Introdução

Como o tema era de livre arbítrio então o tema escolhido foi um “Mercado de Compra e Vendas Online”, uma vez que é uma forma de comprar cada vez mais usada. Para a construção da aplicação focámo-nos numa aplicação que pudesse servir de modelo, neste caso a Amazon, uma vez que é a empresa com maior sucesso da atualidade.

A implementação de um sistema de base de dados trouxe sobretudo vantagens como resposta rápida aos pedidos de informação, múltiplo acesso, flexibilidade e integridade da informação. Neste projeto em particular, se um utilizador efetuar uma operação, esta será rápida, visto que a base de dados contém toda a informação necessária para a realizar. Já que o custo elevado da construção de uma base de dados é a sua maior desvantagem, especialmente se esta for mal concetualizada, este projeto torna-se viável, até porque é relativamente simples e foi investido algum tempo na sua concetualização.

Levantamento e análise de requisitos

O processo de levantamento e análise de requisitos pode ser dividido nos seguintes passos:

• **Entrevista:** Conversa com cliente sobre os requisitos que ele considera mais importante;

• **Recolha de requisitos**: Recolha de requisitos com potenciais clientes do site;

• **Reunir**: Agrupar os vários requisitos dos diferentes utilizadores numa única lista de requisitos

• **Verificação de requisitos**: Garantir que não haja inconsistências, ou conflitos, entre diferentes requisitos;

• **Validação de requisitos**: Garantir que todas as principais funcionalidades foram cumpridas e que o sistema seja implementável;

Requisitos Levantados

Ao fim do processo de levantamento de requisitos estar concluído foi decidido os requisitos para cada um dos tipos de utilizadores tinha acesso.

**Requisitos do utilizador:**

• Ver produtos que estão disponíveis no mercado;

• Filtrar os produtos que pretende ver/comprar do mercado;

• Aceder a informações pessoais;

• Comprar produtos que estão disponíveis no mercado;

• Adicionar/Alterar informações pessoais;

**Requisitos do administrador**

• Aceder a informações de todo o sistema;

• Ver estatísticas do mercado e dos utilizadores;

• Adicionar/Remover métodos de pagamento ou transporte;

Modelo Concetual

**Abordagem de modelação realizada**

O primeiro passo na construção do modelo concetual foi definir as principais entidades, para isso foram analisados os requisitos do sistema e identificar todos os nomes. No passo seguinte foram identificadas as entidades e associados os diferentes atributos a cada entidade, e o seu tipo, por exemplo se eram identificadores da entidade, atributos multivalorados, atributo derivado ou um atributo composto. Foi necessário encontrar as principais relações entre as várias entidades e o tipo de relações e a multiplicidade das relações. Por fim, foi verificado se havia redundância no modelo e se este estava de acordo com os requisitos.

**Descrição do modelo concetual**

Como seria de esperar, uma das entidades é o utilizador. Este é o responsável pelas operações de compra ou venda no site. Este dispõe de diferentes métodos de pagamentos (que são por si só uma entidade) como transferência bancária, cartão de crédito, entre outros. Como foi mencionado acima, o utilizador pode vender ou comprar produtos no site, sendo que os produtos, como entidades, devem ser listados e caraterizados pelo seu vendedor. Várias compras individuais de produtos formam um carrinho de compras, ao qual os produtos serão adicionados após as compras, que serão oficializadas dada a confirmação do carrinho final. Os produtos comprados serão então transportados até ao seu comprador pelo método mais conveniente.

Esquema Lógico

**Nota:** mencionar as tabelas que existem.

**Desenho do modelo lógico:** Todos os atributos correspondem a uma coluna nas respetivas tabelas. Para respeitar a primeira forma normal, foram criadas as tabelas *Telemovel*, *Email* e *Métodos de Pagamento – Utilizador*, assim evitamos as listas de números de telemóvel que o Utilizador poderá ter. Todos os atributos normais são dependentes da chave primária, se existente. E, para além disso, são todos independentes entre si.

Implementação Física

O sistema de gestão de base de dados (SGBD) utilizado foi o proposto pelos docentes da UC, o MySQL.

**Tradução do esquema lógico para o sistema de gestão de bases de dados escolhido em SQL**

Os esquemas têm “traduções” entre si, as entidades no modelo físico são referidas como tabelas e os atributos de cada entidade são colunas. A maior diferença entre os esquemas é que o modelo físico particulariza o tipo de dados, por exemplo, específica que a chave primária dos utilizadores, o NIF, é um inteiro.

**Tradução das interrogações do utilizador para SQL**

Foram criadas algumas interrogações, como visualizar os produtos disponíveis. Foram filtrados os produtos com quantidade maior que 0.

Também podem ser vistos os 5 clientes que mais receberam com o sistema. Esta interrogação é um pouco mais complexa. Para verificar quais são os 5 utilizadores que mais faturaram com o sistema agrupa-se os produtos aos utilizadores. Somam-se os preços, ordena-se por ordem decrescente e retira-se os 5 que mais receberam.

**Definição e caracterização dos mecanismos de segurança em SQL**

Há várias formas de aumentar a segurança como:

Limitar a informação que cada utilizador possa aceder;

*Backups*;

Migração para NoSQL

**Justificação da Migração**

Sistemas NoSQL oferecem maior flexibilidade e escalabilidade, isto é, o crescimento da quantidade de dados no sistema não afeta tanto a quantidade de recursos necessários para a manutenção, o que é uma mais-valia para quem trabalha com recursos limitados, como no nosso caso.

**Estrutura base do sistema NoSQL**

Neste caso, como foi decidido migrar para uma Base de Dados MongoDB, foram criadas coleções para todas as entidades que tinham sido definidas no modelo concetual. Na MongoDataBase foram criadas no total as seguintes 6 coleções, uma para cada entidade.

• Utilizador;

• Produto;

• Compra;

• Carrinho;

• MetodosPagamento;

• Transporte;

Como numa base de dados baseada em documentos é possível um campo ser uma lista, então os atributos multivalorados podiam ser inseridos num documento, ao contrário da base de dados relacional. Também foi decidido, que tal como na base de dados relacional, alguns documentos iriam conter um campo com o identificador de um documento de outra coleção.

**Processo de migração de dados**

A implementação do processo de migração de dados neste trabalho prático foi, relativamente, simples. A implementação esta dividida em 3 packages: Classes, ClassesDAO e ClassesMongo. Estas classes vão tratar da transformação, extração e carregamento, respetivamente.

A fase da extração resume-se a 3 métodos: "getConnection", "createStatement" e o "executeQuery". O getConnection cria uma ligação ao MySQL. O createStatement recebe o statement que irá ser executado pelo MySql e o executeQuery executa esse statement.

O processo da transformação passa por converter os objetos de dados extraídos do MySql para uma coleção da respetiva classe.

Quanto à fase de carregamento, é criado uma ligação ao Mongo através do método MongoClient. Criada a ligação, é feita a criação de uma MongoDatabase e é criada uma coleção para cada classe que criámos na fase de transformação. Depois disso bastou percorrer os "Sets" dos objetos onde estavam armazenadas as informações e criar um documento para cada objeto, com o auxílio do método: "document". Por fim, também existe a classe Migracao que contém o método main, que vai interligar todas as fases do trabalho.

Conclusão

Atualmente, várias coisas que se fazem no dia-a-dia necessitam de bases de dados, como o uso de cartões multibanco, ou inscrição em websites. Num site que pretende atrair várias pessoas, é fundamental a escolha apropriada das caraterísticas de uma base de dados. Qualquer operação nesta loja necessita de aceder à base de dados, desde a compra e venda de produtos até à consulta de operações anteriores, demonstrando assim a sua importância.