

Instituto Federal de Educação, Ciência e Tecnologia de Minas Gerais

Graduação em Ciência da Computação

Formiga, Minas Gerais, Brasil

kimberlylcf@gmail.com

rodrigossoualves7@gmail.com

viniciusbatista023@gmail.com

Kimberly Lamounier Campos Ferreira - 0016190

Rodrigo Sousa Alves - 0011264

Vinicius de Laet Duarte Batista - 0011918

Serviço de Leilão Eletrônico

Para executar o sistema basta abrir um terminal na pasta principal onde se encontram os arquivos e executar o arquivo run.sh da seguinte maneira ./run.sh.

Requisitos do Sistema

ETAPA 1) o SD deverá prover as seguintes funcionalidades **BÁSICAS**:

- Prover alguma interface com o usuário (CLI ou GUI) para acesso rápido às funcionalidades do serviço de leilões; - **OK**
 - O sistema provê uma interface gráfica em CLI, onde é possível escolher se o cliente irá iniciar o sistema como cliente comum ou como cliente gerenciador de réplicas do sistema (por meio de uma flag para determinar se irá tratar comandos recebidos referentes a manipulação de arquivos ou se apenas irá ignorar essas mensagens, atuando por definição como um aplicativo nos modelos de Gerenciamento de Réplicas).
- Definição de um identificador único por usuário do sistema, de maneira que um mesmo usuário seja corretamente identificado ao sair e retornar; - **OK**
 - No sistema o nome de usuário é um identificador único, e mediante senha, é realizado o login, garantindo que um único usuário esteja logado no sistema por vez (garantia obtida utilizando o armazenamento persistente da lista de usuários já cadastrados em um arquivo .json);
- Cadastro de novo item a ser leilado, sem duplicidade, em acordo dos membros (todos ou maioria); - **OK**
 - O cadastro de um novo item é realizado junto da criação de uma nova sala, já que só é permitido o leilão de um item por sala;

- Criação de nova sala de leilão, identificada pelo item a ser leilado e coordenada por algum membro, denominado "leiloeiro"; **OK**
 - A criação de uma sala é realizada por um *Membro* (usuário logado e conectado ao canal), que se torna o "leiloeiro" daquela nova sala.
 - A criação da sala só é realizada junto ao cadastro de um novo *Item*.
 - A identificação das salas é realizado com o auxílio do Counter do JGroups; que atribui a cada sala um *id* único;
- Inclusão de novo membro em determinada sala de leilão, realizada através do leiloeiro da sala. - **ANDAMENTO**
- Envio de lances de um usuário somente para os outros usuários que estejam naquela sala de leilão; - **ANDAMENTO**
- O resultado do leilão de um item deve ser visível para todos os demais membros do cluster JGroups; - **ANDAMENTO**

ETAPA 2) o SD deverá prover as seguintes funcionalidades **INTERMEDIÁRIAS**:

- Armazenamento persistente do estado do sistema (cadastros, alterações); - **PARCIAL**
 - O sistema realiza o armazenamento persistente de alguns dados, como:
 - Login: *nome* de usuário e *senha*;
 - Membro: *nome* dos usuários conectados ao canal;
 - Lance: *nome* do Membro que realizou o lance, e o *valor* do lance;
 - O sistema de persistência garante cadastro, atualização e remoção dos dados.
- No reingresso de um membro, deverá ser obtido o estado do sistema, ou seja, atualizações que ele possa não ter recebido enquanto estava desconectado; - **ANDAMENTO**
 - O sistema já possui os dados armazenados em disco, falta realizar a distribuição dos mesmos;
- O serviço de leilões deverá prover mecanismos de segurança, como criptografia das mensagens trocadas, autenticação dos usuários e autenticidade de seus lances. - **PARCIAL**
 - A criptografia de mensagens é garantida graças ao ENCRYPT importado no xml, que ao ser implementado na pilha de protocolos, garante o envio de mensagens criptografadas;
 - Autenticação dos usuários é garantida graças ao armazenamento persistente dos usuários cadastrados e membros logados;
 - A autenticidade de lances distribuída não foi implementada, mas o registro/histórico de lances sim, foi realizado de modo persistente (local).
- O serviço deverá ser capaz de continuar funcionando caso haja particionamento na rede, devendo consolidar os estados das partições quando se reunirem. - **PENDENTE**
 - Não implementado.

EXTRA) OPCIONALMENTE, o SD poderá prover as seguintes funcionalidades **AVANÇADAS**:

- O serviço de leilão pode utilizar (sub)canais de comunicação mais leves e baratos (ForkChannel), para implementar algumas funcionalidades (ex.: sala de leilão um de item); - **PENDENTE**
- O serviço de leilão pode ser capaz de permitir que usuários que estejam em LANs diferentes possam se conectar ao mesmo cluster, lidando com peculiaridades da infraestrutura de rede; - **PENDENTE**
- O serviço de leilão pode ser capaz de comprimir mensagens grandes para agilizar a sua transferência; - **OK**
 - A compressão de mensagens é garantida graças ao COMPRESS importado no xml, que ao ser implementado na pilha de protocolos, garantindo a compressão das mensagens enviadas;
- O serviço de leilão pode implementar um limitador de taxa de dados (quantidade de dados enviada por unidade de tempo), evitando o congestionamento do serviço ou ataques de negação de serviço (DoS); - **OK**
 - O limitador de taxa de dados é garantida graças ao RATE_LIMITER importado no xml, que ao ser implementado na pilha de protocolos, garantindo proteção contra negação de serviço;
- O serviço de leilão pode ser capaz de realizar controle de fluxo, ajustando a taxa dos transmissores à taxa do receptor mais lento; - **OK**
 - O controle de fluxo é obtido graças ao MFC importado no xml, que ao ser implementado na pilha de protocolos;
- O serviço de leilão poder ser capaz de fragmentar mensagens de aplicação muito grandes em mensagens menores, remontando-as no receptor, agilizando assim sua transferência e evitando fragmentação a nível da camada de rede. - **OK**
 - A fragmentação de mensagens é garantida graças ao FRAG2 importado no xml, que ao ser implementado na pilha de protocolos.