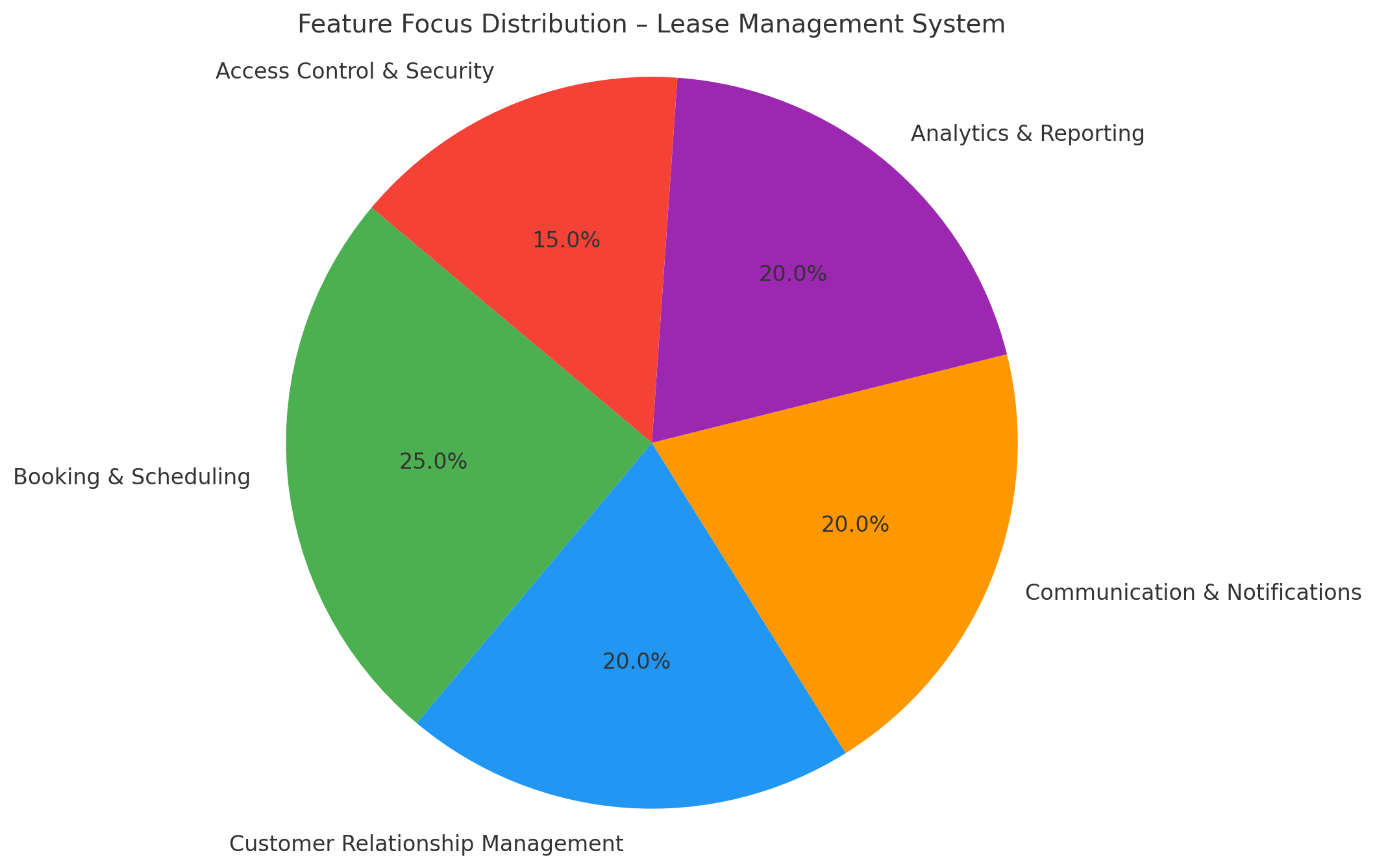
We used a **prioritization matrix** based on the following criteria:

* **Impact on Efficiency**: How significantly the idea reduces manual work or streamlines lease operations
* **User Value**: The degree to which it enhances tenant or landlord experience.
* **Feasibility**: Technical complexity, resource availability, and timeline for implementation
* **Alignment with Business Goals**: Contribution to revenue optimization, compliance, and property performance.

Each idea was scored collaboratively by the team and plotted on an **Impact vs. Effort** chart.

Top-Priority Features Selected:

* **Centralized Lease Calendar** (Booking & Scheduling):  
  High impact, low complexity—critical for tracking lease terms and avoiding overlaps.
* **Automated Notifications** (Communication & Notifications):  
  Easy to implement and essential for improving tenant engagement and reducing delays.
* **Tenant Profiles with Payment History** (Customer Relationship Management):  
  Adds value to both operations and user experience, with manageable technical effort.



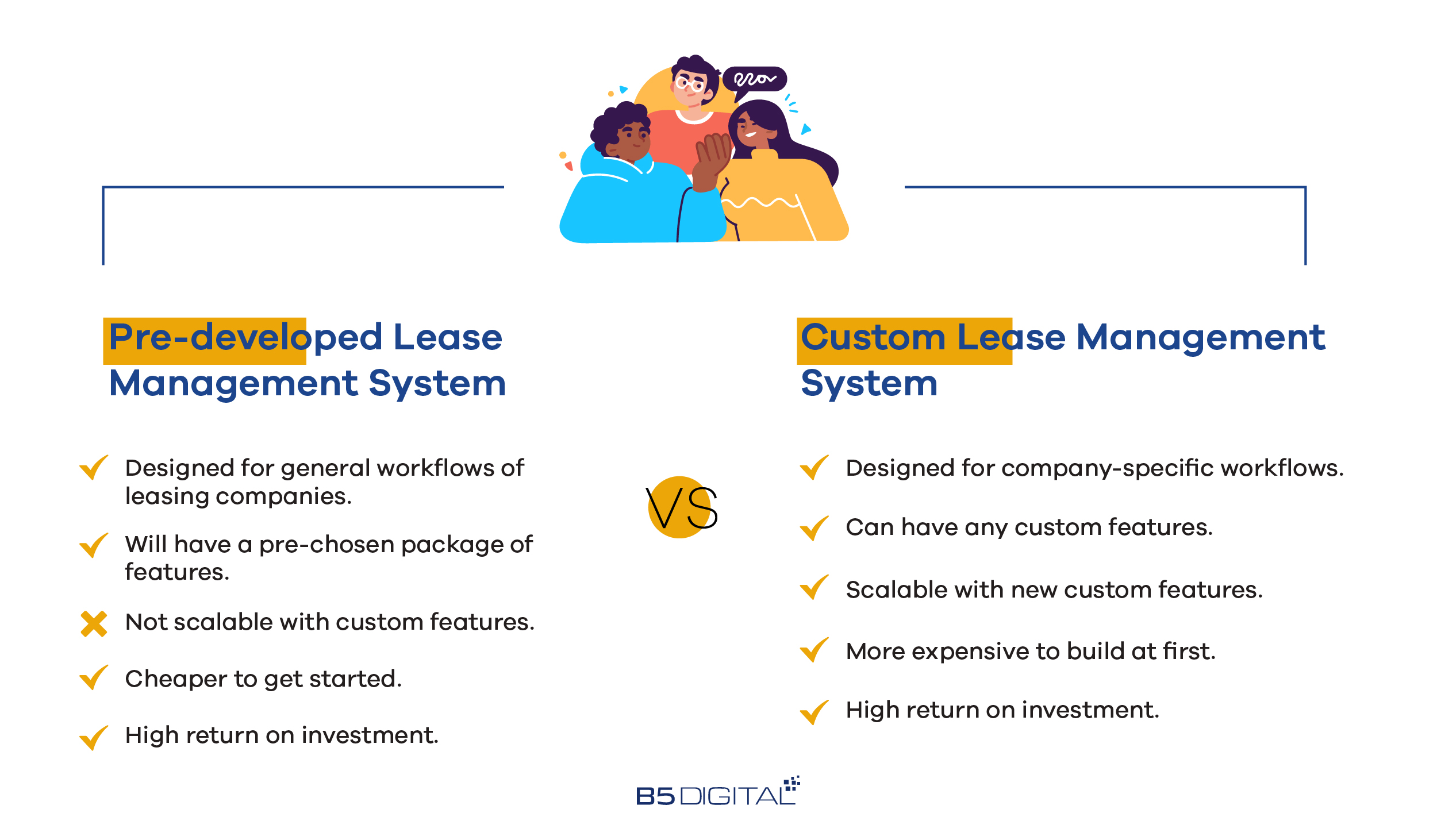
**2. Empathy Mapping- Empathize & Discover**

Empathy Map Covers ;

## Lease management (also known as lease administration) is the practice of organizing renting agreements—whether for property, equipment, or other assets—to ensure compliance, avoid missed deadlines, maximize value, and support strategic decision‑making. Here's a comprehensive overview:

| **Says** | **Thinks** |
| --- | --- |
| “I need renewal reminders before lease end.” | “Did I miss a clause change or rent escalation?” |
| “Getting compliance right is critical.” | “Will this tool reduce my workload and errors?” |
| “I struggle with scattered documents.” | “I wish everything was in one searchable place.” |

| **Does** | **Feels** |
| --- | --- |
| Uploads lease docs, sets reminders | Overwhelmed during lease cycle peaks |
| Cross-checks dates, sends notices | Anxious about compliance errors or missed payments |
| Coordinates with legal/accounting teams | Relieved when tasks flow smoothly |
| Runs status reports | Confident with proactive insights |

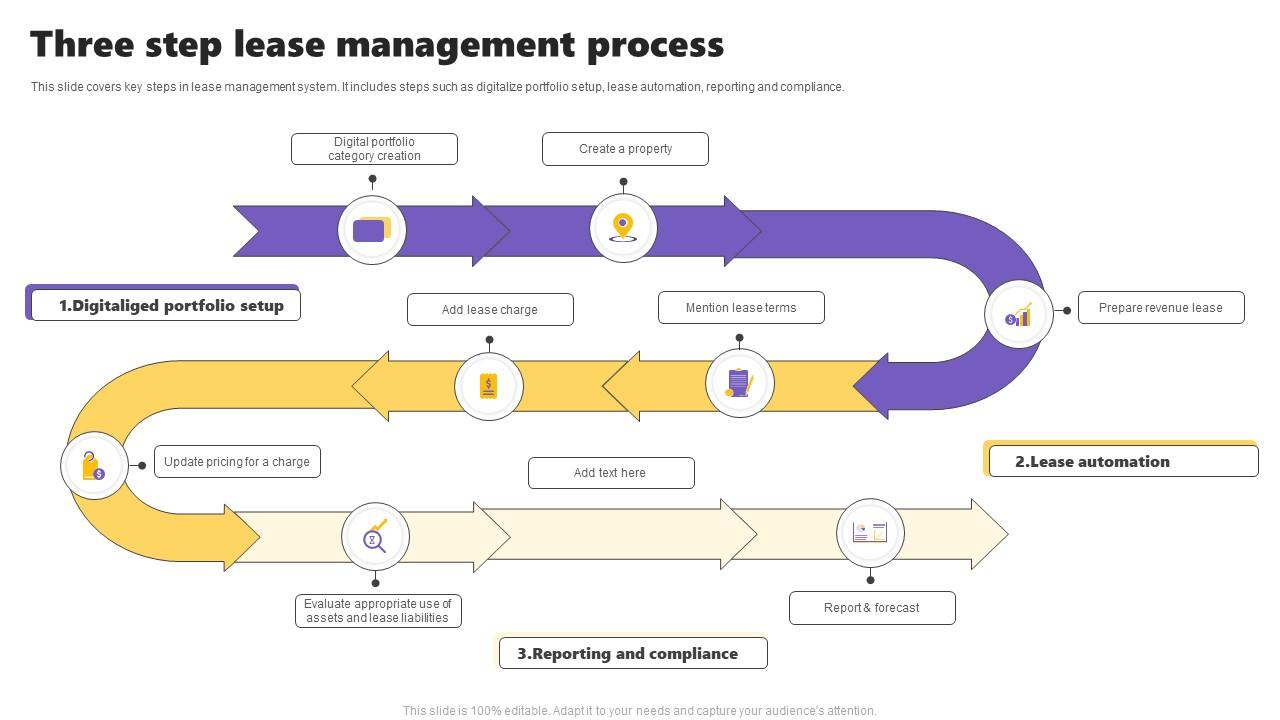


By stepping into the user’s shoes, we ensured that our Salesforce CRM features (formulas, flows, triggers, dashboards) directly addressed their key frustrations.

**3.Define the Problem Statements**

**Customer Problem Statement Template**

Lease administrators and property managers currently handle lease agreements, tenant communications, and payment tracking through disconnected and often manual methods. This leads to lease renewals being missed, delayed payment collections, inconsistent tenant records, and poor visibility into lease portfolio performance. A centralized Salesforce CRM system for lease management can consolidate lease data, automate reminders and payment tracking, streamline tenant communication, and offer real-time reporting through dashboards, workflows, and automation tools. This statement captures the core challenges faced by property managers and tenants and guided the design of our object models, validation rules, process automations, and reporting structures.



**REQUIREMENT ANALYSIS**

The Requirement Analysis phase aims to gather, analyze, and document the business needs and system requirements for a Lease Management System. This phase ensures that all stakeholders’ expectations are understood clearly and translated into precise functional and non-functional requirements to guide system design and development.

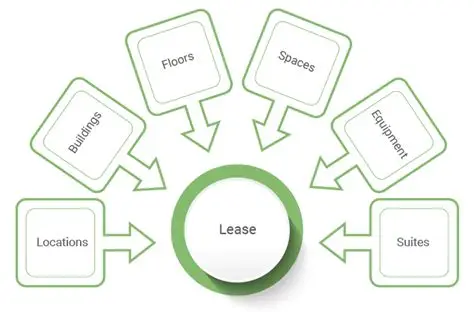
In our project, **“ Lease Management’’** refers to the administration and oversight of lease agreements between landlords (property owners) and tenants (individuals or businesses). It involves managing lease lifecycle activities such as drafting agreements, rent collection, lease renewals, compliance monitoring, and tenant relations.

1. **Customer Journey Map-Understanding User Experience Flow**

**Puropse:**

1. To **understand the user’s experience** step-by-step as they interact with a product, service, or system.
2. To **identify user needs, emotions, and pain points** at each stage of their journey.
3. To **highlight gaps and friction** that cause dissatisfaction or delays.
4. To **improve the overall user experience** by designing solutions that address real problems.
5. To **align teams and stakeholders** around a shared view of the user’s perspective.
6. To **drive customer-centric improvements** in processes, products, or services.
7. To **support strategic decision-making** based on actual user behaviors and feedback.

**Journey steps:**



**2.Data Flow Diagram:**

**Purpose:** Mapping Information Flow Between Objects

The **Data Flow Diagram (DFD)** models how information moves between Salesforce objects and components in the banquet hall CRM system. It helped us structure relationships between:

* **Lease Application Submission**

Tenant submits a lease application with personal and financial details.

Application data is stored in the Lease Applications Database.

* **Verification and Background Check**

The system verifies application documents and performs credit/background checks.

Results are used to approve or reject the application.

* **Approval Notification**

Tenant is notified of approval or rejection.

Legal and property management teams are informed if approved.

* **Lease Agreement Preparation**

Lease agreement is drafted based on approval.

Sent to the legal department for review and compliance checks

* **Agreement Execution**

Final lease agreement is sent to tenant for signature.

Signed agreement is received and stored securely.

* **Payment Management**

Rent invices are generated as per lease terms.

Tenant makes payments which are recorded and verified.

Payment confirmation and receipts are sent to the tenant.

* **Maintenance Request Handling**

Tenant submits maintenance requests.

Requests are validated and assigned to appropriate vendors.

Vendors complete work and submit service reports.

Tenant and property manager are updated on maintenance status.

* **Reporting and Analytics**

System generates reports on lease status, payments, and maintenance.

Reports help property managers track performance and make decisions.

Real-time dashboards provide quick insights.

* **Data Storage**

Lease Applications Database stores tenant applications.

Lease Agreements Database holds signed contracts.

Payment Records Database logs all financial transactions.

Maintenance Requests Database tracks repair jobs and vendor reports.

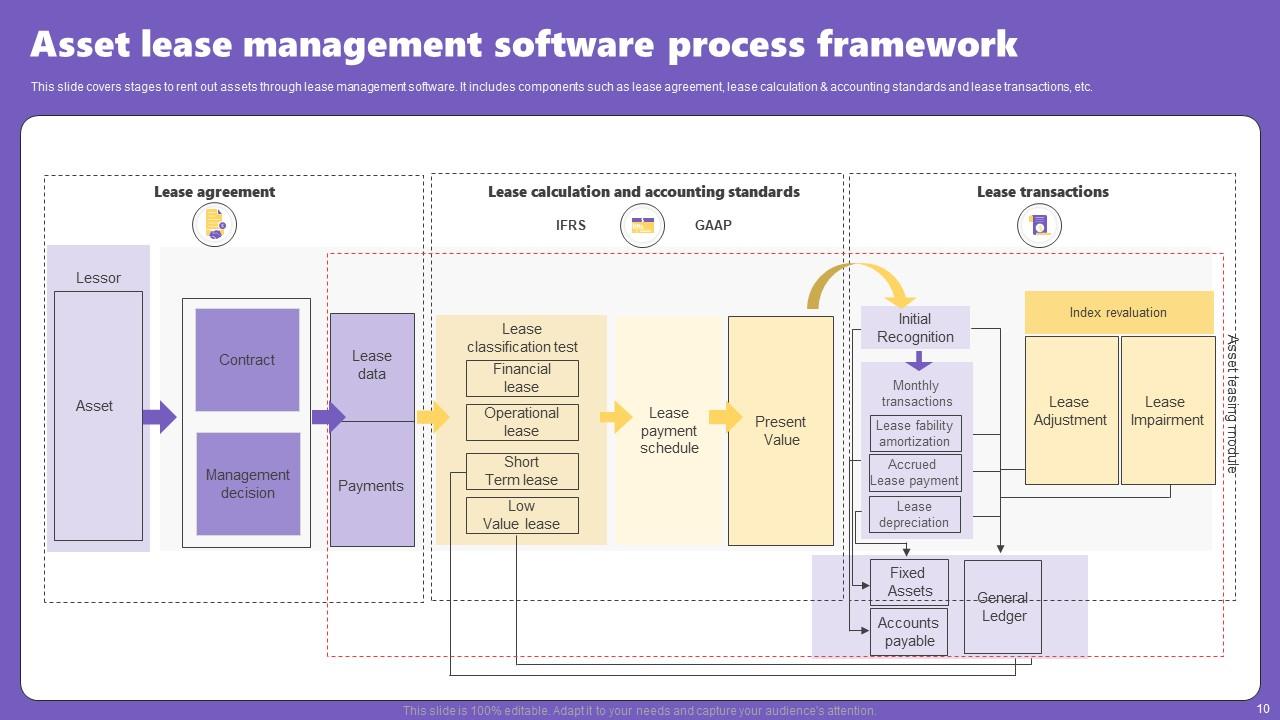
Vendor Information Database contains approved vendor details.

Reports Database archives generated reports.

* **Overall System Benefits**

Automates lease administration process

Ensures data accuracy and transparency.



**3. Solution Requirements:**

* . Functional Requirements

**Lease Application:** Tenants can apply online with document uploads.

**Approval Workflow:** Auto-verification, background checks, approval/rejection, notifications.

**Lease Agreements:** Auto-generated agreements, e-signatures, secure storage.

**Payments:** Rent invoice generation, online payments, receipts, late payment handling.

**Maintenance Requests:** Tenants raise issues, system assigns vendors, tracks status

**Reports & Dashboards:** Real-time lease, payment, and maintenance analytics.

**User Roles:** Role-based access for tenants, owners, vendors, legal, and finance teams.

### ⚙️ Non-Functional Requirements

**Performance:** Fast, responsive, handles multiple users.

**Security:** Encrypted data, secure logins, access control.

**Scalability:** Supports more properties and users as system grows.

**Usability:** Simple UI, mobile-friendly.

**Reliability:** 99.9% uptime, regular backups.

**Maintainability:** Easy to update, well-documented, modular.

### 🔗 Integration Requirement

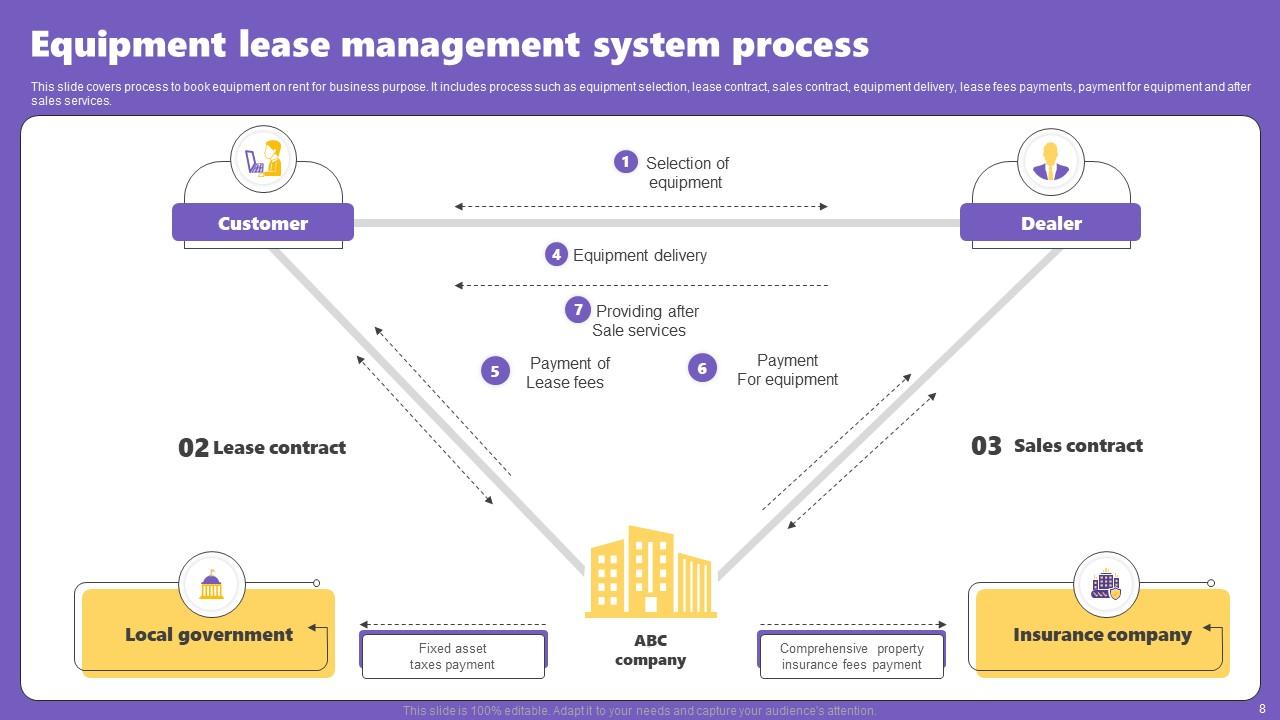
Payment gateway

Background check services

Email/SMS notification systems

Accounting tools (optional)

1. **Technology Stack:**



**Summary:**

The Lease Management Project aims to streamline and automate the processes involved in managing residential and commercial property leases. By identifying key user needs and operational challenges, the project focuses on improving efficiency, communication, and data accessibility for all stakeholders—tenants, landlords, and property managers.

Through a structured ideation process, we conducted brainstorming sessions, empathy mapping, and idea prioritization to define a solution that addresses the most critical pain points in the leasing workflow. Features like centralized booking and scheduling, tenant relationship management, automated reminders, insightful analytics, and secure access control form the foundation of our product roadmap.

**Project Design Phase**

The **Design Phase** of the Lease Management Project focuses on transforming the prioritized ideas and user insights into a clear, structured system design. This phase ensures that the solution is not only functional but also user-friendly, scalable, and aligned with technical and business requirements

.**1. Problem–Solution Fit:**

**Problem Recap**:

Traditional lease management processes are often manual, fragmented, and inefficient. Property managers, landlords, and tenants face challenges such as:

* Missed lease renewals and payment deadlines
* Lack of centralized records and communication
* Time-consuming administrative work
* Limited visibility into lease performance and financial trends
* Security concerns with sensitive documents and access control

**Does the Proposed Solution Fit?**

Yes — the **problems identified in the empathy and discovery phases are well-matched by the functional solutions** outlined in your design. This ensures your lease management system is **relevant, user-centric, and impactful**.

Let me know if you’d like this mapped visually in a table or diagram for your recap slide!

**2. Proposed Solution:**

Our proposed CRM application is designed to digitize and streamline banquet hall booking operations using Salesforce’s powerful customization capabilities.

**Key Functional Features:**

**● Custom Objects:**

Lease Agreement Management,Tenant and Landlord ManagementProperty,Asset Management,Booking and Scheduling,Reporting and Analytics.

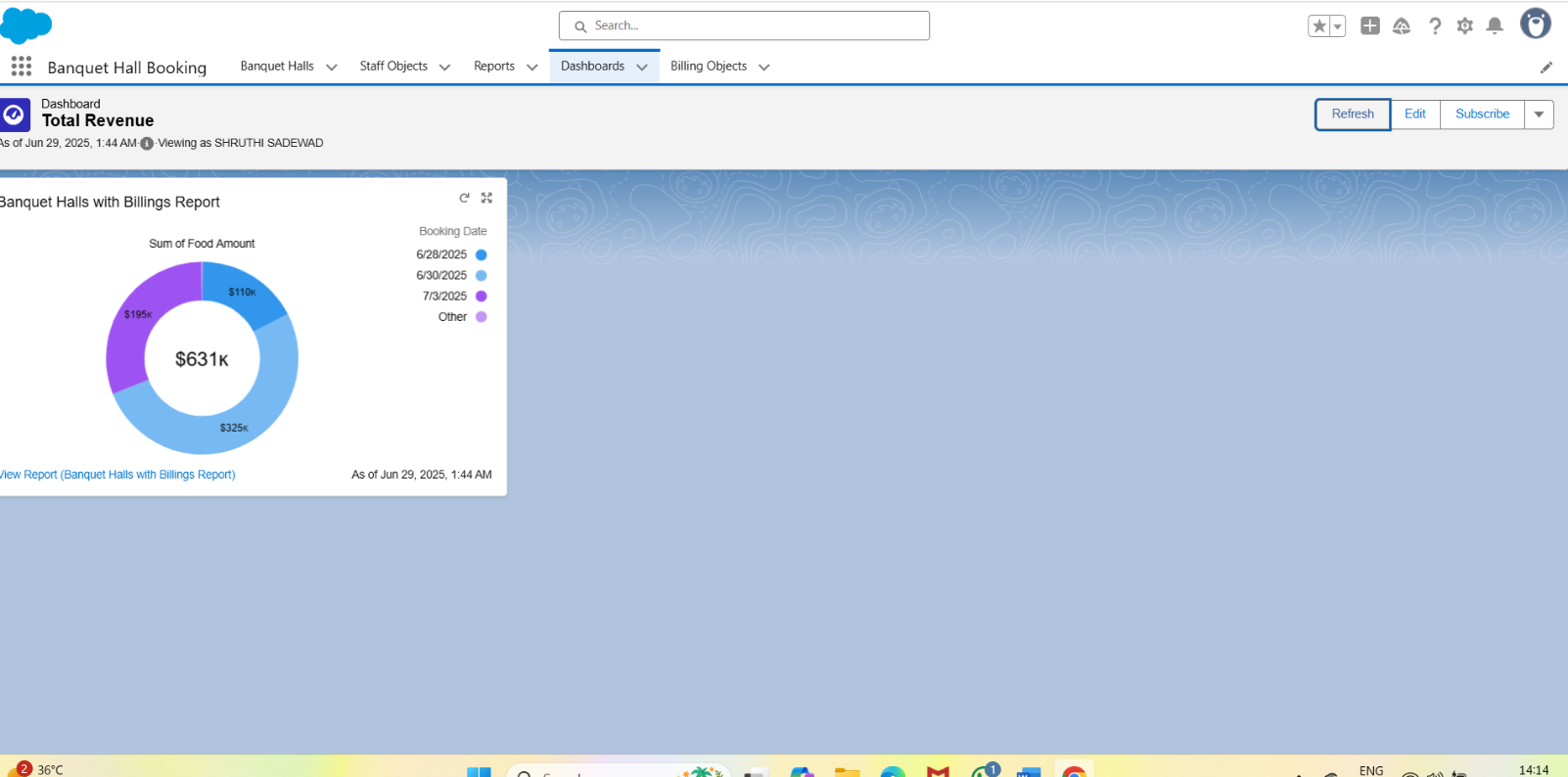
**● Automation & Validation:**

System automatically notifies tenants and landlords before lease expiry or renewal dates.

**● UI & Navigation:**

· Overview of key metrics: active leases, upcoming renewals, pending payments, maintenance requests.

· Quick action buttons for common tasks like creating a new lease or raising a maintenance ticket.

****

**Project Planning Phase**

The Project Planning Phase transformed major project milestones into manageable, time-bound sprints aligned with the lease managament development timeline. This approach ensured clear task ownership, streamlined collaboration, and consistent progress tracking. By breaking down features like lease agreement management,tenant and landlord management Property,Asset Management,Booking and Scheduling,Reporting and Analytics. the team was able to execute efficiently and stay aligned with project goals and stakeholder expectations.

**Project Planning Template**

Sprint Schedule – Based on Project Milestones

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sprint** | **Functional**  **Requirement**  **(Epic)** | **T ask (Mapped from**  **Milestone)** | **Priority** | **Team Members** |
| Sprint-1 | Developer  Setup & Basic  Objects | Creating Developer  Account & Activating  Org | High | Member 1 |
| Sprint-1 | Custom Object  Creation | Creating custom  Objects-A Banquet  booking application | High | Member 1,2 |
| Sprint-2 | UI Tabs & App  Creation | Creating Tabs &  Lightning App | High | Member 3 |
| Sprint-2 | Field  Configuration | Creating fields,  formula fields,  picklists, relationships | Medium | Member 1,3 |
| Sprint-3 | Layouts &  Validations | Page Layouts +  Validation Rules | High | Member 2,4 |
| Sprint-3 | Flows &  Triggers | Automations using  Flows and Apex  Triggers | High | Member 2,3 |
| Sprint-4 | Reports &  Dashboards | Generate Reports and  create Dashboards | High | Member 4 |
| Sprint-4 | Final  Integration &  Conclusion | Final Review, Testing,  and Functional  Summary | Medium | All Members |

Project Tracker & Sprint Timeline

Duration: Each sprint is 6 days, aligned with your June 2025 internship

Schedule.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sprint** | **Duration** | **Sprint start**  **date** | **Sprint end**  **date** | **Sprint release**  **date** |
| Sprint-1 | 6 Days | 03 Jun 2025 | 08 Jun 2025 | 08 Jun 2025 |
| Sprint-2 | 6 Days | 09 Jun 2025 | 14 Jun 2025 | 14 Jun 2025 |
| Sprint-3 | 6 Days | 15 Jun 2025 | 20 Jun 2025 | 20 Jun 2025 |
| Sprint-4 | 6 Days | 21 Jun 2025 | 26 Jun 2025 | 26 Jun 2025 |

**Summary:**

The Project Planning Phase allowed our team to convert 12 major

milestones into 4 streamlined sprints with assigned priorities and contributors.

By aligning sprints with real internship dates and breaking tasks down into

functional chunks, we ensured steady progress and simplified execution

**Project Executable Files**

This phase outlines the actual Salesforce configurations, data models, and outcomes implemented during the execution of the Banquet Hall Booking CRM application. It ensures that all key elements—custom objects, automation flows, validations, and reports—are clearly documented and aligned with real-world use cases. The components developed in this phase are designed to be traceable, reusable, and easily assessable for future improvements or audits. It captures the practical realization of booking flows, vendor assignment rules, and payment tracking modules, offering clarity, replication, and validation across the entire system lifecycle.

**6.1 Project Files**

**Project Executable Files**

The following project files were executed in the Salesforce Developer Org:

Milestone 1: Developer Account Setup

Milestone 2: Object

Milestone 3: Tab

Milestone 4: Lightning App Setup

Milestone 5: Field Creation

Milestone 6: Validation Rules

Milestone 7: Email templates

Milestone 8: Approval process

Milestone 9: Apex Trigger

Milestone 10: Flows

Milestone 11: Scchedule classess

**List of Milestone Tasks with Supporting Screenshots and**

**Descriptions**

**Milestone 1: Developer Account Setup**

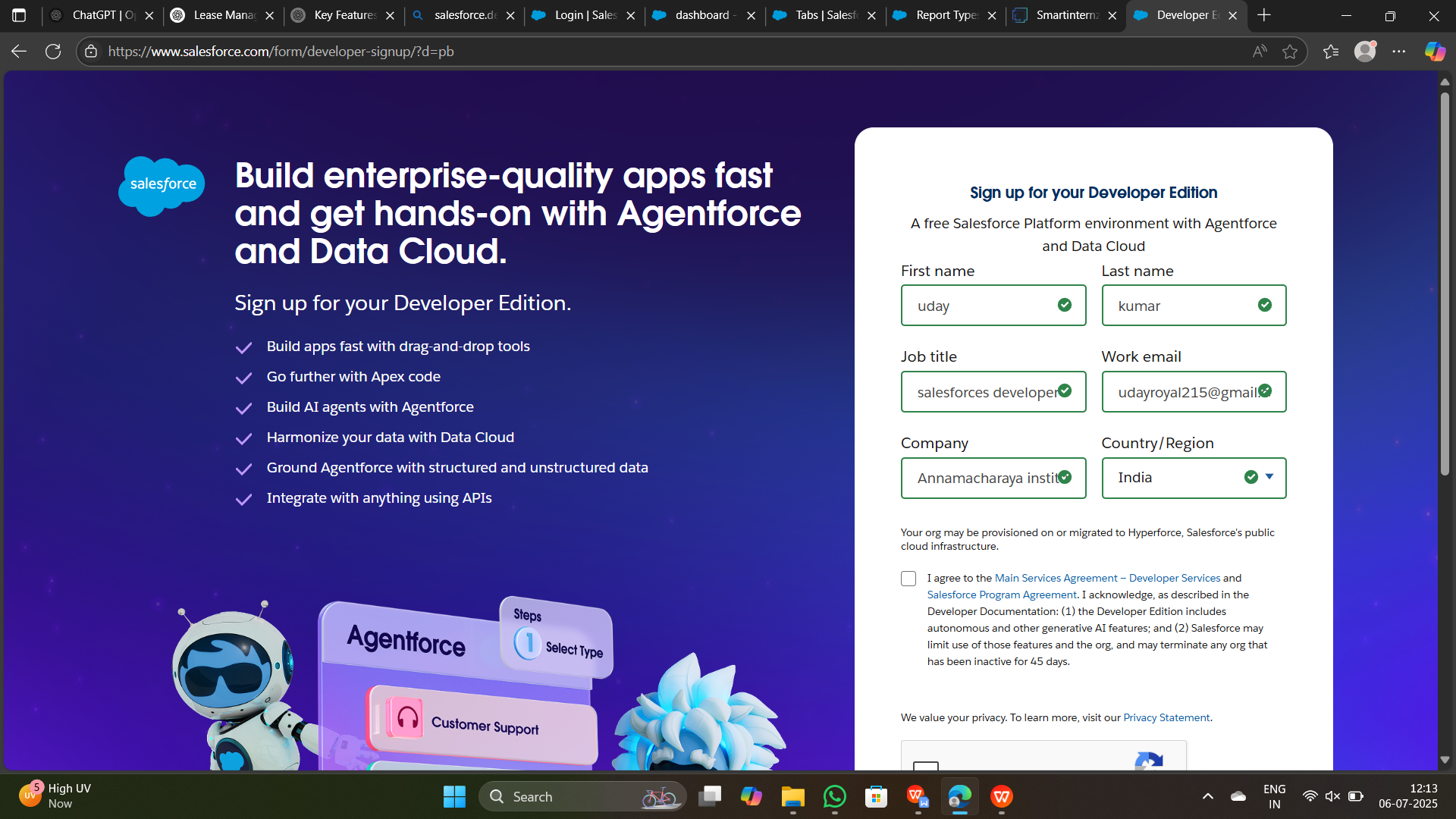
● Created and activated a Salesforce Developer Org.

● Link: <https://developer.salesforce.com/signup>

● Setup the base environment for Lease development.

● Verified access to Object Manager, Flow Builder, and App Builder

**OUTPUT SCREENSHOT**



**Milestone 2: Object Creation**

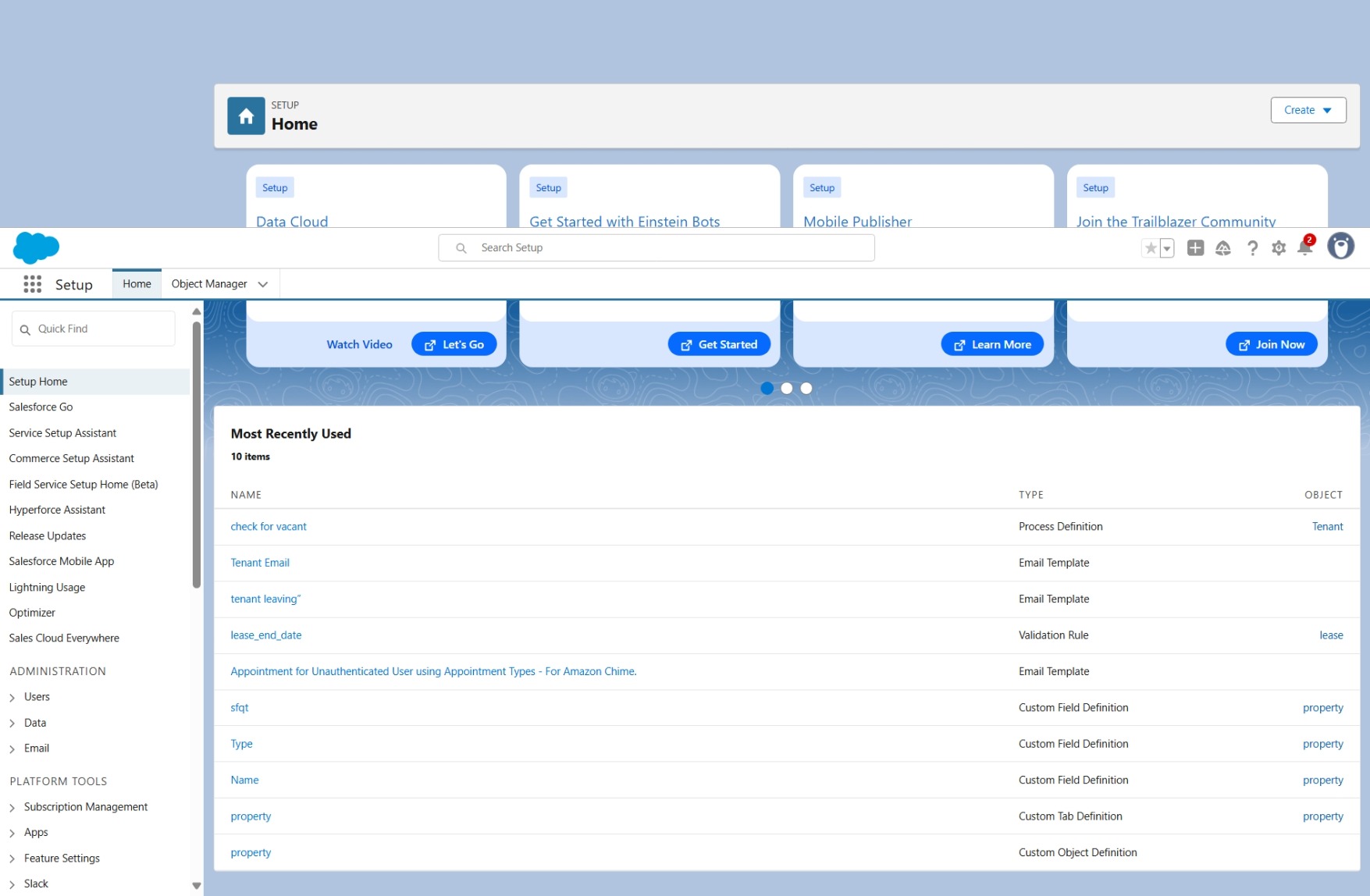
● Created 3 custom objects:

○ Lease Agreement Management,Tenant and Landlord ManagementProperty

● Established foundational schema for Lease Managament.

● Configured relationships using lookup fields.

**OUTPUT SCREENSHOT**



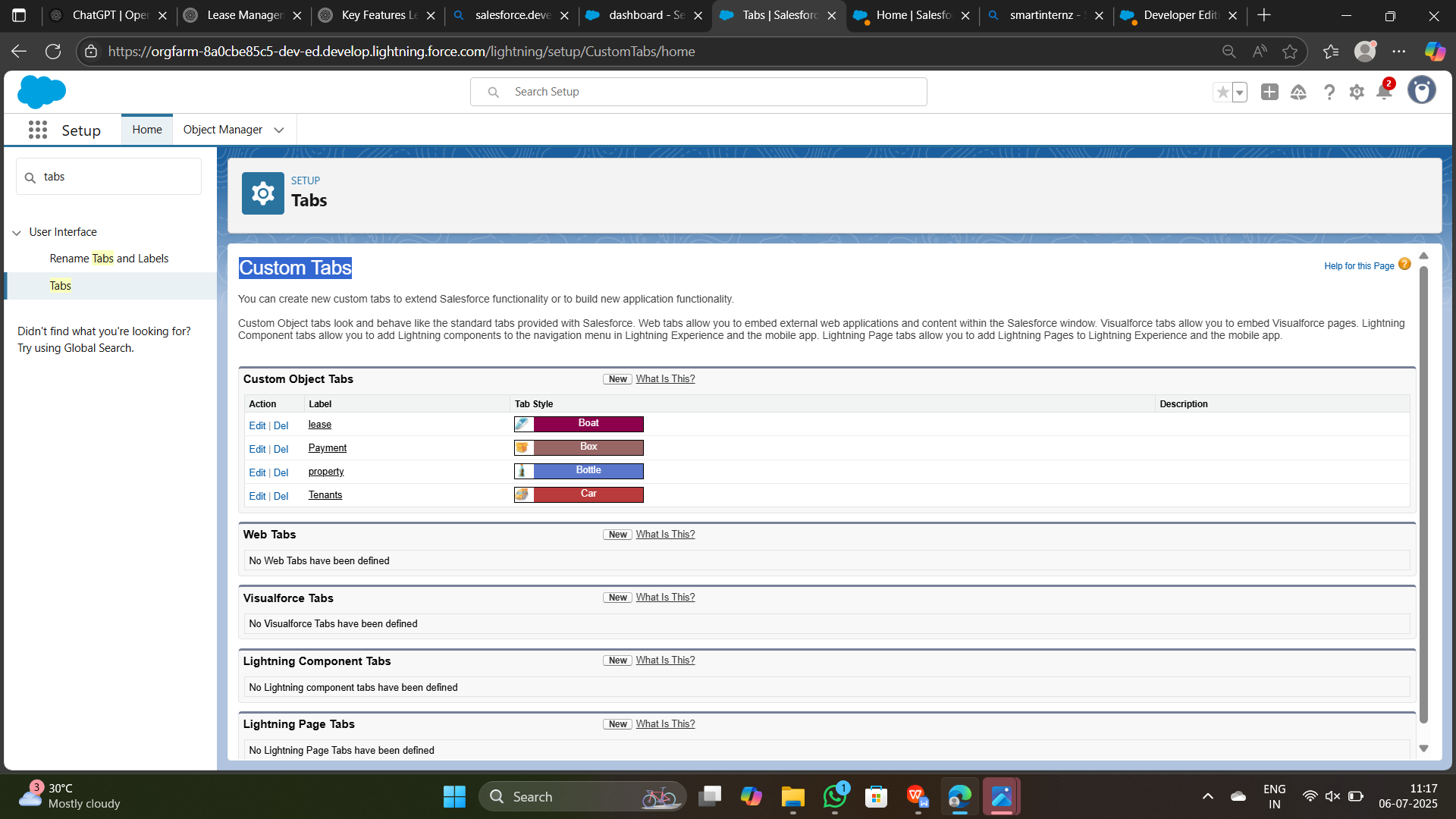
**Milestone 3: Tabs**

**●** Created tabs for each custom object.

● Enabled easy navigation and object access in the app.

● Ensured users can create/view records from the UI.

**OUTPUT SCREENSHOT**



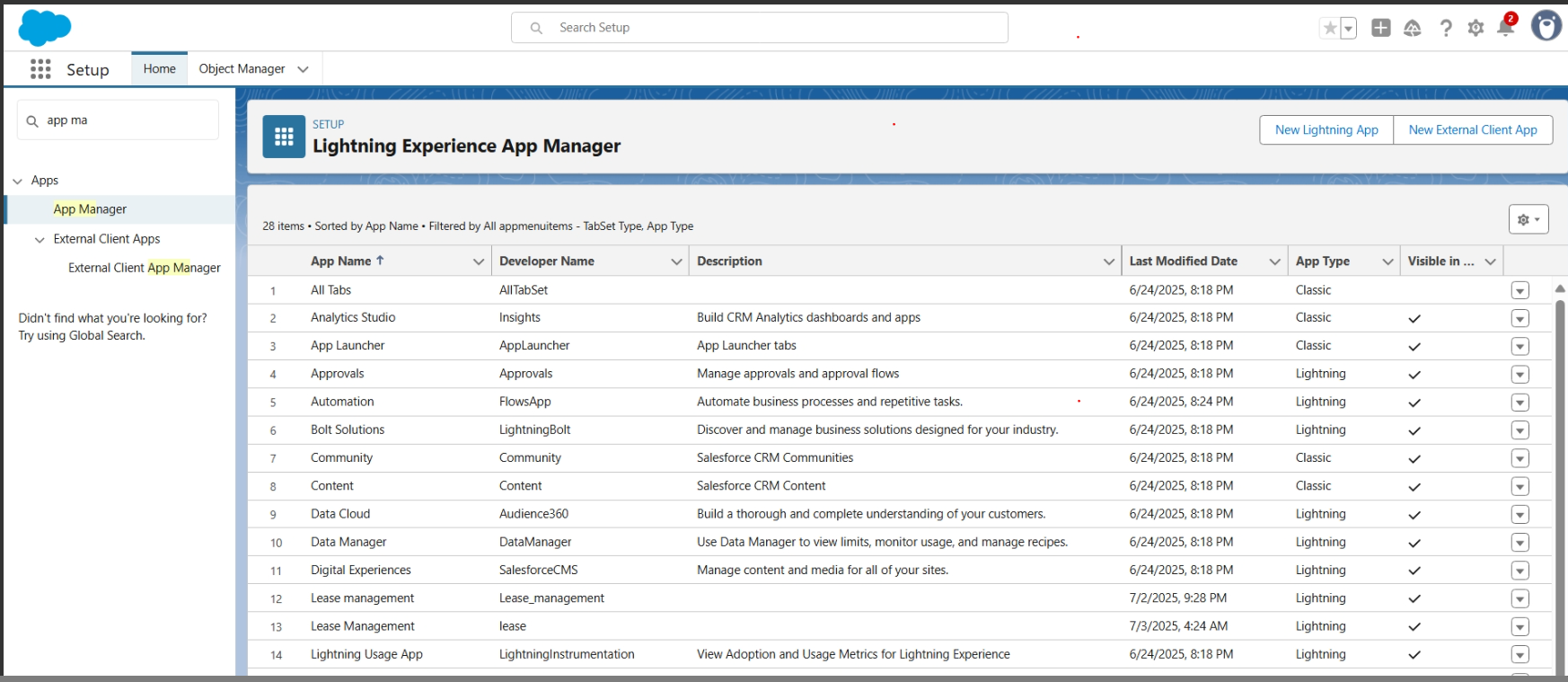
**Milestone 4: Lightning App Setup**

● Built a custom Lightning App named " Lease managament".

● Added relevant tabs to centralize operations.

● Simplified user workflow by grouping features.

**OUTPUT SCREENSHOT**



**Milestone 5: Field Creation**

● Added custom and formula fields like:

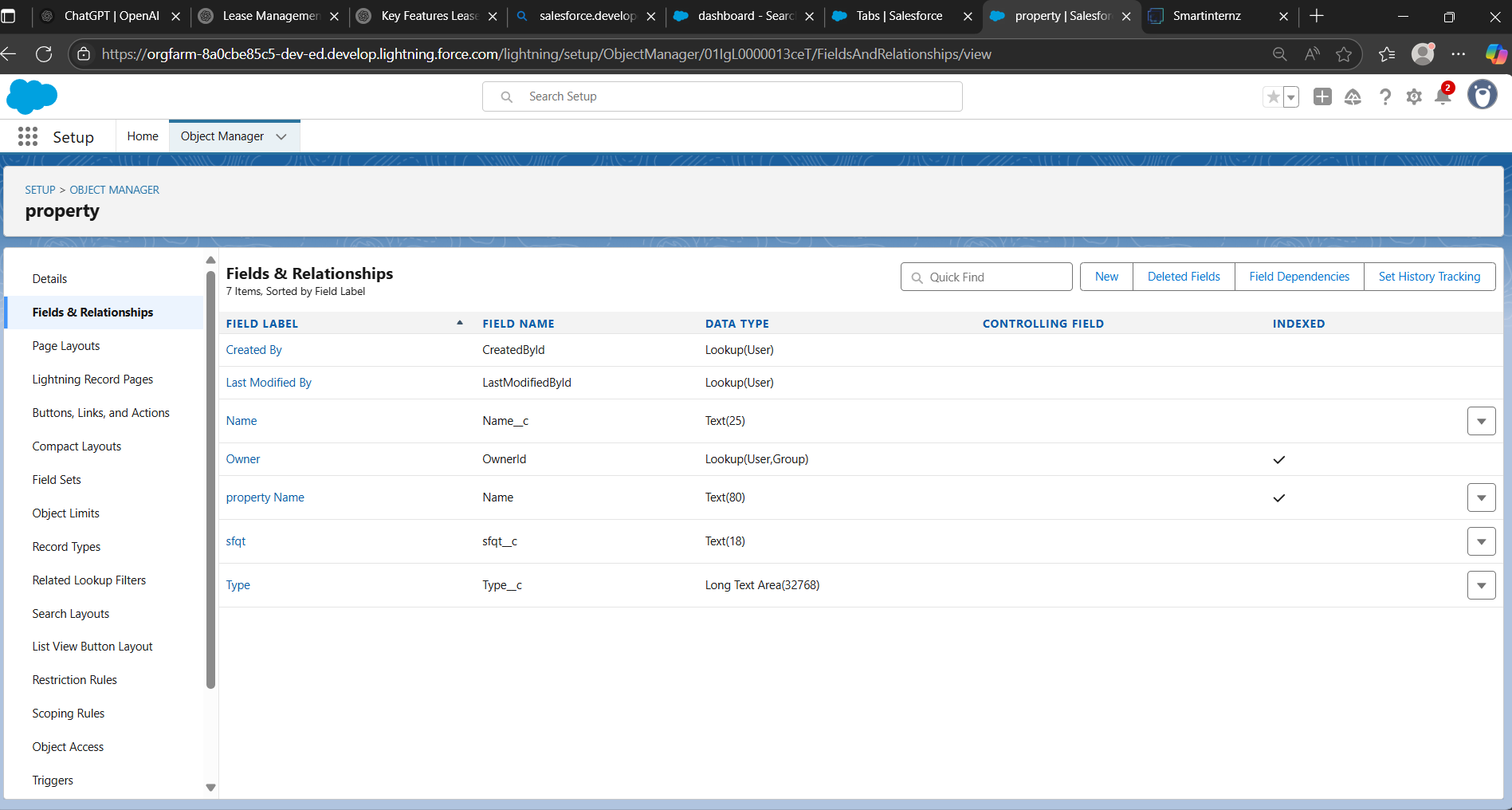
* Lease manageament , Property, Lease

Formula:

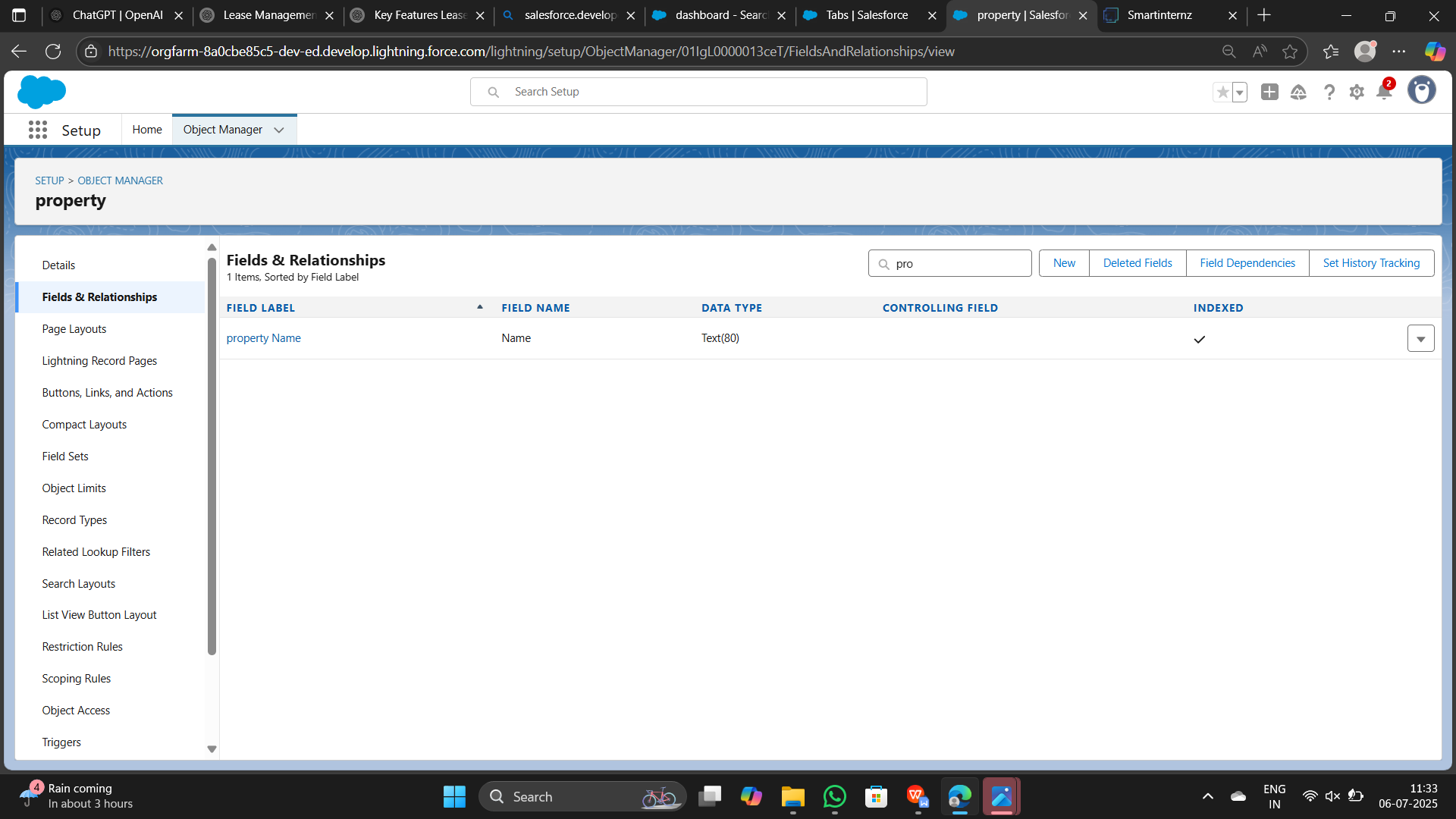
* IF( Customer\_Name\_\_r.Archestra\_\_c = TRUE, 50000, 0)
* IF( Customer\_Name\_\_r.DJ\_\_c = TRUE, 70000, 0)
* IF( Customer\_Name\_\_r.Magician\_\_c , 15000, 0)
* IF( Customer\_Name\_\_r.Shenai\_and\_Mridangam\_\_c = TRUE, 25000, 0)
* IF( Customer\_Name\_\_r.Kolatam\_\_c , 15000, 0)

**OUTPUT SCREENSHOT**

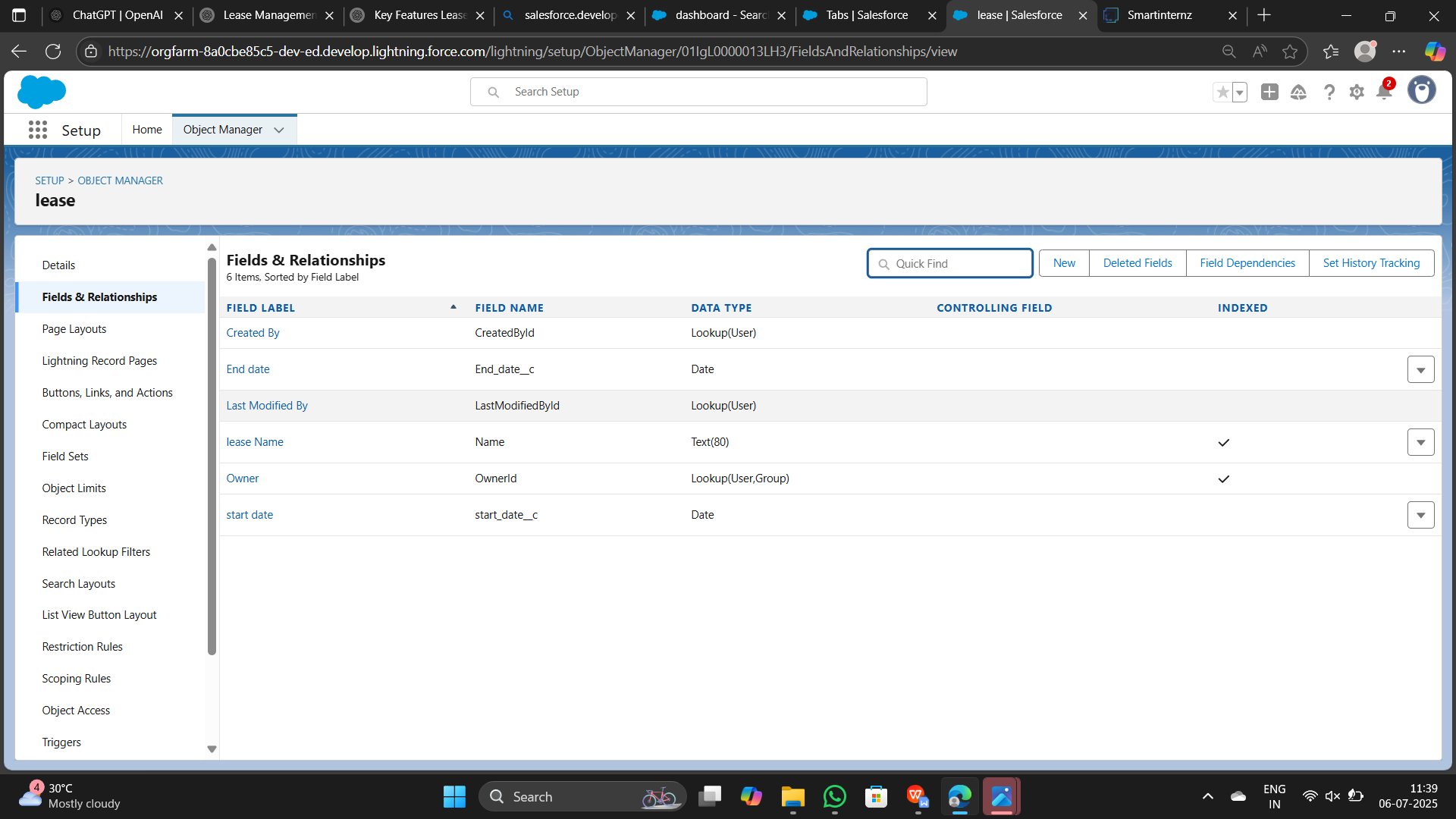
**Lease management Object Fields**



**Property Object Fields**



**Lease Object Fields**



**Milestone 6: Validation Rules**

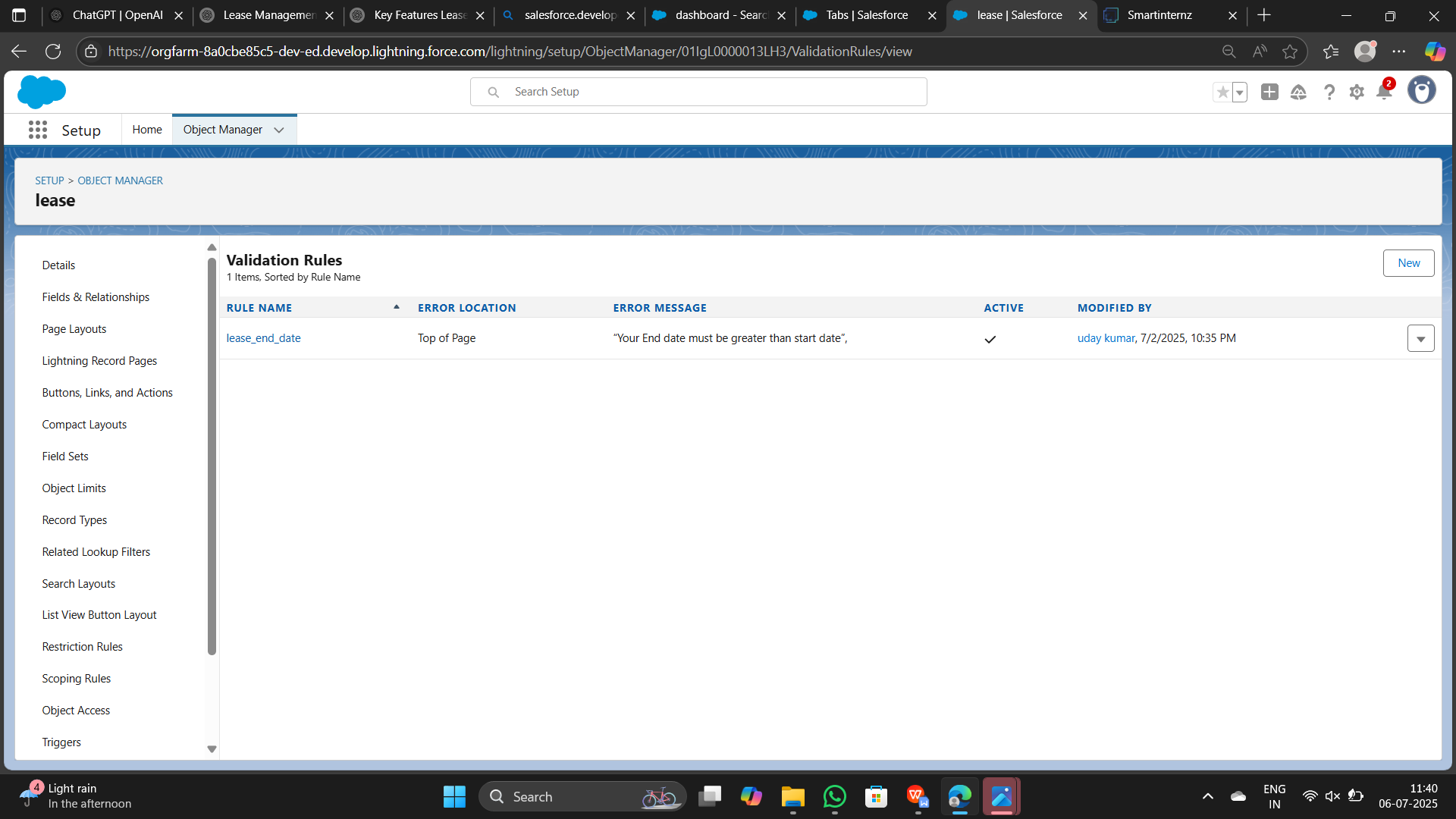
● Customized page layouts per object.

● Grouped fields logically for usability.

● Enhanced record readability and data entry experience.

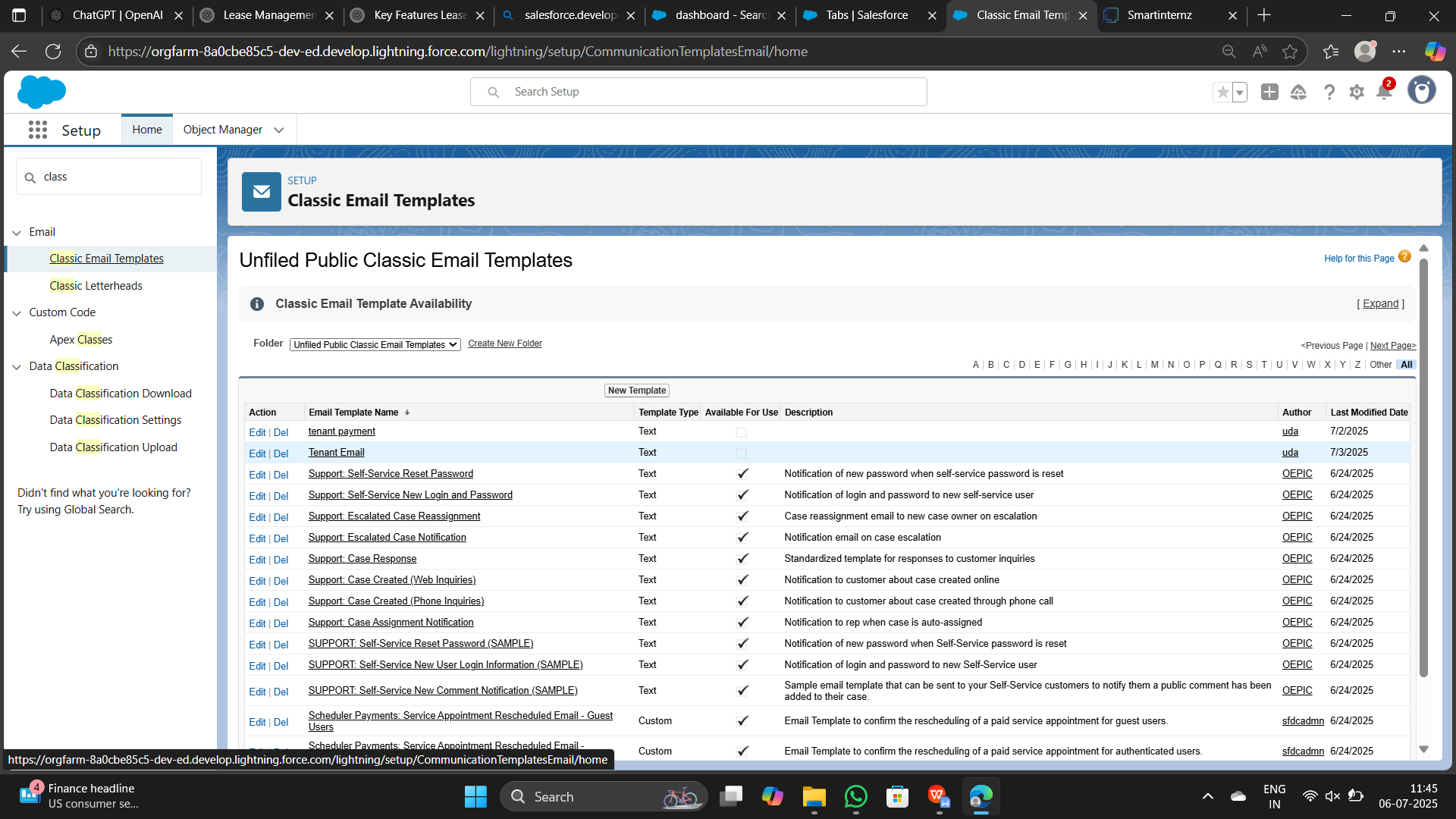
**OUTPUT SCREENSHOT**

**Lease Managament object Validation rule**

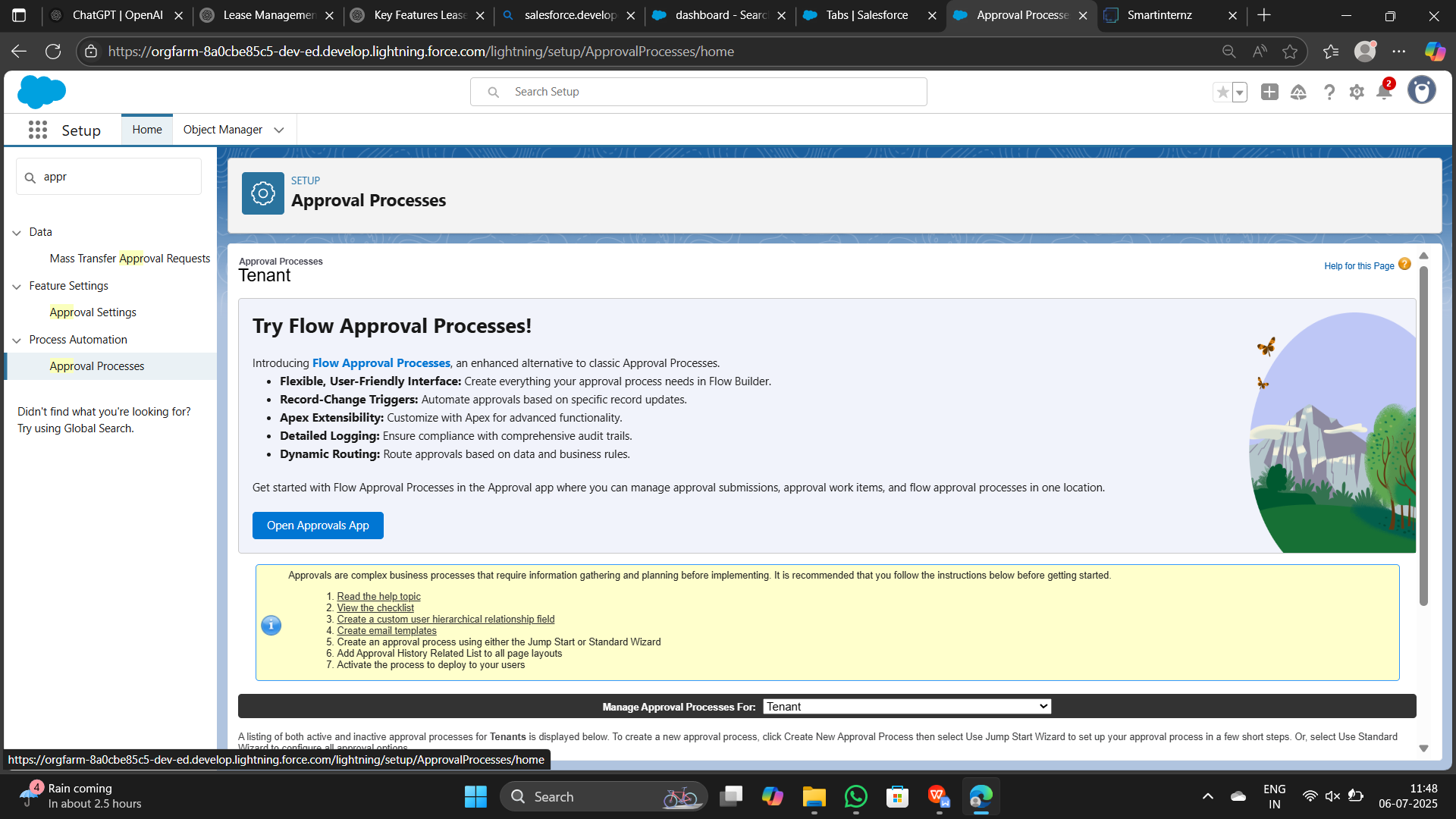


**Milestone 7: Email templates**

Type of record is created



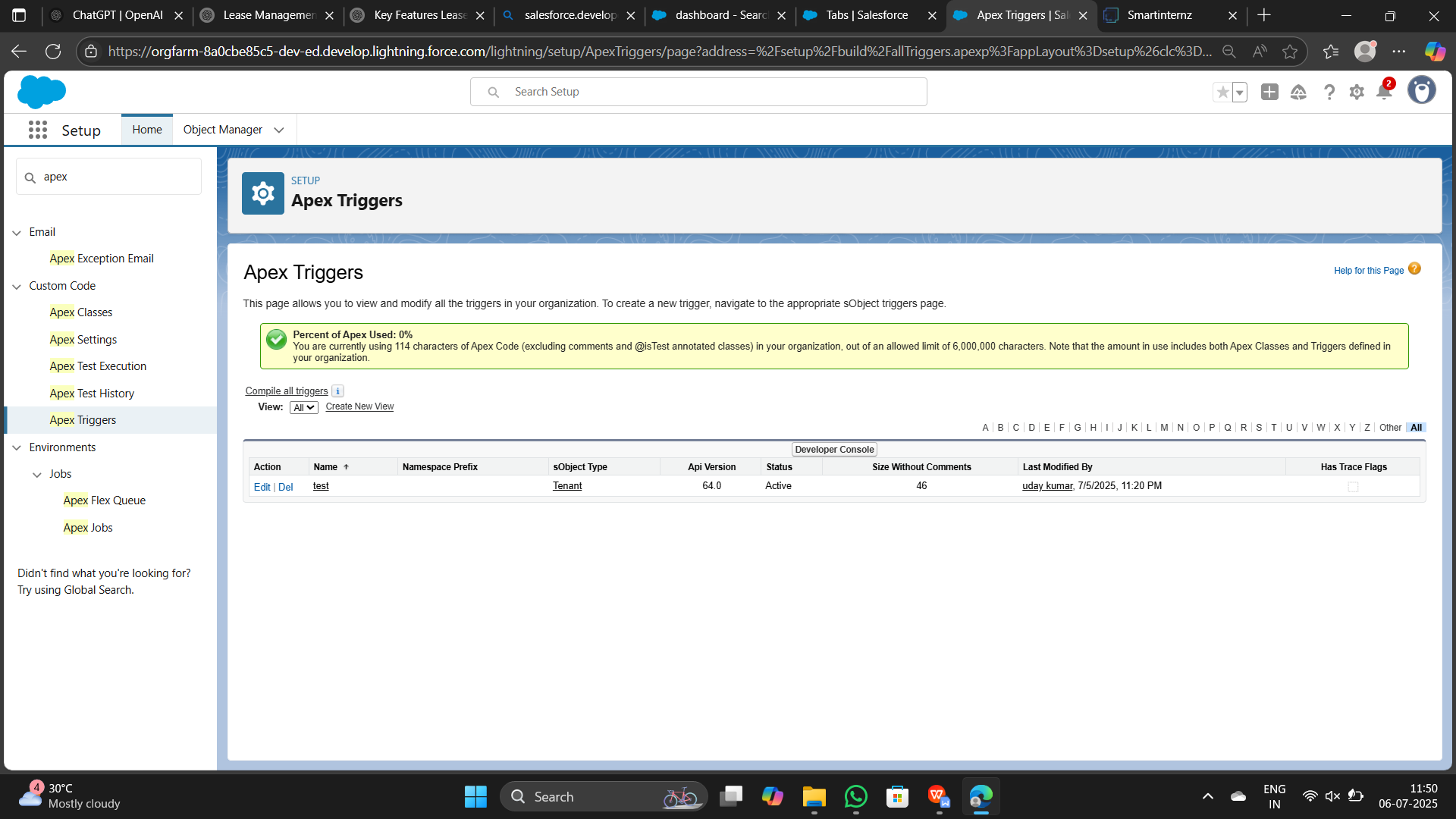
**Milestone 8: Approvel Process**

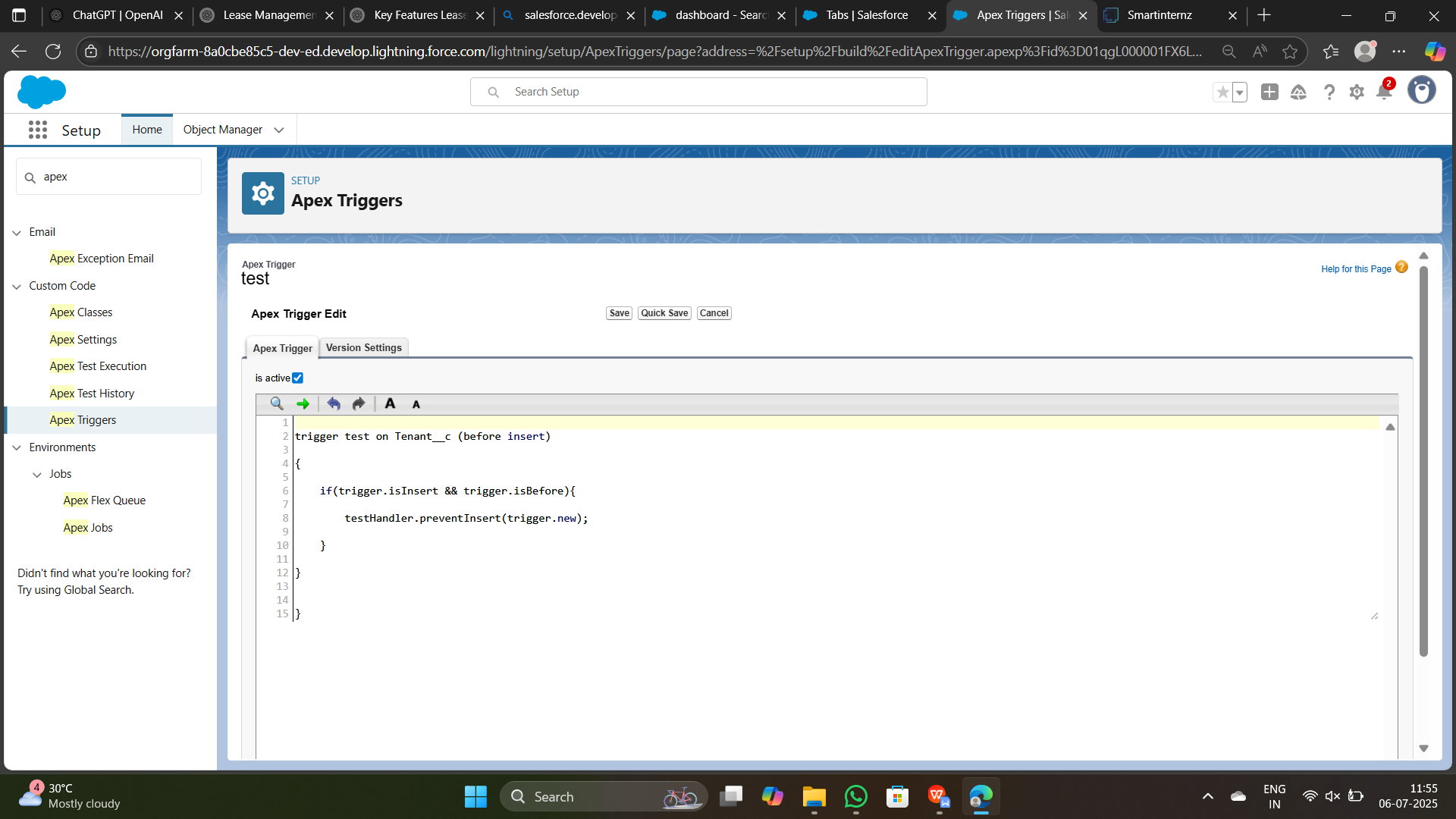


**Milestone 9: Apex trigger**

* Designed a record-triggered flow
* Improved automation and reduced manual input errors

**OUTPUT SCREENSHOT**



**Milestone 11: Reports**

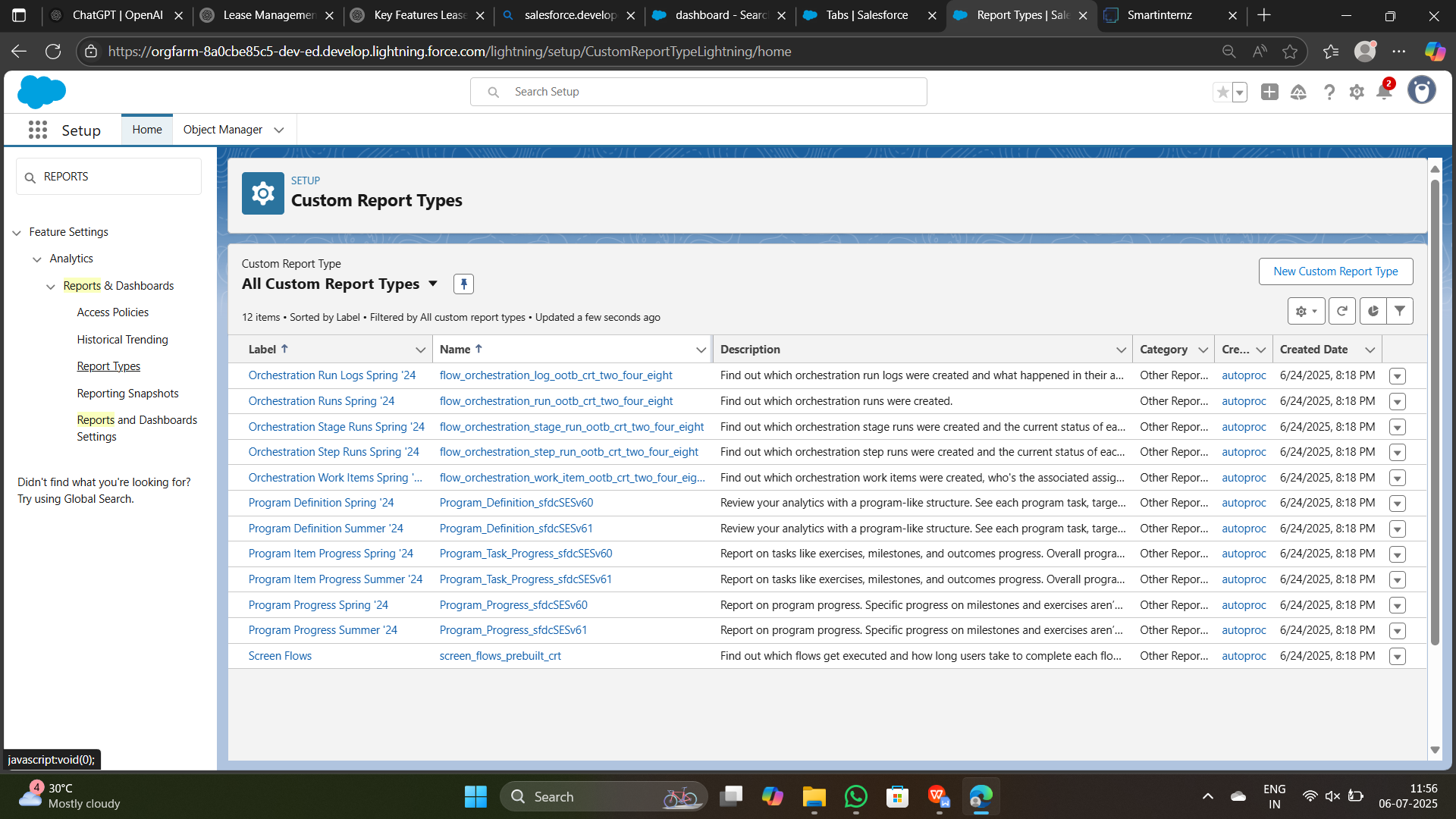
● Created reports for:

* Lease Agreement Management,Tenant and Landlord ManagementProperty.

● Used grouping, filters, and field summaries.

● Enabled data-driven decision-making.

**OUTPUT SCREENSHOT**



**Milestone 12: Final Review**

● Conducted full system testing.

● Checked all object links, flows, triggers, reports.

● Confirmed the Lease managamens fully functional and stable.

**Dataset**

The system was tested with the following types of data:(Sample Records and

Input Values Used During Testing)

### 1. ****Lease Application Module****

| **Field** | **Sample Input** |
| --- | --- |
| Tenant Name | Rakesh Kumar |
| Email | rakesh.kumar@gmail.com |
| Phone Number | 9876543210 |
| Document Uploaded | PAN Card (PDF), Aadhaar (JPEG) |
| Employment Status | Salaried |
| Monthly Income | ₹75,000 |
| Requested Lease Term | 12 months |

### 2. ****Lease Agreement Generation****

| **Field** | **Sample Input** |
| --- | --- |
| Lease ID | LSE1001 |
| Property Address | Flat 202, Lakeview Apartments, Hyderabad |
| Rent Amount | ₹18,000/month |
| Security Deposit | ₹36,000 |
| Start Date | 01-Aug-2025 |
| End Date | 31-Jul-2026 |
| Signed By | Tenant: Rakesh Kumar, Owner: A. Sharma |

### 3. ****Payment Module****

| **Field** | **Sample Input** |
| --- | --- |
| Payment ID | PAY98765 |
| Tenant Name | Rakesh Kumar |
| Amount Paid | ₹18,000 |
| Payment Date | 01-Aug-2025 |
| Payment Mode | UPI (Google Pay) |
| Transaction Status | Success |

### 4. ****Maintenance Request Module****

| **Field** | **Sample Input** |
| --- | --- |
| Request ID | MRQ120 |
| Tenant Name | Rakesh Kumar |
| Issue Type | Plumbing Leak |
| Description | Water leakage in kitchen sink |
| Date Reported | 05-Aug-2025 |
| Assigned Vendor | AquaFix Services |
| Status | Completed on 06-Aug-2025 |

### 5. ****User Role Testing****

| **Role** | **Sample Actions Tested** |
| --- | --- |
| Tenant | Submit application, make payment, raise maintenance request |
| Property Manager | Approve lease, assign vendor, view reports |
| Vendor | View assigned work, update service status |
| Legal Staff | Review agreements |
| Admin | Manage users, configure settings |

### Summary

The above records helped simulate **real-world data** during testing. This ensured:

All user inputs were accepted and processed correctly.

Validations worked (e.g., email formats, amount ranges).

Workflows like lease approval, payment confirmation, and maintenance handling functioned seamlessly.

Note: All test records were created using Salesforce's UI and validated via

flows and formula fields

## ****7. FUNCTIONAL AND PERFORMANCE TESTING – Lease Management System****

### ****7.1 Performance Testing****

The Lease Management System was tested using realistic lease scenarios and sample data representing tenants, properties, lease terms, payments, and maintenance requests. Key validations performed during the testing phase included.

**Trigger for role validation** (Tenant, Property Manager, Vendor) to ensure each user performed only permitted operations.

**Flow for auto-generating rent invoice** based on lease duration, monthly rent, and agreement start date.

**Field validation for input accuracy**, such as contact numbers, lease dates, payment entries, and required fields.

| **S.No** | **Parameter** | **Value / Observation** | **Screenshot Suggestion** |
| --- | --- | --- | --- |
| 1. | **Model Summary** | Salesforce CRM setup for the Lease Management System leverages custom objects (Lease, Tenant, Property, Payment, Maintenance), relationships, and automation flows. Test data import only succeeds if formats and relationships are correct. | CRM Object Schema & Relationships |
| 2. | **Field Validations** | Tested rules such as:  • Tenant must be 18+ years  • Phone number must be exactly 10 digits  • Required fields like Lease Start Date, Property ID, and Monthly Rent must not be blank.  System correctly blocks invalid entries. | Lease Form with Validation Errors |
| 3. | **Automation Accuracy (Flow + Trigger)** | Flow: Auto-calculate next due date & rent based on lease terms  Trigger: Ensures only “Property Manager” can approve lease or assign vendors  **Test Result:** All logic executed successfully during test scenarios | Flow Builder – Rent Generation Flow |
| 4. | **Reports Testing** | Lease Summary Report groups data by property and status.  Total revenue and active lease count calculated accurately using formula fields.  Export to Excel/PDF worked without errors. | Lease Summary Report – Grouped by Property |
| 5. | **Dashboard Verification** | Dashboards display total revenue, active leases, upcoming rent due, and open maintenance requests.  Filters and live data refresh confirmed working correctly. | Dashboard – Revenue & Lease Overview |
| 6. | **Data Accuracy (Manual + Automated)** | Manual and automated tests were conducted for multiple lease types and scenarios (monthly/quarterly rent, different start dates, vendor combinations).  Output values matched expectations consistently. | Lease Records with Expected Payment Output |

### ****Summary****

All components of the Lease Management System were thoroughly tested for:

✅ **Validation rule enforcement** (Age, Contact, Required Fields)

✅ **Automation flow and trigger logic accuracy** (Rent Calculation, Role Restriction)

✅ **Dashboard and report correctness** (Real-time Lease Insights)

✅ **Data relationships integrity** (Tenant–Lease–Property–Payment linkages)

## ****8. ADVANTAGES & DISADVANTAGES****

### ✅ ****Advantages****

● Real-time data validation and automation of lease-related workflows

● Modular design enables quick customization for different property types and lease terms

● Centralized system for managing tenants, leases, payments, and maintenance in one platform

● Automated notifications and reminders improve communication and reduce manual follow-ups

● Visual dashboards and reports enhance tracking of revenue, lease status, and service issues

### ⚠️ ****Disadvantages****

● Requires working knowledge of Salesforce platform for setup and customization

● Offline functionality is limited due to cloud-based dependency

● High reliance on correct setup of object relationships for flows and triggers to function accurately

● External service integrations (e.g., payment gateways, background checks) may require additional configuration and licensing

## ****9. CONCLUSION****

The Lease Management System project successfully delivered a customized Salesforce CRM solution tailored to lease lifecycle management. It streamlined operations including tenant onboarding, lease agreement generation, rent collection, and maintenance coordination. The system addressed common pain points such as manual data entry, delayed communications, and fragmented tracking by introducing automation, validations, and real-time reporting.

This implementation also served as a practical learning experience in both declarative and low-code Salesforce development. Key takeaways for the team included mastering object relationships, automation with flows and triggers, and the design of dynamic dashboards and reports. Overall, the project significantly improved operational efficiency and offered a scalable foundation for managing multiple properties and lease agreements within a unified platform.

## ****10. FUTURE SCOPE****

The Lease Management CRM can be further enhanced with the following capabilities:

* **Third-party payment gateway integration** for seamless rent collection and refund processing
* **Automated communication workflows**, such as SMS/email alerts for rent due dates, lease renewals, and maintenance updates
* **Mobile-access interface** via Salesforce Experience Cloud to allow tenants and property managers to view leases, make payments, and raise service requests on the go
* **AI-driven analytics and forecasting** using Salesforce Einstein to predict rent defaults, lease renewal probabilities, and maintenance frequency trends
* **Tenant self-service portals** for application tracking, document uploads, and support ticket management
* **Integration with accounting tools** (e.g., QuickBooks, Zoho Books) for synchronized financial reporting and ledger updates
* **Dynamic lease template generation** supporting multilingual and multi-region contracts for larger property portfolios