TOPIC: PEER TO PEER FILE TRANSFER

Functionality—(Peer)

A peer can add files, download files and choose to disconnect from the network.

1) Add files:

>When a peer chooses to add files, he is shown with a list of files that exist in his directory and he can choose which of the files he wishes to add, thereby making those files available for other peers to download.

>When he types in a filename, it is validated; (i.e. if an incorrect file name is entered or if the file has already been added an appropriate message is generated asking the peer to renter correct filename.)

2) Download file:

>When a peer chooses to download a file, he is displayed with a list of peers who have the file and are connected to the server.

> He can choose one of the peers (say peer j) from whom he wants to download the file. Upon clicking ok, (the server sends the IP address and port number of peer j and connection is established between the two peers). Thus the file is transferred and placed in the peer’s directory and the array list that contains list of available files and PeerFileInfo array lists are updated.

> Here, peer to peer communication takes place, the peer(j) who send the file acts as server. Both the peers communicate using the peer(j’s) port and a different socket. This enabling multiple peers to transfer or ask for files simultaneously.

3) Exit:

>When a peer chooses to exit, he gets disconnected from the network and all his information is automatically removed from the array lists. This is handled using functions like removepeerfileinfo, removepeerinfo.

>Two or more peers can have same file when one of the peers disconnect, only those files specific and unique to him are deleted from the array list.

> A message is sent to server saying peer has exited from the network.

Functionality—(Server)

* The server just waits for requests from multiple clients to connect to it; each of the connection is handled as a separate thread.
* All the previously mentioned array lists are updated.
* The server maintains information about all the clients and sends them to clients when necessary.
* When a peer(i) chooses to download a file from a particular peer(j) , per(j’s) port number and IP address are sent to peer(i).
* On pressing q the server switches off, that is server quits, this is handled as a separate thread, that just waits for user input; once q is entered the server stops.

JAVA FEATURES EXPLOITED:

>> Socket programming:

>> Java GUI:

> Java Swing

> Java AWT

>>OOP Concepts:

>Inheritance-

>Polymorphism:

-> Overriding -

-> Overloading-

>Aggregation

>>Inner Classes:

>>Java Collections: Java Collection framework provides ArrayList Class. In this project three different ArrayLists are used to maintain the list of added files, list of which peer has which file, and list of peers.

>>Multi-Threading: