

Sistema de gestão de aulas TCP/IP

Pretende-se implementar um sistema que permita gerir as presenças dos alunos nas aulas e as perguntas e respostas, bem como os ficheiros relevantes para a disciplina.

Para efeitos de simplicidade, assume-se que o servidor é lançado no início de cada aula e começa a contar o tempo decorrido das aulas. Assume-se também que uma aula tem uma duração de 2 horas.

O sistema deve ser implementado com base num servidor TCP que aceita ligações de múltiplos clientes em simultâneo na porta 5555, cada uma correspondendo a um utilizador diferente. O cliente TCP deverá permitir utilizar todas as funcionalidades do servidor, de uma forma amigável ao utilizador (e.g., através de comandos simples ou de um menu de opções).

O sistema utilizará um protocolo simples do tipo pedido-resposta. Cada mensagem trocada não poderá ter mais do que 1024 bytes. Para efeitos deste enunciado, nos exemplos de comunicação entre cliente e servidor serão marcados a **azul** os pedidos do cliente e a **laranja** as respostas do servidor.

Componentes do sistema

1. Gestão de presenças

A primeira componente do sistema deve permitir aos alunos registar a sua presença online, através de um “login” no sistema. Quando o aluno “entra”, o servidor regista a sua presença, seguindo a seguinte regra: se está menos de 20 minutos atrasado, a presença é completa; se está 20 ou mais minutos atrasado, conta meia presença; se está mais de 45 minutos atrasado, não é registada a presença (mas pode aceder normalmente ao resto do sistema).

O registo no sistema é feito através do comando “IAM” (ao que o servidor deve responder com “HELLO”):

```
IAM 112345
```

```
HELLO 112345
```

Implementação **opcional**: autenticação com password; password inválida produz erro:

```
IAM 112345 WITHPASS password123
```

```
ERROR 112345
```

2. Gestão de perguntas e respostas

A componente de perguntas e respostas permite aos alunos colocarem dúvidas durante a aula, que poderão ser respondidas pelos colegas ou pelo professor.

Esta componente suporta 3 comandos diferentes: “ASK”, “ANSWER” e “LISTQUESTIONS”:

```
ASK Quem foi o primeiro rei de Portugal?

QUESTION 1: Quem foi ...

ANSWER 1 D. Afonso Henriques

REGISTERED 1

LISTQUESTIONS

(1) Quem foi o primeiro rei de Portugal?

    (112345) D. Afonso Henriques

    (professor) D. Afonso Henriques “O Conquistador” 1143-1185

(2) Quanto é 6 a dividir por 2?

    NOTANSWERED

ENDQUESTIONS
```

Nota: assume-se que aquando do envio do “LISTQUESTIONS” acima, já ocorreu o envio da segunda questão e da resposta do user “professor” (e.g., “alguém” usou ‘IAM professor’, ‘ASK Quanto é 6 a dividir por 2?’, etc.)

Implementação **opcional**: criar um comando para apagar a resposta, caso esta esteja errada.

3. Gestão de ficheiros

Esta componente permite fazer upload e download de ficheiros relevantes para a aula. Existem 3 comandos: “PUTFILE”, “LISTFILES” e “GETFILE”.

```
PUTFILE livro.pdf 2780

(conteúdo do ficheiro - bytes)

UPLOADED livro.pdf

LISTFILES
```

```
(1) livro.pdf
```

```
ENDFILES
```

```
GETFILE 1
```

```
FILE 1 livro.pdf 2780
```

```
Çoe feohwç heevdqudw dç fç dlskjdaçsaldj...saodqodç idgfgfewiuiqufhgjsakjhgsa898738tgewuqf dks  
gfdkdsajhgskdjhsagkjshgaksagdkj.....
```

(nota: aqui não é necessário informação de término, pois sabemos exactamente quantos bytes tem o ficheiro)

4. Persistência de dados

Para que não haja necessidade de enviar os mesmos dados de cada vez que se reinicia o servidor (por motivos de upgrade ou correcção de falhas), é necessário implementar um sistema para persistência dos dados em ficheiros.

O sistema deve guardar automaticamente os dados, num intervalo pré-definido (por exemplo de 5 em 5 minutos).

Os dados a guardar devem incluir as perguntas e as respostas e, **opcionalmente**, as marcações de presenças anteriores. **Opcionalmente** poderá haver forma de utilizar esta informação de presenças para construir uma folha de cálculo com presenças e faltas, incluindo as datas respectivas.

Implementação

O sistema pode ser implementado em **qualquer linguagem de programação** à escolha dos alunos, utilizando sockets TCP. Todas as funcionalidades descritas acima devem ser implementadas exactamente como apresentadas.

As formas de implementação das funcionalidades ficam ao critério dos alunos e as escolhas e técnicas usadas fazem parte da avaliação. Para este efeito, incluem-se todas as funcionalidades cuja forma de funcionamento não está explícita no presente enunciado.

Todas as funcionalidades apresentadas como **opcionais** são, como o nome indica, facultativas e poderão fornecer pontos bónus quando implementadas correctamente.

É necessário implementar o servidor com as componentes apresentadas acima, bem como um cliente que permita tirar proveito dessas componentes de uma forma “user-friendly”, conforme descrito no início deste enunciado.

Qualquer dúvida de interesse geral deverá ser colocada no fórum específico do moodle, de forma a todos poderem beneficiar da resposta.

Entrega e grupos de trabalho

- O trabalho será elaborado em grupos de 2 alunos;
- A entrega é permitida até às 23:59 do dia 14/06/2023. Após esta data serão descontados 2^{n-1} valores (sendo n o número de dias de atraso);
- O código deve ser organizado em duas directorias, 'srv' e 'cli', onde residem, respectivamente, o servidor e o cliente;
- Na directoria "mãe", haverá um ficheiro 'relatorio.pdf' (sem acentos) correspondente ao relatório de implementação e instruções de funcionamento;
- O relatório deve ser detalhado e explicar o design, a arquitectura e as decisões de implementação tomadas durante a elaboração do trabalho;
- O conjunto de ficheiros e directorias deve ser compactada num ficheiro chamado lnnnnn_lmmmmm.tgz (nnnnn é o menor número dos alunos do grupo e mmmmm é o maior número dos alunos do grupo);
- O ficheiro lnnnnn_lmmmmm.tgz deve ser enviado via moodle pelo aluno nº nnnnn dentro do prazo estabelecido.