



## **Trabalho 2 – Modelo Cliente/Servidor (RPC/RMI)**

O objetivo deste trabalho é desenvolver um sistema distribuído baseado no modelo Cliente/Servidor utilizando RPC/RMI que simule a criação de contas de clientes e transações bancárias nestas contas. A escolha da linguagem de programação está liberada desde que sigam o modelo RPC/RMI. O sistema deve ter no mínimo 3 tipos de processos: administração, agência e caixa automático. Devem ser implementadas as seguintes funcionalidades:

- Processo Administração (servidor): Realiza abertura e fechamento de contas (para agências), e também executa as operações de saque, depósito e consulta de saldo destas contas (para agências e caixas automáticos). Deve garantir semântica de execução *exactly-once* para operações que sejam não-idempotentes e controle de concorrência para recursos compartilhados;
- Processo Agência (cliente): Solicita abertura e fechamento de contas e também pode solicitar depósito, retirada e consulta de saldo em conta existente. Abertura de conta, depósito e retirada devem ser operações garantidamente não-idempotentes (semântica de execução *exactly-once*);
- Processo Caixa Automático (cliente): Solicita depósito, retirada e consulta de saldo em conta existente. As duas primeiras devem ser operações garantidamente não-idempotentes (semântica de execução *exactly-once*) mesmo que ocorra algum erro na confirmação da operação (simular com injeção de falhas).

A avaliação do trabalho será feita com base no acompanhamento do desenvolvimento do trabalho em laboratório, na apresentação do trabalho em aula e no envio de um relatório técnico no Moodle. O relatório deve focar em:

- descrever como o problema foi mapeado no modelo cliente/servidor usando RPC/RMI;
- descrever como foi implementado e testado o controle de concorrência;
- descrever como foi implementado e testado o controle de idempotência.

Formato do relatório técnico:

- arquivo formato .pdf;
- cabeçalho reduzido com identificação do grupo e do trabalho;

- primeiras duas páginas em coluna dupla com margens reduzidas (1cm) e fonte 10;
- terceira página com dumps de tela mostrando a execução dos programas em funcionamento;
- a partir da quarta página código fonte formatado em coluna simples dos programas utilizados no trabalho (sem limite).

#### **Critérios de Avaliação:**

- Implementação Funcional (2,0 pontos): processos e responsabilidades corretas no uso do modelo RPC/RMI (1,0), implementação completa das funcionalidades (1,0);
- Controle de Concorrência (2,0 pontos): implementação de controle de concorrência (1,0), experimentos para demonstrar correto funcionamento (1,0);
- Controle de Idempotência (3,0 pontos): implementação de garantia de semântica exactly-once (1,0) e tratamento de falhas de rede (1,0), simulação/injeção de falhas para testar semântica exactly-once (1,0).
- Relatório Técnico (1,0 ponto): descrição do mapeamento cliente/servidor usando RPC/RMI (0,5), explicação do controle de concorrência e do controle de idempotência controle de idemposegurança (0,5).
- Apresentação em Aula (2,0 pontos): demonstração das funcionalidades (1,0), conhecimento sobre a implementação (1,0).

#### **Grupos:**

O trabalho pode ser realizado em grupos de no mínimo 2 e no máximo 4 integrantes. É responsabilidade dos alunos formarem os grupos, que devem ser organizados/cadastrados na ferramenta do Moodle (Auto-seleção de grupos).

#### **Entrega:**

Submeter o reletório, todo o código fonte desenvolvido e um documento com instruções de compilação e uso, na forma de um arquivo .zip, no seguinte padrão de nome T2\_NomeAluno1NomeAluno2.zip, na sala de entrega do Moodle. O prazo limite para entrega está definido na atividade do Moodle.