Byte-Me Project 6 Documentation

**Performance Testing**:

We used JMeter to do performance testing on our app. We tested the speed of GET requests for the site’s home page. 5 tests were done in total, one for 1, 10, 100, 500, and 1000 users. The ramp up period (the amount of time is takes to “ramp up” from the first user to the last user) we chose was 10 seconds for each test. We provided the results of the tests in a folder called JMeter Test Data. The results are available in both graph (png) and table (csv) forms.

Link to performance test data: <https://github.com/ww6vh/Byte-Me/tree/master/cs4501/app/JMeter%20Test%20Data>

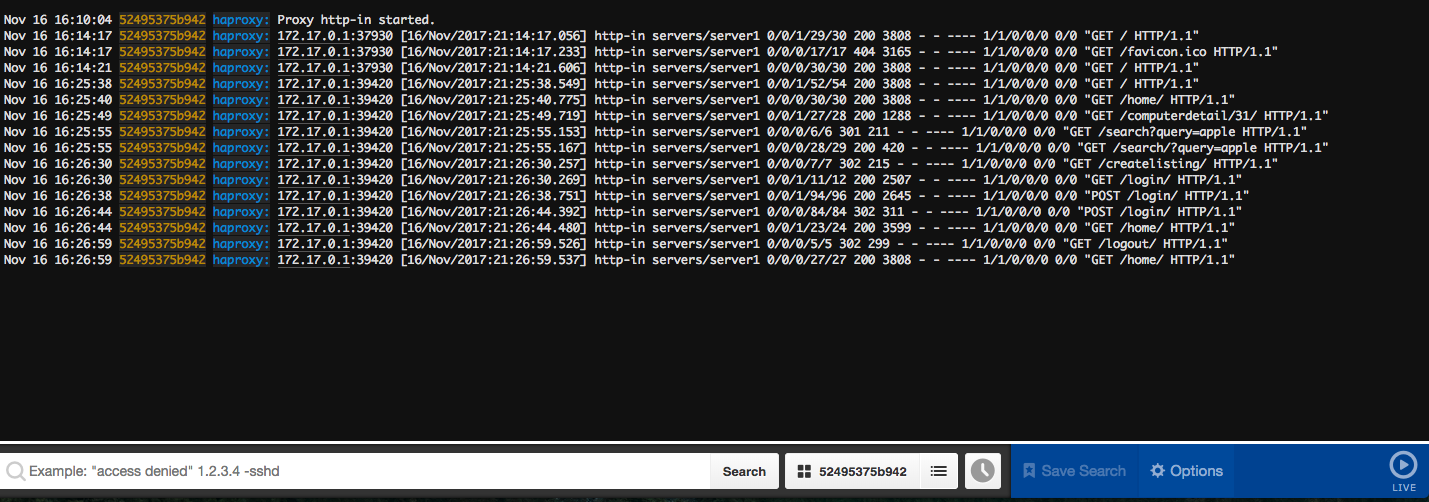
**Load Balancing**:

For load balancing, we added a HAProxy container and a new web front container in docker-compose.yml. We created a Dockerfile and configuration file for HAProxy.

Link to HAProxy files:

https://github.com/ww6vh/Byte-Me/tree/master/cs4501/app/haproxy

HAProxy Papertrail log screenshot:

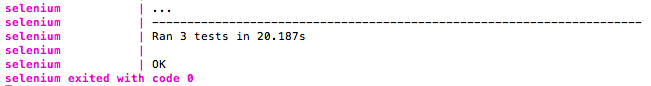


**Integration Tests**:

We added two new containers to docker-compose.yml: Selenium and Selenium-Chrome. We wrote a simple selenium test script to test the home page, signup and login in the web front end.

Link to Selenium test file: https://github.com/ww6vh/Byte-Me/blob/master/cs4501/app/selenium\_test.py

Selenium test screenshot:



**Continuous Integration**:

We used TravisCI to automate builds and run unit tests.

Link to .travis.yml:

https://github.com/ww6vh/Byte-Me/blob/master/.travis.yml

Screenshot:

