## **Message Queuing and Analytics**

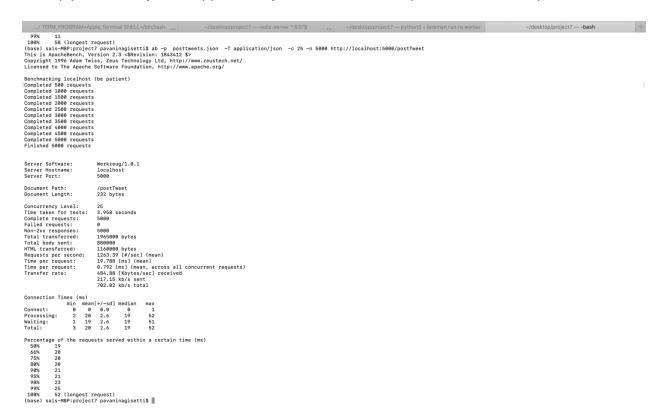
 Using sample posttweets.json we perform load test on the synchronous version of the postTweet(username, text) method. Mean requests per second is 1098.07 as you can see below:

ab -p posttweets.json -T application/json -c 25 -n 5000 http://localhost:5000/postTweet

```
Finished 5000 requests
Server Software:
                                             Werkzeug/1.0.1
Server Hostname:
Server Port:
                                             127.0.0.1
5100
Document Path:
Document Length:
                                             /postTweet/
232 bytes
Concurrency Level:
Time taken for tests:
                                             4.553 seconds
5000
Complete requests:
Complete requests:
Failed requests:
Non-2xx responses:
Total transferred:
Total body sent:
HTML transferred:
                                             0
5000
1965000 bytes
                                            1965000 bytes
910000
1160000 bytes
1099.07 [#/sec] (mean)
22.767 [ms] (mean)
0.911 [ms] (mean, across all concurrent requests)
421.43 [Kbytes/sec] received
195.16 kb/s sent
616.59 kb/s total
Requests per second:
Time per request:
Time per request:
Transfer rate:
Connection Times (ms)
                         Connect:
Processing:
Waiting:
Total:
Percentage of the requests served within a certain time (ms) 50\% \hspace{0.2in} 22
   50%
66%
75%
80%
90%
95%
98%
                  23
24
24
25
99% 34
100% 39 (longest request)
(base) sais-MBP:project7 pavaninagisetti$
```

2) Using sample posttweets.json we perform load test on the asynchronous version of the postTweet(username, text) method. Mean requests per second is 1263.39 as you can see below:

ab -p posttweets.json -T application/json -c 25 -n 5000 http://localhost:5000/postTweet



3) The /trending endpoint currently retrieves hashtags from all the tweets.

