CS 3873: Net-Centric Computing
Lab 2: Socket Programming

Student Name: Mahmoud Moustafa Student Number: 3648276

[Mandatory] Declaration: “I warrant that this is my own work.”

Signed by Mahmoud Moustafa

[Optional] “I hereby give my permission for this work to be used (with my name and identifying information removed) for UNB Faculty of Computer Science program accreditation purposes.”

Signed by _____

Report for Lab Exercise 2:
Socket Programming

LAB ACTIVITIES:

In this lab, a client and a server are written in Java to implement simple file transfer functions with sockets.

PROGRAM RUNNING INSTRUCTIONS:

Please compile and run the uploaded Java programs using the following commands:

Compile using “javac TCPFile*.java”

Run the server first “java TCPFileServer”

Finally run the client “java TCPFileClient [Hostname]” In my case was java TCPFileClient id415m21

APPENDIX A: CLIENT

The following is the Java source code for the client:

```
import java.io.*;
import java.net.*;
import java.util.Scanner;
class TCPFileClient{
  public static void main(String[] args) throws Exception {
    String sentence;
    String modifiedSentence;
    File file = new File("PoemShakespeare.txt");
    Scanner inFromUser = new Scanner (file);
    Socket clientSocket = new Socket(args[0], 6789);
    DataOutputStream outToServer = new DataOutputStream(
    clientSocket.getOutputStream());
    BufferedReader inFromServer = new BufferedReader(new InputStreamReader(
```

```
   clientSocket.getInputStream()));
   int x = 0;
   long startTime = 0, endTime;
   while(inFromUser.hasNextLine()){
    sentence = inFromUser.nextLine();
    if (x == 0){
     startTime = System.currentTimeMillis();
     x++;
    }

    outToServer.writeBytes(sentence + '\n');

    modifiedSentence = inFromServer.readLine();
    System.out.println("FROM SERVER: " + modifiedSentence);
   }
   outToServer.writeBytes("Done" + '\n');
   modifiedSentence = inFromServer.readLine();
   System.out.println("FROM SERVER: " + modifiedSentence);

   if(modifiedSentence.equalsIgnoreCase("Bye")){
    endTime = System.currentTimeMillis();
    clientSocket.close();
    System.out.println("Number of data sent in bytes: " + outToServer.size());
    System.out.println("Total time: " + (endTime - startTime) + "ms");
   }


  }
}
```

APPENDIX B: SERVER

The following is the Java source code for the server:

```
import java.io.*;
import java.net.*;

class TCPFileServer {
  public static void main(String args[]) throws Exception {

    String capitalizedSentence;
    ServerSocket welcomeSocket = new ServerSocket(6789);
    System.out.println ("Waiting for connection.....");
    Socket connectionSocket = welcomeSocket.accept();
    BufferedReader inFromClient = new BufferedReader(
    new InputStreamReader(connectionSocket.getInputStream()));
    DataOutputStream outToClient = new DataOutputStream(
    connectionSocket.getOutputStream());
    String clientSentence = inFromClient.readLine();

    while (clientSentence != null) {

     System.out.println("From client at " + connectionSocket.getInetAddress()
     + ": " + clientSentence);
     // System.out.println(clientSentence);
     if (clientSentence.equalsIgnoreCase("Done")){
       outToClient.writeBytes("Bye"+'\n');
     }
     else{
       capitalizedSentence = clientSentence.toUpperCase() + '\n';
       outToClient.writeBytes(capitalizedSentence);
     }


     clientSentence = inFromClient.readLine();
    }
```

```
  }
}
```