

Faculdade de Engenharia da Universidade do Porto

Mestrado Integrado em Engenharia Electrotécnica e de Computadores

Ano lectivo 2007 / 2008

SISTEMAS DE INFORMAÇÃO EMPRESARIAIS

GUIÃO DO TRABALHO PRÁTICO

INTRODUÇÃO À PROGRAMAÇÃO WEB

José António Faria (jfaria@fe.up.pt)

FEUP, Setembro de 2007

1. Introdução

Este trabalho prático tem por objectivo principal a familiarização com um conjunto de tecnologias que suportam o desenvolvimento de aplicações web. Entre essas tecnologias estão:

- a base de dados Postgres
- a linguagem HTML
- a linguagem de script PHP
- a linguagem de script Javascript

As aulas práticas vão desenrolar-se em duas fases. Numa primeira fase, familiarizar-se-á com o ambiente de desenvolvimento e as tecnologias anteriores através de um conjunto de exercícios. Depois, desenvolverá uma pequena aplicação web.

Os pontos seguintes, descrevem em detalhe as tarefas a realizar.

No caso de encontrar problemas ao longo da realização do trabalho, deve contactar:

- o administrador do sistema: Engº José Gaspar (jgaspar@fe.up.pt, extensão: 1521)
- um dos docentes: José A. Faria (jfaria@fe.up.pt) ou Armando J. Sousa (asousa@fe.up.pt).

1.1. Contas

O trabalho será desenvolvido em grupos de um ou dois alunos.

Para cada grupo de trabalho serão criadas contas nos servidores Linux e no sistema de gestão de bases de dados Postgres do DEEC cujos usernames serão do tipo *siexx*, onde *xx* corresponde ao número do grupo.

Inicialmente, as passwords Linux e Postgres serão idênticas aos usernames, mas deverá alterá-las o mais rapidamente possível:

- para alterar a password Linux ao nível do NIS, utilize o comando *yppasswd* (e não o comando *passwd*).
- para alterar a password Postgres, deve enviar uma mensagem de email ao docente indicando a nova password pretendida.

Estas contas serão destruídas no final da disciplina! Por isso, não se esqueça de copiar para a sua conta pessoal toda a informação que pretenda conservar.

2. Linguagem HTML

Este exercício tem por objectivo a familiarização e o domínio da linguagem de formatação HTML através do desenvolvimento de uma página web.

Pode optar por desenvolver a sua página pessoal (no caso de ainda não o ter feito), ou por desenvolver a página de uma disciplina, por exemplo da disciplina Qualidade.

Em qualquer dos casos, a página deve proporcionar uma **apresentação gráfica cuidada** (alinhamento, cores, fontes e disposição dos vários elementos da página) e, além dos elementos básicos do HTML (formatação, links, imagens, ...) deve incluir tabelas, navegação entre páginas, folhas de estilo.

Para definir o layout das páginas deve recorrer a tabelas ou a divisões (tags html *table* e *div*)

Neste exercício, dado que o objectivo é o de adquirir um domínio razoável do HTML, **deve editar directamente o código HTML sem recorrer a ferramentas de edição dedicadas** (mais tarde poderá fazê-lo mas, para já não).

Antes de iniciar o desenvolvimento da página:

- crie no seu directório Home o subdirectório public-html e
- dê permissões de leitura e execução a todos os utilizadores nesse directório, através da execução do comando Linux `chmod 755 public_html`

Apoio:

Se ainda não está familiarizado com o HTML, pode começar por consultar um, ou mais, dos tutoriais mencionados na página da disciplina, bem como os exemplos introdutórios também disponíveis nessa página.

3. Base de dados Postgres

3.1. Recorrendo ao utilitário psql ou à ferramenta phpPgAdmin para administração via web da base de dados Postgres (informação disponível em <http://phpPgAdmin.sourceforge.net/>), crie uma pequena base de dados de teste no Postgres com as seguintes três tabelas - *Peças*, *Máquinas*, *Clientes* e *Encomendas* - cada uma das quais contendo os seguintes campos:

- tabela **Peças**: Código, Descrição, Cor, Peso, Cliente
- tabela **Máquinas**: Código, Descrição, Capacidade, Operador
- tabela **Encomendas**: Número, Código_peça, Código_máquina, Quantidade, Data_entrega

De seguida, introduza registos nestas tabelas de forma a que existam:

- Encomendas de vários tipos de peças a executar numa mesma máquina;
- Encomendas da mesma peça a executar em diferentes máquinas.

Apoio:

Se ainda não está familiarizado com a base de dados Postgres, pode começar por consultar o pequeno tutorial de introdução ao Postgres e ao psql fornecido no anexo 1 deste guião.

Para informações detalhadas, em <http://bonfim.fe.up.pt/doc/postgresql-doc/html/>, encontra a documentação oficial do Postgres, nomeadamente o Tutorial e o Users' Guide.

3.2. Depois de criadas e preenchidas as tabelas da sua base de dados, execute, recorrendo ao psql, as seguintes interrogações SQL:

- Códigos das peças cujo cliente é xx
- Números das Encomendas executadas na máquina cujo código é xx e cujas peças se destinam ao cliente yy
- Nº total de Encomendas planeadas para a máquina cuja descrição é xx
- Nº total de peças cujo código é xx a fabricar na máquina cujo código é yy
- Quantidade média das Encomendas executadas na máquina cujo nome é xx e cujo cliente é yy
- ...

3.3. Recorrendo aos mecanismos oferecidos pelo próprio sistema de gestão de base de dados, garanta que a sua base de dados respeita as seguintes restrições de integridade:

- os códigos de peça e de máquina são únicos e obrigatórios
- as Encomendas são numeradas sequencial e automaticamente
- o peso das peças e a quantidade das Encomendas são sempre superiores a zero
- a data de conclusão das Encomendas é sempre posterior à data de criação da ordem
- a eliminação do registo associado a uma máquina implica a eliminação de todas as Encomendas planeadas para essa máquina

- a alteração do código de uma máquina implica a alteração desse código nas Encomendas planeadas para essa máquina
- não podem ser eliminadas registos de peças para as quais existam Encomendas planeadas
- não podem ser alterados os nomes de peças para as quais existam Encomendas planeadas

Apoio:

Para executar este ponto, consulte atentamente o comando CREATE TABLE no manual do Postgres

4. Linguagem PHP

4.1. Comece por verificar que o módulo PHP está devidamente instalado no servidor web através da execução do script *teste-php-ok.php*, disponível na página da disciplina.

4.2. De seguida, verifique se a biblioteca de ligação à base de dados também está devidamente instalada executando o script *teste-php-pg-ok.php*, também disponível na página da disciplina.

Tenha em atenção que poderá ter de alterar o nome da base de dados, o username, a password e a tabela acedida na query de acordo com a base de dados que criou no ponto anterior.

4.3. Depois de efectuadas estas verificações, crie uma página estática com o título - *Lista de Peças* - e uma tabela que reproduza o conteúdo da tabela *Peças* que criou no ponto anterior.

4.4. De seguida, edite um script PHP que gere uma página equivalente, mas onde a informação contida na tabela seja obtida dinamicamente por consulta da base de dados. Para criar esse script, utilize o seguinte "template".

```
<html><head>
<title>Template para acesso a base de dados em PHP</title>
</head>

<body>
<?php    /* Início do script PHP */

/* Estabelecimento da ligação à base de dados */
/* Não se esqueça de alterar os parâmetros da função pg_connect */
$conn = pg_connect("hostname=host dbname=nome_bdd user=username
                    password=pass ");

if (!$conn)
{
    print "Nao foi possivel estabelecer a ligacao";
    exit;
}

else
    echo "Ligação estabelecida";

/* Escrita do identificador da conexão para efeitos de debug */
echo "<br>conn = ".$conn."<br>";

/* Definição da query SQL */
/* Não se esqueça de alterar o nome da tabela a consultar na query */
$query = "select * from table;";

$result = pg_exec($conn, $query); /* Execução da query */

/*Escrita do nº de registos do resultado para debug*/
$rows = pg_numrows($result);
echo "rows = ".$rows."<br>";
if ($rows == 0)
    print "0 tuples";

/* Percorrer os registos do resultado */
$i = 0;
while ($i < $rows)
{
    $row = pg_fetch_row($result, $i); /* Obter a linha nº i do resultado */
    echo $row[0]."<br>"; /* escrever o primeiro campo da linha nº i */
    echo $row[1]."<br>"; /* escrever o segundo campo da linha nº i */
    $i++;
}

pg_close($conn); /* Terminar a ligação à base de dados */

?> /* Terminar o script PHP */
</body></html>
```

Apoio:

- Em www.php.net encontra a documentação oficial do PHP, nomeadamente o **Users' Guide**.

- O manual da linguagem está disponível em <http://bonfim.fe.up.pt/doc/phpdoc/html/>.
- Na página da disciplina, pode encontrar alguns exemplos introdutórios.
- Se preferir começar por um texto em português, pode consultar o livro “Programação com PHP”, de Carlos Serrão e Joaquim Marques, Editora FC@.net.

4.5. Crie um nova página web com um formulário contendo:

- um campo de texto, através do qual possa especificar o valor de um campo de uma das tabelas que criou anteriormente;
- um botão para submeter esse valor a um script PHP que o utilizará como parâmetro de uma consulta à base de dados, apresentando o resultado dessa consultas sob a forma de tabelas.

Por exemplo, o formulário permitirá ao utilizador especificar o nome de um cliente, com base no qual o sistema fornecerá três tabelas contendo:

- as peças desse cliente
- as Encomendas planeadas para peças desse cliente
- as máquinas em que serão executadas as Encomendas das peças desse cliente

4.6. Crie uma página na qual o utilizador possa especificar dois parâmetros, um através de "radio buttons" e o outro através de um menu "pull down", a utilizar em consultas à base de dados. Como anteriormente, os resultados da consulta devem ser devolvidos sob a forma de tabela.

Por exemplo, através do "radio button" e do menu "pull down", o utilizador especificará um cliente e uma máquina, sendo-lhe devolvida a lista das Encomendas executadas nessa máquina para esse cliente.

Se o utilizador não especificar nenhuma máquina, serão listadas todas as Encomendas para esse cliente e, se não for especificado nenhum cliente, serão listadas todas as Encomendas para a máquina especificada.

A lista de "radio button" e o menu "pull down" são gerados dinamicamente em função dos registos na base de dados (assim, por exemplo, se uma nova máquina for registada na base de dados, o menu "pull down" deve passar a incluir automaticamente essa nova máquina).

4.7. Crie uma nova página através da qual seja possível alterar, eliminar e introduzir novos registos na tabela Peças.

Garanta as seguintes restrições de integridade através de código PHP (portanto sem recorrer aos mecanismos do próprio Postgres como fez no ponto 3.3:

- os códigos das peças são únicos e obrigatórios
- o peso das peças é sempre superior a zero
- não podem ser eliminadas registos de peças para as quais existam Encomendas planeadas
- a alteração do código das peça é propagada para a tabela de Encomendas

OS EXERCÍCIOS DOS PONTOS 2, 3 E 4 DEVEM SER CONCLUÍDOS ATÉ AO DIA 7 DE OUTUBRO

5. Linguagem JavaScript

Recorrendo a Javascript, altere a página que desenvolveu no ponto 4.7 por forma a que:

- as opções *alterar* / *eliminar* / *introduzir* registos na tabela peças sejam acedida através de botões cujo aspecto gráfico se altera quando o cursor se encontra sobre um desses botões;
- os dados introduzidos pelo utilizador são validados no lado do cliente antes de serem enviados ao servidor e, no caso de serem detectados erros, são afixadas mensagens de alerta ao utilizador apropriadas;
- depois de seleccionada uma dada opção, por exemplo *introduzir novo registo*, o formulário para a introdução de dados aparece numa nova janela.

Apoio:

- Se ainda não está familiarizado com a linguagem JavaScript poderá começar por consultar um dos muitos tutoriais de introdução a esta linguagem disponíveis na web, entre os quais os mencionados na página da disciplina.
- Para uma informação completa poderá consultar *Client-Side JavaScript Guide* e *Client-Side JavaScript Reference* em <http://developer.netscape.com/docs/manuals/>
- Na página da disciplina, também poderá encontrar um conjunto de scripts em JavaScript que exemplificam o tipo de funcionalidades que se pretende que implemente neste exercício.
- Se preferir começar por um texto em português, poderá consultar o livro “JavaScript - Animação e programação em páginas web”, de Pedro Coelho, Editora FC@.net, 2ª edição de 2002.

6. Desenvolvimento de pequena aplicação web

Depois de, nos pontos anteriores, ter adquirido um conhecimento mínimo sobre um conjunto de tecnologias, neste ponto, vai desenvolver uma pequena aplicação web onde integrará essas tecnologias.

Crie uma base de dados com as seguintes cinco tabelas

- tabela **Peças**: Código, Descrição, Cor, Peso, Cliente
- tabela **Máquinas**: Código, Descrição, Capacidade, Operador
- tabela **Clientes**: Código, Nome, Morada, Telefone
- tabela **Encomendas**: Número, Código_peça, Código_cliente, Código_máquina, Quantidade, Data_entrega
- tabela **Stocks**: Código_peça, Qt_armazém, Qt_disponível*

** quando é registada uma nova encomenda, a quantidade correspondente é abatida a Qt_disponível*

Desenvolva uma interface web através da qual o utilizador possa:

- introduzir novos registos nas tabelas da base de dados;
- eliminar registos existentes;
- efectuar diversos tipos de consultas.

Ao desenvolver essa interface, preocupe-se com a facilidade de utilização por parte do utilizador recorrendo aos elementos introduzidos ao longo dos pontos anteriores, tais como frames, menús dinâmicos, mecanismos de integridade, validação dos dados introduzidos pelo utilizador,

Preocupe-se, também, com a qualidade gráfica da interface podendo, se quiser, utilizar ferramentas de edição web de alto nível.

Relativamente às funcionalidades da interface, sugere-se que seja oferecido um menu através do qual o utilizador selecciona o tipo de operação a efectuar (Introduzir, Eliminar, Consultar). Nos dois primeiros casos, o utilizador deve especificar de seguida a tabela onde pretende introduzir ou eliminar registos.

Quando se trata da introdução de um novo registo, o utilizador deve especificar os valores dos campos respectivos.

No caso da eliminação de registos, devem ser listados os registos existentes, após o que o utilizador seleccionará o registo, ou registos, a eliminar.

No caso de ser seleccionada a opção Consulta, o utilizador deverá poder efectuar consultas sob diferentes critérios à sua escolha.

Claro que também não se deve esquecer de definir as restrições de integridade apropriadas para as tabelas da base de dados.

Especificações Adicionais

As especificações seguintes são opcionais. Como tal, a sua implementação não é obrigatória mas, naturalmente, valoriza o trabalho que irá apresentar para avaliação.

1. Considere que existem dois tipos de utilizador: *normal* que apenas pode consultar os dados e *gestor* que pode alterar / eliminar / introduzir dados.
 - Cada utilizador, é identificado por *username* / *password* e tem de introduzir a sua identificação apenas uma vez no início de uma nova sessão de trabalho, mantendo-se essa identificação válida ao longo de toda a sessão.
2. Defina uma folha de estilos para a sua aplicação contemplando, pelo menos, os seguintes elementos: divisões, títulos e tabelas.
3. Trate os erros no acesso à base de dados por forma a que mensagens de erro geradas pelo SGBD não sejam visíveis pelo utilizador.
4. Defina uma vista que reúna os dados seguintes tabela: Número da ordem, código da peça, descrição da peça, código da máquina, descrição da máquina, código do cliente, nome do cliente, quantidade, data de conclusão
5. Defina uma regra que registe automaticamente numa tabela auxiliar todas as alterações (Update ou Delete) efectuadas sobre os registos da tabela Encomendas.
6. Utilize transações para garantir a atomicidade das operações que envolvam manipulação dos dados de mais do que uma tabela (por exemplo, quando é registada uma nova encomenda na tabela respectiva, a quantidade disponível na tabela é simultaneamente actualizada).

ESTE EXERCÍCIO DEVE SER CONCLUÍDO ATÉ AO DIA 19 DE OUTUBRO

---- BOM TRABALHO ---

FEUP, Setembro de 2007

José António Faria

ANEXO I: Introdução à base de dados Postgres

O Postgres é um sistema de gestão de bases de dados relacionais disponível para Linux e desenvolvido na Universidade de Berkley.

Além do modelo relacional clássico, o Postgres suporta um conjunto adicional de conceitos como classes, herança, tipos, funções, restrições, regras e integridade das transacções (Nota: em princípio, não precisará de utilizar estes conceitos no âmbito do trabalho a realizar em SIE).

1. Acesso à base de dados

As bases de dados Postgres podem ser acedidas:

- utilizando o utilitário *psql* que permite aceder interactivamente à base de dados através de uma interface caracter;
- utilizando uma linguagem que disponha de uma interface para o Postgres, como acontece com as linguagens Perl, Tcl ou PHP;
- através de SQL embutido num programa em C (biblioteca LIBPQ).

No âmbito deste trabalho vai aceder à base de dados através:

- do utilitário *psql* para criar as tabelas, introduzir valores e efectuar testes;
- da linguagem *PHP*, para efectuar consultas no contexto da aplicação que irá desenvolver.

2. Invocação do utilitário psql

Este utilitário permite-lhe aceder interactivamente à base de dados Postgres. Para o iniciar proceda da seguinte forma:

1. efectuar telnet para a máquina onde está instalado o servidor de bases de dados (*bonfim*)
2. invocar o *psql*: *psql base_de_dados*, onde *base_de_dados* é o nome da base de dados a que pretende aceder.
3. a partir daqui, a prompt passa a *base_de_dados=>* e pode começar a efectuar consultas SQL sobre essa base de dados.

3. Criação das bases de dados

Por defeito, ao ser criado um novo utilizador do Postgres, é dada autorização de acesso a esse utilizador à base de dados *Postgres*, a partir da qual serão criadas as bases de dados dos utilizadores.

Da primeira vez que executar o invocar o comando *psql*, terá obrigatoriamente de se ligar a esta base de dados. **Não deve criar nenhuma nova tabela nesta base de dados**, mas apenas aproveitar a ligação a *Postgres* para criar uma nova base de dados para o seu grupo, onde criará as tabelas que utilizará ao longo dos exercícios propostos de seguida.

Para criar essa base de dados, por exemplo com um nome idêntico ao seu username, proceda da seguinte forma:

1. efectue telnet para o servidor da base de dados: *telnet bonfim*

2. invoque o psql: *psql postgres*
3. crie a nova base de dados: *postgres=> CREATE DATABASE SIExx;*
(não se esqueça do ";" no final do comando)
4. se o comando tiver sido correctamente executado, o sistema responderá com: *CREATDB*
5. termine a ligação à base de dados postgres fazendo: a *postgres=> \q*
6. ligue-se à base de dados que acabou de criar: *psql SIExx*

4. Queries

No psql, as queries SQL terminam com o caracter ";", ou com o meta-comando \g. O caracter "fim de linha" não termina a query.

Tipicamente, começará por criar uma ou mais tabelas, depois introduzirá valores nessas tabelas, após o poderá efectuar as primeiras consultas.

Para uma primeira familiarização com a base de dados sugere-se que siga o tutorial PostgreSQL ao longo do qual vai, sucessivamente:

a. criar uma tabela:

```
CREATE TABLE weather (cidade VARCHAR(15), temperatura INTEGER);
```

b. introduzir registos nessa tabela:

```
INSERT INTO weather VALUES ('Porto', 25);
```

c. efectuar consultas à tabela:

```
SELECT * FROM weather WHERE cidade = 'Porto';
```

```
SELECT cidade FROM weather
```

```
SELECT cidade FROM weather INTO cidades
```

Nota: esta última query contém um comando CREATE implícito.

d. actualizar o valor dos registos:

```
UPDATE weather SET temperatura = 15 WHERE cidade = 'Porto'
```

e. eliminar registos:

```
DELETE FROM weather WHERE cidade = 'Porto';
```

f. eliminar tabelas:

```
DROP TABLE weather
```

5. Metacomandos

O *psql* aceita um conjunto de metacomandos, os quais são iniciados pelo caracter "\". Estes comandos são executados pelo próprio *psql*, não pela base de dados. Para uma descrição completa consulte o manual. Alguns metacomandos que, provavelmente, serão úteis no âmbito deste trabalho são os seguintes (para os quais deve consultar a documentação):

- \d: mostra todas as relações.
- \d *relação*: mostra todas as colunas da *relação*.
- \help ou \h [comando]: ajuda relativa ao comando
- \i *ficheiro*: lê o input de *ficheiro*, como se tivesse sido introduzido a partir do teclado
- \l ou \list: lista todas as bases de dados existentes no servidor e o respectivo proprietário
- \?: ajuda sobre os meta-comandos

6. Outras funcionalidades SQL

Para uma apresentação das funcionalidades da linguagem SQL disponíveis no Postgres, nomeadamente funções agregadas, como por exemplo, count, sum avg, max, min, consulte o tutorial Postgres disponível em </usr/doc/postgresql-doc/indexpage.html>.