O projeto final de Banco de dados deve conter as tabelas obrigatórias:

● Cliente

● Pedido

● PedidoItem

● Produto

● Categoria

● NotaFiscal

● Usuário

Levantamento de requisitos:

E-commerce tradicional

Uma empresa do comércio varejista deseja realizar a venda de seus produtos pela

internet e para isso precisa de um sistema **capaz de realizar o controle de estoque e**

**cadastro de clientes**. A empresa é responsável pela produção de seus próprios

produtos e por isso não precisa registrar dados de fornecedores externos. O sistema

não tem como objetivo registrar detalhes da produção. Sendo assim, a seguir são

descritos os requisitos e os dados que a empresa julga serem necessários para o

funcionamento do sistema:

1. O sistema deve ser capaz de armazenar informações sobre os produtos da

empresa, sendo eles: código, nome, descrição, quantidade em estoque, data

de fabricação e valor unitário.

2. Ao cadastrar um produto no sistema, os funcionários da empresa devem ser

capazes de associá-lo à uma categoria. Cada produto só poderá pertencer à

uma categoria.

3. Se o produto já estiver cadastrado no sistema, o colaborador deverá apenas

atualizar a quantidade de itens no estoque.

4. Para cada categoria de produto, é necessário registrar: código, nome e

descrição.

5. O sistema deverá armazenar dados dos clientes, que devem ser: código,

nome completo, nome de usuário, e-mail, cpf, data de nascimento e

endereço.

**6. Através do sistema, os clientes podem realizar pedidos.**

7. Cada pedido deve conter um ou mais produtos, a data em que foi realizado e

por quem foi realizado.

8. Um pedido pode ter mais de um item de um mesmo produto, de acordo com

a quantidade disponível em estoque.

9. Sobre os funcionários, é necessário guardar apenas um código, nome e cpf.

PLUS: Caso consigam, seria interessante saber qual funcionário cadastrou o produto.

Aplicando o que aprendemos:

1. Diagrama ER (primeira versão)

2. Diagrama ER (depois da normalização)

3. SQL de criação das tabelas

CREATE DATABASE ecommerce;

CREATE TABLE usuario(

  cod\_usario SERIAL PRIMARY KEY,

  nome\_usuario varchar(50) NOT NULL UNIQUE,

  email varchar(120) NOT NULL UNIQUE,

  senha char(8)

);

CREATE TABLE cliente(

  cod\_cliente SERIAL PRIMARY KEY,

  nome\_completo varchar (50) NOT NULL,

 fk\_nome\_usuario varchar(50) NOT NULL UNIQUE REFERENCES usuario(nome\_usuario) ON UPDATE CASCADE,

  fk\_email varchar(120) NOT NULL UNIQUE REFERENCES usuario(email) ON UPDATE CASCADE,

  cpf char(11) NOT NULL UNIQUE,

  data\_nascimento date NOT NULL,

  endereco varchar(200) NOT NULL

);

CREATE TABLE funcionario(

  cod\_funcionario SERIAL PRIMARY KEY,

  nomeCompleto varchar(50) NOT NULL,

  cpf char(11) NOT NULL UNIQUE

);

CREATE TABLE categoria(

  cod\_catergoria SERIAL PRIMARY KEY,

  nome varchar(50) NOT NULL UNIQUE,

  descricao varchar(200)

);

CREATE TABLE produto(

  cod\_produto SERIAL PRIMARY KEY,

  nome varchar(50) NOT NULL UNIQUE,

  descricao varchar(200),

  qtd\_em\_estoque int,

  data\_fabricacao date,

  valor\_unitario float,

  fk\_cod\_catergoria int REFERENCES categoria(cod\_catergoria) ON UPDATE CASCADE,

  fk\_cod\_funcionario int REFERENCES funcionario(cod\_funcionario) ON UPDATE CASCADE

);

CREATE TABLE pedido(

  cod\_pedido SERIAL PRIMARY KEY,

  fk\_cod\_cliente int REFERENCES cliente(cod\_cliente) ON UPDATE CASCADE,

  data\_pedido date NOT NULL

);

CREATE TABLE pedidoItem(

  fk\_cod\_pedido int NOT NULL REFERENCES pedido(cod\_pedido) ON UPDATE CASCADE,

  cod\_item int NOT NULL,

  fk\_cod\_produto int NOT NULL REFERENCES produto(cod\_produto) ON UPDATE CASCADE,

  qtd\_produto int NOT NULL,

  PRIMARY KEY (fk\_cod\_pedido, cod\_item)

);

CREATE TABLE nota\_fiscal(

  numero\_notafiscal SERIAL PRIMARY KEY, --tbm o codigo da tabela

  fk\_cod\_cliente int REFERENCES cliente(cod\_cliente) ON UPDATE CASCADE,

  fk\_cod\_pedido int REFERENCES pedido(cod\_pedido) ON UPDATE CASCADE

);

4. SQL de inserção de dados nas tabelas (pelo menos 5 registros em

cada uma)

**INSERT INTO usuario(nome\_usuario, email, senha)**

**VALUES ('rafaalves', 'rafaalves.s@gmail.com', 'rafa1234'),**

**('gillima', 'gil.Lima@gmail.com','gil12345'),**

**('tardin', 'tardincarol@gmail.com', '@carol78'),**

**('marcella', 'malzuguir@gmail.com', 'ma#12345'),**

**('lfita', 'lfita@gmail.com', '87654321'),**

**('dfaria', 'diegofaria13@gmail.com', 'diegofa#');**

**INSERT INTO cliente(nome\_completo, fk\_nome\_usuario, fk\_email, cpf, data\_nascimento, endereco)**

**VALUES**

**('Rafael Alves de Souza', 'rafaalves', 'rafaalves.s@gmail.com', '15426575470', '1999-10-08', 'Rua carmelia Lucinda N:188'),**

**('Gilnei Lima da Costa', 'gillima', 'gil.Lima@gmail.com', '12423688471', '1987-07-01', 'Rua Jorge Pinto N:03'),**

**('Carolina Tardin', 'tardin', 'tardincarol@gmail.com', '38429576120', '1995-12-5', 'Rua Carmani Freitas N:732'),**

**('Marcella Alzuguir', 'marcella', 'malzuguir@gmail.com', '55729475620', '1990-04-10', 'Rua Pablo Escobar N:3445'),**

**('Leandro Fita', 'lfita', 'lfita@gmail.com', '45426577461', '1998-03-12', 'Rua Lacerda Costa N:664'),**

**('Diego Faria', 'dfaria', 'diegofaria13@gmail.com', '20124287461', '1989-04-30', 'Rua Lipp Souza N:999');**

**INSERT INTO funcionario(nomeCompleto, cpf)**

**VALUES**

**('Maria da Silva', '75864669040'),**

**('Joao Oliveira', '65688040061'),**

**('Juliana Neves', '94367851001'),**

**('Miguel Xavier', '31742757090'),**

**('Joana Duarte', '49293208016');**

**INSERT INTO categoria(nome, descricao)**

**VALUES**

**('Smartphones', 'Aparelho celular com acesso a internet'),**

**('Eletrodomésticos', 'Aparelhos domésticos eletrônicos'),**

**('Roupa', 'Camisas, calças, bermudas, vestidos, blusas e mais.'),**

**('Produtos de limpeza', 'Produtos para limpeza de ambientes'),**

**('Beleza', 'Produtos de beleza');**

**INSERT INTO produto(nome, descricao, qtd\_em\_estoque, data\_fabricacao, valor\_unitario, fk\_cod\_catergoria, fk\_cod\_funcionario)**

**VALUES**

**('Calça Jeans', 'Calça masculina reta', 20, '2020-08-30', 60.99, 3, 2),**

**('Camisa', 'Camisa gola V, branca', 15, '2020-07-26', 45.99, 3, 1),**

**('Bermuda', 'Bermuda cargo verde musgo', 45, '2020-12-06', 75.99, 3, 4),**

**('Desodorante', 'Desodorante 24h, floral, spray', 55, '2020-09-10', 9.90, 4, 4),**

**('Saia', 'Saia reta vermelha', 15, '2020-06-05', 85.99, 3, 4),**

**('Celular', 'Smartphone 5, Android, azul', 100, '2021-03-25', 805.99, 1, 4);**

**INSERT INTO pedido(fk\_cod\_cliente,data\_pedido)**

**VALUES**

**(1, '2021-07-25'),**

**(2, '2021-08-14'),**

**(3, '2021-08-18'),**

**(4, '2021-09-05'),**

**(5, '2021-09-05'),**

**(6, '2021-09-10');**

**INSERT INTO pedidoItem(fk\_cod\_pedido, cod\_item, fk\_cod\_produto, qtd\_produto)**

**VALUES (1,1,1,5), (1,2,5,20), (1,3,3,15), (1,4,1,20), (2,1,2,30), (2,2,1,0), (3,1,1,2), (3,2,2,4), (4,1,3,12), (4,2,1,10), (5,1,3,21);**

**INSERT INTO nota\_fiscal(fk\_cod\_cliente, fk\_cod\_pedido)**

**VALUES (1,1), (2,2), (3,3), (4,4), (5,5);**

5. Um comando SQL de atualização em algum registro em uma tabela

UPDATE cliente

SET nome\_completo = 'Caio Dantas' WHERE cod\_cliente = 5

UPDATE cliente

SET endereco  = 'Rua Coltinho Souza Lheiras' WHERE cod\_cliente = 3

6. Um comando SQL de exclusão de algum registro em uma tabela

DELETE FROM funcionario

WHERE nomeCompleto LIKE 'Joana %'

7. Cinco SQLs de consulta

a. Pelo menos 2 com algum tipo de junção

SELECT prod.nome, cli.nome\_completo, ped.data\_pedido FROM produto AS prod

INNER JOIN cliente AS cli ON prod.cod\_produto = cli.cod\_cliente

INNER JOIN pedido AS ped ON prod.cod\_produto = ped.cod\_pedido

SELECT prod.nome AS Produto, cat.nome AS Categoria FROM produto AS prod

INNER JOIN categoria AS cat ON prod.fk\_cod\_catergoria = cat.cod\_catergoria

SELECT nome\_completo, nome,  valor\_unitario, qtd\_produto, qtd\_produto \* valor\_unitario

FROM cliente as c INNER JOIN pedido as p ON c.cod\_cliente = p.fk\_cod\_cliente

INNER JOIN pedidoItem as pi ON pi.fk\_cod\_pedido = p.cod\_pedido

INNER JOIN produto as pdt ON pdt.cod\_produto = pi.fk\_cod\_produto;

b. Pelo menos 1 com usando count() e group by()

SELECT fk\_cod\_produto AS codProduto, COUNT(cod\_item) as tot\_PEDIDO, SUM(qtd\_produto) as total\_PRODUTO

FROM pedidoItem

GROUP BY fk\_cod\_produto

ORDER BY codProduto;

c. 1 SQL para construção de nota fiscal

SELECT numero\_notafiscal, cod\_cliente, nome\_completo, fk\_email, cpf, data\_nascimento, endereco, pi.fk\_cod\_pedido, cod\_item, fk\_cod\_produto, qtd\_produto

FROM nota\_fiscal as nf INNER JOIN cliente as c ON nf.fk\_cod\_cliente = c.cod\_cliente

INNER JOIN pedido as p ON nf.fk\_cod\_pedido = p.cod\_pedido

INNER JOIN pedidoItem as pi ON p.cod\_pedido = pi.fk\_cod\_pedido;