

Disciplina Lab. Desenv. Aplic. Móveis e Distribuás	Departamento Engenharia de Software	Turno Noite	Período 5°
Professor Hugo de Paula (hugo@pucminas.br)			

Trabalho Dicionário Distribuído em dupla

1 Objetivos do trabalho

O objetivo deste trabalho é o de praticar a comunicação entre processos utilizando a tecnologia Java RMI (*Remote Method Invocation*).

2 Especificação

Utilizando uma arquitetura cliente-servidor, projete e implemente um servidor que permita que clientes busquem concorrentemente o significado, adicionar uma palavra ou remover uma palavra de um dicionário distribuído.

Do ponto de vista da estrutura de dados, um dicionário é uma coleção de pares chave-valor, onde cada chave única mapeia para um valor correspondente. A chave em um dicionário é geralmente usada como um identificador exclusivo para acessar o valor associado a ela. Esta estrutura permite uma busca eficiente de elementos com base em sua chave, tornando-a ideal para muitas situações onde é necessário armazenar e recuperar informações associadas a identificadores únicos.

O trabalho foi projetado para demonstrar o funcionamento da invocação de métodos remotos baseada no Java RMI. Você deverá garantir que o sistema funcione com múltiplos clientes acessando o sistema concorrentemente. Segundo a especificação do Java RMI¹:

“A method dispatched by the RMI runtime to a remote object implementation may or may not execute in a separate thread. The RMI runtime makes no guarantees with respect to mapping remote object invocations to threads. Since remote method invocation on the same remote object may execute concurrently, a remote object implementation needs to make sure its implementation is thread-safe.”

Requisitos arquiteturais:

1. A arquitetura deverá ser cliente-servidor, com um único servidor servindo múltiplos clientes.
2. A interação deverá ser feita por Java RMI.
3. Os dados devem ser persistidos de alguma forma: XML, JSON, *Java Object Serialization*, etc.

¹<https://docs.oracle.com/javase/8/docs/technotes/tools/windows/s4-rmi-tools.html>

4. Espera-se que sejam tratados os erros adequadamente: entrada/saída, rede, etc.

Requisitos funcionais:

1. Consultar o significado de uma palavra.

Entrada: *palavra*. **Saída:** *significado*. **Erro:** *Palavra não encontrada e padrões do Java*.

2. Adicionar nova palavra.

Entrada: *palavra, significado*. **Saída:** *sucesso ou duplicado*. **Erro:** *padrões do Java*.

3. Remover palavra.

Entrada: *palavra*. **Saída:** *sucesso ou não encontrado*. **Erro:** *padrões do Java*.

4. Desenvolver interface gráfica simples para o cliente.

5. Endereço do servidor deve ser configurável no cliente (pode usar argumento de linha de comando).

3 Resultados Esperados

Deverá ser entregue:

1. Todo o código, comentado.
2. Um arquivo README.TXT contendo uma explicação sucinta do código e instruções para compilação e execução do mesmo.