



Universidade do Minho

Escola de Engenharia

Mestrado Integrado em Engenharia Informática

Sistemas Distribuídos

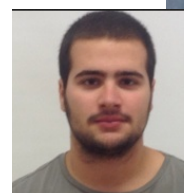
Ano Letivo de 2016/2017

Trabalho Prático: “Leilões”

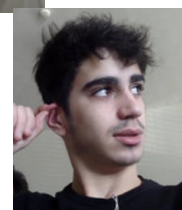
A27748 - Gustavo José Afonso Andrez



A74634 - Rogério Gomes Lopes Moreira



A76507 - Samuel Gonçalves Ferreira



Docentes:

Rui Carlos Mendes Oliveira

Carlos Miguel Ferraz Baquero Moreno

Paulo Sérgio Soares Almeida

Vitor Francisco Mendes Freitas Gomes Fonte

2 de janeiro de 2017

Índice

Índice	2
Introdução	3
Cliente	4
Comprador	5
Vendedor	5
Servidor	6
Comentários e Conclusão	6

Introdução

O objetivo deste trabalho é, recorrendo às competências desenvolvidas nas várias disciplinas do curso, e em particular, na disciplina de Sistemas Distribuídos conceber uma aplicação distribuída que permita a gestão de um serviço de leilões podendo ser acedida por dois tipos de utilizadores: vendedores e compradores. Os vendedores criam leilões e podem-nos encerrar. Os compradores licitam em leilões. Os utilizadores interagem com o programa do servidor *multi-threaded* a partir de um programa cliente, ambos escritos em Java. A comunicação é feita via *sockets* TCP, o endereço é o *localhost* e a porta utilizada é a 9999.

Cliente

O programa cliente pode comunicar com o servidor, através de *sockets* TCP para fazer uma das seguintes ações: registar um utilizador, iniciar sessão de um utilizador, iniciar um leilão, licitar num leilão ou terminar um leilão. As ações que cada um pode realizar dependem do tipo de conta do utilizador em questão. As mensagens são enviadas ao servidor sempre com o formato:

`"<código_operação> + <argumentos necessários para a operação>"`

Quando o utilizador inicia a aplicação é-lhe apresentado um menu que lhe permite fazer *login*, registar-se ou sair da aplicação.

Caso o utilizador escolha registar-se é-lhe pedido que indique o *username*, a palavra-passe e o tipo de utilizador pretendidos. Caso o tipo seja diferente de 1 ou 2 então o sistema imprime no ecrã que o tipo é inválido. Se o tipo de utilizador for válido, é enviada uma *string* ao servidor começada pela palavra *addUser*, seguida pelo *username*, pela *password* e pelo tipo de utilizador. O utilizador fica registado se o *username* estiver disponível, ou seja, se não houver mais nenhuma conta com o mesmo *username*. Neste caso é-lhe atribuído um ID único que é retornado ao cliente pelo servidor.

Caso o utilizador escolha fazer o login é-lhe pedido o seu *username* e a sua palavra-passe, estas informações são enviadas ao servidor via *socket* numa *string* identificada no início por *login* (o servidor sabe assim que é um pedido de login), este encarrega-se de verificar se o *username* e a palavra-passe são válidos e, em caso negativo – o servidor envia uma *string* ao cliente com o valor -1; em caso afirmativo - o servidor envia na *string* ao o id do utilizador que acabou de fazer login. Quando as credenciais são válidas é feito um novo pedido ao servidor identificado por *getTipoUser* que retorna o tipo de utilizador que está a fazer login (1 se for comprador, 2 se for vendedor). Caso o valor retornado pelo servidor seja diferente de 1 e 2 é imprimido no ecrã que o tipo de utilizador não é válido e o login não é feito (houve algum erro nos dados do sistema). Caso o valor retornado seja 1 ou 2 é apresentado o menu correspondente ao tipo de utilizador.

Comprador

Caso o utilizador seja um comprador é lhe apresentado um menu que lhe permite consultar a lista de leilões, fazer uma licitação ou simplesmente sair da conta.

Se a opção for consultar a lista de leilões é apresentada no ecrã a lista dos leilões. Esta lista é obtida fazendo um pedido ao servidor com a palavra inicial *getLeiloes*. Como resposta, o servidor envia informação sobre todos os leilões (Id do Leilão, Descrição, Preço Base e Licitação Atual) que é depois armazenada num *ArrayList<String>* e impressa no ecrã. Nos leilões em que o comprador autenticado efetuou a licitação mais alta, no final da linha respetiva aparece o caracter “+”.

Se a opção for licitar num leilão é-lhe pedido o ID do leilão no qual pretende licitar e o valor da licitação. Depois de inseridos esses dados, é enviada ao servidor uma *string* com a palavra inicial *licita*, seguida desses dados. O servidor valida os dados da licitação e caso esta seja válida ela é registada no servidor e é retornado ao cliente o ID do leilão (o ID é sempre um inteiro positivo). No entanto, uma licitação pode não ser válida caso o valor licitado seja menor que o preço base ou menor que a licitação mais alta até ao momento. Nestes casos, o servidor retorna ao cliente um número negativo, indicando que a licitação não é válida.

Vendedor

Caso o utilizador seja um vendedor é lhe apresentado um menu que lhe permite consultar a lista de leilões, adicionar um novo leilão, terminar um leilão ou simplesmente sair da conta.

Caso a opção selecionada seja consultar a lista dos leilões é enviada uma *string* ao servidor com a palavra inicial *getLeiloes* seguida do ID do utilizador autenticado. São enviadas do servidor as *strings* com o Id do Leilão, a Descrição, o Preço Base e Licitação Atual de todos os leilões em curso. Nos leilões em que o vendedor autenticado é o criador, no final da linha respetiva aparece o caracter “ * ”.

Se a opção for adicionar um leilão são pedidos ao vendedor o Preço Base do produto a vender e uma Descrição com o código *addLeilao*.

Caso o vendedor queira terminar um leilão tem que inserir o ID do leilão e é enviada uma mensagem ao servidor com o código *terminaLeilao*. Se o ID for válido o leilão é terminado e o vendedor e todos os licitadores do leilão são notificados com o valor de fecho do leilão e o *username* do vencedor. Caso contrário o sistema imprime uma mensagem informando que o ID não é válido.

Servidor

O servidor está encarregado de fazer toda a gestão dos pedidos dos vários clientes e comunica com o cliente via *sockets* TCP baseado em texto, mantendo em memória a informação relevante descrita em cima. Cada cliente tem associado a si duas *threads* - uma para as notificações e outra para responder aos pedidos dos clientes. Cada *thread* não escreve em mais do que um *socket*.

A informação utilizada para as notificações (encerramento de leilões) está armazenada no servidor onde existe um *HashMap<Integer,ArrayList<String>>* que a cada *userID* faz corresponder uma lista de notificações. Esta informação é gerida pelas classes *Handler* e *Notificator* em que primeira é responsável por adicionar ao *HashMap* as notificações sempre que um leilão terminar a segunda é responsável por enviar para o cliente as notificações correspondentes que, após envio, são eliminadas do *HashMap*. No caso em que os licitadores estão autenticados aquando do fecho do leilão, a notificação aparece imediatamente no ecrã. Caso contrário serão notificados após próxima autenticação.

Assim, o servidor é composto pelas seguintes partes:

- *Notificator* - Responsável por enviar as notificações aos clientes;
- *Handler* – Responsável por gerir os pedidos do cliente e encaminhar para a respetiva função;
- *Server* – Função *main* do Servidor;
- Sistema – Todas as funções e variáveis relativas ao sistema. É nesta parte que estão declarados o *ArrayList* dos utilizadores, o *ArrayList* dos leilões, o ID do último leilão e o ID do último utilizador inseridos no sistema e o *HashMap* das notificações.

Comentários e Conclusão

Como conclusão salientamos os seguintes aspetos:

- Foram cumpridos todos os requisitos do sistema.
- O sistema é resistente a clientes lentos.
- Independentemente de estar ou não autenticado, ou seja, online no momento do término de um leilão, um licitador é sempre informado do término de um leilão em que participou,