



FCTUC FACULDADE DE CIÊNCIAS  
E TECNOLOGIA  
UNIVERSIDADE DE COIMBRA



## Sistema de Votos

# Introdução

Foi-nos proposto realizar um projeto no âmbito da unidade curricular de Sistemas Distribuídos. O trabalho proposto consiste em montar um sistema de votos, com servidores e terminais de voto. No decorrer deste relatório será explicado detalhadamente a arquitetura da plataforma bem como o funcionamento de cada um dos servidores.

# Arquitetura da Plataforma

A Plataforma é constituída por dois servidores RMI (DataServer.java e DataServerBackUp.java), um ou mais servidores TCP (TCPServer.java), uma ou mais consolas de administração (Admin.java) e um ou mais terminais de voto (Terminal.java)

O servidor de RMI é um processo que acede aos dados armazenados em ficheiros de objetos e responde aos pedidos efetuados pelos servidores de TCP provenientes dos eleitores ligados aos terminais ou mesas de voto (em caso de pedido de desbloqueio).

Os servidores de TCP, para além de mesas de voto, onde os eleitores se identificam e serem dirigidos a um terminal de voto, servem também de intermediários entre os eleitores e os dados disponíveis na aplicação, fazendo *parsing* dos pedidos das mensagens dos eleitores de acordo com o protocolo implementado, obtendo a resposta que, posteriormente, é enviada para o correspondente eleitor. Estes servidores são providos de várias *Threads* que são criadas sempre que um novo cliente se conecta à aplicação e eliminadas quando o utilizador se desconecta.

Os *scripts* utilizados para o desenvolvimento desta aplicação estão organizados da seguinte forma:

```
RMI Server:
    -DataServer.java
    -DataServerBackUp.java
    -RMI.java
TCP Server:
    -TCPServer.java
    -Server_rmi.java
Terminal:
    -Terminal.java
Administrador:
    -Admin.java
    -Admin_rmi.java
Classes:
    -Departament.java
    -Eleicoes.java
    -Faculty.java
    -Lista_candidata.java
    -Lista.java
    -Mesas.java
    -Pessoa.java
    -ProtocolParser.java
    -Results.java
```

## Funcionamento do Servidor TCP

O servidor TCP contém uma pequena interface que representa a mesa de voto e é provido de várias *threads* que são criadas quando um novo utilizador se liga a plataforma, essa *thread* permanece em funcionamento até que o cliente se desliga da aplicação. Cada *thread* é responsável apenas por um cliente. As ligações aos terminais são feitas através do protocolo TCP.

A pequena interface tem como objetivo, autenticar um eleitor, para que se lhe possa ser atribuído um terminal.

O servidor faz o *parsing* dos pedidos das mensagens percebidas pelos eleitores e tratamento dos mesmos. Nota: A classe [ProtocolParser.java](#), responsável pelo *parsing* dos pedidos, não é da nossa autoria. Foi disponibilizado pelo Professor Alcides Fonseca no ano letivo 2016/2017. O protocolo separa os argumentos por “,” e as entradas dos argumentos por “.”. Exemplo: “type:login, username: aluno, password:2014”

Todas as configurações respetivas ao servidor TCP encontram-se no ficheiro de configuração “TCPServerConfig.txt”. Neste mesmo ficheiro estão preparadas duas configurações que pode escolher na inicialização do servidor.

## Funcionamento do Servidor RMI

O Servidor RMI é o responsável por completar os pedidos efetuados pelos eleitores depois de estes serem tratados no servidor tcp e também por efetuar quaisquer mudanças desejadas por um administrador. Contém inúmeros métodos para realizar todas as tarefas descritas no enunciado do projeto sendo alguns destes, métodos de *callback*, tais como notificar todos os administradores de um voto efetuado, notificar as mesas online ou ainda o termino/começo de uma eleição.

Este Servidor faz-se acompanhar de outro Servidor RMI que vai servir de suporte caso o principal falhe. Comunicam por protocolo UDP enviando pacotes de bytes um para o outro. Desta forma conseguimos iniciar o servidor secundário como primário ao fim de 30 segundos sem resposta do servidor primário.

# Administração

A consola de administração tem o poder administrativo sobre a base de dados do projeto. A consola de administração ao contrário dos terminais liga-se “diretamente” ao servidor RMI (*lookup*) e contém uma interface. Esta interface permite-lhe de forma fácil fazer alterações no decorrer das eleições. Todas as notificações são impressas automaticamente na consola.

## Distribuição de Tarefas

Todo o Projeto foi realizado em conjunto (servidores, *callbacks*, *failovers* etc...) à exceção do Servidor RMI que foi grande parte autoria do elemento Fábio Antunes.