

Project RMI [18 points]**Deadline November 27th**

This activity intends to stimulate the students to develop a prototype called Mytube using the RMI technology. This system should allow users to remotely upload and download digital contents within an RMI system.

Basic Functions [60%]

- The Client can upload a digital content (text, music or video) to an RMI Server together with a title and description.
- The RMI Server generates a unique key for each content and store the content locally. The content can be physically stored in a folder named with its unique key, in a relational database, in a NoSQL database, etc., the decision is yours.
- The Client access to the contents doing a partial search on titles and descriptions. The RMI Server receive a string and return a list of locally stored contents that contains totally or partially the string on the title or on the description.
- The Client can download a digital content. Once a client choose a content from a list the RMI Server transfers physically the content to the client.
- The RMI Server provides a list of all the contents locally stored.

Advanced Functions [40 %]

- An RMI client can modify the title or delete contents previously uploaded only by itself.
- The RMI Server expands the search in the local repository to the global repository. A Client sends a search request to an RMI Server sending a text string. Once the RMI Server receives the string executes a local search on the local repository and sends a new search request to other RMI Servers. The result is a list of ALL contents in the distributed system that contain the corresponding string in the title or description.
- The Client can download a digital content independently of its location. When a Client request for a digital content that is not locally stored. The RMI Server redirect the petition to the source server in a transparent way for the client.

Optional Features (extra points)

When the implemented functionalities work properly and the group implemented almost one of the advanced functionalities, the next features provides extra points.

- Alternatively you can use a relational database or NoSQL database.

Considerations

You face the development of a distributed application. Your solution should provide correct implementation of the functionalities and consider efficiency, fault tolerance and performance issues. You should properly answer questions such as:

- How to generate a global unique key for all contents in the system?
- How to store physically the digital contents?
- How to reduce the number of transfers of digital contents?
- How to implement an efficient global search?
- It is important to provide security over the contents.
- It is important to provide transparency about the content location.

- Is not necessary to implement a graphical interface.
- To show the execution of your distributed application, you must execute the application in a distributed environment with different servers and clients. The implementation should consider this situation.

Report content

Create a report with the following contents:

1. Provide the UML class diagram. Describe for each class the main functionality, methods, data structures, etc..
2. Summarize the main design decisions done in this project such as class hierarchy, data structures, etc., and justify them.
3. Describe the implemented functionalities and provide the sequence diagrams for the functions: upload a content, execute a global search and download a content in a remote RMI server.
4. Run two different RMI servers. Describe and explain the outcome.
5. Run different clients in different hosts. Describe and explain the outcome.
6. Execute the main use cases. Describe and explain the outcome.

Instructions

Work in pairs to develop the distributed application project. Submit a report with the contents specified above. The final source code should be published in a repository where the professors have access. Do not forget to indicate in the report the time spent on the activity. **There will be a face-to-face presentation with the professors. In the case a group deliver the project after the deadlines there will be penalizations on the score.**

References

RMI Tutorial. <http://docs.oracle.com/javase/tutorial/rmi/>

Remote Method Invocation Home. <http://www.oracle.com/technetwork/java/javase/tech/index-jsp-136424.html>