

Tecnologias de Distribuição e Integração (TDIN)

Trabalho # 1
.NET Remoting
2014-15

Cenário

Pretende-se desenvolver um sistema (*Diginote Exchange System*) de compra/venda e cotação de valores digitais chamados '*diginotes*', baseado em *.NET Remoting*. Apesar de alguma semelhança das '*diginotes*' com '*bitcoins*', para simplificar os seus aspetos de segurança, o mesmo sistema de compra/venda centraliza a existência das '*diginotes*' e quem é o seu dono atual. Cada '*diginote*' é representada por um objeto, tem um número de série único, e um valor (também para simplificar, sempre igual a 1). As que existem estão registadas no sistema (através do seu número de série), assim como o dono atual. Os utilizadores registados no sistema (nome, *nick name* e *password*) deverão ser os atuais donos e quaisquer outros que pretendam adquirir '*diginotes*'. O sistema permite apenas a venda e compra de '*diginotes*', verificando a sua existência, e tomando nota do novo dono quando há uma transação. Em cada instante há uma cotação (para iniciar deverá ser igual a €1.00). Quem pretende comprar ou vender '*diginotes*' à cotação atual emite uma ordem de compra ou venda respetivamente, fazendo o sistema o emparelhamento entre compradores e vendedores, de acordo com regras específicas enunciadas em operações.

Operações

- *Login/logout*/registo no sistema – Os utilizadores deverão identificar-se perante o sistema (ou registar-se) antes de poderem efetuar qualquer outra operação.
- Cotação – Cada utilizador, quando *logged in* e através da sua aplicação quando ligada, deve saber sempre a cotação atual em tempo real.
- Ordem de venda – os detentores de '*diginotes*' podem emitir uma oferta de venda especificando a quantidade a vender. Se houver nesse momento ofertas de compra para essa quantidade, ou apenas para uma parte, a operação é concretizada à cotação atual na quantidade possível. No caso das ofertas de compra não serem suficientes o vendedor deverá indicar o valor de venda, para a quantidade não vendida, que só poderá ser igual ou menor que a cotação atual e passando a ser a nova cotação.
- Ordem de compra – os utilizadores que pretendem comprar '*diginotes*' podem emitir uma ordem de compra indicando a quantidade a comprar. Se houver oferta total ou parcial a correspondente transação é efetuada à cotação atual. Caso a oferta não seja suficiente o emitente da ordem terá de especificar um valor igual ou maior que a cotação atual, passando a ser a nova cotação.
- Em qualquer momento os emitentes de ordens de venda ou compra ainda não satisfeitas podem respetivamente descer ou subir o valor das suas ordens.
- Sempre que a cotação desce e haja outras ofertas de venda pendentes, estas ficam suspensas por um período de tempo (para testes vamos usar 1 minuto), podendo o emitente dessas ordens confirmar o novo valor ou retirar a ordem. Findo o período de suspensão e não havendo qualquer resposta do emitente esta mantém-se à nova cotação. Do mesmo modo, sempre que a cotação sobe e haja outras ofertas de compra não satisfeitas, estas são suspensas para

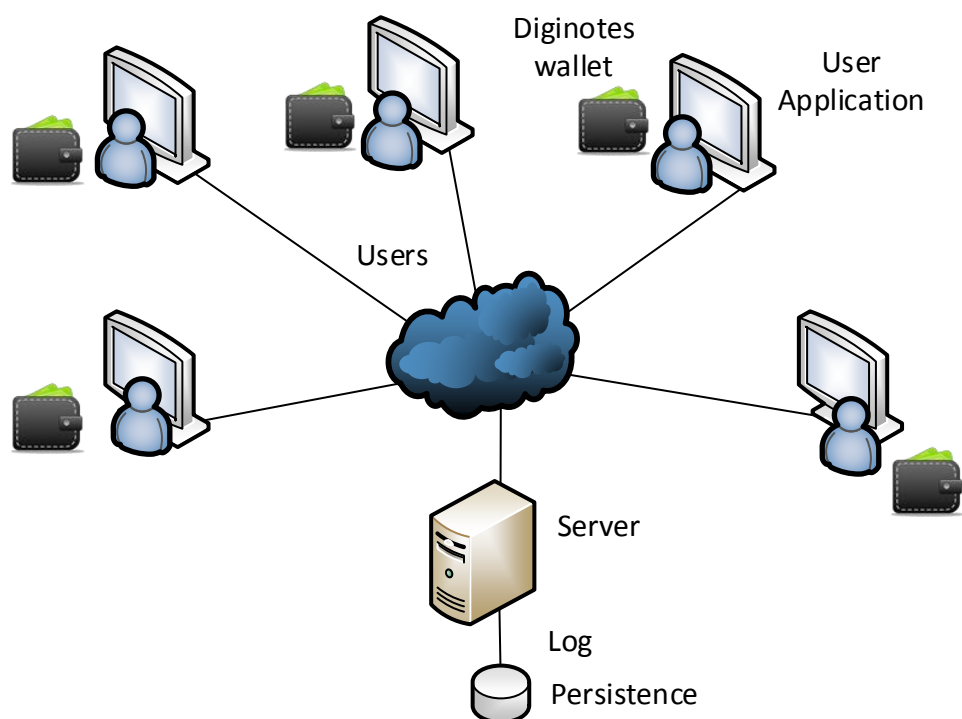
confirmação ou retirada, presumindo-se confirmadas à nova cotação se não houver resposta no período de suspensão.

- Quando se procede a uma transação o sistema deverá sempre começar por satisfazer as ordens ativas mais antigas.

- Quando de uma ordem de venda os números de série das *'diginotes'* a vender devem ser confirmadas pelo sistema (a sua existência e de que o dono é o emitente). Quando se concretiza uma transação, e logo que possível, esses números de série são indicados ao vendedor e comprador. Regista-se também o novo dono das *'diginotes'*.

- O sistema deve emitir um *log* de todas as transações (pode ser um ficheiro de texto) e persistir a informação necessária para poder recuperar de uma falha ou *'shutdown'*.

- Todas as mudanças relevantes devem ser mostradas aos utilizadores ativos no sistema, no mais curto espaço de tempo após terem lugar.



Implementação

Deverá implementar um demonstrador deste projeto usando *.NET Remoting* e desenvolvendo as aplicações GUI e objetos remotos necessários, utilizando as suas boas práticas. As aplicações devem ter uma interface simples, fácil de usar e funcional. Poderá adicionar as funcionalidades que considere convenientes, por exemplo visualização de estatísticas ou gráficos ...

A satisfação de todos os requisitos e regras, interface e facilidade de operação, conjunto de funcionalidades e correto uso das boas práticas são fatores relevantes.

Relatório

Deverá ser entregue um pequeno relatório descrevendo a arquitetura usada (aplicações, módulos, objetos e sua interação), funcionalidades incluídas, testes executados, e modo de funcionamento (capturas dos principais fluxos de utilização).

Deverá conter também todas as instruções para construir e usar todo o demonstrador