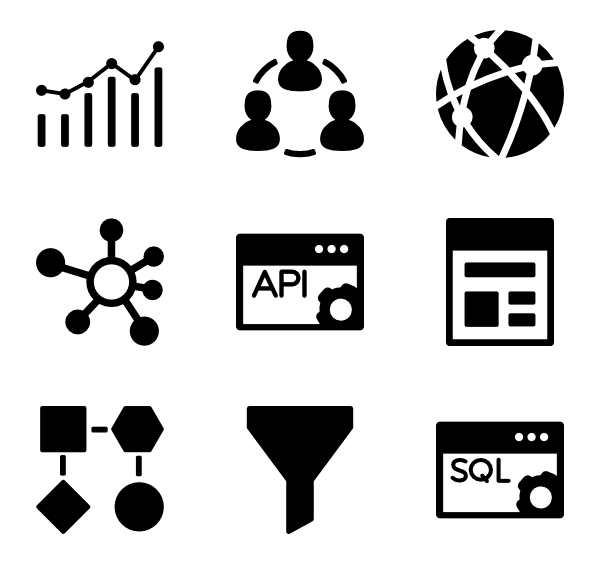
**BASES DE DADOS**

projeto — parte 3  
“Supermarket management”  
ANO letivo 2017/2018

**grupo 12 — TURNO L06 — Quarta-feira, 08h00  
Docente: André Vasconcelos**

|  |  |  |  |
| --- | --- | --- | --- |
| Aluno | Número | Esforço  (em horas) | Esforço (em %) |
| Hélio Domingos | 83473 | 14 | 33,3% |
| Manuel Coimbra | 83505 | 14 | 33,3% |
| Miguel Regouga | 83530 | 14 | 33,3% |



Licenciatura em Engenharia Informática e de Computadores  
Instituto superior técnico

# criação da base de dados

## FiCheiro Schema.sql

DROP TABLE IF EXISTS reposicao;

DROP TABLE IF EXISTS evento\_reposicao;

DROP TABLE IF EXISTS planograma;

DROP TABLE IF EXISTS prateleira;

DROP TABLE IF EXISTS corredor;

DROP TABLE IF EXISTS fornece\_sec;

DROP TABLE IF EXISTS produto;

DROP TABLE IF EXISTS fornecedor;

DROP TABLE IF EXISTS constituida;

DROP TABLE IF EXISTS super\_categoria;

DROP TABLE IF EXISTS categoria\_simples;

DROP TABLE IF EXISTS categoria;

CREATE TABLE categoria( nome varchar(80) NOT NULL, PRIMARY KEY (nome));

CREATE TABLE categoria\_simples( nome varchar(80) NOT NULL, PRIMARY KEY (nome), FOREIGN KEY (nome) REFERENCES categoria(nome) ON DELETE CASCADE ON UPDATE CASCADE);

CREATE TABLE super\_categoria( nome varchar(80) NOT NULL, PRIMARY KEY (nome), FOREIGN KEY (nome) REFERENCES categoria(nome) ON DELETE CASCADE ON UPDATE CASCADE);

CREATE TABLE constituida( super\_categoria varchar(80) NOT NULL, categoria varchar(80) NOT NULL, PRIMARY KEY (super\_categoria, categoria), FOREIGN KEY (super\_categoria) REFERENCES super\_categoria(nome) ON DELETE CASCADE ON UPDATE CASCADE, FOREIGN KEY (categoria) REFERENCES categoria(nome) ON DELETE CASCADE ON UPDATE CASCADE, CHECK (super\_categoria != categoria)

);

CREATE TABLE fornecedor( nif integer NOT NULL unique, nome varchar(80) NOT NULL, PRIMARY KEY (nif)

);

CREATE TABLE produto( ean varchar(80) NOT NULL unique, design varchar(256) NOT NULL, categoria varchar(80) NOT NULL, forn\_primario integer NOT NULL, data date NOT NULL, PRIMARY KEY (ean), FOREIGN KEY (categoria) REFERENCES categoria(nome) ON DELETE CASCADE ON UPDATE CASCADE, FOREIGN KEY (forn\_primario) REFERENCES fornecedor(nif) ON DELETE CASCADE ON UPDATE CASCADE

);

CREATE TABLE fornece\_sec( nif integer NOT NULL, ean varchar(80) NOT NULL, PRIMARY KEY (nif, ean), FOREIGN KEY (nif) REFERENCES fornecedor(nif) ON DELETE CASCADE ON UPDATE CASCADE, FOREIGN KEY (ean) REFERENCES produto(ean) ON DELETE CASCADE ON UPDATE CASCADE

);

CREATE TABLE corredor( nro integer NOT NULL unique, largura numeric(19,4) NOT NULL, PRIMARY KEY (nro)

);

CREATE TABLE prateleira( nro integer NOT NULL, lado varchar(10) NOT NULL, altura numeric(19,4) NOT NULL, PRIMARY KEY (nro, lado, altura), FOREIGN KEY (nro) REFERENCES corredor(nro) ON DELETE CASCADE ON UPDATE CASCADE);

CREATE TABLE planograma( ean varchar (80) NOT NULL unique, nro integer NOT NULL, lado varchar(10) NOT NULL, altura numeric(19,4) NOT NULL, face varchar(10) NOT NULL, unidades integer NOT NULL, loc varchar(30) NOT NULL, PRIMARY KEY (ean, nro, lado, altura), FOREIGN KEY (ean) REFERENCES produto(ean) ON DELETE CASCADE ON UPDATE CASCADE, FOREIGN KEY (nro, lado, altura) REFERENCES prateleira(nro, lado, altura) ON DELETE CASCADE ON UPDATE CASCADE);

CREATE TABLE evento\_reposicao( operador varchar(80) NOT NULL, instante timestamp NOT NULL, PRIMARY KEY (operador, instante), CHECK (localtimestamp >= instante)

);

CREATE TABLE reposicao( ean varchar (80) NOT NULL, nro integer NOT NULL, lado varchar(10) NOT NULL, altura numeric(19,4) NOT NULL, operador varchar(80) NOT NULL, instante timestamp NOT NULL, unidades integer NOT NULL, PRIMARY KEY (ean, nro, lado, altura, operador, instante), FOREIGN KEY (ean, nro, lado, altura) REFERENCES planograma(ean, nro, lado, altura) ON DELETE CASCADE ON UPDATE CASCADE, FOREIGN KEY (operador, instante) REFERENCES evento\_reposicao(operador, instante) ON DELETE CASCADE ON UPDATE CASCADE

);

# Consultas SQL

## ALÍNEA A)

SELECT nome

FROM

(

SELECT fornecedor.nome, COUNT(distinct categoria) c

FROM produto

INNER JOIN fornecedor

ON produto.forn\_primario = fornecedor.nif

GROUP BY nome

UNION ALL

SELECT fornecedor.nome, COUNT(distinct categoria) c

FROM produto

INNER JOIN fornece\_sec

ON produto.ean = fornece\_sec.ean

INNER JOIN fornecedor

ON fornece\_sec.nif = fornecedor.nif

GROUP BY nome

) result

WHERE c = (

SELECT MAX(c)

FROM (

(

SELECT fornecedor.nome, COUNT(distinct categoria) c

FROM produto

INNER JOIN fornecedor

ON produto.forn\_primario = fornecedor.nif

GROUP BY nome

)

UNION

(

SELECT fornecedor.nome, COUNT(distinct categoria) c

FROM produto

INNER JOIN fornece\_sec

ON produto.ean = fornece\_sec.ean

INNER JOIN fornecedor

ON fornece\_sec.nif = fornecedor.nif

GROUP BY nome

)

) result

);

## ALÍNEA B)

SELECT nome, nif

FROM (

SELECT fornecedor.nome, fornecedor.nif, COUNT(distinct categoria) c

FROM produto

INNER JOIN fornecedor

ON produto.forn\_primario = fornecedor.nif

INNER JOIN categoria\_simples

ON categoria\_simples.nome = produto.categoria

GROUP BY fornecedor.nome, nif

) result

WHERE c = (

SELECT COUNT(\*)

FROM categoria\_simples

);

## ALÍNEA C)

SELECT ean

FROM produto

WHERE ean NOT IN (

SELECT ean

FROM reposicao

);

## ALÍNEA D)

SELECT ean

FROM(

SELECT ean

FROM fornece\_sec

GROUP BY ean

HAVING COUNT(nif) > 10) AS result;

## ALÍNEA E)

SELECT ean

FROM reposicao

GROUP BY ean

HAVING COUNT(distinct operador) = 1;

# DESENVOLVIVMENTO DA APLICAÇÃO

A aplicação está sub-dividida em 7 ficheiros diferentes: o ficheiro ***index.html***, que corresponde à página principal da aplicação; o ficheiro ***config.php***, que efetua a ligação à base de dados no SGBD POSTGRES e que é usado por todos os restantes ficheiros, usando o comando *‘include ‘config.php’’*; e os ficheiros ***a.php***, ***b.php***, ***c.php***, ***d.php*** e ***e.php***, que correspondem às componentes da aplicação que permitem efetuar o enunciado.

## fICHEIRO INDEX.HTML

<html> <head><meta http-equiv="Content-Type" content="text/html; charset=utf-8"> </head> <body> <font face="helvetica"> <br><br><br> <div style="text-align: center;"> <div style="line-height: 80%;"> <h3>Instituto Superior Técnico</h3> <h1><b>Projeto de Bases de Dados</b></h1> <h2>Ano Letivo 2017/2018 • Entrega 3</h2> </div> <br> <div style="line-height: 25%"> <b>Grupo 12</b><p>Hélio Domingos, 83473 • Manuel Coimbra, 83505 • Miguel Regouga, 83530 <p>Turno L06 • Quartas-Feiras, 08h00</p> <p>Docente: André Vasconcelos</p> </div> </div> <br><br> <div style="text-align: center;"> <style> .demo {width:1000px; height:100px; border:1px solid #C0C0C0; border-collapse:collapse; padding:50px; margin: 0 auto; text-align: center; } .demo th {border:1px solid #C0C0C0; padding:20px; background:#F0F0F0; font-size: 100px; } .demo td {border:1px solid #C0C0C0; padding:20px; width: 300px; } </style> <table class="demo"> <thead> <tr> <th>(<a href='a.php'>A</a>)</th> <th>(<a href='b.php'>B</a>)</th> <th>(<a href='c.php'>C</a>)</th> <th>(<a href='d.php'>D</a>)</th> <th>(<a href='e.php'>E</a>)</th> </tr> </thead> <tbody> <tr> <td>Inserir e remover categorias e sub-categorias</td> <td>Inserir e remover um novo produto e os seus respectivos fornecedores (primários ou secundários), garantindo que esta operação seja atómica</td> <td>Listar eventos de reposição de um dado produto, incluindo o número de unidades repostas</td> <td>Alterar a designação de um produto</td> <td>Listar todas as sub-categorias de uma super-categoria, a todos os níveis de profundidade</td> </tr> </tbody> </table></div> </font> </body> </html>

## fICHEIRO CONFIG.PHP

<?php $host = "db.ist.utl.pt"; $user ="ist424833"; $password = "miguelregouga"; $dbname = $user; $db = new PDO("pgsql:host=$host;dbname=$dbname", $user, $password); $db->setAttribute(PDO::ATTR\_ERRMODE, PDO::ERRMODE\_EXCEPTION); ?>

## fICHEIRO A.PHP

O ficheiro *a.php* corresponde à alínea a) do desenvolvimento da aplicação e permite a adicionar e remover categorias e sub-categorias. Para tal, são mostradas ao utilizador as tabelas que poderão ajudar o utilizador a perceber o estado da base de dados, antes e após a efetuação das alterações. Para que o utilizador tenha menos probabilidade de se enganar a escrever, criamos um menu *dropdown* quando necessário; este menu é criado com base no estado atual da base de dados, sendo que, para tal, é feita uma *query* que obtém as categorias, categorias simples ou super-categorias atualmente presentes. Após um dos 4 formulários estar preenchido, é feita a ligação à base de dados, efetuada a *query* correspondente à funcionalidade que o utilizador desejou e é atualizada a página; em caso de sucesso, são mostradas as tabelas com as alterações efetuadas. Em caso de insucesso, é mostrado um erro.

<html> <head><meta http-equiv="Content-Type" content="text/html; charset=utf-8"> </head> <body> <font face="helvetica"> <button style="width: 200px; height: 40px;" onclick="window.location='index.html';">< Voltar</button><br><br> <h2>Inserir e remover categorias e sub-categorias</h2> <br><br> <?php try {include 'config.php'; $mode = isset($\_REQUEST['mode']) ? $\_REQUEST['mode'] : ''; $type = isset($\_REQUEST['type']) ? $\_REQUEST['type'] : ''; $nome = $\_POST['nome']; $nome\_sub = $\_POST['nome\_sub']; $nome\_super = $\_POST['nome\_super']; $sql = "START TRANSACTION;"; $db->query($sql); if ($mode == "remove") {if ($type == "categoria") {$sql = "DELETE FROM categoria WHERE nome = '$nome';"; $db->query($sql); } if ($type == "sub-categoria") {$sql = "DELETE FROM constituida WHERE super\_categoria = '$nome\_super' AND categoria = '$nome\_sub';"; $db->query($sql); $sql = "DELETE FROM categoria WHERE nome = '$nome\_sub';"; $db->query($sql); } } if ($mode == "add") {if ($type == "categoria") {$sql = "INSERT INTO categoria VALUES('$nome');"; $db->query($sql); if ($\_REQUEST['cat']=="catsimp") {$sql = "INSERT INTO categoria\_simples VALUES('$nome');"; $db->query($sql); } if ($\_REQUEST['cat']=="supercat") {$sql = "INSERT INTO super\_categoria VALUES('$nome');"; $db->query($sql); } } if ($type == "sub-categoria") {$sql = "INSERT INTO categoria VALUES('$nome\_sub');"; $db->query($sql); $sql = "INSERT INTO categoria\_simples VALUES('$nome\_sub');"; $db->query($sql); $sql = "INSERT INTO constituida VALUES('$nome\_super', '$nome\_sub');"; $db->query($sql); } } $sql = "END;"; $db->query($sql); echo("<div style=\"margin: 0 auto; padding-right: 30px; display: inline-block;\"><h3>Constituida</h3>\n"); $sql = "SELECT \* FROM constituida;"; $result = $db->query($sql); echo("<table style=\"margin: 0 auto;\" border=\"1\">\n"); echo("<tr><td><b>super\_categoria</b></td><td><b>categoria</b></td></tr>\n"); foreach($result as $row) {echo("<tr><td>"); echo($row['super\_categoria']); echo("</td><td>"); echo($row['categoria']); echo("</td></tr>\n"); } echo("</table></div>\n"); echo("<div style=\"margin: 0 auto; padding-right: 30px; display: inline-block;\"><h3>Categoria</h3>\n"); $sql = "SELECT \* FROM categoria;"; $result = $db->query($sql); echo("<table style=\"margin: 0 auto;\" border=\"1\">\n"); echo("<tr><td><b>nome</b></td></tr>\n"); foreach($result as $row) {echo("<tr><td>"); echo($row['nome']); echo("</td></tr>\n"); } echo("</table></div>\n"); echo("<div style=\"margin: 0 auto; padding-right: 30px; display: inline-block;\"><h3>Categoria Simples</h3>\n"); $sql = "SELECT \* FROM categoria\_simples;"; $result = $db->query($sql); echo("<table style=\"margin: 0 auto;\" border=\"1\">\n"); echo("<tr><td><b>nome</b></td></tr>\n"); foreach($result as $row) {echo("<tr><td>"); echo($row['nome']); echo("</td></tr>\n"); } echo("</table></div>\n"); echo("<div style=\"margin: 0 auto; padding-right: 30px; display: inline-block;\"><h3>Super Categoria</h3>\n"); $sql = "SELECT \* FROM super\_categoria;"; $result = $db->query($sql); echo("<table border=\"1\">\n"); echo("<tr><td><b>nome</b></td></tr>\n"); foreach($result as $row) {echo("<tr><td>"); echo($row['nome']); echo("</td></tr>\n"); } echo("</table></div>\n"); $db = null; $prep = null; } catch (PDOException $e) {echo("<p>Ocorreu um erro na ligação à base de dados. Verifique que introduziu corretamente os dados e que tem ligação à internet.\n\n Detalhes: {$e->getMessage()}</p><br><a href=\"a.php\"> < Voltar</a>"); } ?> <br> <br><br> <h3>Adicionar nova categoria</h3> <form action='a.php' method='post'> <p><input type='hidden' name='mode' value='add'/></p> <p><input type='hidden' name='type' value='categoria'/></p> <p>Nome da nova categoria: <input type='text' name='nome' required/></p> <p><input type="radio" name="cat" value="catsimp"/> Categoria Simples<input type="radio" name="cat" value="supercat" /> Super Categoria<br /></p> <p><input type='submit' value='Submeter'/></p> </form> <br> <h3>Adicionar nova sub-categoria</h3> <form action='a.php' method='post'> <p><input type='hidden' name='mode' value='add'/></p> <p><input type='hidden' name='type' value='sub-categoria'/></p> <p>Nome da nova sub-categoria: <input type='text' name='nome\_sub' required/></p> <p>Nome da super-categoria: <select name='nome\_super' required> <?php include 'config.php'; $sql = "SELECT nome FROM super\_categoria;"; $result = $db->query($sql); foreach($result as $row) {echo '<option value="'.htmlspecialchars($row['nome']).'">' . $row['nome'] . '</option>'; } ?> </select></p> <p><input type='submit' value='Submeter'/></p> </form> <br> <h3>Remover categoria</h3> <form action='a.php' method='post'> <p><input type='hidden' name='mode' value='remove'/></p> <p><input type='hidden' name='type' value='categoria'/></p> <p>Nome da categoria a remover: <select name='nome' required> <?php include 'config.php'; $sql = "SELECT nome FROM categoria;"; $result = $db->query($sql); foreach($result as $row) {echo '<option value="'.htmlspecialchars($row['nome']).'">' . $row['nome'] . '</option>'; } ?> </select></p> <p><input type='submit' value='Submeter'/></p> </form> <br> <h3>Remover sub-categoria</h3> <form action='a.php' method='post'> <p><input type='hidden' name='mode' value='remove'/></p> <p><input type='hidden' name='type' value='sub-categoria'/></p> <p>Nome da sub-categoria a remover: <select name='nome\_sub' required> <?php include 'config.php'; $sql = "SELECT nome FROM categoria\_simples;"; $result = $db->query($sql); foreach($result as $row) {echo '<option value="'.htmlspecialchars($row['nome']).'">' . $row['nome'] . '</option>'; } ?> </select></p> <p>Nome da super-categoria da sub-categoria a remover: <select name='nome\_super' required> <?php include 'config.php'; $sql = "SELECT nome FROM super\_categoria;"; $result = $db->query($sql); foreach($result as $row) {echo '<option value="'.htmlspecialchars($row['nome']).'">' . $row['nome'] . '</option>'; } ?> </select></p> <p><input type='submit' value='Submeter'/></p> </form> <br><br> </body> </html>

## fICHEIRO B.PHP

O ficheiro *b.php* corresponde à alínea b) do desenvolvimento da aplicação e permite inserir e remover um novo produto e os seus respetivos fornecedores (primários ou secundários), garantindo que a operação seja atómica. Como na alínea anterior, são mostradas ao utilizador as tabelas que poderão ajudar o utilizador a perceber o estado da base de dados, antes e após a efetuação das alterações – neste caso, é mostrada apenas a tabela *produto*. Para que o utilizador tenha menos probabilidade de se enganar a escrever, criamos um menu *dropdown* quando necessário – neste caso, apenas faz sentido no campo ‘categoria do produto’. Para garantir a atomicidade da operação, são feitas *queries* ‘START TRANSACTION’ e ‘END’, antes e após das *queries* principais da componente da aplicação serem efetuadas. Para o número de fornecedores secundários (e tendo em conta que um produto pode ter zero ou mais fornecedores secundários), é usada uma função *Javascript* que permite a criação das caixas Nome e NIF correspondentes ao número de fornecedores secundários que o utilizador desejar. Depois de introduzir o número de fornecedores secundários no campo correspondente, é necessário clicar em “Submeter número de fornecedores secundários”, completar o formulário e só depois de completo clicar em “Submit”. Após um dos formulários estar preenchido, é feita a ligação à base de dados, efetuada a *query* correspondente à funcionalidade que o utilizador desejou e é atualizada a página; em caso de sucesso, são mostradas as tabelas com as alterações efetuadas. Em caso de insucesso, é mostrado um erro.

NOTA: quando é removido um produto, os fornecedores desse mesmo produto não são removidos da base de dados; caso o mesmo tenha fornecedores secundários, a ligação entre o produto e o fornecedor secundário é sim removida da tabela *fornece\_sec*, mas esses fornecedores secundários não são removidos da tabela *fornecedores*.

<html> <head><meta http-equiv="Content-Type" content="text/html; charset=utf-8"> <script type="text/javascript"> function addFields(){var number = document.getElementById("member").value; var container = document.getElementById("container"); while (container.hasChildNodes()) {container.removeChild(container.lastChild); } for (i=0;i<number;i++){container.appendChild(document.createTextNode("NIF do Fornecedor Secundário " + (i+1))); var input = document.createElement("input"); input.type = "text"; input.name = "forece\_sec\_nif\_" + i; input.id = "forece\_sec\_nif\_" + i; container.appendChild(input); container.appendChild(document.createTextNode("Nome do Fornecedor Secundário " + (i+1))); var input = document.createElement("input"); input.type = "text"; input.name = "forece\_sec\_nome\_" + i; input.id = "forece\_sec\_nome\_" + i; container.appendChild(input); container.appendChild(document.createElement("br")); } } </script> </head> <body> <font face="helvetica"> <button style="width: 200px; height: 40px;" onclick="window.location='index.html';">< Voltar</button><br><br> <?php try {include 'config.php'; $ean = $\_POST['ean']; $design = $\_POST['design']; $categoria = $\_POST['categoria']; $data = $\_POST['data']; $forn\_primario\_nif = $\_POST['forn\_primario\_nif']; $forn\_primario\_nome = $\_POST['forn\_primario\_nome']; $forn\_secundario\_nif = $\_POST['forn\_secundario\_nif']; $forn\_secundario\_nome = $\_POST['forn\_secundario\_nome']; $mode = isset($\_REQUEST['mode']) ? $\_REQUEST['mode'] : ''; $sql = "START TRANSACTION;"; $db->query($sql); if ($mode == "remove") {$sql = "DELETE FROM produto WHERE ean = '$ean';"; $db->query($sql); $sql = "DELETE FROM fornece\_sec WHERE ean = '$ean;'"; $db->query($sql); } if ($mode == "add") {$numberSecs = $\_REQUEST['member']; $fornecedoresSecs\_NIF = array(); $fornecedoresSecs\_NOME = array(); for ($i = 0; $i < $numberSecs; $i++) {$natual = $i; $text1 = 'forece\_sec\_nif\_'; $conc1 = $text1 . $natual; $atualForneceSecNIF = $\_REQUEST[$conc1]; $fornecedoresSecs\_NIF[$i] = $atualForneceSecNIF; $text2 = 'forece\_sec\_nome\_'; $conc2 = $text2 . $natual; $atualForneceSecNOME = $\_REQUEST[$conc2]; $fornecedoresSecs\_NOME[$i] = $atualForneceSecNOME; } $sql = "INSERT INTO fornecedor (nif, nome) SELECT $forn\_primario\_nif, '$forn\_primario\_nome' WHERE not exists (SELECT nif FROM fornecedor WHERE nif=$forn\_primario\_nif);"; $db->query($sql); $sql = "INSERT INTO produto VALUES($ean, '$design', '$categoria', $forn\_primario\_nif, '$data');"; $db->query($sql); for ($i = 0; $i < $numberSecs; $i++) {$sql = "INSERT INTO fornecedor (nif, nome) SELECT $fornecedoresSecs\_NIF[$i], '$fornecedoresSecs\_NOME[$i]' WHERE not exists (SELECT nif FROM fornecedor WHERE nif=$fornecedoresSecs\_NIF[$i]);"; $db->query($sql); $sql = "INSERT INTO fornece\_sec values ($fornecedoresSecs\_NIF[$i], $ean);"; $db->query($sql); } } $sql = "END;"; $db->query($sql); echo("<h3>Produto</h3>\n"); $sql = "SELECT \* FROM produto;"; $result = $db->query($sql); echo("<table border=\"1\">\n"); echo("<tr><td><b>ean</b></td><td>design</td><td>categoria</td><td>forn\_primario</td><td>data</td></tr>\n"); foreach($result as $row) {echo("<tr><td>"); echo($row['ean']); echo("</td><td>"); echo($row['design']); echo("</td><td>"); echo($row['categoria']); echo("</td><td>"); echo($row['forn\_primario']); echo("</td><td>"); echo($row['data']); echo("</td></tr>\n"); } echo("</table>\n"); if (isset($\_POST['submeter'])) {$ean = $\_POST['ean']; echo("<h3>Fornecedores</h3>\n"); $sql = "SELECT \* FROM fornecedor;"; $result = $db->query($sql); echo("<table border=\"1\">\n"); echo("<tr><td><b>nif</b></td><td>nome</td></tr>\n"); foreach($result as $row) {echo("<tr><td>"); echo($row['nif']); echo("</td><td>"); echo($row['nome']); echo("</td></tr>\n"); } echo("</table>\n"); echo("<h3>Fornecedores Secundários</h3>\n"); $sql = "SELECT \* FROM fornece\_sec;"; $result = $db->query($sql); echo("<table border=\"1\">\n"); echo("<tr><td><b>nif</b></td><td>ean</td></tr>\n"); foreach($result as $row) {echo("<tr><td>"); echo($row['nif']); echo("</td><td>"); echo($row['ean']); echo("</td></tr>\n"); } echo("</table>\n\n"); $db = null; $prep = null; } } catch (PDOException $e) {echo("<p>Ocorreu um erro.<br>Verifique que introduziu os dados na forma correta.<br>Assegure que a categoria com o nome introduzido não existe atualmente na base de dados.<br><br> Detalhes: {$e->getMessage()}</p><br><a href=\"b.php\"> < Voltar</a>"); } ?> <h3>Adicionar produto e seus respetivos fornecedores</h3> <form action='b.php' method='post'> <p><input type='hidden' name='mode' value='add'/></p> <p>EAN do Produto: <input type='text' name='ean' required/></p> <p>Designação do Produto: <input type='text' name='design' required/></p> <p>Categoria do Produto: <select name='categoria' required> <?php include 'config.php'; $sql = "SELECT nome FROM categoria;"; $result = $db->query($sql); foreach($result as $row) {echo '<option value="'.htmlspecialchars($row['nome']).'">' . $row['nome'] . '</option>'; } ?> </select></p> <p>NIF do Fornecedor Primário do Produto: <input type='text' name='forn\_primario\_nif' required/></p> <p>Nome do Fornecedor Primário do Produto: <input type='text' name='forn\_primario\_nome' required/></p> <p>Data de início do fornecimento (formato AAAA-MM-DD): <input type='text' name='data' required/></p> <p>Número de Fornecedor secundários:<input type="text" id="member" name="member" required value=""><br /> <a href="#" id="filldetails" onclick="addFields()">Submeter número de fornecedores</a> <div id="container"/></div></p> <p><input type='submit' name='submeter' value='Submeter'/></p> </form> <br> <br> <h3>Remover produto e seus respetivos fornecedores</h3> <form action='b.php' method='post'> <p><input type='hidden' name='mode' value='remove'/></p> <p>EAN do Produto: <select name='ean' required> <?php include 'config.php'; $sql = "SELECT ean FROM produto;"; $result = $db->query($sql); foreach($result as $row) {echo '<option value="'.htmlspecialchars($row['ean']).'">' . $row['ean'] . '</option>'; } ?> </select></p> <p><input type='submit' name='submeter' value='Submeter'/></p> </form> </body> </html>

## fICHEIRO C.PHP

O ficheiro *c.php* corresponde à alínea c) do desenvolvimento da aplicação e permite listar eventos de reposição de um dado produto, incluindo o número de unidades repostas. Para tal, é mostrada ao utilizador a tabela ‘produto’. Para que o utilizador tenha menos probabilidade de se enganar a escrever, criamos um menu *dropdown* que contém o EAN dos produtos atualmente presentes na base de dados. Após a seleção do produto que o utilizador deseja consultar os seus eventos de reposição, é efetuada a *query* de consulta e é atualizada a página; em caso de sucesso, são mostradas as tabelas com as alterações efetuadas. Em caso de insucesso, é mostrado um erro.

<html> <head><meta http-equiv="Content-Type" content="text/html; charset=utf-8"> </head> <body> <font face="helvetica"> <button style="width: 200px; height: 40px;" onclick="window.location='index.html';">< Voltar</button><br><br> <?php try {include 'config.php'; echo("<h3>Produto</h3>\n"); $sql = "SELECT ean, design FROM produto;"; $result = $db->query($sql); echo("<table border=\"1\">\n"); echo("<tr><td><b>ean</b></td><td>design</td></tr>\n"); foreach($result as $row) {echo("<tr><td>"); echo($row['ean']); echo("</td><td>"); echo($row['design']); echo("</td></tr>\n"); } echo("</table>\n"); if (isset($\_POST['submeter'])) {$ean = $\_POST['ean']; echo("<h3>Eventos de Reposição do produto $ean</h3>\n"); $sql = "START TRANSACTION;"; $db->query($sql); $sql = "SELECT ean, operador, instante, unidades FROM evento\_reposicao NATURAL JOIN reposicao WHERE ean = '$ean';"; $result = $db->query($sql); $sql = "END;"; $db->query($sql); echo("<table border=\"1\">\n"); echo("<tr><td><b>ean</b></td><td>operador</td><td>instante</td><td>unidades</td></tr>\n"); foreach($result as $row) {echo("<tr><td>"); echo($row['ean']); echo("</td><td>"); echo($row['operador']); echo("</td><td>"); echo($row['instante']); echo("</td><td>"); echo($row['unidades']); echo("</td></tr>\n"); } echo("</table>\n"); $db = null; $prep = null; } } catch (PDOException $e) {echo("<p>Ocorreu um erro.<br>Tente novamente mais tarde.<br><br> Detalhes: {$e->getMessage()}</p><br><a href=\"c.php\"> < Voltar</a>"); } ?> <h3>Listar eventos de reposição de um produto</h3> <form action='c.php' method='post'> <p>EAN do Produto: <select name='ean' required> <?php include 'config.php'; $sql = "SELECT ean FROM produto;"; $result = $db->query($sql); foreach($result as $row) {echo '<option value="'.htmlspecialchars($row['ean']).'">' . $row['ean'] . '</option>'; } ?> </select></p> <p><input type='submit' name='submeter' value='Submeter'/></p> </form> </body> </html>

## fICHEIRO d.PHP

O ficheiro *d.php* corresponde à alínea d) do desenvolvimento da aplicação e permite alterar a designação de um dado produto. Para tal, é mostrada ao utilizador a tabela ‘produto’. Para que o utilizador tenha menos probabilidade de se enganar a escrever, criamos um menu *dropdown* que contém o EAN dos produtos atualmente presentes na base de dados. Após a seleção do produto que o utilizador deseja alterar e da nova designação que lhe pretende dar é efetuada a *query* e é atualizada a página; em caso de sucesso, é mostrada a tabela ‘produto’ com as alterações efetuadas. Em caso de insucesso, é mostrado um erro.

<html> <head><meta http-equiv="Content-Type" content="text/html; charset=utf-8"> </head> <body> <font face="helvetica"> <button style="width: 200px; height: 40px;" onclick="window.location='index.html';">< Voltar</button><br><br> <?php try {include 'config.php'; $design = $\_POST['design']; $ean = $\_POST['ean']; $sql = "START TRANSACTION;"; $db->query($sql); if (isset($\_POST['submeter'])) {$sql = "UPDATE produto SET design = '$design' WHERE ean = '$ean';"; $db->query($sql); } $sql = "END;"; $db->query($sql); echo("<h3>Produto</h3>\n"); $sql = "SELECT \* FROM produto;"; $result = $db->query($sql); echo("<table border=\"1\">\n"); echo("<tr><td><b>ean</b></td><td>design</td><td>categoria</td><td>forn\_primario</td><td>data</td></tr>\n"); foreach($result as $row) {echo("<tr><td>"); echo($row['ean']); echo("</td><td>"); echo($row['design']); echo("</td><td>"); echo($row['categoria']); echo("</td><td>"); echo($row['forn\_primario']); echo("</td><td>"); echo($row['data']); echo("</td></tr>\n"); } echo("</table>\n"); $db = null; $prep = null; } catch (PDOException $e) {echo("<p>Ocorreu um erro.<br>Tente novamente mais tarde.<br><br> Detalhes: {$e->getMessage()}</p><br><a href=\"d.php\"> < Voltar</a>"); } ?> <h3>Alterar designação de um produto</h3> <form action='d.php' method='post'> <p>EAN do produto: <select name='ean' required> <?php include 'config.php'; $sql = "SELECT ean FROM produto;"; $result = $db->query($sql); foreach($result as $row) {echo '<option value="'.htmlspecialchars($row['ean']).'">' . $row['ean'] . '</option>'; } ?> </select></p> <p>Nova designação: <input type='text' name='design' required></p> <p><input type='submit' name='submeter' value='Submeter'/></p> </form> </body> </html>

## fICHEIRO e.PHP

O ficheiro *e.php* corresponde à alínea e) do desenvolvimento da aplicação e permite listar todas as sub-categorias de uma super-categoria, a todos os níveis de profundidade. Para tal, é mostrada ao utilizador a tabela ‘super\_categoria’. Para que o utilizador tenha menos probabilidade de se enganar a escrever, é criado um menu *dropdown* que contém os nomes das super-categorias atualmente presentes na base de dados. Após a seleção da super-categoria é efetuada a *query* — que, neste caso, é uma *query* recursiva, e permite percorrer toda a ‘árvore’ de sub-categorias da super-categoria dada — e é atualizada a página; em caso de sucesso, são apresentadas todas as sub-categorias da super-categoria escolhida. Em caso de insucesso, é mostrado um erro.

<html> <head><meta http-equiv="Content-Type" content="text/html; charset=utf-8"> </head> <body> <font face="helvetica"> <button style="width: 200px; height: 40px;" onclick="window.location='index.html';">< Voltar</button><br><br> <?php try {include 'config.php'; echo("<h3>Super-Categorias</h3>\n"); $sql = "SELECT DISTINCT nome FROM super\_categoria;"; $result = $db->query($sql); echo("<table border=\"1\">\n"); echo("<tr><td><b>nome</b></td></tr>\n"); foreach($result as $row) {echo("<tr><td>"); echo($row['nome']); echo("</td></tr>\n"); } echo("</table>\n"); if (isset($\_POST['submeter'])) {$nome = $\_POST['nome']; echo("<h3>Sub-Categorias da Super-Categoria $nome</h3>\n"); $sql = "START TRANSACTION;"; $db->query($sql); $sql = "WITH RECURSIVE result (super\_categoria, categoria) AS (SELECT super\_categoria, categoria FROM constituida where super\_categoria='$nome' UNION ALL SELECT constituida.super\_categoria, constituida.categoria FROM constituida INNER JOIN result on constituida.super\_categoria = result.categoria) SELECT categoria FROM result;"; $result = $db->query($sql); $sql = "END;"; $db->query($sql); echo("<table border=\"1\">\n"); echo("<tr><td><b>categoria</b></td></tr>\n"); foreach($result as $row) {echo("<tr><td>"); echo($row['categoria']); echo("</td></tr>\n"); } echo("</table>\n"); $db = null; $prep = null; } } catch (PDOException $e) {echo("<p>Ocorreu um erro.<br>Tente novamente mais tarde.<br><br> Detalhes: {$e->getMessage()}</p><br><a href=\"e.php\"> < Voltar</a>"); } ?> <br><br><h3>Listar todas as sub-categorias de uma super-categoria, a todos os níveis de profundidade</h3> <form action='e.php' method='post'> <p>Nome da Super-Categoria: <select name='nome' required> <?php include 'config.php'; $sql = "SELECT nome FROM super\_categoria;"; $result = $db->query($sql); foreach($result as $row) {echo '<option value="'.htmlspecialchars($row['nome']).'">' . $row['nome'] . '</option>'; } ?> </select></p> <p><input type='submit' name='submeter' value='Submeter'/></p> </form> </body> </html>