Socket Project
Nathan Berman
1210954365

Message Formats:
“register <username>” this command is sent from the client to the server in the form ‘RegisterUser#Username#IPAddress#Port’ so that the server knows the username of the new client as well as the ip/port it will be communicating on.

“setup-dht <size> <leader>” this command is sent by any registered client to the server and tells the server to run SetupDHT in the form ‘SetupDHT#Size#UsernameOfLeader’ which the server can use to see if it has enough clients to host that size of DHT and if it has a user whose name matches the desired leader for the DHT.

“dht-complete <leader>” is a command sent in the form ‘CompleteDHT#Username’ to the server by the leader-client after it has finished setting up the distributed hash table. This command tells the server that the clients are no longer in the process of setting up the DHT and are available for incoming commands.

“query-dht <Alpha-3 Code>” is a command sent in the form ‘QueryDHT#QueryUser#CountryCode’ this command tells the server to send a new message of the form ‘QueryDHT#CountryCode#QueryUserIP#QueryUserPort’ to a random user in the DHT. if the user has the queried information stored in its local hash table then the message gives the user the information to send the response in the form ‘QueryResponse#CountryData’ to the inquiring user.

“leave-dht <username>” is a command of the form “LeaveDHT#Username” which tells the server to tell the specified user to leave the DHT in the form “LeaveDHT”. This involves a few steps like telling the leaving member’s predecessor to change the IP/Port that it points to to the old DHT member’s neighbors, as well as making the immediate neighbor to the old DHT member the new leader and rebuilding the DHT.

‘join-dht <username>” is similar to the leave-dht command but it instead adds a new member to the end of the hash table and then rebuilds it. This command is sent in the form “JoinDHT <username>”.

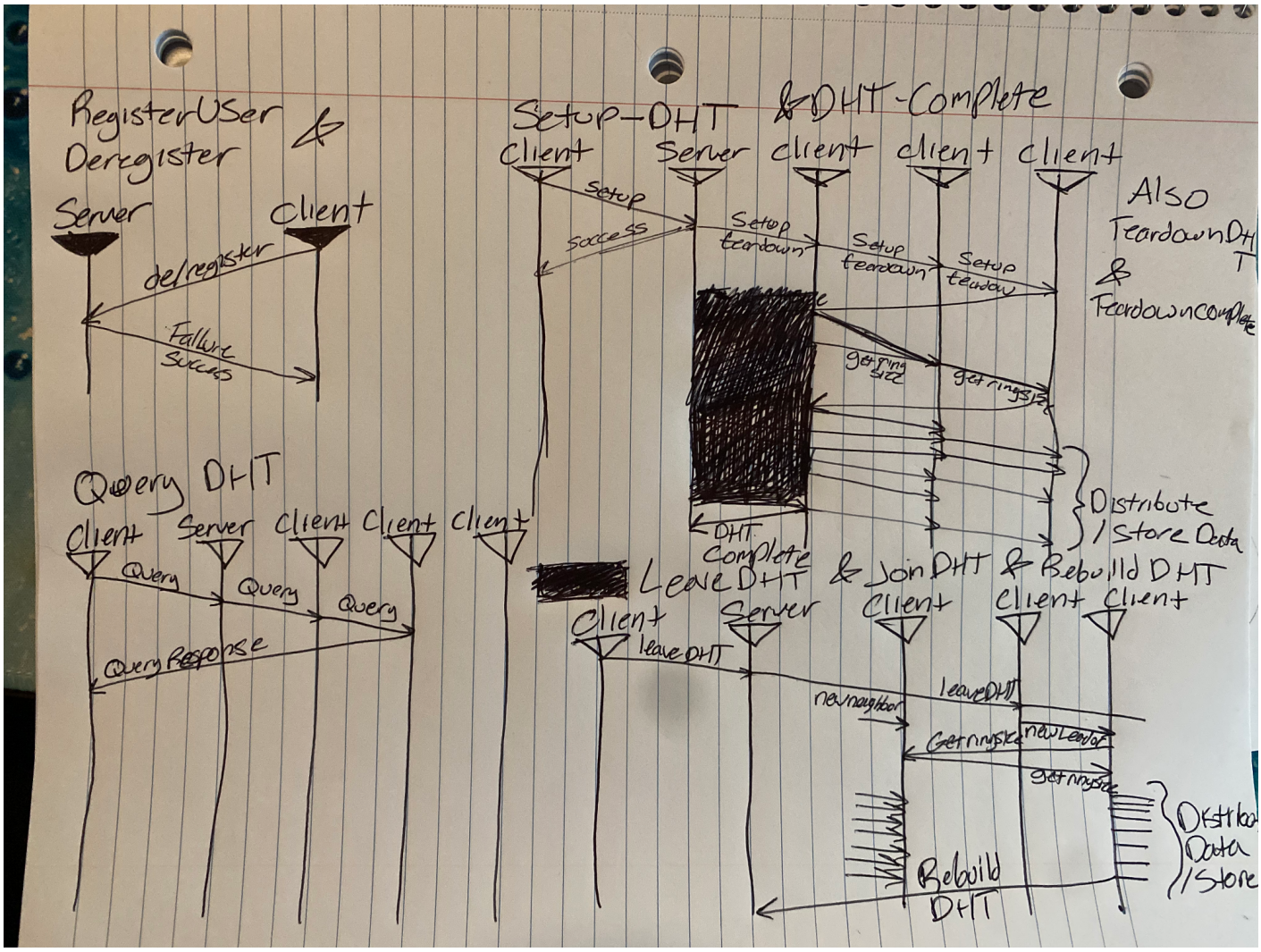
“dht-rebuilt” is an automatically used command which I made in the form “RebuildDHT#Username#LeaderIP#LeaderPort#Polarity” where the username is of the user that either just joined or left the DHT and the polarity is the descriptor of the RebuildDHT function where +1 means rebuild after adding a member and -1 means rebuild after losing a member.

“deregister” is a command that takes no manual inputs in my application and instead uses the stored username so that users can only deregister themselves. This command is sent in the form “DeregisterUser#Username”.

“teardown-dht <leader>” is a command in the form ‘TeardownDHT#Leader’ that tells the server to tell the leader of the current DHT to delete all of its current data as well as all of its workers. The leader only deletes its own information after its workers have deleted theirs.

“teardown-complete” is sent in the form “TeardownCompleteDHT#Leader” and is an automatic command sent by the leader of the DHT after it has finished deleting all of the data stored in the DHT.

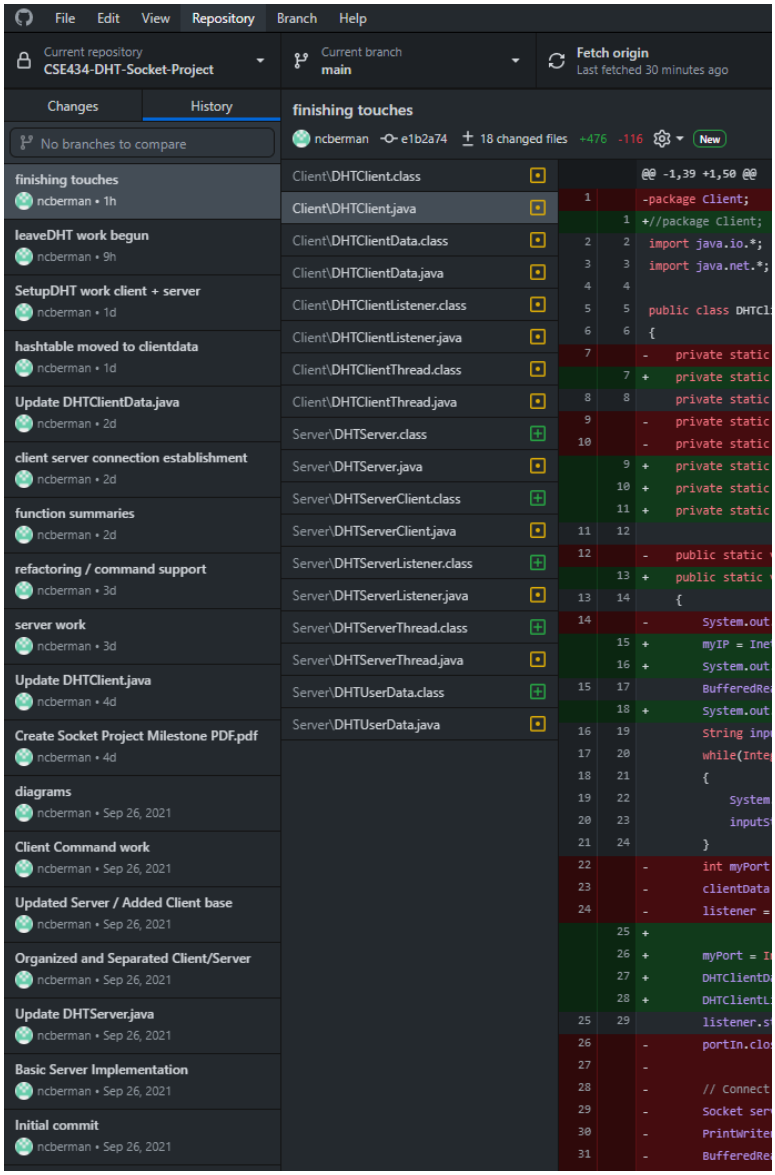
Time Space Diagrams:
Many of these diagrams are shared between similar functions, SetupDHT and Teardown are very similar as are LeaveDHT and JoinDHT.



Data Structures:

There are only a few different data structures I have implemented within my project. As it currently stands I used a convenient and sufficient method for organizing registered server members, that being a linked list. This structure does not scale very well with larger scale user bases but for the milestone I think it will be sufficient. For the contents of the list I created my own custom class to represent UserData, this structure allows you to instantiate new list members quickly and simply and then edit later using setter and getter functions. The last data structure I used is Java's Hash table structure which behaves in the same way as a typical Hash Table without having to design my own structure like I would have to in C, I just had to make the hash table a table of lists since their standard hash table structure used the non-dynamic hash table structure whereas I find it much easier to use the linked hash table.

Version Control:



Video Recording:
https://youtu.be/IfvEykxGV2s
User Registration: 2:50
Setting Up DHT: 4:55
Querying DHT: 5:50
Leaving DHT: 8:30
Re-Querying DHT: 10:10
Joining DHT: 11:35
Re-Querying DHT: 12:45
Teardown DHT: 13:55
User Deregistration: 15:00