

Trabalho Prático 1 – Threads e Objetos Distribuídos, 5 valores.**Grupos de 4 alunos;**

- Submeter no Moodle o código fonte do projeto, mais relatório, até dia 25 de Abril às

23:59.

- Defesas: 27, 28 e 29 de Abril.

- Sistema Login ??
- 2 Clientes: Produtores e Consumidores
- Servidor RMI => Notícias
- Servidor Backup/Arquivo
- Ficheiro de conf com limite de tópicos (guardar tópicos) ???
- Ficheiro Backup para 50% limite de notícias inseridas no servidor
- Notícias serem guardadas em file

Enunciado:

Pretende-se criar um servidor de notícias que terá dois tipos de processos clientes: os produtores de notícias (**publishers**) e os consumidores de notícias (**subscribers**).

Os produtores de notícias registados deverão poder:

- P1 - Adicionar um tópico (por exemplo desporto, politica nacional, educação, ...);
- P2 - Consultar tópicos existentes;
- P3 - Inserir uma notícia de um dado tópico;
- P4 - Consultar todas as notícias publicadas.

Os consumidores de notícias registados deverão poder:

- C1 - Subscrever um tópico (a operação pode ser repetida tantas as vezes quantos os tópicos disponíveis);
- C2- Consultar notícias de um dado tópico entre um determinado intervalo de datas. Se para o intervalo pedido, as notícias já estiverem em arquivo, receberá o endereço (IP + porto) de um servidor de arquivo ao qual poderá solicitar as notícias em falta;
- C3- Consultar a última notícia de um dado tópico.

Poderão existir consumidores de notícias não registados que podem executar as operações C2 e C3.

O servidor de notícias deverá ser um servidor RMI, com um, ou mais, objetos remotos que permitam responder às operações descritas.

- Sempre que um produtor publicar uma notícia de um dado tópico, deve ser enviada uma notificação (callback) a todos os consumidores registados que subscreveram esse tópico (tratar situações em que o consumidor não está ligado).

- O número de notícias para cada tópico deve ser limitado por um parâmetro dado num ficheiro de configuração (o limite é o mesmo, qualquer que seja o tópico).

- As notícias devem ser guardadas em ficheiros de objetos.

- Quando o limite do número de notícias de um dado tópico é atingido, 50% das notícias desse tópico deverá ser copiado para um ficheiro de backup sendo eliminadas do ficheiro original.

3
4

Uma **notícia** está sempre associada a um tópico, associada a um produtor, tem um limite de 180 caracteres e tem um **timestamp** correspondente à data e hora em que é recebida pelo servidor.

Os processos clientes (produtores e consumidores) terão uma **interface** de texto com o utilizador.

O **servidor de backup** deverá ser um **servidor multi-threaded** com o qual os processos consumidores poderão comunicar por **sockets** para **solicitar notícias em arquivo**.

Na versão final deverá usar as **permissões adequadas** evitando o uso de **allPermission**.
? ?

Desafios para após todas as funcionalidades estarem tratadas:

- Os processos clientes poderem executar em **máquinas diferentes** da máquina do(s) servidor(es).
- Execução da aplicação **fora do contexto do editor**.

O relatório deverá incluir as secções:

- **Introdução**;
- **Descrição da arquitetura da aplicação**;
- **Detalhes dos processos clientes e servidor**;
- **Um pequeno manual de configuração e instalação**.

No final, o aluno de contacto de cada grupo deve:

- Submeter no Moodle o código fonte do projeto, mais o relatório, até dia 25 de Abril às 23:59*.

(*) Tolerância de algumas horas

Votos de Bom Trabalho