

DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO
BACHARELADO EM INFORMÁTICA
SISTEMAS OPERACIONAIS I
2^o SEM/2011
Trabalho em Grupo – Nr 3

CONSTRUÇÃO DE APLICATIVO CLIENTE – SERVIDOR USANDO PROCESSOS MULTI-THREADS E SEMÁFOROS

1. Objetivo do Trabalho

Estimular o aluno a trabalhar em equipe para projetar e desenvolver soluções para problemas que envolvam o estudo e conhecimento de threads, semáforos para controle de exclusão mútua e sincronização e uso de comunicação via socket.

2. Escopo do Trabalho

- ✓ Especificar as Regras do Negócio;
- ✓ Realizar a Modelagem da Aplicação;
- ✓ Conceber a lógica, implementar e documentar os algoritmos necessários;
- ✓ Elaborar um relatório técnico descritivo da aplicação contendo, pelo menos, a descrição:
 - Das Regras do Negócio;
 - Da Modelagem da Solução;
 - Da Lógica de Semáforos implementada;
 - Das Dificuldades encontradas;
 - Dos Resultados alcançados;
 - Código fonte documentado (em anexo).
- ✓ Estar preparado para fazer uma demonstração prática do funcionamento simulado do aplicativo desenvolvido.

Entregar todo o material elaborado, fontes, executáveis, relatório e apresentação, em CD ou DVD.

3. Equipes de Trabalho

Devem ser formadas preferencialmente com um máximo de 4 e um mínimo de 3 alunos cada.

4. Prazo de Entrega do Trabalho

Todos os grupos deverão entregar uma cópia do CD impreterivelmente até o dia **30/11**.

5. Penalidades

Caso o grupo atrase a entrega do CD, seu grau final sofrerá um decréscimo na razão de 0.5 pontos por dia.

6. Avaliação

Serão considerados os seguintes aspectos:

- ✓ Forma e conteúdo do material escrito;
- ✓ Qualidade da solução adotada;

- ✓ Qualidade do material para apresentação em sala;
- ✓ Participação dos membros do grupo;
- ✓ Cumprimento do tempo da apresentação;
- ✓ Objetividade e desenvoltura na apresentação.

7. Desenvolvimento

Na página é disponibilizado um tutorial sobre sockets e exemplos para comunicação através de uma rede TCP/IP.

Todos os aplicativos devem ser multithread e podem ser desenvolvidos em C ou Java para os sistemas operacionais Linux, Windows XP ou 7.

Na modelagem dos respectivos sistemas, o grupo deve levantar os requisitos operacionais necessários para a boa operação das diferentes funções e verificar a consistência das mesmas.

8. Temas

A seguir são descritos três temas distintos, sendo que por sorteio, cada equipe receberá como tarefa apenas um dos respectivos temas.

Tema 1 – Servidor de Arquivos Remoto

Projetar e construir um servidor de arquivos remoto que poderá receber e enviar arquivos a um ou mais clientes através de uma conexão socket em rede TCP/IP.

O servidor deve estar preparado para atender clientes sob demanda, em paralelo. Cada cliente deverá ser atendido por uma *thread* distinta no servidor.

Características da aplicação:

▪ **Módulo Cliente**

- a) Recebe como parâmetro de entrada o endereço IP do servidor

ex: `./cliente 192.168.0.1`

- b) Deve exibir uma interface com as diferentes funções disponíveis e permanecer em loop para execução interativa das funcionalidades até que o usuário tecle a função de saída. As funcionalidades disponíveis devem ser:

- ✓ Listar conteúdo do diretório no servidor
- ✓ Listar o diretório local
- ✓ Enviar arquivo para o servidor
- ✓ Enviar arquivos por extensão
- ✓ Receber arquivo
- ✓ Receber arquivos por extensão
- ✓ Renomear arquivo no servidor
- ✓ Apagar um arquivo no servidor
- ✓ Apagar arquivos por extensão
- ✓ Pegar dados sobre o arquivo (data da criação, da última modificação e do último acesso)

- c) Ao solicitar um arquivo, o cliente deverá informar se este arquivo será ou não alterado (modo = r ou w). Dois ou mais clientes podem abrir um mesmo arquivo no modo de leitura “r”. Um cliente somente pode solicitar um arquivo no modo escrita (w) se este não estiver sendo usado por nenhum outro cliente. Um arquivo liberado no modo “w” deve ficar bloqueado para leitura e escrita até que seja restituído pelo cliente que o estiver utilizando.
- d) A função “renomear” só pode ser permitida para arquivos que não estejam bloqueados para escrita (w) nem leitura (r).

▪ **Módulo Servidor**

- a) Aguarda solicitações de um ou mais clientes e dispara uma thread para atender cada uma destas solicitações.
- b) Não tem interface com o usuário.
- c) Deve informar na tela, modo console, o *log* as operações à medida que são realizadas.
- d) É encerrado mediante comando explícito.
- e) Deve permitir leituras concorrentes para um mesmo arquivo.
- f) Se um arquivo recebido possuir o nome de um arquivo já existente no servidor e que não tenha sido lido anteriormente pelo próprio cliente para atualização, o servidor deverá enviar ao cliente uma notificação de erro.
- g) Deve garantir consistência operacional.

O grupo deve preparar uma demonstração com 01 servidor e pelo menos 03 clientes concorrentes preferencialmente em diferentes estações de trabalho.

Tema 2 – Terminal de Auto-Atendimento

Modelar e construir um sistema de atendimento a correntistas de uma rede bancária através de terminais de auto-atendimento. As regras de negócio prevêm:

- a) O cliente deve se identificar fornecendo os dados do cartão e sua senha de 6 dígitos (nunca um mesmo cliente pode estar sendo atendido em dois terminais simultaneamente);
- b) Uma vez *logado* o cliente pode fazer:
 - ✓ Transferências para diferentes contas de diferentes agências do mesmo banco, limitadas a um valor acumulado por dia de R\$5.000,00;
 - ✓ Docs para contas em outros bancos, limitados a um valor acumulado de R\$3.000,00. Cada Doc tem um custo de R\$10,00;
 - ✓ Até 3 retiradas em espécie por dia, independente do terminal, limitadas a um valor acumulado de R\$2.000,00;
 - ✓ Depósitos em espécie unicamente nas agências do próprio banco, que são “automaticamente conferidos” e creditados na conta corrente do beneficiário;
- c) Uma conta solidária pode ter até 3 titulares;

- d) Contas solidárias aceitam operações simultâneas, porém os limites para cada cliente ficam reduzidos em 20% no caso de 2 titulares e 30% no caso de 3.
- e) A comunicação dos terminais de auto-atendimento com o servidor deve ser feita via socket em rede TCP/IP.

Tratando-se de uma operação remota, a principal preocupação dos desenvolvedores do sistema deverá estar na confiabilidade oferecida pelo mesmo às operações realizadas.

O grupo deve preparar uma demonstração com 01 servidor e pelo menos 03 terminais clientes simulados, preferencialmente, em diferentes estações de trabalho.

Tema 3 – Agencia Bancária

Projetar e construir um sistema para simular o atendimento presencial a correntistas de uma agência bancária. As regras de negócio prevêm o atendimento pessoal do cliente na agência, sendo que ao cliente é permitido:

- f) Sacar até R\$1.000,00 se cliente da agência e tiver saldo na conta ou no cheque especial;
- g) Sacar até R\$500,00 se cliente de outra agência e tiver saldo na conta ou no cheque especial;
- h) Depositar dinheiro na sua conta ou na conta de outro cliente do banco, mesma agência ou não.
- i) Transferir dinheiro da sua conta para de outro cliente do banco, mesma agência ou não.

Considere um cenário com 01 agência, a AgDCC, onde ocorre o atendimento aos clientes e 03 outras agências remotas, cada uma das agências possui até 10 clientes cadastrados.

A chegada dos clientes na AgDCC deve ser simulada de forma aleatória (clientes chegando em intervalos de tempo aleatório), sendo que a probabilidade do cliente pertencer a própria agência é de 70%. Ao chegar na agência o cliente entra em uma fila única para atendimento. A agência possui 3 caixas para atendimento.

Para saques ou depósitos em contas de outra agência, o caixa consulta um servidor remoto instalado na outra agência.

A AgDCC atende 1000 clientes por dia em média, que costumam chegar 30% em intervalos de 1 u.t. (unidade de tempo), 50% em 2 u.t. e 20% em 3 u.t..

O grupo deverá montar um cenário com as entidades mínimas e os requisitos necessários a operação do sistema de agências e clientes conforme descrito no problema.

Lógica da aplicação:

O grupo deve criar um cenário para simulação com a AgDCC rodando em uma máquina e duas outras agências rodando em uma ou duas outras máquinas. Na AgDCC deve ser gerada uma interface ou criado um script para rodar a simulação.

Na simulação os clientes devem chegar à agência segundo as probabilidades acima definidas, entrar na fila de atendimento. Ao ser atendido por um caixa, o cliente aleatoriamente deve fazer um saque, um depósito ou uma transferência de um valor gerado aleatoriamente segundo as restrições do problema.

Se a operação envolver qualquer acesso a agências remotas, este deve ser feito através de conexão socket com o endereço IP dos respectivos servidores destas agências.

Cada agência deve manter dois arquivos texto, um com as informações da conta de cada um dos seus clientes e outro com o registro de todas as movimentações realizadas com ou sem sucesso.

Cada agência deve também manter uma tabela com o endereço IP das outras agências.