M7024E Laboratory 5: Option 1 - Cloud Benchmarking

In this laboratory exercise, you will explore tools to benchmark Cloud applications. As described in last week's lecture, cloud monitoring and benchmarking is crucial for application provisioning and Cloud capacity planning, efficient resource utilization and cost savings, and making sure that the applications deployed on the Clouds are market-ready.

1 Objectives

Welcome to the fifth and final laboratory of the M7024E course. The objective of this lab is to:

- Understand concepts regarding Cloud benchmarking.
- Using widely available tools for application performance benchmarking such as Httpperf¹ and Apache JMeter², benchmark an application.

2 Exercises

Exercise a: In this exercise, you will learn how to benchmark a simple Web application (it can be your RESTFul service). You will generate the syn- thetic workload (emulating a number of users making several requests per second, see lecture slides) and benchmark the performance of your application.

- 1. Download and install Apache JMeter on your lab machine/laptop/Cloud instance. Follow the instructions given on the "Getting Started" page on the Apache JMeter website or anywhere on the WWW. Stick to the GUI mode for running JMeter.
- 2. As a previous lab exercise, you created a Webpage and served it via an application server such as Apache Webserver. Test the performance of your Web service using JMeter. It is also OK to run your application and JMeter on the same machine, if you want.
 - (a) Build a "Test Plan" to emulate a number of users, say 10, 50, 100,..., or more. Follow chapter 4 of the "User's Manual" which explains how to test a Web service.

 $^{^1\}mathrm{httperf}$ Webpage:
 https://github.com/httperf/httperf [ONLINE]. Access date: 30th Nov. 2022.

 $^{^2\}mathrm{Apache\ JMeter:\ https://jmeter.apache.org/\ [ONLINE]}.$ Access date: 30th Nov. 2022.

- (b) Execute your "Test Plan" by varying the number of users and the "ramp-up" period.
- (c) Use the Table/Graph to analyze the results and document them.
 - i. Explain in detail your "Test plan". For example, explain (via table) the parameters that you used, their relevance and settings.
 - ii. Analyze and document the performance of your application w.r.t the load you generate via JMeter. For instance, what is the average number of requests/sec. that saturates the application server, the latency, the throughput and other relevant metrics that you find useful.