

Project Design Phase-II

Technology Stack (Architecture & Stack)

Date	30 OCTOBER 2025
Team ID	2F2CA610622DBCB9AF44E9641A0CB552
Project Name	LEASE MANAGEMENT
Maximum Marks	4 Marks

Technical Architecture:

The deliverable includes the architectural diagram of the Lease Management System and details of the components and technologies used (Table 1 & Table 2).

The architecture focuses on a web-based multi-user system that enables admins, property owners, and tenants to manage leases, rent payments, and maintenance requests efficiently.

The system uses a three-tier architecture (Presentation Layer, Application Layer, and Database Layer) deployed on a cloud-based platform for scalability and availability.

Reference: <https://developer.ibm.com/patterns/ai-powered-backend-system-for-order-processing-during-pandemics/>

Guidelines:

- Included all processes (application logic, validation, and automation modules).
- Clear separation between **local and cloud infrastructure**.
- Integration of **external APIs** for notifications or payment gateways.
- Defined **data storage and backup services**.
- Optional link for **machine learning model** (for future rent prediction or tenant analysis).

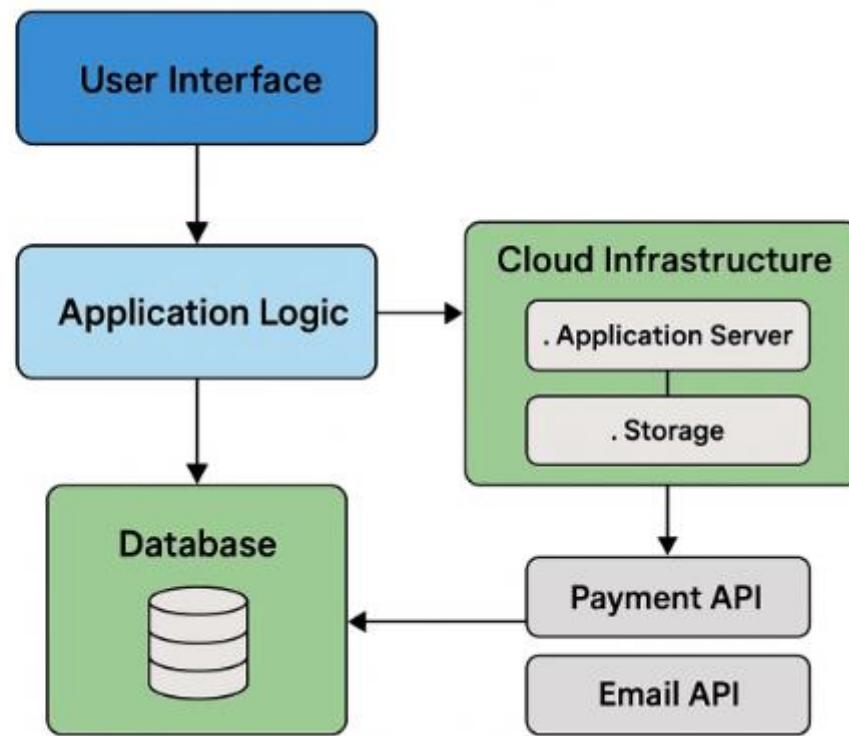


Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	Admin interacts via Web dashboard in ServiceNow.	HTML5, CSS3, JavaScript, React.js
2.	Application Logic-1	Checks if user is linked to an incident before allowing deletion	Node.js, Express.js
3.	Application Logic-2	Validates user status from incident table	Server-side validation using Node.js
4.	Application Logic-3	Sends notification if deletion is blocked.	Email/SMS API (e.g., Twilio, Nodemailer)

5.	Database	Stores user, incident, and assignment data	MySQL / PostgreSQL
6.	Cloud Database	Managed by ServiceNow cloud backend.	AWS RDS / Google Cloud SQL
7.	File Storage	Minimal use; logs stored internally	AWS S3 / Google Cloud Storage
8.	External API-1	(Optional) HRMS integration to verify user status	Razorpay / PayPal REST API
9.	External API-2	Not applicable	Gmail API / Twilio API
10.	Machine Learning Model	Not applicable for current use case	Python (Scikit-learn / TensorFlow)
11.	Infrastructure (Server / Cloud)	Hosted and managed on ServiceNow SaaS platform	AWS EC2 / Azure App Service / Google Cloud

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Not applicable (ServiceNow is proprietary)	React.js, Node.js, Express.js
2.	Security Implementations	Role-based access control, ACLs, secure scripts	JWT Authentication, HTTPS, bcrypt
3.	Scalable Architecture	SaaS-based, horizontally scalable via ServiceNow architecture	AWS / Google Cloud
4.	Availability	Highly available with ServiceNow cloud hosting	Cloud Load Balancer, Auto Scaling
5.	Performance	Optimized via asynchronous flows and indexed tables	Node.js, MySQL Indexing, Caching (Redis)