Uptut

PERFORMANCE TESTING

PORTFOLIO MANAGEMENT

Case Study | 2023

info@uptut.com

+91 92679 97641

PORTFOLIO MANAGEMENT TOOL

Agile Cockpit

Software

Netherlands

Performance testing for ‘Portfolio Management (Agile Cockpit)’ application is to ensure that the application can handle the expected load and provide smooth experience to end-users. Here's a case study outlining the performance testing approach for such an application.

INTRODUCTION TO PERFORMANCE TESTING

Performance testing is a type of software testing that focuses on evaluating the speed, responsiveness, scalability, stability, and overall performance of a software application or system under different conditions and workloads. The primary goal of performance testing is to ensure that an application performs satisfactorily and efficiently in real-world scenarios, even when subjected to various levels of stress and demand.

Performance testing encompasses several different types of testing, each with its specific objectives:

1 Load Testing:

Load testing involves assessing an application's performance under expected and peak load conditions. It determines how the application behaves when multiple users or transactions access it simultaneously. The goal is to identify performance bottlenecks and to ensure the application can handle its expected user load without degrading performance.

2 Stress Testing:

Stress testing pushes the application beyond its intended capacity to identify its breaking point. It helps determine how the system behaves when it experiences extreme loads or resource constraints. The objective is to uncover vulnerabilities and assess the system's ability to recover gracefully.

3 Scalability Testing

Scalability testing measures an application's ability to scale horizontally or vertically to accommodate an increasing number of users or transactions. It aims to determine the system's capacity for growth and whether additional resources (e.g., servers) can be added as needed.

4 Endurance Testing

Endurance testing, also known as soak testing, involves running an application under a sustained load for an extended period to identify memory leaks, resource consumption issues, or other long-term performance problems. It ensures the system can maintain performance over time.

To conduct performance test, specialized software is typically used to load the system with simulated user requests and record various performance metrics. Based on these metrics, bottleneck in performance can be identified and optimized.

An important aspect of conducting performance tests is ensuring that the test environment is as realistic as possible and that the tests are conducted under realistic load conditions. This way, the results ofthe tests can usually be most accurately applied to actual usage conditions

CLIENT - PORTFOLIO MANAGEMENT

TESTING OBJECTIVE

- To identify business critical scenarios and perform different types of Performance testing, to find the bottlenecks in the application.
- To ensure the application meets desired load (1000 concurrent users load).
- To identify and validate the application Performance bottleneck under maximum user load.
- Evaluate the applications response time under various scenarios.

TOOLS AND TECHNOLOGIES

Load Testing Tool
Apache JMeter

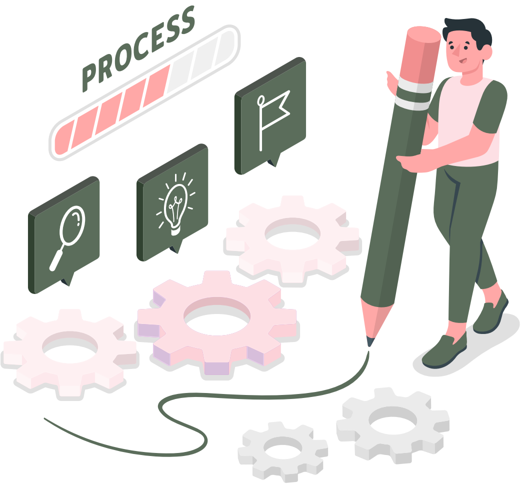
Test Data and Report
Generation - MS Office, Notepad

Amazon Web Services
(AWS) EC2

SOLUTION PREPARATION

How did we customise it to fit their requirement

- Initially we did POC of the staging application.
- Identified business critical scenarios.
- We used Apache JMeter for scripting and load generation.
- To avoid network latency, we launch EC2 instance in Paris region as the application is deployed in the **Netherlands.**
- We clubbed all scenarios into single script and then started with 50 users load - result was good for 50 users then we increase user by 100 and result was degraded for 100 users.



EXECUTION

- We performed proof of concept (POC) for login, logout scenarios to ensure technical flexibility
- **We created Test Plan & identify below scenario's –**

Prioritization Roadmap

Dashboard Epic Feature

Work Item Search

Create Epic Feature

Create Portfolio

Calculate Ranking

Report Static Navigation

Create Scenario

- Prepared workload model as per scenarios identification.
- Script execution.
- Executed Benchmark test.
- Selected AWS EC2 instance to reduce network latency (Paris Location).
- Executed load testing as per client requirement.
- Separated the scenarios with respect to performance of the applications.

OUR RECOMMENDATIONS

Based on the findings, the following recommendations were made:

- Use the Load balancer to distribute the incoming traffic
- Analyse the CPU, Memory utilization during the load test
- Optimize the APIs that are exceeding the SLAs
- Recommendations on fixing identified defects like Gateway Timeout and Internal Server errors

ParamJeet kaur
Product Owner, Agile Cockpit

"We can always rely on Uptut for tech expertise"

The Uptut team has been really resourceful in helping us improve the quality of our application and making it a high-performing asset. The processes and test cases were exceptionally designed and the support throughout the various rounds of testing is really appreciable. The detailed report has enabled us to fix our product holistically..

Develop skills with your team

Offer unmatched customer experience through speed and stability with Performance Testing

Talk to an Expert

✉ info@uptut.com

☎ +91 92679 97641