CNT 4714 – Programming Assignment 2 – Spring 2014

Title: “Program Assignment 3: Socket-based File Server”
Points: 100 points
Due Date: Friday February 14, 2014 by 11:59 pm (WebCourses time)

Objectives: To practice programming socket-level communications with a server. To get a better understanding of low-level client-server communications.

Description: In this programming assignment you will create a simple file server that will return the contents of a file under the server’s control to a client upon a client’s request for the file.

Your server will use TCP-based socket-level communication with a client. Similarly, your client will connect to the server using a TCP-based socket-level communication protocol to request the contents of a file located within the realm of the contacted server.

We will assume all communications between server and client will be on the localhost machine, thus no actual network connection will be required for this project.

Both the server and the client should be GUI-based applications. The server GUI, see below, will simply reflect what actions the server is performing at any moment. The server will simply start, wait for a request, service the request, and then wait for the next request in an endless cycle. The client GUI, see below, will provide a text area for the client to enter the name of a file (we’ll assume that only text files will be retrieved so that output formatting issues will not arise). The client GUI will also provide a scrollable window in which the server will display the contents of the requested file. See the screen shots below to get a better feel for what the server and client GUIs should look like.

If the client enters the name of a file which is not on the server, the server should return a file not found message of the form: “filename does not exist”.

While the server will loop infinitely waiting for and handling client connections, the clients should restart each time (e.g., don’t infinitely loop the client, but start another client instance for each file retrieval).

Include the following screen shots as part of your assignment submission:

(1) An example text file that you used.

(2) The client GUI when the client has entered the file name, but before the request to the server is made.
(3) The client GUI once the server has retrieved and displayed the file.
(4) The server GUI at some point after interaction with a client has begun.
(5) The client GUI after requesting a non-existent file.
(6) A screen shot showing the server and at least 2 different client instances.

References:
Notes: Lecture Notes for Java Networking.

Restrictions:
Your source files shall begin with comments containing the following information:

```
/*  Name:
    Course: CNT 4714 Spring 2014
    Assignment title: Program 2 – Socket-based File Server
    Due Date: February 14, 2014
*/
```

Input Specification: a file (use a text file for this program) on your system. File pathname to be entered by the client in their GUI window.

Output Specification: the contents of the specified file should be returned and printed to the client's GUI. An error message is returned to the client GUI if the specified file does not exist.

Deliverables:

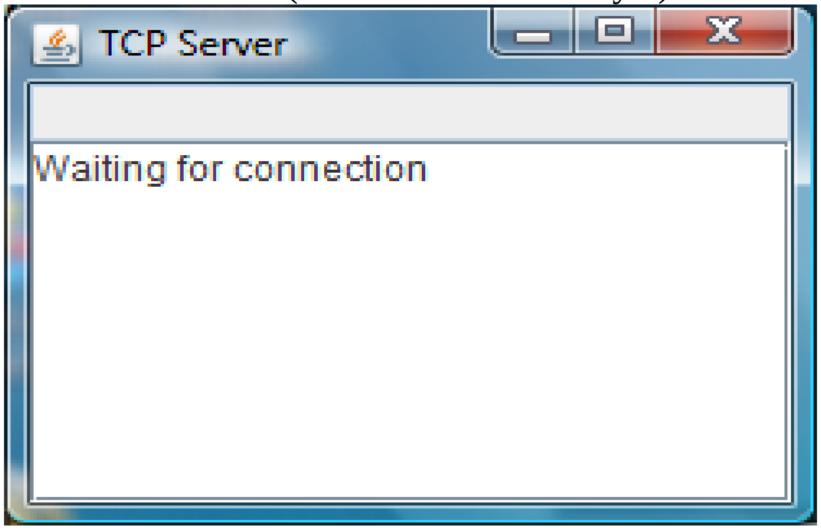
All deliverables should be submitted via WebCourses no later than 11:59pm Friday February 14, 2014.

(1) The text file that you will have the server retrieve (see 3 below).
(2) A screen shot of the client GUI with the filename entered in the input area, but before the server has retrieved the file.
(3) A screen shot illustrating the client GUI after the server has successfully retrieved and printed the file contents.
(4) A screen shot of the server GUI at some point during the interaction with a client.
(5) A screen shot illustrating the client GUI when the file specified by the client does not exist.
(6) A screen shot showing the server and at least 2 different client instances being served by the server without server shutdown in between.
(7) All of your .java files.

Additional Information:

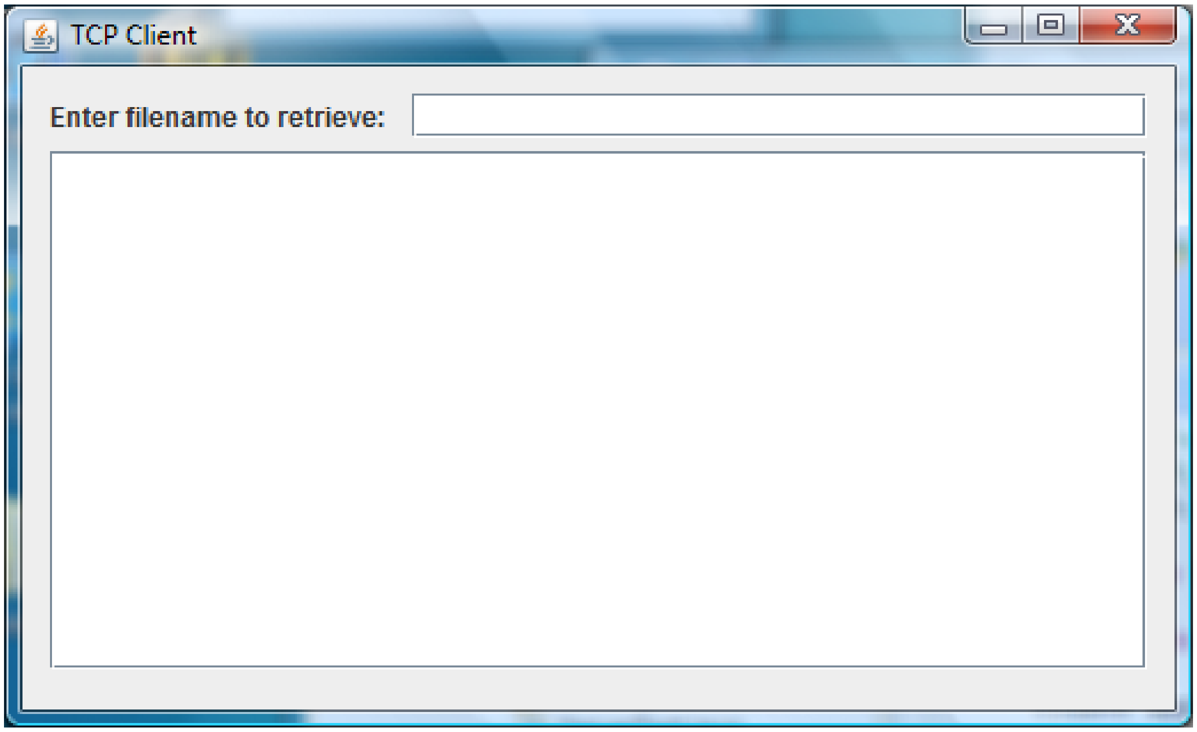
Shown below are example screen shots of the output from this program to help illustrate how your application is to operate and display the results.

Initial Server GUI (no clients connected yet):

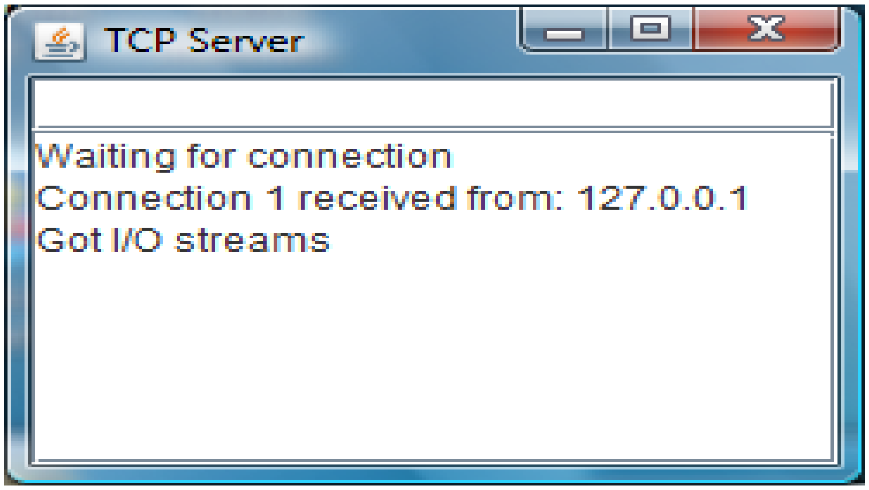


Waiting for connection

Initial Client GUI:

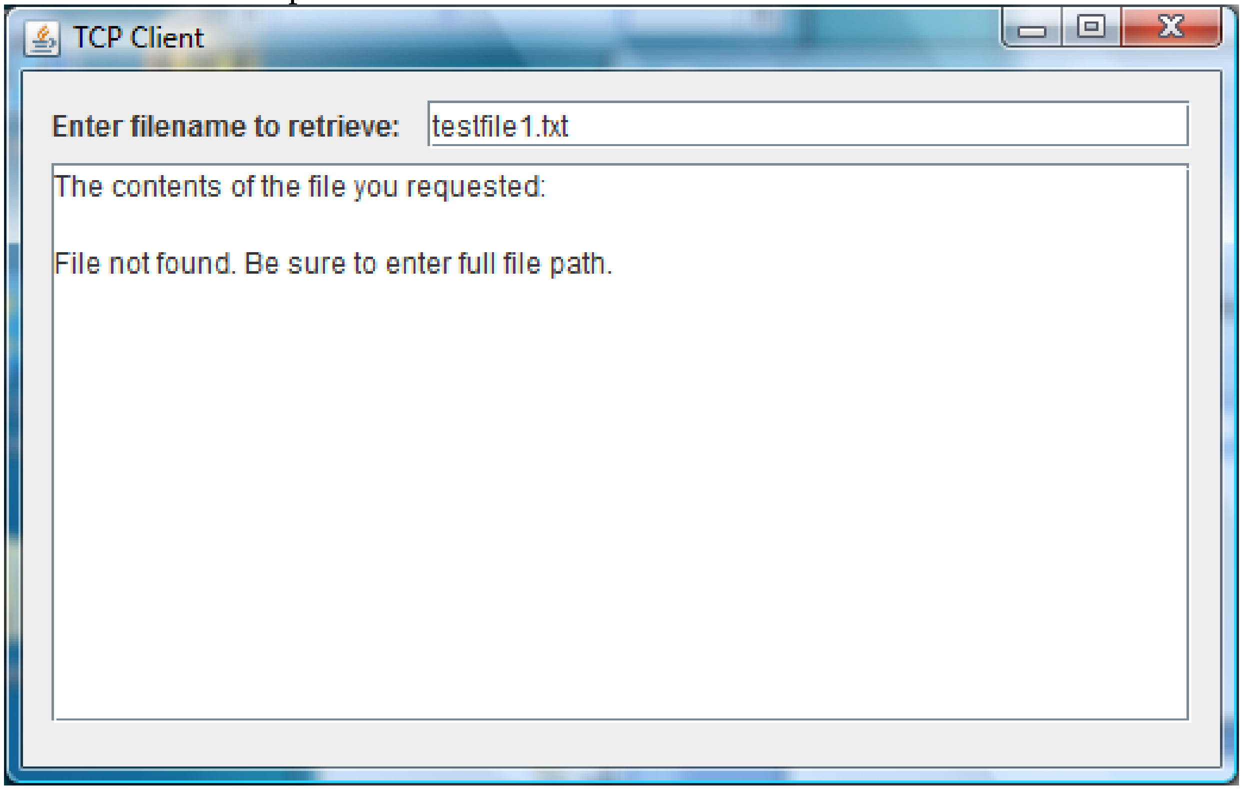


Server GUI after client connection is established:



```
Waiting for connection
Connection 1 received from: 127.0.0.1
Got I/O streams
```

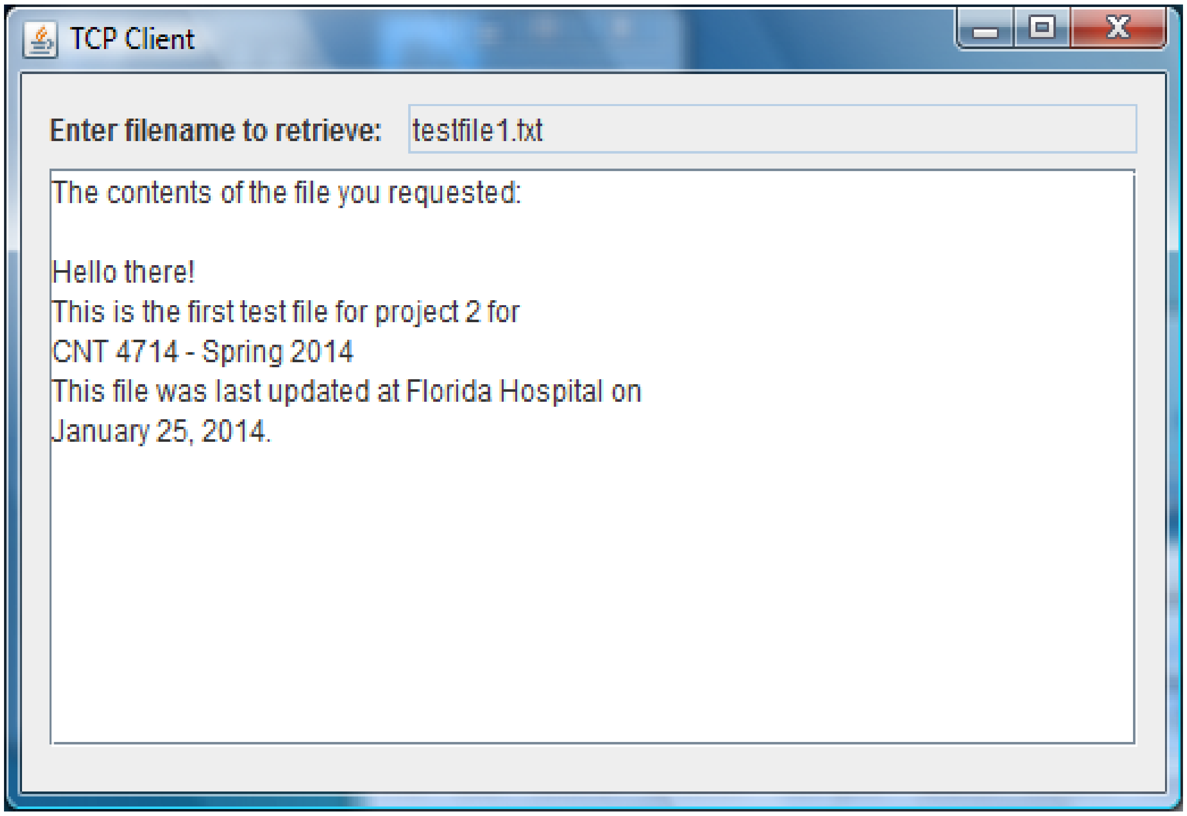
Client enters a file pathname:



```
The contents of the file you requested:

File not found. Be sure to enter full file path.
```

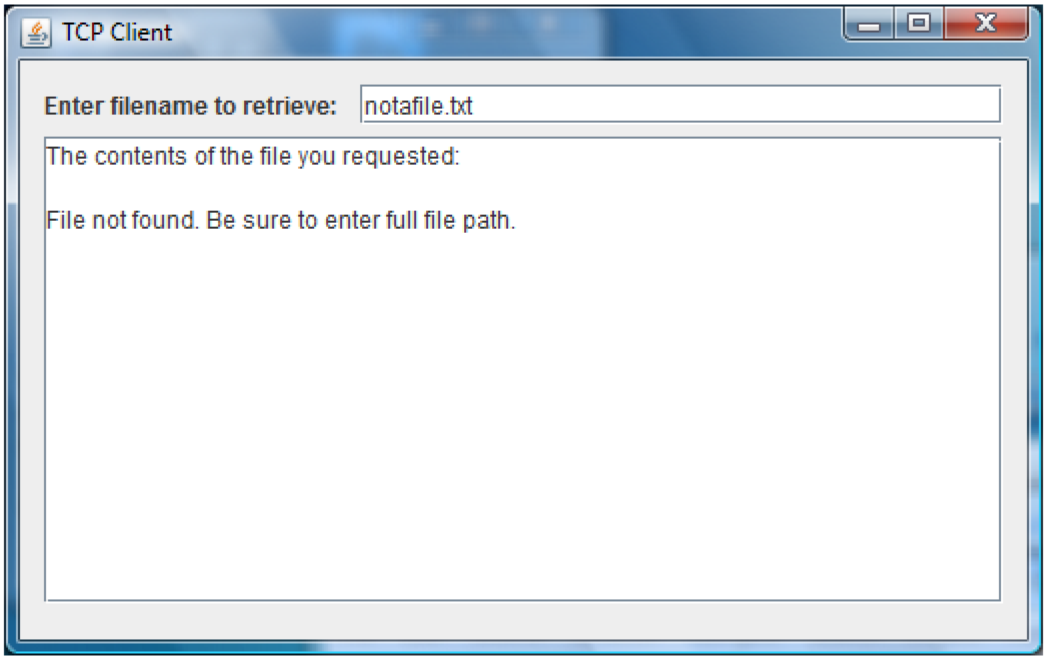
Client GUI after server returns contents of the file to the client:



```
The contents of the file you requested:

Hello there!
This is the first test file for project 2 for
CNT 4714 - Spring 2014
This file was last updated at Florida Hospital on
January 25, 2014.
```

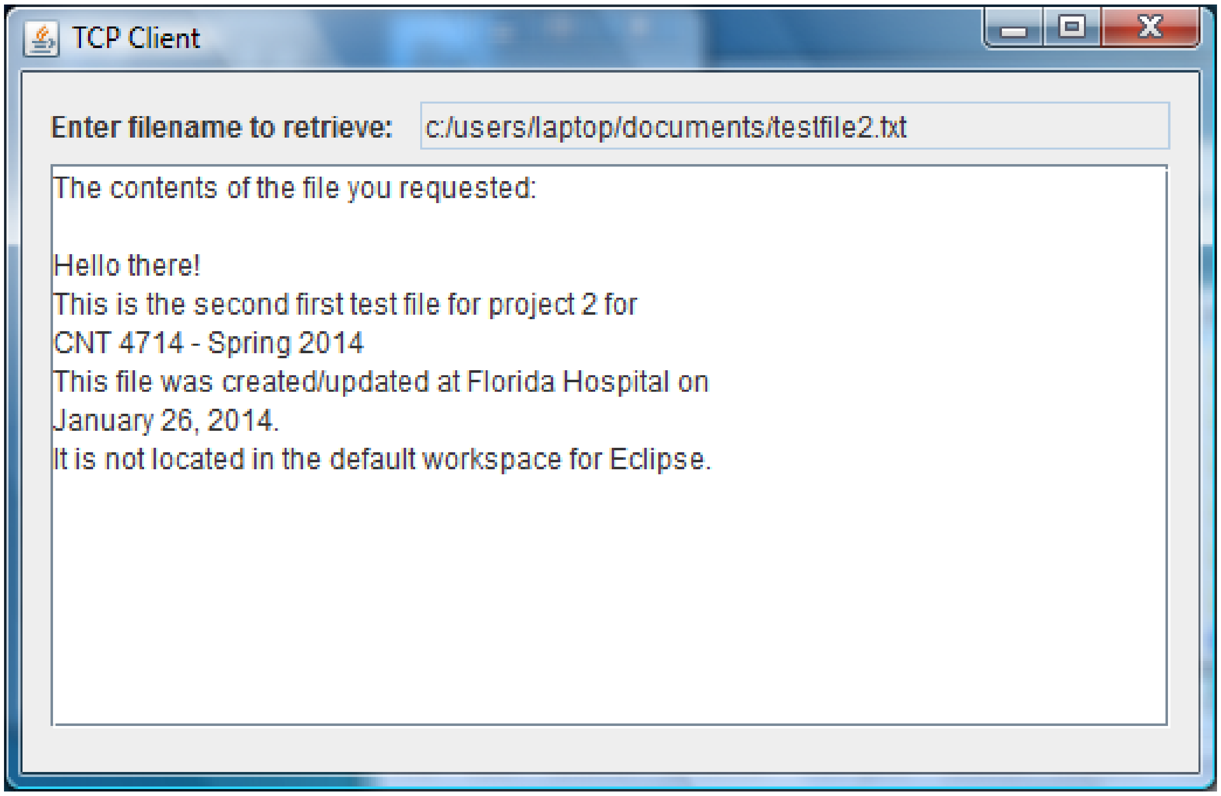
Client enters the name of a non-existent file:



```
The contents of the file you requested:

File not found. Be sure to enter full file path.
```

Note that as long as the requested file is located on the server, it should be returned to the user regardless of the path. In other words, the file does not need to be in an Eclipse workspace in order for your server to find it. Any path should work as illustrated below:

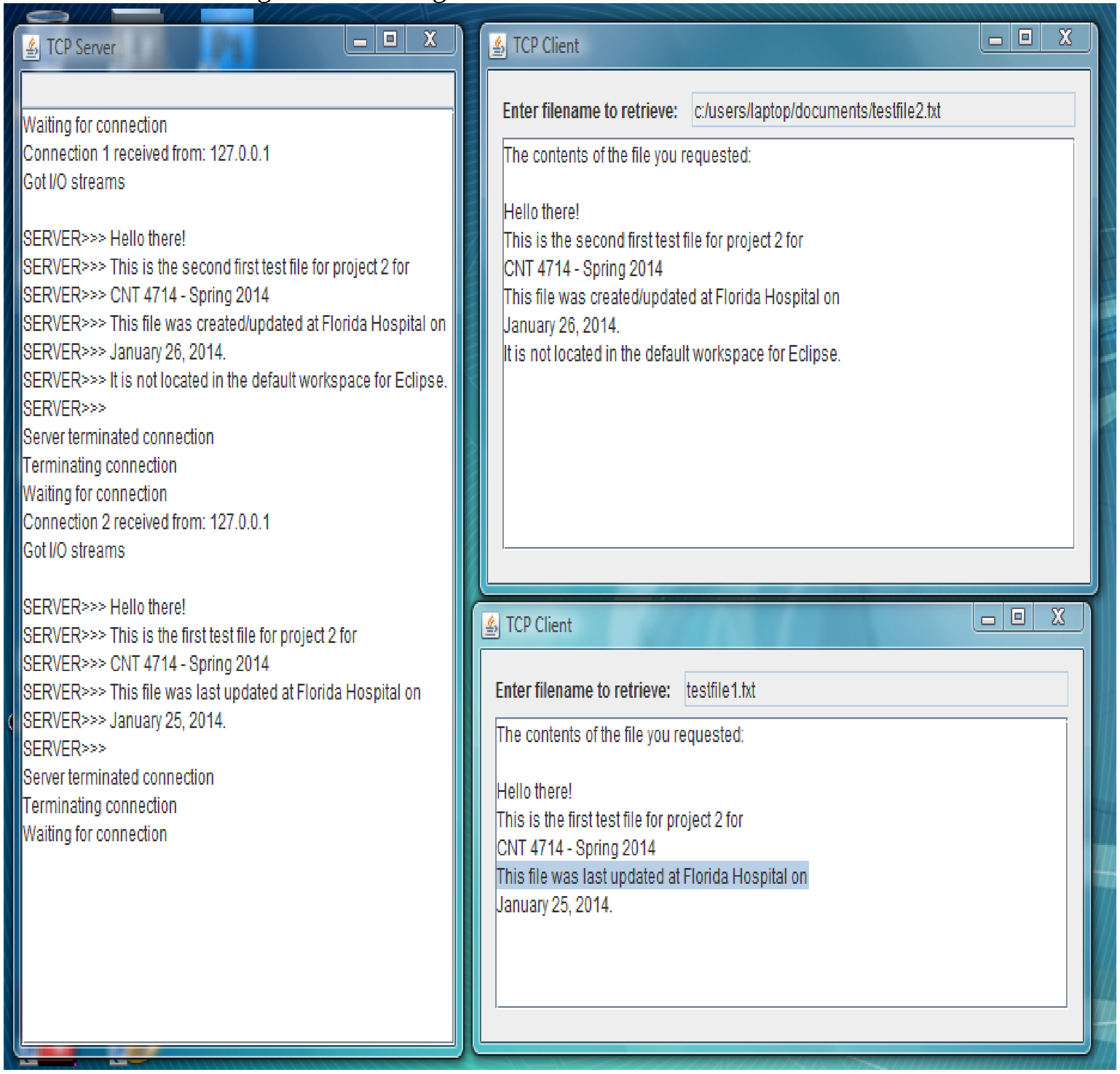


Enter filename to retrieve: c:/users/laptop/documents/testfile2.txt

```
The contents of the file you requested:

Hello there!
This is the second first test file for project 2 for
CNT 4714 - Spring 2014
This file was created/updated at Florida Hospital on
January 26, 2014.
It is not located in the default workspace for Eclipse.
```

Screen shot illustrating server running with two different client instances:



```
Waiting for connection
Connection 1 received from: 127.0.0.1
Got I/O streams

SERVER>>> Hello there!
SERVER>>> This is the second first test file for project 2 for
SERVER>>> CNT 4714 - Spring 2014
SERVER>>> This file was created/updated at Florida Hospital on
SERVER>>> January 26, 2014.
SERVER>>> It is not located in the default workspace for Eclipse.
SERVER>>>
Server terminated connection
Terminating connection
Waiting for connection
Connection 2 received from: 127.0.0.1
Got I/O streams

SERVER>>> Hello there!
SERVER>>> This is the first test file for project 2 for
SERVER>>> CNT 4714 - Spring 2014
SERVER>>> This file was last updated at Florida Hospital on
SERVER>>> January 25, 2014.
SERVER>>>
Server terminated connection
Terminating connection
Waiting for connection
```

Enter filename to retrieve: c:/users/laptop/documents/testfile2.txt

```
The contents of the file you requested:

Hello there!
This is the second first test file for project 2 for
CNT 4714 - Spring 2014
This file was created/updated at Florida Hospital on
January 26, 2014.
It is not located in the default workspace for Eclipse.
```

Enter filename to retrieve: testfile1.txt

```
The contents of the file you requested:

Hello there!
This is the first test file for project 2 for
CNT 4714 - Spring 2014
This file was last updated at Florida Hospital on
January 25, 2014.
```