

 <p>TECNOLOGIAS E ARQUITETURA</p> <p>Departamento de Ciências e Tecnologias da Informação</p>	<p>Sistemas Operativos Ano Letivo 2020/2021 1º Semestre versão 1.1</p>
---	--

Projeto Cliniq-IUL (Parte 1)

O presente trabalho visa aplicar os conhecimentos adquiridos durante as aulas de Sistemas Operativos e será composto por três partes, com o objetivo de desenvolver os diferentes aspetos da plataforma **Cliniq-IUL**. Iremos procurar minimizar as interdependências entre partes do trabalho. Este enunciado detalha apenas as funcionalidades que devem ser implementadas na parte 1, sendo que os outros dois enunciados serão publicados no decorrer do semestre.

A plataforma **Cliniq-IUL** destina-se à marcação e fornecimento de consultas médicas online. Na Plataforma **Cliniq-IUL**, existem os seguintes conceitos:

- **Paciente:** a pessoa que precisa de uma consulta médica de uma determinada especialidade (ou clínica geral). O paciente tem um determinado perfil – número de identificação, nome, localidade, número de telemóvel, email, e saldo disponível para pagamento das consultas pedidas.
- **Médico:** a pessoa que efetua a consulta de especialidade (ou clínica geral). Um médico tem um determinado perfil – Nome, número de cédula profissional, especialidade médica, email, rating como médico (de 0 a 10, atribuído pelo paciente), número de consultas efetuado, e saldo do médico recebido pelas consultas efetuadas.
- **Pedido de Consulta:** mensagem enviada pelo Paciente a marcar uma Consulta para um determinado médico.
- **Consulta:** Consulta de um Paciente pelo Médico com uma determinada duração e com um determinado custo (preço da consulta).

Em termos gerais, quando um Paciente precisa de uma Consulta, utiliza o seu telemóvel para se ligar ao servidor Tigre e requisitar um Médico disponível. Os médicos ativos (disponíveis) estão ligados ao servidor Tigre a aguardar pedidos de consulta.

Entrega, relatório e autoavaliação

Ao contrário dos anos anteriores, o trabalho de SO este ano será realizado **individualmente**, logo sem recurso a grupos.

A entrega da Parte 1 do trabalho será realizada através da criação de **um** ficheiro ZIP cujo nome é o nº do aluno, e.g., **“a12345.zip”** (**ATENÇÃO:** não serão aceites ficheiros RAR, 7Z ou outro formato) onde estarão todos os ficheiros criados **INCLUINDO** o relatório. O relatório será entregue no formato TXT ou PDF. Não serão aceites documentos DOCX ou outro formato.

A entrega desta parte do trabalho deverá ser feita por via eletrónica, através do e-learning:

- e-learning da UC Sistemas Operativos
- Seleccionam a opção sub-menu “Conteúdo/Content”
- Seleccionem o link “Trabalho Prático 2020/2021 Parte 1”
- Dentro do formulário “Visualizar Exercício de carregamento: Trabalho Prático 2020/2021 Parte 1”, seleccionem “Anexar Arquivo” e anexem o vosso ficheiro .zip. Podem submeter o vosso trabalho as vezes que desejarem, apenas a última submissão será contabilizada.

Observação. A submissão de um ficheiro substitui a versão anteriormente submetida. Realça-se que o formulário acima destina-se apenas à entrega da Parte 1.

Deverá ser criado um relatório em formato TXT ou PDF com informações sobre a elaboração do projeto. O relatório será entregue no formato TXT ou PDF. Não serão aceites documentos DOCX ou outro formato. Esse ficheiro deve chamar-se *relatorio.txt* ou *relatorio.pdf*, deverá ter aproximadamente 1 página A4 e deverá ser submetido dentro do ficheiro ZIP do trabalho.

Política em caso de fraude

Os alunos podem partilhar e/ou trocar ideias entre si sobre os trabalhos e/ou resolução dos mesmos. No entanto, o trabalho entregue deve corresponder ao esforço individual de cada aluno. São consideradas fraudes as seguintes situações:

- Trabalho parcialmente copiado
- Facilitar a cópia através da partilha de ficheiros
- Utilizar material alheio sem referir a sua fonte.

Em caso de deteção de algum tipo de fraude, os trabalhos em questão não serão avaliados, sendo enviados à Comissão Pedagógica ou ao Conselho Pedagógico, consoante a gravidade da situação, que decidirão a sanção a aplicar aos alunos envolvidos. Serão utilizadas as ferramentas *Moss* e *SafeAssign* para deteção automática de cópias. Recorda-se ainda que o Anexo I do Código de Conduta Académica, publicado a 25 de janeiro de 2016 em Diário da República, 2ª Série, nº 16, indica no seu ponto 2 que:

Quando um trabalho ou outro elemento de avaliação apresentar um nível de coincidência elevado com outros trabalhos (percentagem de coincidência com outras fontes reportada no relatório que o referido software produz), cabe ao docente da UC, orientador ou a qualquer elemento do júri, após a análise qualitativa desse relatório, e em caso de se confirmar a suspeita de plágio, desencadear o respetivo procedimento disciplinar, de acordo com o Regulamento Disciplinar de Discentes do ISCTE - Instituto Universitário de Lisboa, aprovado pela deliberação nº 2246/2010, de 6 de dezembro.

O ponto 2.1 desse mesmo anexo indica ainda que:

No âmbito do Regulamento Disciplinar de Discentes do ISCTE-IUL, são definidas as sanções disciplinares aplicáveis e os seus efeitos, podendo estas variar entre a advertência e a interdição da frequência de atividades escolares no ISCTE-IUL até cinco anos.

Parte I - Shell Script (bash)

Data de entrega: de 24 outubro de 2020

Nesta fase do trabalho será criado um conjunto de scripts para administração e gestão do sistema.

1) **cria_pacientes.sh**

Este script deverá ler o ficheiro **/etc/passwd** do servidor Tigre, transformar os 10 primeiros utilizadores alunos que nele constam em “Pacientes do sistema Cliniq-IUL” e registá-los, criando um novo ficheiro (apagando um já existente) chamado **pacientes.txt**, cuja sintaxe é a seguinte:

<Nº registo (nº identificação);<nome>;<localidade>;<nº telemóvel>;<e-mail>;<saldo>

Como tal, para esses dez primeiros utilizadores alunos, este script deverá:

- Extrair o campo nº de aluno (sem a letra “a”). Este campo será o nº de identificação do paciente;
- Extrair o Nome do aluno que será o nome do paciente;
- O e-mail deverá ser criado automaticamente tendo por base o número de identificação e acrescentando “@iscte-iul.pt”;
- O saldo inicial de todos os pacientes criados é 100.

Exemplo de um registo completo, já com todos os campos preenchidos:

61999;João Silva;Lisboa;912345678;a61999@iscte-iul.pt;100

Os dados dos campos **localidade** e **Nº telemóvel** serão adicionados posteriormente dado que não há ainda essa informação, pelo que os campos devem ficar vazios.

Exemplo de linhas criadas no ficheiro após a importação:

61999;João Silva;;;a61999@iscte-iul.pt;100

61777;Luis Passos;;;a61777@iscte-iul.pt;100

55555;Maria Aguiar;;;a55555@iscte-iul.pt;100

2) cria_medico.sh

Este script é executado quando se pretende inscrever um médico na plataforma. O script é executado com os seguintes argumentos: **nome, nº de cédula profissional, especialidade médica, e-mail**.

Exemplo de invocação do script, **que deverá receber os valores pedidos passados por argumentos**:

```
./cria_medico.sh "Ana Correia" 12345 "dentista" aguiar@gmail.com  
./cria_medico.sh "João Vieira" 98765 "clinica geral" vieira@gmail.com
```

Com os campos inseridos como descrito acima, o script deve verificar se o médico em questão já está registado, e se não, acrescentá-lo ao ficheiro **medicos.txt**, mantendo a seguinte sintaxe:

<nome>;<número cédula profissional>;<especialidade médica>;<e-mail>;<rating>;<nº de consultas efetuadas>;<saldo>

Exemplo do ficheiro medicos.txt:

```
Ana Correia;12345;dentista;aguiar@gmail.com;0;0;0  
João Vieira;54321;clinica geral;vieira@gmail.com;0;0;0
```

Nesta fase, o Rating, o nº de consultas efetuadas e o Saldo são mantidos a “0” (zero).

O script deve mostrar toda a lista de médicos no final da sua execução.

3) stats.sh

Este script é executado pelo administrador do sistema para obter informações sobre o sistema.

Nomeadamente, deve devolver a seguinte informação, consoante os argumentos passados:

- Número de Pacientes inscritos na Plataforma de uma **determinada localidade**;
- Número de Médicos inscritos na Plataforma com **saldo superior a um determinado valor**.

Exemplo de invocação do script, **que deverá receber os valores pedidos passados por argumentos**:

```
./stats.sh Lisboa 500 # Exemplo com Localidade=Lisboa e Saldo mínimo=500
```

4) avalia_medicos.sh

No caso de haver médicos inscritos na plataforma cujo número de consultas efetuadas seja superior a 6 e o seu rating inferior a 5, o administrador pode colocar o respetivo médico na lista negra da plataforma. Para isso, deverá existir um ficheiro “**lista_negra_medicos.txt**” onde será inserida a informação do médico (linha completa) que não cumpra com o requisito pedido.

No final da execução do script deverá ser impressa a “lista negra” de médicos inscritos na plataforma.

5) menu

Este script agrega os restantes, mostrando um menu com as opções:

1. Cria pacientes
2. Cria médicos
3. Stats
4. Avalia médicos
0. Sair

Cada uma das opções anteriores irá invocar o respetivo script descrito nas alíneas anteriores. No caso das opções 2 e 3, este script deverá pedir ao utilizador (administrador) as informações necessárias para execução dos scripts correspondentes, injetando as mesmas como argumentos desses scripts.

Após a execução de cada sub-script, o menu deverá voltar a ser apresentado, e nova opção pedida. Apenas a opção 0 (zero) permite sair deste script “menu”. Até escolher esta opção, o menu deverá ficar em ciclo, permitindo realizar múltiplas operações iterativamente.

Boa sorte!!!