



MINISTÉRIO DA EDUCAÇÃO
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE MINAS GERAIS
CAMPUS BAMBUÍ
Bacharelado em Engenharia da Computação
Disciplina Banco de Dados I

Izabella Terêncio da Silva Assis

Kevenn Henrique de Paula Silva

TRABALHO PRÁTICO - ETAPA 1

Bambuí-MG

2021

Izabella Terêncio da Silva Assis
Kevenn Henrique de Paula Silva

TRABALHO PRÁTICO - ETAPA 1

Trabalho apresentado à disciplina de Banco de Dados I, do Bacharelado em Engenharia de Computação, como requisito parcial para obtenção dos créditos.

Bambuí-MG

2021

RESUMO

Nos dias atuais, cada vez mais as organizações de diferentes categorias têm a necessidade de possuir controle sistematizado de sua estrutura, com vistas a desenvolver a melhor gestão possível. Particularmente, em organizações como supermercados há a constante necessidade de controlar a entrada e saída de produtos (controle de estoque), bem como parâmetros relacionados à venda dos mesmos, e o controle manual seria altamente dispendioso, desgastante e possuiria maior probabilidade de falhas e atrasos, além de resultar na clara insatisfação dos clientes durante a realização de suas compras.

PALAVRAS-CHAVE: Supermercado; Banco de dados; *Software*; Estoque; Produto; Gerenciamento.

1 INTRODUÇÃO

O presente trabalho visa projetar um banco de dados com modelos textuais e gráficos, que ampare as atividades de gerenciamento de estoque e venda de produtos de um estabelecimento do ramo varejista, utilizando-se de técnicas de engenharia de requisitos para levantamento dos requisitos da organização e sua análise, como base para projeção e análise de modelos de banco de dados coerentes. A automatização de tais atividades de controle permite que os gestores da organização administrem de forma eficaz os recursos da organização, prevejam demandas e controlem seus processos de entrada e saída de mercadorias, possibilitando maior rapidez e assertividade na tomada de decisões, por meio de coleta, tratamento e análise de informações.

2 DESCRIÇÃO

O estabelecimento possui a necessidade de armazenar e gerenciar dados sobre seus produtos, desde sua chegada por meio dos fornecedores, até sua saída por meio de vendas aos clientes. Cada fornecedor recebe uma ordem de compra que contém produtos a serem fornecidos e pode entregar um ou mais lotes contendo os produtos solicitados. Também são emitidas notas fiscais de entrada, a respeito dos produtos que foram entregues. Cada produto pertence a uma categoria e em cada venda podem ser vendidos um ou mais produtos. Para estas, é necessário conhecer o meio de pagamento utilizado pelo cliente, podendo este ser cartão, dinheiro ou crediário. Para o caso em que o cliente utilize pagamento por crediário, é necessário conhecer o número da conta do mesmo.

2.1 REGRAS DE NEGÓCIO

As regras de negócio de uma organização podem ser entendidas como convenções e normas necessárias à estruturação e ao funcionamento da mesma, garantido sua existência, e caracterizando uma base para a modelagem de requisitos de um sistema. Elas são criadas com o objetivo de normatizar e controlar a estrutura da empresa, visando seu funcionamento de forma adequada segundo os padrões estabelecidos. Abaixo, são descritas algumas regras de negócio necessárias para o entendimento e desenvolvimento dos modelos desenvolvidos mais à frente.

- **Funcionários (RN01):** O cadastro de um funcionário é realizado após a comprovação da assinatura de sua carteira de trabalho.
- **Promoções (RN02):** Promoções não podem ser realizadas caso o estoque de determinado produto estiver abaixo de 15% do total.
- **Promoções (RN03):** A um mesmo produto não pode ser aplicada mais de uma promoção de forma simultânea. Promoções resultam em descontos no preço unitário dos produtos.
- **Passagem pelo caixa registrador (RN04):** Todo produto do estabelecimento deve passar pelo caixa registrador para que a compra seja efetuada.
- **Reposição de estoque (RN05):** O estoque dos produtos deve ser atualizado sempre que um novo carregamento chegar.
- **Organização de mostruário (RN06):** Cada gôndola possui produtos de apenas uma categoria (alimentício, limpeza, higiene, etc.). Ou seja, os produtos devem ser distribuídos no mostruário levando-se em conta setores específicos.
- **Prazo de validade (RN07):** Os produtos que possuírem prazo de validade vencido devem ser retirados do mostruário e desconsiderados do estoque.
- **Vendas (RN08):** A compra de um produto por um cliente somente é concluída após a realização do pagamento do mesmo.
- **Clientes (RN09):** Os clientes possuem cadastro no estabelecimento somente para fins de uso de crediário (conta), o qual é concedido após uma análise de crédito.
- **Crediário (RN10):** Cada conta possui apenas um titular e cada cliente pode ter apenas uma conta no estabelecimento, a qual possui prazo de vencimento fixo de 40 dias corridos a partir da data da compra.

2.2 REQUISITOS

Segundo Sommerville (2011, p.59), os requisitos funcionais de um sistema são responsáveis por descrever o que tal sistema deve fazer, sendo declarações do que o sistema deve fornecer e de como o mesmo reage e se comporta em determinadas situações. Tais requisitos variam de acordo com o sistema a ser desenvolvido, o público-alvo de usuários e diversas outras condições, e podem ser expressos de maneiras diferentes de acordo com o público leitor ao qual são principalmente orientados. Aqui serão descritos de forma relativamente breve alguns requisitos, que servirão de base ao entendimento e à contextualização dos modelos desenvolvidos para o banco de dados, mais adiante.

- **RF01** - O sistema deve permitir o cadastro de dados pessoais dos clientes com a possibilidade de consulta, exclusão e alteração dos seguintes dados: nome, CPF, endereço e número de telefone, bem como um código de conta único.
- **RF03** - O sistema deve permitir o cadastro de dados de produtos do estoque com a possibilidade de consulta, exclusão e alteração dos seguintes dados: descrição (contendo nome e marca), código, preço unitário, quantidade em estoque, peso, *status* (ativo ou inativo) e desconto.
- **RF04** - O sistema deve permitir o cadastro de dados de fornecedores com a possibilidade de consulta, exclusão e alteração dos seguintes dados: razão social (nome), endereço, endereço de *e-mail*, telefone, CNPJ, CEP, país e *status* (ativo ou inativo).
- **RF05** - O sistema deve possibilitar que funcionários autorizados (por meio de autenticação com nome de usuário e senha) gerenciem e atualizem os dados cadastrados.

De forma semelhante aos requisitos funcionais acima citados, tem-se também os chamados requisitos não funcionais, que de acordo com Sommerville (2011, p.60) não se relacionam de forma direta com os serviços ofertados pelo sistema, mas são ligados a questões de implementação do mesmo. São pontos relacionados a questões de segurança, portabilidade, manutenibilidade e compatibilidade, por exemplo, que são aqui apresentados para melhor compreensão do sistema e de suas limitações:

- **RNF01** - O acesso ao sistema será restrito ao administrador e aos funcionários devidamente credenciados.
- **RNF02** - O acesso ao sistema é realizado por meio da validação do usuário, informando nome de usuário e senha.
- **RNF03** - O usuário administrador possui acesso completo a todas as funcionalidades do sistema, podendo alterar, excluir e adicionar novos cadastros.

3 PROJETOS

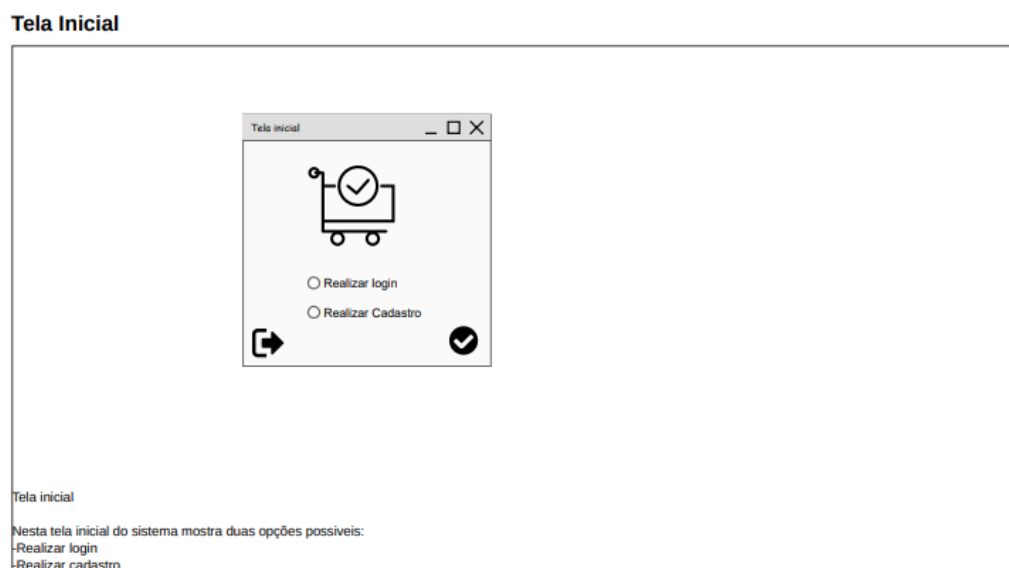
Para o desenvolvimento do banco de dados é necessária uma base lógica caracterizada pela modelagem do mesmo utilizando-se de recursos textuais e recursos gráficos, aplicados na tarefa de descrever as entidades participantes do sistema e os relacionamentos estabelecidos entre tais entidades, bem como as características de cada um (seus atributos), com o objetivo de fornecer uma visualização geral e informações suficientes para o entendimento do sistema e como base para as etapas posteriores do desenvolvimento. Primeiro será apresentado o projeto de interfaces, que possibilitará uma perspectiva inicial sobre a interface de interação com o usuário do sistema. Em seguida, são trazidos os projetos conceitual e lógico, que são os dois níveis de abstração de bancos de dados normalmente considerados.

3.1 PROJETO DE INTERFACES

Nesta etapa é apresentado o modelo de telas de interação com o usuário administrador, responsável por gerenciar os dados do banco. Seu intuito é oferecer aprendizado suave ao usuário, com ícones intuitivos e uso de *design* minimalista para fornecer a informação e as funcionalidades de forma clara ao usuário, atendendo às suas necessidades, bem como às do sistema¹.

Este projeto foi construído com o auxílio do *software* Pencil², e foi inicialmente pensado para proporcionar praticidade no gerenciamento de dados da organização.

Figura 1 - Tela Inicial



Fonte: elaborado pelos autores, 2021.

¹ O arquivo produzido contendo o projeto de interfaces pode ser lido na íntegra e em melhor qualidade a partir da URL <<https://drive.google.com/drive/folders/1tiyO2nIUf8plFhOnvz9zW3PQ-W55yJ>>.

² Informações e *download* do mesmo podem ser encontrados em <<https://pencil.evolus.vn/>>.

Figura 2 - Tela de *login*

Login



The image shows a web browser window titled "Login". On the left is a user icon. To its right are two radio buttons: "Realizar login" (selected) and "Realizar cadastro". Below these is a form titled "Insira os dados de usuário" containing two input fields: "NOME DE USUÁRIO:" and "SENHA:". At the bottom left is a back arrow icon, and at the bottom right is a checkmark icon.

Login

Após escolher a opção "Realizar login", o sistema é redirecionado para esta tela, onde é possível efetuar o login.

Fonte: elaborado pelos autores, 2021.

Figura 3 - Tela de cadastro de usuário

Cadastro



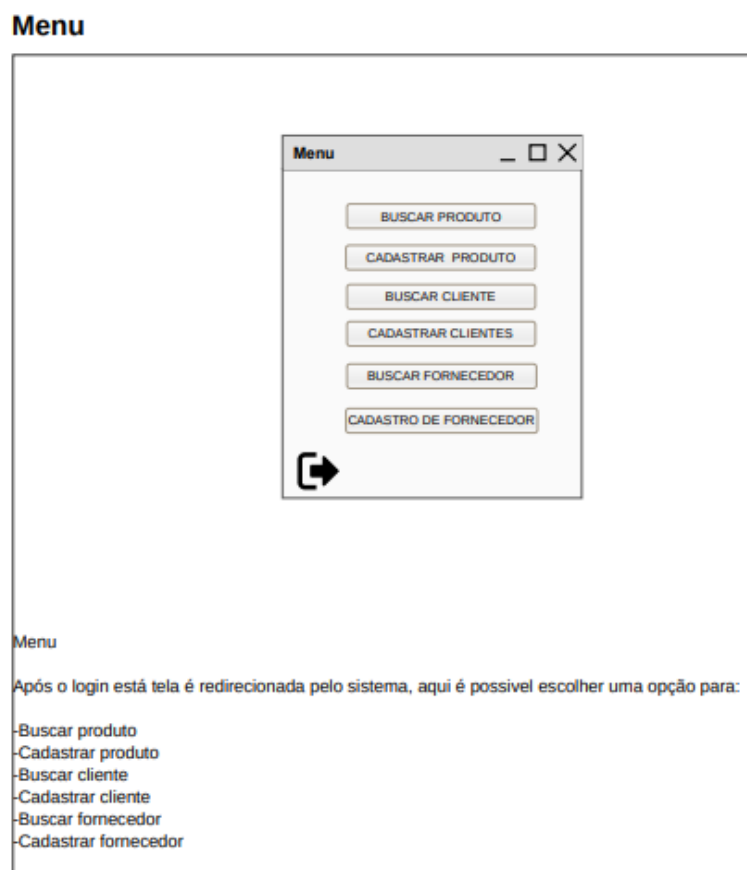
The image shows a web browser window titled "Cadastro". On the left is a user icon. To its right are two radio buttons: "Realizar login" and "Realizar Cadastro" (selected). Below these is a form titled "Insira os dados de usuário" containing four input fields: "NOME DE USUÁRIO:", "E-MAIL:", "SENHA:", and "CONFIRMAR SENHA:". At the bottom left is a back arrow icon, and at the bottom right is a checkmark icon.

Cadastro

Após escolher a opção "Realizar cadastro", esta tela é redirecionada, onde é possível efetuar o cadastro.

Fonte: elaborado pelos autores, 2021.

Figura 4 - Tela de menu



Fonte: elaborado pelos autores, 2021.

Figura 5 - Tela de busca de produto



Fonte: elaborado pelos autores, 2021.

Figura 6 - Tela de cadastro de fornecedores

Cadastrar Fornecedor

The screenshot shows a window titled 'Cadastro de fornecedor' with a standard Windows title bar (minimize, maximize, close buttons). Inside the window, there is a sub-header 'CADASTRO DE DADOS DE UM FORNECEDOR' accompanied by a small printer icon. Below this is a form titled 'Dados do fornecedor' containing eight input fields, each with a label to its left: 'NOME:', 'E-MAIL:', 'ENDEREÇO:', 'TELEFONE:', 'CNPJ:', 'CEP:', 'STATUS (ativo ou inativo):', and 'NACIONALIDADE:'. At the bottom left of the form area is a 'Back' button (left arrow), and at the bottom right is a 'Submit' button (checkmark).

Cadastrar fornecedor

Após escolher a opção de "Cadastrar fornecedor", esta tela do sistema será carregada, aqui é possível cadastrar os seguintes campos:

- nome
- email
- endereço
- telefone
- CNPJ
- CEP
- status (ativo ou inativo)
- nacionalidade

Fonte: elaborado pelos autores, 2021.

Figura 7 - Tela de busca de fornecedores

Buscar Fornecedor

Cadastro de Fornecedores

BUSCAR FORNECEDOR

CNPJ / Razão social

BUSCAR

CNPJ	Razão social	Endereço	CEP
------	--------------	----------	-----

Vertical toolbar icons: +, trash, save, edit, right arrow, left arrow.

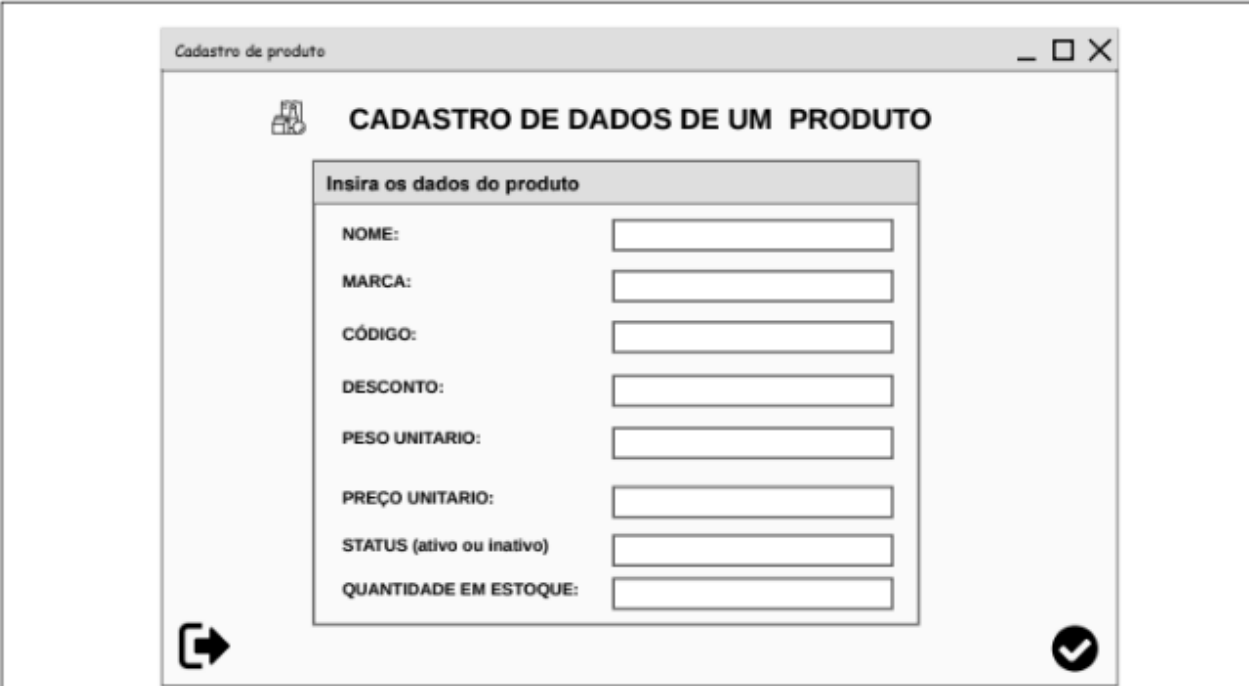
Buscar Fornecedor

Após escolher a opção de "Buscar fornecedor" anteriormente, esta tela do sistema será carregada, aqui é possível buscar um fornecedor e excluí-lo. Também há uma opção onde, ao selecionar um fornecedor, será redirecionado a uma nova tela onde será possível atualizar os seus dados cadastrados.

Fonte: elaborado pelos autores, 2021.

Figura 8 - Tela onde é possível realizar o cadastro um produto

Cadastrar Produto



The screenshot shows a window titled "Cadastro de produto" with a standard Windows-style title bar (minimize, maximize, close buttons). Inside the window, there is a sub-header "CADASTRO DE DADOS DE UM PRODUTO" next to a small icon of a product box. Below this is a section titled "Insira os dados do produto" which contains a form with eight input fields, each preceded by a label: "NOME:", "MARCA:", "CÓDIGO:", "DESCONTO:", "PESO UNITARIO:", "PREÇO UNITARIO:", "STATUS (ativo ou inativo)", and "QUANTIDADE EM ESTOQUE:". At the bottom left of the form area is a back arrow icon, and at the bottom right is a checkmark icon. Below the window screenshot, the text "Cadastrar produto" is followed by a paragraph explaining that this screen is used to register product data, and then a list of the fields to be registered.

Cadastrar produto

Após escolher a opção de "Cadastrar produto", está tela do sistema será carregada, aqui é possível cadastrar os seguintes campos:


- nome
- marca
- código
- desconto
- peso unitario
- preço unitario
- status(ativo ou inativo)
- quantidade em estoque

Fonte: elaborado pelos autores, 2021.

Figura 9 - Tela de atualização do cadastro de fornecedores

Atualizar fornecedor

Atualizar fornecedor



ATUALIZAR DADOS DE UM FORNECEDOR

Dados do fornecedor

NOME:

E-MAIL:

ENDEREÇO:


TELEFONE:


CNPJ:

CEP:

STATUS (ativo ou inativo):

NACIONALIDADE:





Atualizar fornecedor

Após selecionar um fornecedor, esta tela será redirecionada pelo sistema, onde é possível atualizar os dados do fornecedor já cadastrado.

Fonte: elaborado pelos autores, 2021.

Figura 10 - Tela de cadastro de clientes

Cadastrar Cliente



The screenshot shows a window titled "Cadastro de cliente" with a standard Windows-style title bar (minimize, maximize, close buttons). Inside the window, there is a header area with a person icon and the text "CADASTRO/ATUALIZAÇÃO DE DADOS DE UM CLIENTE". Below this is a form titled "Dados do cliente" which contains five input fields: "NOME:", "CPF:", "ENDEREÇO:", "Nº TELEFONE:", and "CÓDIGO:". Each field has a corresponding text label and an empty input box. At the bottom left of the form area is a "Go" button (a square with a right-pointing arrow), and at the bottom right is a "Confirm" button (a circle with a checkmark).

Cadastrar cliente

Após escolher a opção de "Cadastrar cliente", esta tela do sistema será carregada, aqui é possível cadastrar os seguintes campos:

- nome
- CPF
- endereço
- Nº telefone
- código

Fonte: elaborado pelos autores, 2021.

Figura 11 - Tela de busca de clientes

Buscar Cliente

Cadastro de Clientes

BUSCAR DE CLIENTE

Nome / Código da conta

BUSCAR

Conta	Nome	Endereço	Telefone

➡

⊕
🗑️
💾
✎
➡
⬅️

Buscar cliente

Após escolher a opção de "Buscar cliente" anteriormente, esta tela do sistema será carregada, aqui é possível buscar um cliente e excluí-lo. Também há uma opção onde, ao selecionar um cliente, será redirecionado a uma nova tela onde será possível atualizar os seus dados cadastrados.

Fonte: elaborado pelos autores, 2021.

Figura 12 - Tela de cadastro de vendas

Venda

The screenshot shows a window titled 'Venda' with a standard Windows-style title bar (minimize, maximize, close buttons). Inside the window, there is a header area with a small icon of a cash register and the text 'VENDA DE UM PRODUTO'. Below this is a section titled 'Dados da venda' which contains five input fields: 'CÓDIGO CAIXA:', 'NÚMERO', 'VALOR TOTAL:', 'DATA:', and 'HORA:'. Each field has a corresponding text input box. At the bottom left of the window is a back arrow icon, and at the bottom right is a checkmark icon. Below the window, there is a small text label 'Venda' and a paragraph: 'Nesta tela do sistema é possível verificar os dados da venda.'

Fonte: elaborado pelos autores, 2021.

Figura 13 - Tela de registro dos produtos já vendidos

Tabela dos produtos vendidos

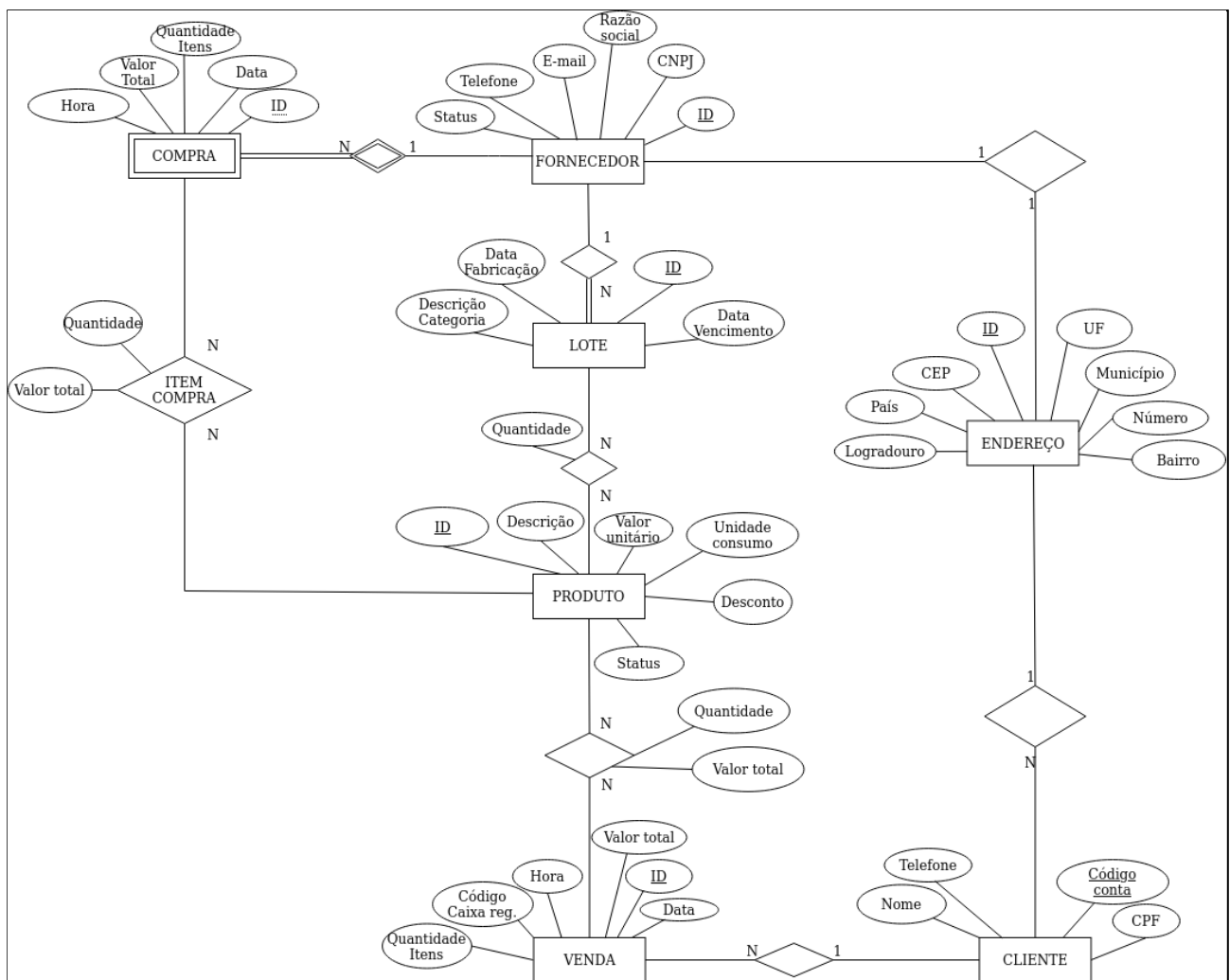
The screenshot shows a window titled 'Tabela de produtos vendidos' with a standard Windows-style title bar. Inside the window, there is a header area with a small icon of a cash register and the text 'PRODUTOS VENDIDOS'. Below this is a search section with a text input field labeled 'Código / Nome' and a button labeled 'BUSCAR'. Below the search section is a table with four columns: 'Código', 'Nome', 'Preço unitario', and 'Quantidade'. The table is currently empty. To the right of the table is a vertical toolbar with six icons: a plus sign, a trash can, a document, a pencil, a right arrow, and a left arrow. At the bottom left of the window is a back arrow icon. Below the window, there is a small text label 'Tabela de produtos vendidos' and a paragraph: 'Nessa tela é possível verificar todos os produtos que foram vendidos.'

Fonte: elaborado pelos autores, 2021.

3.2 PROJETO CONCEITUAL

A descrição que é desenvolvida por este modelo possui alto nível de abstração e não busca desenvolver quaisquer detalhes a respeito de quaisquer Sistemas Gerenciadores de Banco de Dados. De acordo com Elmasri e Navathe (2005, p.90), no modelo conceitual são identificados os dados que podem aparecer no banco de dados, mas não há preocupação em detalhar o armazenamento de tais dados a nível de SGBD. A técnica aqui utilizada abrangida pelo modelo relacional é a abordagem entidade-relacionamento, como pode ser visto abaixo³.

Figura 14 - Diagrama Entidade-Relacionamento



Fonte: elaborado pelos autores, 2021.

³ Arquivo original em melhor qualidade pode ser encontrado em
<<https://app.diagrams.net/#G1BkCFgUpTZcHjInzOk2vhYO8TEvy8Wwhw>>.

3.3 PROJETO LÓGICO

Após a confecção do modelo conceitual na seção anterior, foi possível realizar seu mapeamento para um modelo mais próximo do esquema físico de um banco de dados do que o anterior, denominado modelo lógico. Em um primeiro momento, foi realizado um mapeamento para uma representação textual, o qual foi incluído no texto aqui desenvolvido pois possibilitou maior facilidade na construção da representação gráfica posterior.

```
fornecedor(+id_fornecedor INTEGER, +cnpj CHAR(14), +razao_social VARCHAR(100),
          email VARCHAR(65), +status CHAR(1), telefone VARCHAR(15), *+id_endereco
          INTEGER)
          *fornecedor.id_endereco: endereco.id_endereco
```

```
endereco(+id_endereco INTEGER, +pais VARCHAR(80), +cep VARCHAR(20), +uf CHAR(2),
          +municipio VARCHAR(80), +logradouro VARCHAR(80), +numero INTEGER,
          +bairro VARCHAR(80))
```

```
lote(+id_lote INTEGER, +data_fabricacao DATE, +data_vencimento DATE,
      *+id_fornecedor INTEGER, +descricao_categoria VARCHAR(100))
      *lote.id_fornecedor: fornecedor.id_fornecedor
```

```
compra(+id_compra INTEGER, *+id_fornecedor INTEGER, +data DATE, +hora TIME
        +quantidade_produtos INTEGER, +valor_total REAL)
        *compra.id_fornecedor: fornecedor.id_fornecedor
```

```
item_compra(*+id_compra INTEGER, *+id_fornecedor INTEGER, *+id_produto INTEGER,
             +quantidade_item REAL, +valor_total_item REAL)
             *item_compra.id_compra: compra.id_compra
             *item_compra.id_fornecedor: compra.id_fornecedor
             *item_compra.id_produto: produto.id_produto
```

```
produto(+id_produto INTEGER, +quantidade REAL, +descricao VARCHAR(255),
          +valor_unitario REAL, desconto REAL, +status CHAR(1), +unidade_consumo
          VARCHAR(5))
```

```
lote_produto(*+id_lote INTEGER, *+id_produto INTEGER, +quantidade_produtos
              INTEGER)
              *lote_produto.codigo_lote: lote.codigo_lote
              *lote_produto.codigo_produto: produto.codigo_produto
```

```

produto_venda(*+id_produto INTEGER, *+id_venda INTEGER, +quantidade INTEGER,
              +valor_total REAL)
*produto_venda.id_produto: produto.id_produto
*produto_venda.id_venda: venda.id_venda

cliente(+codigo_conta INTEGER, +cpf CHAR(11), +telefone VARCHAR(15), +nome
        VARCHAR(80), *+id_endereco INTEGER)
*cliente.id_endereco: endereco.id

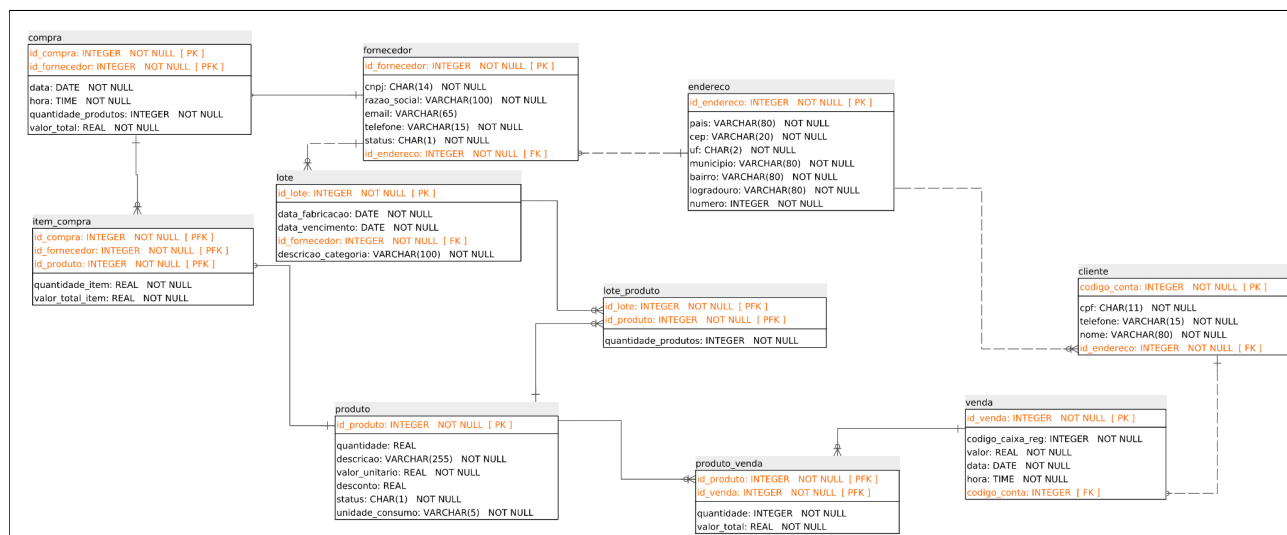
venda(+codigo_caixa_reg INTEGER, +id_venda INTEGER, +valor REAL, +data DATE, +hora
      TIME, *codigo_conta INTEGER)
*venda.codigo_conta: cliente.codigo_conta

```

3.4 PROJETO LÓGICO GRÁFICO

Nesta etapa, o esquema da seção anterior, com suas entidades e relacionamentos transcritos para estruturas mais elaboradas, foi transformado em uma representação lógica gráfica, que traz maior facilidade da identificação dos elementos do banco de dados⁴, por meio do uso do *software* SQL Power Architect Community Edition⁵.

Figura 15 - Representação gráfica do modelo lógico do banco de dados



Fonte: elaborado pelos autores, 2021.

⁴Arquivos originais nos formatos .pdf e .architect podem ser encontrados na pasta
<https://drive.google.com/drive/folders/1dDGU4Y3IEI3KwLFM1HFLHZNo0GRQ116R?usp=sharing>.

⁵Informações e download podem ser encontrados na página oficial
http://www.bestofbi.com/page/architect_download_os.

3.5 CONSULTAS SQL

Para a estruturação do banco de dados de fato, foi elaborado um *script* SQL contendo as instruções necessárias para a criação do banco de dados, bem como de seus componentes (tabelas e respectivas colunas), o qual encontra-se no arquivo `supermercado.sql`.

Ainda, foram elaboradas consultas sobre os dados inseridos no banco de dados, as quais seguem abaixo com suas respectivas descrições:

1. Consulta para obter o valor total comprado de cada fornecedor:

```
SELECT f.id_fornecedor AS codigo,
       f.razao_social AS fornecedor,
       SUM(c1.valor_total) AS valor_comprado
FROM   fornecedor AS f,
       compra AS c1, compra AS c2
WHERE  c1.id_fornecedor = f.id_fornecedor
       AND c1.id_compra = c2.id_compra
GROUP BY codigo, fornecedor;
```

2. Listagem dos dez produtos com maior movimentação, considerando entrada (compras) e saída (vendas) dos mesmos:

```
SELECT descricao AS produto,
       qtd_saida + qtd_entrada AS total_movimentacao
FROM (
    SELECT p.id_produto AS idp1,
           SUM(vp.quantidade) AS qtd_saida
    FROM   produto AS p
    JOIN   venda_produto AS vp
          ON p.id_produto = vp.id_produto
    GROUP BY idp1
) d,
(
    SELECT p.id_produto AS idp2,
           SUM(cp.quantidade) AS qtd_entrada
    FROM   produto AS p
    JOIN   compra_produto AS cp
          ON p.id_produto = cp.id_produto
    GROUP BY idp2
) e, produto AS p
WHERE  p.id_produto = idp1
       AND p.id_produto = idp2
ORDER BY total_movimentacao DESC
LIMIT 10;
```

3. Consulta para listagem do total de vendas geral, agrupado por mês:

```
SELECT DISTINCT date_part('month', v1.data) AS mes_venda,
       SUM(v1.valor_total) AS valor_total_vendas
FROM   venda AS v1, venda AS v2, venda AS v3
WHERE  v1.id_venda = v2.id_venda
       AND v1.id_venda = v3.id_venda
       AND v2.id_venda = v3.id_venda
```

```
GROUP BY mes_venda
ORDER BY mes_venda;
```

4. Consulta para a atualização da coluna de desconto dos produtos do gênero alimentício que estejam no mês de vencimento com valor de 20% (0.20) - para possíveis posteriores cálculos de redução dos preços:

```
UPDATE produto AS p
SET desconto = 0.20
FROM lote AS l,
     lote_produto AS lp
WHERE lp.id_lote = l.id_lote
     AND lp.id_produto = p.id_produto
     AND extract(year from l.data_vencimento) = extract(year from current_date)
     AND extract(month from l.data_vencimento) = extract(month from current_date)
     AND l.descricao_categoria = 'ALIMENTICIO';
```

5. Consulta para listagem de produtos comprados no ano de 2021:

```
SELECT DISTINCT c.data AS data_compra,
                p.id_produto AS codigo,
                p.descricao AS produto,
                p.unidade_consumo AS unidade,
                c.quantidade AS quantidade_produtos,
                p.valor_unitario AS valor
FROM produto AS p,
     compra AS c,
     compra_produto AS cp
WHERE c.data BETWEEN '2021-01-01' AND '2021-12-31'
     AND cp.id_compra = c.id_compra
     AND cp.id_produto = p.id_produto
ORDER BY codigo;
```

6. Consulta para obtenção do total comprado de cada cidade:

```
SELECT DISTINCT e1.municipio,
                SUM(d.valor_comprado) AS total_vendas
FROM endereco AS e1,
     endereco AS e2,
     fornecedor AS f,
     (
         SELECT f.id_fornecedor AS codigo,
                SUM(c1.valor_total) AS valor_comprado
         FROM fornecedor AS f,
              compra AS c1, compra AS c2
         WHERE c1.id_fornecedor = f.id_fornecedor
              AND c1.id_compra = c2.id_compra
         GROUP BY codigo
     ) AS d
WHERE d.codigo = f.id_fornecedor
     AND f.id_endereco = e1.id_endereco
     AND e1.id_endereco = e2.id_endereco
GROUP BY e1.municipio;
```

7. Consulta para listagem do total comprado e do total vendido de cada produto:

```
SELECT DISTINCT p.id_produto AS codigo_produto,
                p.descricao AS produto,
                d.total_vendido,
                e.total_comprado
FROM (
    SELECT p.id_produto AS codigo,
           SUM(vp.quantidade) AS total_vendido
    FROM produto AS p
    JOIN venda_produto AS vp
        ON p.id_produto = vp.id_produto
    JOIN venda AS v
        ON v.id_venda = vp.id_venda
    GROUP BY codigo
) d,
(
    SELECT p.id_produto AS codigo,
           SUM(cp.quantidade) AS total_comprado
    FROM produto AS p
    JOIN compra_produto AS cp
        ON p.id_produto = cp.id_produto
    JOIN compra AS c
        ON c.id_compra = cp.id_compra
    GROUP BY codigo
) e,
produto AS p
WHERE d.codigo = e.codigo
      AND p.id_produto = d.codigo
      AND p.id_produto = e.codigo
ORDER BY codigo_produto;
```

8. Consulta para listar em que mês e ano um produto teve uma maior quantidade de vendas:

```
WITH pq AS (
    SELECT
        vp.id_produto, EXTRACT ( year FROM v.data) AS ano,
        EXTRACT (month from v.data) AS mes,
        SUM(vp.quantidade) AS quantidade
    FROM
        venda AS v,
        venda_produto AS vp
    where
        v.id_venda = vp.id_venda
    GROUP BY
        vp.id_produto,
        EXTRACT (YEAR FROM v.data),
        EXTRACT (MONTH FROM V.DATA)
)
, pqm AS (
    SELECT
        id_produto,
        MAX(quantidade) AS quantidade
    FROM
        pq
    GROUP BY
        id_produto
)
SELECT
```

```

    p.descricao AS produto,
    pq.ano,
    pq.mes,
    pq.quantidade
FROM
    produto AS p,
    pq,
    pqm
WHERE
    p.id_produto = pq.id_produto
    AND p.id_produto = pqm.id_produto
    AND pq.quantidade = pqm.quantidade
ORDER BY
    p.descricao;

```

9. Consulta para listar o total comprado por cada cliente:

```

select
    c.nome,
    SUM(v.valor_total) as total
from
    cliente as c,
    venda as v
where
    c.codigo_conta = v.id_venda
group by
    c.nome;

```

10. Consulta para listar todos os produtos que não foram vendidos, com seus devidos nomes e IDs:

```

with pvnc as (
    select
        id_produto
    from
        produto
    except
    select
        vp.id_produto
    from
        venda as v,
        venda_produto as vp
    where
        v.id_venda = vp.id_venda
)
select
    p.descricao as nome,
    pvnc.id_produto
from
    produto as p,
    pvnc
where
    p.id_produto = pvnc.id_produto;

```

REFERÊNCIAS

ELMASRI, Ramez; NAVATHE, Shamkant B. **Sistemas de banco de dados**. São Paulo: Pearson Addison Wesley, 2005.

SOMMERVILLE, Ian. **Engenharia de *Software***. 9. ed. São Paulo: Pearson Prentice Hall, 2011.