



Universidade De Évora  
Curso: Engenharia Informática  
UC: Sistemas Distribuídos  
Docente: José Saias  
Entregue Novembro 2022

# Room Rent

Relatório Do Trabalho Prático



Alunos:

Joana Carrasqueira, nº 48566

João Condeço, nº 48976

# Índice

Introdução.....	2
Server .....	2
Client.....	3
Manager .....	4
Conclusão .....	5

## Introdução

Este trabalho prático tem como objetivo conciliar os conhecimentos adquiridos na cadeira de Sistemas Distribuídos para desenvolver um novo sistema, com a finalidade de gerir os anúncios de quartos para arrendar, bem como as respetivas solicitações/procura. Para isso foi necessário implementar as seguintes aplicações:

- ⇒ Client
- ⇒ Manager
- ⇒ Server

Este projeto foi desenvolvido em *Java* com o auxílio da ferramenta de automação de compilação *Maven*.

## Server

A aplicação server tem como objetivo fornecer uma interface remota que permite ao cliente comunicar com a base de dados. Quando este inicia, é pedido ao utilizador as credencias necessárias para aceder à base de dados indicada nos argumentos de execução. Efetuado este passo, se tudo correr bem, o servidor fica online até ser encerrado pelo utilizador, caso contrário, o servidor termina logo no início.

### ⇒ Compilação:

```
mvn clean  
mvn compile  
mvn package
```

### ⇒ Execução:

```
mvn exec:java -Dexec.mainClass="work.rmi.Server" -Dexec.args="port server user"
```

## Client

A aplicação client tem como objetivo permitir ao utilizador comum interagir com o programa. Fornecendo a este as seguintes funcionalidades:

- ⇒ sign in – Inicia a sessão no sistema;
- ⇒ sign up - Regista utilizador no sistema;
- ⇒ add - Permite adicionar anúncios;
- ⇒ show ads - Mostra os anúncios atualmente ativos;
- ⇒ show search - Permite procurar com filtros;
- ⇒ search by name - Permite procurar anúncios pelo nome do anunciante;
- ⇒ search by id - Mostra os anúncios através do id inserido;
- ⇒ send msg ou send message - Envia uma mensagem para um anúncio ou para um utilizador;
- ⇒ check msg ou check message - Permite ver as mensagens de um anúncio ou de um utilizador;
- ⇒ check users - Permite ver os utilizadores do sistema;
- ⇒ logout - Termina a sessão do utilizador no sistema;
- ⇒ help ou ? - Mostra este menu;
- ⇒ quit - Termina o programa.

Existem certas funcionalidades que exigem que o sign in seja feito previamente.

### ⇒ **Compilação:**

```
mvn clean  
mvn compile  
mvn package
```

### ⇒ **Execução:**

```
mvn exec:java -Dexec.mainClass="work.rmi.Client" -Dexec.args="host port"
```

# Manager

A aplicação manager tem como objetivo permitir ao utilizador de gestão interagir com o programa e administrar as informações. Fornecendo a este as seguintes funcionalidades:

- ⇒ sign in - Inicia a sessão no sistema;
- ⇒ show ads - Mostra os anúncios atualmente ativos;
- ⇒ show search - Permite procurar com filtros;
- ⇒ search by name - Permite procurar anúncios pelo nome do anunciante;
- ⇒ search by id - Mostra os anúncios através do id inserido;
- ⇒ activate ad - Muda o estado do anúncio para ativo do id inserido;
- ⇒ deactivate ad - Muda o estado do anúncio para inativo do id inserido;
- ⇒ upgrade user - Muda um utilizador cliente para gestor do id inserido;
- ⇒ check users - Permite ver os utilizadores do sistema;
- ⇒ remove ad - Remove um anúncio de acordo com o id inserido;
- ⇒ remove user - Remove um utilizador de acordo com o username inserido;
- ⇒ logout - Termina a sessão do utilizador no sistema;
- ⇒ help ou ? - Mostra este menu;
- ⇒ quit - Termina o programa.

O sign in é exigido logo de início para garantir que o utilizador comum não tem acesso aos dados e funcionalidades de administrador.

## ⇒ Compilação:

```
mvn clean  
mvn compile  
mvn package
```

## ⇒ Execução:

```
mvn exec:java -Dexec.mainClass="work.rmi.Manager" -Dexec.args="host port"
```

## Conclusão

Durante a elaboração deste projeto decidimos acrescentar uma “simulação” de sistema de registo e autenticação de utilizadores. Graças a este adicionamos a funcionalidade troca de mensagens entre utilizadores sem a necessidade de existir um anúncio. Para auxiliar os utilizadores a comunicarem entre si, também foi acrescentada a funcionalidade de ver a lista de todos os utilizadores.

A dificuldade mais significativa encontrada foi acrescentar as dependências do *PostgreSQL* ao ficheiro *pom.xml*. Anteriormente a isto quando executávamos o programa no terminal, este acusava que não existia o driver para interagir com a base dados.