

Performance test report - Mar 12, 2024 (#8)

Open in Postman

Postman collection: NBA GraphQL
Report exported on: Mar 12, 2024, 11:48:43 (CDT)

Test setup

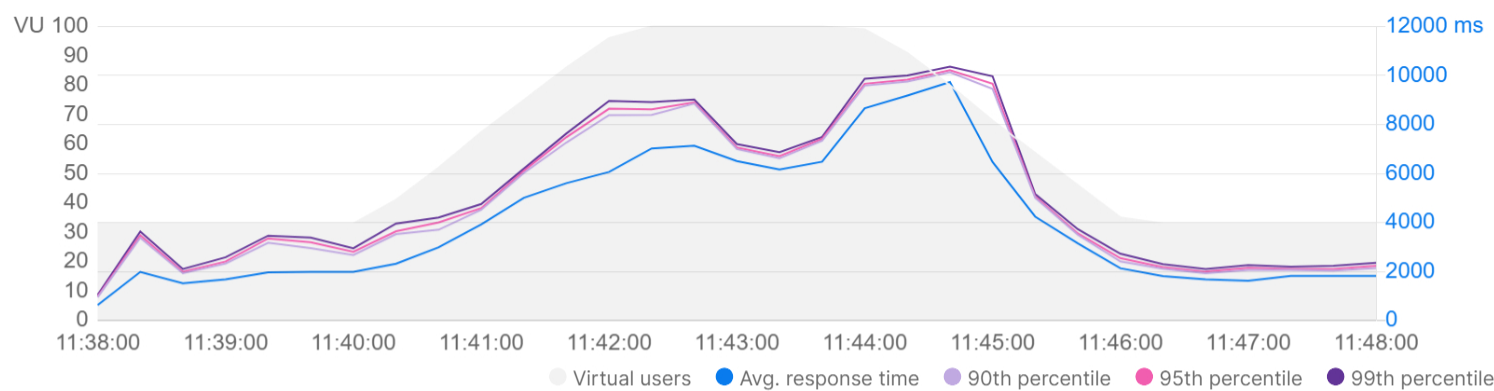
Virtual users	Start time	Load profile
100 VU	Mar 12, 11:38:12 (CDT)	Peak
Duration	End time	Environment
10 minutes	Mar 12, 11:48:19 (CDT)	GraphQL

1. Summary

Total requests sent	Throughput	Average response time	Error rate
8,348	13.76 requests/second	3,918 ms	0.00 %

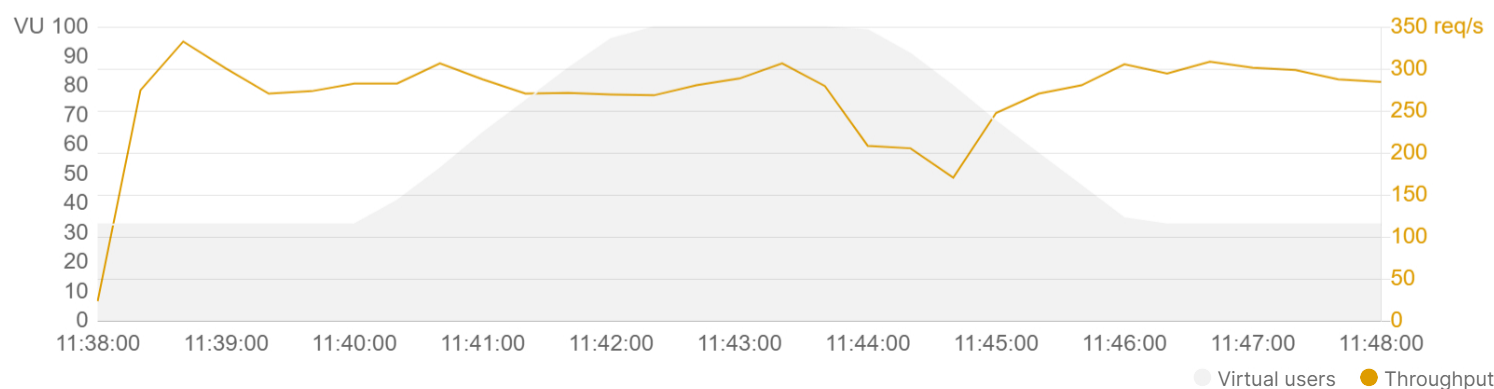
1.1 Response time

Response time trends during the test duration.



1.2 Throughput

Rate of requests sent per second during the test duration.



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 Harden Advanced + Total Copy {{url}}/graphql/	4,024	8,051	9,235	9,959	727	10,238
GET LeBron Advanced + Total {{url}}/graphql/	4,011	7,582	9,335	9,927	766	10,498
GET Kobe Advanced + Total {{url}}/graphql/	3,998	7,772	9,148	9,871	741	10,263
GET Iverson Advanced + Total {{url}}/graphql/	3,920	7,455	8,994	9,810	711	10,284
GET Advanced LeBron {{url}}/graphql/	3,793	7,665	8,850	9,702	229	10,156

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 %
GET Advanced LeBron {{url}}/graphql/	1,459	2.40	229	3,793	7,665	10,156	0
GET Totals Harden {{url}}/graphql/	1,389	2.29	598	3,773	7,169	10,133	0
GET Iverson Advanced + Total {{url}}/graphql/	1,384	2.28	711	3,920	7,455	10,284	0
GET Kobe Advanced + Total {{url}}/graphql/	1,380	2.27	741	3,998	7,772	10,263	0
GET LeBron Advanced + Total {{url}}/graphql/	1,371	2.26	766	4,011	7,582	10,498	0
GET Harden Advanced + Total Copy {{url}}/graphql/	1,365	2.25	727	4,024	8,051	10,238	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](#).