



**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E
TECNOLOGIA DE SANTA CATARINA
CAMPUS FLORIANÓPOLIS**

Projeto Final Análise de Sistemas I e Banco de Dados I

Loja de Artigos Esportivos

ANA ELISA GHANEM ZANON
JULIANA HACHMANN

**Florianópolis, SC
Dezembro, 2022**

Sumário

1. Introdução	3
2. Descrição do Sistema	3
3. Telas	5
4. Requisitos do Sistema	10
4.1 Requisitos Funcionais (RF) e Regras de negócio (RN)	10
4.2 Requisitos Não Funcionais (RNF)	12
5. Diagramas de Casos de Uso	13
6. Diagrama de Entidade-Relacionamento (DER)	16
7. Scripts SQL	17
7.1 Script DDL	17
7.2 Script DML	19
7.3 Visualizações	21
8. Referências	23

1. Introdução

O aplicativo tem como proposta a **criação de um sistema de uma loja de artigos esportivos** que possibilite que funcionários da empresa utilizem para cadastrar e buscar informações para a empresa.

Durante a **entrevista** foi apontado pelo cliente que necessitaria de um aplicativo para celular (Android) e *desktop* para acesso às informações e que fosse um aplicativo prático de uso. Seguindo o princípio de desenvolvimento *mobile first*, foram criadas as telas do sistema para uso em celulares, que servirão de base para a implementação do sistema em resoluções maiores posteriormente.

Observou-se durante a análise de requisitos que seria necessário as funcionalidades de cadastro, busca, exclusão e alteração de informações sobre: produtos, categorias de produto, clientes, funcionários, categoria de funcionário, cupons de desconto, comissão de funcionários e vendas.

Para este sistema foi criado um banco de dados utilizando o Sistema Gerenciador de Banco de Dados (SGBD) *MySQL*, para que as informações sejam gravadas e acessadas pelo cliente.

2. Descrição do Sistema

O sistema será utilizado especialmente pelos funcionários dos Caixas nas lojas físicas da empresa e pelos gerentes responsáveis pelo cadastro de produtos e funcionários. Assim, o sistema foi desenhado para comportar dois tipos de usuários com permissões de acesso diferentes. O usuário com permissão de acesso restrita pode cadastrar e buscar produtos e suas categorias, clientes e vendas. Já o usuário “Gerente”, possui acesso completo às funcionalidades adicionais de cadastro de novos funcionários, visualização de relatórios e alterações nas configurações do sistema.

Por meio do sistema, no seu estado atual, é possível aos usuários:

- Cadastrar as vendas, calculando o valor total a partir dos dados dos produtos e também calculando descontos a partir de cupons. Além disto, este cadastro

gera automaticamente o registro das vendas feitas por cada funcionário e por cada cliente;

- Criar e enviar cupons de desconto automaticamente, a partir de critérios estipulados pelo gerente, que configura regras de valor e frequência de compras para que o cliente receba os cupons. Estes cupons podem ser enviados previamente aos clientes ou gerados no momento de cadastro da venda;
- Gerar relatórios de venda por período;
- Cadastrar novos produtos e novas categorias de produtos, com informações detalhadas sobre eles;
- Controlar automaticamente o estoque, que é atualizado de acordo com os registros de venda de produtos;
- Cadastrar novos clientes e atualizar cadastros já existentes no banco de dados;
- Cadastrar novos funcionários e novas categorias de funcionários, atualizar suas informações e permissões de acesso (por meio das categorias), bem como ter um controle dos funcionários que estão ativos ou em férias e de seus salários atuais.


Além destas funcionalidades, o sistema e o banco de dados poderão ser interligados a outros que não estão no escopo deste projeto. Como exemplo, citamos o sistema de RH da empresa, que poderá solicitar os dados de vendas de funcionários para o pagamento de comissões. Sistemas geradores de Notas Fiscais e de Pagamentos poderão ser interligados aos dados das vendas. Além disso, o cadastro de produtos e estoque pode servir a um futuro sistema de *Marketplace* a ser implementado na empresa.

3. Telas

Link Quant-UX para teste:

<https://www.quant-ux.com/#/share.html?h=a2aa10aTpZ4wmYruVKzsH5HoZTC3OFqDrJhAioaZRO2xzLWuYrYyo8PcKwqu>

Login




Loja de Artigos Esportivos

☐ Salvar Login


Login

Login Gerente


Cadastro Func


 Seja Bem Vindo(a) João!
CPF: 098.974.938-82


Funcionário


O que deseja fazer? 

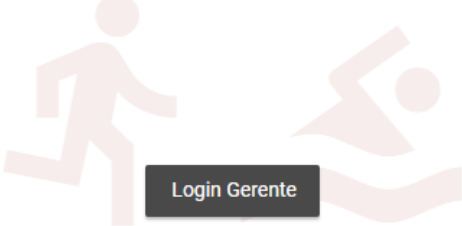
Cadastrar

 Cadastrar uma Venda

 Cadastrar um Produto


 Cadastrar uma Categoria


 Cadastrar um Cliente




Login Gerente

Cadastro Gerente


 **Seja Bem Vindo(a) Maria!**
CPF: 938.973.934-43 Gerente


O que deseja fazer? 


Cadastrar


 Cadastrar uma Venda


 Cadastrar um Produto


 Cadastrar uma Categoria

 Cadastrar um Cliente



 Cadastrar um Funcionário

 Cadastrar Cat. Funcionário


 Configurar um Cupom

 Vendas

Cadastro de venda

  Cadastrar uma Venda

Código da Venda
001

Data da Venda
Selecione uma data 

Valor
Digite o Valor

Código do Cupom
Digite o Valor

Forma de Pagamento
Cartão de Crédito

Código do Cliente
Digite o Código do Cliente

Código do Atendente
Digite o Código do Atendente

Salvar

Cadastro de produto

  Cadastrar um Produto

Código do Produto
001

Categoria do Produto
Tênis

Nome do Produto
Digite o Nome do Produto

Descrição
Digite a Descrição



Preço
Digite o preço

Quantidade em Estoque
1

Link
Digite o link do produto

Salvar

Cadastro de categoria de produto

  Cadastrar Categorias



Código da Categoria
001

Nome da Categoria
Digite o Nome da Categoria

Descrição
Digite a Descrição da Categoria


Salvar

Cadastro de Cliente

 **Cadastrar um Cliente**

Nome

CPF

Data da Venda
 

Telefone do Cliente


Endereço do Cliente

E-mail do cliente

Observações

Salvar

Vendas



 **Vendas**

Selecione o Mês

Valor

Voltar

Cadastro de categoria de funcionário

 **Cadastrar Categorias de Funcionários**



Código da Categoria

Nome da Categoria

Descrição

Salvar

Configurações de Cupom

 **Configurar um Cupom**


Código do Cupom

Valor do Desconto

Periodicidade do Desconto


Salvar

Cadastro de Funcionário

 **Cadastrar um Funcionário**

Nome do Funcionário
Digite o Nome do Funcionário

CPF
Digite o CPF do Funcionário

Data de Nascimento
Selecione uma data 

Telefone do Funcionário
Telefone do Funcionário


Endereço do Funcionário
Endereço do Funcionário

E-mail do Funcionário
E-mail do Funcionário

Salário
Digite o valor do salário

Categoria do Funcionário
1 - Atendente ▼

Período
Matutino ▼


Data da Contratação
Selecione uma data 


Status
Ativo ▼

Observações
Digite as observações sobre o funcionário

Salvar

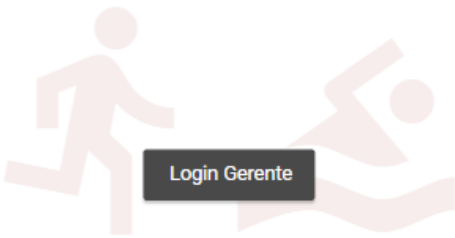
Opções

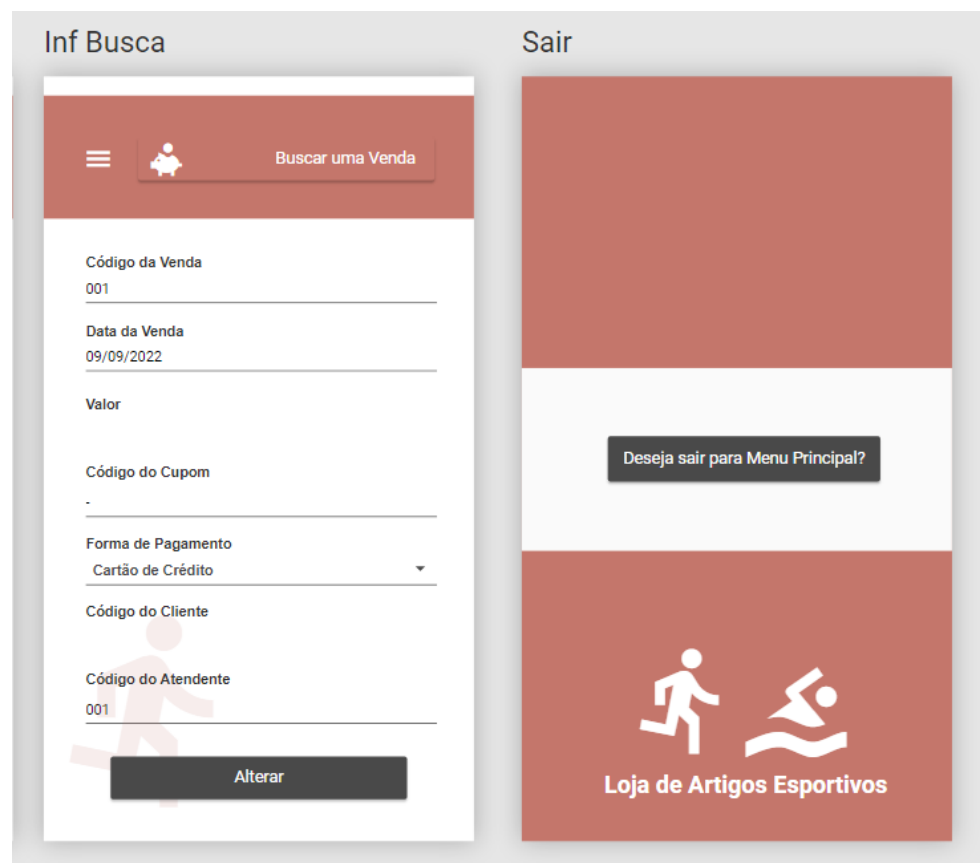
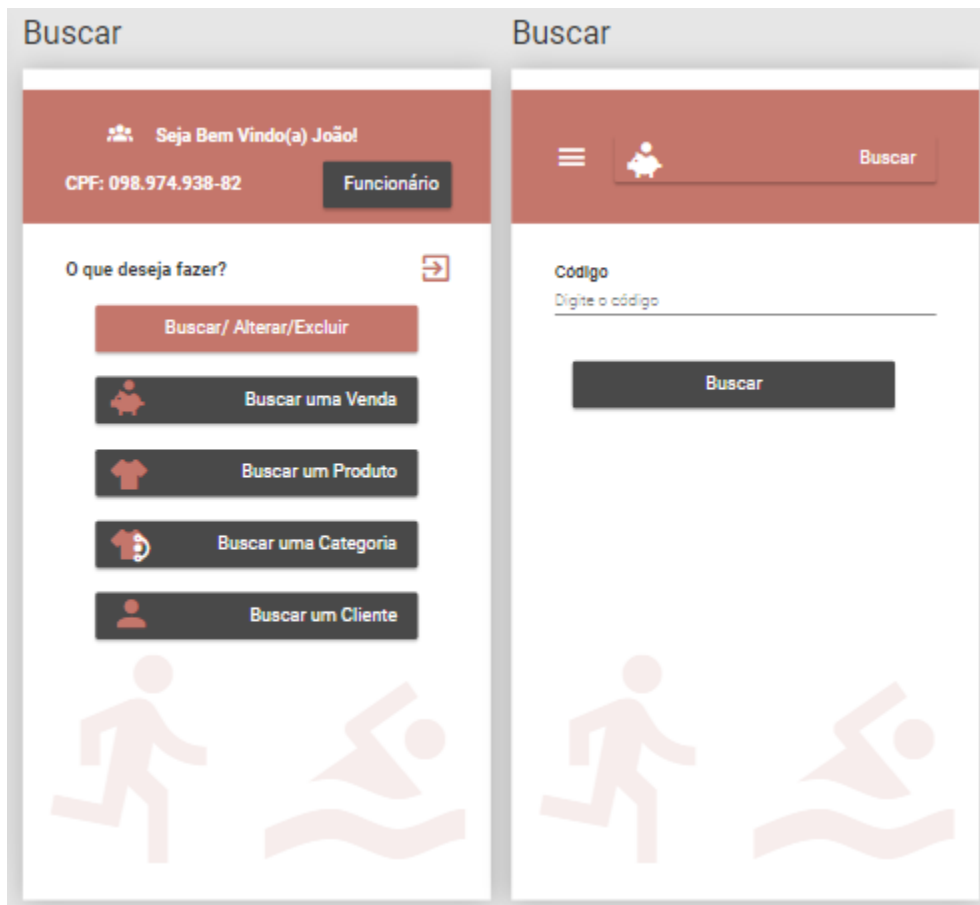
 **Seja Bem Vindo(a) João!**
CPF: 098.974.938-82 **Funcionário**

O que deseja fazer? 

Cadastrar

Buscar/ Alterar/Excluir

**Login Gerente**



4. Requisitos do Sistema

4.1 Requisitos Funcionais (RF) e Regras de negócio (RN)

RF.001 - O sistema deve permitir **cadastrar produto**. (CRUD)

RN.001 - O cadastro de produto deve ter os seguinte campos:

- Código produto (campo obrigatório, auto-incremental);
- Categoria (campo obrigatório);
- Descrição (campo obrigatório);
- Preço (campo obrigatório);
- Quantidade em estoque (campo obrigatório);
- Imagem produto.

RF.002 - O sistema deve permitir **cadastrar o cliente**.

RN.002 - O cadastro do cliente deve ter os seguinte campos:

- CPF do cliente (campo obrigatório);
- Nome (campo obrigatório);
- Data nascimento (campo obrigatório);
- Endereço (campo obrigatório);
- E-mail (campo obrigatório);
- Telefone (campo obrigatório);
- Observações

RN.003 - No campo CPF, criar a rotina de validar CPF.

- Fórmula: pegar fórmula internet

RF.003 - O sistema deve permitir **cadastrar o funcionário**.

RN.004 - O cadastro do funcionário deve ter os seguinte campos:

- CPF do funcionário (campo obrigatório);
- Nome (campo obrigatório);
- Data nascimento (campo obrigatório);
- Endereço (campo obrigatório);
- E-mail (campo obrigatório);
- Telefone (campo obrigatório);
- Salário (campo obrigatório);
- Código Categoria (campo obrigatório);
 - Selecionar: Atendente (1), Caixa (2), Gerente (3), Serviços Gerais (4), Financeiro (5) e Outros (6).
- Vendas*
 - Somente para atendente
- Período (campo obrigatório);
 - matutino ou vespertino
- Data da contratação (campo obrigatório);
 - Máscara: data DD/MM/AAAA

- Status (ativo, férias, ausência);
- Observações.

RN.005 - No campo CPF, criar a rotina de validar CPF.

- Fórmula: pegar fórmula internet

RF.004 - O sistema deve permitir **cadastro de categoria de produto**.

RN.006 - O cadastro de categoria de produto deve ter os seguinte campos:

- Código (auto-incremental);
- Nome (campo obrigatório);
- Descrição.

RF.005 - O sistema deve permitir **cadastro de venda**.

RN.007 - O cadastro de venda deve ter os seguinte campos:

- Código da venda (auto-incremental);
- Data de venda (campo obrigatório);
 - Máscara: data DD/MM/AAAA
- Valor total (campo obrigatório);
- Número do cupom de desconto;
- Código do produto (campo obrigatório);
- Quantidade de produtos (campo obrigatório);
- Forma de pagamento (campo obrigatório);
 - Selecionar: Cartão de crédito parcelado, cartão de crédito á vista, cartão de débito, dinheiro
- CPF do cliente (campo obrigatório);
- CPF do caixa (campo obrigatório);
- CPF do atendente (campo obrigatório);

RN.008 - O sistema deverá validar se o funcionário está logado e se a categoria permite fazer vendas;

RN.009 - O sistema deve enviar um e-mail para o cliente com a nota fiscal da venda;

RN.010 - O sistema deverá validar o cupom de desconto dentro da data de validade;

RN.011 - O sistema deverá calcular o valor de desconto de acordo com o tempo de cadastro, criando um cupom de desconto;

RN.012 - O valor total da venda deverá ser calculado somando o valor total dos produtos subtraídos pela porcentagem de desconto;

RN.013 - O sistema deverá registrar o valor da venda no cadastro do atendente, para posterior cálculo de comissão;

RF.006 - O sistema deve permitir **criar avisos para os clientes** sobre cupons de desconto

RN.014 - O sistema de avisar o cliente que recebeu um cupom de desconto informando a data de validade do cupom;

RN.015- O gerente poderá definir um valor_desconto e tempo_desconto nas configurações do sistemas;

RF.007 - O sistema deve permitir **cadastrar a configuração do sistema**

RN.016 - O cadastro da configuração de sistemas deve ter os seguintes campos:

- Valor do desconto (campo obrigatório);
 - valor_desconto
- Tempo para ser habilitado ao desconto (campo obrigatório);
 - tempo_desconto

RN.008 - O sistema deverá emitir um **relatório de vendas** em um período definido pelo usuário.

4.2 Requisitos Não Funcionais (RNF)

- Usabilidade

RNF.001 - O Sistema deve possuir uma interface gráfica intuitiva para o funcionário;

RNF.002 - O sistema deve permitir a utilização de teclas de atalhos;

- Segurança

RNF.002 - Gerar um log de todas as transações realizadas: usuário, transação e data

RNF.003 - Acesso a informações somente administrador do sistema e dentro do IP da loja.

- Portabilidade

RNF.006 - O sistema deve ser compatível com a versão do Chrome (103) e Firefox (106.0.4).

RNF.007 - O sistema deve ser compatível com a criptografia SSL de 64 bits na transmissão de dados via internet.

RNF.008 - O sistema deve suportar até 10 funcionários simultâneos.

RNF.009 - O sistema deve persistir os dados no SGDB - MYSQL, que permita backup e restore os dados.

RNF.010 - O tamanho dos discos de dados devem permitir escalabilidade do sistema a longo prazo;

- Desempenho

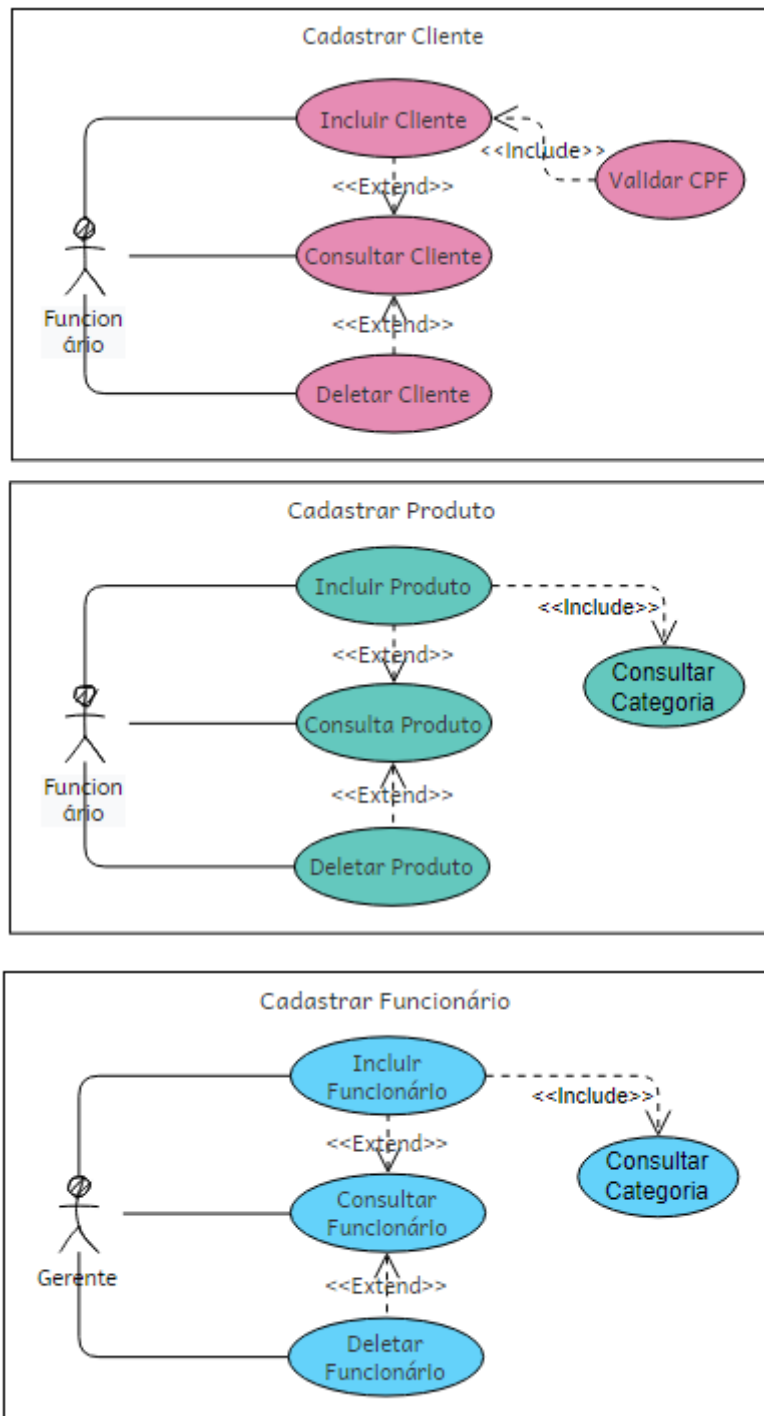
RNF.011 - O sistema deverá emitir um relatório de vendas em 5 segundos;

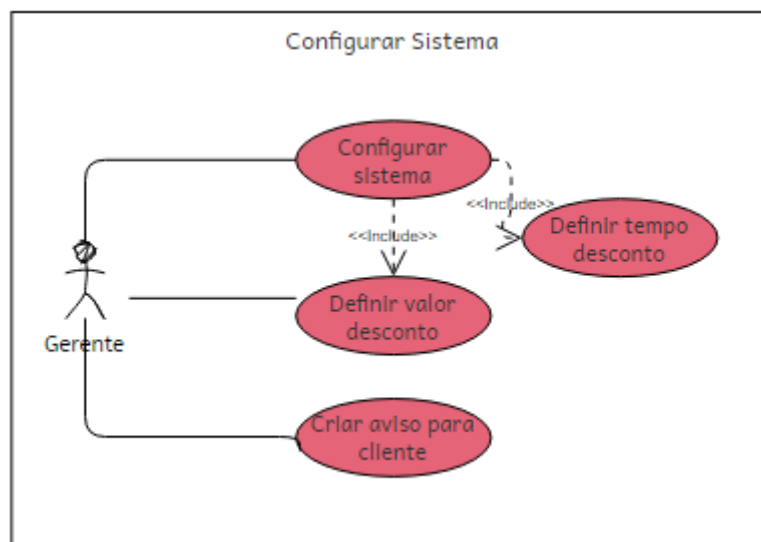
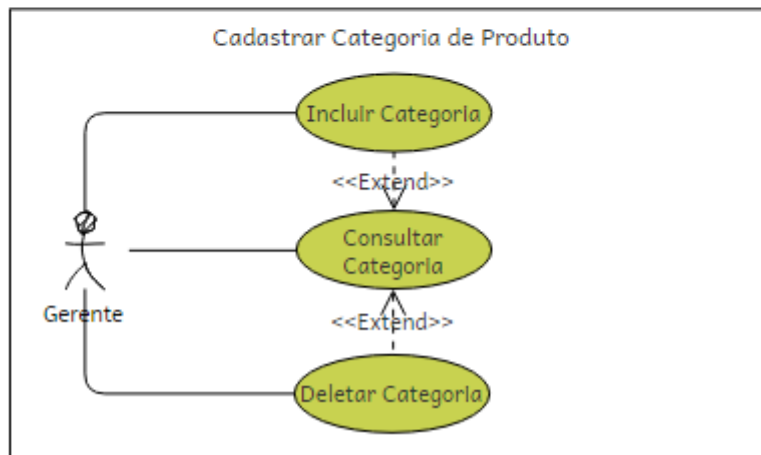
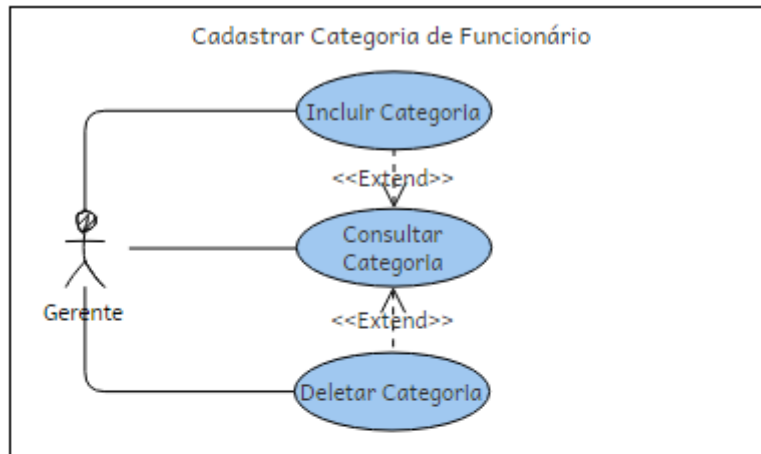
RNF.012 - O sistema deverá efetivar o cadastro em no máximo 2 segundos;

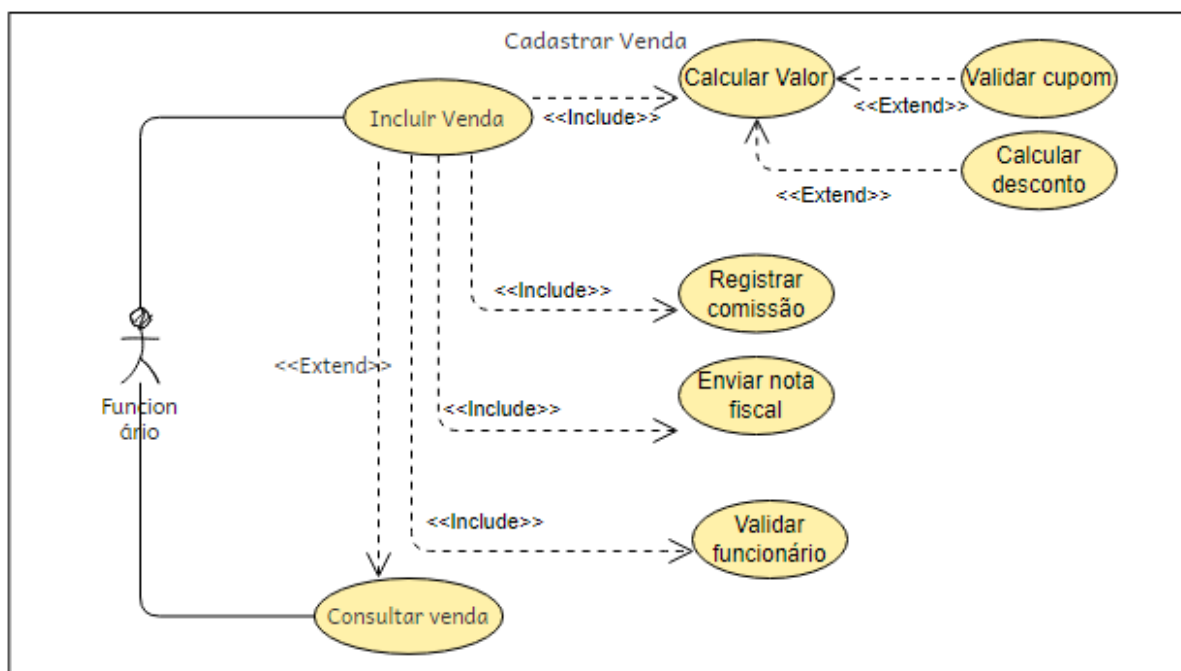
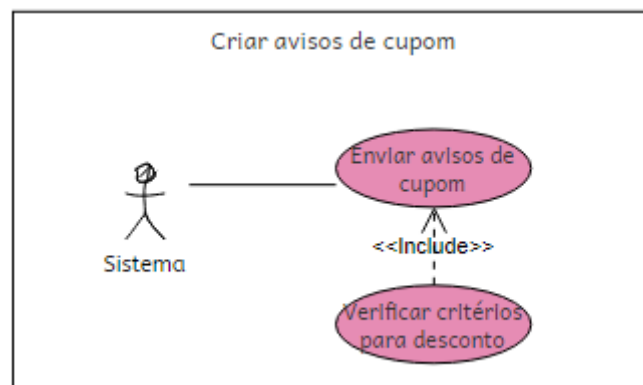
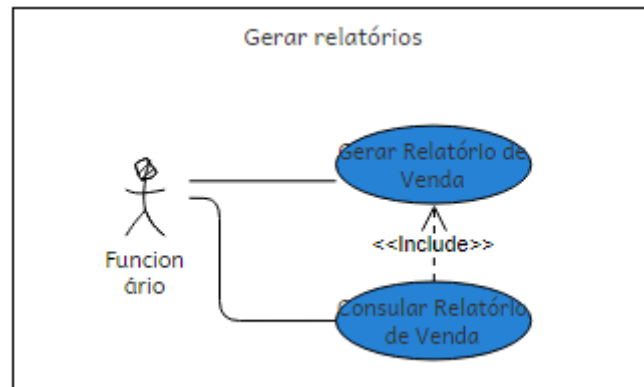
5. Diagramas de Casos de Uso

Link Online Visual Paradigm:

<https://online.visual-paradigm.com/share.jsp?id=323238313435392d34>



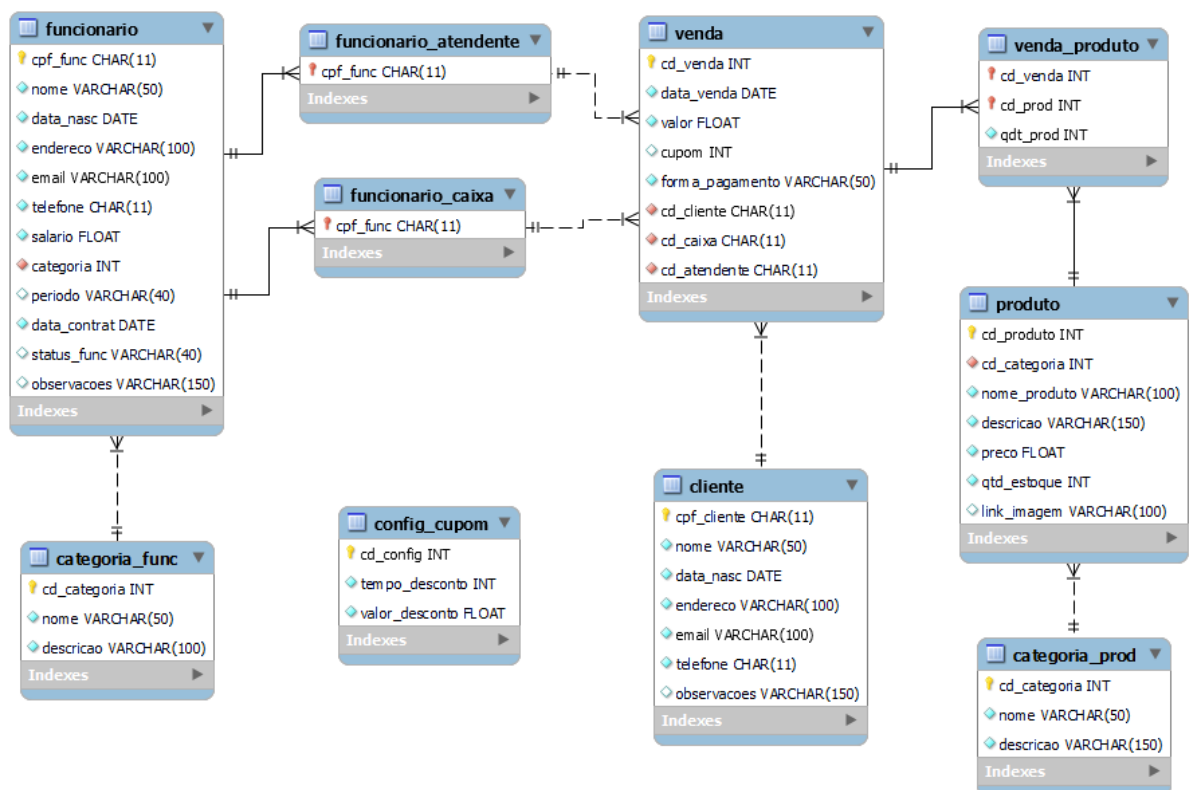




6. Diagrama de Entidade-Relacionamento (DER)

Link para diagrama em maior resolução:

https://drive.google.com/file/d/1r_fbDViY01YvD_IgXgkPGACQJ1ChlhFA/view?usp=sharing



7. Scripts SQL

Link para arquivo SQL:

<https://drive.google.com/file/d/1QI9DbYJJtjGwg9bLmIFMbBKPaJJBrRTT/view?usp=sharing>

Link para arquivo TXT:

https://drive.google.com/file/d/1nyeaww3TsfDKQDuj_GRtXeKzDLShZ0BU/view?usp=share_link

7.1 Script DDL

```
CREATE DATABASE loja;

USE loja;

-- DDL

CREATE TABLE categoria_func (
    cd_categoria INT NOT NULL UNIQUE AUTO_INCREMENT,
    nome VARCHAR(50) NOT NULL,
    descricao VARCHAR(100) NOT NULL,
    PRIMARY KEY (cd_categoria)
);

CREATE TABLE funcionario (
    cpf_func CHAR(11) NOT NULL UNIQUE,
    nome VARCHAR(50) NOT NULL,
    data_nasc DATE NOT NULL,
    endereco VARCHAR(100) NOT NULL,
    email VARCHAR(100) NOT NULL,
    telefone CHAR(11) NOT NULL,
    salario FLOAT NOT NULL,
    categoria INT NOT NULL,
    periodo VARCHAR(40),
    data_contrat DATE NOT NULL,
    status_func VARCHAR(40),
    observacoes VARCHAR(150),
    PRIMARY KEY (cpf_func),
    FOREIGN KEY (categoria) REFERENCES categoria_func (cd_categoria)
);

CREATE TABLE funcionario_caixa (
    cpf_func CHAR(11) NOT NULL UNIQUE,
    PRIMARY KEY (cpf_func),
    FOREIGN KEY (cpf_func) REFERENCES funcionario (cpf_func)
);
```

```
CREATE TABLE funcionario_atendente (  
    cpf_func CHAR(11) NOT NULL UNIQUE,  
    PRIMARY KEY (cpf_func),  
    FOREIGN KEY (cpf_func) REFERENCES funcionario (cpf_func)  
);
```

```
CREATE TABLE categoria_prod (  
    cd_categoria INT NOT NULL UNIQUE AUTO_INCREMENT,  
    nome VARCHAR(50) NOT NULL,  
    descricao VARCHAR(150) NOT NULL,  
    PRIMARY KEY (cd_categoria)  
);
```

```
CREATE TABLE produto (  
    cd_produto INT NOT NULL UNIQUE AUTO_INCREMENT,  
    cd_categoria INT NOT NULL,  
    nome_produto VARCHAR(100) NOT NULL,  
    descricao VARCHAR(300) NOT NULL,  
    preco FLOAT NOT NULL,  
    qtd_estoque INT NOT NULL,  
    link_imagem VARCHAR(300),  
    PRIMARY KEY (cd_produto),  
    FOREIGN KEY (cd_categoria) REFERENCES categoria_prod (cd_categoria)  
);
```

```
CREATE TABLE cliente (  
    cpf_cliente CHAR(11) NOT NULL UNIQUE,  
    nome VARCHAR(50) NOT NULL,  
    data_nasc DATE NOT NULL,  
    endereco VARCHAR(100) NOT NULL,  
    email VARCHAR(100) NOT NULL,  
    telefone CHAR(11) NOT NULL,  
    observacoes VARCHAR(150),  
    PRIMARY KEY (cpf_cliente)  
);
```

```
CREATE TABLE config_cupom (  
    cd_config INT NOT NULL UNIQUE AUTO_INCREMENT,  
    tempo_desconto INT NOT NULL,  
    valor_desconto FLOAT NOT NULL,  
    PRIMARY KEY (cd_config)  
);
```

```
CREATE TABLE venda (  
    cd_venda INT NOT NULL UNIQUE AUTO_INCREMENT,  
    data_venda DATE NOT NULL,  
    valor FLOAT NOT NULL,  
    cupom INT,  
    forma_pagamento VARCHAR(50) NOT NULL,  
    cd_cliente CHAR(11) NOT NULL,
```

```

        cd_caixa CHAR(11) NOT NULL,
        cd_atendente CHAR(11) NOT NULL,
        PRIMARY KEY (cd_venda),
        FOREIGN KEY (cd_cliente) REFERENCES cliente (cpf_cliente),
        FOREIGN KEY (cd_caixa) REFERENCES funcionario_caixa (cpf_func),
        FOREIGN KEY (cd_atendente) REFERENCES funcionario_atendente (cpf_func)
    );

CREATE TABLE venda_produto (
    cd_venda INT NOT NULL,
    cd_prod INT NOT NULL,
    qdt_prod INT NOT NULL,
    CONSTRAINT pk_venda_produto PRIMARY KEY (cd_venda, cd_prod),
    FOREIGN KEY (cd_venda) REFERENCES venda (cd_venda),
    FOREIGN KEY (cd_prod) REFERENCES produto (cd_produto)
);

```

Script SQL DDL - Definição da base de dados e tabelas

7.2 Script DML

```

-- DML

INSERT INTO categoria_func VALUES
(1, "Atendente", "Responsável pela recepção e atendimento direto ao cliente"),
(2, "Caixa", "Responsável pelos pagamentos"),
(3, "Gerente", "Gerente de loja"),
(4, "Serviços Gerais", "Serviços Gerais de Limpeza e Almoxarifado"),
(5, "Outro", "Outras funções");

INSERT INTO funcionario VALUES
("05972750929", "Juliana Hachmann", "1986-08-28", "Serv Ant Thiago, 510",
"juliana@gmail.com", "48998345454", 1300.50, 1, "Matutino", "2022-12-09", "Ativo", " "),
("05794792874", "Ana Elisa Zanon", "1997-07-04", "Mauro Ramos, 56",
"anaz@gmail.com", "48968344454", 1300.50, 1, "Vespertino", "2022-12-09", "Ativo", " "),
("08796512357", "Marcos Elio", "1996-12-12", "Mauro Ramos, 193",
"marcos@gmail.com", "48996652332", 2500.87, 2, "Matutino", "2022-12-10", "Ativo", " ");

INSERT INTO funcionario_caixa VALUES
("08796512357");

```

INSERT INTO funcionario_atendente VALUES

("05972750929"),
("05794792874");

INSERT INTO categoria_prod (nome, descricao) VALUES

("Basquete", "Artigos para todas as modalidades de Basquete"),
("Tênis", "Artigos para Tênis de Quadra"),
("Corrida", "Artigos para todas as modalidades de Corrida");

INSERT INTO produto VALUES

(1, 1, "Bola de Basquete Spalding Streetball", "Possuindo a qualidade Spalding em cada detalhe, a bola de basquete tem matéria prima de borracha, circunferência de 76 cm e pressão de 8 psi", 99.99, 50,
"https://imgcentauro-a.akamaihd.net/900x900/966449CD/bola-de-basquete-spalding-streetball-img.jpg"),
(2, 3, "Pochete Asics Perso com Porta Celular", "A alça é regulável e o design da pochete permite usá-la de forma confortável na hora da corrida ou caminhada", 49.99, 20,
"https://imgcentauro-a.akamaihd.net/900x900/97175502A4/pochete-asics-perso-com-porta-celular-img.jpg"),
(3, 2, "Raquete de Tênis Wilson Advantage XL L2 - Adulto", "Perfeita para você que ama jogar com os amigos em quadra e se destacar nas partidas é a Raquete de Tênis Wilson Advantage XL L2 - Adulto ", 319.99, 20,
"https://imgcentauro-a.akamaihd.net/900x900/97897006/raquete-de-tenis-wilson-advantage-xl-l2-adulto-img.jpg");

INSERT INTO cliente VALUES

("51921141441", "Thaís Santos Martins", "1953-10-24", "Serv Nunes, 520",
"thaismartins@gmail.com", "48992848088", " "),
("50520976541", "André Cardoso Ribeiro", "1991-01-12", "Av Pequeno Príncipe, 1234",
"andreriibeiro@hotmail.com", "48984213971", " "),
("85525514540", "Luana Cunha", "1982-07-30", "Av Hercílio Luz, 45",
"luanacunha@yahoo.com.br", "48988535899", " ");

INSERT INTO config_cupom VALUES

(1, 3, 0.05),
(2, 12, 0.1),
(3, 36, 0.3);

INSERT INTO venda VALUES

(1, "2022-12-10", 99.99, 0, "Dinheiro", "51921141441", "08796512357", "05972750929"),
(2, "2022-12-11", 319.99, 0, "Débito", "85525514540", "08796512357", "05794792874"),
(3, "2022-12-12", 519.96, 0, "Crédito", "50520976541", "08796512357", "05794792874");

INSERT INTO venda_produto VALUES

(1, 1, 1),
(2, 3, 1),
(3, 1, 1),
(3, 2, 2),
(3, 3, 1);

7.3 Visualizações

- Visualizando todas as compras de todos os clientes:

```
CREATE VIEW View_Compras AS SELECT cliente.nome AS Cliente, venda.data_venda  
AS Data_da_Compra, venda.valor AS Valor, venda.forma_pagamento AS  
Forma_de_Pagamento FROM cliente INNER JOIN venda ON cliente.cpf_cliente =  
venda.cd_cliente;  
  
SELECT * FROM View_Compras;
```

! Cliente	Data_da_Compra	Valor	Forma_de_Pagamento
Thaís Santos Martins	2022-12-10	99.99	Dinheiro
Luana Cunha	2022-12-11	319.99	Débito
André Cardoso Ribeiro	2022-12-12	519.96	Crédito

Captura de tela de *script* rodado na aplicação *Sql Lite Online*

- Visualizando todos os produtos e suas quantidades na venda '3':

```
CREATE VIEW Produtos_Por_Venda AS SELECT cliente.nome AS Cliente,  
venda.cd_venda AS CD_Venda, venda.data_venda AS Data, produto.nome_produto AS  
Produto, produto.preco AS Preco_Unitario, venda_produto.qdt_prod AS Quantidade  
FROM cliente INNER JOIN venda ON cliente.cpf_cliente = venda.cd_cliente INNER JOIN  
venda_produto ON venda.cd_venda = venda_produto.cd_venda INNER JOIN produto ON  
venda_produto.cd_prod = produto.cd_produto;  
  
SELECT * FROM Produtos_Por_Venda WHERE CD_Venda = 3;
```

i	Cliente	CD_Venda	Data	Produto	Preco_Unit...	Quantidade
	André Cardoso Ribeiro	3	2022-12-12	Bola de Basquete ...	99.99	1
	André Cardoso Ribeiro	3	2022-12-12	Pochete Asics Per...	49.99	2
	André Cardoso Ribeiro	3	2022-12-12	Raquete de Tênis ...	319.99	1

Captura de tela de *script* rodado na aplicação *Sql Lite Online*

- Visualizando todas as vendas do mês de determinado funcionário:

```
CREATE VIEW Vendas_Func AS SELECT funcionario.cpf_func AS CPF,
funcionario.nome AS Funcionario, venda.valor AS Valor_Venda, venda.data_venda AS
Data, venda.cd_venda AS CD_Venda, cliente.nome AS Cliente FROM funcionario INNER
JOIN venda ON funcionario.cpf_func = venda.cd_atendente INNER JOIN cliente ON
venda.cd_cliente = cliente.cpf_cliente;

SELECT * FROM Vendas_Func WHERE cpf = "05794792874" AND data >= "2022-12-01"
AND data < "2023-01-01";
```

i	CPF	Funcionario	Valor_Venda	Data	CD_Venda	Cliente
	05794...	Ana Elisa Zanon	319.99	2022-12-11	2	Luana Cunha
	05794...	Ana Elisa Zanon	519.96	2022-12-12	3	André Cardoso Ribeiro

Captura de tela de *script* rodado na aplicação *Sql Lite Online*

8. Referências

1. Quant-ux - Disponível para uso web em “<https://www.quant-ux.com/#/>”
2. Visual Paradigm - Disponível para uso web em “<https://www.visual-paradigm.com/>”
3. MYSQL - Disponível para download em “<https://www.mysql.com/>”