Performance Test Report - Jan 22, 2025 (#4)

Open in Postman

Postman collection: API_Automation_PositiveTests

Report exported on: Jan 22, 2025, 17:10:37 (GMT)

Test setup

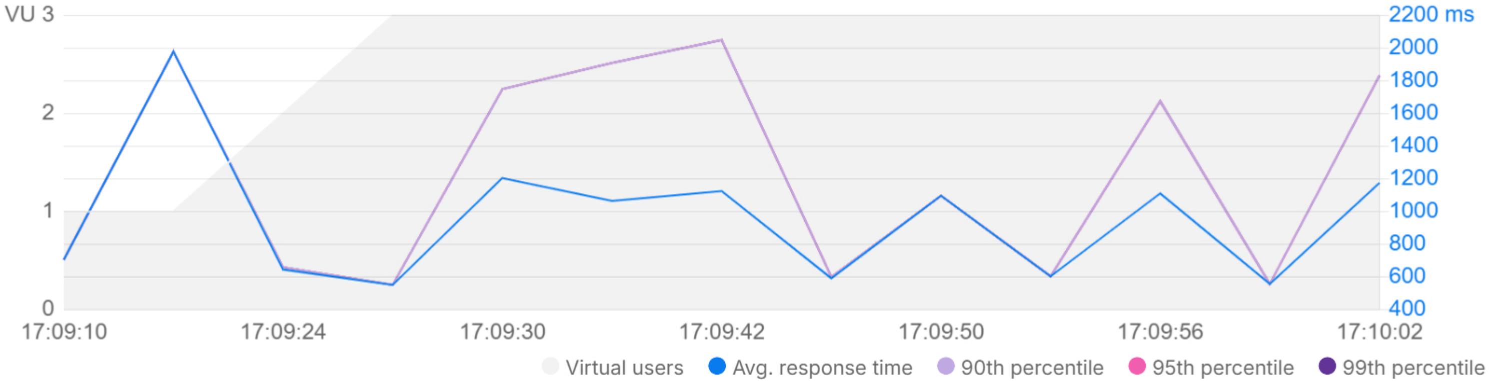
Virtual users 3 VU	Start time Jan 22, 17:08:56 (GMT)	Load profile Ramp up (15 seconds)
Duration 1 minute	End time Jan 22, 17:10:05 (GMT)	Environment -

1. Summary

Total requests sent	Throughput	Average response time	Error rate
22	0.32 requests/second	979 ms	0.00 %

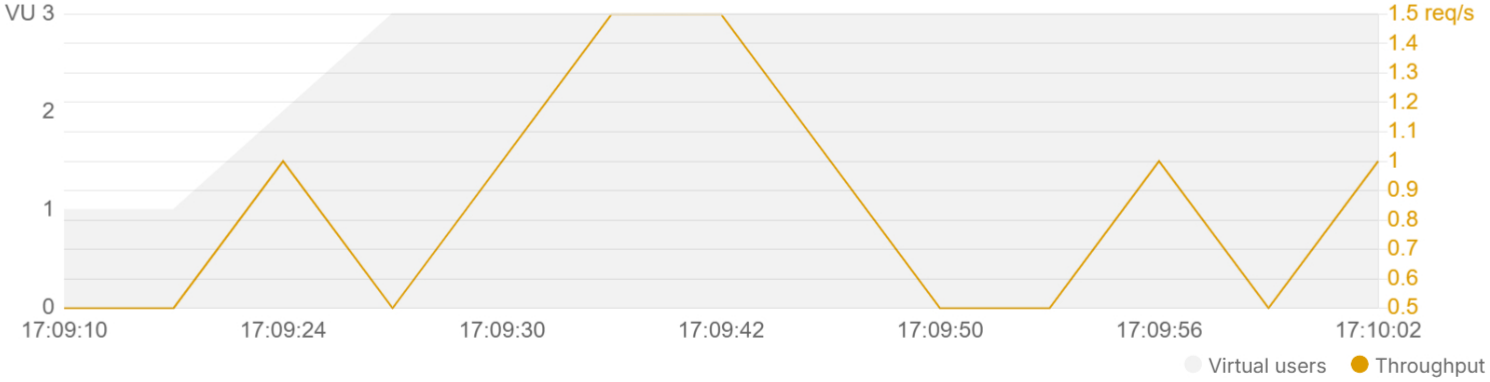
1.1 Response time

Response time trends during the test duration.



1.2 Throughput

Rate of requests sent per second during the test duration.



POSTMAN

1.3 Requests with slowest response times

Top 5 slowest requests based on their average response times.

Request	Resp. time (Avg ms)	90th (ms)	95th (ms)	99th (ms)	Min (ms)	Max (ms)
GET GetPlace {{base_url}}/maps/api/place/get/json?place_id={{place_id}}&key=qaclick123	1,866	2,050	2,050	2,050	1,676	2,050
POST AddPlace {{base_url}}/maps/api/place/add/json?key= qaclick123	694	1,098	1,098	1,098	548	1,098
PUT UpdatePlace {{base_url}}/maps/api/place/update/json?place_id={{place_id}}&key=qaclick123	617	693	693	693	519	693
DELETE DeletePlace {{base_url}}/maps/api/place/delete/json?key=qaclick123	598	634	634	634	553	634

2. Metrics for each request

The requests are shown in the order they were sent by virtual users.

Request	Total requests	Requests/s	Min (ms)	Avg (ms)	90th (ms)	Max (ms)	Error %
POST AddPlace {{base_url}}/maps/api/place/add/json?key= qaclick123	7	0.10	548	694	1,098	1,098	0
GET GetPlace {{base_url}}/maps/api/place/get/json?place_id={{place_id}}&key=qaclick123	6	0.09	1,676	1,866	2,050	2,050	0
PUT UpdatePlace {{base_url}}/maps/api/place/update/json?place_id={{place_id}}&key=qaclick123	5	0.07	519	617	693	693	0
DELETE DeletePlace {{base_url}}/maps/api/place/delete/json?key=qaclick123	4	0.06	553	598	634	634	0

3. Errors

This run has no errors

All requests were sent successfully and returned a 2xx response code.

Testing API performance on Postman

Postman enables you to simulate user traffic and observe how your API behaves under load. It also helps you identify any issues or bottlenecks that affect performance.

Learn more about testing API performance.