**Instruction to execute load test**

1. Install apache jmeter 5.4.1

2. Download plugins manager.jar from [**https://jmeter-plugins.org/install/Install/**](https://jmeter-plugins.org/install/Install/)and keep it in lib->ext folder of jmeter home directory

3. Open jmeter using jmeter.bat in ‘Jmeterhome -> bin’

4. load the script file in jmeter and it will automatically direct you to install additional jars necessary for the test.

5. Once installation is done open **jmeter.properties** file in **apache-jmeter-5.4.1\bin** and modify the below lines and save

**jmeter.save.saveservice.output\_format=csv**

**httpsampler.ignore\_failed\_embedded\_resources=true**

**jmeter.save.saveservice.subresults=false**

6. To execute the load test from command line , navigate to jmeter bin directory from windows cmd and execute the below

jmeter.bat -n -t magnitude.jmx -Jusers=10 -Jrampup=10 -Jsteadystate=300 -Jemailprefix=mm29 -l SEP27\_10users\_run30.jtl -e -o SEP27\_10users\_run30

Juser -> user count

Jramup -> rampup time in sec

Jsteadystate -> duration in sec

Jemailprefix -> prefix for email id ( has to unique for everyrun)

Make sure the result file name is unique ( the names after -l and -o flags)

Two reports will be generated in the bin folder

Ex: SEP27\_10users\_run30.jtl , SEP27\_10users\_run30 ( folder in which index.html reports exists)

**Script and reports**

1. Scriptname - magnitude.jmx

2. Reports generated - SEP26\_10users\_run21\_xml.jtl

3. For reference purpose I have also attached HTML report for another run which excludes static assets ( SEP26\_10users\_run26.zip - extract and open index.html))

**Executive Summary**

This test was executed to determine if the system could maintain response times under the highest anticipated load.Below analysis is based on **SEP26\_10users\_run21\_xml.jtl** report file

**Inscope Urls**

GET <http://automationpractice.com/>

POST <http://automationpractice.com/index.php>

GET <http://automationpractice.com/index.php?controller=order>

GET <http://automationpractice.com/index.php?controller=order&step=1>

POST <http://automationpractice.com/index.php>

GET <http://automationpractice.com/index.php?controller=payment&module=bankwire&fc=module>

**User flow**

Guest user loads homepage -> Guest user adds 5 different products to card -> Guest registers -> User navigates through the complete checkout process

**SLA(Service Level Agreement)**

The below SLA’s were decided based on the business criticality of an E-commerce application

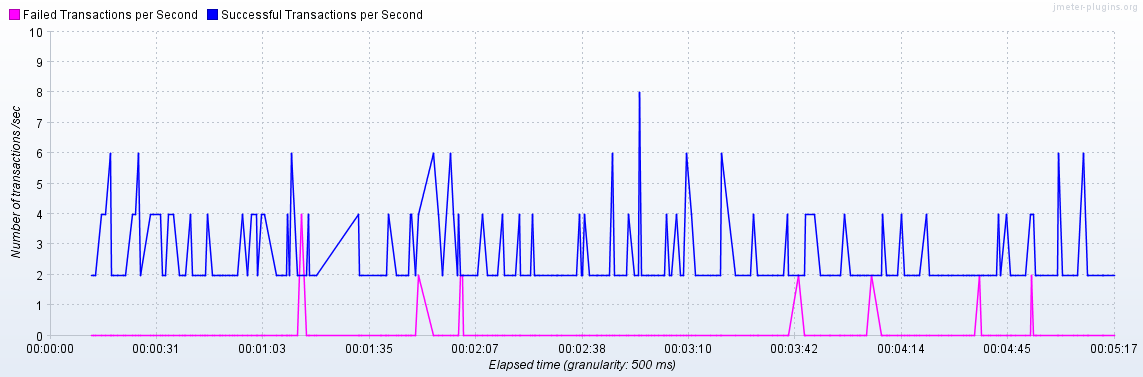
| **SLA ID** | **Description** |
| --- | --- |
| SLA-01 | Expected Average response time of all the transactions is 1 sec |
| SLA-02 | Maximum Average response time of all transactions should not exceed 2 sec |
| SLA-03 | Total Error percentage should not exceed 5% |
| SLA-04 | Error percentage of individual transaction should not exceed 5% |

**Performance Test Result Description**

* + 1. Summary

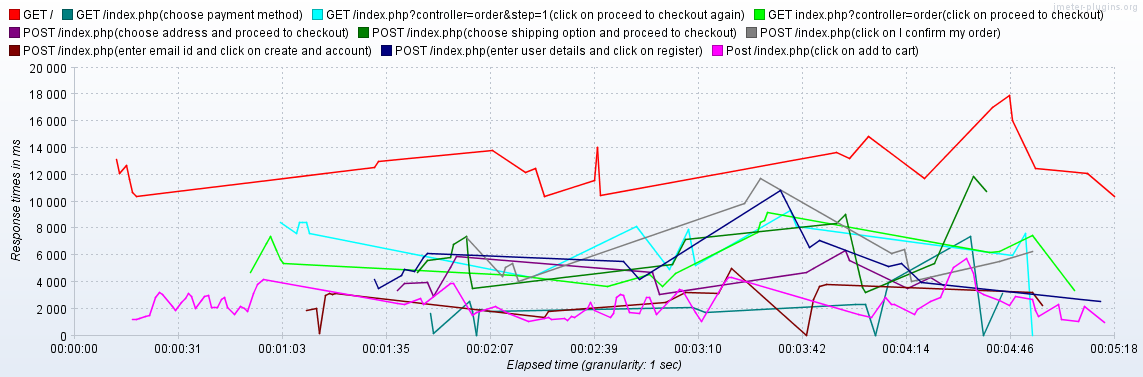
|  | **Test Details** |
| --- | --- |
| **Purpose** | This test was executed to determine if the system could maintain response times under the highest anticipated load. This test was designed to collect performance metrics on transaction throughput, response times. |
| **No. of Tests** | 1 |
| **Duration** | Ramp-up:10 sec  Steady State: 300 sec  Ramp-down: 10 sec |
| **User Load / Volume** | 10 Vusers |
| **Test Status** | Failed |
| **Observation / Reason for Failure** | 1. Transactions exhibited high response time which breached SLA01 and SLA02 2. Error percentage of few transactions has crossed 5% breaching SLA04 |
| **Overall RAG status** | RED |

**Transactions per second:**



The throughput of the system varied between **2- 6 TPS** during the test

**Response Times over time:**



The above graph indicates that the system started exhibiting higher response times from the beginning of the test and maintained the same behaviour throughout the test. This is an indication that the system is not set up/scaled to handle the mentioned tps and load.

**Average response times and error%:**

| Label | # Samples | AverageResponse time(ms) | Error % |
| --- | --- | --- | --- |
| GET / | 31 | 12759 | 0.00% |
| Post /index.php(click on add to cart) | 133 | 2469 | 0.00% |
| GET index.php?controller=order(click on proceed to checkout) | 24 | 6043 | 0.00% |
| GET /index.php?controller=order&step=1(click on proceed to checkout again) | 23 | 7141 | 4.35% |
| POST /index.php(enter email id and click on create and account) | 22 | 2534 | 13.64% |
| POST /index.php(enter user details and click on register) | 18 | 5602 | 0.00% |
| POST /index.php(choose address and proceed to checkout) | 17 | 4327 | 0.00% |
| POST /index.php(choose shipping option and proceed to checkout) | 17 | 6536 | 0.00% |
| GET /index.php(choose payment method) | 17 | 1954 | 29.41% |
| POST /index.php(click on I confirm my order) | 12 | 6377 | 0.00% |
| TOTAL | 314 | 4727 | 2.87% |

* Response times of all the transactions are very high breaching all the SLA’s set for response time **(SLA01 and SLA02)**
* Error % of GET /index.php(choose payment method) and POST /index.php(enter email id and click on create and account) has breached **SLA-04** (>5%)

**Transactions breaching SLA-04:**

| Label | # Samples | Error % | Error |
| --- | --- | --- | --- |
| GET /index.php(choose payment method) | 17 | 29.41% | Response code:Non HTTP response code: org.apache.http.NoHttpResponseException  Response message:Non HTTP response message: automationpractice.com:80 failed to respond |
| POST /index.php(enter email id and click on create and account) | 22 | 13.64% | Response code:Non HTTP response code: org.apache.http.NoHttpResponseException  Response message:Non HTTP response message: automationpractice.com:80 failed to respond |

High error percentage was seen only in pages related to payment and account creation.Code/ queries related to these endpoints has to be checked for any possible performance issues

**Recommendations:**

* Tuning needs to be performed at a system level rather than at individual page level
* Attention has be concentrated GET /index.php(choose payment method) and POST /index.php(enter email id and click on create and account)
* Post tuning another set of tests should be conducted to check the performance