05-07-2022 Task 12

Test Purpose

* Prepare a complex report on the entire performance testing process.

Test plans

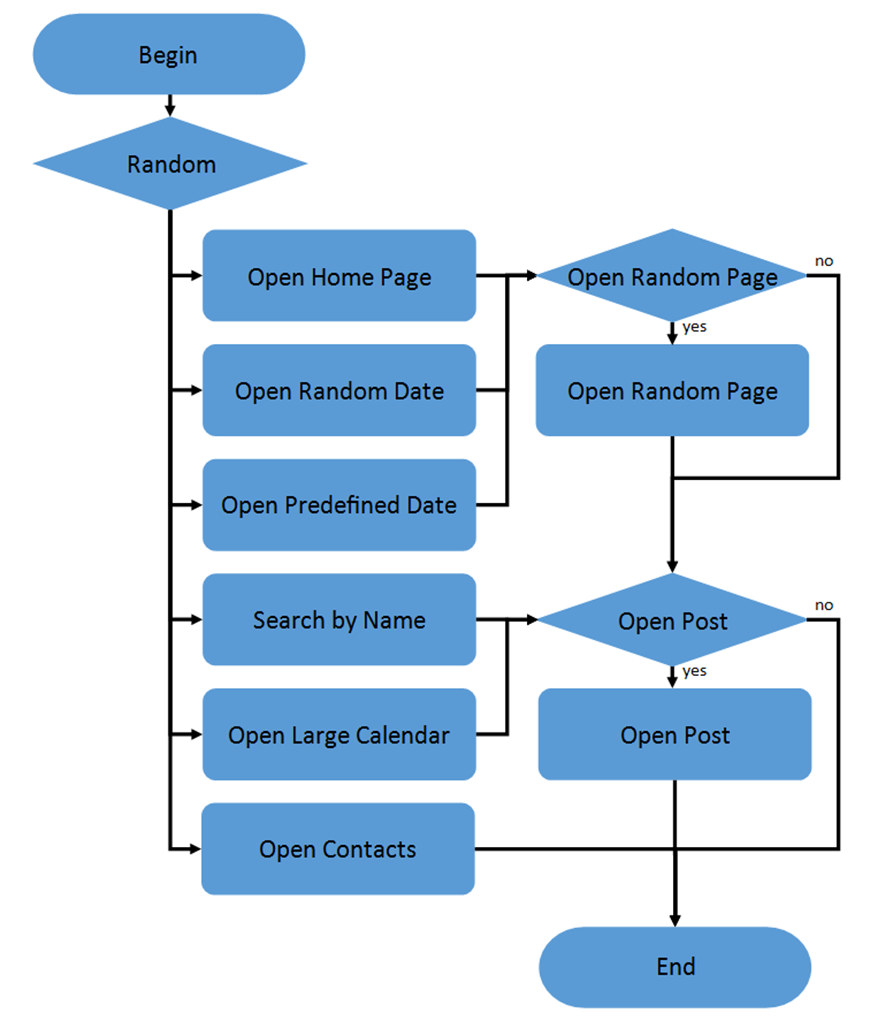
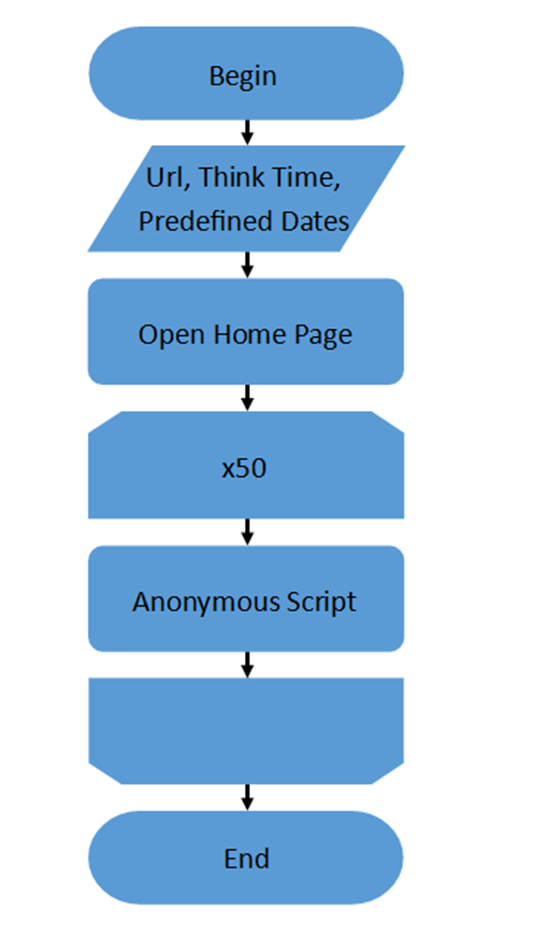
| **Start date:** 05.05.2022  **End date:** 05.07.2022 | **Introduction:** BlogEngine.NET 3.2 | |
| --- | --- | --- |
| **In Scope**   * Smoke testing * Capacity testing * Load testing * Regression testing * Scalability testing * Volume testing * Stability testing * Stress testing * Using DB(MySQL) | **Entry Conditions**   * Installed virtual machine with Windows-10 * Installed IIS * Installed web-site BlogEngine.Net * Installed JMeter * Installed Grafana * Installed InfluxDB * Installed Telegraf * Installed MySQL | **People**  QA: Nehrii Stanislav |
| **Exit Conditions**   * DB MySQL was had duplicate data * IIS Manager was downed time to time | **Test Environment**  Load testing tool: Apache JMeter 5.4.3  Load generator: Oracle VirtualBox 6.1, IIS Manager v10.0.19041.1  Software/hardware configuration: I7 3.0 GHz, 32 GB, Disk 96.3 GB, Windows 10 64bit  DB: MySQL 8.0.29 |
| **Out of Scope** | **Risks**   * unavailability/absence of test environments * database setup | **Timescales**  1. Planning and estimating - 1 day  2. Вevelopment of test scripts - 5 weeks  3. Running Tests - 5 weeks  4. Reporting - 2 weeks |

Test strategy

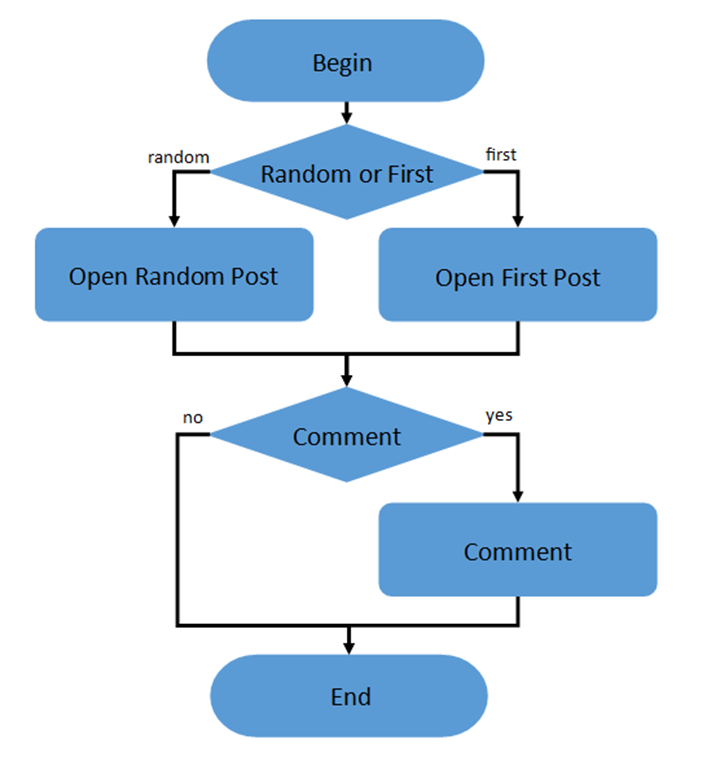
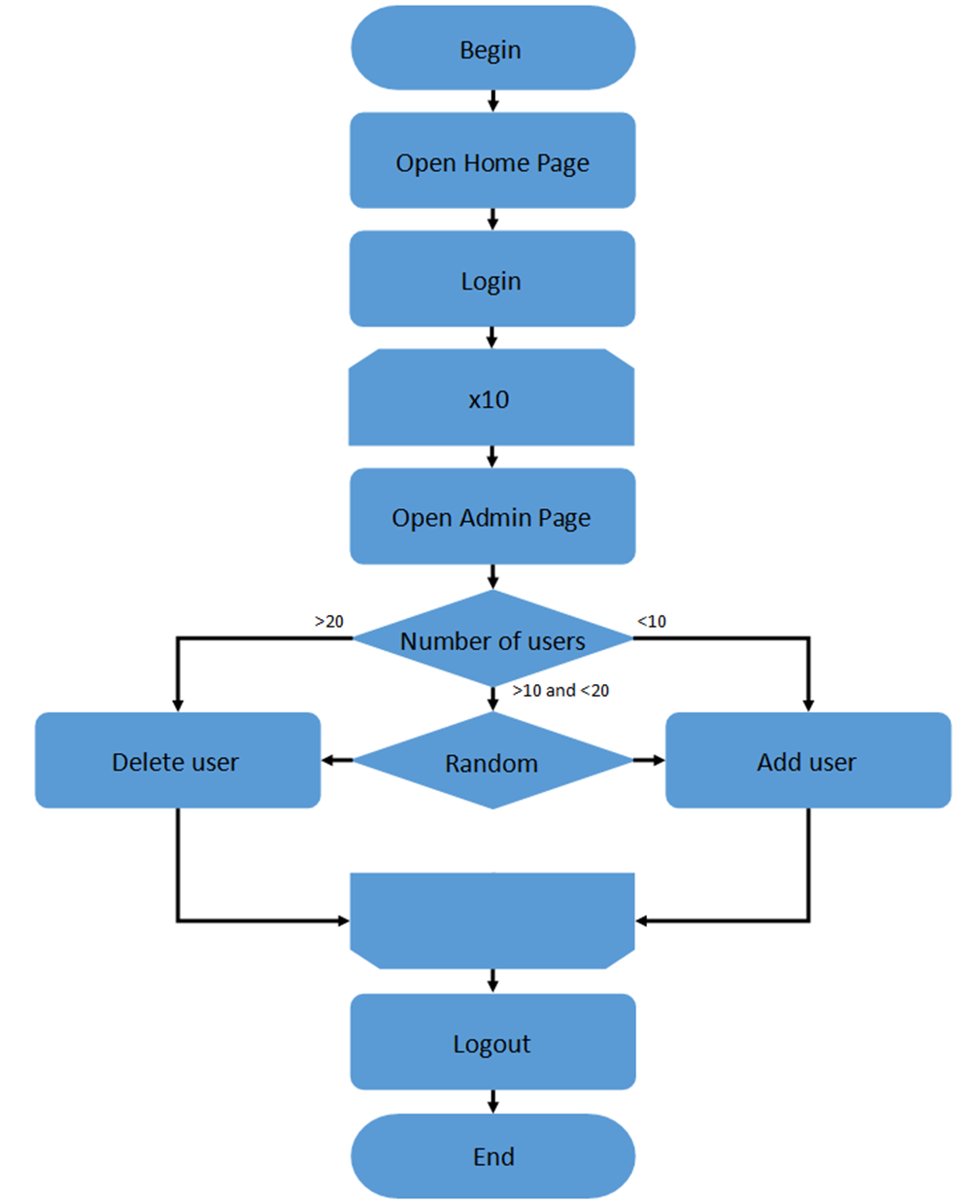
|  | **Smoke  testing** | **Capacity testing** | **Load testing** | **Regression testing** | **Scalability testing** | **Volume testing** | **Stability testing** | **Stress testing** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Anonymous script** | [HW-3](https://github.com/Son1cPower/Performance-Testing-Global-Mentoring-Program-2022-Q1Q2/tree/HW_1_2_3_4_5) [HW-6](https://github.com/Son1cPower/Performance-Testing-Global-Mentoring-Program-2022-Q1Q2/tree/HW_6) [HW-7](https://github.com/Son1cPower/Performance-Testing-Global-Mentoring-Program-2022-Q1Q2/tree/HW_7) | [HW-3](https://github.com/Son1cPower/Performance-Testing-Global-Mentoring-Program-2022-Q1Q2/tree/HW_1_2_3_4_5) [HW-6](https://github.com/Son1cPower/Performance-Testing-Global-Mentoring-Program-2022-Q1Q2/tree/HW_6) [HW-7](https://github.com/Son1cPower/Performance-Testing-Global-Mentoring-Program-2022-Q1Q2/tree/HW_7) [HW-11](https://github.com/Son1cPower/Performance-Testing-Global-Mentoring-Program-2022-Q1Q2/tree/HW_11) | [HW-8](https://github.com/Son1cPower/Performance-Testing-Global-Mentoring-Program-2022-Q1Q2/tree/HW_8) [HW-11](https://github.com/Son1cPower/Performance-Testing-Global-Mentoring-Program-2022-Q1Q2/tree/HW_11) | [HW-6](https://github.com/Son1cPower/Performance-Testing-Global-Mentoring-Program-2022-Q1Q2/tree/HW_6) | [HW-8](https://github.com/Son1cPower/Performance-Testing-Global-Mentoring-Program-2022-Q1Q2/tree/HW_8) | [HW-6](https://github.com/Son1cPower/Performance-Testing-Global-Mentoring-Program-2022-Q1Q2/tree/HW_6) [HW-9](https://github.com/Son1cPower/Performance-Testing-Global-Mentoring-Program-2022-Q1Q2/tree/HW_9) [HW-11](https://github.com/Son1cPower/Performance-Testing-Global-Mentoring-Program-2022-Q1Q2/tree/HW_11) | [HW-10](https://github.com/Son1cPower/Performance-Testing-Global-Mentoring-Program-2022-Q1Q2/tree/HW_10) | [HW-11](https://github.com/Son1cPower/Performance-Testing-Global-Mentoring-Program-2022-Q1Q2/tree/HW_11) |
| **Admin script** | [HW-4](https://github.com/Son1cPower/Performance-Testing-Global-Mentoring-Program-2022-Q1Q2/tree/HW_1_2_3_4_5) [HW-7](https://github.com/Son1cPower/Performance-Testing-Global-Mentoring-Program-2022-Q1Q2/tree/HW_7) | [HW-7](https://github.com/Son1cPower/Performance-Testing-Global-Mentoring-Program-2022-Q1Q2/tree/HW_7) [HW-11](https://github.com/Son1cPower/Performance-Testing-Global-Mentoring-Program-2022-Q1Q2/tree/HW_11) | [HW-4](https://github.com/Son1cPower/Performance-Testing-Global-Mentoring-Program-2022-Q1Q2/tree/HW_1_2_3_4_5) [HW-8](https://github.com/Son1cPower/Performance-Testing-Global-Mentoring-Program-2022-Q1Q2/tree/HW_8) [HW-11](https://github.com/Son1cPower/Performance-Testing-Global-Mentoring-Program-2022-Q1Q2/tree/HW_11) |  | [HW-8](https://github.com/Son1cPower/Performance-Testing-Global-Mentoring-Program-2022-Q1Q2/tree/HW_8) | [HW-9](https://github.com/Son1cPower/Performance-Testing-Global-Mentoring-Program-2022-Q1Q2/tree/HW_9) [HW-11](https://github.com/Son1cPower/Performance-Testing-Global-Mentoring-Program-2022-Q1Q2/tree/HW_11) | [HW-10](https://github.com/Son1cPower/Performance-Testing-Global-Mentoring-Program-2022-Q1Q2/tree/HW_10) | [HW-11](https://github.com/Son1cPower/Performance-Testing-Global-Mentoring-Program-2022-Q1Q2/tree/HW_11) |
| **Editor script** | [HW-5](https://github.com/Son1cPower/Performance-Testing-Global-Mentoring-Program-2022-Q1Q2/tree/HW_1_2_3_4_5) [HW-7](https://github.com/Son1cPower/Performance-Testing-Global-Mentoring-Program-2022-Q1Q2/tree/HW_7) | [HW-7](https://github.com/Son1cPower/Performance-Testing-Global-Mentoring-Program-2022-Q1Q2/tree/HW_7) [HW-11](https://github.com/Son1cPower/Performance-Testing-Global-Mentoring-Program-2022-Q1Q2/tree/HW_11) | [HW-5](https://github.com/Son1cPower/Performance-Testing-Global-Mentoring-Program-2022-Q1Q2/tree/HW_1_2_3_4_5) [HW-8](https://github.com/Son1cPower/Performance-Testing-Global-Mentoring-Program-2022-Q1Q2/tree/HW_8) [HW-11](https://github.com/Son1cPower/Performance-Testing-Global-Mentoring-Program-2022-Q1Q2/tree/HW_11) |  | [HW-8](https://github.com/Son1cPower/Performance-Testing-Global-Mentoring-Program-2022-Q1Q2/tree/HW_8) | [HW-9](https://github.com/Son1cPower/Performance-Testing-Global-Mentoring-Program-2022-Q1Q2/tree/HW_9) [HW-11](https://github.com/Son1cPower/Performance-Testing-Global-Mentoring-Program-2022-Q1Q2/tree/HW_11) | [HW-10](https://github.com/Son1cPower/Performance-Testing-Global-Mentoring-Program-2022-Q1Q2/tree/HW_10) | [HW-11](https://github.com/Son1cPower/Performance-Testing-Global-Mentoring-Program-2022-Q1Q2/tree/HW_11) |

Test cases for testing BlogEngine

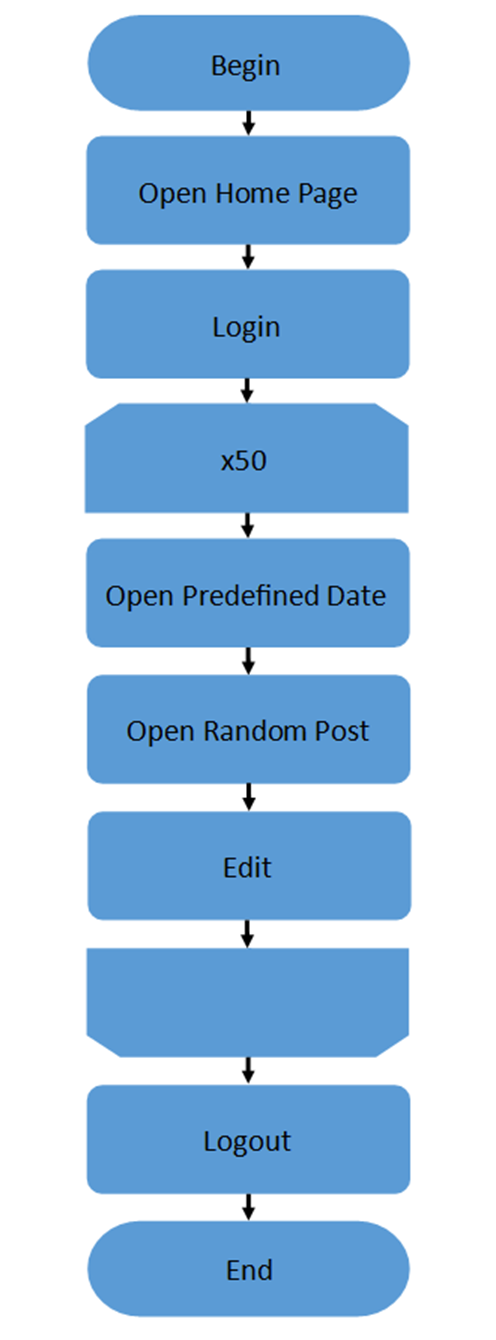
Main script Anonymous script



Open Post script Admin script

Editor script



Task 6

Test Purpose

* Perform Capacity testing for Task 3 with two sets of generated posts (100 and 1000).
* Perform Capacity testing for Task 6 with two sets of generated posts (100 and 1000).
* Compare results between:
  + Task 3 and Task 6: 100 posts
  + Task 3 and Task 6: 1000 posts
  + Task 6: 100 and 1000 posts

Test Summary

* We can observe a significant difference between those two tasks, Task 6 which implemented a distributed scenario shows us that under this scenario app able to handle much more users and use system resources more efficiently, this is possibly due to throughput which is less than in Task 6.
* Also we observe that the existing configuration can handle about 40 threads, when we have more than 40 threads we can observe that errors can start to grow and after 50-51 threads our application is down.

Compare results between: Task 3 and Task 6: 100 posts

|  | Task 3: 100 posts | Task 6: 100 posts |
| --- | --- | --- |
| Active threads (Mean) | 25 (Max:51) | 29 (Max:57) |
| TPS max | 17 | 24 |

Compare results between: Task 3 and Task 6: 1000 posts

|  | Task 3: 1000 posts | Task 6: 1000 posts |
| --- | --- | --- |
| Active threads (Mean) | 22 (Max:50) | 27 (Max:50) |
| TPS max | 11.20 | 18.2 |

Compare results between: Task 6: 100 and 1000 posts

|  | Task 6: 100 posts | Task 6: 1000 posts |
| --- | --- | --- |
| Active threads (Mean) | 29(Max:57) | 27(Max:50) |
| TPS max | 24 | 18.2 |

Task 8

Test Purpose

* Perform load testing for different numbers of CPUs: 1, 2, 4.
* Perform load testing for different sizes of RAM: 2Gb, 5Gb, 8Gb.

Test Summary

We can observe a significant difference between different sizes of RAM and different numbers of CPUs.

For cases when we added cores, it shows us that under this scenario Response time is decreased.

For cases when we added memory, it shows us that under this scenario Throughput is increased.

Compare results between: Cores: 1, 2, 4(Memory: 24GB)

|  | Cores: 1 | Cores: 2 | Cores: 4 |
| --- | --- | --- | --- |
| CPU Usage | 51,9% | 20.8% | 10,4% |
| Memory Usage | 709 mb | 712 mb | 717 mb |
| Response time  median | 401ms | 382ms | 336ms |
| Response time  90 pct | 7.66s | 1.24s | 7.05s |
| Throughput | 4.91 | 4.39 | 4.59 |
| Error | 2% | 8% | 2% |

Compare results between: Memory: 2Gb, 5Gb, 8Gb (Cores: 2)

|  | Memory: 2Gb | Memory: 5Gb | Memory: 8Gb |
| --- | --- | --- | --- |
| Memory Usage | 443 mb | 354 mb | 382 mb |
| CPU Usage | 9.96% | 20.5% | 9.89% |
| Response time  median | 310ms | 430ms | 289ms |
| Response time  90 pct | 4.33s | 1.42s | 1.16s |
| Throughput | 4.50 | 6.45 | 7.40 |
| Error | 2% | 5% | 2% |

Task 9

Test Purpose

* Perform load testing for 2000 pure text posts.
* Perform load testing for 2000 mixed posts (text and attached 1Mb photo).
* Compare results between the results of the testing
  + 2000 pure text posts and 2000 mixed posts

Test Summary

* We can observe a difference between those two test.Test with 2000 mixed posts (text and attached 1Mb photo) shows us that under this scenario used much more system resources, like CPU, also it has more response time and it has more failed transactions.

Compare results between: 100, 1000, 2000 and 5000 posts

|  | **100 posts** | **1000 posts** | **2000 posts** | **5000 posts** |
| --- | --- | --- | --- | --- |
| Active threads |  | Max: 71 | Max: 63 | Max: 80 |
| Throughput | 7.2 (Max: 14.6) | 7.95 (Max: 17.8) | 10.6 (Max: 27.8) | 6.95 (Max: 16.4) |
| avg Response time | 309 ms | 966 ms | 821 ms | 2.86 s |
| Failed transaction% | 4% | 9% | 36% | 29% |
| avg CPU | 17.2% | 18.8% | 50.1% | 64% |
| avg Memory | 497 mb | 389 mb | 424 mb | 420 mb |
| Response time  median | 119ms | 1.58s | 845ms | 11.75s |
| Response time  90 pct | 38s | 15.94s | 7.68s | 26.31 |

Compare results between: 2000 text posts and 2000 mixed posts

|  | **2000 mixed posts** | **2000 text posts** |
| --- | --- | --- |
| avg TPS | 3.8 (Max: 7.2) | 3.49 (Max: 6.4) |
| avg Response time | 584 ms | 416 ms |
| Transaction count | 4561 | 4207 |
| Failed transaction | 31 (1%) | 26 (1%) |
| avg CPU | 44.3% | 33.1% |
| avg Memory | 371 mb | 381 mb |
| Response time  median | 305ms | 212ms |
| Response time  90 pct | 1.15s | 1.14s |

Task 10

Test Purpose

* Perform long-time testing for 2200 text posts.

Test Summary

* We can observe that the application works stable for 2 hours. CPU had a few short-term leaps. Response time less than 1.5s for 95% requests. Failed transactions are less than 1%. The hardest request for our application is "Edit Post", it had 1% failed transactions. Also I ran 3 times this test, and two of them were failed without visible reasons for me, because application just stopped answered and CPU load was 100%, looks like problem with IIS.

Endurance testing : 2200 text posts

|  | **2200 text posts** |
| --- | --- |
| avg Throughput | 2 (Max: 3.6) |
| avg Response time | 287 ms |
| Transaction count | 14376 |
| Failed transaction | 37 (<1%) |
| avg CPU | 7.2% |
| avg Memory | 453 mb |
| Response time  median | 297 ms |
| Response time  90 pct | 942 ms |
| Response time  95 pct | 1.45 s |

Task 11

Test Purpose

* Perform load testing for 2200 text posts.
* Perform load testing for 2200 mixed posts.

Test Summary

* We can observe that response time is less and uses less CPU for tests with 2200 text posts.

Load testing : 2200 text posts and 2200 mixed posts

|  | **2200 text posts** | **2200 mixed posts** |
| --- | --- | --- |
| avg Throughput | 1.89 (Max: 3.8) | 2.77 (Max: 5.2) |
| avg Response time | 512 ms | 447 ms |
| Transaction count | 3782 | 5535 |
| Failed transaction | 15% | 5% |
| avg CPU | 23.2% | 26.3% |
| avg Memory | 431 mb | 433 mb |
| Response time  median | 159 ms | 306 ms |
| Response time  90 pct | 243.5 ms | 1.97 s |
| Response time  95 pct | 383.7 ms | 2.44 s |