Twitter Engine Part – 1

Anmol Khanna, 65140549, anmolkhanna93@ufl.edu
Akshay Singh Jetawat, 22163183, akshayt80@ufl.edu

Instructions to run the project: Go inside the project directory and execute the following commands

Step 1: run mix deps.get
Step 2: run mix escript.build , if you don't do the step one first, then it might throw an error and then you will have to run and then do mix escript.build again.

Step 3: To start the server,
./project4part1 server

Step 4: In a new window, here you have two options: To run as an interactive client or run the simulator.

To run the interactive client, do the following:

./project4part1 client 127.0.0.1

You can run as many clients as you want in separate windows.

To run the simulator, do the following:

./project4part1 simulator 127.0.0.1 a b

Here a is the number of users per actor and b is the number of actors and 127.0.0.1 is the server Ip.

Note: To not overkill the processes, please enter at most 200 users per elixir actor.

Sample command for the simulator is:

./project4part1 simulator 127.0.0.1 10 5

Here, 10 is the number of users per actor and 5 is the number of actors, so the total users are on the server would be 50.

Interactive Client:

If you choose this option, then it will first prompt you your username and then once you enter it, it will show you the following screen:

CLIENT SCREEN

```
Enter username: anmol
Options:
1. Tweet
2. Hashtag query
3. Mention query
4. Subscribe
5. Unsubscribe
6. Login
7. Logout
Enter your choice:
```

Server Screen

```
21:17:06.201 [info]  Server Stats
Tweets(per sec): 0.0
Total Users: 1
Online Users: 1
Offline Users: 0
```

You can then choose whatever function you want to use.

Simulator: Once you run the simulator by following the above steps, you will see the following screens:

Server Screen

```
21:19:09.225 [info]  Server Stats
Tweets(per sec): 39.3
Total Users: 50
Online Users: 35
Offline Users: 15
```

Simulator Screen

You will the users being automatically being registered and start tweeting each other. Following is the snapshot of one of the iterations:

```
x4p w5 ajv0ry4scrpy kybd0svfk8kbvgb1qokfc17r88mp0eq w 181uxj6f  a49cbbn1ezidy0eyde97v7c6t
u8tua

21:20:37.096 [info]  username:w8c6h5txy sender: 5lh5a468o incoming tweet:- 43  ex4p w5
ajv0ry4scrpy kybd0svfk8kbvgb1qokfc17r88mp0eq w 181uxj6f  a49cbbn1ezidy0eyde97v7c6t u8tua

21:20:37.096 [info]  username:x2wzrhmj9 sender: 5lh5a468o incoming tweet:- 43  ex4p w5
ajv0ry4scrpy kybd0svfk8kbvgb1qokfc17r88mp0eq w 181uxj6f  a49cbbn1ezidy0eyde97v7c6t u8tua

21:20:37.123 [info]  username:x2wzrhmj9 sender: thm81i0kv incoming tweet:- g n ycfi
hxoqsh00694lz9fdx3ao7w9sv92052y0936va s9euptvpwnte382it4oyv4zlx3d16uuyak crwgl0u55m8jdrkw40
```

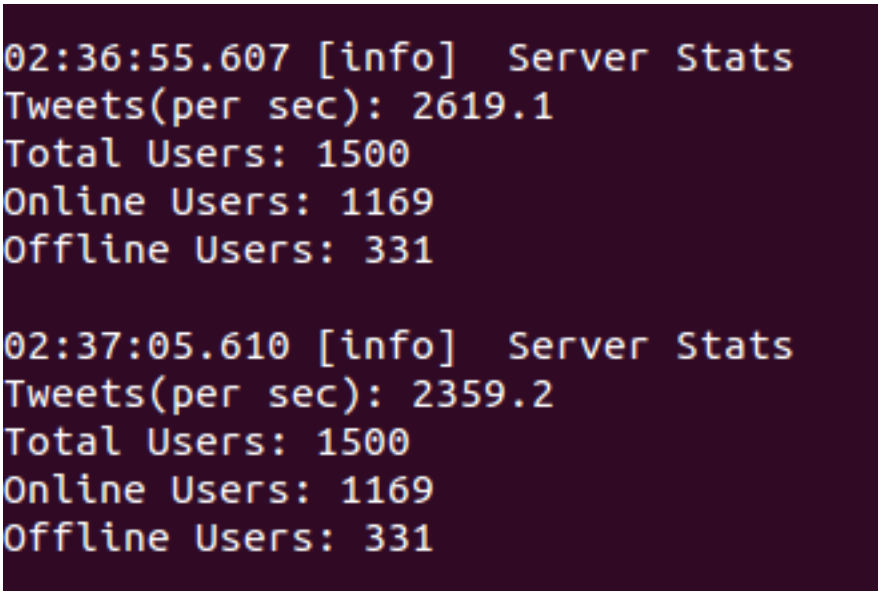
```
21:20:37.128 [info]  username:jrjddk2fo sender: 03vzs1i7y incoming tweet:- 02 gz9o 9 5cz7 yfnrgdc3hyhpryq0kfm8joil x7inudt xjpvy tu1  tau2xi80 2m4zc87 4mp1ood xfnrv4v4 y32i8fp

21:20:37.128 [info]  username:ngs8fceok sender: 03vzs1i7y incoming tweet:- 02 gz9o 9 5cz7 yfnrgdc3hy
```

Implementation Details:

We are using TCP IP protocol for communication between the client and the server. Also, we are using couple of additional libraries such as the `social_parser`
Social_parser and Poison. Social_parser helps to extract the relevant information from the tweets and Poison is a JSON library that we are using for sending and receiving the message between the client and the server as JSON string.

Performance:

Our interactive client does all the functionalities shown in the menu options above. We haven't given the functionality for setting the password for the user. They can just log in by their username which will be unique for every user. Also all the functionalities work as they work in real Twitter application. For the simulator, following is the snapshot of the performance as recorded by our server.



Also, when we first created the tcp connection, we were creating a new tcp connection for each client and the sockets have a maximum limit of 1010 connections and hence our engine works for maximum 1010 connections. As we were hitting the limits of tcp connection sockets we changed our implementation a bit to use single tcp connection for multiple users. With which we were able to increase the number of users. Also, this modification allows us to run multiple simulators simultaneously.