

Faculdade de Engenharia da Universidade do Porto



Relatório do projeto de Base de dados

Supermercado “Baratinho”

Estudantes & Autores:

José Leandro, up202008061

Manuel Alves, up201906910

Tiago Barbosa, up202004926

Índice

Contexto	3
Diagrama	4
Esquema Relacional	5
Análise de Dependências Funcionais e Formas Normais	6
Restrições à base de dados.....	8
Interrogação de Base de Dados	10
Gatilhos da Base de Dados	11

Contexto

No âmbito da unidade curricular de Base de dados, foi nos pedido para criar uma base de dados acerca de um tema. O tema escolhido foi um supermercado que intitulamos “Baratinho”.

Pretende-se armazenar a informação em relação a todas as pessoas que frequentam o supermercado guardando o seu nome, NIF, idade, sexo, morada e número de telefone. Se a pessoa for cliente é também guardado o seu número de cliente. Se pertencer ao Staff armazena-se o seu número de funcionário, data de início de contrato, data de fim de contrato e o seu dia de folga semanal. Cada funcionário tem uma especialidade com um salário base e carga horário inerentes.

É necessário ter em conta que consideramos que as pessoas que são funcionários também podem ser clientes e as pessoas, dentro do contexto desta base de dados, só podem ser funcionários ou clientes.

Para a base de dados vai ser também armazenada informação sobre os vários produtos, cada produto tem um nome, preço, quantidade em stock, marca associada e pertence a um determinado tipo de produto.

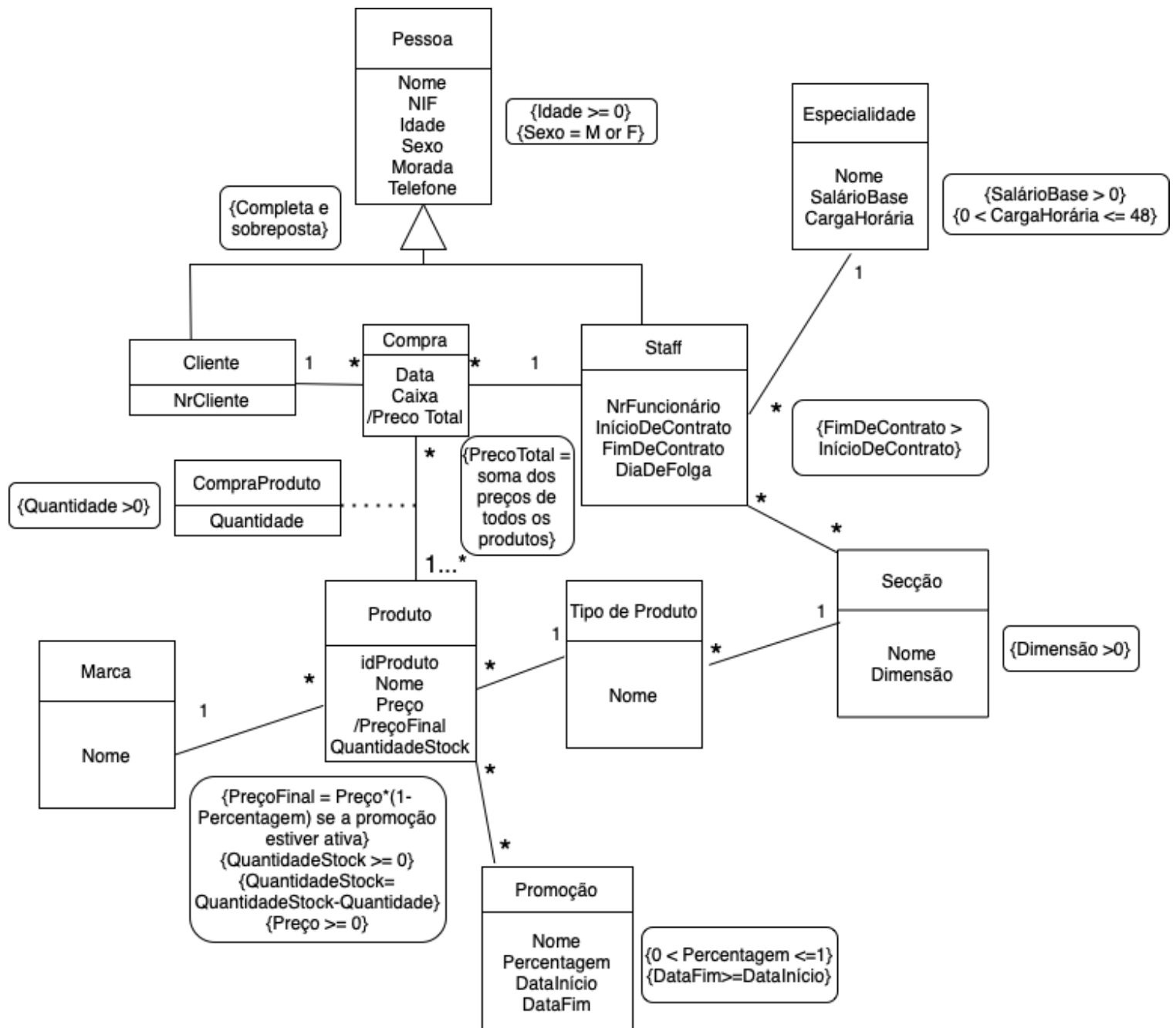
O supermercado é constituído por várias secções às quais estão associados vários tipos de produto e vários funcionários, mas cada tipo de produto só está associado a uma secção. Acerca das secções é guardado o seu nome e dimensão em metros quadrados.

Neste supermercado existem também promoções associadas aos vários produtos e pretende-se guardar o seu nome, percentagem de desconto, data de início e fim da campanha.

Acerca de todas as compras realizadas no supermercado é necessário conhecer o cliente e o funcionário envolvidos, os produtos adquiridos, a

quantidade de cada produto, a data, o número da caixa e o preço total da compra.

Diagrama



Esquema Relacional

- Pessoa(NIF, Nome, Idade, Sexo, Morada, Telefone)
- Cliente(NIF->Pessoa, NrCliente)
- Staff(NIF->Pessoa, NrFuncionário, InícioDeContrato, FimDeContrato, DiaDeFolga, idEspecialidade->Especialidade)
- Especialidade(idEspecialidade, Nome, SalárioBase, CargaHorária)
- Produto(idProduto, Nome, Preço, /PreçoFinal, QuantidadeStock, idMarca->Marca, idTipodeProduto->Produto)
- Marca(idMarca, Nome)
- TipoDeProduto(idTipoDeProduto, Nome, idSecção->Secção)
- Secção(idSecção, Nome, Dimensão)
- Compra(idCompra, Data, Caixa, /PreçoTotal, NIFCliente->Cliente, NIFStaff->Staff)
- Promoção(idPromoção, Nome, Percentagem, DataInício, DataFim)
- StaffSecção(NIF->Staff, idSecção->Secção)
- ProdutoPromoção(idProduto->Produto, idPromoção->Promoção)
- CompraProduto(idCompra->Compra, idProduto->Produto, Quantidade)

Análise de Dependências Funcionais e Formas Normais

- **Pessoa(NIF, Nome, Idade, Sexo, Morada, Telefone)**

{NIF}-> {Nome, Idade, Sexo, Morada, Telefone}

- **Cliente(NIF->Pessoa, NrCliente)**

{NIF}->{NrCliente}

- **Staff(NIF->Pessoa, NrFuncionário, InícioDeContrato, FimDeContrato, DiaDeFolga, idEspecialidade->Especialidade)**

{NIF}->{Pessoa, NrFuncionário, InícioDeContrato, FimDeContrato, DiaDeFolga, idEspecialidade}

- **Especialidade(idEspecialidade, Nome, SalárioBase, CargaHorária)**

{idEspecialidade}-> {Nome, SalárioBase, CargaHorária}

- **Produto(idProduto, Nome, Preço, /PreçoFinal, QuantidadeStock, idMarca->Marca, idTipodeProduto->Produto)**

{idProduto}->{Nome, Preço, /PreçoFinal, QuantidadeStock, idMarca,idTipodeProduto}

- **Marca(idMarca, Nome)**

{idMarca}->{Nome}

- **TipoDeProduto(idTipoDeProduto, Nome, idSecção->Secção)**

{idTipoDeProduto}->{Nome,idSecção}

- **Secção(idSecção, Nome, Dimensão)**

{idSecção}->{Nome,Dimensão}

- **Compra(idCompra, Data, Caixa, /PreçoTotal, NIFCliente->Cliente, NIFStaff->Staff)**

{idCompra}->{Data,Caixa, /PrecoTotal,NIFCliente,NIFStaff}

- **Promoção(idPromoção, Nome, Percentagem, DataInício, DataFim)**

$\{idPromoção\} \rightarrow \{Nome, Percentagem, DataInício, DataFim\}$

- **StaffSecção(NIF→Staff, idSecção→Secção)**

Não há dependências funcionais.

- **ProdutoPromoção(idProduto→Produto, idPromoção→Promoção)**

Não há dependências funcionais.

- **CompraProduto(idCompra→Compra, idProduto→Produto, Quantidade)**

$\{idCompra, idProduto\} \rightarrow \{Quantidade\}$

Para todas estas dependências, do lado esquerdo está uma chave da relação ou os atributos do lado direito são primos. Logo, não violam nem a 3ª Forma Normal nem a Forma Normal de Boyce-Codd.

Restrições à base de dados

Não pode haver duas pessoas com o mesmo NIF (PRIMARY KEY & NOT NULL). Cada pessoa tem que ter um nome e uma morada (NOT NULL). Cada pessoa não pode ter uma idade negativa e tem que ter um sexo 'M' ou 'F' (CHECK & NOT NULL). Cada pessoa tem um número de telefone único (UNIQUE).

Cada cliente tem um NIF que referencia a tabela Pessoa (chave estrangeira). O número de cliente é único e existe (UNIQUE & NOT NULL).

Cada especialidade tem um id único (PRIMARY KEY) que é gerado na sua criação (AUTOINCREMENT). Tem que ter um nome (NOT NULL). O salário base tem que ser positivo e cada trabalhador só pode trabalhar 8 horas por dia e tem que folgar um dia da semana, limitando a sua carga horária (CHECK & NOT NULL).

Cada staff tem um NIF que referencia a tabela Pessoa (chave estrangeira). O número de funcionário é único e existe (UNIQUE & NOT NULL). Cada funcionário tem um início de contrato (NOT NULL) e se tiver um fim de contrato esse tem que ser após o início (CHECK). Cada staff tem um dia de folga que tem que corresponder a um dia da semana (CHECK & NOT NULL). Cada funcionário tem uma chave estrangeira que referencia a sua especialidade.

Cada marca tem um id único (PRIMARY KEY) que é gerado na sua criação (AUTOINCREMENT). Tem que tem um nome (NOT NULL).

Cada secção tem um id único (PRIMARY KEY) que é gerado na sua criação (AUTOINCREMENT). Tem que tem um nome (NOT NULL). Cada secção tem uma dimensão válida (CHECK & NOT NULL).

Cada tipo de produto tem um id único (PRIMARY KEY) que é gerado na sua criação (AUTOINCREMENT). Tem que tem um nome (NOT NULL). Cada tipo de produto está associado a uma secção (chave estrangeira).

Cada produto tem um id único (PRIMARY KEY) que é gerado na sua criação (AUTOINCREMENT). Tem que tem um nome (NOT NULL). O preço tem que ser positivo (CHECK & NOT NULL). O preço final, se uma promoção estiver ativa para

esse produto, será a aplicação dessa promoção no preço base (não é possível tratar esta restrição nesta etapa do projeto). A quantidade de stock não pode ser negativa (CHECK & NOT NULL) e será alterada se o produto estiver envolvido numa compra (não é possível tratar esta restrição nesta etapa do projeto). Cada produto tem uma marca e um tipo de produto associados (chave estrangeira).

Cada compra tem um id único (PRIMARY KEY) que é gerado na sua criação (AUTOINCREMENT). Tem que tem uma data e uma caixa (NOT NULL). O preço total da compra corresponde à soma de todos os produtos comprados (não é possível tratar esta restrição nesta etapa do projeto). Cada compra está associada a um cliente e a um staff (chave estrangeira) e tem que ter pelo menos um produto associado.

Cada promoção tem um id único (PRIMARY KEY) que é gerado na sua criação (AUTOINCREMENT). Tem que tem um nome (NOT NULL). A percentagem de desconto tem que estar entre 0% e 100% (CHECK & NOT NULL). Cada promoção tem que ter data de início e fim (NOT NULL) e a data de fim tem que ser após a de início (CHECK).

Cada relação entre staff e secção tem o NIF do staff associado (chave estrangeira) e o id da secção associado (chave estrangeira) cujo conjunto é único (PRIMARY KEY).

Cada relação entre produto e promoção tem o id do produto associado (chave estrangeira) e o id da promoção associado (chave estrangeira) cujo conjunto é único (PRIMARY KEY).

Cada relação entre compra e produto tem o id do compra associado (chave estrangeira) e o id do produto associado (chave estrangeira) cujo conjunto é único (PRIMARY KEY). A quantidade de produto comprado tem que ser positiva (CHECK & NOT NULL).

Todos os atributos que referenciam outra tabela estão com a definição de ON DELETE CASCADE ON UPDATE CASCADE.

Interrogação de Base de Dados

1. Quantos produtos há?
2. Qual a idade média dos clientes aderentes à MEO (92...)?
3. Quantas compras realizou cada funcionário?
4. Quais as compras que tiveram valor superior a 20€ ?
5. Quais os funcionários que também são clientes?
6. Quais os produtos que se encontram numa secção maior que 20 m² ?
7. Quais os funcionários que iniciaram funções a partir de 2021 ordenados pela data de contratação?
8. Qual o total gasto por cada cliente no supermercado desde que são clientes?
9. Quanto se poupa em cada produto com as promoções ?
10. Quais os clientes que beneficiaram nas suas compras com a promoção de Natal?

Gatilhos da Base de Dados

1. Gatilho que verifica se um produto está com desconto
2. Gatilho que calcula o preço total de uma compra
3. Gatilho que garante que uma promoção é no futuro

Todos os elementos contribuíram de forma igual para a realização do projeto estando sempre todos disponíveis para o realizar.