

Postman collection: MeDQATask

Report exported on: Feb 24, 2025, 21:23:13 (GMT+1)

Test setup

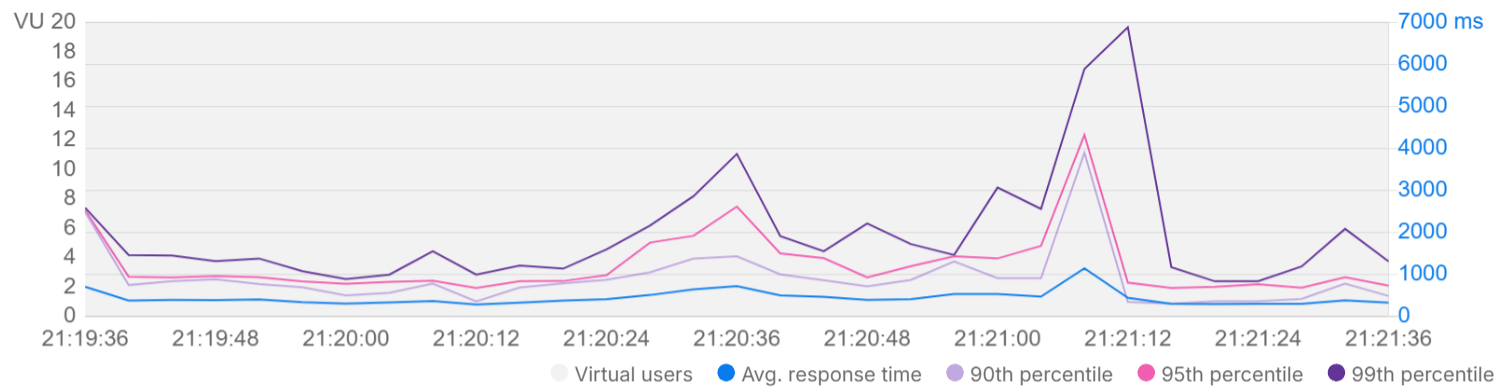
Virtual users	Start time	Load profile
20 VU	Feb 24, 21:19:32 (GMT+1)	Fixed
Duration	End time	Environment
2 minutes	Feb 24, 21:21:39 (GMT+1)	First Environment

1. Summary

Total requests sent	Throughput	Average response time	Error rate
4,103	32.36 requests/second	421 ms	0.34 %

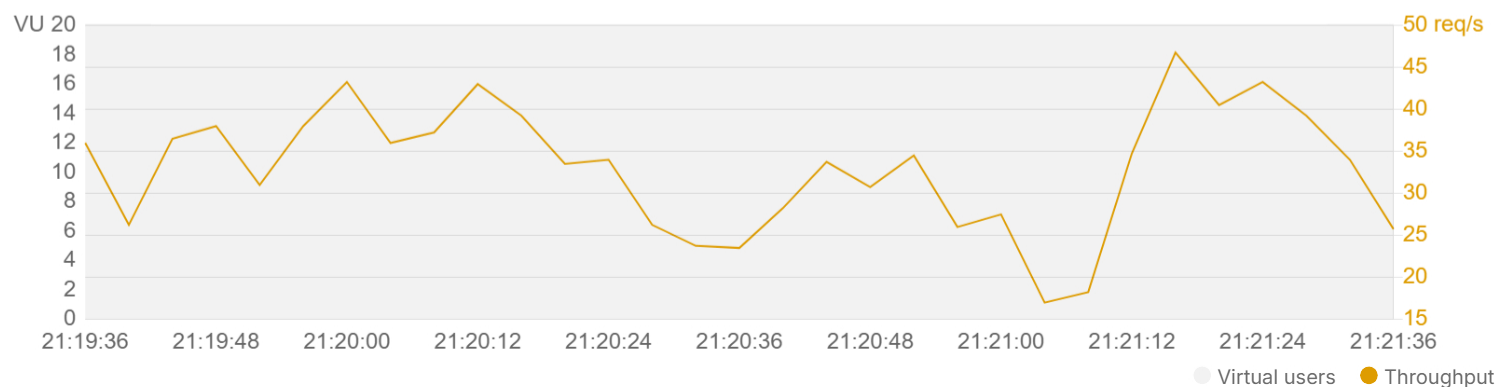
1.1 Response time

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 Retrieve All Bookings {{target_url}}/booking	877	1,473	1,882	3,702	132	6,886
GET Check liveliness {{target_url}}/ping	540	1,313	2,498	3,237	125	9,948
POST Create Token {{target_url}}/auth	379	791	832	1,172	117	10,311
GET Retrieve Booking by Date {{target_url}}/booking?checkin=2025-02-23&checkout=2025-02-27	365	511	845	1,877	112	4,293
GET Retrieve Booking by Name {{target_url}}/booking?firstname=sally&lastname=brown	356	680	863	1,458	122	3,300

1.4 Requests with most errors

Top 5 requests with the most errors, along with the most frequently occurring errors for each request.

Request	Total error count	Error 1	Error 2	Other errors
GET Get Booking {{target_url}}/booking/{{randomBookingId}}	11	404 Not Found (11)	-	0
PUT Update Booking {{target_url}}/booking/{{bookingId}}	1	405 Method Not Allowed (1)	-	0
PATCH Partial Update Booking {{target_url}}/booking/{{bookingId}}	1	405 Method Not Allowed (1)	-	0
DELETE Delete Booking {{target_url}}/booking/{{bookingId}}	1	405 Method Not Allowed (1)	-	0

2. Metrics for each request

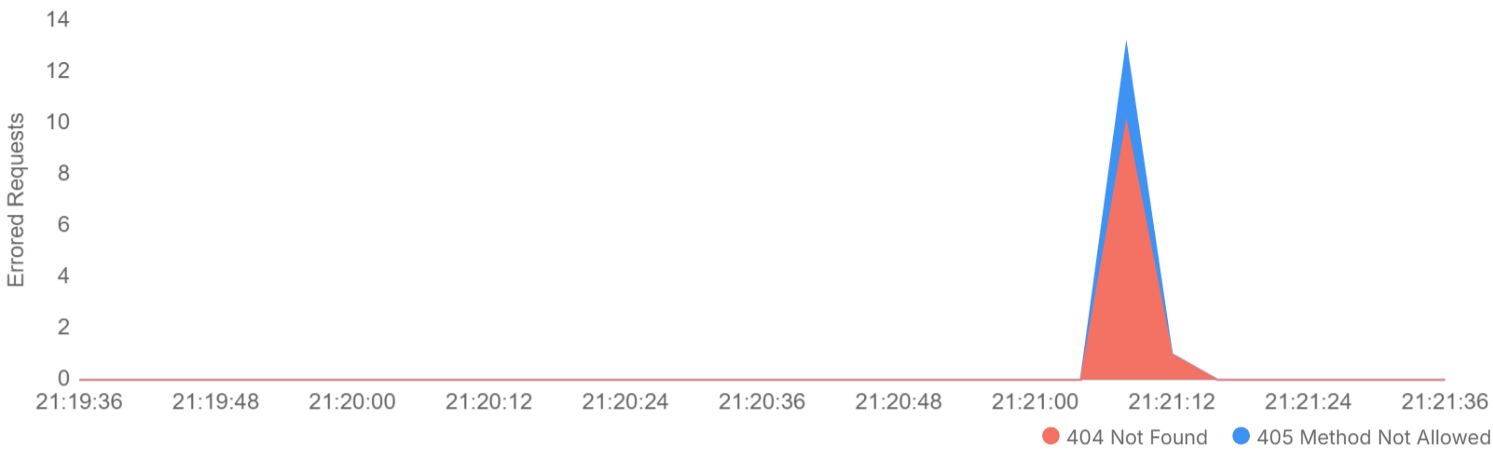
The requests are shown in the order they were sent by virtual users.

Request	Total requests	Requests/s	Min (ms)	Avg (ms)	90th (ms)	Max (ms)	Error %
GET Check liveliness {{target_url}}/ping	414	3.27	125	540	1,313	9,948	0
POST Create Token {{target_url}}/auth	413	3.26	117	379	791	10,311	0
GET Retrieve All Bookings {{target_url}}/booking	412	3.25	132	877	1,473	6,886	0
GET Retrieve Booking by Name {{target_url}}/booking?firstname=sally&lastname=brown	411	3.24	122	356	680	3,300	0
GET Retrieve Booking by Date {{target_url}}/booking?checkin=2025-02-23&checkout=2025-02-27	410	3.23	112	365	511	4,293	0
GET Get Booking {{target_url}}/booking/{{randomBookingId}}	410	3.23	127	348	652	5,892	2.68
POST Create Booking {{target_url}}/booking	410	3.23	118	342	427	6,302	0
PUT Update Booking {{target_url}}/booking/{{bookingId}}	408	3.22	119	352	725	3,744	0.25
PATCH Partial Update Booking {{target_url}}/booking/{{bookingId}}	408	3.22	126	317	425	1,675	0.25
DELETE Delete Booking {{target_url}}/booking/{{bookingId}}	407	3.21	123	325	426	1,839	0.25

3. Errors

3.1 Error distribution over time

Top 5 error classes observed during the test duration.



3.2 Error distribution for requests

Errored requests grouped by error class, along with the error count for each class.

Error class	Total counts
404 Not Found	11
GET Get Booking	11
405 Method Not Allowed	3
PUT Update Booking	1
PATCH Partial Update Booking	1
DELETE Delete Booking	1



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](#).