

Class TCPServer

```
java.lang.Object
  java.lang.Thread
    TCPServer
```

All Implemented Interfaces:

Runnable

```
public class TCPServer
  extends Thread
```

This class is waiting to receive a message from a client using TCP/IP protocol and then starting a new TCP thread to handle the response. The response is forwarding the message to all of the clients or disconnecting the client

Nested Class Summary

Nested classes/interfaces inherited from class java.lang.Thread

Thread.State, Thread.UncaughtExceptionHandler

Field Summary

Fields

Modifier and Type	Field and Description
private MainServer	mainServer The object to access the main server
private ServerGUI	serverGUI the object to access the GUI for the server to update the client list
private ServerSocket	serverSocket The socket to accept client connections

Fields inherited from class java.lang.Thread

MAX_PRIORITY, MIN_PRIORITY, NORM_PRIORITY

Constructor Summary

Constructors

Constructor and Description

TCPServer(**ServerSocket** serverSocket, MainServer mainServer, ServerGUI serverGUI)

The constructor which is storing the socket the client is communicating through and the object to access the main web server

Method Summary

All Methods Instance Methods Concrete Methods

Modifier and Type	Method and Description
void	run () This method is called when the thread is started in the constructor of the main server class.

Methods inherited from class **java.lang.Thread**

activeCount, checkAccess, clone, countStackFrames, currentThread, destroy, dumpStack, enumerate, getAllStackTraces, getContextClassLoader, getDefaultUncaughtExceptionHandler, getId, getName, getPriority, getStackTrace, getState, getThreadGroup, getUncaughtExceptionHandler, holdsLock, interrupt, interrupted, isAlive, isDaemon, isInterrupted, join, join, join, resume, setContextClassLoader, setDaemon, setDefaultUncaughtExceptionHandler, setName, setPriority, setUncaughtExceptionHandler, sleep, sleep, start, stop, stop, suspend, toString, yield

Methods inherited from class **java.lang.Object**

equals, finalize, getClass, hashCode, notify, notifyAll, wait, wait, wait

Field Detail

serverSocket

```
private ServerSocket serverSocket
```

The socket to accept client connections

mainServer

```
private MainServer mainServer
```

The object to access the main server

serverGUI

```
private ServerGUI serverGUI
```

the object to access the GUI for the server to update the client list

Constructor Detail

TCPServer

```
public TCPServer(ServerSocket serverSocket,  
                 MainServer mainServer,  
                 ServerGUI serverGUI)
```

The constructor which is storing the socket the client is communicating through and the object to access the main web server

Parameters:

serverSocket - The socket new clients will connect to

mainServer - Object to access the main server

Method Detail

run

```
public void run()
```

This method is called when the thread is started in the constructor of the main server class. It waits to receive a message from a TCP client and then starts a new thread to handle disconnecting the client or forwarding the clients message. The client stays connected to this socket

Specified by:

`run` in interface `Runnable`

Overrides:

`run` in class `Thread`