

**NDC Assignment 2**

**Derek McCarthy B00007439**

Table of Contents

[Introduction 3](#_Toc513369519)

[Configuration Research 4](#_Toc513369520)

[Tests Cases 5](#_Toc513369521)

[Results 6](#_Toc513369522)

[Test Case 1 Results 6](#_Toc513369523)

[Test Case 2 Results 7](#_Toc513369524)

[Test Case 3 Results 8](#_Toc513369525)

[Test Case 4 Results 9](#_Toc513369526)

[Test Case 5 Results 10](#_Toc513369527)

[Conclusion 12](#_Toc513369528)

[References 13](#_Toc513369529)

# Introduction

In this report I am going to discusses the impact of a user’s interaction with a Python guestbook hosted on the Google App Engine (GAE) servers. I will discuss the different configurations available on the GAE servers and the associated cost of the configuration and also any trade-off associated with that configuration. To gauge the impact of the different number of users I will use a number of different test case scenarios.

# Configuration Research

Google App Engine (GAE) is a framework which allows users to develop and host their applications on Googles cloud-based infrastructure. It allows you to run your application in two different environments Flexible and Standard. With the Standard environment applications are ran in a sandbox. It is mostly used when developers want to run their applications for “free or at very low cost”. With the Flexible environment applications hosted on GAE run within the Docker containers which scale automatically. Docker allows applications to be packaged into containers allowing them to run on any application that is using a Linux operating system. While most developers choose to use one environment for their applications they can also decide to use both environments simultaneously. The cost of each environment varies with the flexible being the most expensive.

**Standard Environment Cost**

With the standard environment you pay a flat rate for an instance class which is basically a computer system with fixed memory limit and CPU which you choose to suit your needs. The type of instance classes ranges from the F1 which is the default option which has a memory limit of 128 MB and a CPU limit of 600 MHz to the B8 option which has a memory limit of 1 GB and a CPU limit of 4.8 GHz. Below are all the available instance classes in which you can choose from.

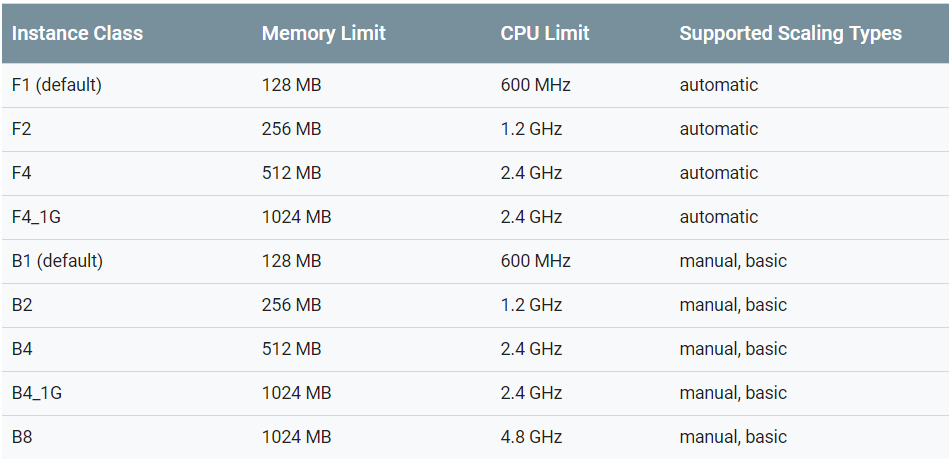


Figure 1 - Standard Environment Instance Class

**Flexible environment Cost**

Unlike the standard environment the flexible environment is set up on virtual machines where you choose the ram, permanent storage and CPU you desire. Each of these resources are charged on a per-second basis with the minimum amount to be purchased being 1 minute.

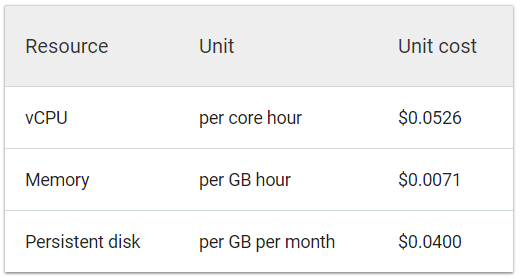


Figure - Flexible Environment Cost Chart

# Tests Cases

To test how Google App Engine handles requests from several users with different amounts of data parameters I will use several different test cases. To do this I will use Apache JMeter to simulate the users which is carried out by declaring several threads with each thread representing a user. The test cases and the parameters I will use are,

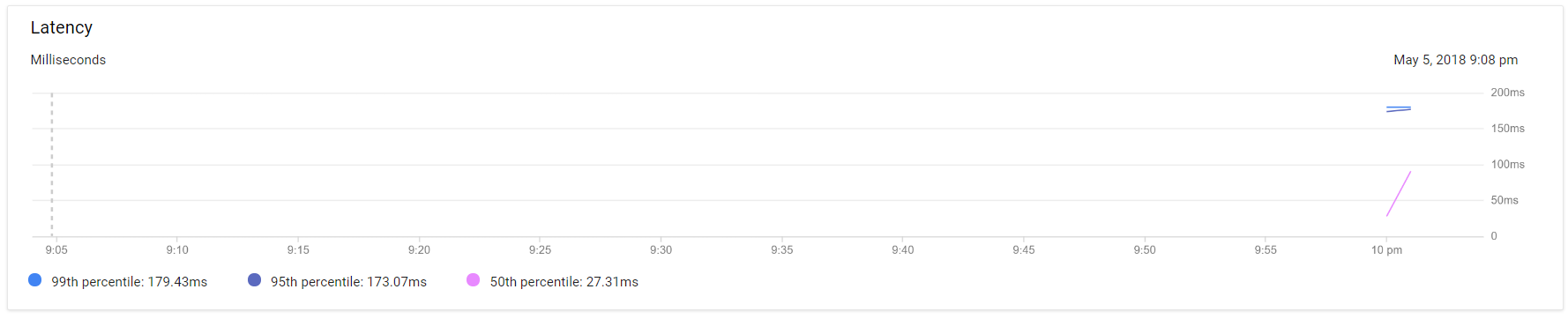
1. For the first test case I will use one user who will enter one word as the data parameter to the guestbook.
2. For the second test case I will use ten users who will enter 10 words each as the data parameter to the guestbook.
3. For the third test case I will use 100 users who will enter 1000 words each as the data parameter.
4. For the fourth test case I will use 250 users who will enter 2000 words each as the data parameter.
5. And finally, for the fifth test case I will use 500 users who will enter 4000 words each as the data parameter.

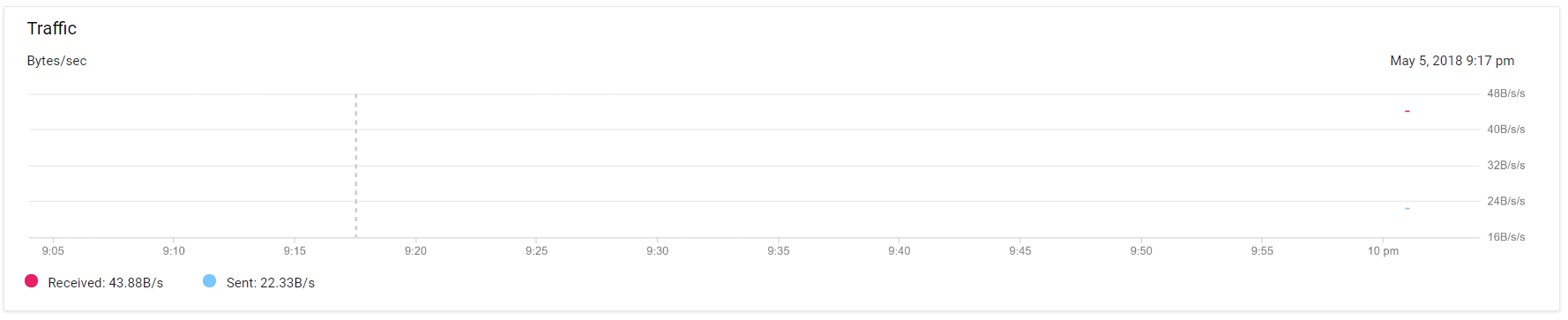
# Results

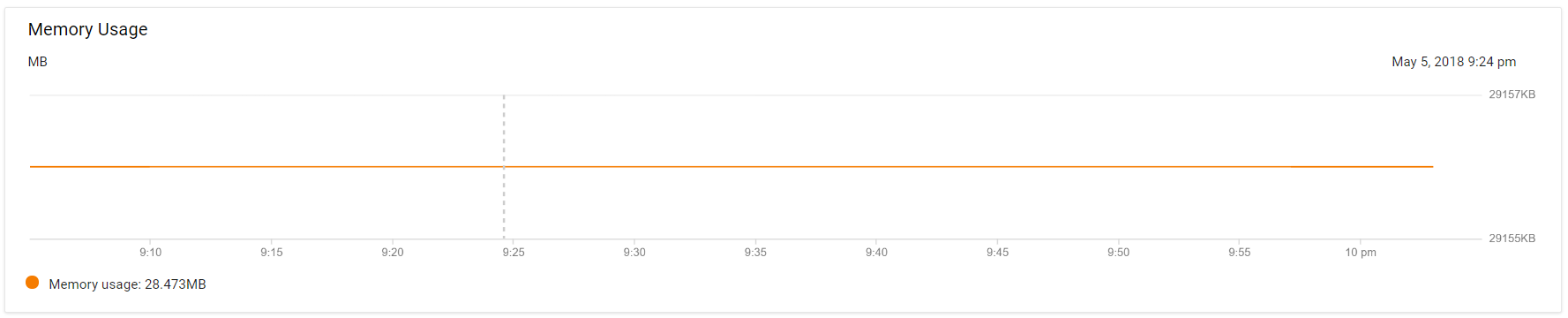
To analyse the impact the tests, have on the GAE server I will look at the impact of the Latency, Traffic and Memory usage of the server. I will include the screenshots of all 5 test and for the fifth test I will explain the results.

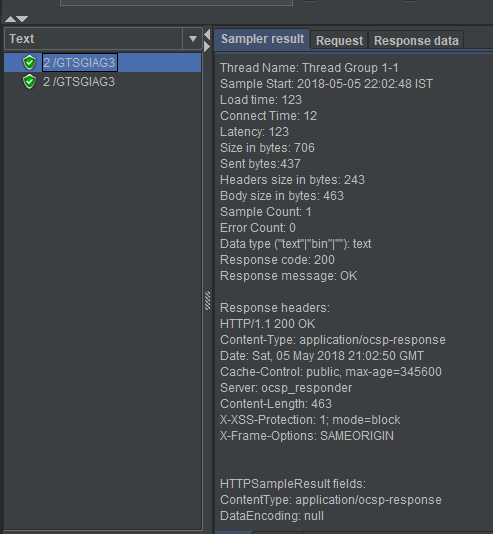
## Test Case 1 Results

In this test I tested the GAE server with 1 user who added 1 word to the guestbook. Below are the results from this test.



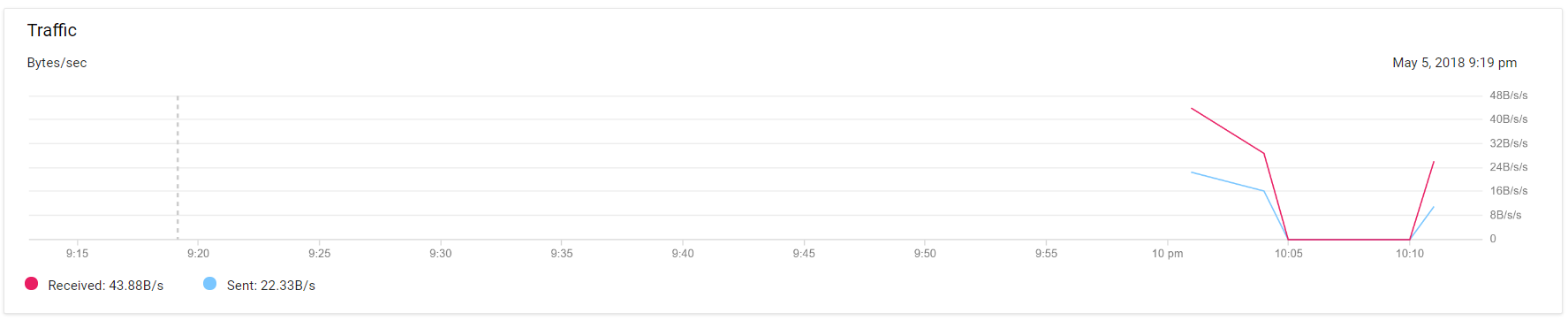


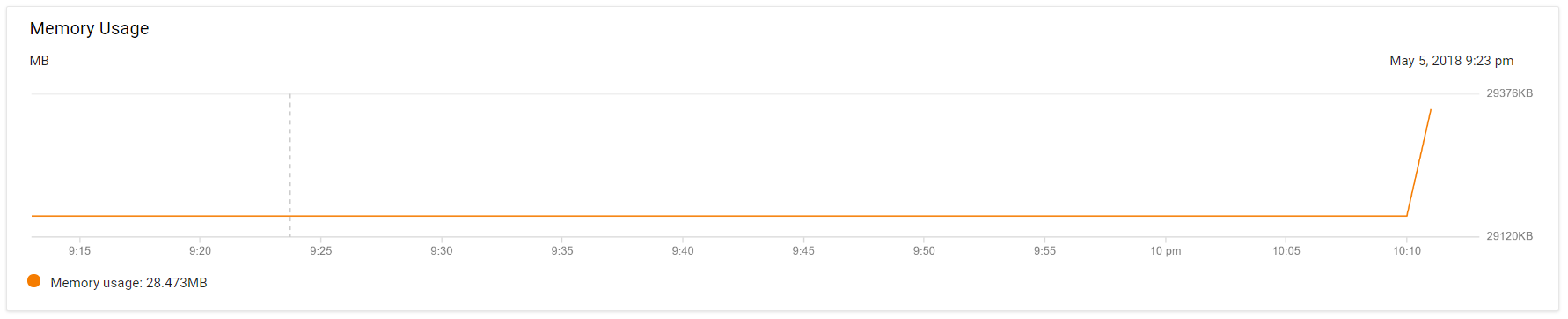


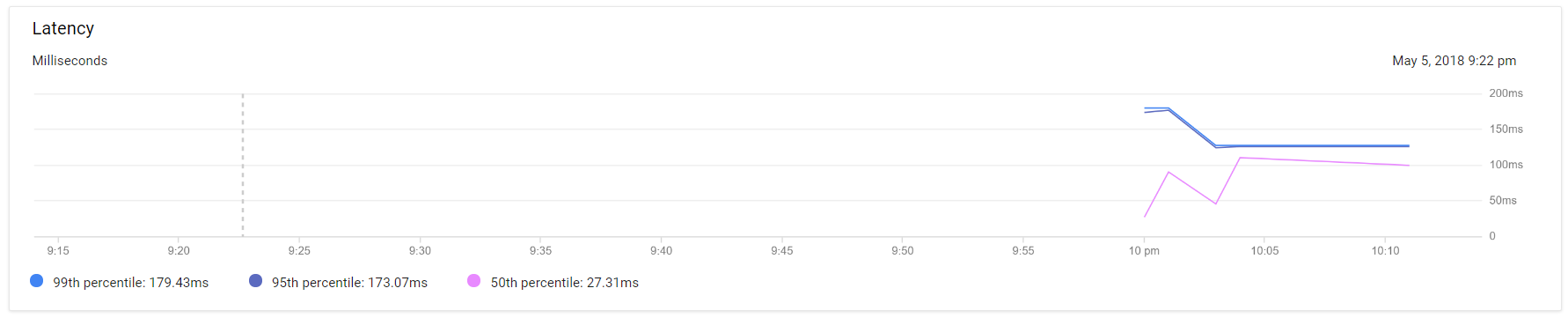


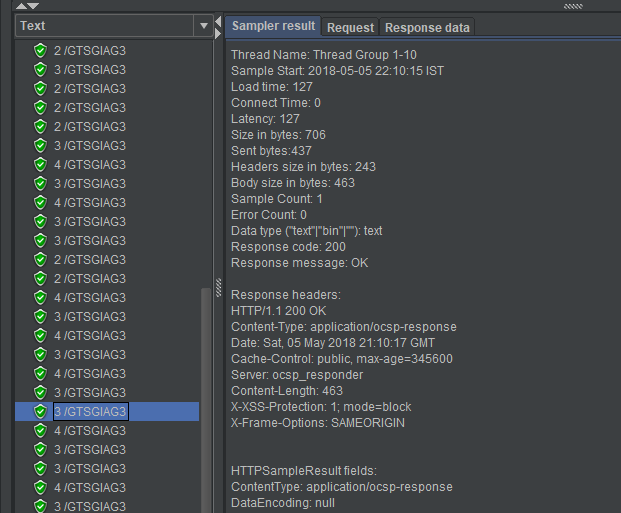
## Test Case 2 Results

In this test I tested the GAE server with 10 users who added 10 words to the guestbook. Below are the results from this test.



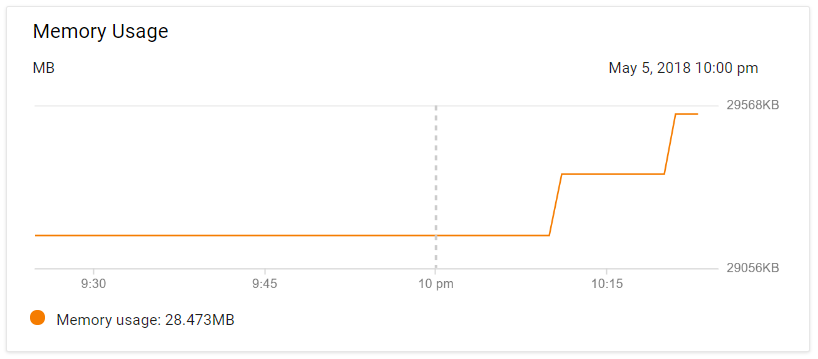


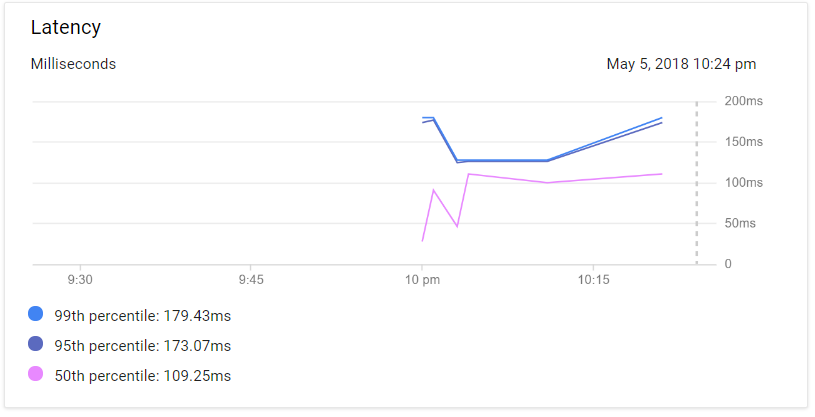


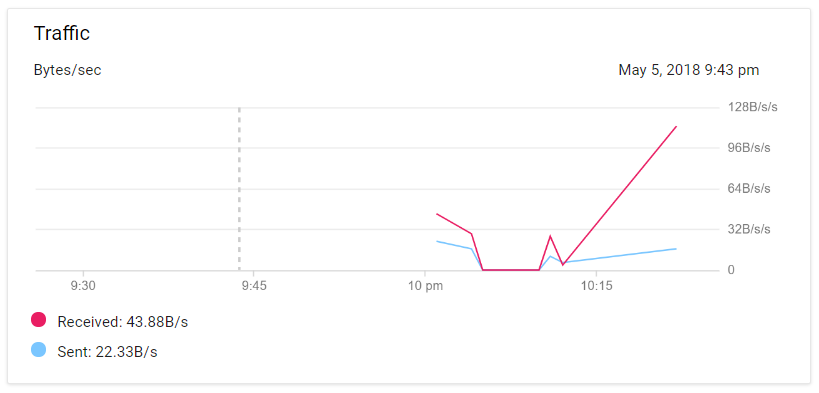


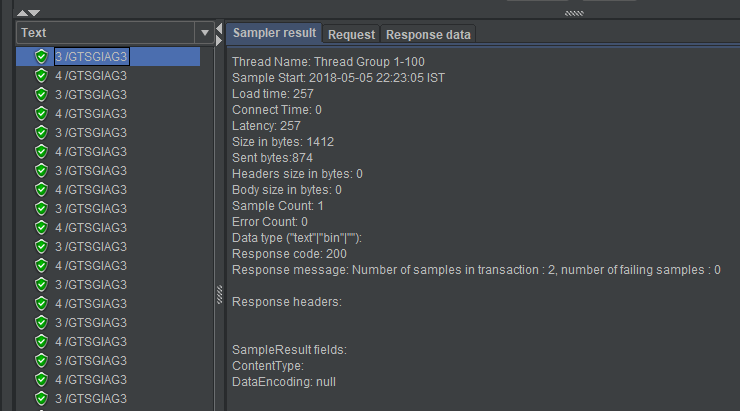
## Test Case 3 Results

In this test I tested the GAE server with 100 users who added 1000 words to the guestbook. Below are the results from this test.



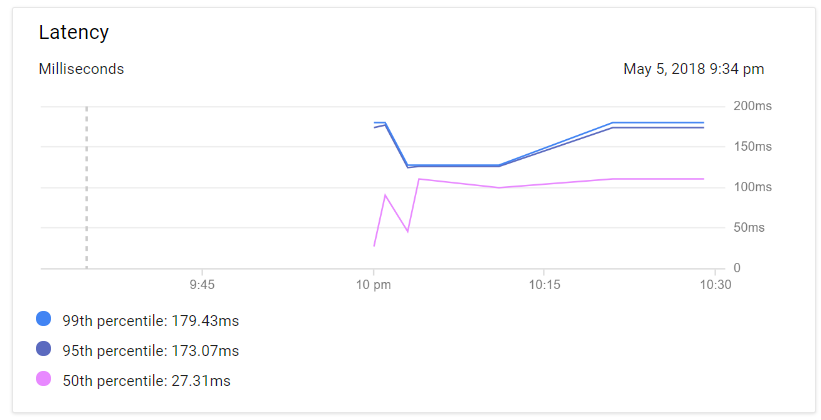


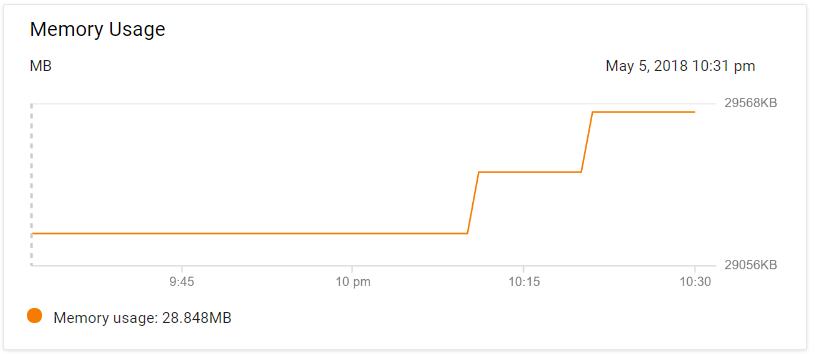


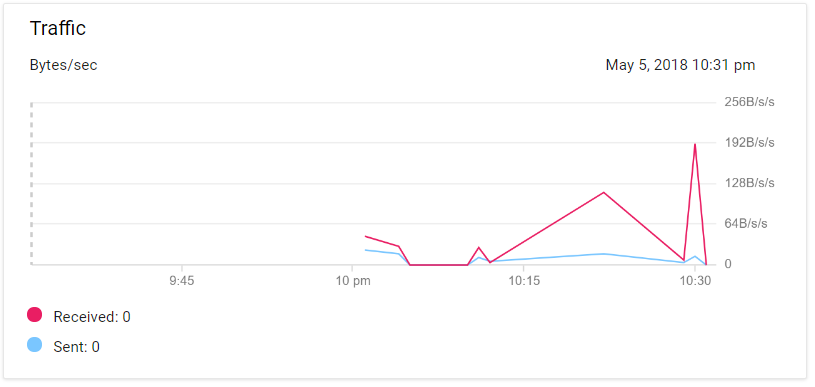


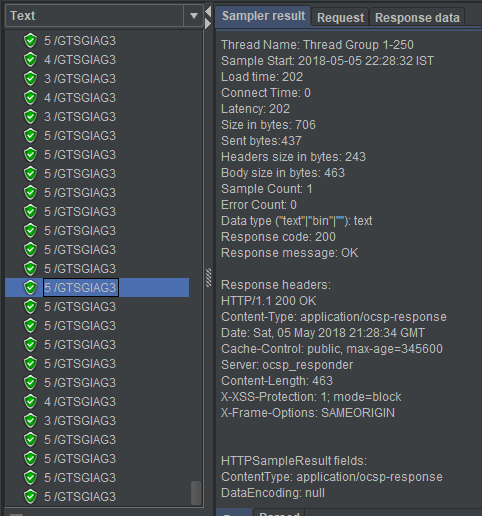
## Test Case 4 Results

In this test I tested the GAE server with 250 users who added 2000 words to the guestbook. Below are the results from this test.



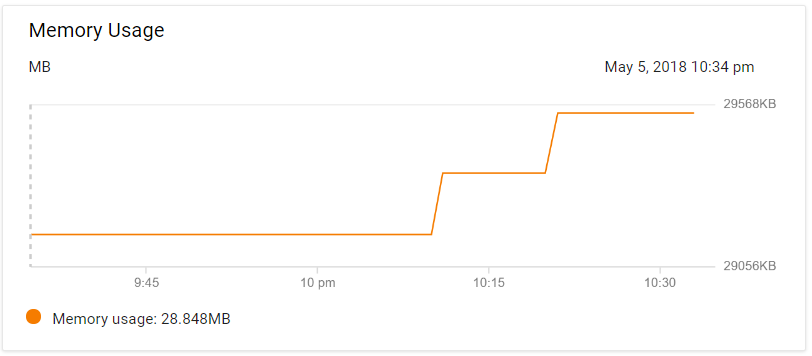




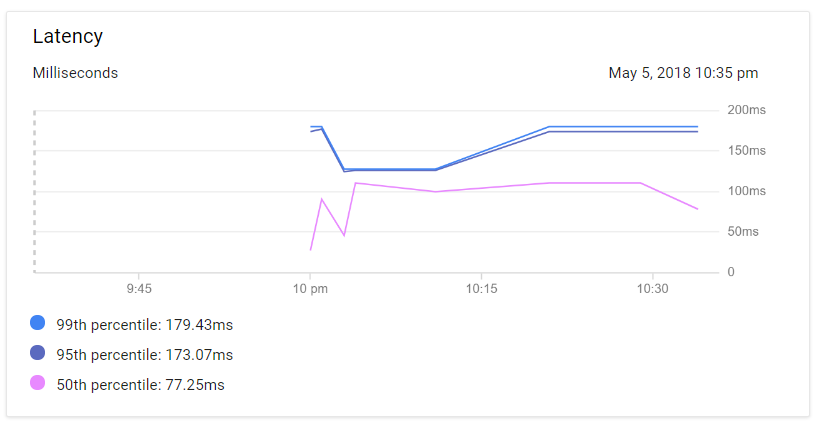


## Test Case 5 Results

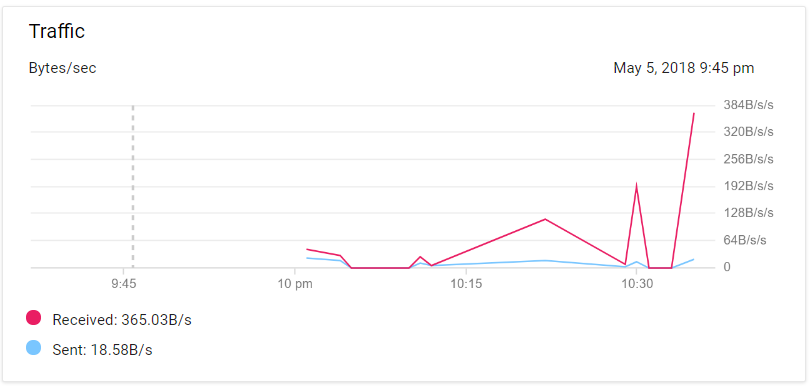
In this test I tested the GAE server with 500 users who added 4000 words to the guestbook. Below are the results from this test and an overall explanation of all 5 tests.



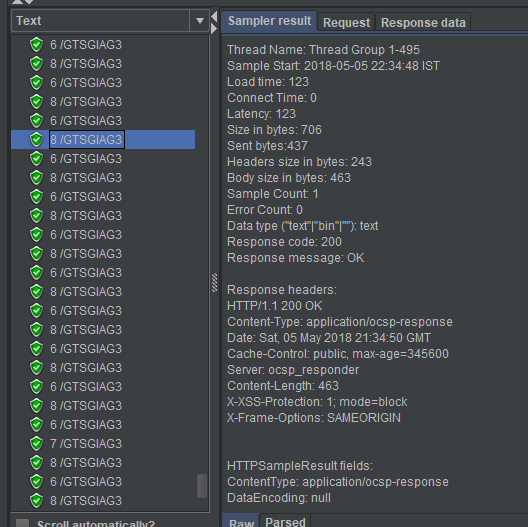
In this screenshot we can see the memory usage of the GAE server after running my 5 test cases. For the first two tests the memory usage of the server averaged at 28.660 MB. For the third test there was an increase usage of 0.188 MB and for the fourth and fifth test there was an increase 0.257 MB. These increases in memory usage are to be expected as the data being received increased.



In this screenshot we can see the latency results of the 50th, 95th and 99th percentile which is a metric similar to average, minimum and maximum for measuring data.



In this screenshot we can see the results of the traffic received and sent from my tests on the GAE web server. From these results we can see for the first two tests the data received had a similar impact on the traffic. For the third test we can see a drastic jump in traffic which has doubled from the first two tests. For the fourth test there was an increase of 66 B/s/s of data received. And finally, for the fifth test the data received almost doubled from the previous data received.



In this screenshot we can see the results of one of the threads (user’s) from JMeter after running the recording/simulation. From these results I am mostly interested in the latency for comparisons with my results from the GAE server. For this thread the latency was 123ms which falls between the 50th and 95th percentile. As for other for the other 499 users the latency fluctuated with some threads being faster than others. But overall the majority of threads latency was in around 115ms – 130ms.

# Conclusion

From the results I found the number of users didn’t make much of a difference to the latency results from the GAE server. For the memory usage there is was a drastic change in the memory usage as the number of users and data increased. There was also a spike in the traffic usage as the number of users and data increased. As for the JMeter results there wasn’t much of a difference between them and the GAE server results, there were just slight differences between threads.

# References

<https://cloud.google.com/appengine/docs/the-appengine-environments>

<https://www.techopedia.com/definition/31267/google-app-engine-gae>

<https://www.sdxcentral.com/cloud/containers/definitions/what-is-docker-container-open-source-project/>

<https://cloud.google.com/appengine/docs/standard/#instance_classes> [Fig 1 and Fig 2]