



FACULDADE DE
CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE DE
COIMBRA

Relatório Meta 1 de SD

Online Voter

Servidor multicast

Funcionamento

O servidor multicast é composto por duas threads:

- Uma thread que cuida dos inputs de novos utilizadores (identificação e escolha de eleição) e da seleção do respectivo terminal;
- Uma outra que gere as mensagens enviadas pelo terminal de voto (autenticação e voto).

A thread que gere as mensagens do terminal de voto, verifica a validade do CC (aqui usado como identificador) e da password inseridos na multicast Table, de acordo com a base de dados. Nesta mesma thread, para evitar a duplicação de votos, existe um mecanismo de proteção, que passa por manter uma estrutura comum a ambas a threads, que mantém informação sobre cada voting Terminal. Esta estrutura inclui informação como dados pessoais, informações sobre o voto, bem como uma flag que indica se o voto foi guardado ou não.

Ambas as threads comunicam por RMI com o RMIServer, a primeira para obter os dados da pessoa aquando da sua identificação e a segunda para enviar o voto.

Failover

Sempre que o servidor multicast comunica com o RMIServer, no caso de ocorrer uma RemoteException, é usada uma função que tenta conectá-lo com o RMIServer novamente.

Se o RMIServer primário não estiver funcional, o secundário é contactado. Se este também falhar, o server multicast espera por 1 segundo e tenta novamente estabelecer ligação com um dos RMIServers, se após determinado tempo (a indicar pelo programador) não for possível estabelecer comunicação com nenhum dos servidores, o programa termina indicando ao utilizador que não existem servidores disponíveis no momento.

Comunicação RMI

Na instanciação do multicast server, são-lhe passados, para além de outros parâmetros, os ips e os portos dos RMIServers, para que este se consiga conectar a estes.

Comunicação Multicast

Na instanciação do multicast server, os ips e portos multicast de discovery e de voting, para que este consiga comunicar com os terminais de voto associados a esta mesa (estes mesmos ips são passados aos terminais de voto).

Protocolo de Comunicação

Quando um utilizador se identifica na voting table, a mensagem 1 é enviada pelo grupo multicast de discovery. As mensagens 2, 3 e 4 são também enviadas pelo grupo de multicast de discovery.

As restantes mensagens são trocadas pelo grupo multicast de voting.

O terminal de voto ao receber a mensagem 5.2 termina o processamento do utilizador atual e coloca-se como disponível.

Voting Table		
1	id *;type availability	Procura por um terminal disponível
3	id N;type lock	Indica a um terminal livre que este vai servir o utilizador
6	id N;type list;itemcount X;listitem1 Y;listitemN Z OR id N;type error	Transmite a informação sobre a votação(listas) ao Terminal de voto
8	id N;type unlock	Desbloquear o terminal (caso haja falha no login ou voto registado com sucesso)

Voting Terminal		
2	id *;type availability	Echo message que indica que o terminal está disponível
4	id N;type locked	Indica à mesa de voto que recebeu mensagem 2
5	id N;type auth;cc X;password Y	Transmite informação de identificação
7	id N;type vote;list {blank, null, listname};time Y/M/D-HH:mm	Transmite a informação sobre a voto à Mesa de voto

Servidor RMI

Funcionamento

O RMIServer é composto por uma thread.

É instanciado com os portos onde fica o próprio e onde fica o secundário. Foi assumido que ambos os servers se encontram na mesma máquina, utilizando ficheiros objeto para guardar e carregar as respetivas informações (pessoas, eleições, listas de voto).

Ao iniciar, se não conseguir contactar o RMIServer de backup, este liga-se como primário, criando o objeto remoto.

Sempre que uma alteração é efetuada aos dados, estes são guardados na base de dados, para que, se este RMIServer parar, o secundário ter a informação mais atualizada.

Quando um novo voto é efetuado, se a consola de administração se encontrar na opção para visionamento de realtime data, a eleição e o número de votos até ao momento em cada mesa vão ser apresentados nesta, através de um método remoto que é acedido pelo RMIServer. Isto é possível através de callbacks do RMIServer. Neste estão guardadas as consolas de administração, que têm de se inscrever a este através de um método remoto disponibilizado pelo RMIServer.

Heartbeats

Quando é iniciado, se não se conseguir ligar ao segundo servidor, este fica como primário. Caso consiga conectar-se, fica à espera até que deixe de receber os heartbeats desse servidor. O servidor tenta estabelecer a ligação 5 vezes (consecutivas) antes de ficar como primário.

API

Consultar a documentação Javadoc.

Distribuição de tarefas

Tarefa	Realizador
RMIServer	Ambos
Admin Console	Ambos
Voting Terminal	Gabriel
Voting Table	Duarte

*Nota : Quando se indica que apenas 1 membro do grupo realizou uma tarefa apenas significa que uma parte significativamente maior dessa tarefa foi feita por esse membro havendo no entanto entajuda na realização dessa mesma tarefa.

Testes

Test	Status
Verificar que o RMIServer secundário se liga quando o primário vai a baixo	pass
Verificar que não se perdeu informação,	pass
verificar que basta reiniciar o terminal de voto, após um crash, para que este volte a funcionar normalmente	pass
Verificar que cada pessoa apenas vota uma vez em cada eleição	pass
Verificar que se o user colocar credenciais de acesso erradas ou que não são as dele, mesmo sendo de outro user, o terminal bloqueia	pass

Verificar que não dá para votar numa eleição depois desta terminar	pass
Verificar que a pessoa apenas pode votar nas eleições para o seu departamento	pass
Verificar que determinada eleição apenas aparece nas mesas adicionadas previamente a esta	pass
Verificar que não dá para criar duas listas com o mesmo nome para a mesma eleição e mesmo tipo de pessoas em intervalos de tempo sobrepostos	pass
Verificar que as consolas de administração, quando na opção de mostrar real time data, recebem informação de quando um voto é realizado	pass