Network Project Report

Design a Transport Protocol

Group6

Zhe Xing, NetId:zxx151430

Chun Hui Lin, NetId:cxl153930

Chen Wei Liu, NetId:cxl153830

**1. The structure of project**

1).We create connection between Client side and Server side.

2).Client side sends “sync” message to Server side to request Dir from server.

3).Server side creates Dir and sends Dir to Client side.

4).Client side creates Dir and compares two Dirs.

5).Creates addfile.txt and deletefile.txt.

6).Client side send files that server side does not have to server side according to addfile.txt.

7).Client side sends deletefile.txt to server side.

8).Server side receive deletefile.txt and server deletes unnecessary files.

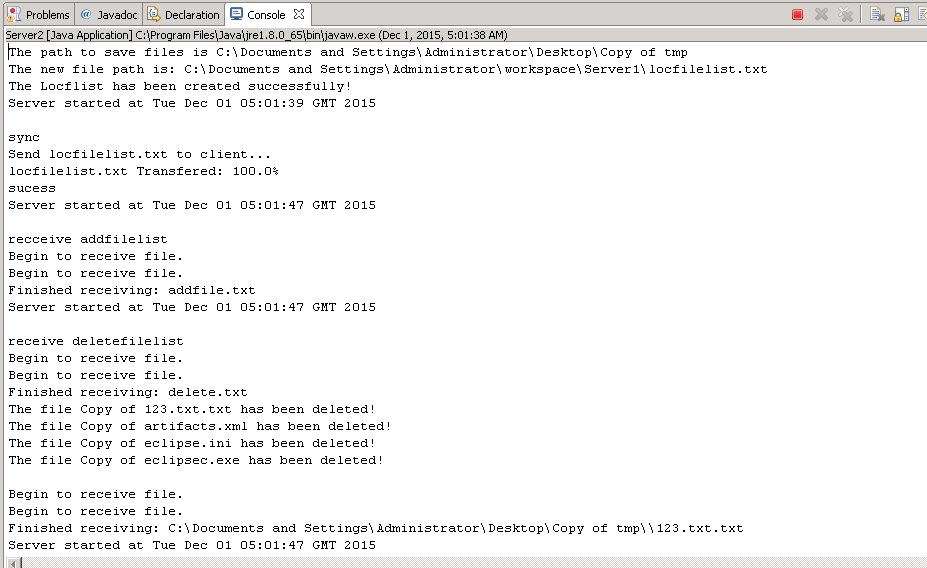
**2. TCP-socket**

We choose to use Socket and SocketServer Class to create connection and to send files.

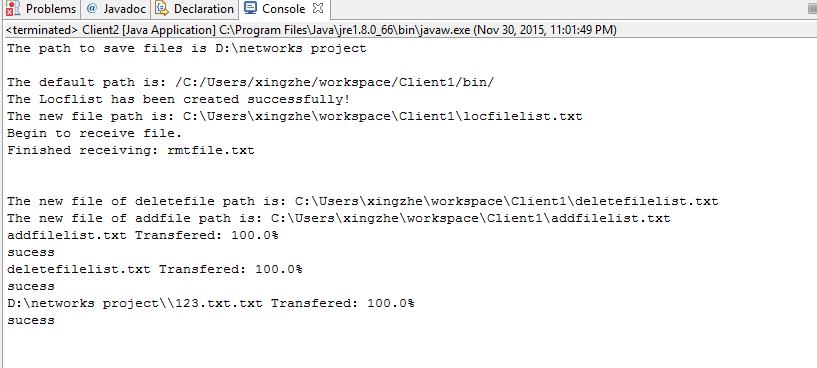
These two class are in TCP. The TCP has higher security than UDP. So we choose TCP-Socket.

**3. The result**

Server side:



Client side:



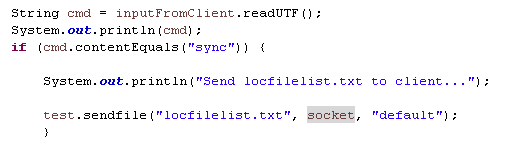
**4. What I did (Zhe Xing)**

Because this project needs three nodes laptop, cloud and Android phone, I set up Amazon EC2 cloud environment as cloud node and Android studio as Android phone node because of we do not have real Android phone.

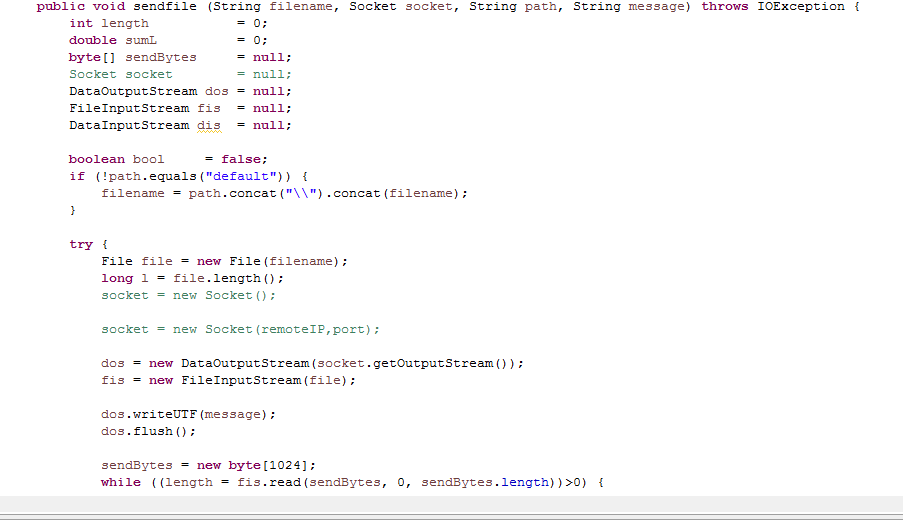
I write all the codes from server side (cloud) and I set up Android studio environment and I finish a large part of codes from client side.

1. Send “sync” message to Server side and server side creates dir when it receive “sync” message.

1):



In this section, I write a method called sendfile in Filesnew2 class.





This method has four parameters: filename, socket, path, message.

filename: the name of file to send.

socket: the socket I set up in main().

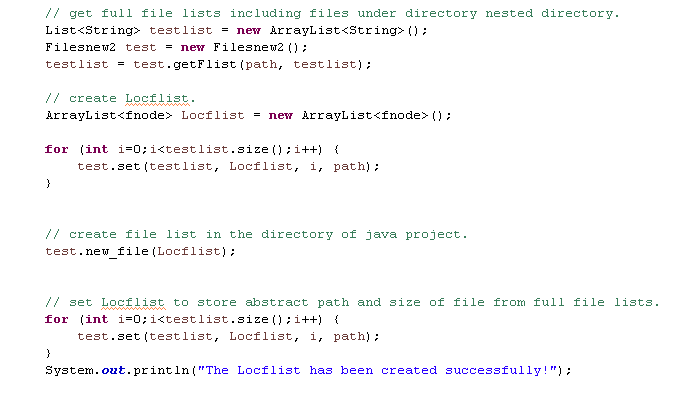
Path: the path of file.



The file may be saved in two places: the path of dir and workspace-the “default”. If the path is not default path, I will use these codes to transfer file’s path to absolute path.

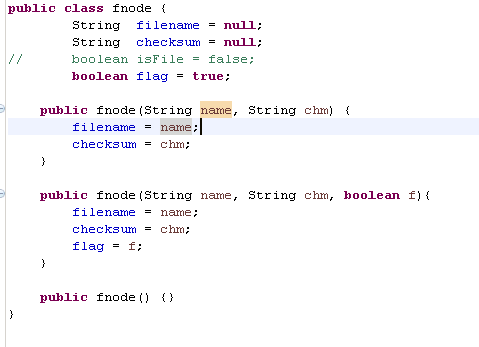
Message: using to represent the style of file to send. Because in this project, I will use several times to call this method.

2):creates dir in server side.

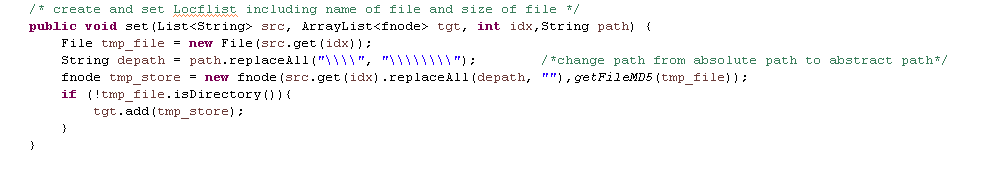


To create dir, I need several steps.

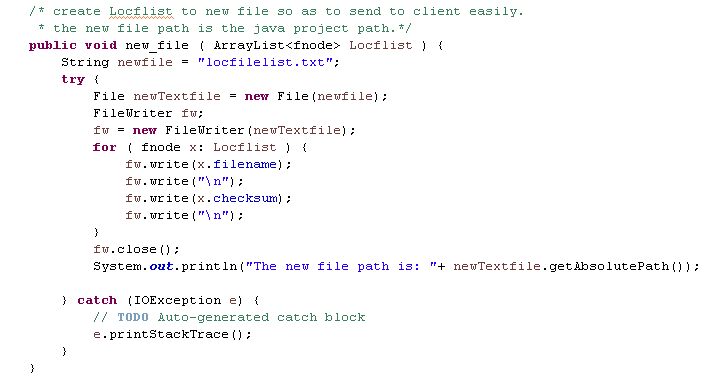
Firstly, I customize array fnode. In this class, each node has three parameters: filename, checksum and flag



Second, I customize a method called set(). This method is to transfer List<String> to ArrayList<fnode>. Because, the list of string just contains one parameter-filename., after using this method, each node has three parameters. Subsequently, we will use these features to compare dirs between client and server.

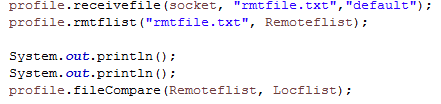


Thirdly, I customize a method called new\_file(). I use this method to transfer list to a file that will be sent to client side.



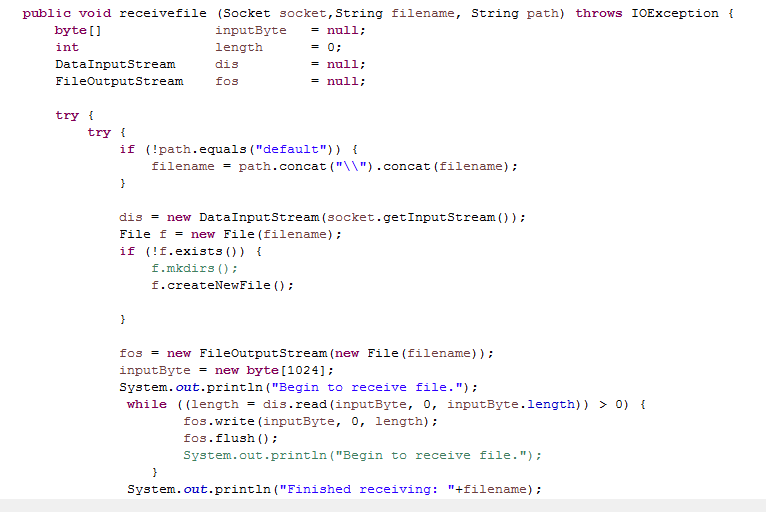
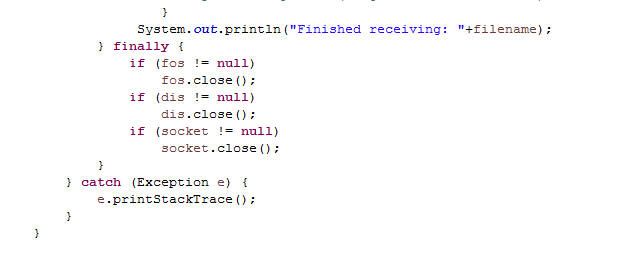
After called these methods, I have server’s directory. And in client side, I will call the same methods to create client’s directory.

1. Client side receives server’s directory.

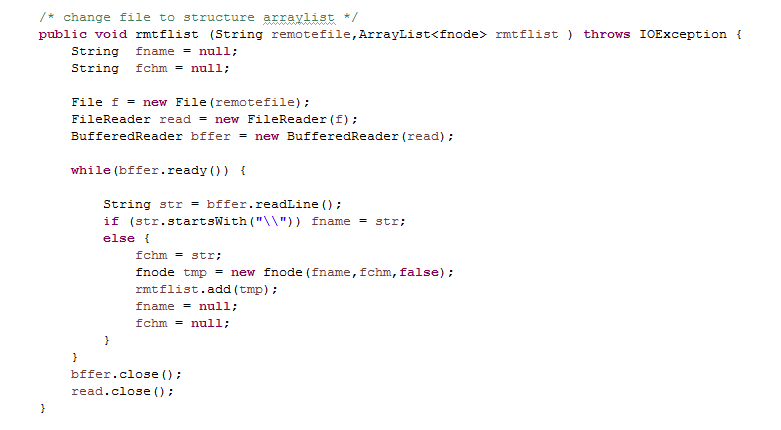


In this section, after receiving directory of server, I will compare two directories from server and client.

1): I write a method called receivefile(). This method is to receive files from opposite side.

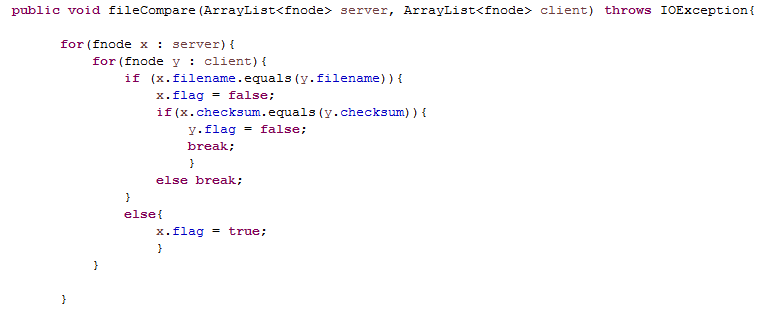
 

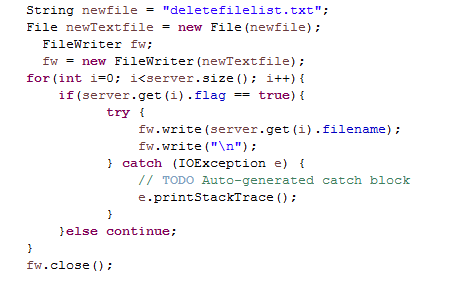
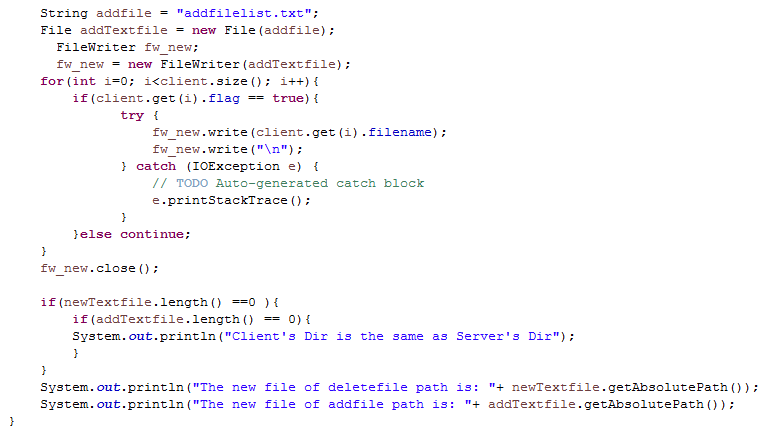
2): After receiving dir from server, I will call rmtflist method that is used to change file to Arraylist<fnode>. Because I use filename and checksum to compare two directories.



I call FileReader and BufferedReader to read each line in file and I will get two parameters filename and checksum.

3): Last, I customize filecompare method to compare two directories. At the same time, it will have two file deletefilelist and addfilelist.



In this method, I used an algorithm to compare two directories.

First step, I set flag from client to 1 and from server to 0.

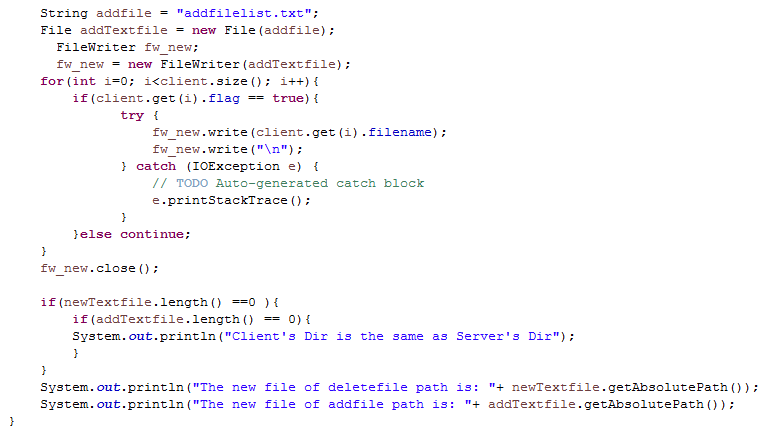
Then, I take one node from server to compare each node from client.

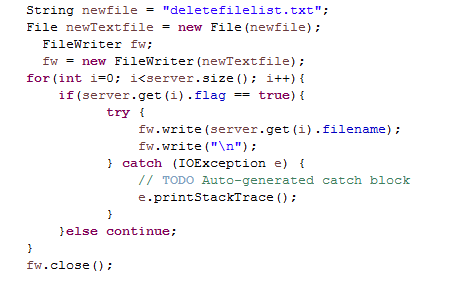
Then, if the filename of node from server side is difference from client side. I will set the flag from server to 1.

If the filename and ckecksum of node from server is the same as client. I will set the flag from client to 0.

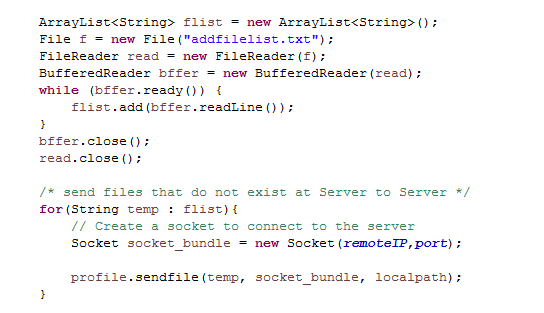
The other nodes’ flag do not change.

According the node flag “1” from client, these node will create a file called addfilelist.



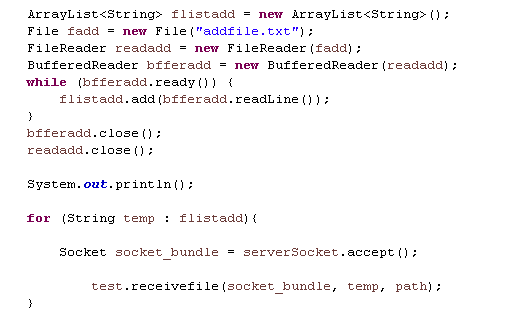
According the node flag “1” from server, these node will create a file called deletefilelist. 

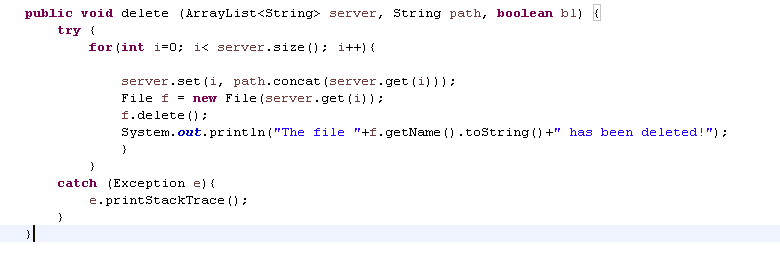
1. Client side send files that do not exist in server side to server side.



It will send files to server side according to addfilelist.txt.

I will also call sendfile method to send files.

And server side will use receive method to receive files. 

1. Server side deletes files that do not exist in client side. I write a method called delete. 
2. I set up Android Studio.

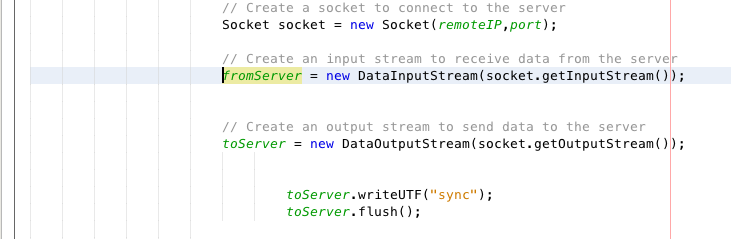
The codes from eclipse can also run in Android Studio. But I need to export code from eclipse. I need a software called ADT(Android Developer Tools). Using this software, the codes from eclipse can run in Android Studio.

**5. What I did(Chun Hui Lin/Che Wei Liu)**

1. Socket

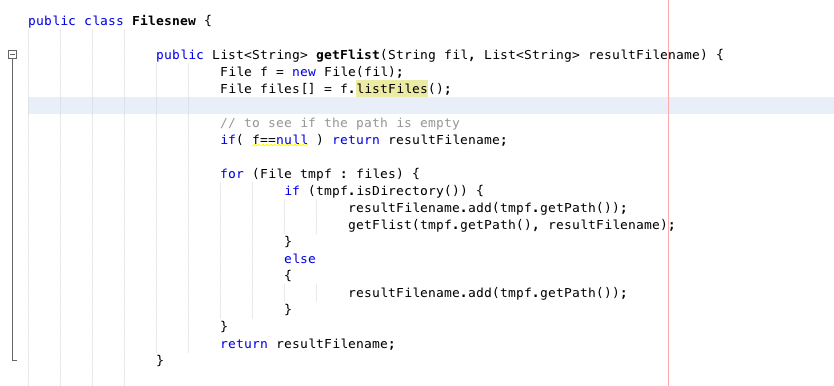
On the server side, it has a socket that is bound to a port number. The server will wait and listen to the socket for a client to make a connection request.

On the client side, client also has a local port number that will use during connection. When server accepts the connection, the server gets a new socket bound to the same local port and also has its remote endpoint set to the address and port of the client.



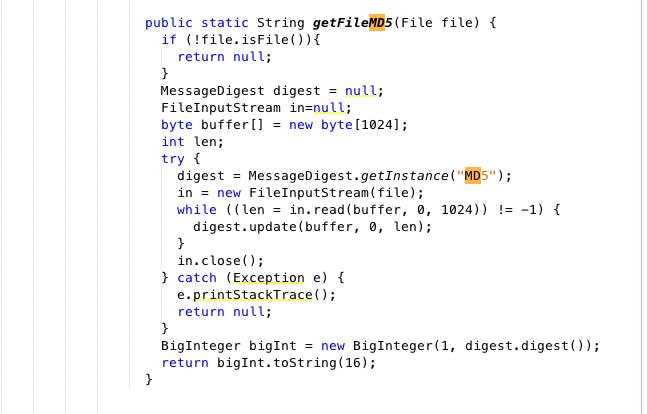
2. File list

To list all the files in the server side and the client side. Use recursive to count all the files.



3. Checksum

Use MD5 to produce checksum then we can use it to compare files.



4. fnode

In this code area, we utilize the fnode which function is to separate and analyze a file. A file must contain a filename and a checksum. We can employ the MD5 to assist us to peruse the checksum of every file. We first limpidly isolate the filename and checksum to two list; accordingly, we can differentiate the filename and checksum of the both side, client and server. After the comparison, the following programming can do the response.

