

[Open in Postman](#)

Edit comment Stress Test (from 0 to 6,000) - Oct 29, 2025 (#26)

Postman collection: [mirai-api](#)

Report exported on: Oct 29, 2025, 2:35:52 (CST)

Test setup

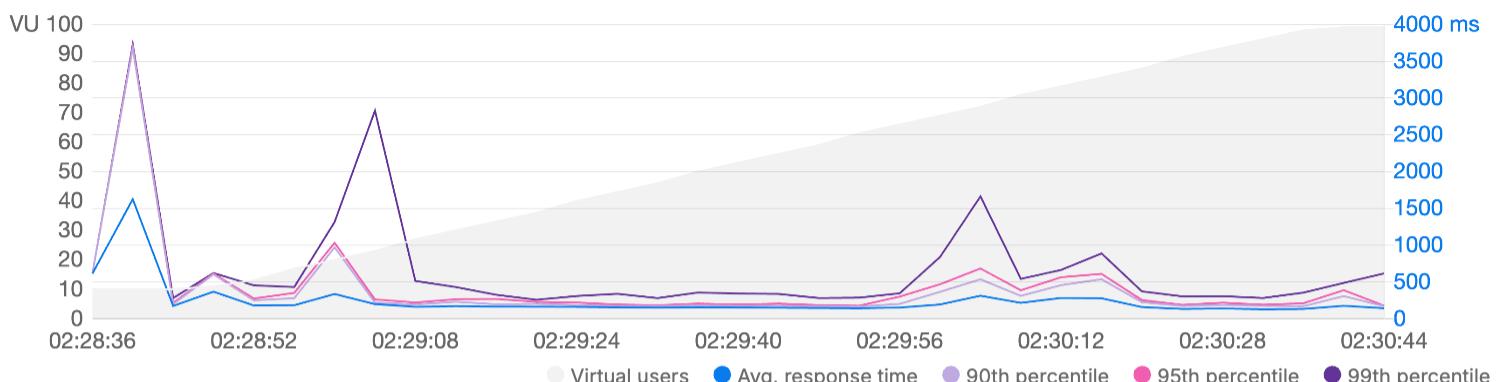
Virtual users	Start time	Load profile
100 VU	Oct 29, 2:28:37 (CST)	Ramp up (1 minute 50 seconds)
Duration	End time	Environment
2 minutes	Oct 29, 2:30:44 (CST)	Mirai environment

1. Summary

Total requests sent	Throughput	Average response time	Error rate
5,753	45.23 requests/second	181 ms	28.63 %

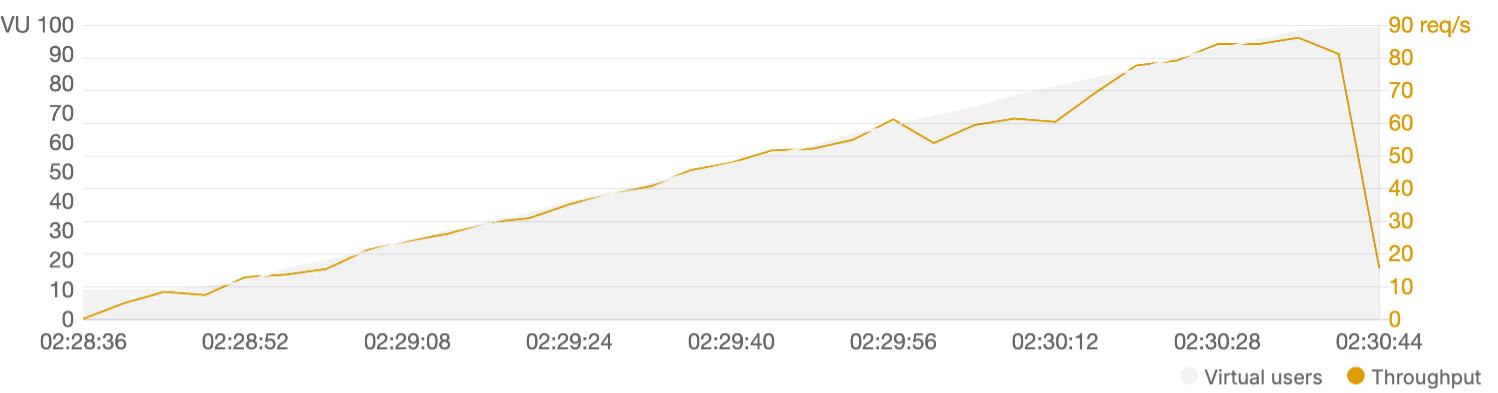
1.1 Response time

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)
PUT Edit comment {{ ...Mirai }}/forums/68e7e2e51c2daa89776d242e/comments/690111e3400 99bf3e8a85ada	181	274	365	641	80	3,730

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
PUT Edit comment {{ ...Mirai }}/forums/68e7e2e51c2daa89776d242e/comments/690111e3400 99bf3e8a85ada	1,647	503 Service Unavailable (1647)	-	0

2. Metrics for each request

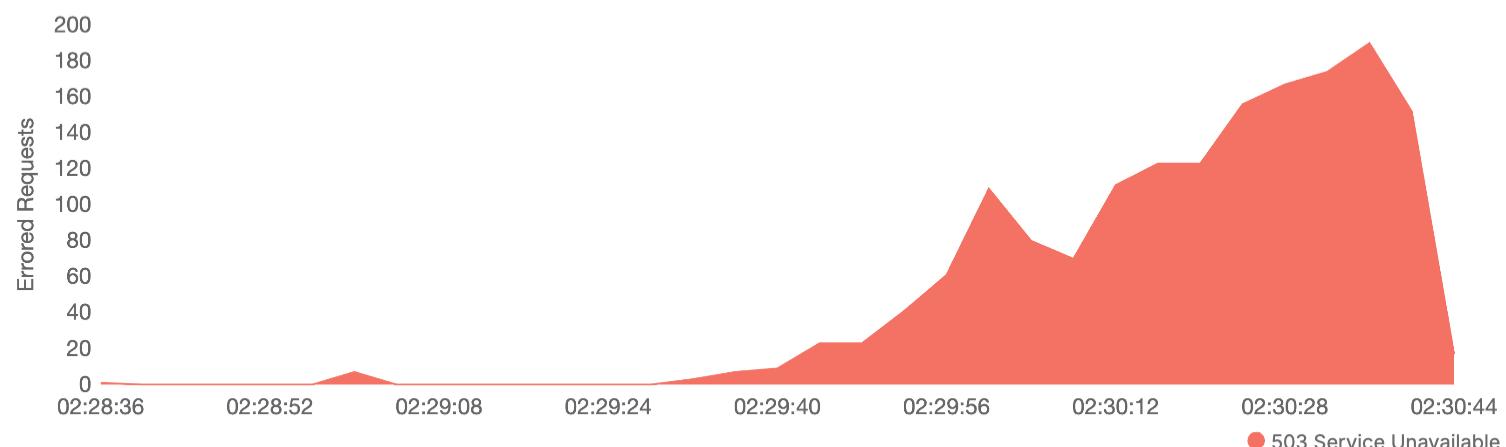
The requests are shown in the order they were sent by virtual users.

Request	Total requests	Requests/s	Min (ms)	Avg (ms)	90th (ms)	Max (ms)	Error %
PUT Edit comment <code>{{ ...Mirai }}/forums/68e7e2e51c2daa89776d242e/comments/690111e340099bf3e8a85ada</code>	5,753	45.23	80	181	274	3,730	28.63

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
503 Service Unavailable	1647
PUT Edit comment	1,647



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](#).