

UNIVERSIDADE DO OESTE DE SANTA CATARINA – UNOESC

CAMPUS SÃO MIGUEL DO OESTE

CIÊNCIA DA COMPUTAÇÃO – 3ª FASE

ARTHUR BORGER KOCHER

ARTHUR COSTA GRUBER

BRUNO GABRIEL KONZEN

SISTEMA PARA UMA PADARIA: BREADTRACK

SÃO MIGUEL DO OESTE

2023

ARTHUR BORGER KOCHER

ARTHUR COSTA GRUBER

BRUNO GABRIEL KONZEN

SISTEMA PARA UMA PADARIA: BREADTRACK

Trabalho apresentado ao curso de Graduação em Ciência da Computação da Universidade do Oeste de Santa Catarina - UNOESC, Campus São Miguel do Oeste, como requisito para obtenção de nota para o componente curricular de Banco de Dados I.

Orientador: Roberson Junior Fernandes Alves

SÃO MIGUEL DO OESTE

2023

RESUMO

Este trabalho apresenta um estudo de caso sobre a implementação de um Sistema de Gerenciamento de Banco de Dados (SGBD) em uma padaria, com o objetivo de aprimorar o gerenciamento de informações e processos relacionados ao estabelecimento. O SGBD escolhido para esse projeto foi o PostgreSQL devido à sua ampla adoção, eficiência e escalabilidade. Inicialmente, foi realizada uma análise das necessidades da padaria, identificando os principais requisitos para o SGBD. Os requisitos incluíam o armazenamento e a organização de dados de clientes, produtos, estoque, vendas, pedidos de fornecedores e funcionários. A modelagem do banco de dados foi realizada utilizando-se a abordagem relacional, onde foram definidas as entidades, atributos e relacionamentos relevantes para a padaria. Com o banco de dados modelado, procedeu-se à implementação do SGBD, criando as tabelas, estabelecendo as chaves primárias e estrangeiras, e populando o banco com os dados existentes na padaria. Ao longo da implementação, foram realizados testes para verificar a eficácia do SGBD. Os resultados demonstraram que o PostgreSQL atendeu satisfatoriamente às necessidades da padaria, proporcionando um desempenho adequado, consultas eficientes e segurança dos dados. Este trabalho demonstrou a importância e os benefícios de um SGBD na gestão de uma padaria. Recomenda-se que outras padarias considerem a adoção de um SGBD adequado às suas necessidades, a fim de obterem maior eficiência e competitividade no mercado.

Palavras-chave: Sistema de Gerenciamento de Banco de Dados (SGBD); Padaria; Implementação; PostgreSQL; Eficiência.

SUMÁRIO

1	INTRODUÇÃO.....	5
1.1	Justificativa.....	6
1.2	Definição do Problema	6
1.3	Objetivo Geral	6
1.4	Objetivos Específicos	7
2	METODOLOGIA	8
3	DESENVOLVIMENTO	10
3.1	Requisitos (Funcionais e Não-Funcionais).....	10
3.2	Modelagem Conceitual.....	12
3.3	Modelagem Relacional.....	14
3.3.1	Dicionário de Dados (Data Dictionary).....	16
3.4	Banco de Dados.....	19
3.4.1	Scripts da criação da base de dados.....	19
3.4.2	Relatórios previsto pelo modelo de negócio.	21
3.4.3	Implementação das regras de integridade.....	23
4	ANEXOS.....	24
5	CONCLUSÃO	25
6	REFERÊNCIAS	27

ÍNDICE DE FIGURAS

Figura 1 - Diagrama Entidade Relacionamento.....	13
Figura 2 - Modelo Relacional.....	15
Figura 3 - Dicionário de Dados.....	17
Figura 4 - Dicionário de Dados (2).	18
Figura 5 - Dicionário de Dados (3).	19
Figura 6 - Exemplo da criação da tabela Cliente.....	20
Figura 7 - Exemplo de criação de dados na tabela Cliente.	21
Figura 8 - Comandos SELECT (Resposta 1 e 2).	22
Figura 9 - Comandos SELECT (Resposta 3 e 4).	22
Figura 10 - Regras de Integridade.....	23

1 INTRODUÇÃO

No mundo contemporâneo, as empresas estão cada vez mais dependentes de sistemas de gerenciamento de banco de dados para otimizar suas operações e garantir a eficiência de suas atividades. Nesse contexto, o setor de alimentos, mais especificamente as padarias, também têm se beneficiado significativamente da adoção de sistemas modernos para o gerenciamento de suas informações.

Este trabalho acadêmico tem como objetivo apresentar um estudo sobre o desenvolvimento e implementação de um sistema gerenciador de banco de dados para uma padaria, utilizando ferramentas como dBeaver, Visual Paradigm e o brModelo, o PostgreSQL como SGBD, e a linguagem SQL.

O PostgreSQL é um Sistema Gerenciador de Banco de Dados Relacional (SGBDR) de código aberto e um dos mais populares do mercado. Sua robustez, confiabilidade e flexibilidade o tornam uma escolha ideal para ambientes empresariais, incluindo padarias. Ele oferece recursos avançados, como suporte a transações, controle de concorrência, integridade dos dados e suporte a consultas complexas, que podem ser explorados para desenvolver um sistema de banco de dados eficiente e adaptado às necessidades específicas de uma padaria (Silberschatz, Korth & Sudarshan, 2019).

Ao longo deste trabalho, serão abordados os principais aspectos envolvidos no desenvolvimento desse sistema gerenciador de banco de dados para a padaria, desde a modelagem do banco de dados, a definição das tabelas e relacionamentos, até a implementação das funcionalidades necessárias para atender às demandas do estabelecimento.

Dessa forma, espera-se que este estudo contribua para a compreensão dos benefícios da adoção de um sistema de banco de dados eficiente em uma padaria, bem como demonstre a aplicação prática do PostgreSQL nesse contexto. Além disso, pretende-se apresentar possíveis melhorias e otimizações que podem ser implementadas no sistema, a fim de aprimorar ainda mais a gestão e o desempenho da padaria.

1.1 Justificativa

A padaria é um estabelecimento que envolve uma série de processos complexos, desde a produção dos produtos, como pães, bolos e salgados, até a gestão de estoque, controle de vendas, atendimento ao cliente e gerenciamento financeiro. Um sistema de banco de dados eficiente é essencial para integrar todas essas atividades, permitindo o armazenamento, organização e recuperação de dados de forma rápida e segura.

A escolha do PostgreSQL como SGBD para esse sistema se baseia em sua reputação como uma solução robusta, confiável e flexível. Sua popularidade no mercado e a ampla adoção por empresas de diversos segmentos comprovam sua eficiência. Além disso, o PostgreSQL oferece recursos avançados, como suporte a transações, controle de concorrência e consultas complexas, que podem atender às demandas específicas de uma padaria.

1.2 Definição do Problema

O problema abordado neste trabalho é a falta de um sistema gerenciador de banco de dados eficiente e adequado às necessidades de uma padaria. Atualmente, muitas padarias ainda dependem de processos manuais ou sistemas desatualizados para realizar tarefas como controle de estoque, registro de vendas, gerenciamento financeiro e atendimento ao cliente.

1.3 Objetivo Geral

Desenvolver e implementar um sistema gerenciador de banco de dados para uma padaria, utilizando o PostgreSQL como SGBD, a fim de otimizar as operações internas, melhorar o controle de estoque, registrar vendas, gerenciar informações de clientes e proporcionar uma gestão eficiente e integrada para a empresa.

1.4 Objetivos Específicos

- Realizar a análise das necessidades e demandas específicas da padaria em termos de gerenciamento de dados, levando em consideração os processos de produção, estoque, vendas e atendimento ao cliente.
- Projetar a estrutura do banco de dados, definindo tabelas, atributos e relacionamentos necessários para atender às demandas identificadas. Implementar o sistema gerenciador de banco de dados utilizando o PostgreSQL, configurando e ajustando os parâmetros adequados para garantir a eficiência e a segurança dos dados.
- Desenvolver as funcionalidades do sistema, incluindo módulos de controle de estoque, registro de vendas, gerenciamento de clientes e geração de relatórios relevantes para a tomada de decisões. Testar e validar o sistema gerenciador de banco de dados, realizando testes de desempenho, confiabilidade e usabilidade para garantir a qualidade e eficiência do sistema.

2 METODOLOGIA

Para o desenvolvimento e implementação do sistema gerenciador de banco de dados, foi adotada uma abordagem de desenvolvimento iterativa e incremental. Esse método permite que o sistema seja construído e aprimorado em etapas, possibilitando a validação contínua dos requisitos e a entrega gradual de funcionalidades.

dBeaver: É uma ferramenta de administração e desenvolvimento de banco de dados de código aberto, que oferece uma interface intuitiva para executar consultas, gerenciar esquemas, tabelas e visualizar dados. O dBeaver foi utilizado para realizar operações de gerenciamento do PostgreSQL, como a criação de tabelas, relacionamentos, consultas e manipulação dos dados.

Visual Paradigm: É uma ferramenta de modelagem visual de dados e processos. Foi utilizada para a criação dos diagramas de modelagem do banco de dados, permitindo a definição das entidades, atributos, relacionamentos e restrições necessárias para o sistema.

brModelo: É uma ferramenta gratuita de modelagem de banco de dados que permite a criação de diagramas entidade-relacionamento (DER). O brModelo foi utilizado para a modelagem inicial do banco de dados, facilitando a visualização e definição das estruturas e relacionamentos do sistema.

A linguagem SQL (Structured Query Language) foi utilizada para a implementação do sistema gerenciador de banco de dados. O SQL é uma linguagem de programação específica para a manipulação de dados em bancos de dados relacionais. No desenvolvimento do sistema, foram utilizadas as principais operações do SQL, como a criação de tabelas, inserção, atualização e exclusão de registros, além de consultas complexas para recuperar e filtrar informações. Também foram utilizadas cláusulas SQL avançadas, como JOINS para combinar dados de diferentes tabelas, agregações para calcular valores sumarizados e consultas parametrizadas para tornar o sistema mais flexível e seguro.

O desenvolvimento do sistema gerenciador de banco de dados para a padaria seguiu um ciclo de desenvolvimento iterativo. Inicialmente, foram realizadas

análises detalhadas dos requisitos e necessidades da padaria em relação ao gerenciamento de dados. Com base nessas análises, foram criados os diagramas de modelagem utilizando o Visual Paradigm e o brModelo, definindo as entidades, atributos e relacionamentos do banco de dados. Em seguida, utilizando o dBeaver, foram criadas as tabelas do banco de dados de acordo com a modelagem definida, considerando as características específicas da padaria, como controle de estoque, registro de vendas, gerenciamento de clientes, entre outros. Posteriormente, a implementação das funcionalidades do sistema foi realizada por meio da escrita de consultas SQL adequadas para inserção, atualização, exclusão e consulta de dados no banco de dados PostgreSQL.

Durante o desenvolvimento, foram realizados testes e validações do sistema para garantir a integridade dos dados e a correta execução das operações. O desenvolvimento do sistema gerenciador de banco de dados para a padaria seguiu um fluxo contínuo de análise, modelagem, implementação e testes, visando atender aos objetivos específicos estabelecidos e fornecer uma solução eficiente para a gestão das informações da padaria.

3 DESENVOLVIMENTO

Nesta etapa, foi realizada uma análise minuciosa das necessidades e demandas da padaria em relação ao gerenciamento de suas informações. Foram identificadas as principais áreas a serem abordadas, como controle de estoque, registro de vendas, gerenciamento de clientes e gestão financeira. Essa análise serviu como base para o desenvolvimento do sistema, garantindo que as funcionalidades atendam de forma eficaz aos requisitos do estabelecimento. Partimos então para uma análise dos requisitos funcionais e não-funcionais da padaria. Tivemos uma lista grande, mas obviamente incompleta, pois, como todo sistema de banco, sempre há algo a mais para ser complementado. Seguindo o modelo da apostila e pelo conhecimento na área de padarias, levantamos os requisitos e demos início a modelagem de nosso banco.

3.1 Requisitos (Funcionais e Não-Funcionais)

- Gerenciamento de produtos: Cadastro de produtos (pães, bolos, salgados etc.) com informações como nome, descrição, preço e ingredientes. Controle de estoque, incluindo o registro de entrada e saída de produtos. Possibilidade de definir alertas para produtos com estoque baixo.
- Sistema de pedidos: Registro e gerenciamento de pedidos de clientes. Opção de criar pedidos personalizados, permitindo escolher produtos e quantidades específicas. Cálculo automático do valor total do pedido.
- Gestão de clientes: Cadastro de clientes com informações como nome, endereço, telefone e histórico de compras. Possibilidade de criação de programas de fidelidade para recompensar clientes frequentes.
- Ponto de Venda: Emissão de cupom fiscal ou nota fiscal eletrônica. Aceitação de diferentes formas de pagamento, como dinheiro, cartões de crédito/débito e pagamento móvel. Registro e controle de vendas por operador de caixa.

- Gestão de funcionários: Cadastro de funcionários, incluindo informações pessoais, carga horária e funções. Controle de ponto eletrônico. Geração de relatórios de desempenho e produtividade dos funcionários.
- Controle financeiro: Registro de receitas e despesas. Cálculo de lucro e margem de lucro. Geração de relatórios financeiros, como balanço patrimonial e demonstrativo de resultados.
- Agendamento de encomendas: Possibilidade de agendar a produção de produtos específicos para datas futuras. Notificação automática de encomendas prontas para retirada.
- Integração com delivery: Integração com aplicativos de delivery para receber pedidos e gerenciá-los. Atualização automática do status dos pedidos para entrega.
- Relatórios e análises: Geração de relatórios com dados sobre vendas, estoque, clientes e desempenho do negócio. Análise de dados para identificar tendências e tomar decisões estratégicas.
- Escalabilidade: O sistema deve ser dimensionável para acomodar o crescimento futuro da padaria, suportando um aumento no número de produtos, clientes e pedidos.
- Usabilidade: Interface intuitiva e de fácil utilização para os funcionários, mesmo para aqueles com pouca experiência em sistemas computacionais. Resposta rápida do sistema para minimizar o tempo de espera durante as transações.
- Desempenho: Tempo de resposta rápido para todas as operações, como cadastro de produtos, registro de vendas e geração de relatórios.
- Escalabilidade: Capacidade de expandir o sistema facilmente para atender a um aumento no número de produtos, clientes e transações. Suporte a múltiplas filiais, permitindo que a padaria gerencie várias unidades.
- Manutenção: Facilidade de manutenção do sistema, permitindo atualizações de software, correções de bugs e melhorias funcionais.
- Regulamentação: Conformidade com as leis e regulamentos aplicáveis, como requisitos fiscais e normas de proteção de dados.

- Promoções: O sistema deve permitir a criação e aplicação de promoções especiais para determinados produtos ou períodos específicos, como descontos sazonais ou ofertas especiais para ocasiões especiais.
- Personalização: O sistema deve ser capaz de lidar com pedidos personalizados dos clientes, como solicitações de decoração específica em bolos ou ingredientes alternativos para atender a restrições dietéticas.
- Cálculo de custos: O sistema deve ser capaz de calcular o custo dos produtos com base nos ingredientes utilizados, levando em consideração a porcentagem de desperdício e os custos operacionais.
- Controle de validade: O sistema deve ser capaz de rastrear a validade dos produtos para evitar a venda de itens vencidos, emitindo alertas quando a data de validade estiver próxima.

Como nossa disciplina de banco nesse semestre não buscou nos formar no design do SGBD, partes como a personalização do sistema provavelmente estarão presentes no futuro.

3.2 Modelagem Conceitual

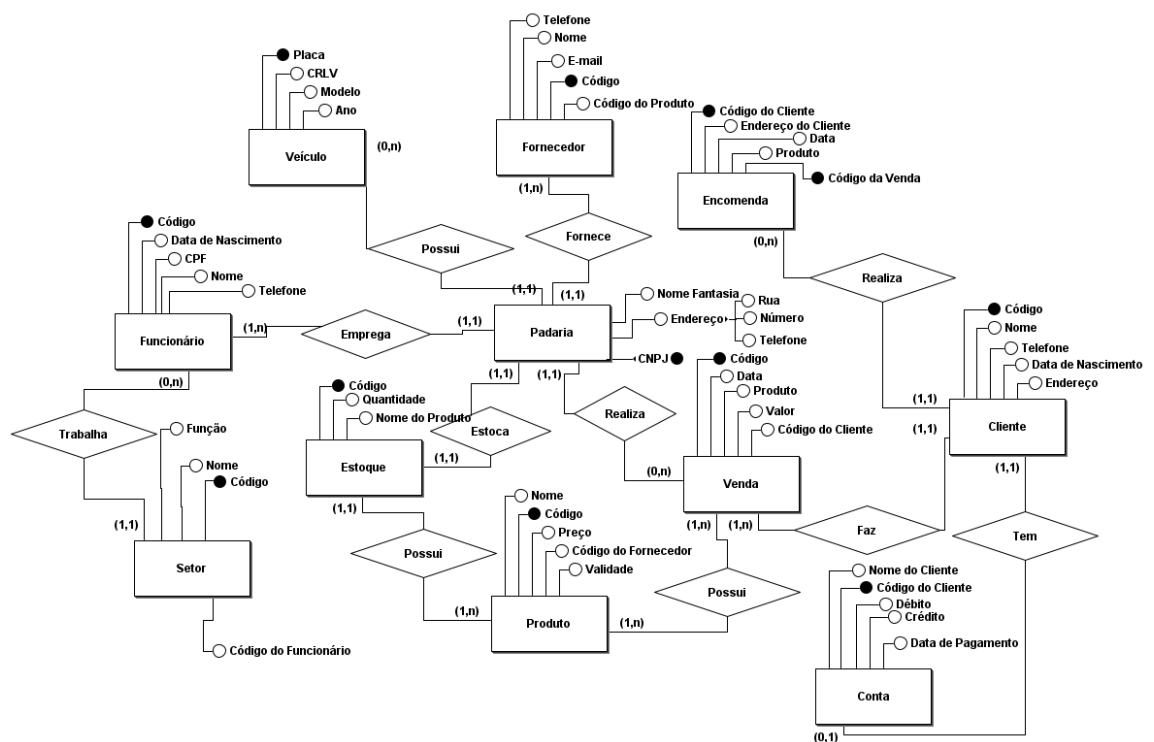
A modelagem conceitual é uma etapa essencial no processo de desenvolvimento de um sistema gerenciador de banco de dados. Essa etapa visa representar, de forma abstrata, as entidades, atributos e relacionamentos envolvidos no domínio do problema a ser resolvido, porém sabemos que ela não era solicitada no projeto, mas acreditamos que sua ajuda foi essencial.

A modelagem conceitual permite visualizar de forma clara e organizada as entidades que compõem o sistema, seus atributos e os relacionamentos entre elas. Essa representação gráfica facilita a compreensão e a comunicação entre os membros da equipe do Sistema BreadTrack. Com o brModelo, foi possível definir as entidades específicas da padaria, como produtos, clientes, fornecedores, vendas e estoque. Cada entidade foi identificada com seus atributos correspondentes, como

nome, código, descrição, preço, entre outros. Além disso, os relacionamentos entre as entidades foram estabelecidos, como a associação entre produtos e fornecedores, produtos e vendas, e clientes e vendas.

Ao final da modelagem conceitual, o resultado obtido foi um diagrama entidade-relacionamento que serviu como base para a implementação do banco de dados. Esse diagrama serviu como guia para a criação das tabelas e definição dos relacionamentos no PostgreSQL. Segue abaixo na Figura 1 o DER. Conforme está ilustrado no Anexo – A, no fim do documento, segue o arquivo sobre a Modelagem Conceitual.

Figura 1 - Diagrama Entidade Relacionamento.



Fonte: Autoria própria.

Em resumo, a modelagem conceitual desempenhou um papel fundamental no desenvolvimento do sistema gerenciador de banco de dados para a padaria, permitindo a representação visual das entidades, atributos e relacionamentos envolvidos, bem como a definição de restrições e regras de negócio. O uso da

ferramenta brModelo facilitou a criação dessa modelagem conceitual, contribuindo para o sucesso do projeto.

3.3 Modelagem Relacional

Durante o processo de modelagem relacional, as tabelas foram criadas com base nas entidades definidas na modelagem conceitual. Cada tabela representa uma entidade específica da padaria, como produtos, clientes, fornecedores e vendas.

Os atributos de cada entidade foram mapeados como colunas nas tabelas, representando as características e informações relacionadas a cada entidade. Além disso, os relacionamentos entre as entidades foram implementados na forma de chaves estrangeiras. Por exemplo, a tabela de vendas possui uma chave estrangeira que faz referência à tabela de clientes, estabelecendo assim a relação entre vendas e clientes.

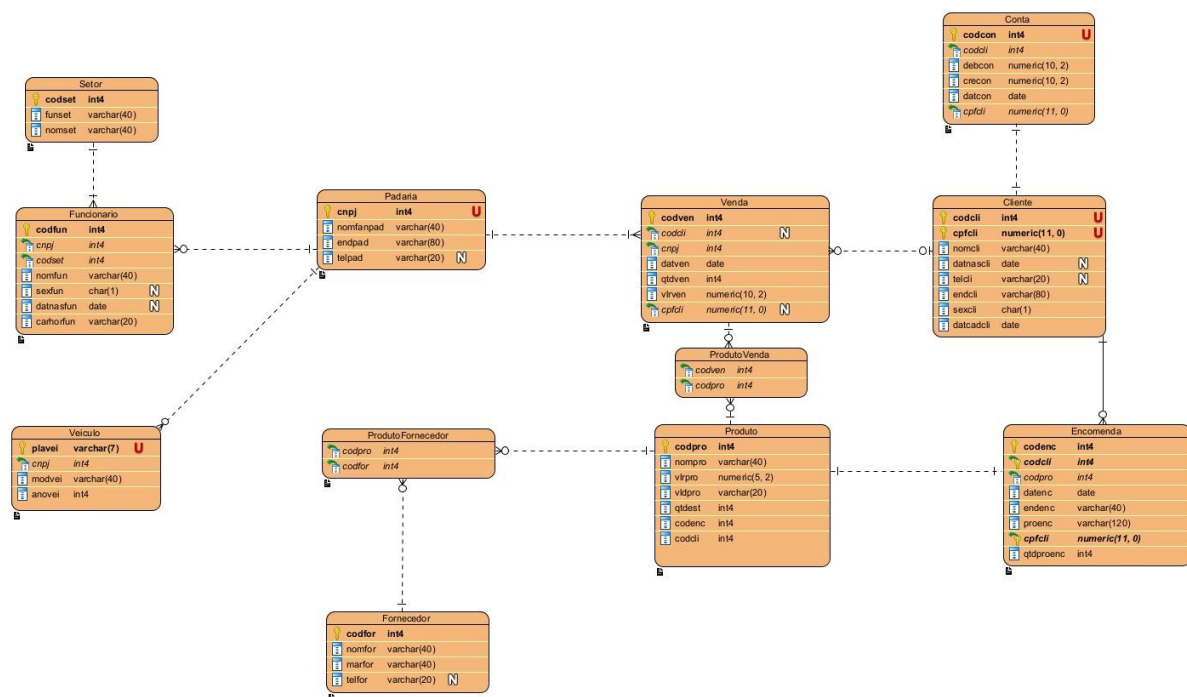
A modelagem relacional também permite a definição de restrições de integridade, como chaves primárias, chaves estrangeiras e restrições de valor. As chaves primárias são utilizadas para identificar de forma única cada registro em uma tabela, enquanto as chaves estrangeiras estabelecem os relacionamentos entre as tabelas.

As restrições de valor garantem que os dados inseridos nas tabelas estejam de acordo com as regras estabelecidas. Essa modelagem relacional proporciona uma estrutura organizada e consistente para o banco de dados da padaria, permitindo o armazenamento e a recuperação eficiente das informações. Além disso, ela facilita a realização de consultas e operações sobre os dados, como busca, inserção, atualização e exclusão.

Em resumo, a modelagem relacional foi utilizada como base para a implementação do banco de dados do sistema gerenciador da padaria. Essa abordagem permitiu a representação lógica dos dados em formato de tabelas, estabelecendo os relacionamentos entre as entidades e definindo as restrições de integridade necessárias.

Na Figura 2, podemos ver o modelo relacional de nosso projeto, feito no Visual Paradigm com base em nosso modelo conceitual também. No projeto do VPP (Visual Paradigm Project) podemos ver a lógica entre as tabelas e também as relações e também as interações entre todos os componentes que formam nosso banco. Conforme está ilustrado no Anexo – B, no fim do documento, segue o arquivo sobre a Modelagem Relacional.

Figura 2 - Modelo Relacional.



Fonte: Autoria própria.

Para o desenvolvimento do diagrama, tivemos como ponto de foco a tabela “Padaria”, essa tabela é responsável por armazenar dados da padaria como empresa. Após isso, criamos outras tabelas mantendo para agregar e normalizar nosso projeto.

Tabelas como: “Funcionário”, “Setor”, “Cliente”, “Conta”, “Venda”, “Produto” foram criadas para evitar a quebra das três regras de normalização e garantindo assim, o funcionamento correto de nosso sistema ganhando mais consistência.

Após a implementação das funcionalidades, foram realizados testes e validações do sistema para garantir seu correto funcionamento. Foram executados





testes unitários para cada funcionalidade, verificando se os dados eram armazenados corretamente no banco de dados, se as consultas retornavam os resultados esperados e se as regras de integridade e validação eram respeitadas. Além disso, foram conduzidos testes de integração para verificar a interação adequada entre as diferentes funcionalidades.

3.3.1 Dicionário de Dados (Data Dictionary)

As próximas figuras (Figura 3, Figura 4 e Figura 5) mostram o Dicionário de Dados de nosso projeto. Conforme está ilustrado no Anexo – C, no fim do documento, segue o arquivo sobre o Dicionário de Dados.







Figura 3 - Dicionário de Dados.

1. Data Dictionary

Entity Name	Entity Description					
Column Name	Column Description	Data Type	Length	Primary Key	Nullable	Unique
 Cliente						
codcli	Código do Cliente	int4	0	true	false	true
cpfcli	CPF do Cliente	numeric	11	true	false	true
datcadcli	Data de cadastro do cliente	date	0	false	false	false
datnascli	Data de Nascimento do Cliente	date	0	false	true	false
endcli	Endereço do Cliente	varchar	80	false	false	false
nomcli	Nome do Cliente	varchar	40	false	false	false
sexcli	Sexo do cliente	char	1	false	false	false
telcli	Telefone do Cliente	varchar	20	false	true	false
 Conta Tabela de Cadastro de Conta do Cliente						
codcli	Código do Cliente	int4	0	false	false	false
codcon	Código da Conta do Cliente	int4	10	true	false	true
cpfcli		numeric	11	false	false	false
crecon	Crédito da Conta do Cliente	numeric	10.2	false	false	false
datcon	Data da Criação da Conta do Cliente	date	0	false	false	false
debcon	Débito da Conta do Cliente	numeric	10.2	false	false	false
 Encomenda Tabela de Cadastro de Encomenda						
codcli	Código do Cliente	int4	0	true	false	false
codenc	Código da Encomenda	int4	0	true	false	false
codpro		int4	10	false	false	false
cpfcli		numeric	11	true	false	false
datenc	Data da Encomenda	date	0	false	false	false
endenc	Endereço da Encomenda	varchar	40	false	false	false
proenc	Produto da Encomenda	varchar	120	false	false	false
qtdproenc	Quantidade de produtos da encomenda	int4	4	false	false	false
 Fornecedor Tabela de Cadastro do Fornecedor						



Fonte: Autoria própria.

Figura 4 - Dicionário de Dados (2).

codfor	Código do Fornecedor	int4	10	true	false	false
marfor	Marca do Fornecedor	varchar	40	false	false	false
nomfor	Nome do Fornecedor	varchar	40	false	false	false
telfor	Telefone do Fornecedor	varchar	20	false	true	false
 Funcionario	Tabela de Cadastro de Funcionário					
carhorfun	Carga Horária do Funcionário	varchar	20	false	false	false
cnpj		int4	14	false	false	false
codfun	Código do Funcionário	int4	10	true	false	false
codset	Código do Setor do Funcionário	int4	10	false	false	false
datnasfun	Data de Nascimento do Funcionário	date	0	false	true	false
nomfun	Nome do Funcionário	varchar	40	false	false	false
sexfun	Sexo do Funcionário	char	1	false	true	false
 Padaria	Tabela de Cadastro da Padaria					
cnpj	CNPJ da Padaria	int4	14	true	false	true
endpad	Endereço da Padaria	varchar	80	false	false	false
nomfanpad	Nome Fantasia da Padaria	varchar	40	false	false	false
telpad	Telefone da Padaria	varchar	20	false	true	false
 Produto	Tabela de Cadastro de Produto					
codcli		int4	10	false	false	false
codenc		int4	10	false	false	false
codpro	Código do Produto	int4	10	true	false	false
nompro	Nome do Produto	varchar	40	false	false	false
qtdest	Quantidade de Estoque	int4	10	false	false	false
vldpro	Validade do Produto	varchar	20	false	false	false
vlrpro	Valor do Produto	numeric	5.2	false	false	false
 ProdutoFornecedor	Fornecedores do produto					
codfor		int4	10	false	false	false
codpro		int4	10	false	false	false
 ProdutoVenda						
codpro		int4	10	false	false	false
codven		int4	10	false	false	false
 Setor	Tabela de Cadastro do Setor					
codset	Código do Setor	int4	10	true	false	false

Fonte: Autoria própria.

Figura 5 - Dicionário de Dados (3).

funset	Função do Setor	varchar	40	false	false	false
nomset	Nome do Setor do Funcionário	varchar	40	false	false	false
 Veiculo	Tabela de Cadastro de Veículo					
anovei	Ano do Veículo	int4	4	false	false	false
cnpj	CNPJ da Padaria	int4	14	false	false	false
modvei	Modelo do Veículo	varchar	40	false	false	false
plavei	Placa do Veículo	varchar	7	true	false	true
 Venda	Tabela de Cadastro da Venda					
cnpj	CNPJ da Padaria	int4	14	false	false	false
codcli	Código do Cliente	int4	0	false	true	false
codven	Código da Venda	int4	10	true	false	false
cpfcli		numeric	11	false	true	false
datven	Data da Venda	date	0	false	false	false
qtdven	Quantidade de Produto	int4	10	false	false	false
vlrven	Valor da Venda	numeric	10.2	false	false	false

Fonte: Autoria própria.

3.4 Banco de Dados

Depois da construção e normalização do modelo relacional com o uso da ferramenta Visual Paradigm, foi dada sequência no projeto com a criação dos scripts.

Após a execução dos scripts de criação, a base de dados estará pronta para receber e armazenar os dados relacionados às entidades da padaria. Os scripts também garantem a integridade dos dados, por meio das restrições definidas, como chaves primárias e chaves estrangeiras.

E com isso criamos a nossa DataBase do sistema BreadTrack para posteriormente construirmos os scripts necessários para o projeto.

3.4.1 Scripts da criação da base de dados

Os scripts são responsáveis por criar as tabelas que representam as entidades do sistema, definindo seus atributos e estabelecendo as chaves primárias

e chaves estrangeiras. Além disso, os scripts também podem incluir a criação de índices para melhorar a performance das consultas.

A execução dos scripts de criação da base de dados resulta em uma estrutura sólida e organizada para o armazenamento e manipulação dos dados da padaria. Através desses scripts, a base de dados é criada com todas as tabelas necessárias e suas configurações correspondentes.

Cada tabela é definida com seus atributos e relacionamentos, garantindo a integridade dos dados e a eficiência das operações. Dessa forma, os scripts de criação da base de dados são essenciais para estabelecer a estrutura necessária para o funcionamento do sistema gerenciador de banco de dados da padaria. Eles fornecem a base sólida para o armazenamento das informações e permitem a realização de consultas e operações de forma eficiente.

Previamente geramos os scripts de criação da base de dados com base em nosso Modelo Relacional, e após isso realizamos os inserts nas tabelas para popular nosso banco de dados com informações.

Segue na Figura 6 um exemplo de uma (das 11) tabelas que criamos com o script.

Figura 6 - Exemplo da criação da tabela Cliente.

```
1 CREATE TABLE Cliente (  
2     codcli    int4 NOT NULL UNIQUE,  
3     cpfcli    numeric(11, 0) NOT NULL UNIQUE,  
4     nomcli    varchar(40) NOT NULL,  
5     datnascli date,  
6     telcli    varchar(20),  
7     endcli    varchar(80) NOT NULL,  
8     sexcli    char(1) NOT NULL,  
9     datcadcli date NOT NULL,  
10    PRIMARY KEY (codcli,  
11    cpfcli));  
12 COMMENT ON COLUMN Cliente.codcli IS 'Código do Cliente';  
13 COMMENT ON COLUMN Cliente.cpfcli IS 'CPF do Cliente';  
14 COMMENT ON COLUMN Cliente.nomcli IS 'Nome do Cliente';  
15 COMMENT ON COLUMN Cliente.sexcli IS 'Sexo do Cliente';  
16 COMMENT ON COLUMN Cliente.datnascli IS 'Data de Nascimento do Cliente';  
17 COMMENT ON COLUMN Cliente.telcli IS 'Telefone do Cliente';  
18 COMMENT ON COLUMN Cliente.endcli IS 'Endereço do Cliente';  
19
```

Após a criação das tabelas como falamos, tivemos que popular nosso banco com informações genéricas para enfim poder efetuar os relatórios. Os comandos utilizados para popular o banco de dados da padaria incluiu o uso de INSERT INTO, SELECT, VALUES e INSERT INTO ... SELECT. Esses comandos permitiram inserir dados de forma estruturada e consistente, contribuindo para a completa funcionalidade e utilização do sistema gerenciador de banco de dados da padaria. A Figura 7 mostra um exemplo de população de dados na tabela Cliente.

Figura 7 - Exemplo de criação de dados na tabela Cliente.

```
INSERT INTO Cliente (codcli, cpfcli, nomcli, datnascli, telcli, endcli,sexcli, datcadcli)
VALUES
(1, 12345678901, 'João Silva', '1990-01-01', '999999999', 'Rua 25 de Março, 123','M','2018-09-18'),
(2, 98765432109, 'Maria Souza', '1985-05-10', '888888888', 'Avenida Avelino de Matias , 456','F','2009-05-06'),
(3, 45678912304, 'Carlos Ferreira', '1998-08-15', '777777777', 'Rua do Cristo, 789','M','2023-09-21'),
(4, 32165498707, 'Ana Oliveira', '1992-03-20', '666666666', 'Avenida Brasil, 321','F','2020-10-30'),
(5, 74185296300, 'Lucas Santos', '2023-12-25', '555555555', 'Rua Aurelio, 987','M','2005-03-11'),
(6, 95637265311, 'Arthur Costa', '2023-05-29', '444444444', 'Avenida João Pedro, 454','M','2023-07-03'),
(7, 85743421321, 'Matheus Machado', '1989-08-21', '111111111', 'Rua Almeida, 768','M','2021-03-02'),
(8, 87548725382, 'Marcelo Freitas', '2004-06-02', '222222222', 'Rua Garcia, 990','M','2006-07-29'),
(9, 98565263516, 'Clarice Marques', '2023-02-02', '333333333', 'Avenida Carlos Alberto, 321','M','2023-06-01'),
(10, 12319243412, 'Fabio Charnoski', '2000-07-26', '101010101', 'Avenida Paulista, 666','M','2023-02-04');
```

Fonte: Autoria própria.

3.4.2 Relatórios previsto pelo modelo de negócio.

Com o banco em si já gerado, partimos para os relatórios disponibilizados pelo professor. Estudamos e discutimos em grupo a melhor maneira para realizar os selects e distribuir a informação da maneira mais consistente possível. Porém como todo trabalho, encontramos erros em nosso banco que no futuro, caso o projeto continuar, teremos que corrigir para, novamente, entregar os relatórios da maneira mais correta possível. Segue as pesquisas que nos foram encaminhadas:

- 1) Relação de clientes com o nome, sexo e idade em ordem crescente de nome. Relacionar somente clientes cadastrados antes de 2023;
- 2) Relação de produtos (nome e descrição) vendidos nos meses pares de 2022. Ordene o relatório pelo nome do produto de forma ascendente;
- 3) Relação dos top 10 produtos vendidos em 2023;

- 4) Relação de meses, quantidade total de vendas, valor total de vendas por mês. Relacionar somente meses com quantidade de vendas acima de 100. Ordenar o relatório do mês com o maior valor(R\$) em vendas para o mês com menos vendas.

Após a discussão realizamos os comandos selects que estão representados nas próximas Figuras (8 e 9).

Figura 8 - Comandos SELECT (Resposta 1 e 2).

```
scr_relatório_breadtrack.sql
C: > Users > boont > OneDrive > Documentos > UNOESC > Banco de Dados I > BreadTrack > scripts > scr_relatório_breadtrack.sql

1  --1) Relatório de clientes com o nome, sexo e idade em ordem crescente de nome.
2  --Relacionar somente clientes cadastrados antes de 2023;
3  SELECT nomcli, sexcli, EXTRACT(YEAR FROM AGE(datnasccli)) AS idade
4  FROM Cliente
5  WHERE datcadcli < '2023-01-01'
6  ORDER BY nomcli ASC;
7
8
9  --2) Relação de produtos(nome e descrição) vendidos nos meses pares de 2022.
10 --Ordene o relatório pelo nome do produto de forma ascendente;
11 SELECT p.nompro, e.proenc
12 FROM Produto p
13 INNER JOIN Encomenda e ON p.codpro = e.codpro
14 WHERE EXTRACT(YEAR FROM e.datenc) = 2022
15 | -- AND EXTRACT(MONTH FROM e.datenc) % 2 = 0
16 ORDER BY p.nompro ASC;
17
```

Fonte: Autoria própria.

Figura 9 - Comandos SELECT (Resposta 3 e 4).

```
18
19 --3) Relação dos top 10 produtos vendidos em 2023;
20 SELECT p.nompro, SUM(v.qtdven) AS total_vendas
21 FROM Produto p
22 INNER JOIN ProdutoVenda pv ON p.codpro = pv.codpro
23 INNER JOIN Venda v ON pv.codven = v.codven
24 WHERE EXTRACT(YEAR FROM v.datven) = 2023
25 GROUP BY p.nompro
26 ORDER BY total_vendas DESC
27 LIMIT 10;
28
29 --4) Relação de meses, quantidade total de vendas, valor total de vendas por mês.
30 --Relacionar somente meses com quantidade de vendas acima de 100.
31 --Ordenar o relatório do mês com o maior valor(R$) em vendas para o mês com menos vendas.
32 SELECT EXTRACT(MONTH FROM v.datven) AS mes, COUNT(*) AS quantidade_vendas, SUM(v.vlrven) AS valor_total_vendas
33 FROM Venda v
34 where vlrven > 10
35 GROUP BY mes
36 ORDER BY valor_total_vendas DESC, quantidade_vendas ASC;
37
38
```

Fonte: Autoria própria.

3.4.3 Implementação das regras de integridade

A implementação das constraints é realizada durante a criação das tabelas ou posteriormente, por meio de alterações na estrutura do banco de dados. Essas restrições são fundamentais para assegurar a consistência dos dados e prevenir operações indesejadas que possam comprometer a integridade das informações. Ao aplicar as regras de integridade no sistema gerenciador de banco de dados da padaria, é possível evitar inconsistências, garantir a confiabilidade dos dados e facilitar a manutenção do banco de dados a longo prazo.

Portanto, a implementação das regras de integridade no sistema é uma etapa crucial para assegurar a qualidade e a consistência dos dados armazenados na padaria. Elas ajudam a evitar problemas futuros e contribuem para a confiabilidade e o sucesso do sistema gerenciador de banco de dados.

Em nosso projeto, após toda a criação das tabelas, realizamos o fechamento das integridades do sistema. Segue a Figura 10.

Figura 10 - Regras de Integridade.

```

173 ALTER TABLE Venda ADD CONSTRAINT FKVenda891572 FOREIGN KEY (cnpj) REFERENCES Padaria (cnpj);
174 ALTER TABLE Conta ADD CONSTRAINT FKConta96578 FOREIGN KEY (codcli, cpfcli) REFERENCES Cliente (codcli, cpfcli);
175 ALTER TABLE Veiculo ADD CONSTRAINT FKVeiculo134298 FOREIGN KEY (cnpj) REFERENCES Padaria (cnpj);
176 ALTER TABLE Funcionario ADD CONSTRAINT FKFuncionario869187 FOREIGN KEY (codset) REFERENCES Setor (codset);
177 ALTER TABLE Encomenda ADD CONSTRAINT FKEncomenda274345 FOREIGN KEY (codcli, cpfcli) REFERENCES Cliente (codcli, cpfcli);
178 ALTER TABLE ProdutoVenda ADD CONSTRAINT FKProdutoVen34663 FOREIGN KEY (codven) REFERENCES Venda (codven);
179 ALTER TABLE ProdutoVenda ADD CONSTRAINT FKProdutoVen664344 FOREIGN KEY (codpro) REFERENCES Produto (codpro);
180 ALTER TABLE ProdutoFornecedor ADD CONSTRAINT FKProdutoFor834255 FOREIGN KEY (codpro) REFERENCES Produto (codpro);
181 ALTER TABLE ProdutoFornecedor ADD CONSTRAINT FKProdutoFor239414 FOREIGN KEY (codfor) REFERENCES Fornecedor (codfor);
182 ALTER TABLE Venda ADD CONSTRAINT FKVenda345088 FOREIGN KEY (codcli, cpfcli) REFERENCES Cliente (codcli, cpfcli);
183 ALTER TABLE Encomenda ADD CONSTRAINT FKEncomenda511952 FOREIGN KEY (codpro) REFERENCES Produto (codpro);
184 ALTER TABLE Funcionario ADD CONSTRAINT FKFuncionario369823 FOREIGN KEY (cnpj) REFERENCES Padaria (cnpj);
185

```

Fonte: Autoria própria.

Por fim, esse foi nosso desenvolvimento do projeto de sistema. Obviamente não está completo, mas com o tempo vamos trabalhando e adquirindo o conhecimento para aprimorar nossas habilidades com o banco. Lembrando que todos os scripts estarão em anexo nesse trabalho. Conforme está ilustrado no Anexo – D, no fim do documento, segue a pasta completa sobre os scripts de nosso trabalho.

4 ANEXOS

ANEXO A – **Modelagem Conceitual**. Disponível no GitHub em: <https://github.com/brunogkonzen/BreadTrack/blob/main/modelagem/BreadTrack%20-%20Modelagem%20Conceitual.png>.

ANEXO B – **Modelagem Relacional**. Disponível no GitHub em: <https://github.com/brunogkonzen/BreadTrack/blob/main/modelagem/BreadTrack%20-%20ProjetoVPP.vpp>.

ANEXO C – **Dicionário de Dados**. Disponível no GitHub em: <https://github.com/brunogkonzen/BreadTrack/blob/main/modelagem/BreadTrack%20-%20Data%20Dictionary.pdf>.

ANEXO D – **Scripts do Banco de Dados**. Disponível no GitHub em: <https://github.com/brunogkonzen/BreadTrack/tree/main/scripts>.

5 CONCLUSÃO

Durante todo o desenvolvimento do sistema, foram aplicados os conceitos e conhecimentos adquiridos na disciplina de bancos de dados.

Ao longo deste trabalho, foram abordados conceitos fundamentais de sistemas gerenciadores de banco de dados, o papel desses sistemas em padarias e a apresentação do PostgreSQL como uma escolha adequada para esse contexto. Foi discutida a importância da modelagem conceitual e da modelagem relacional na estruturação do banco de dados, bem como a utilização de ferramentas como o brModelo para auxiliar nesse processo.

A metodologia utilizada envolveu a análise das necessidades específicas da padaria, a modelagem do banco de dados, a implementação das funcionalidades necessárias, como controle de estoque, registro de vendas e gerenciamento de clientes, e a realização de testes para validar o sistema.

Durante o desenvolvimento, foi possível perceber os benefícios de um sistema gerenciador de banco de dados eficiente. O PostgreSQL demonstrou-se uma ferramenta robusta, confiável e flexível, oferecendo recursos avançados que permitiram o desenvolvimento de um sistema adaptado às necessidades da padaria.

A partir da implementação do sistema gerenciador de banco de dados, espera-se que a padaria possa desfrutar de uma gestão mais eficiente de suas informações, contribuindo para a tomada de decisões estratégicas, o controle de estoque, a otimização de processos e a satisfação dos clientes.

No entanto, é importante ressaltar que este trabalho acadêmico apresenta uma visão geral do sistema gerenciador de banco de dados para a padaria, e existem possibilidades de aprimoramentos e otimizações que podem ser exploradas no futuro. Essas melhorias podem incluir a implementação de novas funcionalidades, a incorporação de técnicas avançadas de consultas e a integração com outros sistemas da padaria.

Destacamos também a falta de prática de cada um dos membros por nunca ter tido contato com a área de banco de dados, em alguns momentos oscilamos, mas nunca deixamos de tentar, por fim observamos que sem um levantamento de

requisitos e funcionalidades do banco fica muito difícil ter um projeto consistente. Sem o levantamento completo e detalhado dos requisitos e funcionalidades do banco de dados não é possível ter um projeto concreto de bom entendimento.

Em suma, este trabalho buscou contribuir para a compreensão dos benefícios e da aplicação prática do sistema gerenciador de banco de dados em uma padaria. Espera-se que os resultados obtidos e as considerações apresentadas possam servir como base para futuros estudos e aprimoramentos nessa área, visando sempre o aperfeiçoamento da gestão e o sucesso das padarias e estabelecimentos similares.

Agradecemos a oportunidade de trabalhar com o banco de dados, toda experiência ganha vai com toda certeza nos ajudar a construir um futuro sólido e tranquilo e mais uma vez destacamos nossos erros e deixamos claro que estamos preparados para seguir em frente com o projeto e cada vez mais buscando sua perfeição.

6 REFERÊNCIAS

ALVES, Roberson J. F. **Apostila de Banco de Dados. São Miguel do Oeste: Unoesc, 2023. Material didático.** Acesso em: 28, jun 2023.

AUTOR DESCONHECIDO. **Usando scripts de exemplo para criar bancos de dados DB2.** Disponível em: <https://www.ibm.com/docs/pt-br/control-center/6.1.1?topic=users-using-example-scripts-create-db2-databases>. Acesso em: 22, jun 2023.

ORACLE. **O que é um Banco de Dados?** Disponível em: <https://www.oracle.com/br/database/what-is-database/>. Acesso em: 15, jun 2023.

REZENDE, Ricardo. **Conceitos Fundamentais de Banco de Dados.** Disponível em: <https://www.devmedia.com.br/conceitos-fundamentais-de-banco-de-dados/1649>. Acesso em: 01, jul 2023.