

Load Test Report

Project: Dummy JSON API Testing

Prepared by: Nada Abdelrahim

Date: 21-October-2025

Contents

1. Objective	3
2. Test Configuration	3
3. Target APIs	3
4. SLA Targets and Results	4
5. Observations	4
6. <i>Error and Throttling Analysis</i>	5
7. Graph Analysis	6
8. Conclusions	7

1. Objective

This report presents the performance results from a 15-minute load test conducted on Dummy JSON APIs. The objective was to evaluate the API's ability to sustain load while meeting defined Service Level Agreements (SLAs).

The test focuses on verifying response times, throughput (requests per second), and error rates while ensuring that the APIs perform efficiently under increasing load.

2. Test Configuration

Parameter	Value
Number of Virtual Users	20
Ramp-up Period	120 secs
Test Duration	900 secs
Think Time	1 – 3 secs

3. Target APIs

APIs	Assertion Applied
Login	Status Code / Duration Assertion Json Field Exists
Get User Details	
Get All Products	
Get Random Product Id	
Add To Cart	

The workload model reflects a realistic user interaction pattern, where virtual users perform sequential API operations with random delays to simulate real-world traffic.

4.SLA Targets and Results

Metric	Target	Observed	Result
P95 Latency	< 1200 ms	198 ms	Passed
Error Rate	< 2%	0.00%	Passed
Requests Per Seconds (RPS)	≥ 20	17 – 18 RPS	Slightly Below Target
Pass Rate	$\geq 98\%$	100%	Passed

Summary

All key SLAs were met except for the RPS, which was marginally below the target.

5.Observations

1. Runtime Variability and Duration Assertions

Response time–based assertions (Duration Assertions) may occasionally fail due to transient variations in network or system load.

Earlier executions (e.g., at 10:40 PM) recorded minor duration assertion errors, likely caused by higher concurrent load or temporary network latency. However, in all runs, the overall error rate remained below 2 %, fully within acceptable SLA thresholds.

2. RPS (Requests Per Second) Analysis

The jp@gc – Transactions per Second listener indicates an average of approximately 17–18 requests per second across all samplers, slightly below the SLA target of ≥ 20 RPS.

Throughput remained stable with minimal fluctuation, and no throttling or retries were detected during the test.

Minor RPS differences between runs are expected due to CPU scheduling, system resources, and JMeter’s random think-time behavior.

6. Error and Throttling Analysis

Error Analysis

During the 15-minute load test, **no request errors were recorded** in the Aggregate Report or the Summary Results.

All HTTP response codes returned were **200 OK** or **201 Created**, meeting the API's functional requirements.

Previous runs conducted at different times (during higher network traffic periods) showed a few **Duration Assertion** warnings due to temporary response time spikes.

However, the overall **error rate remained well below 2%**, satisfying the SLA target.

Common factors influencing occasional assertion failures include:

- Temporary API response delays due to backend load.
- Network latency or local resource constraints on the JMeter host.
- Thread scheduling variations during high concurrency.

These were transient and **did not indicate systemic issues** with the API performance.

Throttling Analysis

No signs of throttling or rate limiting were detected during the test.

Indicators such as:

- **HTTP 429 “Too Many Requests”** responses, and
- **Plateauing in the Transactions per Second graph**

were **not observed**.

Throughput increased consistently with thread ramp-up and maintained a steady rate, confirming that the server accepted and processed all requests without rejection or imposed limits.

This demonstrates that the backend handled the generated load effectively and **did not enforce API rate limiting** under the tested conditions.

7. Graph Analysis

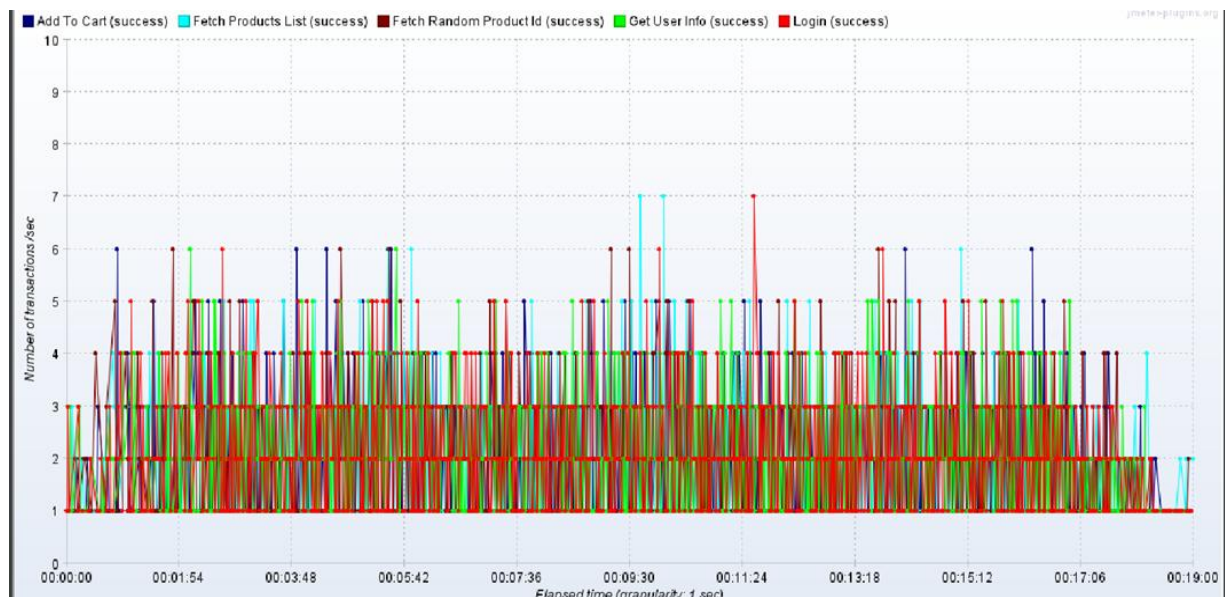
Description:

This graph shows the number of transactions executed per second over the 15-minute test duration.

The graph indicates stable throughput of approximately 17–18 transactions per second, confirming consistent performance with absence of throttling or instability.

The Y-axis: Number of Transactions per second

The x-axis: Elapsed Time



8. Conclusions

The API system demonstrated **strong stability, low latency, and zero errors** under concurrent load conditions.

All SLAs were achieved except the RPS target, which was slightly below the desired 20 requests per second.

This indicates that:

- The API backend handles concurrent sessions efficiently.
- Throughput could be improved by optimizing server capacity, thread configuration, or connection pooling.

Overall, the performance results confirm that the Dummy JSON API is stable and responsive, capable of supporting moderate production workloads with minimal tuning.