Performance Testing Phase

Performance Testing

Date	03-11-2025
Team ID	NM2025TMID07221
Project Name	Lease Management

1. Introduction

Objective:

Performance testing ensures that the Lease Management System (LMS) performs efficiently, reliably, and consistently under various operational loads. In real-estate leasing operations, system delays or failures may lead to Missed rent reminders , Inaccurate financial reporting , Tenant dissatisfaction , Compliance and financial losses .Therefore, performance testing is essential to validate system stability, accuracy, and responsiveness.

Key Goals of Performance Testing

Validate system response times

Ensure scalability for multiple tenants, properties, and users

Measure data accuracy and reliability under load

Identify performance bottlenecks and optimize system behavior

Scope

Testing focuses on critical LMS functions:

1. Lease contract creation and renewal
2. Tenant information updates
3. Automated rent invoicing and reminders
4. Maintenance request handling
5. Reporting and financial dashboards

2. Types of Performance Testing

Testing Type	Purpose	Scenario Example
Load Testing	Evaluate system with expected user traffic	50 property managers viewing lease dashboards at once
Stress Testing	Test system beyond peak capacity	200 tenants submitting maintenance requests simultaneously
Endurance Testing	Check stability over long usage	24-hour continuous operations for payments, lease updates
Spike Testing	Test sudden workload surge	Sudden influx of 100 rent payments in 10 minutes
Scalability Testing	Validate performance as data grows	Adding 1,000+ tenant and property records
Latency Testing	Measure response times	Fetching lease records or generating invoices in <2 seconds

3. Performance Metrics

Critical Metrics to Measure

1. Response Time

Time required to retrieve lease/tenant details

Target: <2 seconds per request

2. Throughput

Number of lease or payment transactions per second

Example: 100 invoices processed per minute without delays

3. Error Rate

Percentage of failed transactions under load

Target: <1% failure rate

4. Resource Utilization

Monitor CPU, memory, database, and network usage

Objective: Stable performance without server slowdown

5. Scalability

Must support increasing properties, tenants, and users without performance drop

6. Data Accuracy

Rent amounts, due dates, and maintenance logs must remain accurate under high load

4. Testing Workflows & Scenarios

Sample Scenario

50 users simultaneously:

Viewing property lease dashboards , Processing rent payments , Submitting maintenance requests , Accessing tenant data , Generating lease renewal reports

Measurement Focus:

Response time for dashboard and record retrieval

Payment processing accuracy

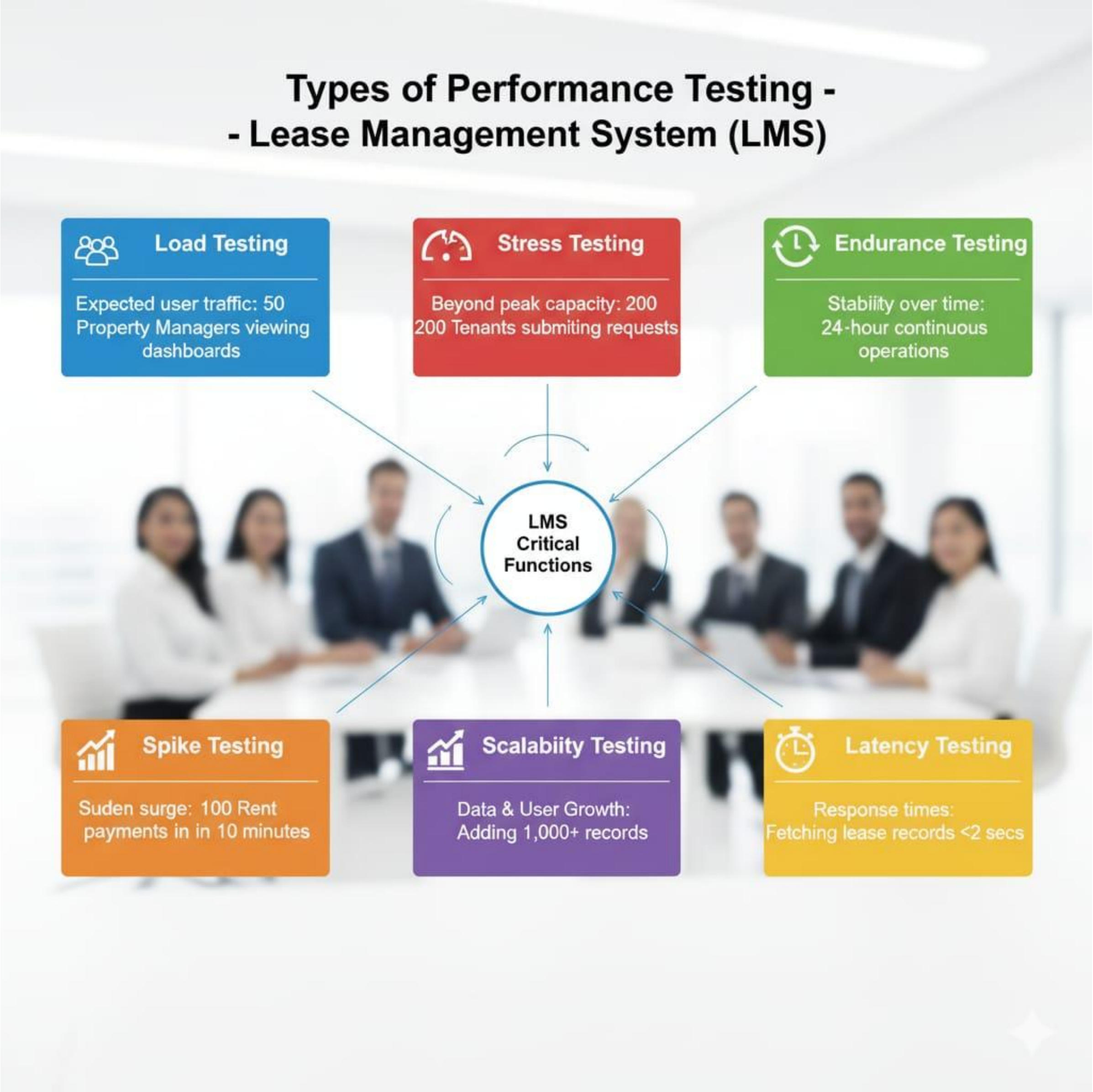
System behavior under parallel operations

Expected Outcome:

All tenant records and lease dashboards load within 2 seconds

Payment logs update correctly without duplication or loss

No timeout errors during peak usage



Types of Performance Testing -
- Lease Management System (LMS)

Load Testing
Expected user traffic: 50 Property Managers viewing dashboards

Stress Testing
Beyond peak capacity: 200 200 Tenants submiting requests

Endurance Testing
Stabiity over time: 24-hour continuous operations

LMS Critical Functions

Spike Testing
Suden surge: 100 Rent payments in in 10 minutes

Scalabiity Testing
Data & User Growth: Adding 1,000+ records

Latency Testing
Response times: Fetching lease records <2 secs

5. Conclusion

Performance testing ensures that the Lease Management System:

- Handles multiple property managers and tenants concurrently
- Maintains accurate lease and payment records
- Provides instant alerts and automated invoices without lag
- Supports future growth in number of properties and users
- Delivers smooth tenant service and operational reliability