**BRUNEL UNIVERSITY**

**SCHOOL OF ENGINEERING AND DESIGN**

**ELECTRONICS & COMPUTER ENGINEERING**

**Modules: EE5506**

**DATA COMMUNICATION SYSTENS WORKSHOP 2008-2009**

**Evaluating UDP Performance in File Transfer**

**SUBMITTED BY:**

**SEYED KAMRAN PEDRAM RAD**

**ID: 0828678**

**INDEX**

INTRODUTION 3

UDP 3

UDP Socket 6

TASK 1 10

TASK 2 16

CONCULOTION 17

REFERENCES 18

**Aim**

In this assignment we used eclipse programs to take UDP socket in programming and also evaluate UDP performance in file transfer while we put some different packet size for transfer. So first of all for describe how UDP Socket works we introduce UDP and UDP Socket.

**User Datagram Protocol (UDP):**

There are two distinct transport layer protocols to the application layer in the Internet**.**

UDP (User Datagram Protocol) is one of these protocols, which makes an unreliable connectionless support to takeapplication.

In comparison TCP (Transmission Control Protocol), the second of these protocols, this provides a reliable, connection-oriented support to take application.

We have to use one of these two transport protocols for design of a network application.

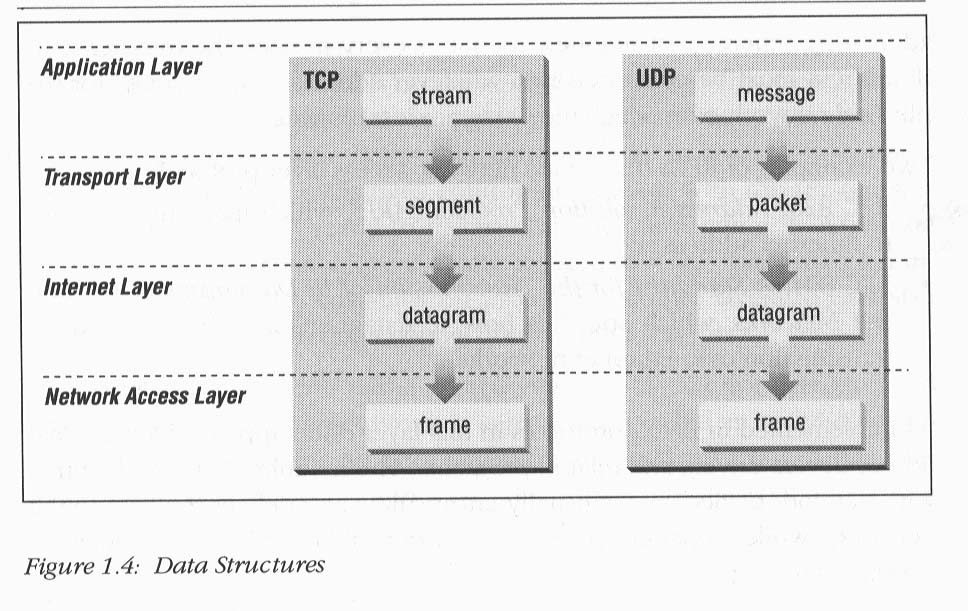
In an internet we have the transport layer packet for TCP as a segment but usually refer to the packet for UDP as a datagram. Although in this case we refer to both TCP and UDP packets as segments, and for the network layer packet have the datagram. A specific port is identified by its number commonly known as the port number. 

Figure 1

We know in process, on the sending side, we have to consider taking the data from application process and passing them straight to the network layer, and on the receiving side we have this process again but vice versa.

UDP, does only as little as a transport protocol can do. It just multiplexing / de multiplexing function and a few light errors checking without add anything to IP. It means when we use UDP, then the application is almost directly talking with IP. UDP carry massages from the application process, attaches source and destination port number part for the multiplexing/de multiplexing service, take two other small fields, and passes the resulting segment to the network layer. The network layer passed the segment to the receiving host. If the segment arrives at the receiving host, UDP uses the destination port number to take the segments message to the correct application process.

The service created by UDP is an unreliable service, which has no guarantee for delivery and no protection from duplication. Almost the client makes up the source port number to a unique number which usually are based on the program that stared connection. When the server in response resend this number it permit to sender to know which conversation packet is to be sent to. The destination of packets that client sent it is make up to a number of well known ports, which usually correspond to a number of different applications.

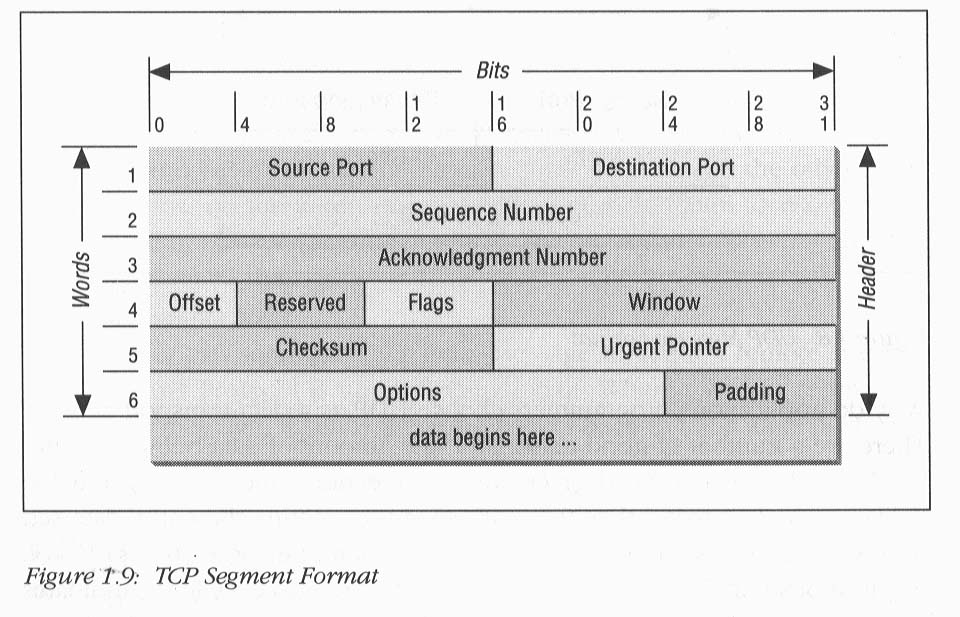


Figure2

The program of server for UDP packets listens by a well-known port number for UDP packet, and tells to local UDP layer to send the suitable packet to server program, it indicate these pack come from which client by examining the receiving IP source address and receive unique. The server needs to resend some response to client which is sent by source port. The client defines the destination port. We can see these in table 1.

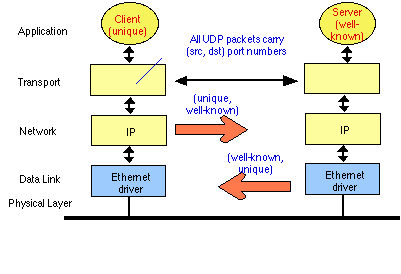


Table1

The network application that needs to transport data between computers is supported by UDP. Application that is supported by UDP is client and server. After many year that people use UDP it is interesting technology to use yet. UDP is liked TCP-on top of the IP. As we can see from table 2, UDP and also TCP are transport layer protocols.

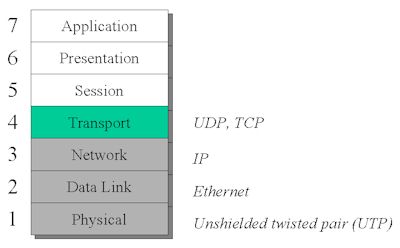


Table 2

When data moves in the network, different attribute is added to a file to provide a frame. This mechanism is called encapsulation. The protocol which is used is important to use of the methods of encapsulation. In the image below we can see the structure of UDP.

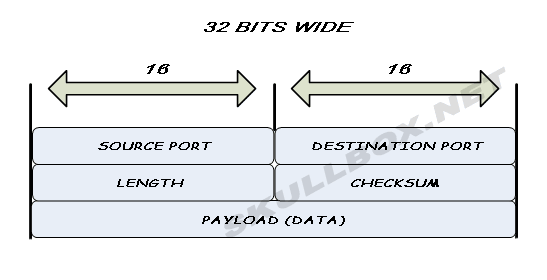
**

Table 3

UDP is popular to use for streaming audio and video, and not used for transfer the important data such as WebPages; because of not correcting the error it offers the higher speed. There is no high quality in UDP and just higher speed rather than TCP.

**UDP Sockets**

One of the Internet socket is a Datagram Socket. It is sending or receiving point for packet delivery services. On a Datagram socket, every packet which is sent or receives has an address and rout. Multiple packets which are between one machine and other machine may arrive in any order and may not arrive at a receiving machine.

User Datagram Protocol broadcasts sends are enable on a Datagram Socket. For receiving the broadcast packets, a Datagram socket should provided by Wildcard address. Also when a Datagram Socket is bound to a more address, Broadcasted packet may be received.

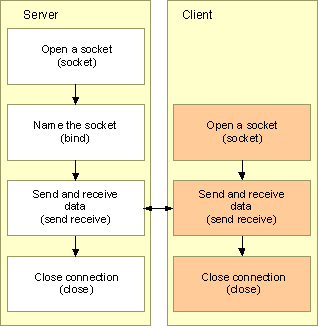


Table 4

Creating a UDP datagram Socket Server needs some works; first of all with Socket we should open a Datagram socket. A UDP Datagram needs to create a socket and bind it to a name, to preparing for association with client. Then with the bind function, name the socket, use SOCKADDR\_IN for the address parameter. After that by using the Sendto-is used in the connectionless socket- and recvfrom-the UDP Datagram socket server- function, exchange data with client. Finally close the connection with the close socket function. Shutdown function is not necessary for UDP socket.

To create a UDP datagram Socket **client**,first we open the socket with socket function, then by using Sendto, and recvfrom exchange data with server. In the end close the connection with the close socket function.

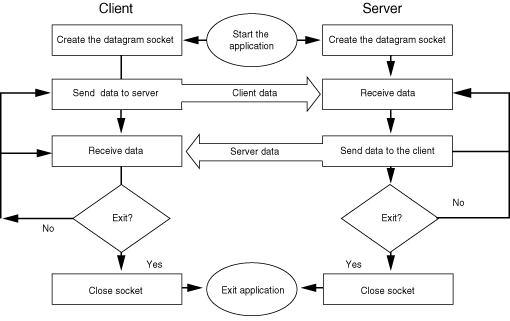


Figure 3

Datagram Protocol is an unreliable protocol which does not have software mechanism for retrying on transmission failures, and it has limitation on message length, (it should be under 65536 bytes). For applications which use broadcasting or multicasting, it is used. It can underline the important applications as the Network File system, the domain name system, the Simple Network management Protocol, and other noted by Stevens.

A program for communicate with some (more than one) host and port number, can use just one UDP socket. So we can also call the User Datagram Protocol (UDP), the connectionless protocols. Although it is better to most of the User Datagram Protocol client programs, by keeping a local record of each host and port number, saying there is a connection.

UDP servers do not create new child processes for every served client, the UDP socket identified by the local address.

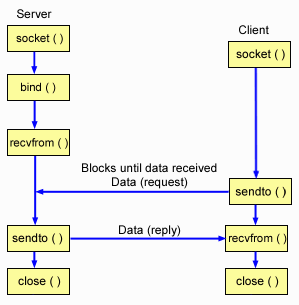


Figure 4

**Task 1**

First of all we made a new project in eclipse and we put an UDPGui name for that.

Then we take four classes in this project for run the program.

We started to write a program for UDP Client as can be seen in figure 5.

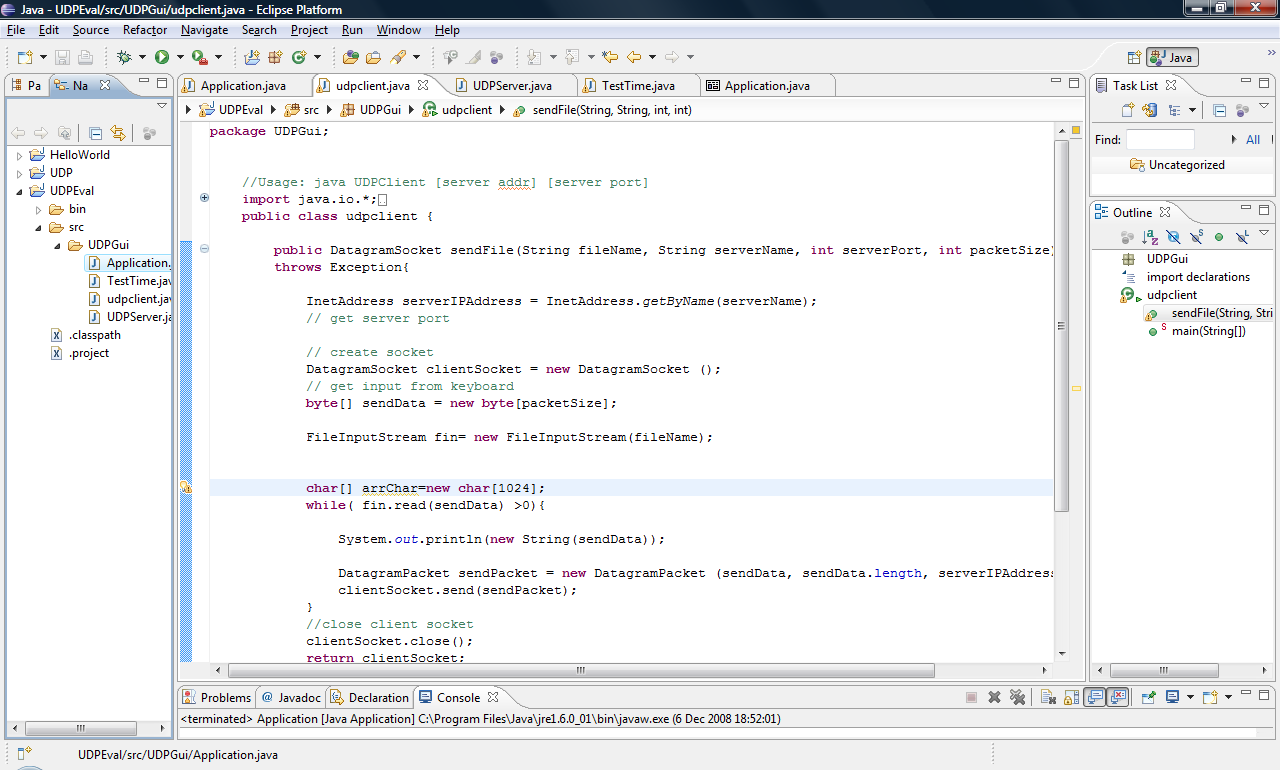


Figure 5

After we checked the error in UDP Client if it was any to resolve it, we wrote a UDP Server in eclipse in new class. See the figure below 6.

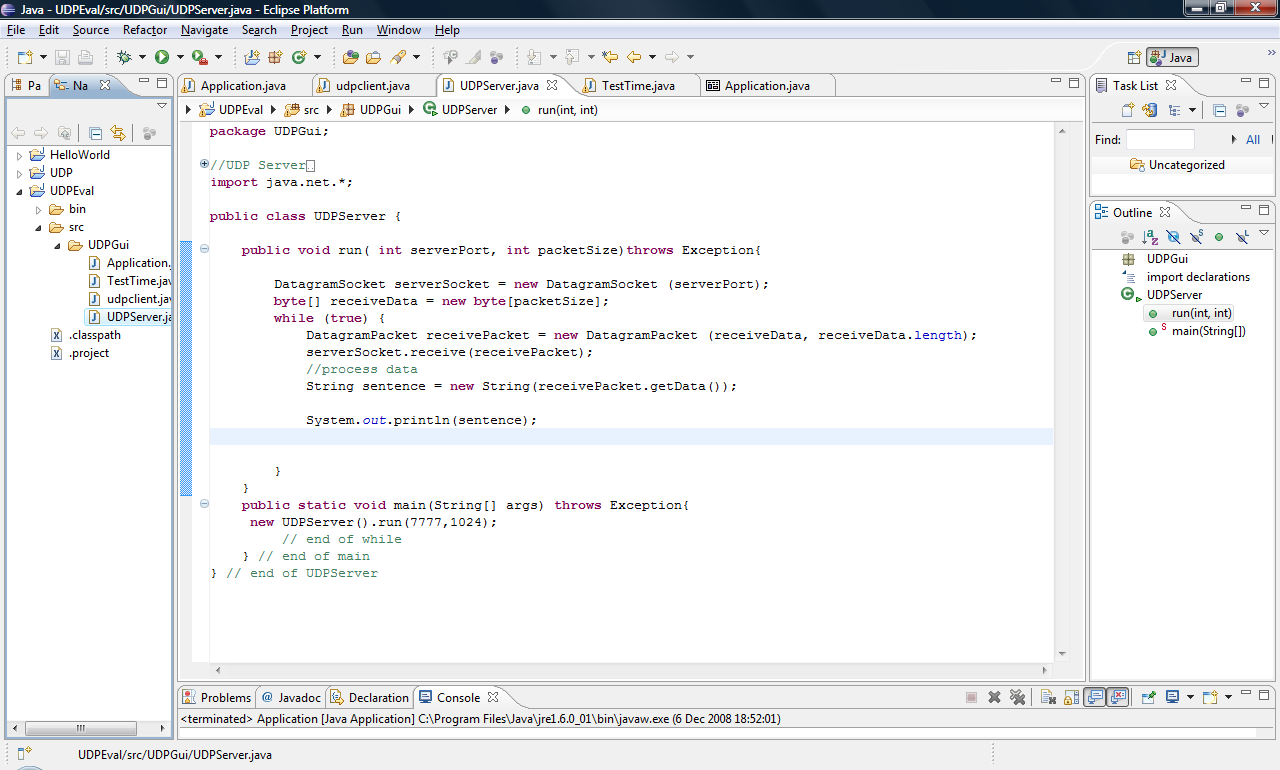


Figure 6

Then we made a Client user inter face and a Server user interface as we show in figure 7 and 8.

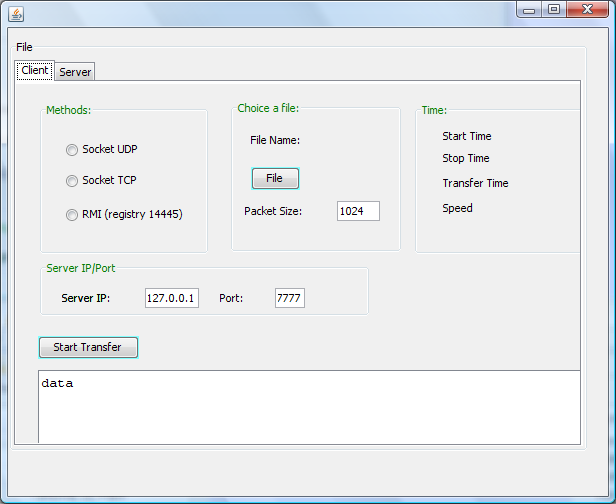


Figure 7.The Client User interface

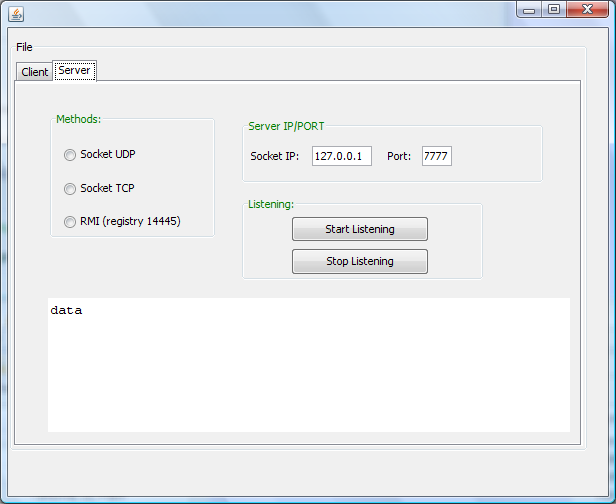


Figure 8.The Server User interface.

So now we can choose a file for transfer

When we put click in a browse we saw the figure as be seen in figure 9 below.

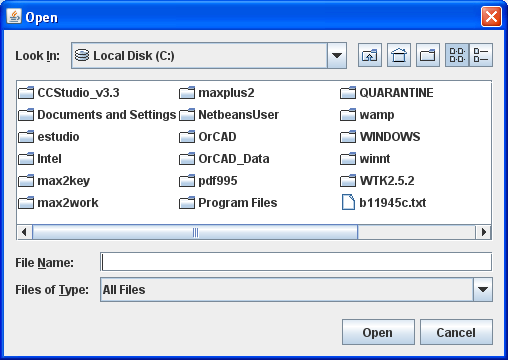


Figure 9

Then we choose anything which we want for transfer, after that we click in start listening in Server interface then we click in start in client to start the transfer and then stop listening. After few second we can see the result as can be seen in figure 10.

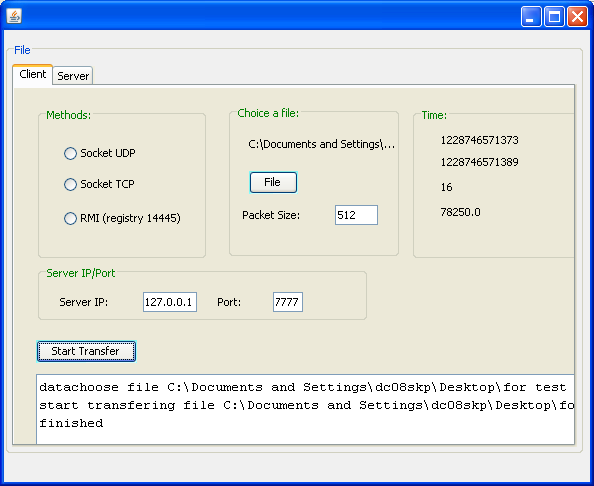


Figure10.

As can be seen the Transfer Time is 16second with speed: 78250 for 512B packet size.

Now we can change the packet size to 1K and see the result, as can be seen in figure 11.

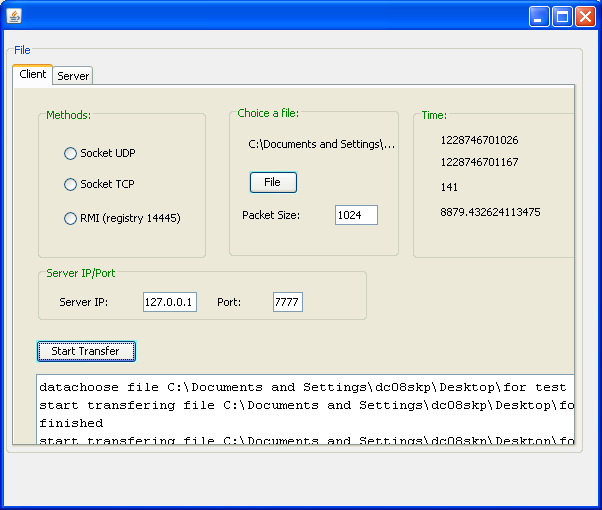


Figure11.

The Transfer Time is 141 second with speed: 8879 for 1KB.

And finally we do it again for 2KB packet size to see the change.

It shows in figure 12.

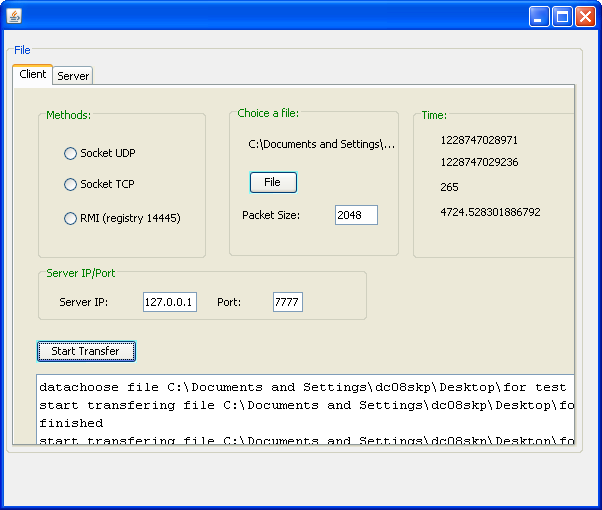


Figure 12

The Transfer Time is 256 second with speed: 4724 for 2KB.

So we see the different Time and speed for 3 different packet size it means when we put a bigger packet size the transfer time is take longer and speed is higher.

During the process in the main page of java we saw the information about what we did in server and client interface which shows in figure 13.

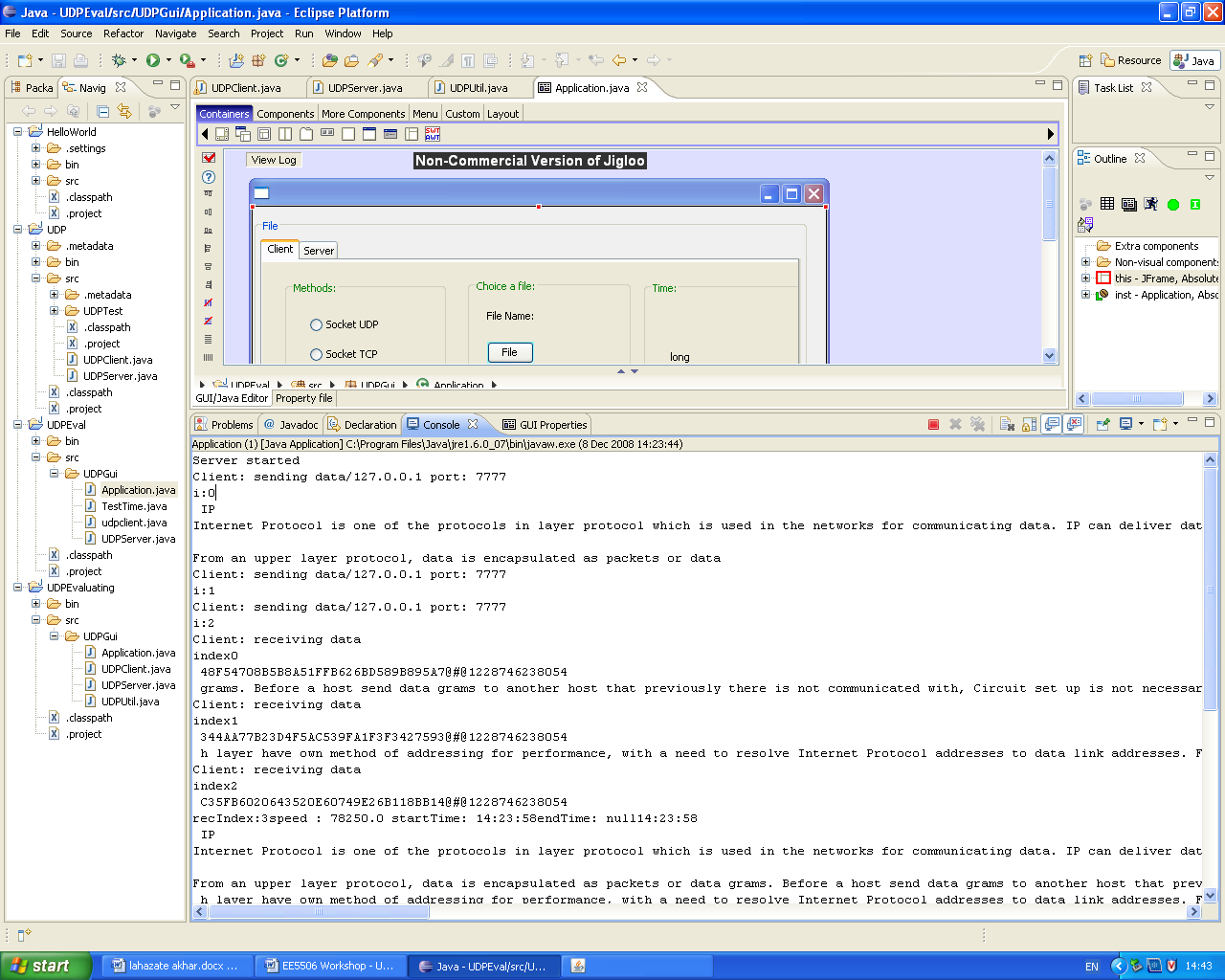


Figure 13

**Task 2**

For this task we used 2 PC two connect them with each other which one of them for Client code and another is for Server code to see the figure that how it look like when we change the packet size and what is the delay in them. See the figure 14. The delays in the column and packet size in row.

Figure 14

As can be seen clearly when we give the larger packet size we see less delay for transfer the dada because we have a bigger size of packet for transfer which was forecast able.

**Conclusion**

In this assignment we used eclipse for evaluating UDP performance in file transfer which we used different packet size.

First we made the UDP Client and UDP Server then we made the application, Client and Server interface.

After we checked the errors we run the program then choose the file for transfer and we click in start listening in server interface then start in client to see the result in it. We did this test for three different packet sizes 512B, 1024KB and 2048KB respectively. We saw the time transfer and speed which when we put the bigger packet size we saw a higher speed with longer transfer time.

Finally we did in two PC for connect between client and server to see the delay between different packet sizes. It was clear when we have larger packet size we have less delay for them. As we know in UDP the important things is speed so it has the very good speed with a little delay.

**References:**

Figure1 & 2: Cs.fsu (Internet website)

Table 1, 2 & 3: Wikipedia (Internet website)

Table 4: Msdn.microsoft (Internet website)

Figure 3: Ssuet (Internet website)

Figure 4: publib.boulder (Internet website)