Date	20-6-2025
Team ID	LTVIP2025MID29669
Project Name	Airlines Management system
Maximum Marks	

Chapter-7

7. Functional and Performance Testing

✅Functional Testing

This verifies that each feature works as intended:

Test Type	Purpose	Example
Unit Testing	Validate individual components	Test Apex class for flight status updates
Integration Testing	Ensure modules work together	Booking system integrates with payment gateway
System Testing	End-to-end validation	Simulate full flight booking and pilot assignment
User Acceptance Testing (UAT)	Confirm system meets user needs	Operators test flight scheduling and notifications
Regression Testing	Check for issues after updates	Re-test pilot assignment after new validation rule

Tools: Salesforce Developer Console, Workbench, Selenium (for UI), Postman (for API)

⚡ Performance Testing

This ensures the system performs well under load:

Test Type	Goal	Example
Load Testing	Assess system under expected traffic	Simulate 500 concurrent flight searches
Stress Testing	Push system beyond limits	Test booking module with 1000+ simultaneous users
Scalability Testing	Evaluate growth handling	Add 10,000 flight records and monitor response time
Recovery Testing	Check system resilience	Force failure and verify data recovery and alerts

Metrics to Monitor:

- Response time for flight search and booking
- CPU and memory usage during peak load
- Error rate under stress
- Time to recover after crash

Tools: JMeter, BlazeMeter, Salesforce Performance Assistant

7.1 Performance testing(trigger testing)

What Is Performance Testing?

Performance testing evaluates how the system behaves under expected and peak loads. It helps identify bottlenecks, slow responses, and system crashes before they affect real users.

- **Load Testing**: Simulates multiple users accessing the system simultaneously.
- **Stress Testing**: Pushes the system beyond normal limits to see how it recovers.
- **Spike Testing**: Tests sudden increases in user load.
- **Endurance Testing**: Checks system stability over extended periods.

Trigger Testing Explained

Trigger testing involves activating specific events or conditions to observe system behavior. In airline systems, this could include:

- **Flight Booking Surge**: Triggering a sudden spike in ticket bookings during a flash sale.
- **Check-in Deadline**: Simulating last-minute check-ins before flight departure.
- **System Failover**: Forcing a server crash to test backup and recovery mechanisms.
- **Payment Gateway Timeout**: Testing how the system handles delays or failures in payment processing.

Tools & Techniques

Some common tools used for performance and trigger testing include:

Tool	Purpose
JMeter	Load and stress testing
LoadRunner	Enterprise-grade performance
Selenium + TestNG	UI automation with triggers
Apache Kafka	Event-driven trigger simulation

Best Practices

- Use **realistic data** and user scenarios.

- Monitor **CPU, memory, and network usage** during tests.
- Automate triggers using scripts or event simulators.
- Validate **response times**, **error rates**, and **system recovery**.