SOFE 4790U Distributed Systems (Fall 2018)

Lab#2 – Distributed Applications with Java RMI

Objective

In this lab you will experiment with developing distributed applications using Java RMI. You will demonstrate your running applications to the TA (50% of the mark), and you will write and submit a lab report (50% of the mark).

Note: The Lab tasks must be completed in the lab. If you don't show up for a lab, you will receive a zero and there is no need for you to submit a lab report.

Task #1: (10 Marks)

In the pre-lab tutorial "Distributed Java Programming with RMI" a file service for allowing clients to download files from remote servers is presented. In this simple task, your job is to run the client/server application successfully and demo it to the TA. To do so:

a) Create a directory structure such as:
   FileApp
      Server
      Client

b) Download the code from Blackboard (FileInterface.java, FileImpl.java, FileServer.java, FileClient.java, and policy.txt)

c) Save the files in the right directories
   Server: FileInterface.java, FileImpl.java, FileServer.java, policy.txt
   Client: FileInterface.java, FileClient

d) Copy a couple of text and binary files into the Server directory (these are sample files of your choice that the client will download from the server)

e) Run the rmiregistry, server and client as discussed in class for any Java RMI application, or as described in the article.

Presentation to the TA:

Demonstrate to the TA the running application.

Task #2: (10 Marks)

In this task, your job is to modify the code in Task#1 so that meaningful messages are printed on the console when: (a) the server starts, (b) the ip address of the connecting client, and (c) when the client downloads a file successfully the client prints a message. Hint: you need to revise the FileServer and FileClient.

Presentation to the TA:

Demonstrate to the TA the running application.

Task#3: (10 Marks)

In this task, your job is to demo to the TA that the application in Task#2 works successfully on different machines. For example, you may run the server and your neighbour would run the client or vice versa.

Presentation to the TA:

Demonstrate to the TA that you are able to download a file from your neighbours machine (yes, you need to know the name of the file in advance).

Task#4: (20 Marks)

In this task, your job is to add a new remote method that also allows the client to upload a file to the server. To do that, you need to revise the FileInterface and add a method such as:

```
public void uploadFile(byte[] content) throws RemoteException
```

And then implement this method in FileImpl, and then revise the FileServer and FileClient.

Presentation to the TA:

Demonstrate to the TA that the client (everything is running on your own laptop) is able to upload a file to the server.

Lab Report: 1-2 pages max (50 Marks)

In your lab report, for each task:

1- Explain how you accomplished the task.
2- Describe any challenges you faced with the tasks and how you solved them.
3- What did you learn?

Submit the lab report (in Word or PDF) on Blackboard by **11:59pm on Friday, October 5. No late submissions will be accepted no matter what is the reason.**