

E3 - Arquitetura

Grupo: 90, Turno: L04, Docente: Tiago Oliveira

| Nome | Nº Aluno | Esforço |
|------------------|-----------------|----------------|
| Mafalda Serafim | 92512 | 8h00 (50%) |
| Rafael Gonçalves | 92544 | 8h00 (50%) |

A arquitetura da aplicação Flask segue uma estrutura hierárquica de páginas html, tendo como “raiz” um index que oferece a possibilidade de escolher entre várias ações. Cada uma destas ações permite por sua vez aceder a uma página própria que pode, oportunamente, oferecer operações mais específicas que redirecionarão o utilizador para páginas que lhe permitem preencher a informação necessária para realizar a ação. Após esta ser realizada com sucesso, é dada ao utilizador a opção de retornar à página index (se for uma operação de edição na base de dados).

Ao inicializar a aplicação (“app.cgi”), é apresentada ao utilizador a homepage (“index.html”). O ficheiro “app.cgi” passa então a estar encarregue de requerer a informação submetida pelo usuário nas páginas html e de chamar recursivamente a função onde se encontra com os novos argumentos. É aqui que se realizam as queries e se pede a renderização da página em questão, de forma a apresentar os resultados das mesmas (quando pertinente para o utilizador). A “app.cgi” contém uma função para cada página html.

O “index.html” disponibiliza ao cliente 5 opções:

1. Inserir, editar e remover instituições e médicos (“instmedic”)
2. Inserir, editar e remover prescrições e análises (“prescanalysis”)
3. Realizar vendas em farmácia (“sales”)
4. Listar substâncias (“substances”)
5. Listar valores de glicémia (“glicemia”)

Ao escolher uma opção, é chamada a função indicada na app, que fará depois o render da página html:

| | |
|----------------------|--------------------|
| instmedic | iedmedico |
| | iedinstituicao |
| prescanalysis | iedprescricao |
| | iedanalise |
| sales | vendasprescricao |
| | vendasemprescricao |
| substances | - |
| glicemia | - |

Tabela 1: Opções fornecidas em cada página

Especificamente, cada página faz o seguinte:

iedmedico:

Recolhe a primary key e a operação a realizar
Recolhe os restantes dados dependendo do tipo de operação
Efetua a operação sobre o médico na base de dados, se possível

iedinstituicao:

Recolhe a primary key e a operação a realizar
Recolhe os restantes dados dependendo do tipo de operação
Efetua a operação sobre a instituição na base de dados, se possível

iedprescricao:

Recolhe a primary key e a operação a realizar
Recolhe os restantes dados dependendo do tipo de operação
Efetua a operação sobre a prescrição na base de dados, se possível

iedanalise:

Recolhe a primary key e a operação a realizar
Recolhe os restantes dados dependendo do tipo de operação
Efetua a operação sobre a análise na base de dados, se possível

Para todas as páginas acima, são verdade as seguintes constatações:

1. Caso a operação pretendida seja “insert”, são recolhidos todos os parâmetros necessários para realizar *insert* na tabela em causa.
2. Caso a operação pretendida seja “edit”, são recolhidos todos os dados que o utilizador pretende alterar, sendo que os restantes deverão permanecer em branco.
3. Caso a operação pretendida seja “delete” verifica-se que o utilizador pretende remover a linha em causa da base de dados, uma vez que já foram requeridas as primary keys.
4. Uma vez que na arquitetura da base de dados, não está precavida a necessidade de dar cascade a uma alteração numa linha, não é possível realizar *edits* ou *deletes* a atributos que sejam foreign key de linhas de outras tabelas.

vendasprescricao:

Recolhe o número do doente e apresenta todas as suas prescrições
Recolhe todos os dados necessários para inserir tanto a venda na base de dados como o registo da prescrição aviada

vendasemprescricao:

Recolhe os dados para a venda

Regista a venda na base de dados, se possível

substances:

Recolhe o número da cédula do médico e o mês do ano

Procura na base de dados todas as linhas compatíveis com a informação introduzida e apresenta as substâncias ao utilizador

glicemia:

Recolhe informação relativa ao concelho (número de região e número de concelho)

Procura na base de dados todas as linhas compatíveis com a informação introduzida e apresenta o par (número de doente, nível de glicémia) ao utilizador.

Nesta última página verifica-se:

1. Se as análises de glicémia forem fora do contexto de uma consulta (o que acontece no nosso populate) o número de doente é apresentado como None.