**Assignment 2- Report**

**Introduction:**

In this program, the inter-process communication and end-to-end argument are achieved using socket programming. Sockets make it possible to distribute work to the computer that is most efficient, exchange information between processes running on the same machine or across a network, and access centralized data with ease. The program has a client and a server who communicate with each other for File Transfer. The user must be authenticated before being able to request files from the server. The file transfer includes encryption and decryption of files for security. The contents of the source and destination must be the same.

**Development:**

To send or request info from one computer to another over the network, we need to have information on its IP Address and Port. This will be helpful to keep track of where the data is going. We have two different java classes running on two different machines. One for the client and one for the server. Using socket programming the connection between the client and server is established.

The user must enter their username and password for authentication. If the user is already registered, then they get logged in directly. If the user is new, then they can get registered before logging in. If the user gives an incorrect password, the program terminates here while indicating so to the user.

Once the authentication is done, the user requests the file from the server. The server checks whether the file exists or not. If the file exists, it encrypts that file and sends it to the client in the form of a byte array. If the file does not exist, the server sends a message to the client indicating that no such file exists.

The method used here for encryption is Caesar cipher, which is a monoalphabetic substitution method. Here each character is replaced by another character with some fixed number of positions, which is the key. For example, If the key is 3 and my string is “apple” then after encryption the text is replaced by “dssoh”.

The encrypted file is decrypted at the client end using the same key and the contents of the file are stored in a new file with the random file name “OutputNN” where NN is any random integer between 1-200. The client checks if the original file and the transferred file contents are the same or not.

Hence the file transfer between the client and server is successful and the connection is closed.

**Pros:**

* User needs to be authenticated for requesting any file from the server.
* Socket programming made it easy for communication between the client and server.
* Sockets are used
* Caesar Cipher algorithm for encryption.
* Encryption and Decryption of files help provide data security for sensitive information.

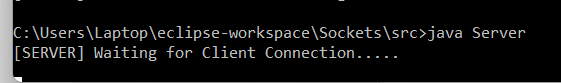
**Cons:**

* One client and one server model.
* Both ends should have the ability to intercept the data.

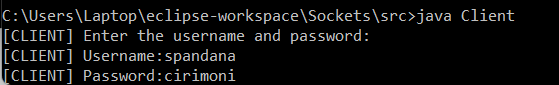
**Test Case:**

Here’s a small test case on the program.

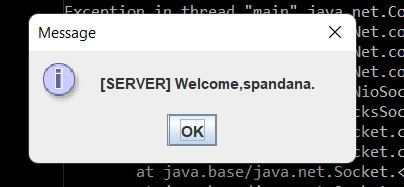
After running the server class:

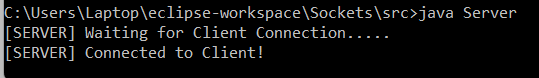


After running the client class: (Providing an existing username and password)

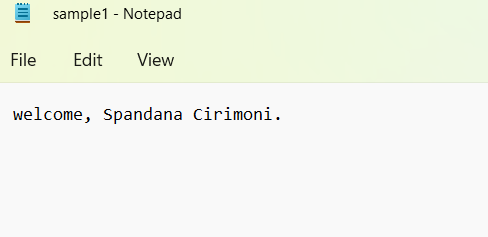


Message:

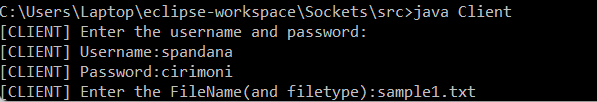


The connection is established:  


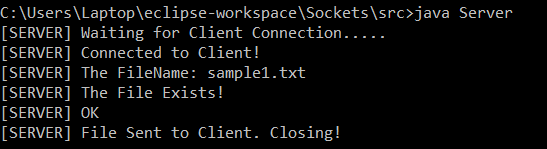
The sample1 text file:



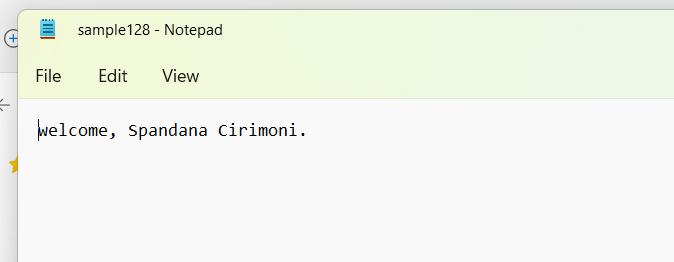
The client requests sample1.txt



The server checks if the file exists and sends the file to the client.



The randomly generated OutputNN file contains :



The client indicates that it has received the file.  
