Fontys University of Applied Sciences

Load and stress testing

GetawayGo

Anna Kadurina
18/01/2025

Table of Contents

Introduction

This document outlines the load balancing tests conducted up till now to evaluate the performance, reliability, and scalability of the system under varying traffic conditions. Load balancing is a critical aspect of modern distributed systems, ensuring that incoming requests are efficiently distributed across multiple servers to optimize resource utilization and maintain high availability.

Load Testing

For the load testing, Azure Load Testing was utilized, which is a cloud based solution that uses Apache Jmeter as its underlying framework. The frontend URL was specified as the target for testing, representing the entry point for end-user interactions.

Metrics Overview

1. Response Time – It measures the time taken for the system to process requests and return a response. A threshold of < 200 ms was set.
2. Throughput (requests/sec) – It indicates the number of requests handled by the system per second, reflecting its capacity to process concurrent users. The goal was set based on expected user traffic.
3. Error Percentage – Represents the percentage of failed requests. A threshold of 0% was maintained.
4. Latency – Captures the delay between a user request and the start of processing.

5. Server-side Metrics – Monitored backend resources.

Configuration

- Load pattern – Linear increase of users over a 5 minute ramp-up period.
- Concurrent users – 50 per engine, scaled with multiple engines for higher loads.
- Test Duration – 10 minutes.
- Auto-stop Criteria – Tests were configured to stop if the error percentage exceeded 90% within a 60-second window.

Results

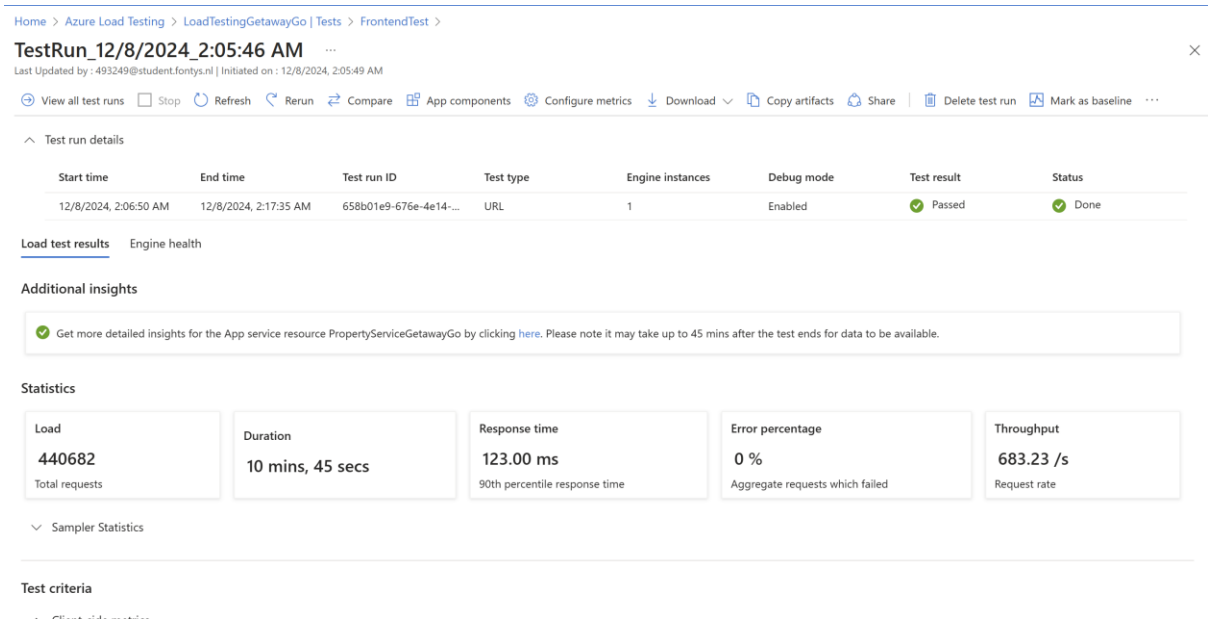


Figure 1 – Statistics of the test run

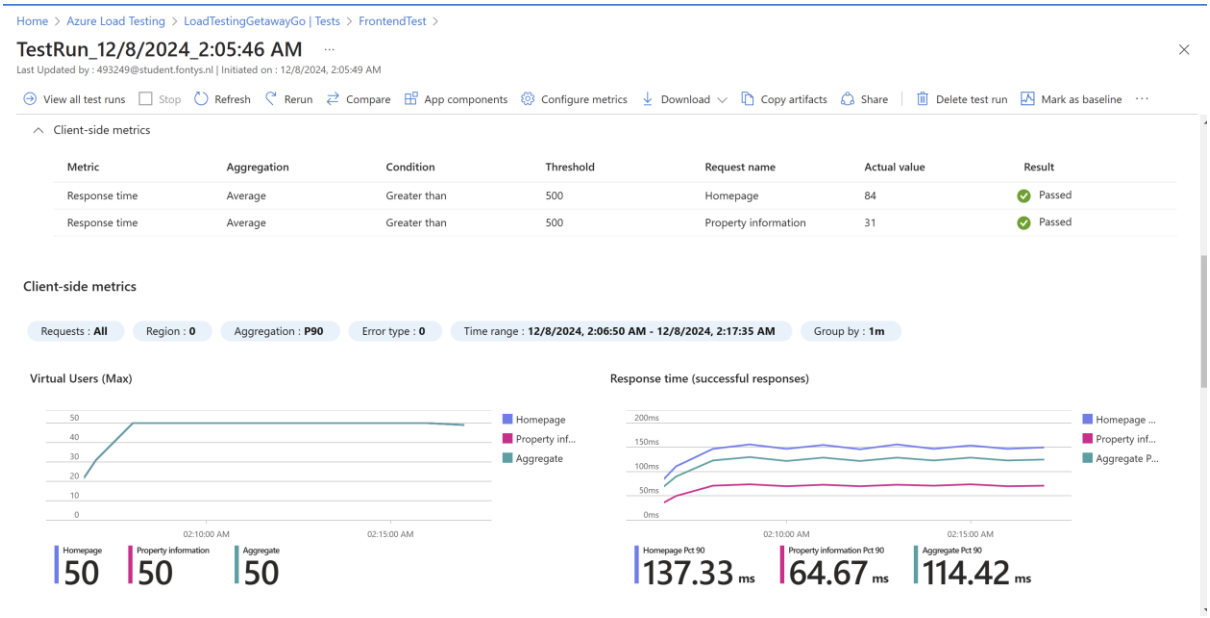


Figure 2 – Client-side metrics

Figure 3 – Client-side metrics

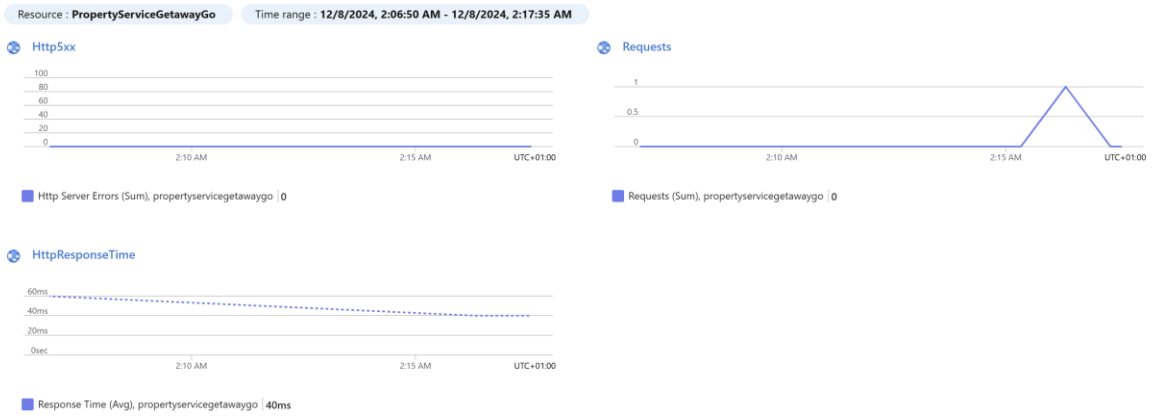
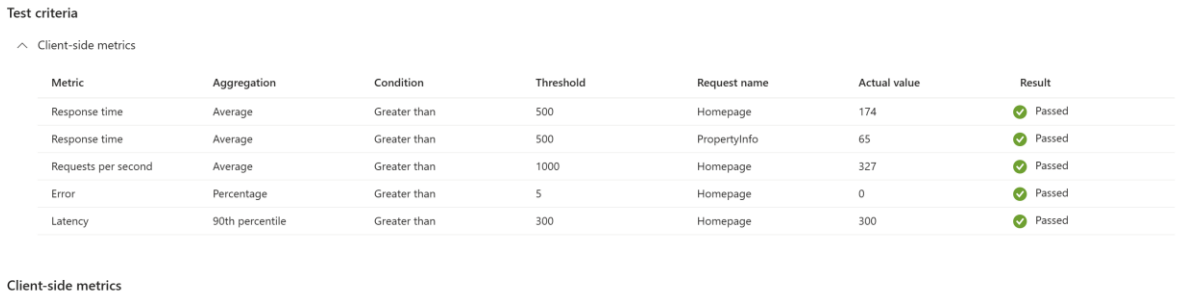


Figure 4 – Server-side metrics


Test criteria

Client-side metrics

Metric	Aggregation	Condition	Threshold	Request name	Actual value	Result
Response time	Average	Greater than	500	Homepage	174	Passed
Response time	Average	Greater than	500	PropertyInfo	65	Passed
Requests per second	Average	Greater than	1000	Homepage	327	Passed
Error	Percentage	Greater than	5	Homepage	0	Passed
Latency	90th percentile	Greater than	300	Homepage	300	Passed

Client-side metrics

Figure 5 – Test criteria

The load testing successfully demonstrated the system's ability to handle high traffic while maintaining optimal performance. Over the test duration of 10 minutes and 45 seconds, the system processed a total of 440,682 requests with an average throughput of 683.23 requests per second.

The average response time was 123 ms, with the 90th percentile response time remaining within acceptable limits, indicating consistent responsiveness under load.

The error rate was 0%, meaning no requests failed during the test.

Stress Testing

Stress testing is designed to evaluate system performance under high load and identify any bottlenecks. This test simulated a significant load on the system, achieving over 3 million total requests in a 20-minute duration. The test was made again with Azure Load Testing but in a Spike pattern, meaning that the users increase drastically over time periods.

Metrics Overview

- Total Requests: 3.08 million requests processed during the test, which demonstrates the system's ability to handle a substantial volume of traffic.
- Response Time: The average response time was 109 ms, which is excellent and well within acceptable thresholds.
- 90th Percentile Response Time: 131 ms shows that most responses were quick, though slightly higher latency occurred under heavy load.
- Error Percentage: 57.74% of requests failed, highlighting potential areas for optimization or resource scaling.
- Throughput: The system processed an impressive 2562 requests per second, which demonstrates its scalability potential.

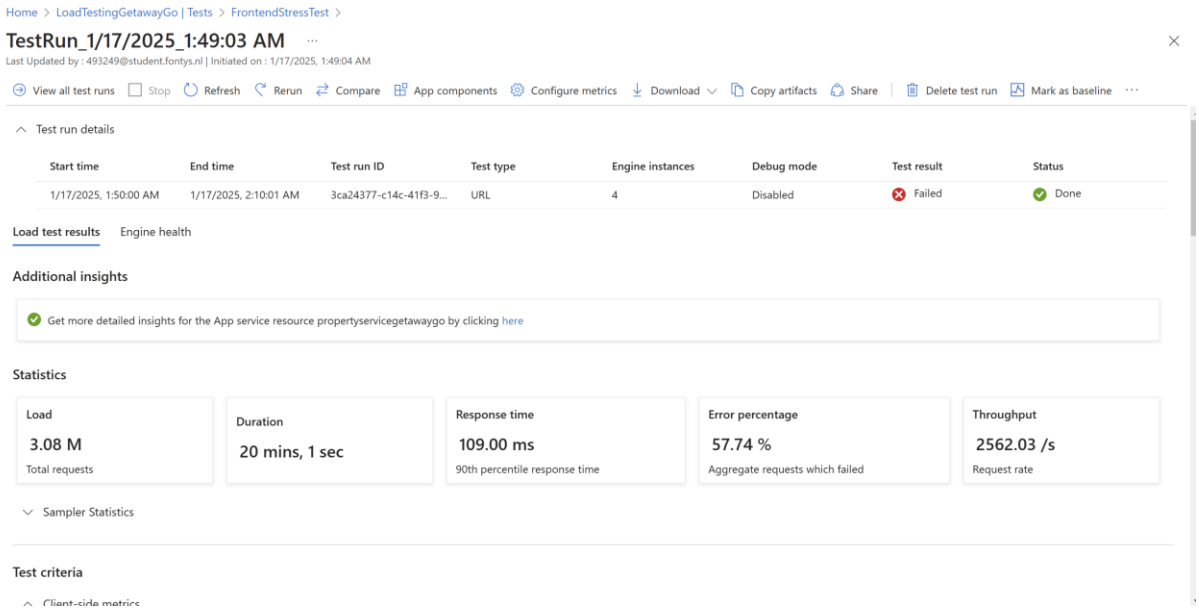


Figure 6 –Statistics of the stress testing

Results

While the system showed strong throughput and response times, the high error rate and latency failures indicate resource limitations under stress.

- Response Time (Passed): The homepage consistently responded with an average of 76 ms, far below the threshold of 2000 ms, which is a strong positive indicator of performance under load.
- Latency (Failed): Average latency exceeded the threshold of 1500 ms. The 95th percentile latency was 131 ms, slightly above the acceptable range, indicating performance degradation under peak load.
- Error Rate (Failed): With a failure rate of 57.74%, the system struggled to maintain reliable performance under extreme conditions.

Figure 7 – Client-side metrics

Conclusion

The load stress tests demonstrated that the system performed admirably given the constraints of using minimal, cost-efficient resources. Over 3 million requests were processed in just 20 minutes, with an impressive average response time of 109 ms and a throughput of 2562 requests per second. These metrics highlight the system's capability to handle significant traffic efficiently under limited resource allocation. The test results affirm that the system is robust and scalable, even with minimal investment. With proper resource scaling, such as leveraging auto-scaling and optimizing high-latency components, the system has the potential to achieve exceptional performance under heavier loads.