Course Matrix/term-group-project-c01w25-project-course-matrix

Performance Test Report

Environment

Device: e2-medium (2 vCPUs, 4 GB Memory)
Operating System: Debian/Linux
Browser: Mozilla Firefox
Server: Supabase backend hosted on google cloud VM
Testing tools: Mozilla Firefox Dev and J-Meter

Performance Metrics

System Responsiveness

Metric	Expected	Measured
First Contentful Paint	< 2.0s	≈1.7s
Largest Contentful Paint	< 2.5s	≈1.9s
Time to Interactive	< 3.0s	≈2.2s
Total Blocking Time	< 500ms	≈100ms

System Responsiveness Results

It seems that our application is quite responsive. We reached all of our response time goals and in fact even had quite a few milliseconds to spare.

The FCP took on average less than 1.7s to load, surpassing the expected metrics by 0.3s. Something to note about our LCP is that due to our application relying so little on images and videos this likely aided its response times by a large margin. This might be the reason why FCP and LCP are so close together in response times only differing by 0.2s.

TTI is 0.8s faster than expected. This is likely because our TBT was much lower than expected, leading to a faster TTI.

In conclusion, our application reached all system responsiveness goals we set out to achieve. As we have more time to refine our app, response times will likely improve as well.

Testing Results under Load

Number of Users	Average Response Time	Peak Response Time	Errors (%)
10	≈250ms	≈1800ms	0.0%
50	≈500ms	≈2000ms	0.3%
100	≈900ms	≈2100ms	1.3%
200	≈1700ms	≈2400ms	1.9%

Stress Testing Results

- The average response times for our application are what we deem acceptable, if not slightly better than what we had anticipated. We aimed for an average response time for queries to be less than 1s and for the most part succeeded.
- When the number of users exceeded 100 we failed to reach the goal of < 1.0s response time. With time to refine our application in the next sprint we should be able to match our goal response time even under more users.
- Peak response times are much higher than average response times. This is likely linked to our AI assistant, generating AI responses often takes a long time and this is likely why we see such a disconnect between average response time and peak response time.
- At max load (200 users) the website noticeably slowed down. Average response times increased drastically from 0.25s to 1.7s but peak response times remained relatively the same. This is likely due to time spent waiting for an AI query to return.