Name: Pavithra Rathinasabapathy

Student ID: 1001698736

**Project IDE:** Eclipse IDE for Java Developers - 2019-12

**Steps for setting up a project:**

* Create a new java project in Eclipse IDE
* Create a new package under src folder with the name “project1DS”
* Copy the files ClientDS.java and ServerDS.java to the project1DS folder
* Run the ServerDS.java as java Application
* Run the ClientDS.java as java application and enter the name to get a connection
* Similarly run this file for next 2 clients in the same way.
* The randomly selected client would receive random number and thread is paused.
* The timer Label in GUI shows the countdown
* Press the “CANCEL” button on the Client GUI to stop waiting
* Press the “QUIT button to manually kill the client.
* The text Area in Client GUI displays all the messages from Server.
* The text area in Server GUI displays all the messages from Client.

Message sent from Client to Server:

1. “DONE” – client thread is finished with waiting
2. “IGNORED” – client thread cancelled waiting. Didn’t wait
3. “QUIT” – User manually killed the client. Close all the connection

Messages from Server to Client:

1. “AVAILABLE” – The user name is available and connection is made to server
2. “NOT AVAILABLE” – The user name is not available and clients rejects the connection. The client GUI is disposed and have to run the client again
3. “PAUSE” - Pause command from server. Expect a random number.

**Assumption** - Server selects client randomly every 10 seconds. Even if the selected clients didn’t pause/wait or wait for less than 10 seconds (e.g., 3 sec). Upon receiving the reply from client, server will wait for that remaining seconds to randomly select next client.

**Project Reference-**

1. https://stackoverflow.com/questions/15247752/gui-client-server-in-java
2. https://stackoverflow.com/questions/22728794/how-to-put-timer-into-a-gui
3. https://www.geeksforgeeks.org/introducing-threads-socket-programming-java/
4. https://www.geeksforgeeks.org/socket-programming-in-java/
5. https://stackoverflow.com/questions/41506997/java-multithreading-synchronizedthis-on-a-thread-class