

Distributed systems lab project

Distributed application using java RMI

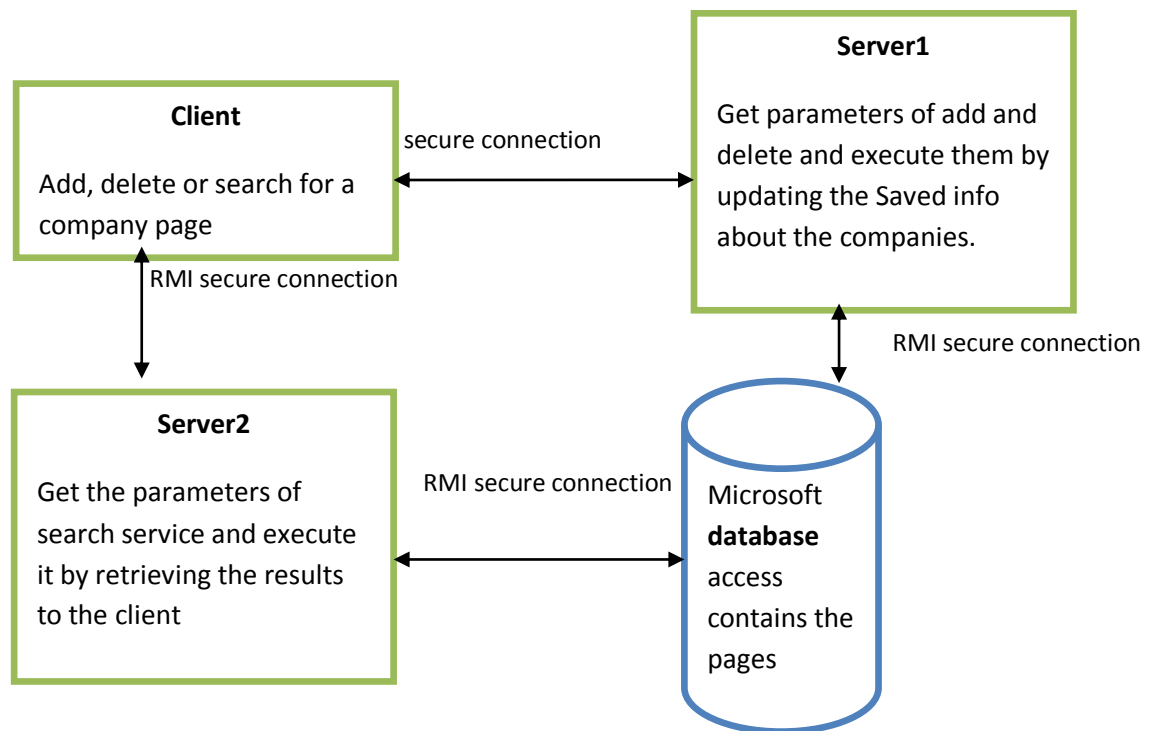
Project Description:

In this project, we aim to build a distributed system using Java RMI technology. Our application will be a simple yellow pages Directory which introduces some services like Add, Delete and Search for a contact information about companies pages.

In this case, three participants will run on different machines. The first will represent the client side (request the services), which has a yellow pages Directory interfaces and can use these services.

Server1 gets the input (as array of add and delete parameters) and update the Database it uses.

Server2 (third participant) which will be responsible for Search operation in database, then, it will send the result to the client.



The main purpose of this project is to learn how to make a load distribution (many services will be distributed on many servers). So, the load will not be centralized on a single machine.

Consider the following scenario:

The connection with servers will begin when user request the main window of the program

The main window will show the available services of the system to allow the user choose among them for example:

- 1. Add new company information.**
- 2. Delete an existing company page.**
- 3. Search for pages saved at the database.**

If the user chooses “Add New company information” Service, a new window will appear and asks the user to insert the company information .if doing so; the data entered and will be sent to server1 in order to be saved at the yellow pages database.

If the user requests the Delete Service, a new window will be displayed and prompt the user to insert the intended company name or Telephone Number then the Data entered will be sent to Server1 and delete the corresponding record from database.

The window of the Search service will prompt the user to insert a search key (i.e. company name or telephone number). When the user click on Search button, the Data entered will be sent to Server2 ; doing search in the database and return the result to the client.

Security Restrictions:

You can use any Encryption algorithm you learned.

Deliverables:

1. **Java files** (Source code) with these names:
 - a. **Client**: This will request the service.
 - b. **Interface1**: This contains the declaration of Add and Delete Methods.
 - c. **Server1**: This contains two classes: one of them make the implementation of interface1 and the other create an object from implementation class and bind with registry.
 - d. **Interface2**: This contains the declaration of Search method.
 - e. **Server2**: This contains two classes: one of them make the implementation of interface2 and the other create an object from implementation class and bind with registry.
2. **Skeletons** and **Stubs**: This will be generated using the implementation classes.
3. **Microsoft Office Access file**: That will store the data of application (with phone_directory.mdb file name).

To Consider:

- No submission will be accepted after 20th January of 2011.
- Discussion times will be determined later.
- No Groups for this project, (But Co-operation among students is allowed).
- Correctness of your program (Syntax errors, run-time errors, and/or incorrect answers will result in a significant number of points being deduced from your grade); try to do the best.
- Compress your work in a folder named with your student name and id like this:

firstName_familyName_220089999.rar

- Upload the file to the Moodle system before the deadline.

I sincerely appreciate that Mr. Abed gave me a permission to use a part of his distributed systems class assignments and consider this project for you.

Ms. Ghadeer