CENTRO UNIVERSITÁRIO UNIFECAF

GRADUAÇÃO EAD

Marcello Henrique Tura Fonseca

Sistema de Gestão Comercial

Controle completo de estoque, vendas e fornecedores em um único ambiente integrado.

Jundiaí

2025

Marcello Henrique Tura Fonseca

Trabalho apresentado como requisito parcial de avaliação da disciplina **SQL e banco de dados** do Curso de Graduação em **Análise e desenvolvimento de sistemas** do Centro Universitário UniFECAF.

Tutor: ***Fernando Leonid***

Jundiaí

2025

**SUMÁRIO**

Sumário

[1. CENÁRIO 1](#_Toc195857743)

[2. lEVANTAMENTO DE REQUISTOS 2](#_Toc195857744)

[3. MODELO CONCEITUAL 3](#_Toc195857745)

[4. MODELO LÓGICO 4](#_Toc195857746)

[5. MODELO FÍSICO 5](#_Toc195857747)

# CENÁRIO

Os proprietários de um **supermercado** precisam de um sistema que viabilize o armazenamento de informações sobre seus produtos, colaboradores e clientes para que quando ocorram as vendas todas as informações possam ser guardadas em uma base de dados segura, para que estes dados sirvam de insumos para realização de futuras vendas que possam fidelizar os clientes que costumam realizar compras neste supermercado. Por exemplo: Armazenando as informações das vendas realizadas para seus clientes, a empresa terá informações sobre as preferências de compras dos clientes, portanto será mais fácil enviar sugestões de compras direcionadas aos produtos que aquele cliente costuma comprar

# lEVANTAMENTO DE REQUISTOS

O levantamento de requisitos é uma etapa essencial para compreender as necessidades do cliente e garantir o sucesso do projeto. Para isso, foram elaboradas 10 perguntas estratégicas que serão utilizadas durante a entrevista com o cliente e simulado as respostas, com o objetivo de coletar informações detalhadas e alinhar expectativas.

-Quais informações dos clientes você precisa armazenar para melhorar a fidelização?

-Precisamos registrar o nome, CPF, RG, e-mail, telefone e endereço dos clientes, esses dados nos permitem segmentar clientes e oferecer promoções personalizadas com base nos hábitos de compra.

-Como você quer que o sistema acompanhe o histórico de compras dos clientes?

-Gostaria que ele armazenasse a data da compra, valor total, numero do pedido e produtos comprados. Isso nos ajudaria a entender o que cada cliente costuma comprar e sugerir produtos relevantes.

-Quais informações sobre os funcionários são necessárias para a gestão interna?

-Precisamos registrar nome, cargo, setor, turno, data de admissão, salário e e-mail. Também seria interessante associar cada funcionário as vendas que ele realiza, para podermos analisar o desempenho.

-Você precisa que o sistema armazene dados sobre fornecedor? Se sim, quais seriam relevantes?

-Sim, queremos manter um controle dos nossos fornecedores. Precisamos armazenar o CNPJ, nome, e-mail e os produtos que cada fornecedor nos entrega, para facilitar negociações e reposições de estoque.

-Como deve ser feito o controle de estoque e dos preços dos produtos?

-O sistema deve armazenar o nome do produto, código de barras, categoria, quantidade disponível, preço de custo e preço de venda. Também seria útil ter um histórico de preços, para acompanhar ajustes e promoções.

-Quais métodos de pagamento o sistema deve aceitar e como quer que eles sejam registrados?

-Aceitamos cartão de credito, debito, pix e dinheiro. Cada venda pode ter mais de um método de pagamento. O sistema deve permitir registrar quanto foi pago em cada forma e vincular isso à compra.

-Como deseja que os endereços dos clientes, funcionários e fornecedores sejam registrados?

-Cada entidade deve ter seu próprio endereço, contendo logradouro, cep, bairro, estado e país, para facilitar entregas e cadastros.

-Como o sistema pode facilitar a comunicação entre o supermercado e os clientes?

-O sistema deve armazenar todos os números de telefone que o cliente possuir e e-mail dos clientes para que possamos contata-los, enviar confirmações de compra e oferecer suporte quando necessário.

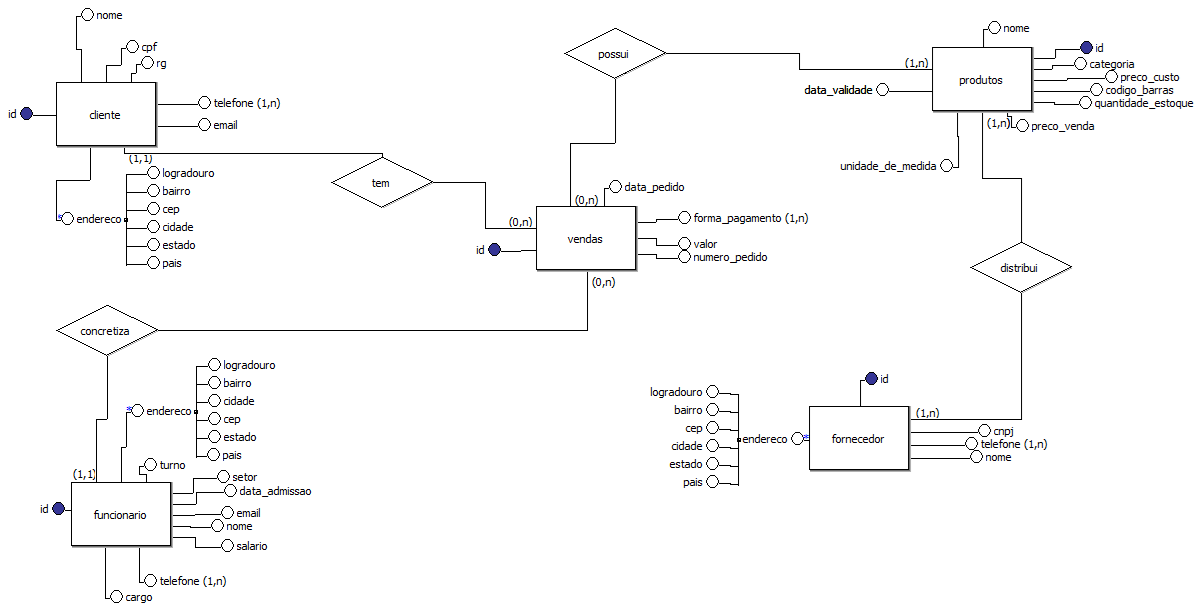
-Como o sistema deve lidar com clientes que realizam múltiplas compras ao longo do tempo?

-O sistema deve vincular todas as compras ao cliente correspondente, permitindo o acompanhamento do histórico de compras e facilitando analises sobre os hábitos de consumo.

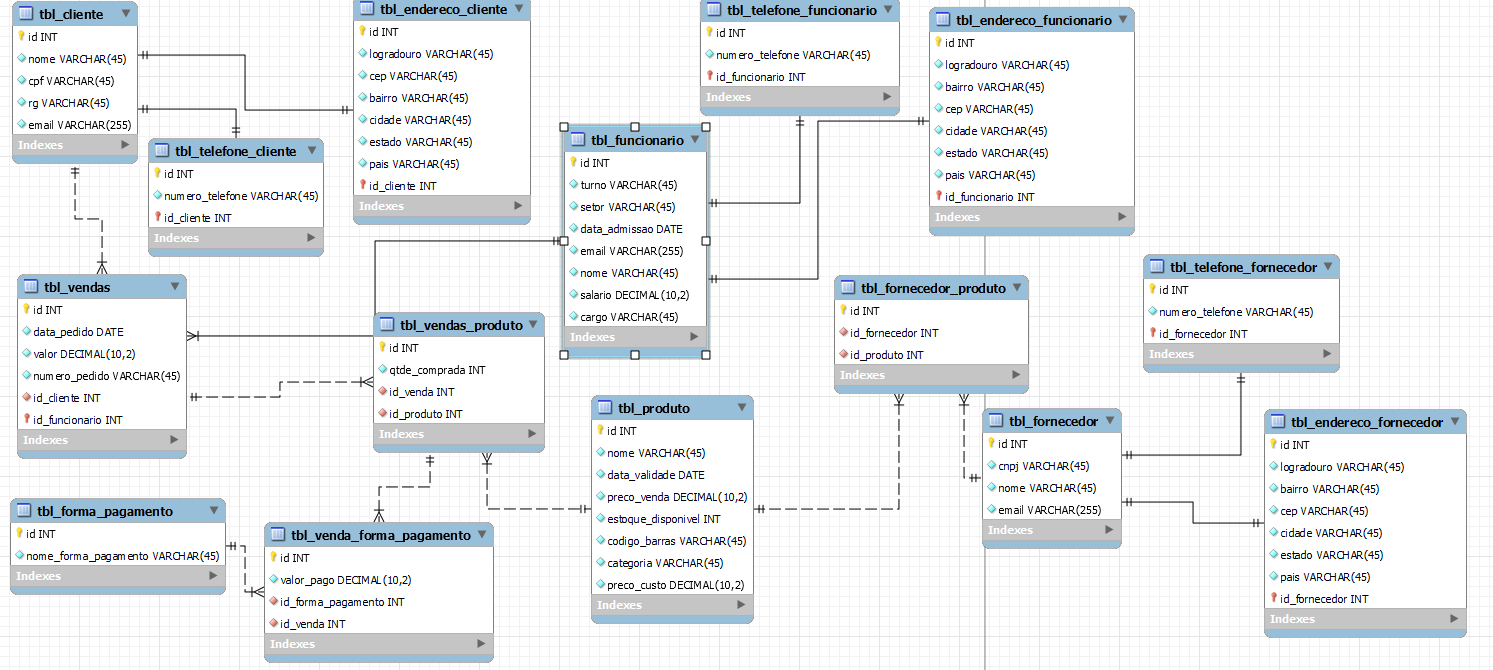
-Como o sistema deve registrar a relação entre os produtos e os fornecedores?

-Cada produto deve estar vinculado ao fornecedor que o fornece, permitindo que possamos rastrear a origem dos itens vendidos e gerenciar compras de reposição de estoque de maneira eficiente.

## MODELO CONCEITUAL



## MODELO LÓGICO



## MODELO FÍSICO

create database db\_sistema\_mercado;

use db\_sistema\_mercado;

create table tbl\_cliente (

id int not null primary key auto\_increment,

nome varchar(45) not null,

cpf varchar(45) not null,

rg varchar(45) not null,

email varchar(255) not null,

unique index unq\_nome (nome),

unique index unq\_cpf (cpf),

unique index unq\_rg (rg),

unique index unq\_email (email)

);

create table tbl\_funcionario (

id int not null primary key auto\_increment,

nome varchar(45) not null,

turno varchar(45) not null,

setor varchar(45) not null,

data\_admissao date not null,

email varchar(255) not null,

salario decimal(10,2) not null,

cargo varchar(45) not null,

unique index unq\_id (id),

unique index unq\_nome (nome),

unique index unq\_email (email)

);

create table tbl\_fornecedor (

id int not null primary key auto\_increment,

cnpj varchar(45) not null,

nome varchar(45) not null,

email varchar(255) not null,

unique index unq\_cnpj (cnpj),

unique index unq\_nome (nome),

unique index unq\_email (email)

);

create table tbl\_produto (

id int not null primary key auto\_increment,

nome varchar(45) not null,

data\_validade date not null,

preco\_venda decimal(10,2) not null,

estoque\_disponivel int not null,

codigo\_barras varchar(45) not null,

categoria varchar(45) not null,

preco\_custo decimal(10,2) not null

);

create table tbl\_forma\_pagamento (

id int not null primary key auto\_increment,

nome\_forma\_pagamento varchar(45) not null

);

create table tbl\_telefone\_cliente (

id int not null primary key auto\_increment,

numero\_telefone varchar(45) not null,

id\_cliente int not null,

constraint FK\_cliente\_telefone\_cliente foreign key (id\_cliente)

references tbl\_cliente(id),

unique index unq\_numero\_telefone (numero\_telefone)

);

create table tbl\_endereco\_cliente (

id int not null primary key auto\_increment,

logradouro varchar(45) not null,

cep varchar(45) not null,

bairro varchar(45) not null,

cidade varchar(45) not null,

estado varchar(45) not null,

pais varchar(45) not null,

id\_cliente int not null,

constraint FK\_cliente\_endereco\_cliente foreign key (id\_cliente)

references tbl\_cliente(id)

);

create table tbl\_vendas (

id int not null primary key auto\_increment,

data\_pedido date not null,

valor decimal(10,2) not null,

numero\_pedido varchar(45) not null,

id\_cliente int not null,

id\_funcionario int not null,

constraint FK\_cliente\_vendas\_cliente foreign key (id\_cliente)

references tbl\_cliente(id),

constraint FK\_funcionario\_vendas\_funcionario foreign key (id\_funcionario)

references tbl\_funcionario(id),

unique index unq\_numero\_pedido (numero\_pedido)

);

create table tbl\_venda\_forma\_pagamento (

id int not null primary key auto\_increment,

valor\_pago decimal(10,2) not null,

id\_forma\_pagamento int not null,

id\_vendas int not null,

constraint FK\_venda\_forma\_pagamento\_forma\_pagamento foreign key (id\_forma\_pagamento)

references tbl\_forma\_pagamento(id),

constraint FK\_vendas\_forma\_pagamento\_venda foreign key (id\_vendas)

references tbl\_vendas(id)

);

create table tbl\_vendas\_produto (

id int not null primary key auto\_increment,

qtde\_comprada int not null,

id\_vendas int not null,

id\_produto int not null,

constraint FK\_vendas\_produto\_vendas foreign key (id\_vendas)

references tbl\_vendas(id),

constraint FK\_vendas\_produto\_produto foreign key (id\_produto)

references tbl\_produto(id)

);

create table tbl\_fornecedor\_produto (

id int not null primary key auto\_increment,

id\_fornecedor int not null,

id\_produto int not null,

constraint FK\_fornecedor\_produto\_fornecedor foreign key (id\_fornecedor)

references tbl\_fornecedor(id),

constraint FK\_fornecedor\_produto\_produto foreign key (id\_produto)

references tbl\_produto(id)

);

create table tbl\_telefone\_funcionario (

id int not null primary key auto\_increment,

numero\_telefone varchar(45) not null,

id\_funcionario int not null,

constraint FK\_telefone\_funcionario\_funcionario foreign key (id\_funcionario)

references tbl\_funcionario(id),

unique index unq\_numero\_telefone (numero\_telefone)

);

create table tbl\_endereco\_funcionario (

id int not null primary key auto\_increment,

logradouro varchar(45) not null,

cep varchar(45) not null,

bairro varchar(45) not null,

cidade varchar(45) not null,

estado varchar(45) not null,

pais varchar(45) not null,

id\_funcionario int not null,

constraint FK\_endereco\_funcionario\_funcionario foreign key (id\_funcionario)

references tbl\_funcionario(id)

);

create table tbl\_telefone\_fornecedor (

id int not null primary key auto\_increment,

numero\_telefone varchar(45) not null,

id\_fornecedor int not null,

constraint FK\_telefone\_fornecedor\_fornecedoor foreign key (id\_fornecedor)

references tbl\_fornecedor(id),

unique index unq\_numero\_telefone (numero\_telefone)

);

create table tbl\_endereco\_fornecedor (

id int not null primary key auto\_increment,

logradouro varchar(45) not null,

bairro varchar(45) not null,

cep varchar(45) not null,

cidade varchar(45) not null,

estado varchar(45) not null,

pais varchar(45) not null,

id\_fornecedor int not null,

constraint FK\_endereco\_fornecedor\_fornecedor foreign key (id\_fornecedor)

references tbl\_fornecedor(id)

);