Project 4 Report

Dhruv Mahajan, UFID: 42111994
Ashvini Patel, UFID: 47949297

What is implemented:

Implemented the full brief.

Part 1) Twitter Engine, which supports the following functions

- Account Registration.
- Subscription to a user's tweets
- It allows querying tweets subscribed to.
- Tweets with specific hashtags.
- Tweets in which the user is mentioned.
- If the user is connected, we deliver the above types of tweets live (without querying)
- If user is disconnected, the server stores all the tweets for the user and delivers them when he reconnects.

Part 2) Tester/simulator to test the engine

- Each Client can:
  - Send tweet. Tweets can have hashtags (e.g. #COP5615isgreat) and mentions (@bestuser)
  - Re-tweets (so that your subscribers get an interesting tweet you got by other means)
  - Query for mentions, hastags or tweets subscribed to.
- We simulated periods of live connection and disconnection for random users

Part 3) Zip-f distribution

The program (client simulator) takes the minimum number of activities as input. Based on the input, the most subscribed to users (Users with a higher rank) set the minimum number of activities they need to do using the following:

Zip-f version 1:

- Top 1% of the clients do at least 20 times the minimum activities.
- Next 9% of the clients do at least 10 times the minimum activities.
- Next 50% of the clients do at least 2 times the minimum activities.
- Rest 40% of the clients do at least the minimum activities**.**

Example: If we simulate 1000 users with 100 minimum activities, the total requests in the system will be: 250,000

Other considerations:

NOTE: The Client simulator and server are two different processes. These can be on the same or different machines on the same local network but they must get the correct IP address from the init.getif() system call.

We assume that:

- **The first address returned by the init.getif() system call is correct**. If the first IP address returned is not the address of the machine on the local network, no node can be named correctly. Hence nodes can't connect.

Activity: An activity represents a single action that a client randomly selects**. Many actions can be triggered by a single activity**. For example, if a client tweets, it needs to be delivered to all the subscribers.

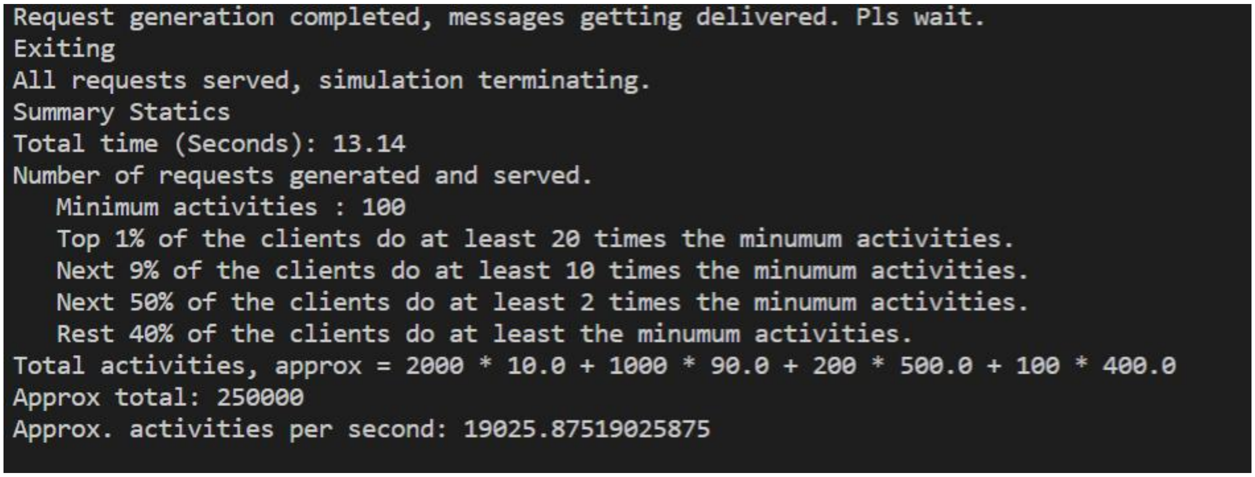
Performance measures

No. of Users	Minimum Activities per user	Total Activities	Time Taken (Seconds)	Activities handled per second	Figure
Zip-f version 1, Two machines					
1,000	100	250,000	13.14	~19,025	Figure 1
10,000	10	250,000	18.469	~13,536	Figure 2
10,000	100	2,500,000	705	~3,542	Figure 3
20,000	10	500,000	171	~2,909	Figure 4
Zip-f version 1, One machine					
10,000	100	2,500,000	86	~28,849	Figure 7
100,000	10	2,500,000	65	~38,332	Figure 8
Zip- version 2, two machines, (only for testing)					
1,000	10	33,000	3.3	~9,731	Figure 5
10,000	100	3,300,000	575	~5,731	Figure 6

We concluded that when using two machines (like in a true client-server environment), **the network is the main bottleneck.**

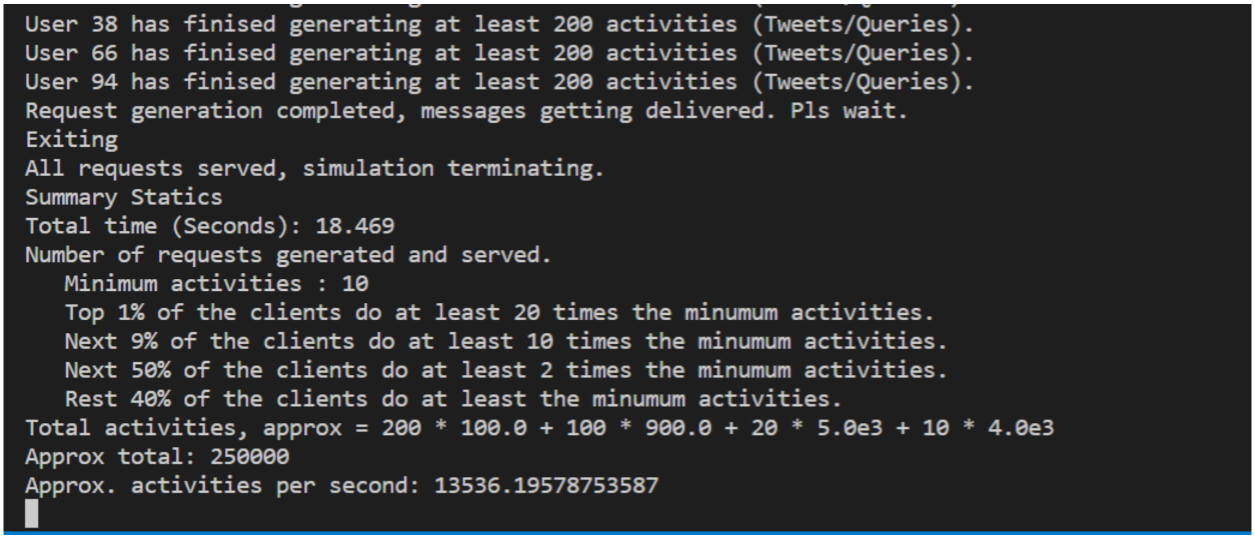
Note Zip-f version 2 is not in the final submission. Just something we tried. It has a very big multiplier in terms of number of activities and hence was not included in the final submission**.**

- Top 1% of the clients do at least 100 times the minimum activities.
- Next 9% of the clients do at least 10 times the minimum activities.
- Next 50% of the clients do at least 2 times the minimum activities.
- Rest 40% of the clients do at least the minimum activities**.**



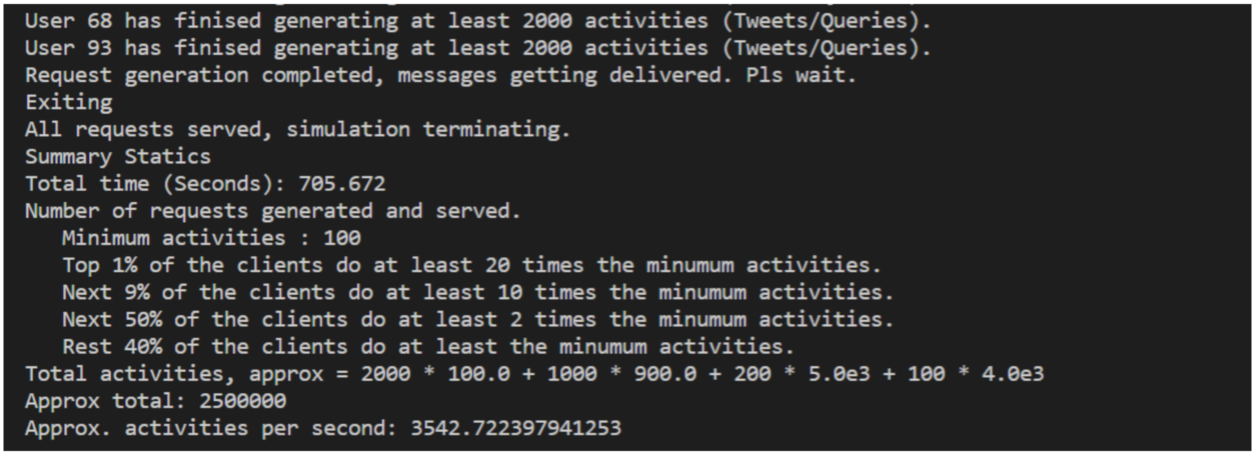
```
Request generation completed, messages getting delivered. Pls wait.
Exiting
All requests served, simulation terminating.
Summary Statics
Total time (Seconds): 13.14
Number of requests generated and served.
   Minimum activities : 100
   Top 1% of the clients do at least 20 times the minumum activities.
   Next 9% of the clients do at least 10 times the minumum activities.
   Next 50% of the clients do at least 2 times the minumum activities.
   Rest 40% of the clients do at least the minumum activities.
Total activities, approx = 2000 * 10.0 + 1000 * 90.0 + 200 * 500.0 + 100 * 400.0
Approx total: 250000
Approx. activities per second: 19025.87519025875
```

Figure 1



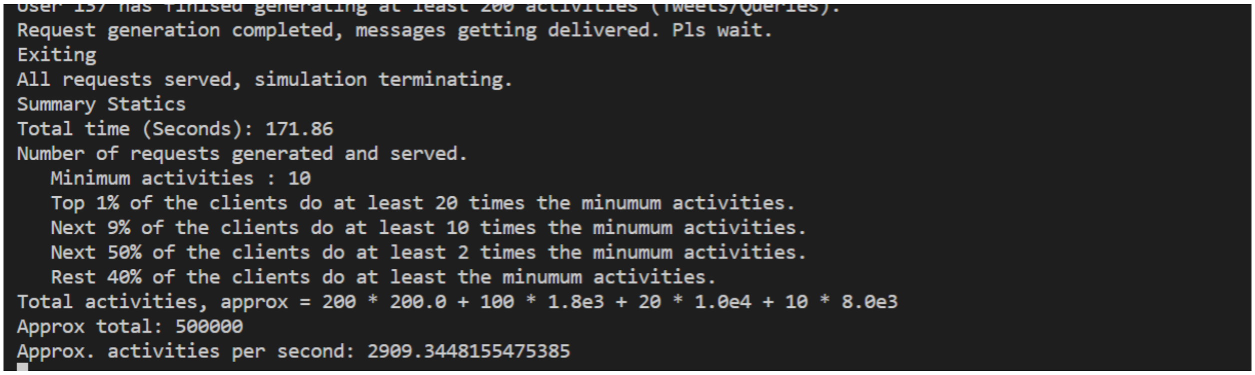
```
User 38 has finised generating at least 200 activities (Tweets/Queries).
User 66 has finised generating at least 200 activities (Tweets/Queries).
User 94 has finised generating at least 200 activities (Tweets/Queries).
Request generation completed, messages getting delivered. Pls wait.
Exiting
All requests served, simulation terminating.
Summary Statics
Total time (Seconds): 18.469
Number of requests generated and served.
   Minimum activities : 10
   Top 1% of the clients do at least 20 times the minumum activities.
   Next 9% of the clients do at least 10 times the minumum activities.
   Next 50% of the clients do at least 2 times the minumum activities.
   Rest 40% of the clients do at least the minumum activities.
Total activities, approx = 200 * 100.0 + 100 * 900.0 + 20 * 5.0e3 + 10 * 4.0e3
Approx total: 250000
Approx. activities per second: 13536.19578753587
```

Figure 2



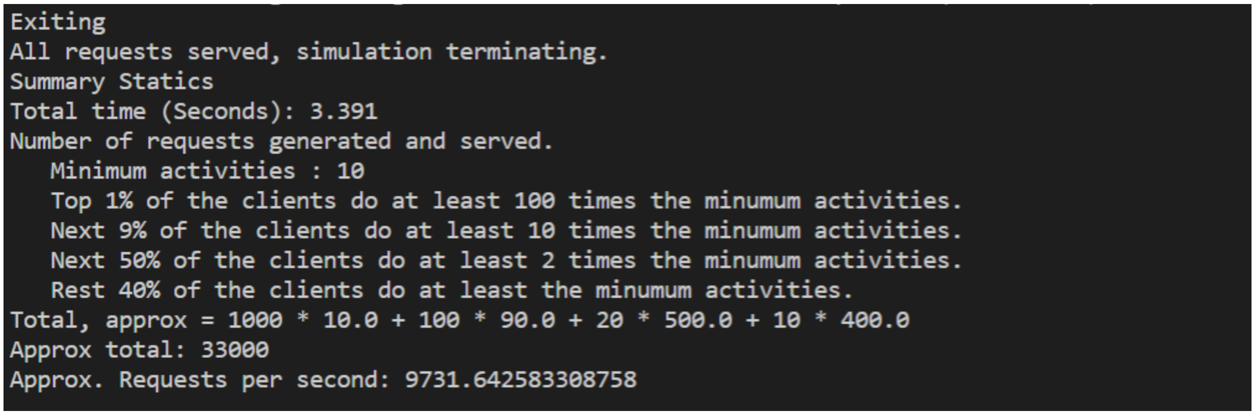
```
User 68 has finised generating at least 2000 activities (Tweets/Queries).
User 93 has finised generating at least 2000 activities (Tweets/Queries).
Request generation completed, messages getting delivered. Pls wait.
Exiting
All requests served, simulation terminating.
Summary Statics
Total time (Seconds): 705.672
Number of requests generated and served.
   Minimum activities : 100
   Top 1% of the clients do at least 20 times the minumum activities.
   Next 9% of the clients do at least 10 times the minumum activities.
   Next 50% of the clients do at least 2 times the minumum activities.
   Rest 40% of the clients do at least the minumum activities.
Total activities, approx = 2000 * 100.0 + 1000 * 900.0 + 200 * 5.0e3 + 100 * 4.0e3
Approx total: 2500000
Approx. activities per second: 3542.722397941253
```

Figure 3



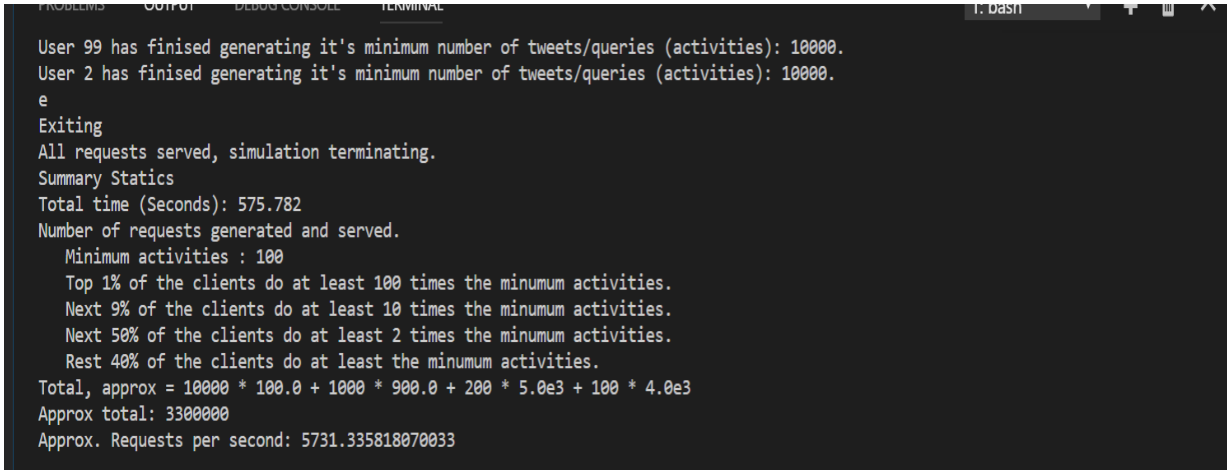
```
User 137 has finised generating at least 200 activities (Tweets/Queries).
Request generation completed, messages getting delivered. Pls wait.
Exiting
All requests served, simulation terminating.
Summary Statics
Total time (Seconds): 171.86
Number of requests generated and served.
   Minimum activities : 10
   Top 1% of the clients do at least 20 times the minumum activities.
   Next 9% of the clients do at least 10 times the minumum activities.
   Next 50% of the clients do at least 2 times the minumum activities.
   Rest 40% of the clients do at least the minumum activities.
Total activities, approx = 200 * 200.0 + 100 * 1.8e3 + 20 * 1.0e4 + 10 * 8.0e3
Approx total: 500000
Approx. activities per second: 2909.3448155475385
```

Figure 4



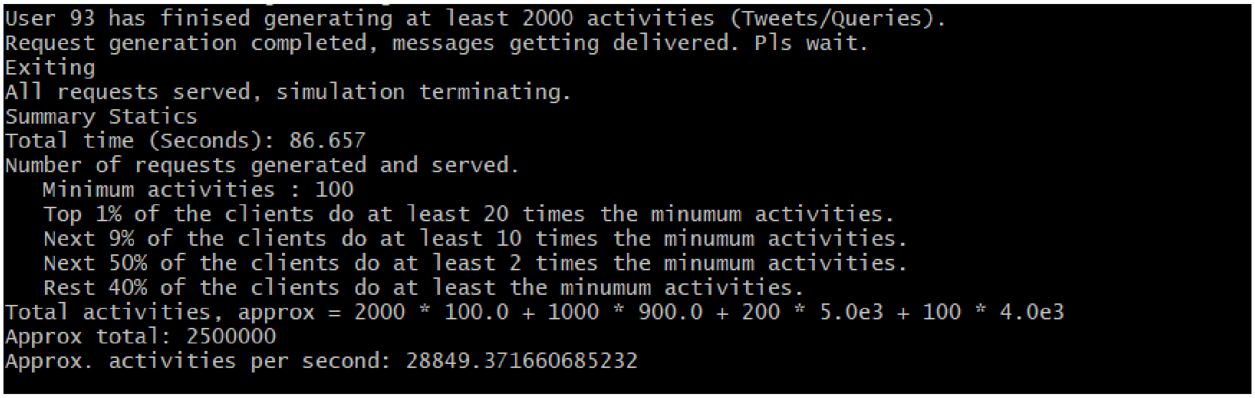
```
Exiting
All requests served, simulation terminating.
Summary Statics
Total time (Seconds): 3.391
Number of requests generated and served.
   Minimum activities : 10
   Top 1% of the clients do at least 100 times the minumum activities.
   Next 9% of the clients do at least 10 times the minumum activities.
   Next 50% of the clients do at least 2 times the minumum activities.
   Rest 40% of the clients do at least the minumum activities.
Total, approx = 1000 * 10.0 + 100 * 90.0 + 20 * 500.0 + 10 * 400.0
Approx total: 33000
Approx. Requests per second: 9731.642583308758
```

Figure 5



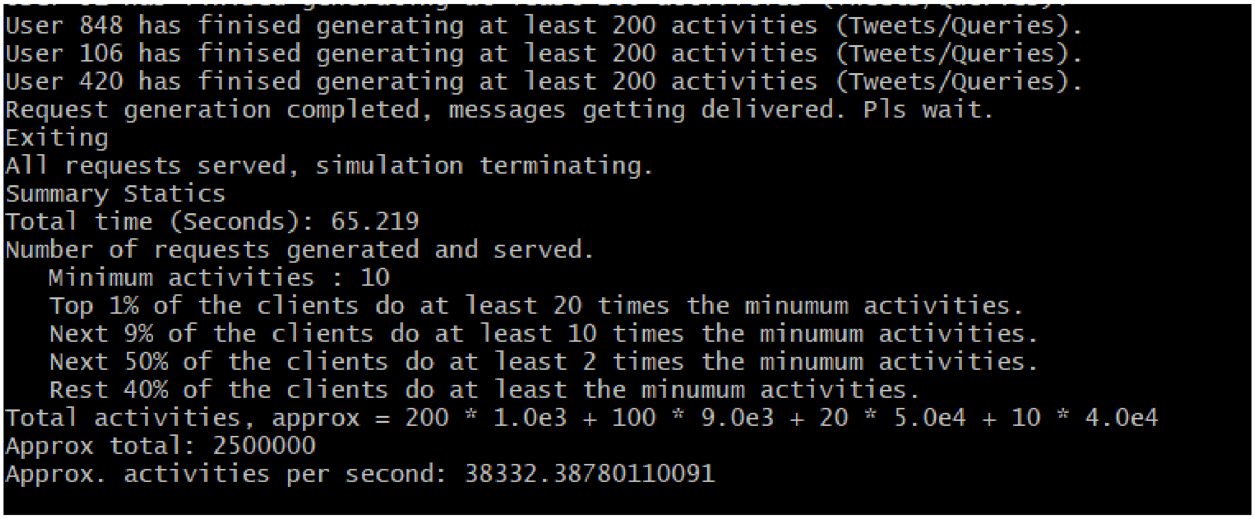
```
User 99 has finised generating it's minimum number of tweets/queries (activities): 10000.
User 2 has finised generating it's minimum number of tweets/queries (activities): 10000.
e
Exiting
All requests served, simulation terminating.
Summary Statics
Total time (Seconds): 575.782
Number of requests generated and served.
   Minimum activities : 100
   Top 1% of the clients do at least 100 times the minumum activities.
   Next 9% of the clients do at least 10 times the minumum activities.
   Next 50% of the clients do at least 2 times the minumum activities.
   Rest 40% of the clients do at least the minumum activities.
Total, approx = 10000 * 100.0 + 1000 * 900.0 + 200 * 5.0e3 + 100 * 4.0e3
Approx total: 3300000
Approx. Requests per second: 5731.335818070033
```

Figure 6



```
User 93 has finised generating at least 2000 activities (Tweets/Queries).
Request generation completed, messages getting delivered. Pls wait.
Exiting
All requests served, simulation terminating.
Summary Statics
Total time (Seconds): 86.657
Number of requests generated and served.
   Minimum activities : 100
   Top 1% of the clients do at least 20 times the minumum activities.
   Next 9% of the clients do at least 10 times the minumum activities.
   Next 50% of the clients do at least 2 times the minumum activities.
   Rest 40% of the clients do at least the minumum activities.
Total activities, approx = 2000 * 100.0 + 1000 * 900.0 + 200 * 5.0e3 + 100 * 4.0e3
Approx total: 2500000
Approx. activities per second: 28849.371660685232
```

Figure 7



```
User 848 has finised generating at least 200 activities (Tweets/Queries).
User 106 has finised generating at least 200 activities (Tweets/Queries).
User 420 has finised generating at least 200 activities (Tweets/Queries).
Request generation completed, messages getting delivered. Pls wait.
Exiting
All requests served, simulation terminating.
Summary Statics
Total time (Seconds): 65.219
Number of requests generated and served.
   Minimum activities : 10
   Top 1% of the clients do at least 20 times the minumum activities.
   Next 9% of the clients do at least 10 times the minumum activities.
   Next 50% of the clients do at least 2 times the minumum activities.
   Rest 40% of the clients do at least the minumum activities.
Total activities, approx = 200 * 1.0e3 + 100 * 9.0e3 + 20 * 5.0e4 + 10 * 4.0e4
Approx total: 2500000
Approx. activities per second: 38332.38780110091
```

Figure 8

Test setup & resource utilization:

All testing and was done using one or two machines with Core i5 processors and 16GB memory each. In case of two, they were connected to the same local network via wireless connections. Since the server is a single process, it doesn’t use up all the on a multicore machine.

RAM Usage:

Server: 500 – 800 MB

Client simulator: Upto 1.5 GB