

DESENVOLVIMENTO DE UM BANCO DE DADOS COM APLICAÇÃO EM SQLITE BROWSER VOLTADO PARA PROCESSOS INTERNOS DE UMA ORGANIZAÇÃO DO SETOR DE MODAS

Carla Cristina Ramos de Freitas. carla.ramos@estudante.ufcg.edu.br
Cecir Almeida Farias. cecir.barbosa@professor.ufcg.edu.br
Isabella Thyfany Cavalcante Palmeira. isabellatcp9@gmail.com
Núbia Naely da Costa Benedito. naelycostabenedito@gmail.com
Rafaela Firmino da Costa Leão. rafaela.firmino@estudante.ufcg.edu.br

Resumo

A pesquisa tem como objetivo desenvolver um banco de dados com a interface elaborada na linguagem de programação *Python* para uma empresa do setor de roupas, a fim de armazenar e organizar os registros inseridos de forma a otimizar as atividades realizadas na organização e trazer maior confiança e automatização no armazenamento das informações. Para a construção desse trabalho foi realizada, por meio do Diagrama de Caso de Uso (USE-CASE) e do Diagrama Entidade-Relacionamento (DER), uma análise das ações realizadas diariamente durante os processos da organização, com isso criou-se o banco de dados para trazer melhorias e otimização à organização. Assim, conclui-se que, com base nos resultados obtidos, o uso do *software* possibilitou a observação da segurança desse sistema, uma vez que, o mesmo se faz essencial para controlar e guardar esses dados.

Palavras-Chaves: *Software, SQLite Browser, Python, Banco de Dados, Otimização*

1. Introdução

No atual cenário onde as organizações vêm se desenvolvendo progressivamente e com isso aumentando sua competitividade diante do mercado, a quantidade de dados armazenados também cresce exponencialmente e dessa maneira percebe-se a necessidade de adotar métodos que facilitem o armazenamento, manuseio e análise dos mesmos. Neste segmento, o desenvolvimento de um banco de dados e um sistema para gerenciamento deles, se torna indispensável para o sucesso da organização, uma vez que, após os dados serem armazenados é possível realizar inúmeras consultas e aplicações sem interferir na integridade dos mesmos.

De acordo com Cristina Matsumoto (2019, apud MACHADO, 1996, P.23.), um banco de dados é uma coleção de fatos registrados que refletem o estado de certos aspectos de interesse do mundo real, de maneira que a todo instante os dados presentes no banco representam uma visão instantânea do estado do mundo real, cada mudança em qualquer item do banco de dados reflete uma mudança ocorrida na realidade. A tecnologia instalada no banco de dados objetiva permitir que os dados possam ser mantidos e definidos de forma independente dos sistemas de aplicação que possam ser utilizados.

Durante o processo de desenvolvimento de um banco de dados são realizadas algumas etapas, como a criação dos Diagramas de Caso de Uso (Use Case), Diagramas de Entidade e Relacionamento (DER) e o Modelo Relacional (MR). A primeira consiste em descrever de forma gráfica e organizada um subconjunto do modelo a fim de simplificar a comunicação com o usuário, já a segunda, consiste em “utilizar elementos gráficos para descrever o modelo de dados de um sistema e o relacionamento entre eles” (NETO, 2020). Nesta sequência, de acordo com Garcia e Sotto (2018, apud (MACÁRIO; BALDO, 2005, p.1.), o Modelo Relacional foi desenvolvido a fim de representar um banco de dados de uma forma inovadora, utilizando uma coleção de ferramentas conceituais para descrever dados, relações entre eles e as restrições de consistência.

No desempenho de gerenciamento dos respectivos bancos de dados, são elaborados sistemas, os quais recebem atribuições através de um grupo de softwares utilizados para acessar e realizar a administração dos mesmos, que possibilitam realizar comandos envolvendo as ações de criar, modificar, consultar e deletar tanto os dados como a sua estruturação.

Sendo assim, a ferramenta *SQLite Browser*, se destaca por conter uma simplicidade em administração, implementação e manutenção dos dados armazenados na ferramenta. De acordo com Silva (2019), no *SQLite Browser* é possível criar e manter diversas tabelas de banco de dados, utilizando-se dos mais diversos comandos, onde esses dados podem ser consultados, manipulados e deletados a qualquer instante.

O objetivo deste trabalho se dá pela necessidade em aprimorar e expandir os conhecimentos relacionados acerca da tese apresentada e aplicá-los em um caso real para um maior enriquecimento do que foi visto em teoria. O desenvolvimento do sistema foi voltado para uma empresa do setor de roupas. A aplicação prática presente neste estudo visa estruturar o Diagrama de Entidade e Relacionamento, Modelo Relacional e o banco de dados através do *SQLite Browser* para contribuir com o desenvolvimento da empresa em questão.

2. Referencial Teórico

2.1 Banco de dados

Atualmente, com o desenvolvimento da tecnologia é perceptível a rapidez em que os dados vão assumindo maior quantidade de volume na memória de um hardware, dessa forma criando uma utilidade estratégica e extraindo informações relevantes que influenciam de maneira mais assertiva nas tomadas de decisões. (MATSUMOTO, 2006).

Diante disso, após diversas descobertas e inovações tecnológicas, com as Revoluções Industriais ocasionadas pelo nascimento da modernização da informação e comunicação, houve a criação do computador, que tornou mais simples, clara e objetiva a forma de armazenamento de dados, trazendo ao mundo atual uma série de plataformas que permitem aos usuários acessos quase que ilimitado e completo de informações.

Em virtude disso, tem-se o banco de dados, o qual é uma coleção de dados interrelacionados que visam atender as necessidades de um grupo específico de aplicações e usuários. De acordo com Cristina Matsumoto (2019, apud BATISTA, 2006), o banco associado ao sistema computacional recebe e disponibiliza informações e dados relevantes para uma consulta que venha a ser realizada posteriormente.

2.2 Sistema Gerenciador de Banco de Dados

Segundo os estudos realizados por Coronel (2010), o conceito de Sistema Gerenciador de Banco de Dados (SGBD), consiste em um conjunto de programas que gerenciam a estrutura do banco de dados e controlam o acesso aos dados armazenados.

Os primeiros sistemas gerenciadores de banco de dados, de acordo com Garcia e col. (apud LÓSCIO, et al, 2011, p.2).

[...] surgiram por volta de 1960 e foram desenvolvidos com base nos primeiros sistemas de arquivos. Dentre as principais características de SGBD, destacamos: controle de ocorrência, segurança, recuperação de falhas, gerenciamento do mecanismo de armazenamento de dados e controle das restrições de integridade do BD.

Ao serem realizadas execuções nos SGBDs, as mesmas devem obedecer a algumas propriedades, visando garantir o correto funcionamento do sistema e a consistência dos dados inseridos no mesmo. “Essas propriedades são chamadas de propriedades ACID, na qual se refere às quatro propriedades de transação de um sistema de banco de dados: Atomicidade, Consistência, Isolamento e Durabilidade” (LÓSCIO et al., 2011).

- **Atomicidade:** Em uma transação envolvendo duas ou mais partes de informações discretas, ou a transação será executada totalmente ou não será executada, garantindo assim que as transações sejam atômicas;
- **Consistência:** A transação cria um estado válido dos dados ou em caso de falha retorna todos os dados ao seu estado antes que a transação foi iniciada;
- **Isolamento:** Uma transação em andamento, mas ainda não válida deve permanecer isolada de qualquer outra operação, ou seja, garantimos que a transação não será interferida por nenhuma outra transação.

Durabilidade: Dados validados são registrados pelo sistema de tal forma que mesmo no caso de uma falha e/ou reinício do sistema, os dados estão disponíveis em seu estado correto.

2.3 Diagrama de Use Case

O Diagrama de Caso de Uso (Use Case), vai definir uma sequência de ações que são executadas pelo sistema, as quais irão gerar um resultado de valor notável para um ator em particular. Assim, o aspecto comportamental de um sistema a ser desenvolvido pode ser exposto pelo devido diagrama (SANTANDER, et al, 2019).

De acordo com Aline Santos, et al, (2006, apud MEDEIROS, 2005), o Diagrama de Caso de Uso foi proposto por Ivar Jacobson, em sua metodologia de desenvolvimento de sistemas orientados à objetos, posteriormente tornando seu uso uma prática frequente na identificação destes mesmos sistemas. Neste contexto o Caso de Uso descreve um comportamento que o software a ser desenvolvido apresentará quando estiver pronto.

Tendo em vista isso, os inúmeros Diagramas de Caso de Uso sempre estarão associados a um determinado modelo, cada um expondo um subconjunto de elementos dentro destes respectivos exemplos que contém uma relevância cada qual para um determinado objetivo.

2.4 Diagramas de Entidades e Relacionamentos (DER)

Um diagrama entidade relacionamento (ER) é um tipo de fluxograma que ilustra como “entidades”, pessoas, objetos ou conceitos, que se relacionam entre si dentro de um sistema (Nogueira, 1988). De acordo com Kawbata, et al (2015, apud Genong, 2010), diagramas ER são mais utilizados para projetar ou depurar bancos de dados relacionais nas áreas de Engenharia de Software, sistemas de informações empresariais, educação e pesquisa.

Com isso, os Diagramas ER estão relacionados com diagramas de estrutura de dados, que incidem sobre as relações de elementos dentro de entidades ao invés de relações com as próprias entidades. Eles espelham estruturas gramaticais, onde entidades são substantivos e relacionamentos são verbos (SILVA, et al.,2008).

Em síntese, o MER é um tipo de modelagem conceitual, que busca representar de forma abstrata, os objetos de um domínio de negócios, descrevendo as suas características o relacionamento entre as respectivas entidades.

2.5 Modelo Relacional (MR)

O Modelo Relacional (MR), é um modelo de dados representativo, o qual fundamenta-se em conceitos da matemática, os primeiros sistemas comerciais baseados no MR foram disponibilizados em 1980 e desde então vem sendo implementado em diversos sistemas (COSTA, 2015).

O Modelo Relacional é caracterizado pela organização das conexões dos dados, ou seja, são tabelas, nas quais são constituídas por colunas e linhas, onde se encontram informações de um determinado assunto de maneira organizada.

Segundo, Macário et al (2018), uma relação consiste de um esquema e de uma instância. O esquema especifica o nome da relação e o nome e o domínio de cada coluna, também denominada atributo ou campo da relação. O domínio do atributo é referenciado no esquema por seu nome e serve para restringir os valores que este atributo pode assumir.

2.6 SQLite Browser

De acordo com Elisângela Costa (2017, apud DATE, 2004, p. 71), a Structured Query Language (SQL) ou Linguagem de Consulta Estruturada foi criada pela IBM Research, no início da década de 1970, para o protótipo de um sistema de banco de dados chamado System R. A linguagem SQL hoje é reconhecida como um padrão para Sistemas Gerenciadores de Bancos de Dados Relacionais, sendo mais compreensível do que outras linguagens.

Apesar de conhecida como sendo uma “linguagem de consulta”, a SQL disponibiliza aos usuários recursos para definir a estrutura dos dados, atualizar, incluir, excluir e alterar dados, especificar restrições de integridade e outros recursos dentro da plataforma (COSTA apud SILBERSCHATZ; KORTH; SUDARSHAN, 1999, p. 109).

3. Metodologia

Nesta seção apresentam-se os métodos (preparação e planejamento) referentes à execução do estudo, para de forma objetiva, poder argumentar a maneira como foi constituído o presente trabalho.

Para execução correta da pesquisa, foram realizadas 3 etapas para compreender e analisar minuciosamente o funcionamento das tarefas cotidianas dentro da empresa e o relacionamento entre os seus respectivos membros. Iniciou-se esse estudo realizando o Diagrama de Caso de Uso (USE CASE), a fim de expressar de maneira prática os agentes presentes na questão e as ações realizadas por eles, após isso, foi realizado o Diagrama de Entidade e Relacionamento (DER), esquematizando como acontece o relacionamento entre as entidades mencionadas, para que a partir dessas análises feitas pudesse dar início à criação do banco de dados utilizando o *software SQLite*, a Figura 01 demonstra as etapas executadas neste trabalho.

Figura 01- Diagrama da metodologia



Fonte: Autores (2023)

O presente estudo apresenta uma pesquisa aplicada, cujo principal objetivo é gerar conhecimentos para aplicação prática e clara, referentes às melhorias e otimização para os interesses da organização.

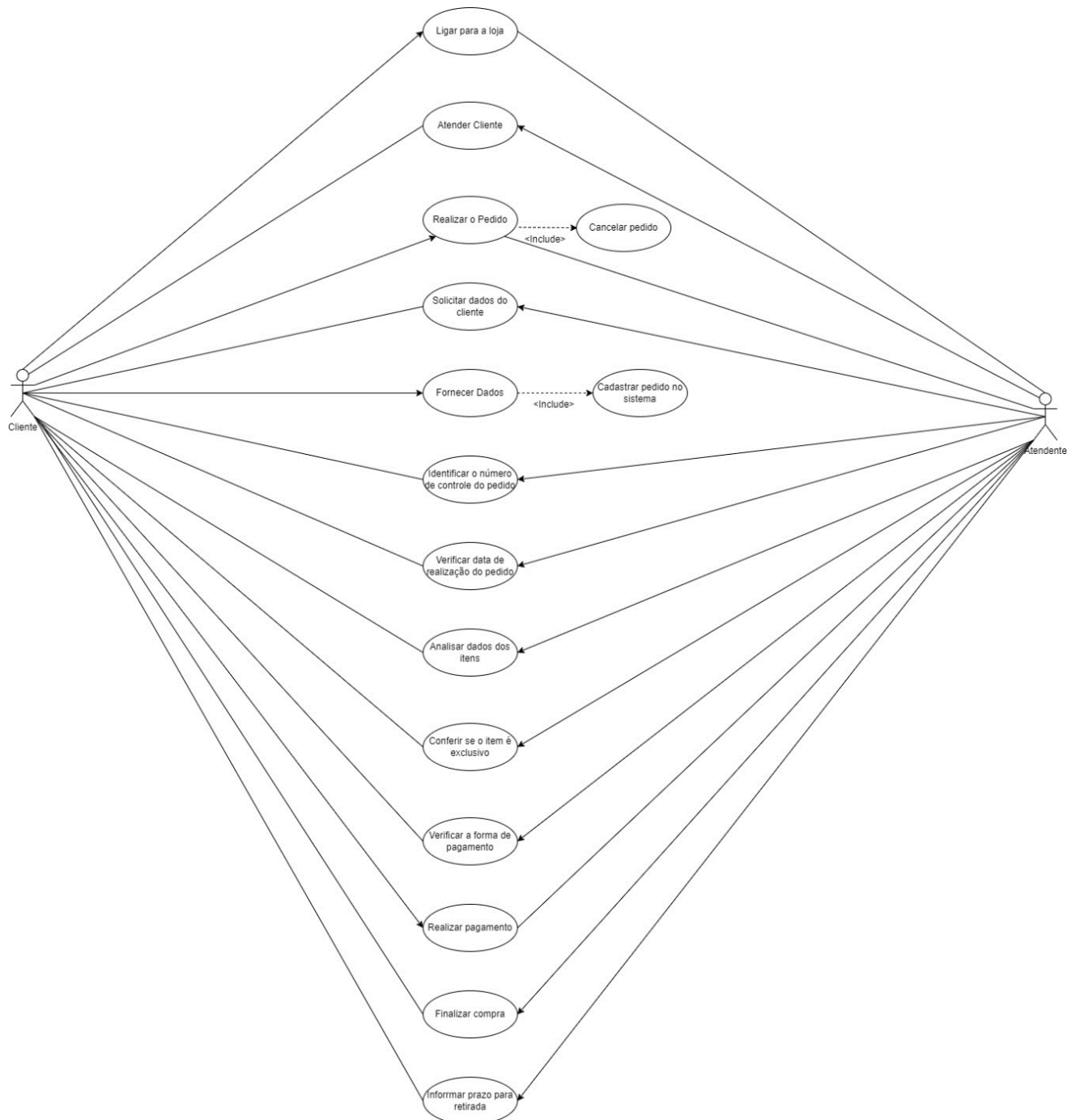
4. Resultados e Discussões

4.1 Objetivos e funcionalidades do sistema

O sistema tem como objetivo criar um banco de dados seguido de uma interface elaborada na linguagem de programação *Python* que visa facilitar o armazenamento de dados, controle de informações, manipulação de registros dos clientes e itens da loja de roupas. Assim, o banco de dados será criado por meio do *SQLite Browser*. Dessa forma, através de consultas, é possível visualizar informações referentes ao cliente e ao item adquirido, além de relacionar tabelas com seus respectivos atributos que viabilizam a interação da possível compra realizada. Nesse contexto, funciona para cadastrar clientes no sistema, identificar o pedido solicitado e determinar a categoria que o item pertence.

4.2 Diagrama de Use Case

Figura 02 - Diagrama de Use Case



Fonte: Autores (2023)

Atores: Cliente, atendente.

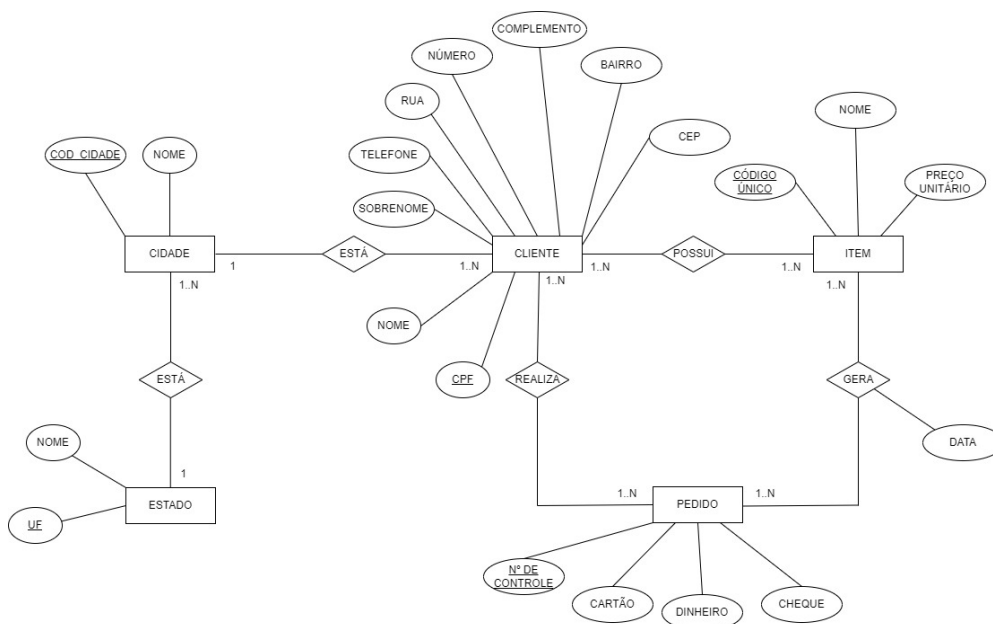
Ações:

- Cliente: ligar para a loja, realizar pedido, cancelar pedido, fornecer dados, realizar pagamento;
- Atendente: atender cliente, solicitar dados do cliente, cadastrar pedido no sistema, identificar o número de controle do pedido, verificar data de realização do pedido, analisar dados dos itens, conferir se o item é exclusivo, verificar a forma de pagamento, finalizar compra, informar prazo para retirada.

De acordo com a Figura 02, pode-se observar como ocorre o Diagrama de Use Case do sistema de informações da organização, a forma como as ações do mesmo é realizada. O modelo apresenta a sequência de ações que são realizadas dentro da empresa entre os atores envolvidos. A esquematização conta com 2 (dois) atores e uma série de ações interligadas entre si.

4.4 Diagrama de Entidade e Relacionamento

Figura 03 - Diagrama de Entidade e Relacionamento



Fonte: Autores (2023)

- Estado (Sigla, nome);
- Cidade (Cod_cidade, nome, sigla);
- Cliente (CPF, nome, sobrenome, telefone, rua, número complemento, bairro, CEP, Cod_cidade);
- Item (Cod_unico, nome, preço_unitário, CPF);

- Pedido (n_controle, cartão, dinheiro, cheque, data, Cod_unico).

De acordo com a Figura 03, é possível observar como ocorre Diagrama de Entidade e Relacionamento da empresa, a forma como os mesmos estão distribuídos. O protótipo da esquematização é composto por 5 (cinco) entidades, 5 (cinco) relacionamentos e os atributos interligados em cada item necessário. Contudo, pode-se observar que na entidade cliente o CPF atua como chave primária, na entidade pedido é número de controle, no item sua chave primária é código único, na entidade cidade é código da cidade e na entidade estado a chave é a sigla.

4.5 Desenvolvimento do Sistema de Banco de Dados

Para a criação do banco de dados foi desenvolvido um código de criação de tabelas e suas respectivas restrições, assim como o tipo de dado que cada restrição iria comportar, é possível observar na Figura 04, a seguir.

Figura 04 – Código para o banco de dados no SQL

```
CREATE TABLE "Cliente" (  
    "CPF" INTEGER NOT NULL,  
    "nome" TEXT NOT NULL,  
    "sobrenome" TEXT NOT NULL,  
    "telefone" INTEGER,  
    "rua" TEXT,  
    "número" INTEGER,  
    "complemento" TEXT,  
    "bairro" TEXT,  
    "CEP" INTEGER,  
    "cod_cidade" INTEGER,  
    PRIMARY KEY ("CPF")  
);  
  
CREATE TABLE "Estado" (  
    "Sigla" TEXT,  
    "nome" TEXT NOT NULL,  
    "cod_cidade" INTEGER,  
    PRIMARY KEY ("Sigla")  
);  
  
CREATE TABLE "Cidade" (  
    "nome" TEXT NOT NULL,  
    "sigla" TEXT,  
    PRIMARY KEY ("sigla")  
);  
  
CREATE TABLE "Pedido" (  
    "n_controle" INTEGER NOT NULL,  
    "cartão" TEXT,  
    "dinheiro" TEXT,  
    "cheque" TEXT,  
    "data" INTEGER,  
    "cod_unico" INTEGER,  
    PRIMARY KEY ("n_controle")  
);  
  
CREATE TABLE "Item" (  
    "cod_unico" INTEGER NOT NULL,  
    "nome" TEXT,  
    "preco_unitario" NUMERIC,  
    "CPF" INTEGER,  
    PRIMARY KEY ("cod_unico")  
);  
  
CREATE TABLE "CidadeCliente" (  
    "cod_cidade" INTEGER UNIQUE,  
    "CPF" INTEGER UNIQUE,  
    FOREIGN KEY ("cod_cidade") REFERENCES "Cidade" ("cod_cidade"),  
    FOREIGN KEY ("CPF") REFERENCES "Cliente" ("CPF"),  
    PRIMARY KEY ("cod_cidade", "CPF")  
);
```

```
CREATE TABLE "ClienteItem" (
  "CPF" INTEGER UNIQUE,
  "cod_unico" INTEGER UNIQUE,
  FOREIGN KEY ("CPF") REFERENCES "Cliente" ("CPF"),
  FOREIGN KEY ("cod_unico") REFERENCES "Item" ("cod_unico"),
  PRIMARY KEY ("CPF", "cod_unico")
);

CREATE TABLE "PedidoItem" (
  "cod_unico" INTEGER UNIQUE,
  "n_controle" INTEGER UNIQUE,
  FOREIGN KEY ("cod_unico") REFERENCES "Pedido" ("cod_unico"),
  FOREIGN KEY ("n_controle") REFERENCES "Item" ("n_controle"),
  PRIMARY KEY ("cod_unico", "n_controle")
);
```

Fonte: Autores (2023)

4.6 Testes do Banco de Dados

Após realizar o desenvolvimento do sistema de banco de dados, foram realizados alguns testes a fim de verificar se o mesmo estava funcionando corretamente. Diante disso, a primeira etapa realizada foi a adição da coluna “CREDIÁRIO” do tipo texto, na tabela “PEDIDO”, como pode ser observado na Figura 05 e 06.

Figura 05 – Antes da alteração

ALTER TABLE Pedido ADD Crediario TEXT		
▼ Pedido		CREATE TABLE "Pedido" ("n_controle" INTEGER, "cartão" TEXT, "dinheiro" TEXT, "cheque" TEXT, "data" INTEGER, "co
n_controle	INTEGER	"n_controle" INTEGER
cartão	TEXT	"cartão" TEXT
dinheiro	TEXT	"dinheiro" TEXT
cheque	TEXT	"cheque" TEXT
data	INTEGER	"data" INTEGER
cod_unico	INTEGER	"cod_unico" INTEGER

Fonte: Autores (2023)

Figura 06 – Resultado da alteração

▼ Pedido		CREATE TABLE "Pedido" ("n_controle" INTEGER, "cartão" TEXT, "dinheiro" TEXT, "cheque" TEXT, "data" INTEGER, "co
n_controle	INTEGER	"n_controle" INTEGER
cartão	TEXT	"cartão" TEXT
dinheiro	TEXT	"dinheiro" TEXT
cheque	TEXT	"cheque" TEXT
data	INTEGER	"data" INTEGER
cod_unico	INTEGER	"cod_unico" INTEGER
Crediario	TEXT	"Crediario" TEXT

Fonte: Autores (2023)

O próximo teste realizado consistiu em inserir dados em todas as tabelas de banco de dados. Com isso, as tabelas cliente, cidade, estado, pedido, item, cidadecliente, clienteitem e pedidoitem, estão demonstradas nas Figuras 07, 08, 09, 10, 11, 12, 13, 14 a seguir.

Figura 07- Adição de dados na tabela “CLIENTE”

```
INSERT INTO Cliente (Nome,Sobrenome, Telefone, Rua, Numero, Complemento, Bairro, CEP, Cod_cidade)
VALUES (15263548599, 'Carla', 'Ramos de Freitas', 83996562545, 'Rua Rafael de Freitas', 286, 'Terceiro andar', 'Centro', 56800000, 1)

INSERT INTO Cliente (Nome,Sobrenome, Telefone, Rua, Numero, Complemento, Bairro, CEP, Cod_cidade)
VALUES (12290456817, 'Leonardo', 'do Nascimento Costa', 83999458216, 'Travessa Tenente Severino Espinhara', 107, 'Não existe', 'Centro', 58500000, 2)
```



	CPF	nome	sobrenome	telefone	rua	número	complemento	bairro	CEP	cod_cidade
	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro
1	12290456817	Leonardo	do Nascimento Costa	83999458216	Travessa Tenente Severino Espinhara	107	Não existe	Centro	58500000	2
2	15263548599	Carla	Ramos de Freitas	83996562545	Rua Tenório Magalhães	286	Terceiro andar	Centro	56800000	1

Fonte: Autores (2023)

Figura 08 - Adição de dados na tabela “CIDADE”

```
INSERT INTO Cidade (Nome,Sigla)
VALUES ('Afogados da Ingazeira', 'Af')

INSERT INTO Cidade (Nome,Sigla)
VALUES ('Monteiro', 'MO')
```

nome	sigla
Filtro	Filtro
Afogados da Ingazeira	Af
Monteiro	MO

Fonte: Autores (2023)

Figura 09 - Adição na tabela “ESTADO”

```
INSERT INTO Estado (Sigla,Nome,Cod_cidade)
VALUES ('PE', 'Pernambuco', 1)

INSERT INTO Estado (Sigla,Nome,Cod_cidade)
VALUES ('PB', 'Paraíba', 2)
```

Sigla	nome	cod_cidade
Filtro	Filtro	Filtro
PE	Pernambuco	1
PB	Paraíba	2

Fonte: Autores (2023)

Figura 10 - Adição na tabela “PEDIDO”

```
INSERT INTO Pedido(n_controle,cod_unico,data,cartão,dinheiro,crediciario)
VALUES (111,1,'11/10','SIM','NÃO','NÃO','NÃO')

INSERT INTO Pedido(n_controle,cod_unico,data,cartão,dinheiro,crediciario)
VALUES (222,2,'12/10','NÃO','SIM','NÃO','NÃO')
```

n_controle	cod_unico	data ▼1	cartão	dinheiro	cheque	Crediciario
Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro
111	1	11/10	SIM	NÃO	NÃO	NÃO
222	2	12/10	NÃO	SIM	NÃO	NÃO

Fonte: Autores (2023)

Figura 11- Adição na tabela “ITEM”

```
INSERT INTO Item(Cod_unico,nome,preco_unitario,CPF)
VALUES (01,'Blusa', '60,00', 15263548599)

INSERT INTO Item(Cod_unico,nome,preco_unitario,CPF)
VALUES (02,'Calça', '130,00', 12290456817)
```

cod_unico	nome	preco_unitario	CPF
Filtro	Filtro	Filtro	Filtro
1	Blusa	60,00	15263548599
2	Calça	130,00	12290456817

Fonte: Autores (2023)

Figura 12- Adição de dados na tabela “CIDADECLIENTE”

```
INSERT INTO CidadeCliente(Cod_cidade,CPF)
VALUES (1, 15263548599)

INSERT INTO CidadeCliente(Cod_cidade,CPF)
VALUES (2, 12290456817)
```

cod_cidade	CPF
Filtro	Filtro
1	15263548599
2	12290456817

Fonte: Autores (2023)

Figura 13 - Adição de dados na tabela “CLIENTEITEM”

```
INSERT INTO ClienteItem(CPF,Cod_unico)
VALUES (15263548599, 1)

INSERT INTO ClienteItem(CPF,Cod_unico)
VALUES (12290456817, 2)
```

CPF	cod_unico
Filtro	Filtro
15263548599	1
12290456817	2

Fonte: Autores (2023)

Figura 14 - Adição de dados na tabela “PEDIDOITEM”

```
INSERT INTO PedidoItem(cod_unico,n_controle)
VALUES (1,111)

INSERT INTO PedidoItem(cod_unico,n_controle)
VALUES (2,222)
```

cod_unico	n_controle
Filtro	Filtro
1	111
2	222

Fonte: Autores (2023)

Em seguida, buscou-se realizar uma consulta acerca do nome do cliente, que está localizado na tabela “CLIENTE”, como pode ser observado na Figura 15.

Figura 15 - Consulta do nome

```

1 SELECT nome
2 FROM Cliente
3 WHERE CPF = 12290456817
4
5

```

	nome
1	Leonardo

Fonte: Autores (2023)

Posteriormente, foram consultados valores máximos e mínimos do atributo “N_CONTROLE” da tabela “PEDIDO” e como resultado obteve-se que o máximo valor do código presente especificamente nessa tabela é 222 e o mínimo é 111. A Figura 16 expõe detalhadamente a consulta realizada.

Figura 16 - Consulta do valor máximo e mínimo

```

1 SELECT MIN(n_controle),MAX(n_controle)
2 FROM Pedido
3
4

```

	MIN(n_controle)	MAX(n_controle)
1	111	222

Fonte: Autores (2023)

A próxima consulta efetuada tem como objetivo mostrar os valores do preço unitário de cada item em ordem crescente e, contudo, utilizou-se o comando apresentado na Figura 17 para conseguir demonstrar esse número de acordo com a sequência do maior para o menor.

Figura 17- Consulta de ordem crescente

```

1 SELECT preco_unitario
2 FROM Item
3 ORDER BY preco_unitario ASC
4
5
6

```

	preco_unitario
1	130,00
2	40,00

Fonte: Autores (2023)

Após isso, foram realizadas alterações nos dados cadastrados da tabela “CLIENTE”, como mostram as Figuras 18 e 19.



Figura 18 - Antes da alteração

	CPF	nome	sobrenome	telefone	rua	número	complemento	bairro	CEP	cod_cidade
	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro
1	12290456817	Leonardo	do Nascimento Costa	83999458216	Travessa Tenente Severino Espinhara	107	Não existe	Centro	58500000	2
2	15263548599	Carla	Ramos de Freitas	83996562545	Rua Tenório Magalhães	286	Terceiro andar	Centro	56800000	1

Fonte: Autores (2023)

Figura 19 - Após a alteração

```
UPDATE Cliente
SET rua = 'Rua Rafael de Freitas'
WHERE cod_cidade = 1
```

	CPF	nome	sobrenome	telefone	rua	número	complemento	bairro	CEP	cod_cidade
	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro
1	12290456817	Leonardo	do Nascimento Costa	83999458216	Travessa Tenente Severino Espinhara	107	Não existe	Centro	58500000	2
2	15263548599	Carla	Ramos de Freitas	83996562545	Rua Rafael de Freitas	286	Terceiro andar	Centro	56800000	1

Fonte: Autores (2023)

Em seguida, a mesma operação se deu para a tabela “ITEM”, as Figuras 20 e 21 expressam como ficou o resultado.

Figura 20 - Antes da alteração

cod_unico	nome	preco_unitario	CPF
Filtro	Filtro	Filtro	Filtro
1	Blusa	60,00	15263548599
2	Calça	130,00	12290456817

Fonte: Autores (2023)

Figura 21 - Após a alteração

```
UPDATE Item
SET preco_unitario = '40,00'
WHERE cod_unico = 1
```

cod_unico	nome	preco_unitario	CPF
Filtro	Filtro	Filtro	Filtro
1	Blusa	40,00	15263548599
2	Calça	130,00	12290456817

Fonte: Autores (2023)

Após isso, uma linha da tabela “PEDIDO” foi deletada, como se observa nas Figuras 22 e 23.

Figura 22 - Antes de deletar

n_controle	cod_unico	data ▼ ¹	cartão	dinheiro	cheque	Crediarior
Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro
111	1	11/10	SIM	NÃO	NÃO	NÃO
222	2	12/10	NÃO	SIM	NÃO	NÃO

Fonte: Autores (2023)

Figura 23 - Após deletar

```
DELETE FROM Pedido
WHERE n_controle = 222
```

n_controle	cod_unico	data ▼ ¹	cartão	dinheiro	cheque	Crediarior
Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro
111	1	11/10	SIM	NÃO	NÃO	NÃO

Fonte: Autores (2023)

Para finalizar os testes do sistema elaborado, realizou-se uma consulta envolvendo as tabelas: cliente, pedido e item. Dessa forma, foram consultados registros, como: CPF, número de controle e preço unitário.

Figura 24 - Resultados das consultas

```
1 SELECT CEP, n_controle, preco_unitario
2 FROM Cliente, Pedido, Item
3
4
5
6
```

	CEP	n_controle	preco_unitario
1	58500000	111	40,00
2	58500000	111	130,00
3	58500000	222	40,00
4	58500000	222	130,00
5	56800000	111	40,00

Fonte: Autores (2023)

5. Considerações Finais

Os sistemas de gestão de base de dados são essenciais para o desenvolvimento e crescimento das organizações. Através da sua utilização, é possível gerar consultas, relatórios e, acima de tudo, aceder rapidamente às informações de forma clara. Com base nisso, este projeto desenvolveu um sistema para criar e administrar a base de dados de uma loja de roupas, utilizando *SQLite Browser*.

Os objetivos deste estudo foram alcançados, uma vez que a estruturação do sistema de gestão de base de dados permite aos proprietários da loja, uma maior segurança nos dados armazenados e um sistema mais organizado. Além disso, a aplicação prática dos conceitos aprendidos em sala de aula resultou num maior enriquecimento do conhecimento, devido à experiência adquirida ao lidar com problemas reais durante o desenvolvimento da base de dados.

Portanto, a criação da respectiva base de dados juntamente com a interface em *Python* que foi desenvolvida após todo o processo de elaboração do banco de dados, visa armazenar e organizar esses registros em questão, de uma maneira mais prática e segura. Dessa forma, possibilitar também, a automatização e otimização da emissão de informações a serem repassadas para seus usuários e controle interno.



REFERÊNCIAS

BALDO, Carla. **O Modelo Relacional**. São Paulo: UFC, 2018. Acesso em: 06 de outubro de 2023.

COMACHIO, Vanderson. **FUNCIONAMENTO DE BANCO DE DADOS EM ANDROID: UM ESTUDO EXPERIMENTAL UTILIZANDO SQLITE**. Universidade Tecnológica Federal do Paraná. Acesso em: 06 de outubro de 2023.

CORONEL, Carlos. **Sistema de Banco de Dados**. São Paulo. Acesso em: 06 de outubro de 2023.

COSTA, Elisângela. **BANCOS DE DADOS RELACIONAIS**. Faculdade de Tecnologia de São Paulo, 2011. Acesso em: 06 de outubro de 2023.

FRANK, Kewry; PEREIRA, Robson; FILHO, Jerônimo. **Diagrama Entidade-Relacionamento: uma ferramenta para modelagem de dados conceituais em Engenharia de Software**. Research, Society and Development, v. 10. Acesso em: 06 de outubro de 2023.

GARCIA, Vinícius; SOTTO, EDER. COMPARATIVO ENTRE OS MODELOS DE BANCO DE DADOS RELACIONAL E NÃORELACIONAL. **Faculdade de Tecnologia de Catanduva (FATEC) –SP –Brasil**. Acesso em: 06 de outubro de 2023.

MATSUMOTO, Cristina. **A importância do Banco de Dados em uma Organização**. Maringá Management: Revista de Ciências Empresariais, v. 3, n.1 - p.45-55, jan./jun. 2006. Acesso em: 06 de outubro de 2023.

NETO, Rafael. **AprenDER: Ferramenta de Apoio à Construção de Diagrama Entidade Relacionamento para Deficientes Visuais**. Instituto Federal Fluminense, Rio de Janeiro. Acesso em: 06 de outubro de 2023.

SANTOS, Aline; RADAEL, Edilson; PINHATA, Rosimeire; ROSSI, Sandreliz; MENDES, Sandro. **Diretrizes a construção de caso de uso eficazes**. Faculdade de Filosofia, Ciência e Letras- Centro Universitário Fundação Santo André. Acesso em: 06 de outubro de 2023.

VERONA, Ana Carolina; CAMPOS, Maria Luiza; BAIÃO, Fernanda. **LGPD em Ambientes de Banco de Dados nas Organizações**. VI Escola Regional de Sistemas de Informação. Acesso em: 06 de outubro de 2023.