

Message Queuing and Analytics

- 1) 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
```

```
Completed 5000 requests
Finished 5000 requests

Server Software: Werkzeug/1.0.1
Server Hostname: 127.0.0.1
Server Port: 5100

Document Path: /postTweet/
Document Length: 232 bytes

Concurrency Level: 25
Time taken for tests: 4.553 seconds
Complete requests: 5000
Failed requests: 0
Non-2xx responses: 5000
Total transferred: 1965000 bytes
Total body sent: 910000
HTML transferred: 1160000 bytes
Requests per second: 1098.07 [#/sec] (mean)
Time per request: 22.767 [ms] (mean)
Time per request: 0.911 [ms] (mean, across all concurrent requests)
Transfer rate: 421.43 [Kbytes/sec] received
195.16 kb/s sent
616.59 kb/s total

Connection Times (ms)
      min   mean[+/-sd] median   max
Connect:    0      0  0.1      0      4
Processing:  3     23  2.6     22     39
Waiting:    1     22  2.6     22     38
Total:       3     23  2.6     22     39

Percentage of the requests served within a certain time (ms)
 50%    22
 66%    23
 75%    24
 80%    24
 90%    25
 95%    26
 98%    30
 99%    34
100%    39 (longest request)
(base) sais-MBP:project7 pavaninagiseti$
```

- 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

```
...r TERM_PROGRAM=Apple_Terminal SHELL=/bin/bash ...  ~/desktop/project7 -- redis-server *:6379 ...  ~/desktop/project7 -- python3 - foreman run rq worker  ~/desktop/project7 -- -bash  +
99% 11
100% 58 (longest request)
(base) sa1s-MBP:project7 pavaninagiseti$ ab -p posttweets.json -T application/json -c 25 -n 5000 http://localhost:5000/postTweet
This is ApacheBench, Version 2.3 <Revision: 1843412>
Copyright 1996 Adam Twiss, Zeus Technology Ltd, http://www.zeustech.net/
Licensed to The Apache Software Foundation, http://www.apache.org/

Benchmarking localhost (be patient)
Completed 500 requests
Completed 1000 requests
Completed 1500 requests
Completed 2000 requests
Completed 2500 requests
Completed 3000 requests
Completed 3500 requests
Completed 4000 requests
Completed 4500 requests
Completed 5000 requests
Finished 5000 requests

Server Software: Werkzeug/1.0.1
Server Hostname: localhost
Server Port: 5000

Document Path: /postTweet
Document Length: 232 bytes

Concurrency Level: 25
Time taken for tests: 3.958 seconds
Complete requests: 5000
Failed requests: 0
Non-2xx responses: 5000
Total transferred: 1965000 bytes
Total body sent: 880000
HTML transferred: 1100000 bytes
Requests per second: 1263.39 [#/sec] (mean)
Time per request: 19.788 [ms] (mean)
Time per request: 0.792 [ms] (mean, across all concurrent requests)
Transfer rate: 484.88 [Kbytes/sec] received
217.15 kb/s sent
702.02 kb/s total

Connection Times (ms)
      min      mean[+/-sd] median   max
Connect:    0      0   0.0      0      1
Processing:  2     20   2.6     19     52
Waiting:    1     19   2.6     19     51
Total:       3     20   2.6     19     52

Percentage of the requests served within a certain time (ms)
 50%    19
 60%    20
 75%    20
 80%    20
 90%    21
 95%    21
 98%    23
 99%    25
100%    52 (longest request)
(base) sa1s-MBP:project7 pavaninagiseti$
```

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

