Performance test report - Mar 21, 2024 (#17)

Open in Postman

Postman collection: NBA GraphQL

Report exported on: Mar 21, 2024, 0:45:43 (CDT)

Test setup

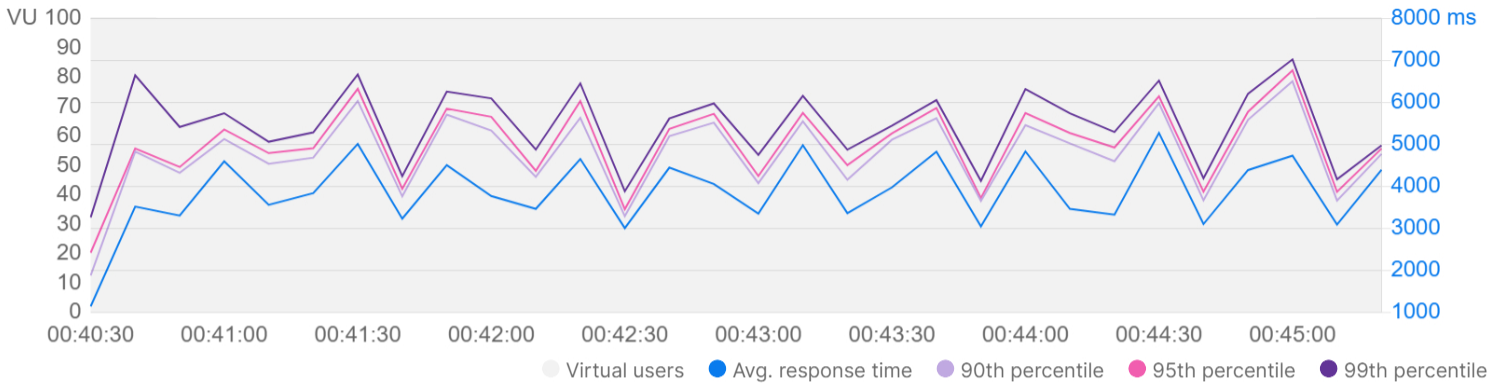
Virtual users	Start time	Load profile
100 VU	Mar 21, 0:40:16 (CDT)	Fixed
Duration	**End time**	**Environment**
5 minutes	Mar 21, 0:45:22 (CDT)	GraphQL

1. Summary

Total requests sent	Throughput	Average response time	Error rate
6,634	21.67 requests/second	3,799 ms	0.00 %

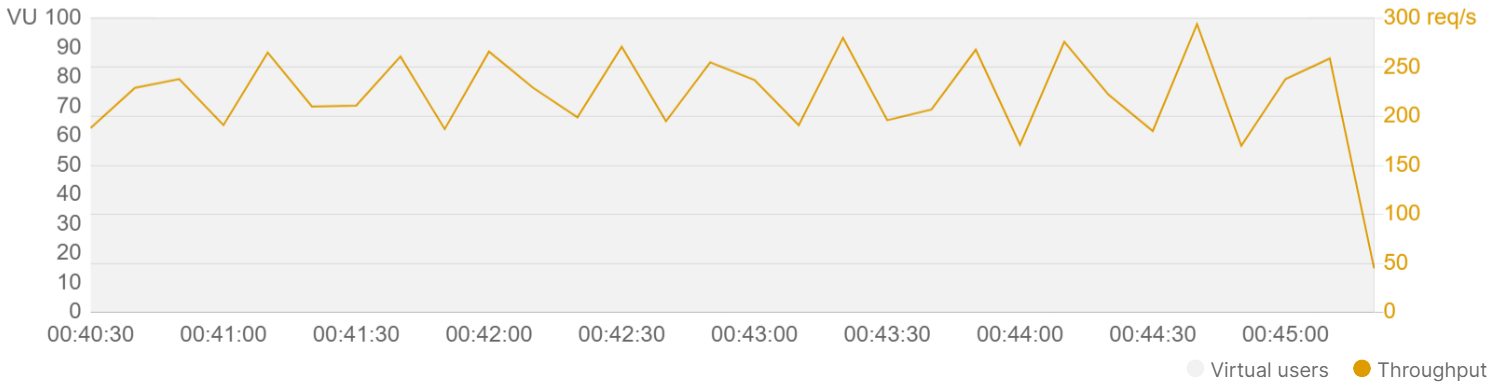
1.1 Response time

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 Season - Combined Data {{url}}/graphql/	4,681	5,920	6,240	6,767	559	7,117
GET Playoffs - Advanced Data {{url}}/graphql/	4,236	5,420	5,741	6,396	805	6,745
GET Playoffs - Totals Data http://209.38.172.107/graphql/	3,646	4,956	5,262	5,720	2,174	6,398
GET Season - Totals Data {{url}}/graphql/	3,553	4,828	5,132	5,781	178	6,506
GET Team Data http://209.38.172.107/graphql/	3,430	4,812	5,142	5,707	2,203	6,547

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 Season - Advanced Data {{url}}/graphql/	1,161	3.79	220	3,273	4,427	6,339	0
GET Season - Totals Data {{url}}/graphql/	1,144	3.74	178	3,553	4,828	6,506	0
GET Season - Combined Data {{url}}/graphql/	1,105	3.61	559	4,681	5,920	7,117	0
GET Playoffs - Advanced Data {{url}}/graphql/	1,081	3.53	805	4,236	5,420	6,745	0
GET Playoffs - Totals Data http://209.38.172.107/graphql/	1,074	3.51	2,174	3,646	4,956	6,398	0
GET Team Data http://209.38.172.107/graphql/	1,069	3.49	2,203	3,430	4,812	6,547	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.