No	Feature	Status	Keterangan
1	Chat	Works	Works well using <Person x> like in the question
2	List	Works	Works well. Sending the file list only to the client that requested
3	Upload	Doesn’t work	The code stops when trying to read the file that’s being sent
4	Downzip	Blank	Not enough time
5	Log	Didn’t finish	Code stops for some reason when trying to save the chat by appending to a variable

Changes made to server.py

```
message = conn.recv(2048).decode()
            currId = getId(conn)
            splitmsg = message.split(' ', 1)
            if message[:-1].lower() == 'list':
                print('<Person ' + currId + '> requested file list')
                # chatlog.append('<Person ' + currId + '> requested 
file list\n')
                fileList = listdir("./chatstorage")
                for files in fileList:
                    files = str(files) + '\n'
                    conn.send(files.encode())
            elif splitmsg[0].lower() == 'upload':
            # Creating a new file at server end and writing the data
                filename = splitmsg[1]
                filename = filename[:-1]
                fo = open("./chatstorage/" + filename, "w")
                data = conn.recv(1024).decode()
                if not data:
                    continue
                fo.write("balls")
                while data:
                    if not data:
                        break
                    else:
                        print("asdflsdjflsdkfj")
                        fo.write("a")
                        print("bruh")
                        data = conn.recv(1024).decode()
                        print("lmao")

                print()
                print('Receiving file from client ', currId)
                print()
                print('Received successfully! New filename is:', filename)
                fo.close()
            elif message[:-1].lower() == 'log':
                pass
            elif message[:-1] == 'downzip':
                pass
            elif message:
                print('<Person ' + currId + '>' + message)
                message_to_send = '<Person ' + currId + '>' + message
                broadcast(message_to_send, conn)
```

```
def getId(conn):
    clientcounter = 1
    for clients in list_of_clients:
        if clients == conn:
            return str(clientcounter)
        else:
            clientcounter += 1
```

Changes made to client.py

```
if splitData[0].lower() == "upload":
            filename = splitData[1]
            # read file and send
            try:
                fi = open(filename[:-1], "r")
                data = fi.read()
                if not data:
                    break
                while data:
                    sock.send(str(data).encode())
                    data = fi.read()
                # File is closed after data is sent
                fi.close()

            except IOError:
                print('You entered an invalid filename!\nPlease enter a valid name')
```