**Report about conducted load test**

**Date:** 1st of June 2022

**Author:** Oleksandr Maksymenko

**Application:** BlogEngine.NET version 3.2

**Environment:** Host machine where script was run + Virtual Machine where blog is deployed

**Test Environment configuration(RAM, CPU etc.):**

|  |  |
| --- | --- |
| **RAM** | 4096 MB |
| **CPU** | 1 Core CPU (Intel(R) Core(TM) i7-8665U CPU @ 1.90GHz 2.11 GHz) |
| **System Type** | Windows 10 64-bit |

1. **Why such testing was conducted:** To check system’s behavior for usual Editor actions.
2. **Test script description:** Script contains actions for Editor flow. Firstly “user” opens Home page, Logs In, after that starts loop which executes 50 times: Open Predefined Date -> Open Random Page (in case editable post was not found on first page) -> Open Random Post -> Edit Post. After loop ends “user” will be logged off.

**Diagram

Description automatically generated**

1. **Tests:** 1 test runs.  
     
   **Test run preconditions:**

* 2 Editor users
* CSV file with user credentials
* Warmup actions before each test run (Log In, navigate through Blog pages, Open Content Page, Log Out)

**Load Model 1:**

|  |  |
| --- | --- |
| **Users** | 2 |
| **Rump up time (s)** | 1 |
| **Duration (s)** | 1500 |

**Load Model 2:**

|  |  |
| --- | --- |
| **Users** | 2 |
| **Rump up time (s)** | 1 |
| **Duration (s)** | 1200 |

1. **Short summary on conducted tests:**

* During test execution 1 we can observe that average response time was stable and not growing during run. “Spikes” happen when one active thread ends and new one starts (Log Out and Login actions)
* For 2nd test run response time is similar to 1st test, but it is lower at the beginning and is a bit higher at the end.
* Throughput was nearly the same (0,62 and 0,63 respectively) in both test executions.
* At the start of both tests, we can see Active Threads equal to 5, that number was used for Warmup actions, but when Main Script started number of AT in Jmeter and in Command Line was 2.
* Expected load was created for each of 2 users (1 opening of Main Page, 1 Log In, 50 Open Predefined Date, 50 Open Random Post, 50 Edit Post, 1 Log Off).
* No Server errors were observed during test run.

1. **Detailed test results #1:** Load Model 1

**A screenshot of a computer

Description automatically generated with medium confidence**

**Chart

Description automatically generated**

**A screenshot of a computer

Description automatically generated with low confidence**

Server’s metrics:

A screenshot of a video game

Description automatically generated

A screenshot of a computer

Description automatically generated with medium confidence

A screenshot of a video game

Description automatically generated

1. **Detailed test results #2:** Load Model 2

**A screenshot of a computer

Description automatically generated with medium confidence**

**Chart

Description automatically generated**

Server’s metrics:

Chart, histogram

Description automatically generated

A screenshot of a computer

Description automatically generated with medium confidence

A screenshot of a video game

Description automatically generated

1. **Conclusion:**

* System handles with 2 concurrent Editor users
* Response time was stable during run, some “spikes” can be observed at the same time when 1 Active Thread ends (Log Out), and new one starts (Login)
* Longest response time in 1st test was for Log Out (700-900 ms); for 2nd run it was Login (600-700 ms, when in 1st run it was ~400ms).
* Server’s metrics were stable except last 1 minute of 1st test run. At that moment CPU had its peak at 100% of load, Memory usage has grown up, Network was sending a lot of data to client.
* During 2nd test execution similar behavior was observed for CPU (reached ~76% load) and Memory, but Network stats were remaining the same. Insignificant growth of response time can be observed in our test when that spike happened.
* Further observation of CPU and Memory behavior showed such spikes happen approximately each 20-25 minutes even when tests are not run, which can say they are not related to application behavior.