

## Atividade 1 - Programação com Sockets TCP

### Sistemas Distribuídos – Prof. Rodrigo Campiolo

**Questão 1)** Faça um servidor para processar as seguintes mensagens dos clientes. O servidor deve suportar mensagens de múltiplos clientes. Use o TCP. As mensagens de solicitação estão no formato String UTF:

## TIME

\* Retorna a hora do sistema como uma String UTF no formato HH:MM:SS

## DATE \_\_\_\_\_

\* Retorna a data do sistema como uma String UTF no formato DD/MM/AAAA.

## FILES

\* Retorna os arquivos da pasta definida por padrao (p. ex. /home/user/shared)

Formato:

\* retorna um inteiro indicando o número de arquivos (int)

\* envia individualmente o nome de um arquivo como uma String UTF.

**DOWN nome-arquivo**

\* Faz o download do arquivo nome-arquivo

Formato:

\* retorna um inteiro 0 se nome não existe ou retorna o tamanho do arquivo (int) em bytes.

\* recebe byte a byte e grava em um diretório padrão

## EXIT

\* Finaliza a conexão

**Questão 2)** Faça uma aplicação com um servidor que gerencia um conjunto de arquivos remotos entre múltiplos usuários. O servidor deve responder aos seguintes comandos:

-> ADDFILE (1): adiciona um arquivo novo.

-> DELETE (2): remove um arquivo existente.

-> GETFILESLIST (3): retorna uma lista com o nome dos arquivos.

-> GETFILE (4): faz download de um arquivo.

O servidor deve registrar as ações em logs.

**As solicitações possuem o seguinte cabeçalho em comum:**

1 byte: requisição (1) – *Message Type* (0x01)

**1 byte:** código do comando – *Command Identifier (0x01 a 0x04)*

1 byte: tamanho do nome do arquivo – *Filename Size*

variável [0-255]: nome do arquivo em bytes – *Filename*

1 byte		1 byte		1 byte		[0 a 255] bytes	
+	-	+	-	+	-	+	-
	Message Type		Command Ident.		Filename Size		Filename
+	-	+	-	+	-	+	-

**As respostas possuem o seguinte cabeçalho em comum:**

1 byte: resposta (2) – *Message Type* (0x02)

**1 byte:** código do comando – *Command Identifier (0x01 a 0x04)*

1 byte: status code (1-SUCCESS, 2-ERROR) – *Status Code*

```

      1 byte      1 byte      1 byte
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
| Message Type | Command Ident. | Status Code |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

— — —

para o **ADDFILE**, adicionam-se os campos na solicitação:

**4 bytes:** tamanho do arquivo (*big endian order*) em bytes.

variável[1 a  $2^{32}$ ]: bytes do arquivo.

\_\_\_\_\_

para o **GETFILESLIST**, adicionam-se os campos na resposta:

**2 bytes:** número de arquivos (*big endian order*)

Repete-se até terminar os nomes:

1 byte: tamanho do nome (1-255)

variável [1 a 255]: nome do arquivo

----

para o **GETFILE**. adicionam-se os campos na resposta:

**4 bytes**: tamanho do arquivo (big endian order) em bytes

**variável [1 a 2<sup>32</sup>]**: bytes do arquivo.

\* ao fazer download do arquivo, grave em uma pasta padrão.

\* para enviar e receber arquivo façam envio e recebimento byte a byte.