



Prof.^a Ana Cristina Barreiras Kochem Vendramin

Universidade Tecnológica Federal do Paraná (UTFPR), Câmpus Curitiba
Engenharia de Computação, Sistemas de Informação
Programa de Pós-Graduação em Computação Aplicada (PPGCA)
Departamento Acadêmico de Informática (DAINF)

Avaliação (valor 2,0)

Inserir transações e controle de concorrência na aplicação 2 (Java RMI / Pyro) ou 3 (Web Services).

Controle de Concorrência (valor 0,3):

- Como a aplicação em questão será *multithread*, é possível que algum recurso possa ser acessado por duas ou mais *threads* ao mesmo tempo. Por esse motivo, se faz necessário a implementação de um mecanismo que controle o acesso concorrente aos recursos compartilhados;
- Lembre-se que a sincronização pode afetar o desempenho da aplicação. Então, como regra deve-se manter bloqueios pelo menor tempo possível e apenas onde for necessário.

Transações (valor 1,7):

- Cada par de operações de compra e venda de um ativo deve ser tratado como uma transação. Deve-se garantir que as quatro propriedades ACID sejam atendidas: Atomicidade, Consistência, Isolamento e Durabilidade. Deve-se evitar: escritas prematuras, leituras sujas, cancelamentos em cascata e impasses. Não é permitido o uso de banco de dados. Durabilidade: dados devem ser persistidos em arquivos. Coordenador e participantes devem se comunicar através do protocolo *two-phase commit*. Um coordenador pode ser um participante. Cada participante da transação deve implementar, ao menos, os seguintes métodos: votar se pode ou não realizar uma operação da transação (preparar), efetivar uma transação e abortar uma transação. O Coordenador deve implementar, ao menos, os seguintes métodos: abrir uma transação, obter estado de uma transação (valor 1,0);
- Participantes e coordenador devem registrar logs das transações. O sistema deve se manter consistente em caso de falhas em algum processo com transação ativa (valor 0,7).

Observações: Desenvolver uma interface com recursos de interação apropriados.