



Projeto de Bases de Dados

Parte 3



Nome	Número	Percentagem Relativa de Contribuição	Número de Horas Investidas
Pedro Maria	84618	33%	12
Margarida Simões	84611	33%	12
Miguel Gonçalves	84613	33%	12

Grupo 26
Turno: Quinta-Feira 8:30 – 10:00.
Docente: Tomás Almeida e Silva Martins Alves.

Criação da Base de Dados

```
drop table categoria cascade;
drop table categoria_simples cascade;
drop table super_categoria cascade;
drop table constituída cascade;
drop table fornecedor cascade;
drop table produto cascade;
drop table fornece_sec cascade;
drop table corredor cascade;
drop table prateleira cascade;
drop table planograma cascade;
drop table evento_reposicao cascade;
drop table reposicao cascade;
```

```
CREATE TABLE categoria(
    nome varchar(64) NOT NULL unique,
    PRIMARY KEY(nome));
```

```
CREATE TABLE categoria_simples(
    nome char(64) NOT NULL,
    PRIMARY KEY (nome),
    FOREIGN KEY(nome) references categoria (nome) on delete cascade);
```

```
CREATE TABLE super_categoria(
    nome char(64) NOT NULL,
    PRIMARY KEY (nome),
    FOREIGN KEY (nome) references categoria(nome) on delete cascade);
```

```
CREATE TABLE constituída(
    super_categoria char(64) NOT NULL,
    categoria char(64) NOT NULL,
    PRIMARY KEY (super_categoria, categoria),
    FOREIGN KEY (super_categoria) REFERENCES super_categoria(nome) on delete cascade,
    FOREIGN KEY (categoria) REFERENCES categoria(nome)on delete cascade);
```

```
CREATE TABLE fornecedor(
    nif numeric(9) NOT NULL unique,
    nome char(64) NOT NULL,
    PRIMARY KEY(nif));
```

```
CREATE TABLE produto(
    ean numeric(13) NOT NULL unique,
    design char(128) NOT NULL,
    categoria char(64) ,
    forn_primario numeric(9),
    data date NOT NULL,
    PRIMARY KEY(ean),
    FOREIGN KEY(categoria) REFERENCES categoria(nome) on delete set null,
    FOREIGN KEY(forn_primario) references fornecedor(nif));
```

```
CREATE TABLE fornece_sec(
    nif numeric(9) NOT NULL,
    ean numeric(13) NOT NULL,
    PRIMARY KEY (nif, ean),
    FOREIGN KEY (nif) references fornecedor(nif),
    FOREIGN KEY(ean) references produto(ean) on delete cascade);
```

```
CREATE TABLE corredor(
    nro integer NOT NULL unique,
    PRIMARY KEY(nro),
    largura double precision NOT NULL CHECK (largura >= 0));
```

```
CREATE TABLE prateleira(
    nro integer NOT NULL,
    lado char(10) NOT NULL,
    altura varchar(12) NOT NULL ,
    PRIMARY KEY(nro, lado, altura),
    FOREIGN KEY(nro) references corredor(nro));
```

```
CREATE TABLE planograma(
    ean numeric(13) NOT NULL,
    nro integer NOT NULL,
    lado char(10) NOT NULL CHECK (lado != ""),
    altura varchar(12) NOT NULL,
    faces integer NOT NULL CHECK (faces >= 0),
    unidades integer NOT NULL CHECK (unidades >= 0),
    loc integer NOT NULL CHECK (loc >= 0),
    PRIMARY KEY (ean, nro, lado, altura),
    FOREIGN KEY (nro, lado, altura) REFERENCES prateleira(nro, lado, altura),
    FOREIGN KEY (ean) REFERENCES produto(ean)on delete cascade);
```

```
CREATE TABLE evento_reposicao(
    operador varchar(64) NOT NULL,
    instante date NOT NULL,
    PRIMARY KEY(operador, instante));
```

```
CREATE TABLE reposicao(
    ean numeric(13) NOT NULL,
    nro integer NOT NULL,
    lado varchar(10) NOT NULL,
    altura varchar(12) NOT NULL,
    operador char(64) NOT NULL,
    instante date NOT NULL,
    unidades integer CHECK (unidades >= 0),
    PRIMARY KEY (ean , nro, lado, altura, operador, instante),
    FOREIGN KEY (operador, instante) REFERENCES evento_reposicao(operador, instante),
    FOREIGN KEY (ean, nro, lado, altura) REFERENCES planograma(ean, nro, lado, altura)on delete
cascade);
```

Consultas em SQL

a)

```
select nome
from fornecedor f, fornece_sec natural join produto
where( f.nif = fornece_sec.nif or f.nif = produto.forn_primario)
group by f.nif
having count(distinct categoria) >= all
      (select count(distinct categoria)
       from fornecedor f, fornece_sec natural join produto
       where( f.nif = fornece_sec.nif or f.nif = produto.forn_primario)
       group by f.nif)
```

b)

```
select nome, nif
from fornecedor f
where not exists(
(select nome
 from categoria_simples)
except
(select p.categoria
 from produto p, fornecedor s
 where p.forn_primario=s.nif and s.nif=f.nif));
```

c)

```
(select ean from produto)
except
(select ean from reposicao);
```

d)

```
select ean
from fornece_sec
group by ean
having count(nif) > 10;
```

e)

```
select ean
from reposicao
group by ean
having count(distinct operador)=1;
```

Arquitetura PHP

A aplicação Web desenvolvida consiste em 9 ficheiros PHP em que um é uma página inicial, `index.php`, contendo formulários HTML para receber e enviar os dados necessários às páginas PHP de acesso à base de dados que então irão aceder à mesma para adicionar, remover ou listar elementos com base nos dados recebidos.

Existem 7 ficheiros PHP de acesso à base de dados e cada alínea do enunciado é cumprida exclusivamente por 1 ou 2 deles, cada um desempenhando uma única função dentro da alínea ou cumprindo a alínea na totalidade. Os ficheiros `produto_ins_parse.php` e `produto_rem.php`, por cumprirem a alínea b), utilizam transacções para garantir a atomicidade e consistência das operações realizadas. Todos os ficheiros usam também prepared statements para realizar as suas queries de modo a evitar SQL Injection na base de dados e assim prevenir ataques de agentes externos. Cada ficheiro verifica ao início se foram recebidos dados e, no caso de se tratar de um código EAN ou de um NIF, se têm o número certo de algarismos, devolvendo uma mensagem de erro se os dados não forem os esperados.

Existe ainda um 9º ficheiro PHP, `Config.php`, que contém as credenciais para acesso à base de dados e função para tratamento de erros que imprime as respetivas mensagens. Esse ficheiro é incluído por todos os ficheiros PHP e a sua função é sempre chamada caso ocorra uma exceção. É também o único ficheiro PHP da aplicação para o qual a página inicial não envia dados.

A alínea a) é cumprida pelos ficheiros `add_cat_e_sub.php` e `rm_cat_e_sub.php`. O 1º recebe da página inicial a supercategoria e/ou a categoria a adicionar à base de dados. Dois casos principais são considerados: se apenas é recebida uma categoria; ou se são recebidas uma supercategoria e uma categoria. Se apenas for recebida uma categoria, e se esta não existir na base de dados, é inserida na mesma como uma categoria simples não associada a nenhuma supercategoria. Se já existir, uma mensagem de erro é impressa.

Se for recebida uma supercategoria e uma categoria, 4 casos podem acontecer. Ambas não existem, e nesse caso a supercategoria é adicionada na base de dados como supercategoria e a categoria inserida tanto como categoria simples como subcategoria da mesma.

A supercategoria já existe e a categoria não, e nesse caso a supercategoria passa de categoria simples para supercategoria se ainda não o for e a categoria é adicionada à base de dados tanto como categoria simples como subcategoria da mesma.

A supercategoria não existe mas a categoria já, e nesse caso a supercategoria é adicionada à base de dados como supercategoria da categoria existente apenas se essa não tiver já outra supercategoria. Se tiver, uma mensagem de erro é impressa.

Ambas já existem na base de dados, e nesse caso se ambas forem categorias simples então a supercategoria passa a ser supercategoria da categoria existente.

Se apenas for recebida uma supercategoria, uma mensagem de erro é impressa.

O ficheiro `rm_cat_e_sub.php` recebe da página inicial o nome da categoria a remover. Apenas é possível remover categorias simples sem supercategoria ou com supercategoria que para além dessa tenha pelo menos uma outra subcategoria, de modo a que uma supercategoria nunca fique “vazia”. Também só é possível remover supercategorias “root”, ou seja, sem supercategorias acima associadas. Caso a categoria não cumpra estes requisitos, ou não exista na base de dados, uma mensagem de erro específica para cada caso será impressa. Caso uma categoria com produtos associados seja removida, os produtos serão mantidos mas o campo da categoria será mudado para null.

A alínea b) é cumprida pelos ficheiros `produto_ins_parse.php` e `produto_rem.php`. O 1º recebe da página inicial o código EAN, a designação, o nome da categoria, o NIF do fornecedor primário, 1 ou

mais NIFs de fornecedores secundários e a data em que o fornecedor primário se tornou fornecedor primário do produto. É possível a adição de qualquer número de NIFs de fornecedores secundários desde que estejam separados por um ; sem espaços (Um exemplo é 100000001;100000002). Se todos os dados estiverem corretos, uma transacção é iniciada e o produto adicionado à base de dados sem planogramas associados. Caso a categoria ou algum dos fornecedores não exista, o fornecedor primário seja também um dos secundários, ou o código EAN já seja usado por outro produto, uma mensagem de erro é impressa.

O ficheiro produto_rem.php recebe da página inicial o código EAN do produto. Se o produto existir, uma transacção é iniciada e o produto removido da base de dados juntamente com todos os planogramas e reposições associados. Caso o produto não exista uma mensagem de erro é impressa. A categoria que contém o produto removido não é afetada, podendo existir sem produtos associados; também os fornecedores que fornecem o produto não são afetados e podem existir sem produtos associados, embora a sua ligação ao produto removido seja igualmente removida.

A alínea c) é cumprida pelo ficheiro listar_repo.php, que recebe da página inicial o código EAN do produto. Se este existir, é apresentada uma tabela com o operador, o instante e o número de unidades repostas em cada evento de reposição do produto. Se não existir, uma mensagem de erro é impressa.

A alínea d) é cumprida pelo ficheiro alter_design.php, que recebe da página inicial o código EAN do produto e uma designação. Se o produto existir, a sua designação é alterada para a nova designação recebida, podendo até ser a mesma que já existia. Se não existir, uma mensagem de erro é impressa.

A alínea e) é cumprida pelo ficheiro listar_cats.php, que recebe da página inicial o nome da supercategoria. Se a supercategoria existir, é apresentada a lista das suas subcategorias e um link por cada para a mesma página mas listando agora as subcategorias da subcategoria e com novos links, de uma forma recursiva. Se a categoria recebida for uma categoria simples, ou se se abrir o link de uma categoria simples, será simplesmente apresentado um ecrã em branco. Se a supercategoria não existir uma mensagem de erro é impressa.

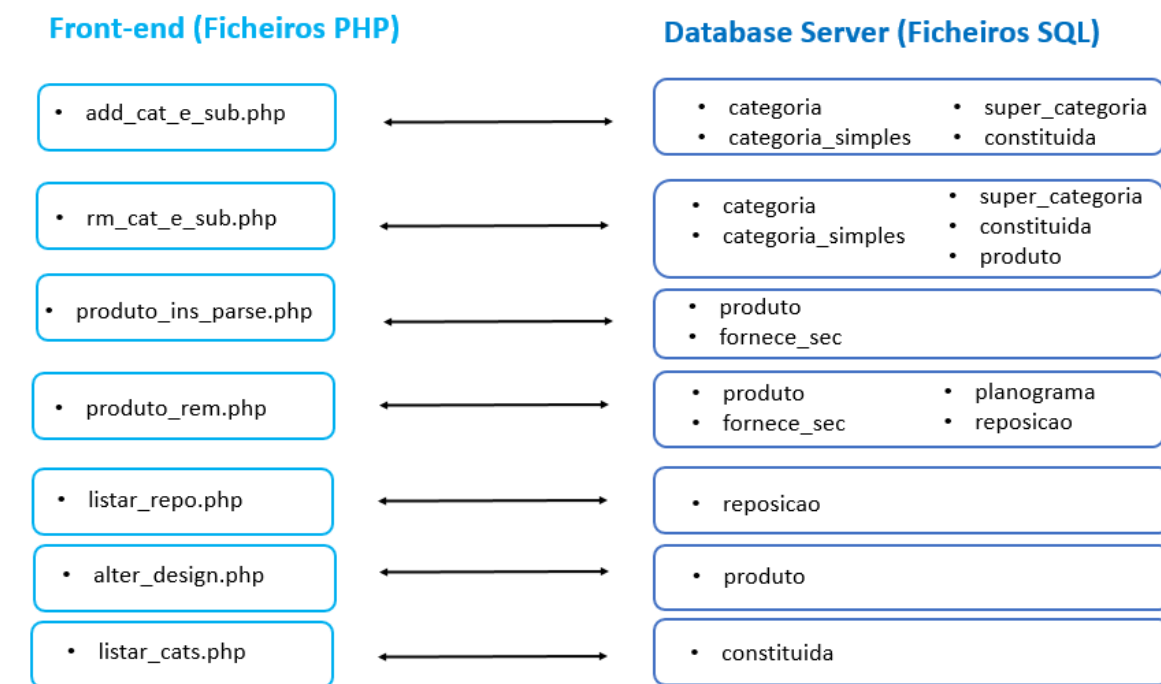


Figura que relaciona os ficheiros PHP com as tabelas da base de dados. Nos ficheiros em que são realizadas inserções e remoções são apresentadas as tabelas em que há alterações nos dados. Nos ficheiros que fazem as listagens são apresentadas as tabelas onde são obtidos os dados para apresentar ao utilizador.