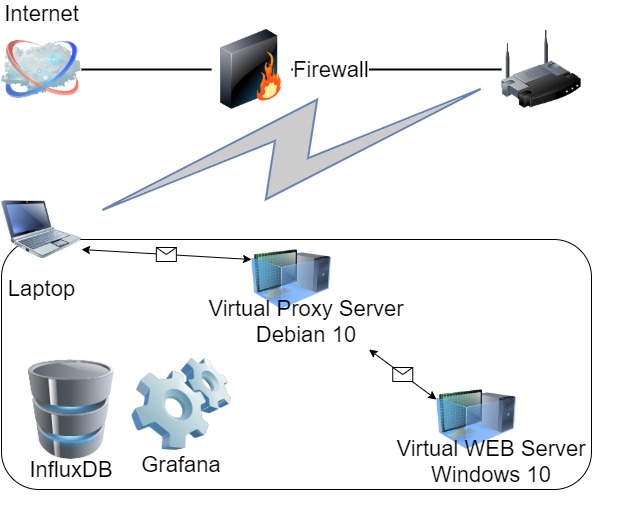
**First type (Server with data storage in file system):**

**Laptop** Windows 10**:**

Intel Core i7-8665U / 32 GB

**Virtual Proxy Server** Debian 10:

2 CPU / 4 GB

**Virtual WEB Server** Windows 10:

4 CPU / 4 GB (configurable)

1. **Capacity testing:**
   1. Load:
      1. Script: [HomeTask7.jmx](https://github.com/DmytroYaroslavtsev/study/blob/master/HomeTask11/HomeTask7.jmx)
      2. Users: Editor – 2, Admin – 1, Anonymous – 200
      3. Rump-up: Editor – 10s, Admin – 5s, Anonymous – 600s
      4. Number of posts: 1000
   2. Graphs:<https://github.com/DmytroYaroslavtsev/study/tree/master/HomeTask7/Capacity>
   3. KPI: Response time (avg, 95pct), throughput, active threads.
   4. Saturation Point: 35 users
2. **Scalability Testing:**
   1. Regular load:
      1. Script: [HomeTask7.jmx](https://github.com/DmytroYaroslavtsev/study/blob/master/HomeTask11/HomeTask7.jmx)
      2. Users: Editor – 2, Admin – 1, Anonymous – 35
      3. Rump-up: Editor – 120s, Admin – 60s, Anonymous – 1800s
      4. Test duration – 1 h
      5. Number of posts: 1000
   2. Variable values: Numbers of CPU (1, 2, 3, 4) with static RAM size (4 GB);

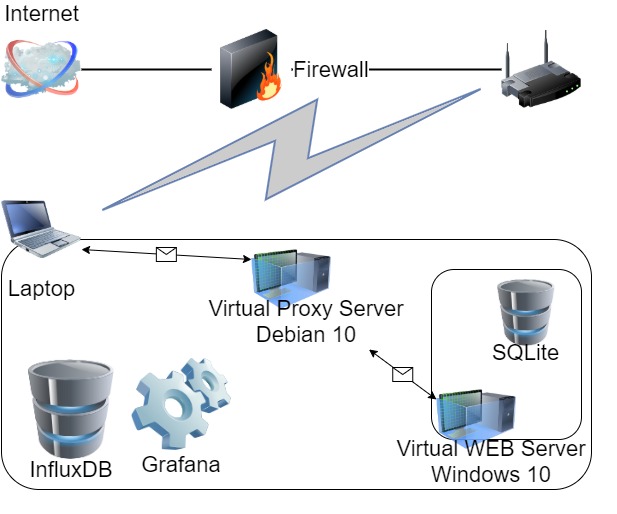
RAM size (2, 3, 4, 5 GB) with static numbers of CPU (4).

* 1. Graphs: <https://github.com/DmytroYaroslavtsev/study/tree/master/HomeTask8/Graphs>
  2. KPI: Get Home Page (First Page) samples results (avg, 90pct, 95pct), throughput, percentage of errors, active threads, CPU utilization, min available memory.
  3. Comparison:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | 4 CPU 2GB | 4 CPU 3GB | 4 CPU 5GB | 4 CPU 4GB | 3 CPU 4GB | 2 CPU 4GB | 1 CPU 4GB |
| Home avg | 1.05 s | 1.02 s | 743.6ms | 834.64ms | 785.09ms | 995ms | 1.78 s |
| Home 90 pct | 2.86 s | 4.70 s | 1.49 s | 3.35 s | 2.42 s | 2.36 s | 3.72 s |
| Home 95 pct | 8.83 s | 7.31 s | 3.33 s | 6.7 s | 5.01 s | 7.43 s | 9.35 s |
| Max throughput | 14.2 req/s | 13.6 req/s | 14.0 req/s | 13.2 req/s | 13.6 req/s | 12.6 req/s | 13.6 req/s |
| % of Errors | 0.03 % | 0 % | 0 % | 0 % | 0 % | 0 % | 0 % |
| Max CPU use | 72 % | 72 % | 62 % | 61 % | 84 % | 100 % | 100 % |
| Avg CPU use | 35 % | 30 % | 35 % | 35 % | 50 % | 70 % | 90 % |
| Min memory | 248.8 MB | 417 MB | 1.885 MB | 1.158 MB | 1.199 GB | 893 MB | 999 MB |

1. **Volume testing:**
   1. Regular load:
      1. Script: [HomeTask7.jmx](https://github.com/DmytroYaroslavtsev/study/blob/master/HomeTask11/HomeTask7.jmx)
      2. Users: Editor – 2, Admin – 1, Anonymous – 35
      3. Rump-up: Editor – 120s, Admin – 60s, Anonymous – 1800s
      4. Test duration – 1h
   2. Variable value: Number of posts (100, 1000, 2000, 5000, 1000 with 1 MB photo)
   3. Graphs: <https://github.com/DmytroYaroslavtsev/study/tree/master/HomeTask9/Graphs>
   4. KPI: Get Home Page (First Page) samples results (avg, 90pct, 95pct), throughput (max), percentage of errors, CPU utilization (max, avg), min available memory
   5. Comparison:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 100 | 1000 | 1000+photo | 2000 | 5000 |
| Home avg | 40.04 ms | 65.08 ms | 1.02 s | 171 ms | 834m |
| Home 90 pct | 35.80 ms | 199.9 ms | 4.7 s | 369.1 ms | 3.35 s |
| Home 95 pct | 223.25 ms | 289.9 ms | 7.31 s | 455.25 ms | 6.7 s |
| Max throughput | 15.4 req/s | 14.8 req/s | 14.2 req/s | 15.8 req/s | 13 req/s |
| % of Errors | 0.01 % | 0.15 % | 3.67 % | 3.6 % | 0.01 % |
| Max CPU use | 31 % | 51 % | 32 % | 100 % | 56 % |
| Avg CPU use | 12 % | 15 % | 10 % | 25 % | 30 % |
| Min memory | 981.3 MB | 889.5 MB | 622 MB | 996.3 MB | 667 MB |

**Second Type (Server with data storage in database):**

**Laptop** Windows 10**:**

Intel Core i7-8665U / 32 GB

**Virtual Proxy Server** Debian 10:

2 CPU / 4 GB

**Virtual WEB Server** Windows 10:

4 CPU / 4 GB (configurable)

**Server Database:** SQLite

1. **Capacity testing:**
   1. Load:
      1. Script: [HomeTask11.jmx](https://github.com/DmytroYaroslavtsev/study/blob/master/HomeTask11/HomeTask11.jmx)
      2. Users: Editor – 2, Admin – 1, Anonymous – 500
      3. Rump-up: Editor – 120s, Admin – 60s, Anonymous – 90000s
      4. Test duration – 25 h
      5. Number of posts: 1000
   2. Graphs: <https://github.com/DmytroYaroslavtsev/study/tree/master/HomeTask11/Graphs/Capacity>
   3. KPI: Response time (avg, 95pct), throughput, active threads.
   4. Saturation Point: 235 users
2. **Load testing:**
   1. Regular load:
      1. Script: [HomeTask11.jmx](https://github.com/DmytroYaroslavtsev/study/blob/master/HomeTask11/HomeTask11.jmx)
      2. Users: Editor – 2, Admin – 1, Anonymous – 188
      3. Rump-up: Editor – 120s, Admin – 60s, Anonymous – 1800s
      4. Test duration – 1 h
      5. Number of posts: 1000
   2. Graphs: <https://github.com/DmytroYaroslavtsev/study/tree/master/HomeTask11/Graphs/Load>
   3. KPI: Get Home Page (First Page) samples results (avg, 90pct, 95pct), throughput, percentage of errors, active threads, CPU utilization, min available memory.
3. **Stress testing:**
   1. Load:
      1. Script: [HomeTask11.jmx](https://github.com/DmytroYaroslavtsev/study/blob/master/HomeTask11/HomeTask11.jmx)
      2. Users: Editor – 2, Admin – 1, Anonymous – 240
      3. Rump-up: Editor – 120s, Admin – 60s, Anonymous – 1800s
      4. Test duration – 1 h
      5. Number of posts: 1000
   2. Graphs: <https://github.com/DmytroYaroslavtsev/study/tree/master/HomeTask11/Graphs/Stress>
   3. Get Home Page (First Page) samples results (avg, 90pct, 95pct), throughput, percentage of errors, active threads, CPU utilization, min available memory.
4. **Volume testing:**
   1. Regular load:
      1. Script: [HomeTask11.jmx](https://github.com/DmytroYaroslavtsev/study/blob/master/HomeTask11/HomeTask11.jmx)
      2. Users: Editor – 2, Admin – 1, Anonymous – 188
      3. Rump-up: Editor – 120s, Admin – 60s, Anonymous – 1800s
      4. Test duration – 1h
   2. Variable value: Number of posts (100, 1000, 2000, 5000, 1000 with 1 MB photo)
   3. Graphs: <https://github.com/DmytroYaroslavtsev/study/tree/master/HomeTask11/Graphs/Volume>
   4. KPI: Get Home Page (First Page) samples results (avg, 90pct, 95pct), throughput (max, avg), percentage of errors, CPU utilization (max, avg), min available memory
   5. Comparison:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 100 | 1000 | 1000+photo | 2000 | 5000 |
| Home avg | 93.5 ms | 294.98 ms | 2.85 s | 6.36 s | 23.41 s |
| Home 90 pct | 301.1 ms | 1.03 s | 5.73 s | 11.97 s | 32.61 s |
| Home 95 pct | 413.2 ms | 1.13 s | 6.46 s | 13.38 s | 35.00 s |
| Max throughput | 64.2 req/s | 62.0 req/s | 58.2 req/s | 58.6 req/s | 38.8 req/s |
| Avg throughput | 37.13 req/s | 37.02 req/s | 36.61 req/s | 29.27 req/s | 15.37 req/s |
| % of Errors | 0.02 % | 0.02 % | 0.02 % | 0 % | 0 % |
| Max CPU use | 71 % | 59 % | 71 % | 76 % | 62 % |
| Min memory | 684.3 MB | 538 MB | 364 MB | 507 MB | 604 MB |

1. **Scalability testing:**
   1. Regular load:
      1. Script: [HomeTask11.jmx](https://github.com/DmytroYaroslavtsev/study/blob/master/HomeTask11/HomeTask11.jmx)
      2. Users: Editor – 2, Admin – 1, Anonymous – 188
      3. Rump-up: Editor – 120s, Admin – 60s, Anonymous – 1800s
      4. Test duration – 1h
   2. Variable values: Numbers of CPU (1, 2, 3, 4) with static RAM size (4 GB);

RAM size (2, 3, 4 GB) with static numbers of CPU (4).

* 1. Graphs: <https://github.com/DmytroYaroslavtsev/study/tree/master/HomeTask11/Graphs/Scalability>
  2. KPI: Get Home Page (First Page) samples results (avg, 90pct, 95pct), throughput (max, avg), percentage of errors, CPU utilization (max, avg), min available memory
  3. Comparison:

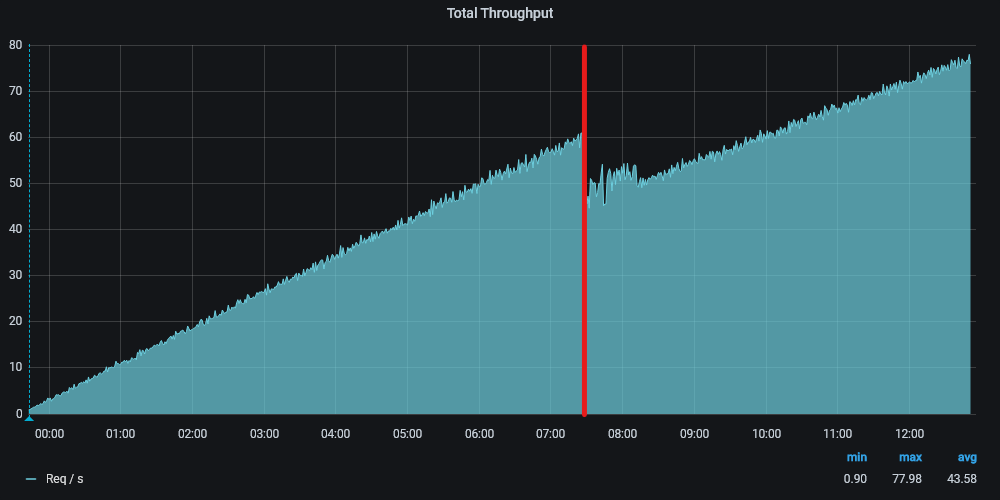
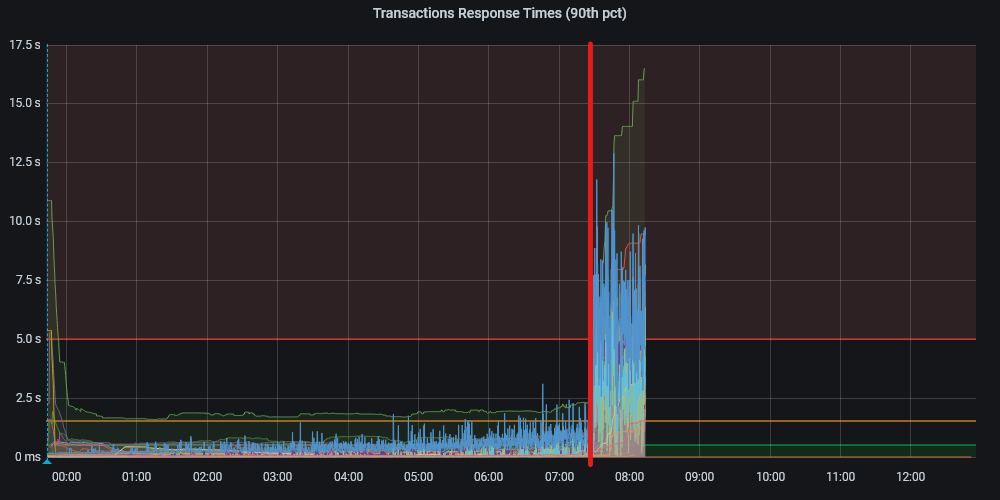
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | 4 CPU 2GB | 4 CPU 3GB | 4 CPU 4GB | 3 CPU 4GB | 2 CPU 4GB | 1 CPU 4GB |
| Home avg | 938.19 ms | 696.35 ms | 294.98 ms | 601.5 ms | 3.49 s | 2.06 s |
| Home 90 pct | 2.94 s | 1.44 s | 1.03 s | 1.84 s | 5.71 s | 3.69 s |
| Home 95 pct | 3.36 s | 1.75 s | 1.13 s | 2.03 s | 6.16 s | 5.06 s |
| Max throughput | 65.20 req/s | 66.8 req/s | 62.00 req/s | 58.2 req/s | 60.00 req/s | 60.40 req/s |
| Avg throughput | 36.61 req/s | 33.62 req/s | 37.02 req/s | 36.37 req/s | 33.87 req/s | 34.21 req/s |
| % of Errors | 0.07 % | 0.02 % | 0.02 % | 0.03 % | 0.01% | 0 % |
| Max CPU use | 61 % | 72 % | 59 % | 89 % | 100 % | 100 % |
| Avg CPU use | 26 % | 30 % | 22 % | 34 % | 54 % | 80 % |
| Min memory | 86 MB | 233 MB | 559 MB | 1.025 GB | 930 MB | 853 MB |

**Summary:**

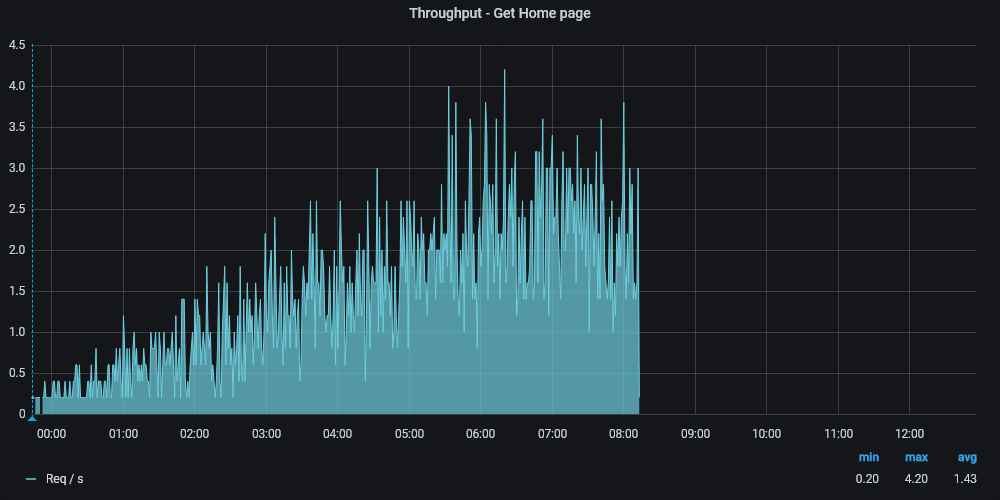
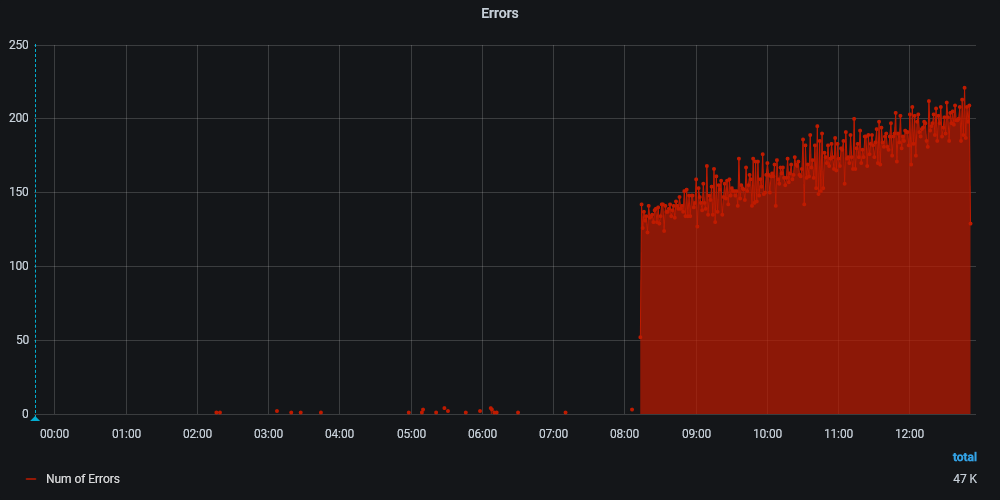
Test results of first type Environment (with data storage in file system) are too low, so there is no point to use such Environment and take this data into account.

**Second type Environment (with data storage in file data base) summary:**

According to Capacity testing saturation point has been determined in time when response time start to increase with high speed and throughput start to decrease.

After that time throughput start to decrease again but with error responses. We could see it from graphs with “Home Page” responses:

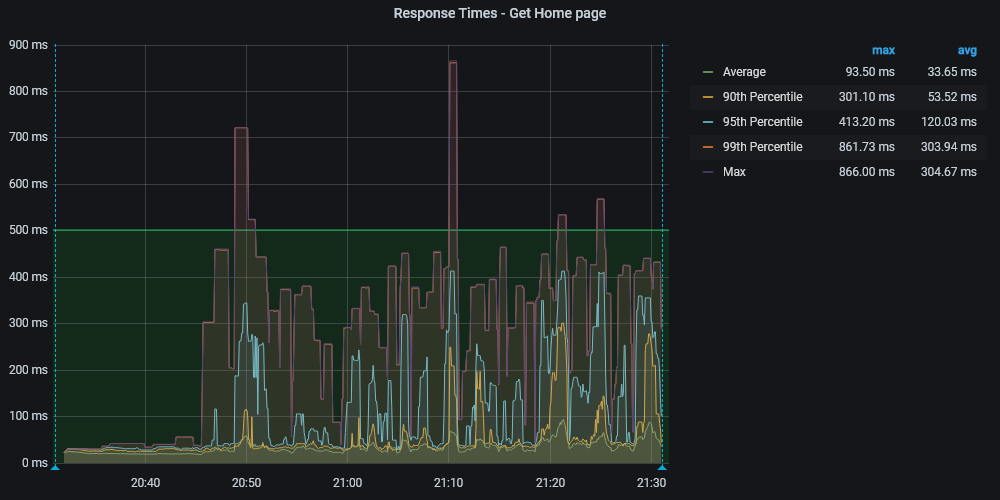
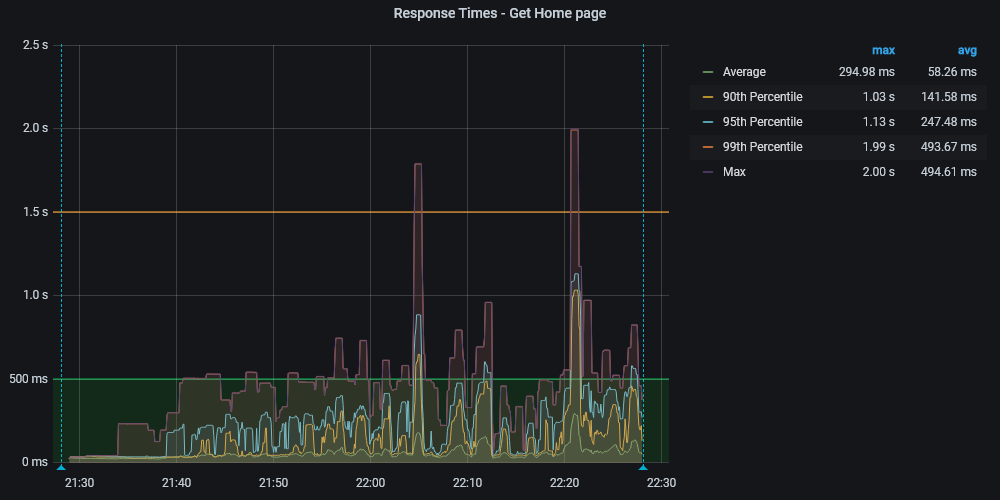
 

Using Capacity testing results was determined regular load (Numbers of users: Editor – 2, Admin – 1, Anonymous – 188; Rump-up time: Editor – 120s, Admin – 60s, Anonymous – 1800s; Test duration – 1h). And after was done load testing and stress testing.

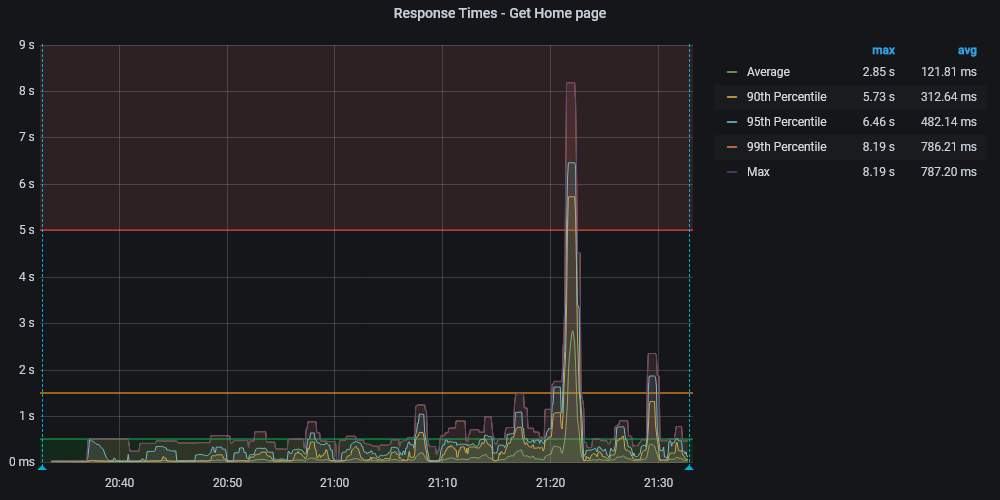
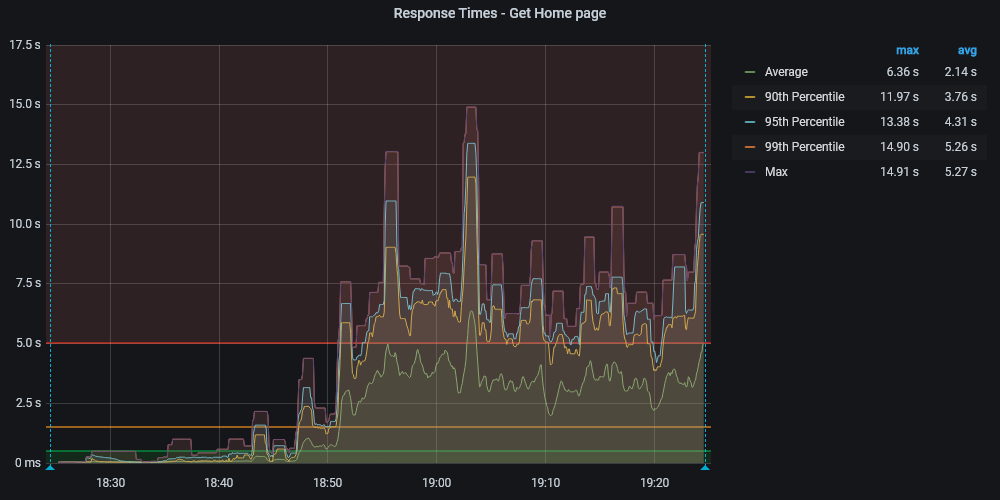
Load testing shows that system can handle regular load.

Stress testing shows that system can handle load near the saturation point.

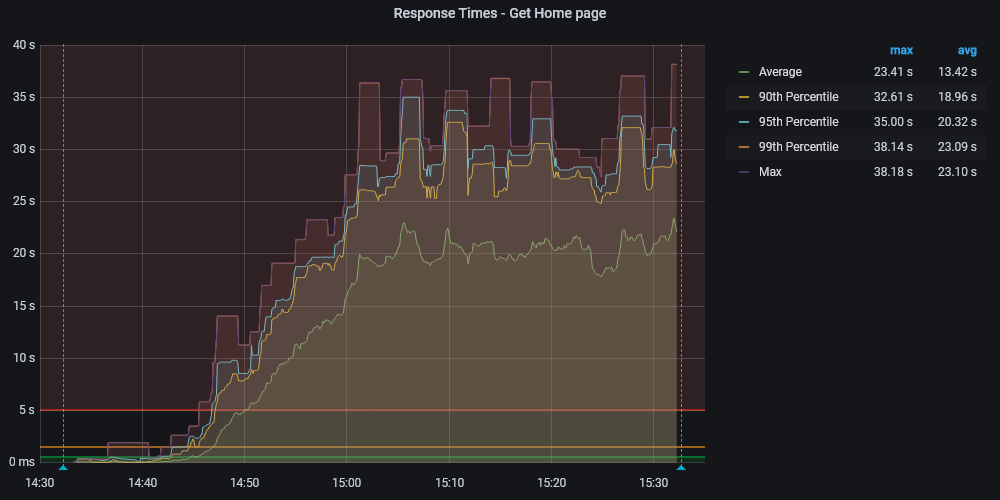
Responses time for (Home Page) was chosen as main KPI for Volume testing. Results of Volume testing show that system degradation depends on data size.

*100 posts 1000 posts*

*1000 posts with 1 Mb picture 2000 posts*



*5000 posts*

Scalability testing was done in all needed cases, but in different time and with different additional condition. So, results have a deviation. (For example: Test with 1 CPU 4 GB was done after some Environment optimization actions. And that actions increase performance of system). But even with results deviation Scalability testing shows dependency of System degradation from the number of CPUs and the size of the RAM of the server.

The comparison between the results with files system data source and DB data source are unavailable because of different load model (regular load).

At all stages of testing was a bottleneck: network configuration (proxy server).

**Conclusions:**

1. Use DB as a data source to increase the performance of the system.
2. Blog .Net server should be configured .Net specialist with optimal settings (For example: use garbage collector in server mode, not in workstation mode; settings internal processes which could load CPU for 100%, like antiviruses, software protectors, etc.)
3. For increasing throughput should be eliminated all network bottlenecks.