**PRIMEIRO PROJETO – ERI-A**

**DIRETÓRIO DE SERVIÇOS - Peso 2**

Individual ou dupla

Data limite para entrega: 07/05/2013

**PROJETO BÁSICO** (5 pontos)

Em redes grandes como na Internet, por exemplo, é impossível ou inviável que os clientes armazenem localmente os endereços dos servidores. Isso porque, além de haver um número muito grande de servidores, novos serviços são sempre disponibilizados e muitos servidores saem do ar. Além disso, podem servir de base para ambientes de *Web Services* ou *Cloud Mashups*, por exemplo.

Assim, existem **meta-servidores** (**diretórios**, ***portmappers***, **mapeadores de portas** ou simplesmente **portais**) que armazenam os endereços dos servidores e os disponibilizam aos clientes.

Este projeto consiste na implementação de um sistema de diretório (*portmapper*) que cadastra os serviços e servidores através de uma lista ligada e presta informações sobre a localização dos servidores aos clientes. A lista deve ser organizada por tipo de serviço prestado. Para cada tipo de serviço devem estar associados os endereços dos servidores que prestam aquele serviço.

No projeto básico, a interação dos clientes e servidores com o *portmapper* pode ser simulada através de um menu de opções com pelo menos as seguintes funções:

* Inclusão de um novo serviço (IP e porta do servidor);
* Remoção de um serviço existente;
* Consulta de um serviço;
* Listagem de todos os serviços;
* Listagem de servidores por tipo.

**OPCIONAL 1 – Balanceamento de Carga** (2 pontos)

Para não sobrecarregar um só servidor, o *portmapper* deve alternar os endereços dos servidores quando vários clientes solicitam determinado serviço.

**OPCIONAL 2 - Tolerância a Falhas A** (3 pontos)

Deve haver uma opção para armazenar e recuperar de um arquivo os dados armazenados no *portmapper*.

**OPCIONAL 3 - Tolerância a Falhas B** (1 ponto)

Todas as alterações realizadas pelo *portmapper* devem ser gravadas em um arquivo de *log.*

**OPCIONAL 4 – Manutenção da Consistência** (2 pontos)

Um mecanismo para a manutenção da consistência (o diretório deve informar somente endereços de servidores ativos) é fazer com que os servidores se cadastrem periodicamente no *portmapper*, em um mecanismo também conhecido como ***heartbeat***. Para simulá-lo, implemente uma função para o recadastramento de servidores e outro para o diretório remover os servidores que não se recadastraram há muito tempo.

**OPCIONAL 5 – Recuperação dos dados de um servidor** (1 ponto)

Caso um mesmo servidor saia do ar e retorne à atividade, seus dados anteriores serão recuperados.

**OPCIONAL 6 – Relatório de Demanda** (1 ponto)

O sistema emitirá relatórios de demanda, que devem informar quais servidores foram informados aos clientes e quantas vezes.

**OPCIONAL 7 – *Garbage Collection*** (1 ponto)

Após um período de tempo determinado pela equipe, caso o servidor não tenha se recadastrado, o sistema irá remover definitivamente seus dados do sistema.

**OPCIONAL 8 – Execução “Real”** (5 pontos)

A execução será realizada por um processo *portmapper*, além de processos clientes e servidores. Os processos deverão enviar mensagens como meio de comunicação.

**OBSERVAÇÕES**

1. No início do arquivo que contiver o programa fonte, deve haver o seguinte cabeçalho preenchido pelo grupo:

Nomes: RAs:

Opcionais funcionando:

Valor do Projeto: \_\_\_\_\_\_\_ pontos

1. O trabalho vale até 10 pontos. Os grupos podem implementar quantos opcionais desejarem, mas a nota do projeto será limitada a 10 pontos. Dessa forma, cada grupo deve escolher um conjunto de opcionais, cujos pontos serão somados aos 5 pontos do projeto básico (obrigatório). Cada grupo tem a liberdade de escolher os opcionais que considerar mais convenientes.
2. O trabalho deverá ser elaborado em linguagem C, a menos que seja previamente combinado de outra forma com o professor.
3. Os projetos poderão ser apresentados e/ou modificados em laboratório em data a ser definida.
4. Não serão aceitos trabalhos após a data de entrega.