# PERFORMANCE TESTING REPORT

# PERFORMANCE TESTING REPORT

**Application:** email.com
**Feature Tested:** Login functionality

## 1. Objective

The objective of this performance test is to evaluate the response time, throughput, and stability of the Gmail homepage (gmail.com) when accessed by multiple concurrent users using Apache JMeter.

## 2. Application Under Test

- **Application Name:** Gmail
- **URL:** https://gmail.com
- **Feature Tested:** Homepage access (GET request)

## 3. Test Tool

- **Apache JMeter (v5.6.3)**

## 4. Test Environment

- Operating System: Windows 10
- System RAM: 8 GB
- Network: Stable Internet Connection
- Testing Tool: Apache JMeter
- Protocol: HTTPS

## 5. Test Scenario

**Scenario Description:**

- Multiple users access the Gmail homepage simultaneously
- Each user sends a GET request to /
- Server responds with homepage content

## 6. Load Configuration

| **Parameter** | **Value** |
|---|---|
| Virtual Users | 50 |
| Ramp-Up Period | 3 sec |
| Loop Count | 3 |
| Total Requests | 150 |
| Request Method | Get |

## 7. Performance Metrics

- Response Time (Average, Min, Max)
- Throughput (requests/second)
- Error Rate (%)
- Standard Deviation

## 8. Pass / Fail Criteria

- Error Rate ≤ 1%
- Average Response Time ≤ 3 seconds
- Stable throughput without major fluctuations

# Study Performance concepts

## Part 1: Study Performance Testing Concepts

### Why is performance testing required?

Performance testing ensures your application can handle expected user load and identifies bottlenecks before production. Key reasons:

- User Experience: Slow applications lose users (3-second load time = 40% abandonment)
- Scalability: Understand how many users your system can handle
- Stability: Ensure the system doesn't crash under load
- Cost Optimization: Identify resource wastage
- Competitive Advantage: Fast applications win users

### Types of Performance Testing

1. **Load Testing:** Tests system behavior under expected load (normal conditions)
2. **Stress Testing:** Tests system beyond normal capacity to find breaking point
3. **Spike Testing:** Tests sudden increase/decrease in load
4. **Endurance Testing:** Tests system over extended period
5. **Volume Testing:** Tests with large amounts of data

### Key Performance Metrics (KPIs)

- Response Time: Time taken to receive response (should be < 2-3 seconds)
- Throughput: Requests processed per second (requests/sec)
- Error Rate: Percentage of failed requests (should be < 1%)
- Concurrent Users: Number of simultaneous users
- Resource Utilization: CPU, Memory, Network usage

# PART 2: Learn Apache JMeter Basics

## Part 2: Learn Apache JMeter Basics

### JMeter Interface Overview

When you open JMeter, you'll see:

1. **Test Plan** (root element) - Container for your entire test
2. **Thread Group** - Simulates users
3. **Samplers** - Actual requests (HTTP, FTP, etc.)
4. **Listeners** - Display results
5. **Configuration Elements** - Setup variables, defaults
6. **Assertions** - Validate responses

### Key Components Explained

**Thread Group** controls:

- **Number of Threads (Users)**: Virtual users
- **Ramp-up Period**: Time to start all users (e.g., 100 users in 10 seconds = 10 users/second)
- **Loop Count**: How many times to repeat the test