

# **Escola Superior de Tecnologia e Gestão**

## **Licenciatura em Engenharia Informática**

### **Sistemas Distribuídos**

**Ano Letivo 2021/22**

### **Ficha 7**

**Elaborado em: 2022/04/20**

**Daniel Santos | Nº2019133865**

## **Índice**

<b>List of Figures .....</b>	<b>ii</b>
<b>1. Exercício 1 .....</b>	<b>1</b>

## List of Figures

FIGURA 1 - SERVER INICIADO .....	1
FIGURA 2 - CLIENT TESTE .....	1
FIGURA 3 - CLIENT OFERTA.....	1
FIGURA 4 - CLIENT REMOVER .....	1
FIGURA 5 - SERVER FUNÇÃO OFERECE .....	2
FIGURA 6 – SERVER.....	2
FIGURA 7 - SERVER OFERECE SYNCHRONIZED.....	2

## 1. Exercício 1

### Execução do programa:

```
run:  
servidor RMI iniciado  
|
```

Figura 1 - Server iniciado

```
Introduza o ip: localhost  
Conectado...  
1.Apresentar item 2.Apresentar itens 3.Adicionar item 4.Remover item 5.Fazer oferta 0.Sair  
Introduza a opção que deseja:3  
Descrição: pao  
Preço minimo: 12  
Indique a data de fecho do item...  
Introduza ano: 9999  
Introduza mês: 12  
Introduza dia: 12  
Introduza hora: 12  
Introduza minuto: 12  
Item adicionado com sucesso  
1.Apresentar item 2.Apresentar itens 3.Adicionar item 4.Remover item 5.Fazer oferta 0.Sair  
Introduza a opção que deseja:2  
Item{id=1, descricao=pao, dataHoraFecho=java.util.GregorianCalendar[time=253403295156821,areField.  
1.Apresentar item 2.Apresentar itens 3.Adicionar item 4.Remover item 5.Fazer oferta 0.Sair  
Introduza a opção que deseja:1  
Introduza id do item:1  
Item{id=1, descricao=pao, dataHoraFecho=java.util.GregorianCalendar[time=253403295156821,areField.
```

Figura 2 - Client teste

```
1.Apresentar item 2.Apresentar itens 3.Adicionar item 4.Remover item 5.Fazer oferta 0.Sair  
Introduza a opção que deseja:5  
Introduza id do item que deseja fazer oferta: 1  
Introduza preço que deseja oferecer: 50  
Oferta a item feito com sucesso
```

Figura 3 - Client Oferta

```
1.Apresentar item 2.Apresentar itens 3.Adicionar item 4.Remover item 5.Fazer oferta 0.Sair  
Introduza a opção que deseja:4  
Introduza id do item que deseja remover: 1  
Item removido com sucesso  
1.Apresentar item 2.Apresentar itens 3.Adicionar item 4.Remover item 5.Fazer oferta 0.Sair  
Introduza a opção que deseja:2  
1.Apresentar item 2.Apresentar itens 3.Adicionar item 4.Remover item 5.Fazer oferta 0.Sair  
Introduza a opção que deseja:|
```

Figura 4 - Client Remover

**Pergunta:** A exclusão mútua no acesso aos métodos de um objeto remoto é garantida pelo Java RMI? Explique e demonstre a sua resposta.

Para testar introduzi um scanner na função oferece():

```
public boolean oferece(int id, float valor) throws RemoteException {
    if (itemList != null) {
        System.out.println("Cliente entrou na função");
        inputString("");
        for (int i = 0; i < itemList.size(); i++) {
            if ((itemList.get(i).getId() == id) && itemList.get(i).getPrecoAtual() <= valor) {
                if (itemList.get(i).getDataHoraFecho().before(Calendar.getInstance())) {
                    itemList.get(i).setTerminado(true);
                    return false;
                }
                itemList.get(i).setPrecoAtual(valor);
                itemList.get(i).setNumeroOfertas(itemList.get(i).getNumeroOfertas() + 1);
                return true;
            }
        }
    }
    return false;
}
```

Figura 5 - Server função oferece

Depois entrei em dois Client e iniciei a função oferece() do lado do Client:

```
servidor RMI iniciado
Cliente entrou na função
Cliente entrou na função
```

Figura 6 – Server

Ambos os Client conseguiram entrar na função ao mesmo tempo pois estes apenas saem quando o server introduzir um enter para deixar.

**Resposta:** Não. O RMI não fornece essa garantia por si só, e duas chamadas no mesmo objeto remoto podem ser executados simultaneamente, a menos que se implemente alguma sincronização. A abordagem correta é sincronizar todos os métodos pois irá garantir que dois clientes não sejam executados ao mesmo tempo no mesmo método.

```
public synchronized boolean oferece(int id, float valor) throws RemoteException {
```

Figura 7 - Server oferece synchronized

