

Projeto BD - Parte 3

Grupo 07 – Turno BD2L02

Professor:

João Tomás Brazão Caldeira

Alunos	Número de Aluno	Percentagem Relativa de Contribuição	Esforço Total
André Lazenga	99052	33%	40h
André Correia	102666	34%	40h
Tiago Firmino	103590	33%	40h

EXPLICAÇÃO DA ARQUITETURA E RELAÇÕES ENTRE OS FICHEIROS DA APLICAÇÃO WEB

Link para a versão de trabalho:

<https://web2.tecnico.ulisboa.pt/ist1102666/clients.cgi>

A aplicação web encontra-se disponível pelas credenciais do aluno 102666.

A nossa aplicação web representa um sistema de gestão de uma base de dados, é constituída por doze ficheiros, entre os quais: login.py, base.py, products.cgi, clients.cgi, orders.cgi, addCustomer.cgi, addOrder.cgi, addSupplier.cgi, addProduct_Supplier.cgi, delete.cgi, edit.cgi e pay.cgi.

O ficheiro login.py trata de realizar o início de sessão com na base de dados do Técnico do aluno. O ficheiro base.py consiste de funções auxiliares para a criação das páginas web, com a formatação base escolhida.

O utilizador interage maioritariamente com os ficheiros products.cgi, clients.cgi e orders.cgi, visto que estes dão display às três páginas principais, podendo navegar entre estas clicando num dos tabs respetivos, localizados no topo da página. Todas as tabelas apresentadas estão paginadas e estão limitadas a vinte entradas por página, facilitando a leitura das mesmas.

Relativamente ao clients.cgi, este apresenta a tabela customer presente na base de dados, mostrando todos os dados. Numa última coluna, a tabela dispõe de um botão com o formato de um “X” vermelho, cuja funcionalidade é remover o cliente e tudo associado ao mesmo. Ao premir esse botão, é enviada a informação do cliente para o ficheiro delete.cgi, onde é tratada a remoção. Na parte inferior da página, após a tabela, encontra-se o botão “Add a Customer”, ao premi-lo é redirecionado para addCustomer.cgi, onde é tratada a adição de um cliente.

Quanto ao orders.cgi, é apresentada a tabela order presente na base de dados, mostrando todos os dados e, numa última coluna, o estado de pagamento da encomenda. Este estado tem duas variantes, “Already Paid” caso a encomenda se encontre paga, caso contrário é apresentado ‘pay’ a verde, que ao ser premido envia os dados para pay.cgi, tratando da simulação de pagamento da encomenda. Na parte inferior da página, tal como na página Customers, encontra-se o botão “Place an Order”, que ao ser premido redireciona para addOrder.cgi, onde é tratada a adição de uma encomenda.

No que diz respeito ao products.cgi, é exibida a tabela product da base de dados, mostrando todos os dados, exceto o ean. Adicionalmente, existem duas colunas, onde a última dispõe de um botão com o formato de um “X” vermelho, com funcionalidade idêntica ao descrito no paragrafo relativo a clients.cgi, aplicada aos produtos. A outra coluna apresenta um botão “See Suppliers” verde, em que ao clicar, é mostrada a tabela com os fornecedores do produto em causa,

mostrando todos os seus dados. Ainda na tabela dos produtos, nas colunas “Description” e “Price”, o texto é apresentado a azul, significando que é possível premir de modo a alterar o seu valor, quando esta ação é realizada o utilizador é redirecionado para edit.cgi, com a informação do produto, para alterar o valor. Na parte inferior da página, tal como nas outras, encontra-se o botão “Add a Product”, cujo ao ser premido é redirecionado para addProduct_Supplier.cgi, onde é realizado a adição do produto e do seu fornecedor. Por fim, na tabela de fornecedores de cada produto, existe, novamente, o botão no formato de um “X” vermelho, para remover um dado fornecedor, caso este seja o único fornecedor, também é removido o produto e tudo associado ao mesmo. Também, há um botão “Add a Supplier”, idêntico aos demais, que ao ser premido, redireciona para addSupplier.cgi, onde é tratada a adição de um fornecedor do produto em causa.

Em todas ações existe um botão “Cancel”, que permite retroceder para a página anterior. Quando o utilizador submete os dados de input, estes são coletados pelos respetivos ficheiros que lançam pedidos para o servidor da base de dados, recebendo a resposta deste, validando a execução da ação.

RESPOSTA AOS ÍNDICES (SECÇÃO 7)

Dadas as consultas presentes no enunciado, e após o estudo do funcionamento de índices sobre as mesmas, chegámos à conclusão que faria sentido criar os índices da seguinte forma:

1. CREATE INDEX product_price_idx ON product (price);
2. CREATE INDEX product_name_idx ON product (name);
3. CREATE INDEX order_date_idx ON “order” (date);

Fez-nos sentido a criação de um índice do tipo B+Tree sobre o atributo price da tabela product (1.), visto que este índice otimizaria a filtragem “price > 50” no caso de existir um maior número de produtos com o valor de price menor que 50, desta maneira o tempo de execução do primeiro caso melhoraria.

Tal como referido anteriormente, optámos pelo uso de um índice do tipo B+Tree para o atributo name da tabela product (2.). Este índice tornaria mais rápida a filtragem “name LIKE ‘A%’” se a quantidade de produtos com o nome começado por “A” fosse inferior à quantidade de produtos começados por qualquer outra letra, acelerando assim a consulta do segundo caso.

Por fim, também escolhemos um índice do tipo B+Tree para o atributo date da tabela “order” (3.). Este índice otimizaria a filtragem “YEAR = 2023” desde que o número de encomendas realizadas em 2023 fosse baixo relativamente ao número total, assim o tempo de consulta do primeiro caso diminuía.

Já que devemos supor que o tamanho das tabelas excede a memória em várias ordens de magnitude, o uso da B+Tree é o que consideramos mais correto, visto

que este tipo de índices são otimizados para se autoadaptar com inserções e deleções de modo a assegurar que todos os leaf nodes se encontram à mesma profundidade, e as entradas são ordenadas pelo valor da search key, mantidas numa estrutura de procura hierárquica. Assim, as inúmeras entradas consultadas pelos dois casos apresentados teriam um menor custo em memória com a utilização destes índices.

Esta hipótese foi testada através do uso de um script que preencheu a base de dados com imensas entradas aleatórias (cumprindo as restrições), e foi observado que o tempo e custo dos dois casos foi menor com o uso destes índices. Chegámos então à conclusão que ambas as consultas se tornaram mais eficientes, confirmando a hipótese teórica.