

# Distributed Operating System Principles Project 4

## Twitter Clone Part II

Date: 15th Dec, 2022

Team members:

Hrushyang Adloori, UFID: 86776220, adloorih@ufl.edu

Goutham Kumar Mekala, UFID: 29386297, mekala.g@ufl.edu

### Implementation Details:

- Initially both server and client files have to be compiled. The compiled .beam files have to be generated in their respective directories or nodes.  
`>c(twitter_socket_engine).`  
`>c(twitter_client).`
- Run the server to establish the main twitter engine.  
`>twitter_socket_engine:initiate().`
- Once the twitter engine is successfully started, we run clients on different nodes.  
`>twitter_client:initiate().`
- All functionalities can be achieved using the command line instructions.

### Implementation of Functionalities:

(We have to type the command string exactly, or else the code prompts an invalid command exception and asks to retry again.)

- Sign up:** Nothing can be done without registering an account. The code handles the error by instructing the user to register the account first. After opting to sign up, give a unique username which has not been used before. If used, the code prompts you to give another username to retry.

```

PS C:\Users\hrush\OneDrive\Documents\Erlang Project\project-4-2\DOSP-4.2
> erl
Eshell V13.0.4 (abort with ^G)
1> c(twitter_socket_engine).
{ok,twitter_socket_engine}
2> twitter_socket_engine:initiate().

*****TWITTER ENGINE INITIATED*****

New Registration Initiated by USER : "gouthi"
JSON Formatted Data :: []

PS C:\Users\hrush\OneDrive\Documents\Erlang Project\project-4-2\DOSP-4.2
2> erl
Eshell V13.0.4 (abort with ^G)
1> c(twitter_client).
{ok,twitter_client}
2> twitter_client:initiate().

*****Welcome to Twitter*****

Sending Connection Request to the Twitter Engine!
Received Response!
Socket Connection Successful!!!
1. signup      2. login
3. tweet      4. re-tweet
5. follow     6. util
7. logout

Enter Input to perform following Action: signup

Enter the User Name: gouthi
SELF: <0.96.0>

Account has been Registered
Received Response!
User Registration Successful!
1. signup      2. login
3. tweet      4. re-tweet
5. follow     6. util
7. logout

Enter Input to perform following Action:

```

Fig.1: Implementation screenshot of Signup and Login

- **login** : After initial sign up, you're logged in by default. But once logged out, we can login to previously registered accounts using this command. We have to enter the exact username used before. Then we continue to achieve further functionalities.
- **tweet** : After logging in to desired account, we can now tweet anything we want that can include message, mentions (containing "@") and hashtags (containing "#"). This tweet is concatenated to the JSON file and sent to the server. This file contains key-value storage, that stores tweets in whole, mentions, hashtags and followers.

```

PS C:\Users\hrush\OneDrive\Documents\Erlang Project\project-4-2\DOSP-4.2
> erl
Eshell V13.0.4 (abort with ^G)
1> c(twitter_socket_engine).
{ok,twitter_socket_engine}
2> twitter_socket_engine:initiate().

*****TWITTER ENGINE INITIATED*****

New Registration Initiated by USER : "gouthi"
JSON Formatted Data :: []

"gouthi" sent the following tweet: "Hi Twitter2.0 #firsttweet#newtotwitter"
JSON Formatted Data: [{"gouthi", [{"followers", []}, {"tweets", []}]}]
["Hi Twitter2.0 #firsttweet#newtotwitter"]

Tweet Successful New Changes: [{"gouthi", [{"followers", []}, {"tweets", [{"Hi Twitter2.0 #firsttweet#newtotwitter"}]}]}]

er\n]]]]]

No followers!
Running!!!

PS C:\Users\hrush\OneDrive\Documents\Erlang Project\project-4-2\DOSP-4.2
7. logout

Enter Input to perform following Action: signup

Enter the User Name: gouthi
SELF: <0.96.0>

Account has been Registered
Received Response!
User Registration Successful!
1. signup      2. login
3. tweet      4. re-tweet
5. follow     6. util
7. logout

Enter Input to perform following Action: tweet

Tweet freely since now it's a free bird :Hi Twitter2.0 #firsttweet#newtotwitter

Tweet Sent !
Received Response!
Tweet Processed by Server!

1. signup      2. login
3. tweet      4. re-tweet
5. follow     6. util
7. logout

Enter Input to perform following Action:

```

Fig.2: Implementation screenshot of tweet

- **follow** : This functionality allows the user to subscribe to any existing user. We need to enter the exact username to subscribe when prompted. By doing this, the user gets live updates of tweets tweeted by the subscribed user.

<pre>2.0 #firsttweet#newtotwitter\n",     "Excited fo r Avatar2#newearth#fdfs#jamesrocks\n"]]]}} No followers! Running!!!  New Registration Initiated by USER : "hru" JSON Formatted Data :: []  "hru" sent the following tweet: "Elon is chan ging the AI industry!@gouthi#openai\n" JSON Formatted Data: [{"hru", [{"followers", []} , {"tweets", []}]]} ["Elon is changing the AI industry!@gouthi#ope nai\n"]  Tweet Sucessful New Changes: [{"hru", [{"followers", [{"tweets", ["Elon is ch anging the AI industry!@gouthi#openai\n"]}]}} No followers! Running!!! ["hru"]  Output after subscribing: [{"gouthi", [{"followers", ["hr u"]}, {"tweets", ["Hi Twitter2.0 #firsttweet#newtotwitter\n", "Excited for Av atar2#newearth#fdfs#jamesrocks\n"]}]}} []</pre>	<pre>Enter Input to perform following Action: tweet  Tweet freely since now it's a free bird :Hi Tw itter2.0 #firsttweet#newtotwitter  Tweet Sent ! Received Response! Tweet Processed by Server!  1. signup      2. login 3. tweet       4. re-tweet 5. follow      6. util 7. logout  Enter Input to perform following Action: tweet  Tweet freely since now it's a free bird :Excit ed for Avatar2#newearth#fdfs#jamesrocks  Tweet Sent ! Received Response! Tweet Processed by Server!  1. signup      2. login 3. tweet       4. re-tweet 5. follow      6. util 7. logout  Enter Input to perform following Action: []</pre>	<pre>7. logout  Enter Input to perform following Action: twee t  Tweet freely since now it's a free bird :Elon is changing the AI industry!@gouthi#openai  Tweet Sent ! Received Response! Tweet Processed by Server!  1. signup      2. login 3. tweet       4. re-tweet 5. follow      6. util 7. logout  Enter Input to perform following Action: foll ow  Who do you want to subscribe to?:gouthi  Subscribed! Received Response! Subscribed!  1. signup      2. login 3. tweet       4. re-tweet 5. follow      6. util 7. logout  Enter Input to perform following Action: []</pre>
--	--	---

Fig.3: Implementation screenshot of Subscribe

- **re-tweet** : We can use this functionality to retweet an existing tweet. For re-tweeting, we need to enter the exact username who has tweeted it previously. Then we need to give input of the tweet he has previously tweeted. This counts as a retweet for the current user.

<pre>"Excited to watch avatar the way of water!#ne wearth\n"]  Tweet Sucessful New Changes: [{"gouthi", [{"followers", ["hru"]}, {"tweets", ["Hi Twitter 2.0 #firsttweet#newtotwitter\n", "Excited fo r Avatar2#newearth#fdfs#jamesrocks\n", "Excited to watch avatar the way of water!#newearth\n"]}] }} Client to send: "hru"  Remaining List: []  Client Socket: #Port&lt;0.6&gt;  No followers! Running!!! Running!!!  Query: The current username is -&gt; "hru" Subscribed User Search Sub_UserName: "gouthi"  "hru" wants to queryUser Requested Retweet fr om: "gouthi" Tweet to be re-posted: "Hi Twitter2.0 #firstt weet#newtotwitter\n"  No followers! Running!!!  "hru" wants to Retweet</pre>	<pre>Tweet freely since now it's a free bird :Excit ed for Avatar2#newearth#fdfs#jamesrocks  Tweet Sent ! Received Response! Tweet Processed by Server!  1. signup      2. login 3. tweet       4. re-tweet 5. follow      6. util 7. logout  Enter Input to perform following Action: tweet  Tweet freely since now it's a free bird :Excit ed to watch avatar the way of water!#newearth  Tweet Sent ! Received Response! Your tweet has been sent Received Response! Tweet Processed by Server!  1. signup      2. login 3. tweet       4. re-tweet 5. follow      6. util 7. logout  Enter Input to perform following Action: []</pre>	<pre>Enter Input to perform following Action: New tweet received! gouthi:Excited to watch avatar the way of wat er!#newearth  Enter Input to perform following Action: re-t weet  Enter the Username whose tweet you want to re -tweet: gouthi  Enter the tweet that you want to repost: Rece ived Response!  Enter the tweet that you want to repost: Hi T witter2.0 #firsttweet#newtotwitter Excited for Avatar2#newearth#fdfs#jamesrocks Excited to watch avatar the way of water!#new earth  Enter the tweet that you want to repost: Hi T witter2.0 #firsttweet#newtotwitter  Retweeted Received Response! Server processed retweet  1. signup      2. login 3. tweet       4. re-tweet 5. follow      6. util 7. logout  Enter Input to perform following Action: []</pre>
--	--	--

Fig.4: Implementation screenshot of Re-tweet

- **logout** : Once we are done, we can log out of this account which suspends the sustained socket connection. While the records are not lost until the server is shutdown.
- **util** : The util functionality contains another sub-menu that contains commands which can be used to query tweets. By opting this command. We are introduced to new set of functionalities that are described below:

(Enter input as integers denoted to the command respectively.)

- ❖ **@MyMentions** : A user can mention another user in their tweet using the symbol “@” followed by the username of the user to be tagged with no spaces. This tags user, when opted this command, the user can see all tweets they are mentioned in.

<pre> {"tweets", ["Hi Twitter 2.0 #firsttweet#newtotwitter\n", "Excited fo r Avatar2#newearth#fdfs#jamesrocks\n", "Excited to watch avatar the way of water!#newearth\n"]}] }} Client to send: "hru"  Remaining List: []  Client Socket: #Port&lt;0.6&gt;  No followers! Running!!! Running!!! Query: The current username is -&gt; "hru" Subscribed User Search Sub_UserName: "gouthi"  "hru" wants to queryUser Requested Retweet fr om: "gouthi" Tweet to be re-posted: "Hi Twitter2.0 #firsttw eet#newtotwitter\n"  No followers! Running!!!  "hru" wants to Retweet Query: The current username is -&gt; "gouthi" My @mentions! Sub_UserName: "gouthi" Tweets Found: ["Elon is changing the AI indust ry!@gouthi#openai\n"]  "gouthi" wants to query] </pre>	<pre> Tweet Sent ! Received Response! Your tweet has been sent Received Response! Tweet Processed by Server!  1. signup      2. login 3. tweet      4. re-tweet 5. follow     6. util 7. logout  Enter Input to perform following Action: util  Querying Options:  1. @MyMentions 2. HashtagQuery 3. Tweets of User  Enter the number of the required Functionality : 1 Received Response! Elon is changing the AI industry!@gouthi#openai  1. signup      2. login 3. tweet      4. re-tweet 5. follow     6. util 7. logout  Enter Input to perform following Action: [] </pre>	<pre> Enter the tweet that you want to repost: Hi Twitte r2.0 #firs ttweet#new totwitter  Retweeted Received R esponse! Server pro cessed ret weet  1. signup      2. login 3. tweet      4. re-tweet 5. follow     6. util 7. logout  Enter Inpu t to perfo rm followi ng Action: </pre>
---	---	---

Fig.5: Implementation screenshot of MyMentions

- ❖ **HashtagQuery** : The user will send a request to the twitter engine to search all tweets that contain the hashtag given as input.

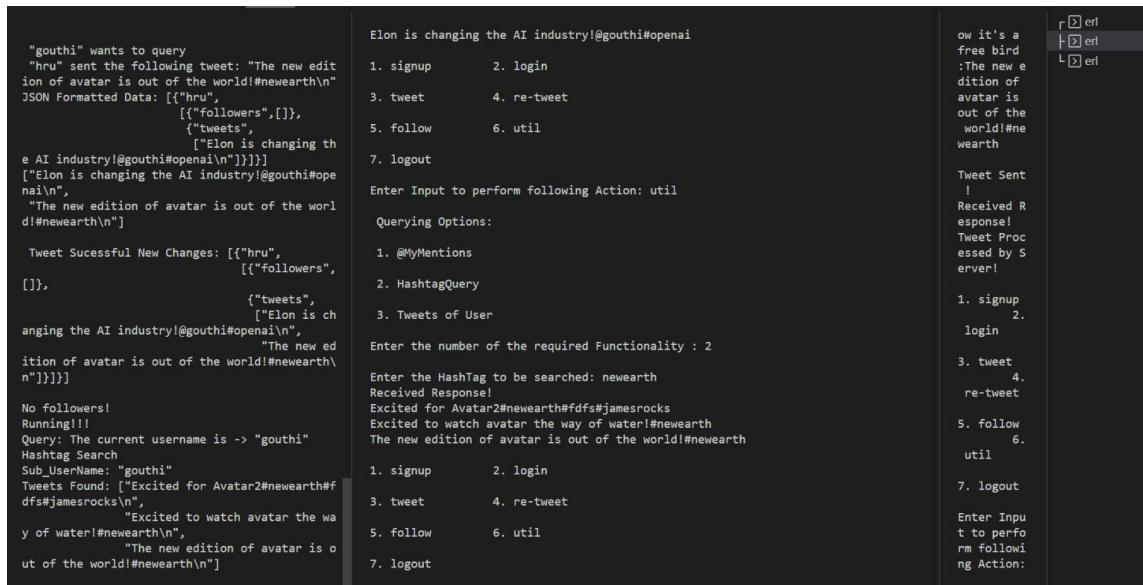


Fig.5: Implementation screenshot of HashtagQuery newearth

❖ **Tweets of User** : Upon giving input of the username, the function previews all the tweets tweeted by that particular user on the client console.

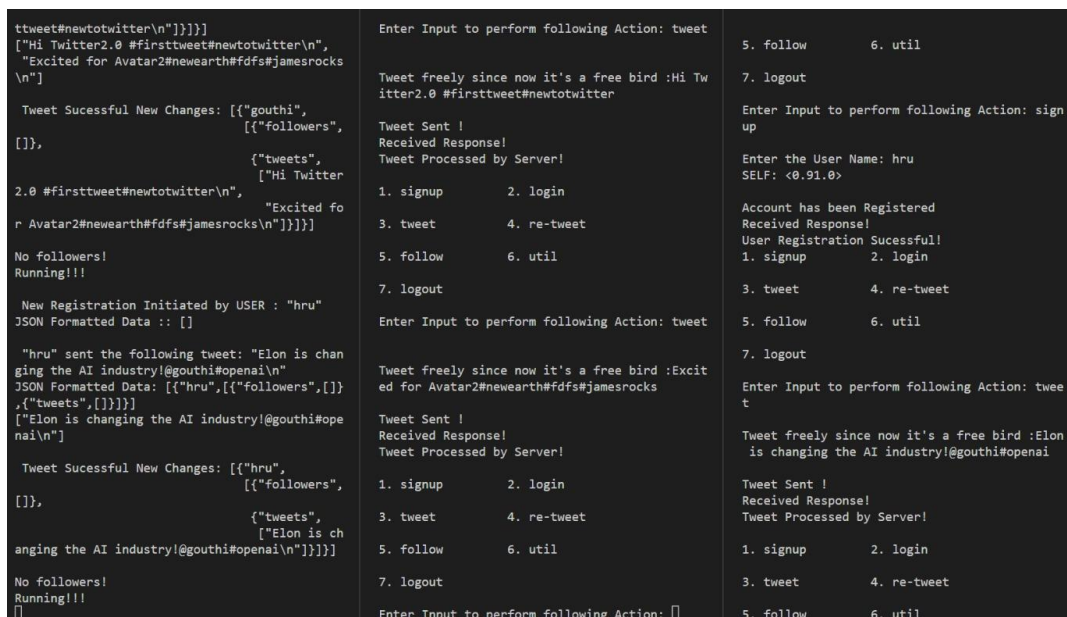


Fig.6: Implementation screenshot of Mentions



<pre> "gouthi" sent the following tweet: "Excited to watch avatar the way of water!#newearth\n" JSON Formatted Data: [{"gouthi",   [{"followers",["hru"]},     {"tweets",       ["Hi Twitter2.0 #firsttweet#newtotwitter\n",         "Excited for Avatar2#newearth#fdfs#jamesrocks\n"]}]},     [{"Hi Twitter2.0 #firsttweet#newtotwitter\n",       "Excited for Avatar2#newearth#fdfs#jamesrocks\n",       "Excited to watch avatar the way of water!#newearth\n"}]}] Tweet Successful New Changes: [{"gouthi",   [{"followers",     {"tweets",       ["Hi Twitter2.0 #firsttweet#newtotwitter\n",         "Excited for Avatar2#newearth#fdfs#jamesrocks\n",         "Excited to watch avatar the way of water!#newearth\n"]}]}]}] Client to send: "hru" Remaining List: [] Client Socket: #Port&lt;0.6&gt; No followers! Running!!! Running!!! [] </pre>	<pre> Tweet freely since now it's a free bird :Excited for Avatar2#newearth#fdfs#jamesrocks  Tweet Sent ! Received Response! Tweet Processed by Server!  1. signup      2. login 3. tweet       4. re-tweet 5. follow      6. util 7. logout  Enter Input to perform following Action: tweet  Tweet freely since now it's a free bird :Excited to watch avatar the way of water!#newearth  Tweet Sent ! Received Response! Your tweet has been sent Received Response! Tweet Processed by Server!  1. signup      2. login 3. tweet       4. re-tweet 5. follow      6. util 7. logout  Enter Input to perform following Action: [] </pre>	<pre> Tweet Sent ! Received Response! Tweet Processed by Server!  1. signup      2. login 3. tweet       4. re-tweet 5. follow      6. util 7. logout  Enter Input to perform following Action: follow  Who do you want to subscribe to?:gouthi  Subscribed! Received Response! Subscribed! 1. signup      2. login 3. tweet       4. re-tweet 5. follow      6. util 7. logout  Enter Input to perform following Action: Received Response!  Enter Input to perform following Action: New tweet received! gouthi:Excited to watch avatar the way of water!#newearth  Enter Input to perform following Action: [] </pre>
---	--	--

Fig.7: Implementation screenshot of live feed of subscribed user

<pre> [{"followers",[]}, {"tweets",   ["Elon is changing the AI industry!@gouthi#openai\n"]}]}] ["Elon is changing the AI industry!@gouthi#openai\n",   "The new edition of avatar is out of the world!#newearth\n"]  Tweet Successful New Changes: [{"hru",   [{"followers",     {"tweets",       ["Elon is changing the AI industry!@gouthi#openai\n",         "The new edition of avatar is out of the world!#newearth\n"]}]}]}]  No followers! Running!!! Query: The current username is -&gt; "gouthi" Hashtag Search Sub_UserName: "gouthi" Tweets Found: ["Excited for Avatar2#newearth#fdfs#jamesrocks\n",   "Excited to watch avatar the way of water!#newearth\n",   "The new edition of avatar is out of the world!#newearth\n"]  "gouthi" wants to queryQuery: The current username is -&gt; "gouthi" Subscribed User Search Sub_UserName: "hru"  "gouthi" wants to query[] </pre>	<pre> Enter the HashTag to be searched: newearth Received Response! Excited for Avatar2#newearth#fdfs#jamesrocks Excited to watch avatar the way of water!#newearth The new edition of avatar is out of the world!#newearth  1. signup      2. login 3. tweet       4. re-tweet 5. follow      6. util 7. logout  Enter Input to perform following Action: util  Querying Options:  1. @MyMentions 2. HashtagQuery 3. Tweets of User  Enter the number of the required Functionality : 3  Whose tweets do you want? hru Received Response! Elon is changing the AI industry!@gouthi#openai The new edition of avatar is out of the world!#newearth  1. signup      2. login 3. tweet       4. re-tweet 5. follow      6. util </pre>	<pre> ow it's a free bird :The new edition of avatar is out of the world!#newearth  Tweet Sent ! Received Response! Tweet Processed by Server!  1. signup      2. login 3. tweet       4. re-tweet 5. follow      6. util 7. logout  Enter Input to perform following Action: [] </pre>
--	--	---

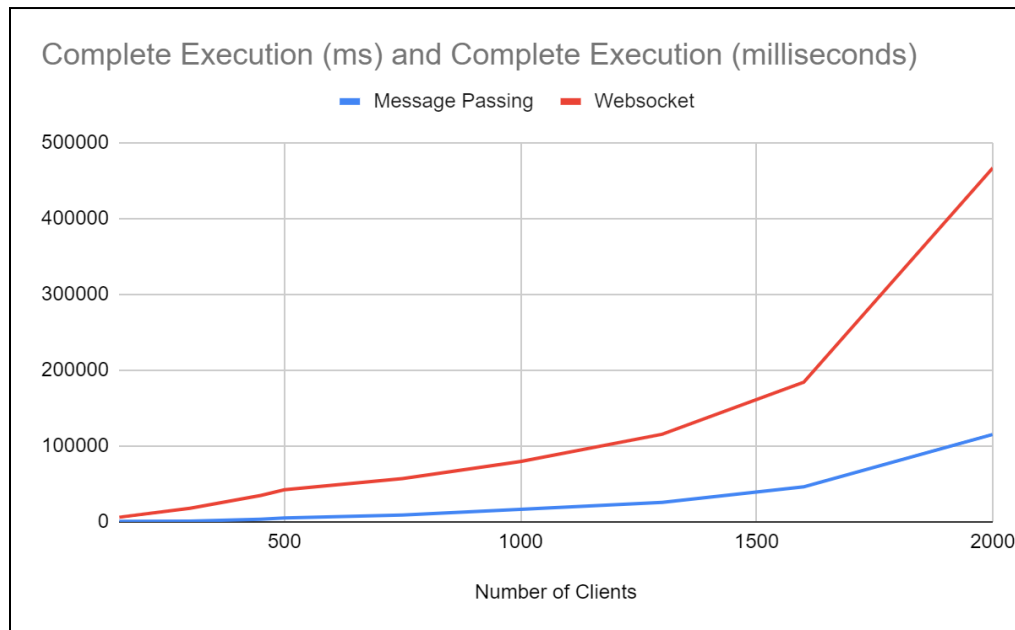
Fig.8: Implementation screenshot of tweets of user "hru" previewed from "gouthi"

## Performance Results:

For message-passing systems, a graph is generated to show how the number of clients affects how long the simulation lasts. A maximum of 6,000 clients have been generated on our system.

### Performance using Websocket interface when Max Subscribers = Number of Clients

Max Subscribers	Time to Tweet, Retweet (ms)	Query Tweets Subscribed (ms)	Query Tweets by Hashtag (ms)	Query Tweets by Mention (ms)	Query All Relevant Tweets (ms)	Complete Execution (ms)
199	270.54	1254.76	378	80.98	67.34	6436
399	2654.43	16543.87	2345.87	1234.65	387.87	64567
799	3245.76	23456.76	9876.98	9543.76	5432.78	112341
1199	4532.67	47543.79	19876.93	6587.34	5463.26	199634
1999	5437.82	76875.34	13679.65	11543.63	1654.37	575437



## Performance using Websocket interface for different number of Max Subscribers / Max Tweets per Account

Max Subscribers	Time to Tweet & Retweet (ms)	Query Tweets Subscribed To (milliseconds)	Query Tweets by Hashtag (ms)	Query Tweets by Mention (ms)	Query All Relevant Tweets (ms)	Complete Execution for Websocket (ms)
200	247.11	224.68	1017.83	594.71	218.51	5467
400	252.4	380.54	1362.83	549.22	197.97	8745
800	244.45	2791.65	1774.67	876.11	589.93	19874
1200	2443.35	18098.63	5158.14	4887.51	2263.81	123456
1999	4104	37803.92	19021.37	12507.01	7031.84	167897

