TDDE46: Software Quality
Lab 6
ROBUSTENES & AVAILABILITY

Objectives:

1. The main objective of this lab to let student familiar/understand with the concept of load and stress testing
2. To perform stress testing in controlled environment
3. To perform load testing

Exercise 1:

1. What is the difference among the following. (4-5 lines)
    a. Load testing
    b. Stress testing
    c. Performance testing
2. Write about any two tools (each) for load testing and stress testing such as the names of tool and how can we use them (getting started) and what are the features.
3. In your opinion, which type of software, as a end user, you would prefer: Justify your selection with one paragraph
    a. available but crashes everyday
    b. Slow but never crashes

Exercise 2:

You will be conducting stress testing on Violet UML software. To start the Violet UML, open a terminal (UNIX shell and) execute the following command after downloading the jar file from course webpage:

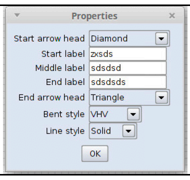
java -jar com.horstmann.violet-0.21.1.jar (also uploaded separately on course page along with manual on http://alexdp.free.fr/violetumleditor/page.php?id=en:installation)

Stress testing is conducted to see: “How much can we push things before the application fails?” and what we meant with “pushing things” is to expose the system under test to abnormal circumstances and high volumes with an intention to break the system under test. The aim is to crash the system under test and noticed whether it crashed Gracefully or abnormal. In graceful termination, we can recover our data but in abnormal termination, all data is lost. Stress testing is particularly important for Web applications since networks can easily get heavily loaded very quickly.

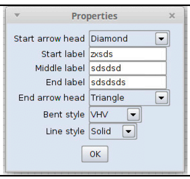
You have been provided with the test plan below to stress test the Violet UML software. Follow the plan and take readings, as mentioned below in tables. Spend 15 minutes on each test plan and if nothing happens, move to next plan. Keep reading/writing the required values, as per requested in below tables, during and after the execution of test plans. Use the following websites for animated applets:

- https://www.falstad.com/circuit/
- https://www.falstad.com/mathphysics.html

Test Plan 1:

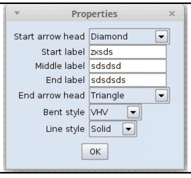
Open the different animated applet in different Firefox tabs	10 applets
Number of Violet instances	1
Keep creating inheritance relationship in the file in each instances of Violet. **The inheritance relationships must be drawn to the same superclass i.e the subclasses are all at the same level.**	until next 15 minutes
Double click on each relationship and fill information	 Properties Start arrow head: Diamond Start label: zxsds Middle label: sdsdsd End label: sdsdsds End arrow head: Triangle Bent style: VHV Line style: Solid OK
How many classes and inheritance relations, you made in these 15 minutes. This will help you noticing performance as you proceed with next test plan 1. We will see if it increases or decreases.	
Software crashed (Gracefully or abnormal):	
Your reflection	
Start and End Time	

Test Plan 2:

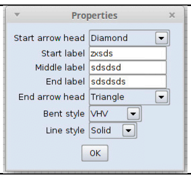
Open the different animated applet in different Firefox tabs	20 applets
Number of Violet instances	3
Keep creating inheritance relationship in the file in each instances of Violet. **The inheritance relationships must be drawn to the same superclass i.e the subclasses are all at the same level.**	until next 20 minutes
Double click on each relationship and fill information	 Properties Start arrow head: Diamond Start label: zxsds Middle label: sdsdsd End label: sdsdsds End arrow head: Triangle Bent style: VHV Line style: Solid OK
How many classes and inheritance relations, you made in these 15 minutes. This will help you noticing performance as you proceed with	

next test plan 1. We will see if it increases or decreases.	
Software crashed (Gracefully or abnormal):	
Your reflection	
Start and End Time	

Test Plan 3:

Open the different animated applet in different Firefox tabs	30 applets
Number of Violet instances	5
Keep creating inheritance relationship in the file in each instances of Violet. **The inheritance relationships must be drawn to the same superclass i.e the subclasses are all at the same level.**	until next 25 minutes
Double click on each relationship and fill information	 Properties Start arrow head Diamond Start label zxsds Middle label sdsdsd End label sdsdsds End arrow head Triangle Bent style VHV Line style Solid OK
How many classes and inheritance relations, you made in these 15 minutes. This will help you noticing performance as you proceed with next test plan 1. We will see if it increases or decreases.	
Software crashed (Gracefully or abnormal):	
Your reflection	
Start and End Time	

Test Plan 4:

Open the different animated applet in different Firefox tabs	40 applets
Number of Violet instances	6
Keep creating inheritance relationship in the file in each instances of Violet. **The inheritance relationships must be drawn to the same superclass i.e the subclasses are all at the same level.**	until next 30 minutes
Double click on each relationship and fill information	 Properties Start arrow head Diamond Start label zxsds Middle label sdsdsd End label sdsdsds End arrow head Triangle Bent style VHV Line style Solid OK
How many classes and inheritance relations, you made in these 15 minutes. This will help you noticing performance as you proceed with next test plan 1. We will see if it increases or decreases.	

Software crashed (Gracefully or abnormal):	
Your reflection	
Start and End Time	

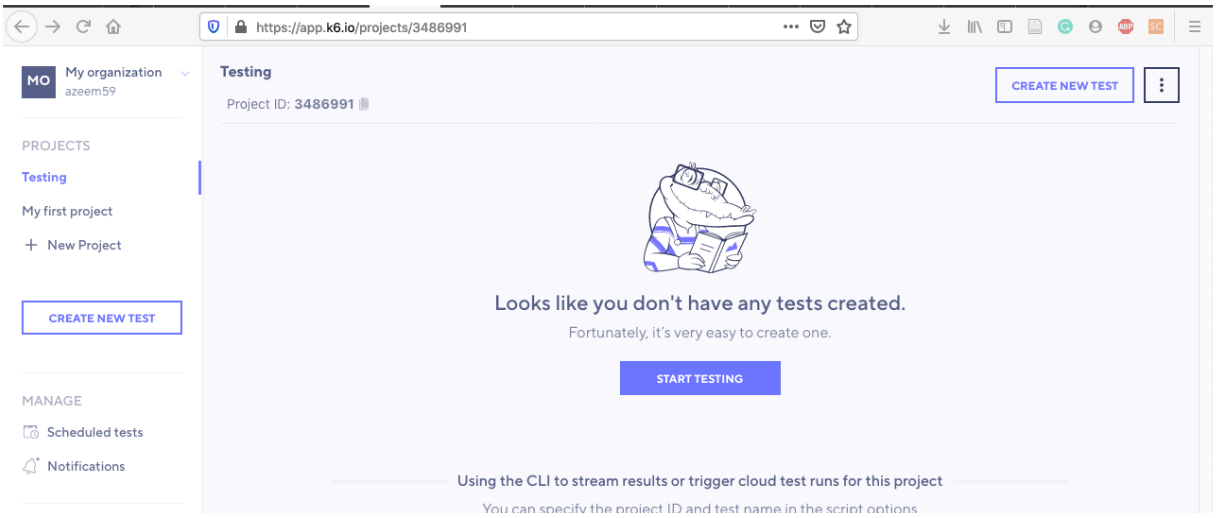
Exercise 3:

your job in this task is to perform load testing on the following two URLs:

- http://advantageonlineshopping.com/#/
- www.liu.se

We will use the online tool called https://k6.io/cloud. You need to create an account for free trial.

- Create a new project or perhaps, one project is already created for your with the name of "My First Project"



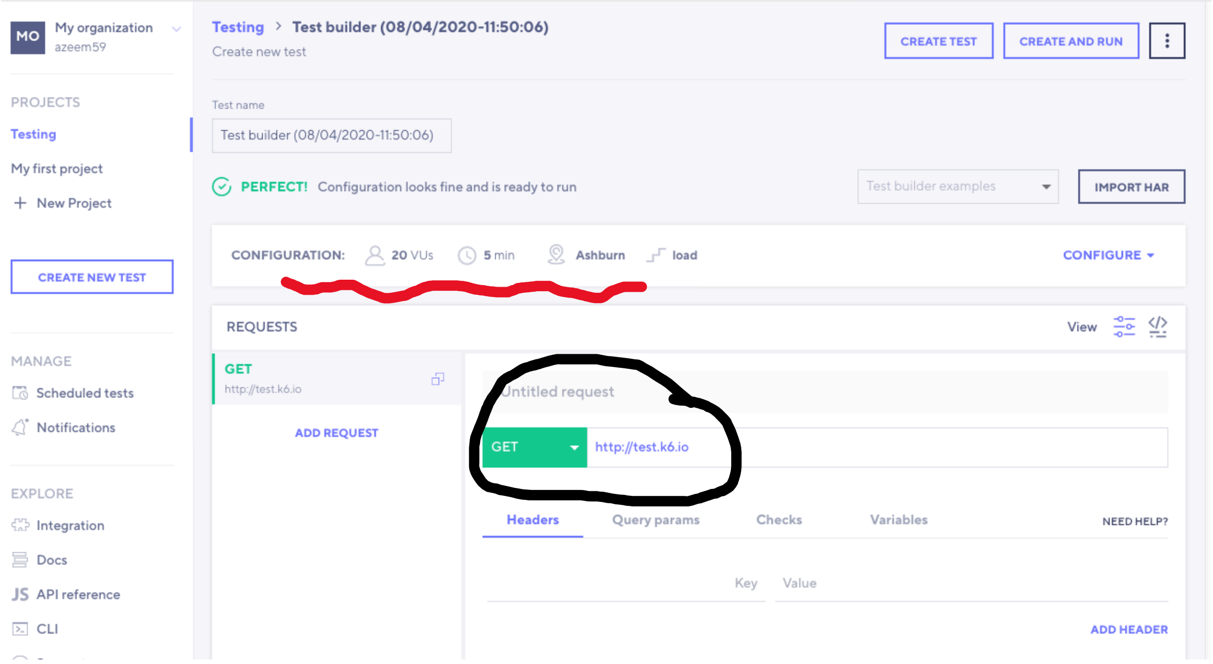
Looks like you don't have any tests created.

Fortunately, it's very easy to create one.

Using the CLI to stream results or trigger cloud test runs for this project

You can specify the project ID and test name in the script options

- Click on "Create New Test" and then "Start Building" under "Test Builder".
- Enter the URL as shown below in circle and then click on top right "Create And Run". You can see the configurations such as number of virtual users, how long test should run etc.



Testing > Test builder (08/04/2020-11:50:06)

Create new test

Test name: Test builder (08/04/2020-11:50:06)

PERFECT! Configuration looks fine and is ready to run

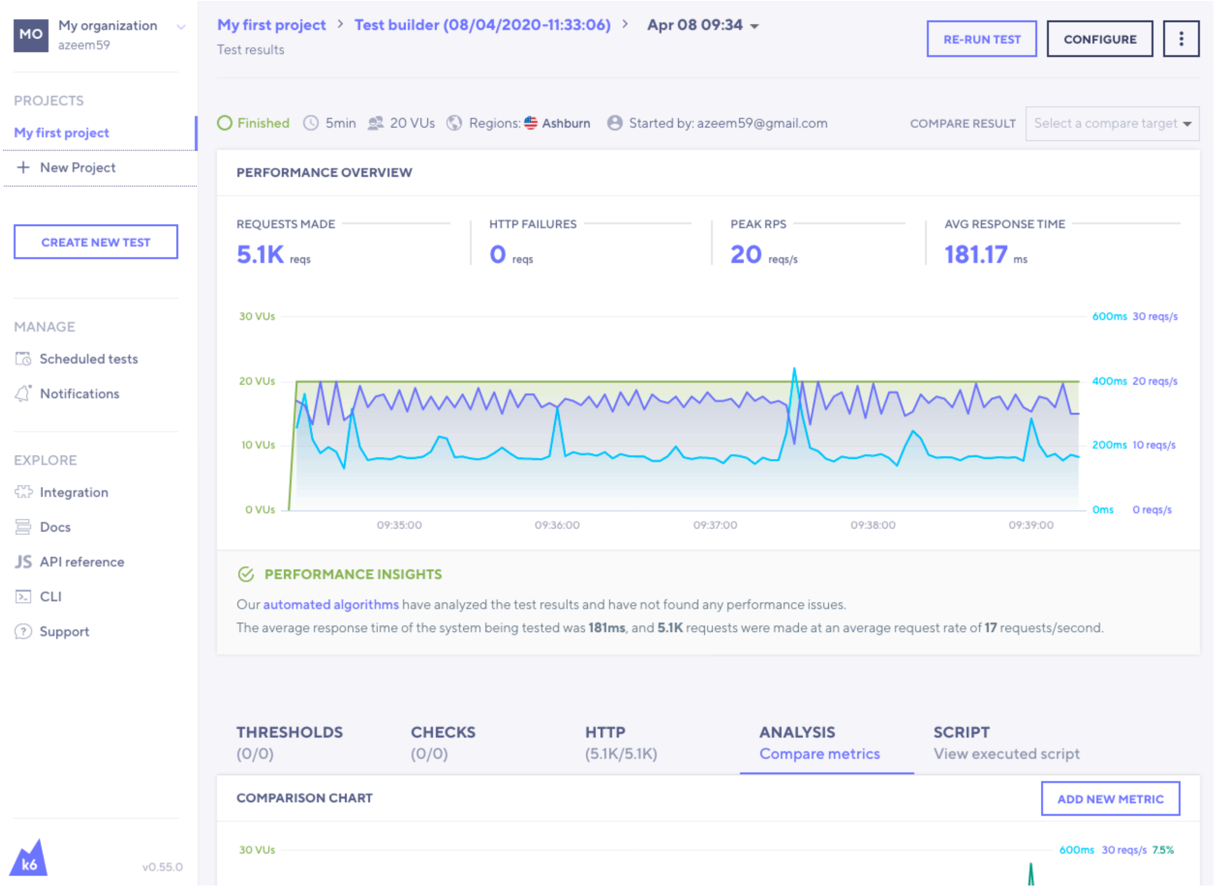
CONFIGURATION: 20 VUs 5 min Ashburn load

GET http://test.k6.io

Untitled request

GET http://test.k6.io

- You will notice that test suite has been started. After some time, it shows a clock for 5 minutes. Wait until the time has passed and you have seen some graphs.
- Once 5 minutes have been passed, you will see the reports as below:



My first project > Test builder (08/04/2020-11:33:06) > Apr 08 09:34

Test results

Finished 5min 20 VUs Regions: Ashburn Started by: azeem59@gmail.com

PERFORMANCE OVERVIEW

REQUESTS MADE	HTTP FAILURES	PEAK RPS	AVG RESPONSE TIME
5.1K reqs	0 reqs	20 reqs/s	181.17 ms

PERFORMANCE INSIGHTS

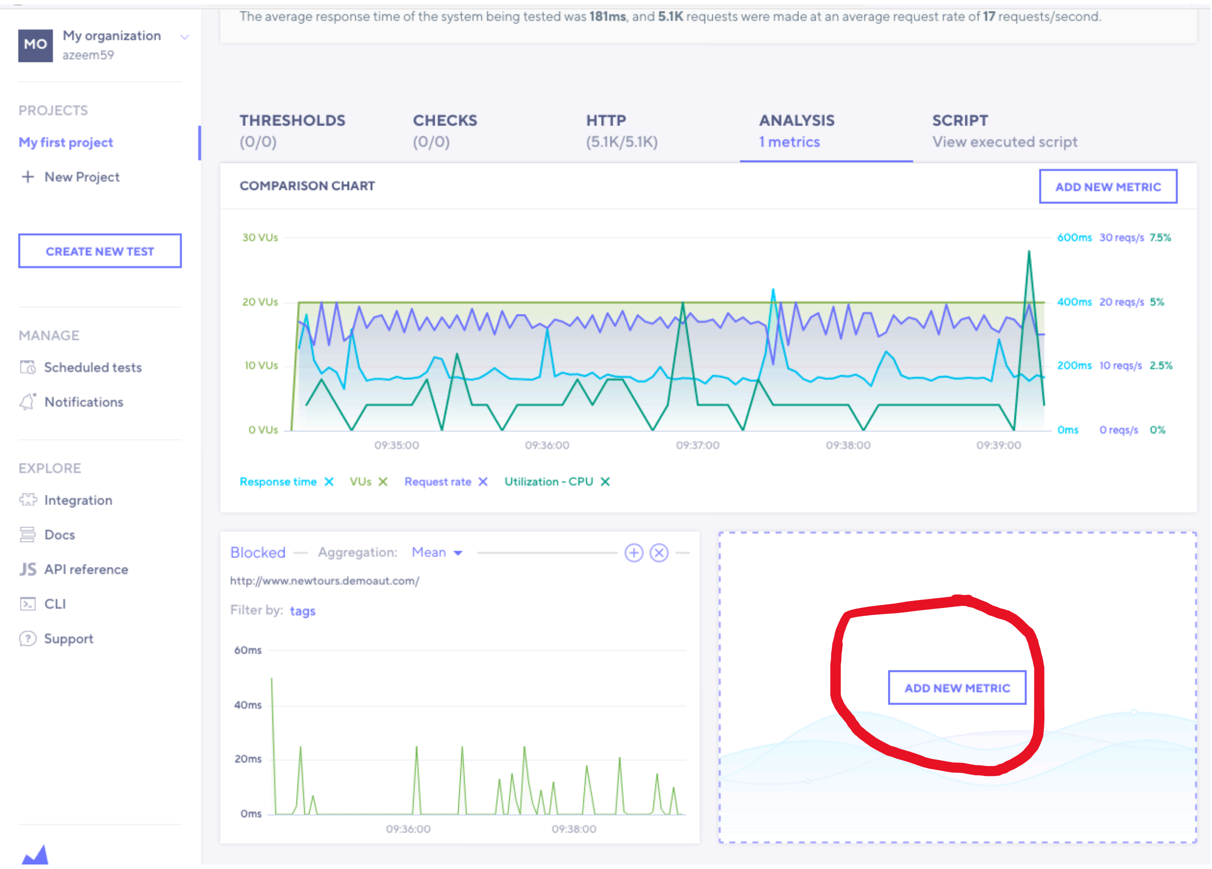
Our automated algorithms have analyzed the test results and have not found any performance issues.

The average response time of the system being tested was 181ms, and 5.1K requests were made at an average request rate of 17 requests/second.

THRESHOLDS	CHECKS	HTTP	ANALYSIS	SCRIPT
(0/0)	(0/0)	(5.1K/5.1K)	Compare metrics	View executed script

COMPARISON CHART

- You can add more metrics by the panel down the page as shown in below diagram. To do that, click on “Analysis – Compare Metrics”



The average response time of the system being tested was **181ms**, and **5.1K** requests were made at an average request rate of **17** requests/second.

1. There are 15-17 metrics provided to explore the efficiency of the webpage. Select any 8 metrics from the list. You are free to choose any metrics of your choice. Your job is to enter both of above two URLS separately (either in sequence or in two different browsers) and use the selected metrics:
2. Print all screen shots of metrics result in the file to be submitted on LISAM.
3. Explain what selected metrics does?
4. Explain why the results of selected metrics are important for you? What do they tell you and how can you use this information to increase quality?
5. Why selected metrics appears different for both URL as both belongs to LiU.
6. Write 4-5 lines, whether these types of testing should be automated or why not?
7. https://k6.io/cloud is a free website and provided limited features. Write one paragraph to convince your manager for why should he spend money to buy a commercial tool for load testing?