

RMI Project

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

Basic Features

- The Client can upload a digital content to an RMI Server together with a title.
- The RMI Server generates an unique key for each content and store the content locally in a folder named with this key.
- The Client can access to contents using a textual description. The RMI Server receive the textual description and return the list of contents (titles) related to this description.
- The Client can download a digital content using a title. The RMI Server finds locally the digital content associated to that specific title and provide it to the client.

Advanced Features

- The RMI Server notifies to all Clients the addition of a new content.
- The Client can access to distributed contents using a textual description. The Client sends the request to an RMI Server. The RMI Server receive the textual description, perform a global search and returns the list of ALL contents (titles) in the distributed system related to this description.
- The Client can download a distributed digital content. When a Client request for a digital content is not locally stored the, The RMI Server redirect the petition to the source server in a transparent way for the client.

Optional Features

- Configure some Security policies. Justify your election and test it.
- An RMI client can modify the title of the contents uploaded by itself.
- An RMI client can delete contents previously uploaded by itself.

Deliveries

Create a report with the following contents:

1. Describe the implemented classes and the corresponding UML class diagram.
2. Describe the implemented functionalities and provide the sequence diagrams.
3. Describe the main design decisions provided in this project and the used data structures.
4. Run an RMI server. Check the displayed messages, including the list of names currently in the registry. Do you see the name that the server registered under (as specified in the program)? How can you check the remote services registered?
5. Compile and execute one Client in a different host. Describe and explain the outcome.
6. Execute the main use cases. Describe and explain the outcome.
7. Run two or more clients on a separate machine. Describe using a sequence diagram and screenshots the events produced. Can you explain if the method calls are executed concurrently or iteratively? Justify the response.

Instructions

Working in pairs try to solve the different exercises. Submit a report with the answers to the questions, listing the source codes and outputs. Do not forget to indicate in the report the time spent on the activity.

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>