**SYSTEM DOCUMENTATION:**

1) The Peer To Peer application consists of 3 modules: DHT\_Server, P2P\_Server and P2P\_Client. The names refer to their corresponding roles.

2) DHT\_Server Operation

When the main DHT Server (#1) is started, it reads from DHT\_Server.txt the names of all DHT Servers and stores it in an ArrayList of DHT\_Records. A Connect\_DHT thread is created to handle TCP communications with other DHT Servers. In Connect\_DHT, 3 Get\_DHT threads are created to establish and maintain TCP keep alive connections with the other 3 DHT Servers. DHT Server 1 sends “Hello” messages to the other DHT Servers, which respond with “Hi” messages.

After DHT Server operations are started, Connect\_P2P is called to establish UDP server connections to accept any requests from P2P Servers or Clients.

DHT Status Codes are used to standardize and synchronize communications with P2P Servers and Clients. (See Appendix)

3) P2P\_Server Operation

P2P Server starts by sending a GETDHT message to DHT Server #1. DHT Server #1 responds with SENDDHT message, which also contains the DHT Records of the other 3 DHT Servers.

Next, P2P Server stores the filenames of files present in its sharedDir directory. This is where the files are stored and uploaded to P2P Clients. Then it sends INFORM message to corresponding DHT Server that hash(filename) method hashes to.

Then, P2P Server establishes TCP connection to port 40411 to listen to P2P Client requests. When a download is successfully performed, P2P Server refreshes the filenames of files present in its sharedDIr directory and resends INFORM message to corresponding DHT Server that hash(filename) method hashes to.

4) P2P\_Client Operation

P2P\_Client start by sending a GETDHT message to DHT Server #1. DHT Server #1 responds with SENDDHT message, which also contains the DHT Records of the other 3 DHT Servers.

The user is then prompted to enter the filename to be downloaded. If file is not found in its sharedDir directory, it queries the DHT Server (hashed to filename) by sending REQINFO message containing the filename. The DHT Server sends INFORM message containing the (content,IP) records of containing the IP address of P2P Servers that have the file.

P2P\_Client then contacts the P2P\_Server (via TCP) by sending GET message containing the requested filename. If the file is found, the P2P\_Server sends 200 message containing the requested file. Otherwise, it can send 400, 404 or 505 messages depending on the situation. (See Appendix)

5) PseudoCode

Revised PseudoCodes for DHT\_Server, P2P\_Server and P2P\_Client are also included.

i) PSeudoCode-DHTServer.docx

ii) PSeudoCode-P2PServer.docx

iii) PSeudoCode-P2PClient.docx

**REFERENCES**

http://www.oracle.com/technetwork/java/socket-140484.html

http://www.codeproject.com/Articles/32633/Sending-Files-using-TCP

http://www.coderpanda.com/java-socket-programming-file-transfer-through-socket-in-java/

http://codereview.stackexchange.com/questions/23088/java-multithreaded-file-server-and-client-emulate-tcp-over-udp

http://www.tutorialspoint.com/java/java\_multithreading.htm

https://github.com/JohnPayment/FlamingCommAudio/blob/master/Design/P2P%20Pseudocode.txt

https://www.google.com/webhp?sourceid=chrome-instant&ion=1&espv=2&ie=UTF-8#q=java+how+to+terminate+a+thread

http://docs.oracle.com/javase/1.5.0/docs/guide/misc/threadPrimitiveDeprecation.html

http://hellotojavaworld.blogspot.ca/2010/11/runtimeaddshutdownhook.html

https://docs.oracle.com/javase/tutorial/essential/concurrency/atomic.html

http://www.programcreek.com/2009/02/java-convert-image-to-byte-array-convert-byte-array-to-image/

http://www.rgagnon.com/javadetails/java-0542.html

**APPENDIX 1**

**DHT\_Server Status Codes**

(Used to communicate between DHT Servers and (P2P Servers or Clients)

1) GETDHT – received from P2P client to query for IPs of other DHT servers.

2) SENDDHT – sent by DHT server to P2P client, contains IPs of other DHT servers.

3) REQINFO – sent by P2P client to DHT server to request records of all P2P servers that have content name.

4) SHUTDOWN – sent from DHT server to DHT server disconnects/closes all sockets and connections with P2P servers and clients

5) INFORM – received from P2P client, contains record (content name, client IP).

6) INFORM – sent by DHT server to P2P client, containing records (content, IPaddr). This is in response to REQINFO from P2P client.

7) QUIT – send by P2P Server to each DHT Server. Each DHT server will delete P2P Server’s content records.

8) CODE404 – sent from DHT server if ip address of client not found.

**P2P\_Client and P2P\_Server Status Codes**

(Used to communicate between P2P Servers and P2P Clients

1) GET – sent by P2P client to request a content file.

2) 200 – sent by P2P server if request succeeded (i.e. client IP found).

3) 400 – sent by P2P server if request message corrupted.

4) 404 – sent by P2P server if content file not found.

5) 505 – sent by P2P server if HTTP version not supported.

**APPENDIX 2**

**Message Structures**

The 1st section pertains to DHT\_ServerStatus Codes (See Appendix 1)

The 2nd section pertains to P2P\_Client and P2P\_ServerStatus Codes (See Appendix 1)

Note that there is an implicit newline “\n” at the end of every line in “Message Structure”, “Message Field Types” and “Field Values and Format” column entries.

|  |  |  |  |
| --- | --- | --- | --- |
| **Message Code** | **Message Structure** | **Message Field Types** | **Field Values and Format** |
| GETDHT | GETDHT | String |  |
| SENDDHT | SENDDHT  Server1, IPAddr,Status  Server2, IPAddr,Status  Server3, IPAddr,Status | String  String,String,String  String,String,String  String,String,String | 2,192.168.0.1,0 (where 0 = Offline, 1 = Online)  3,192.168.0.2,1  4,192.168.0.3,0 |
| INFORM  (from P2P Server to DHT Server) | INFORM  Filename | String  String | File1.jpg |
| QUIT | QUIT | String |  |
| REQINFO | REQINFO  Filename | String  String | File1.jpg |
| CODE404 | CODE404 | String |  |
| INFORM | INFORM  Filename,IPAddr  Filename,IPAddr  … | String  String,String  String,String  … | File3.jpg,192.168.0.100  File3.jpg,192.168.0.101  … |
| SHUTDOWN | SHUTDOWN | String |  |
|  |  |  |  |
| **Message Code** | **Message Structure** | **Message Field Types** | **Field Values and Format** |
| GET | HTTP1.1  GET  Filename | String  String  String | File1.jpg |
| 200 | HTTP1.1  200  filename  filecontent (in Unicode bytes) | String  String  String  String | File2.jpg  bca345yuujko34$%\*JM… |
| 400 | HTTP1.1  400 | String  String |  |
| 404 | HTTP1.1  404 | String  String |  |
| 505 | HTTP1.1  505 | String  String |  |