



UNIVERSIDADE FEDERAL DA GRANDE DOURADOS
Faculdade de Ciências Exatas e Tecnologias - FACET

**GEILSO FARIA RODRIGUES DA
SILVA E
RODRIGO QUINHONE VEIGA DURAIS**

TRABALHO PRÁTICO
**BANCO DE DADOS E LABORATÓRIO DE BANCO DE
DADOS**

**Dourados -
MS 2022**

Resumo:

Relatório técnico-científico, sobre a elaboração de um pequeno sistema de informação usando conceitos de banco de dados, com o uso da plataforma *MySQL Workbench*, nas disciplinas de Banco de dados e Laboratório de Banco de dados, ministradas pela professora Silvana Morita Melo.

SUMÁRIO

1.0- Introdução.....	4
2.0 - Desenvolvimento.....	5
2.0.0 - Levantamento de Requisitos.....	5
2.0.1- Descrição do problema.....	5
2.1.0 - Modelagem.....	5
2.1.1 - Modelo de entidades e relacionamentos.....	6
2.1.2 - Dicionário de dados.....	6
2.2.0 - Projeto.....	10
2.2.1 - Conceito Lógico(Modelo Relacional).....	11
2.3.0 - Implementação e Testes.....	12
2.4.0 - Carga horária por integrante	17
3.0 - Conclusão.....	18
4.0 - Bibliografia.....	18

1.0-INTRODUÇÃO

Segundo Elmasri e Navathe (2010) “ Os bancos de dados e os sistemas de bancos de dados se tornaram componentes essenciais no cotidiano da sociedade moderna. No decorrer do dia, a maioria de nós se depara com atividades que envolvem alguma interação com os bancos de dados ”. Tal fato pode ser explicado devido ao crescente volume de informações experimentado nas últimas décadas em nossa sociedade, fazendo com que espaços físicos como salas, gavetas e armários não sejam suficientes para atender a atual demanda de informações. Em resposta a esse problema surgem os sistemas de banco de dados e sistemas gerenciadores de dados(SGBDs), que propõem o armazenamento de informações no formato digital, economizando tempo e espaço físico.

Eventualmente, praticar sobre os conceitos de bancos de dados é extremamente necessário. Para tal, este relatório mostra em etapas os processos de construção de um banco de dados digital, desde a etapa de levantamento de requisitos até o momento de implementação e testes no SGBD *MySQL Workbench*. Mais adiante o leitor poderá observar a enorme importância do uso de SGBDs, como por exemplo a praticidade de armazenamento de informações.

Este relatório consiste em apresentar um esquema de Banco de Dados de uma loja de confecções e, para um melhor desenvolvimento deste projeto, a metodologia se baseou na divisão em quatro etapas, sendo elas:

1. **Levantamento de requisitos:** Etapa na qual todas as informações que representavam algum aspecto do mundo real foram selecionadas, posteriormente o problema foi transcrito de maneira textual com o auxílio dos dados obtidos;
2. **Modelagem:** Nesta fase todos os dados obtidos na fase de levantamento de requisitos foram transcritos para um modelo entidades e relacionamentos;
3. **Projeto:** Neste momento foi desenvolvido um diagrama de projeto na plataforma *MySQL Workbench* com o auxílio das informações das fases anteriores;
4. **Fase de implementação e testes:** Fase na qual o banco de dados em si foi criado, testado e modificado.

Logo, para facilitar a compreensão do leitor, o desenvolvimento deste relatório foi dividido também em quatro etapas, com cada etapa correspondendo a uma parte do projeto (levantamento de requisitos, modelagem, projeto e a fase de implementação e testes). Cada capítulo mostra em detalhes as etapas de criação, as dificuldades e os aprendizados obtidos.

2.0- DESENVOLVIMENTO

Antes de mais nada, é importante salientar ao leitor que, alguns erros de projeto foram corrigidos apenas em fases posteriores da citada, entretanto, para uma melhor compreensão sobre a elaboração de um banco de dados corretamente, cada fase contará com sua descrição correta mesmo se a etapa citada ainda estava descrita erroneamente.

2.0.0 - Levantamento de Requisitos:

Inicialmente, foi preciso definir sobre qual problema este projeto iria solucionar e os requisitos necessários para tal. Após conversas entre os membros foi decidido que o problema a ser solucionado seria a construção de um banco de dados de uma pequena loja de confecções, pois, este ambiente se mostrou propício para testar muitos conceitos de banco de dados (relacionamentos, cardinalidades, consultas, entre outros). Inicialmente, foi identificado que os requisitos necessários seriam informações sobre a loja, funcionários, clientes, vendas, peças e um controle de vendas.

Em seguida, algumas dúvidas começaram a surgir como: O que é realmente necessário para o desenvolvimento do projeto e o que foge do contexto de solucionar este problema? Quais informações cada componente no banco de dados teria? Quais informações são armazenadas em uma loja de confecções?.

2.0.1- Descrição do problema

Posteriormente, chegou-se na seguinte descrição textual do problema:

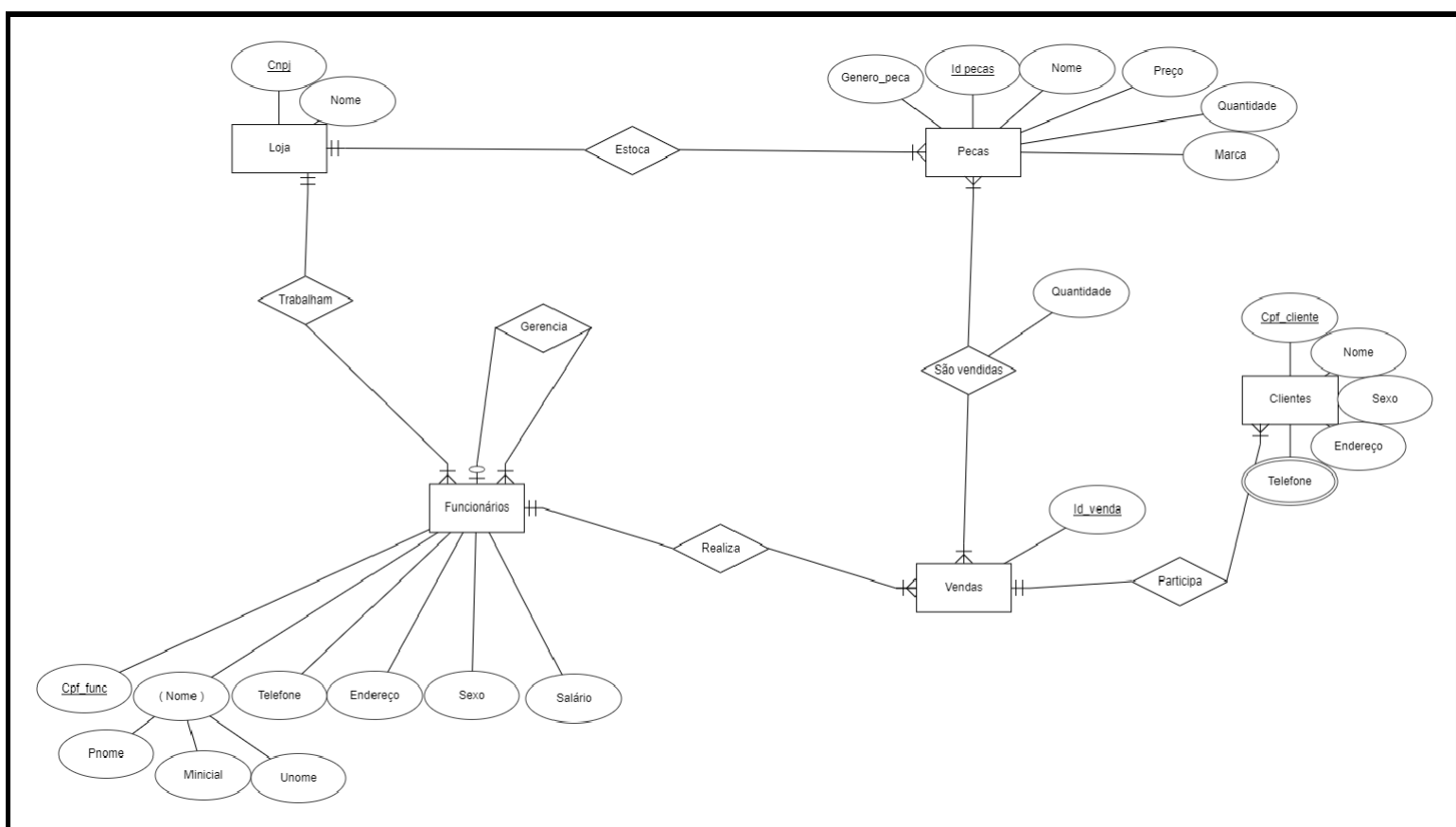
“Uma loja de confecções decide armazenar detalhes de seus clientes, funcionários, peças e vendas em um banco de dados. Esta loja possui um CNPJ e um nome. Trabalham nesta loja vários funcionários e cada um deles tem seu CPF, nome, sexo, salário, endereço e o telefone armazenado no banco de dados. Todos os funcionários possuem um gerente como seu superior. Nessa loja há um estoque para armazenamento das peças e cada peça registra um código, uma marca, seu respectivo preço, gênero (masculina, feminina ou unissex), a quantidade disponível e seu nome (camisa, calça, blusa, etc). Neste banco de dados também é armazenado as vendas realizadas. Como também, as informações pessoais dos clientes, sendo eles: CPF, nome, sexo, endereço e o telefone, que pode ser mais de um (ex: residencial, pessoal, comercial). E vale ressaltar, que cada venda realizada recebe consigo um identificador, além de informações sobre vendedor e clientes. A cada venda realizada mais de uma peça pode ser vendida”.

Nesta descrição cada entidade participante do relacionamento (loja, funcionários, clientes, peças, vendas) possuem uma chave identificadora (CNPJ, CPF do funcionário, código da peça, código da venda) respectivamente, e conta com informações adicionais importantes para manter-se uma consistência no banco de dados como nome, telefone, endereço, entre outros.

2.1.0 - Modelagem

2.1.1 - Modelo de entidades e relacionamentos.

Logo após, todas as informações obtidas nas fases anteriores foram transformadas em modelo de entidades e relacionamentos. Apesar de não ser uma fase ainda de alto nível, algumas informações contribuíram para o desenvolvimento do projeto, fato este que pode ser explicado, por ser um diagrama esquemático e facilita a visualização inicial do projeto. Ademais, nesse momento, algumas informações adicionais foram obtidas, tais como a cardinalidade dos relacionamentos, os relacionamentos que cada entidade fazia e a classe de alguns atributos (se eram compostos, multivalores, simples, etc). Vale lembrar, que tal modelo de entidade relacionamento a seguir passou por modificações ao longo do desenvolvimento do projeto até chegar ao seguinte diagrama entidade relacionamento:



Fonte: Geilso Faria e Rodrigo Quinhone - Modelo Entidade Relacionamento

2.1.2 - Dicionário de dados

Outro fator importante foi o desenvolvimento de um dicionário de dados que se trata de uma coleção de nomes, atributos e definições sobre elementos de dados que estão sendo usados no relatório e que é capaz de fornecer uma visão mais ampla sobre os dados em si, fornecendo mais detalhes do comportamento e descrição dos dados.

Dicionário de dados da loja				
Nome do Campo	Chave	Tipo de dado	Descrição do Campo	Tamanho do campo
CNPJ	Primária	varchar	Cadeia de caracteres que representam o identificador da loja	14
nome_loja		varchar	Cadeia de caracteres que representam o nome da loja	15

Dicionário de dados do cliente				
Nome do Campo	Chave	Tipo de dado	Descrição do Campo	Tamanho do campo
cpf_cliente	Primária	varchar	Cadeia de caracteres que representam o identificador do cliente	11
nome		varchar	Cadeia de caracteres que representam o nome do cliente	40
endereço		varchar	Cadeia de caracteres que representam o endereço do cliente	50
sexo		char	Caractere que representam o sexo do cliente e pode ser apenas 'F' ou 'M'	1

Dicionário de dados do funcionário				
Nome do Campo	Chave	Tipo de dado	Descrição do Campo	Tamanho do campo
cpf	Primária	varchar	Cadeia de caracteres que representam o identificador do funcionário .	11
Cpf_gerente		varchar	Cadeia de caracteres que representam o identificador do gerente.	11
Pnome		varchar	Cadeia de caracteres que representa o primeiro nome do funcionário.	10
Minicial		char	Atributo que representa o primeiro caractere do segundo nome do funcionário.	1
Unome		varchar	Cadeia de caracteres que representa o último nome do funcionário.	10
Endereço		varchar	Cadeia de caracteres que representam o endereço do funcionário.	50
Sexo		char	Caractere que representam o sexo do funcionário e pode ser apenas 'F' ou 'M'.	1

Salário		decimal	Valor que informa o salário do funcionário.	6,2
---------	--	---------	--	-----

Telefone		varchar	Uma cadeia de caracteres que representa o número de telefone do funcionário.	15
Loja_CNPJ	FK	varchar	Cadeia de caracteres que representam o identificador da loja	14

Dicionário de dados das peças

Nome do Campo	Chave	Tipo de dado	Descrição do Campo	Tamanho do campo
Cod_produto	Primária	inteiro	Cadeia de caracteres que representam o identificador da peça.	
tipo_peça		varchar	Cadeia de caracteres que representa de que peça se trata (blusa, calça, short, etc).	30
Marca		varchar	Cadeia de caracteres que representa a marca da peça em questão,	15
Valor		Decimal	Valor que representa o preço da peça em questão.	6,2
quant		inteiro	Atributo multivalorado. Uma cadeia de caracteres que representam um ou mais telefones dos clientes	
genero_peça		char	Caractere que representa o gênero da peça podendo ser M=masculino, F=feminino e U=Unisex	1
CNPJ_loja	FK	varchar	Cadeia de caracteres que representam o identificador da loja	14

Dicionário de dados do telefone dos clientes

Nome do Campo	Chave	Tipo de dado	Descrição do Campo	Tamanho do campo
Telefone	Primária	varchar	Cadeia de caracteres que representam o número de telefone dos clientes.	15
cpf_cliente	FK	varchar	Cadeia de caracteres que representam o identificador do cliente.	11

Dicionário de dados das vendas

Nome do Campo	Chave	Tipo de dado	Descrição do Campo	Tamanho do campo
cod_venda	Primária	inteiro	Número inteiro que representa o código de cada venda	
cpf_cliente			Cadeia de caracteres que representam o identificador do cliente.	

	FK	varchar		11
--	----	---------	--	----

cpf_vendedor	FK	varchar	Cadeia de caracteres que representam o identificador do funcionário .	11
--------------	----	---------	---	----

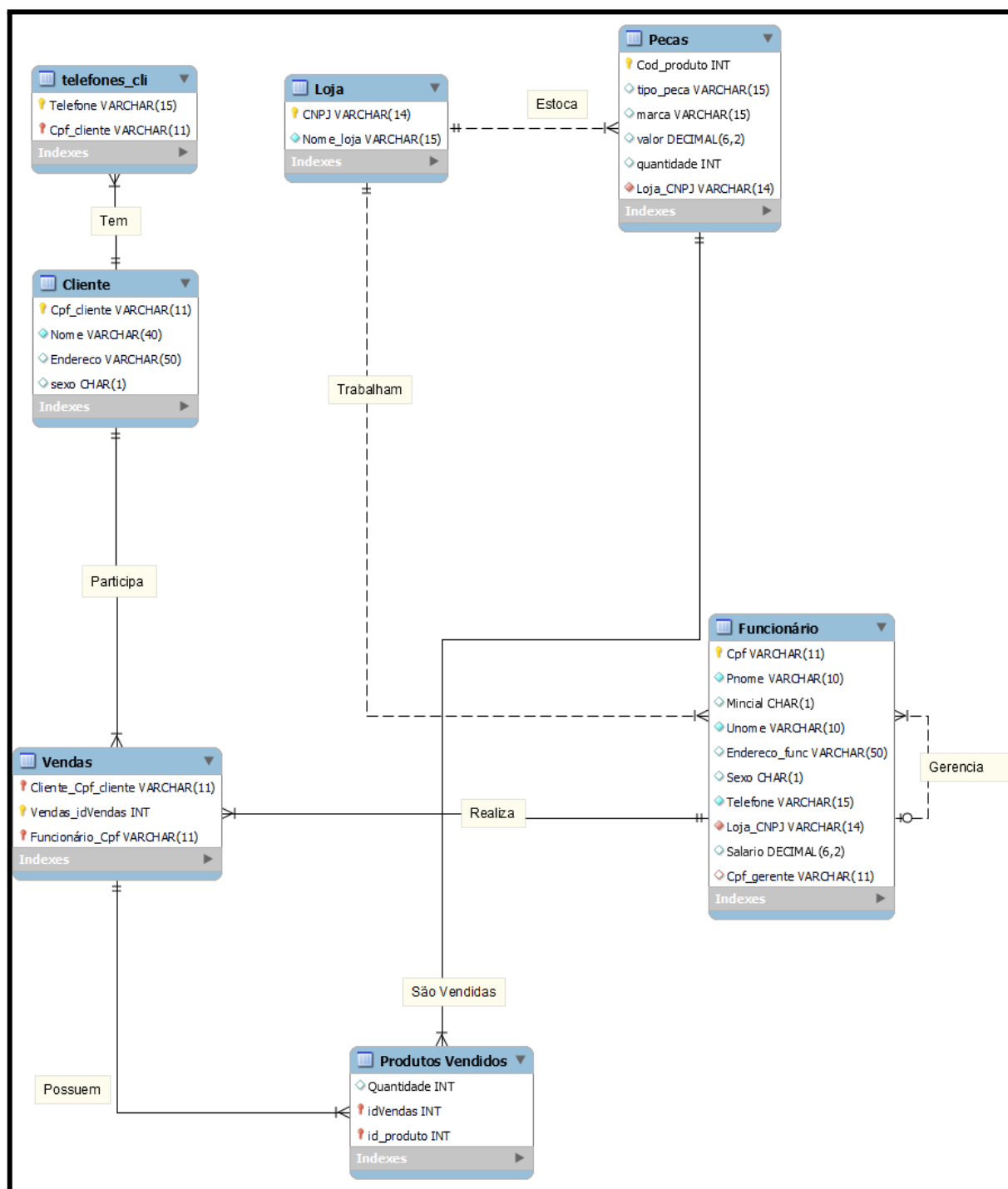
Dicionário de dados dos produtos vendidos				
Nome do Campo	Chave	Tipo de dado	Descrição do Campo	Tamanho do campo
codigo_venda	Primária	inteiro	Número inteiro que representa o código de cada venda	
cod_produtos	Primária	inteiro	Cadeia de caracteres que representam o identificador da peça.	
quantidade	FK	inteiro	Inteiro que representa a quantidade de peças vendidas	11

Fonte: Geilso Faria e Rodrigo Quinhone - Dicionários de dados do projeto

2.2.0 - Projeto

2.2.1 - Conceito Lógico (Modelo Relacional)

Logo após, foi desenvolvido a etapa que precede a fase final do projeto, o modelo relacional. Que se trata de um modelo de dados conceitual que descreve os dados como sendo entidades, atributos e relacionamentos entre tabelas. Esta etapa é uma das etapas que mais chegam perto da fase final, tanto que esta já está integrada no SGBD *MySQL Workbench*. Esta fase gerou o seguinte diagrama de projeto:



Fonte: Geilso Faria e Rodrigo Quinhone - Diagrama do projeto

2.3.0 - Implementação e Testes

Logo, com todas as informações obtidas anteriormente o banco de dados já podia ser construído na plataforma *MySQL Workbench*. Nesse momento todos os relacionamentos, cardinalidades, restrições de chaves, restrição de integridade, modelagem de dados, entre outros, foram transcritos para uma script.sql que representa o banco de dados.

Os testes foram feitos com o intuito de verificar possíveis falhas (falha de inserção, exclusão e alteração/modificação) neste momento foi percebido que seria necessária uma entidade que pudesse especificar cada produto vendido a cada venda realizada. Nesta

etapa, uma anomalia foi encontrada, sendo ela: no momento da aquisição cada peça comprada geraria um código de compra novo, ou seja, seria considerado uma compra nova para cada peça adquirida pelo cliente. Logo, nota-se que isso não pode ocorrer, pois em uma única compra pode-se adquirir mais de um produto.

Abaixo segue o script.sql(código) do banco de dados para a ferramenta *MySQL Workbench*:

```
create schema empresa;
use empresa;

create table loja
(
    nome_loja varchar(15) not null,
    CNPJ varchar(14) primary key
);
create table funcionarios
(
    cpf varchar(11) primary key,
    Pnome varchar(10) not null,
    Minical char,
    Unome varchar(10) not null,
    endereço_func varchar(50),
    sexo char check(sexo='F'or sexo='M'),
    Telefone varchar(15) not null,
    Salário decimal(6,2) not null,
    Cpf_gerente varchar(11),
    Loja_CNPJ varchar(14) not null,
    foreign key (Loja_CNPJ) references loja(CNPJ) on delete cascade on update
    cascade,
    foreign key (Cpf_gerente) references funcionarios(cpf) on delete set null on update
    cascade
);
create table clientes
(
    cpf_cliente varchar(11) primary key not null,
    nome varchar(40) not null,
    endereço varchar(50),
    sexo char check(sexo='F'or sexo='M')
);
create table telefones_clientes
(
    cpf_cliente varchar(11) not null,
    telefone varchar(15) not null,
    primary key (cpf_cliente,telefone),
    foreign key (cpf_cliente) references clientes (cpf_cliente) on delete cascade on
    update cascade
);
create table pecas
(
    Cod_produto int primary key,
    tipo_peça varchar(30),
    marca varchar(15),
    valor decimal(6,2),
```

```

    genero_peca char(1) check(genero_peca='M'or genero_peca='F' or
genero_peca='U'),
    quant int,
    CNPJ_loja varchar(14),
    foreign key (CNPJ_loja) references loja(CNPJ) on delete cascade on update cascade
);
create table vendas
(
    cod_venda int not null,
    cpf_cliente varchar(11) not null,
    cpf_vendedor varchar(15) not null,
    primary key (cod_venda,cpf_cliente,cpf_vendedor),
    foreign key (cpf_cliente) references clientes (cpf_cliente) on delete cascade on
update cascade,
    foreign key (cpf_vendedor) references funcionarios (cpf) on delete cascade on update
cascade
);

create table produtos_vendidos
(
    codigo_venda int not null,
    cod_produtos int,
    quantidade int,
    primary key (codigo_venda,cod_produtos),
    foreign key (codigo_venda) references vendas (cod_venda) on delete cascade on
update cascade,
    foreign key (cod_produtos) references pecas (Cod_produto) on delete cascade on
update cascade
);

```

Fonte: Geilso Faria e Rodrigo Quinhone

Os dados fictícios inseridos foram:

```

insert into loja values('LojaX','54712536610288');

insert into funcionarios values
('12345678966', 'Joao','M','Wong','Rua Sete de Setembro, 35, Dourados, MS','M',
'67992345678',5000.00,null,'54712536610288'),
('00012345678', 'Jorge','A','Silva','Rua São José, 34, Dourados, MS','M',
'67998256423',1900.00,'12345678966','54712536610288'),
('21345678911', 'Aline','B','Lima','Avenida Brasil, 20, Dourados, MS','F',
'67996022844',1900.00,'12345678966','54712536610288'),
('05165432199', 'Joice','T','Rodrigues','Rua Santo Antônio, 31, Dourados, MS','F',
'67991200103',1900.00,'12345678966','54712536610288'),
('66158746320', 'Lucas','O','Leite','Rua Cinco, 78, Dourados, MS','M',
'67994449821',1900.00,'12345678966','54712536610288'),
('85621035598', 'Maria','O','Conceicao','Rua das Gracas, 8, Dourados, MS','F',
'67990050127',1900.00,'12345678966','54712536610288');

insert into clientes values
('11875426510', 'Jonas Teixeira','Rua das Flores, 21, Dourados, MS','M'),

```

```

('00123485964', 'Renata Lima Faria', 'Rua Tiradentes, Dourados, MS', 'F'),
('76592225091', 'Maria da Conceicao', 'Avenida Dom Pedro, 96, Rio Brilhante, MS', 'F'),
('20022019653', 'Tom Rodrigues', 'Avenida Parana, 57, Ponta Pora, MS', 'M'),
('20091921052', 'Joana Pereira', 'Rua Mato Grosso, Dourados, MS', 'F'),
('29933356173', 'Eduarda Leite Faria', 'Rua Primeiro de Maio, Douradina, MS', 'F'),
('48972356110', 'Steve Silva', 'Rua Projetada 1, Dourados, MS', 'M'),
('88896532143', 'Paula Faria', 'Avenida Espirito Santo, Dourados, MS', 'F');

```

insert into telefones_clientes values

```

('11875426510', '67991234567'),
('11875426510', '67992353716'),
('00123485964', '67999235106'),
('76592225091', '67993270154'),
('20022019653', '6796625487'),
('20022019653', '6799900125'),
('20091921052', '6798270102'),
('29933356173', '6799039040'),
('29933356173', '6799825345'),
('88896532143', '6799925267');

```

insert into pecas values

```

(1234, 'Camisa', 'Marca A', 78.89, 'M', 50, '54712536610288'),
(4321, 'Vestido', 'Marca B', 99.99, 'F', 20, '54712536610288'),
(2199, 'Blusa', 'Marca C', 49.99, 'F', 56, '54712536610288'),
(3258, 'Short', 'Marca A', 39.99, 'M', 19, '54712536610288'),
(2193, 'Moletom', 'Marca C', 129.99, 'U', 9, '54712536610288'),
(3567, 'Luva', 'Marca C', 19.99, 'U', 33, '54712536610288'),
(2610, 'Lencol Casal', 'Marca L', 149.49, 'F', 6, '54712536610288'),
(1123, 'Toalha de Banho', 'Marca L', 29.99, 'M', 3, '54712536610288'),
(9851, 'Cinto de Couro', 'Marca C', 39.99, 'M', 5, '54712536610288'),
(7430, 'Calca Jeans', 'Marca A', 99.99, 'M', 41, '54712536610288'),
(7431, 'Calca Jeans', 'Marca A', 99.99, 'F', 35, '54712536610288'),
(3059, 'Sapato Infantil', 'Marca D', 89.99, 'M', 2, '54712536610288'),
(7899, 'Roupa de Banho', 'Marca B', 65.99, 'F', 6, '54712536610288'),
(2793, 'Casaco', 'Marca C', 199.99, 'F', 2, '54712536610288');

```

insert into vendas values

```

(001, '11875426510', '00012345678'),
(002, '00123485964', '00012345678'),
(003, '76592225091', '21345678911'),
(004, '20022019653', '05165432199'),
(005, '20091921052', '66158746320'),
(006, '29933356173', '05165432199'),
(007, '48972356110', '21345678911'),
(008, '88896532143', '00012345678'),
(009, '88896532143', '00012345678'),
(010, '20091921052', '05165432199'),
(011, '48972356110', '21345678911');

```

insert into produtos_vendidos values

```

(001, 1234, 2),
(001, 4321, 1),
(001, 2199, 6),
(002, 3258, 1),

```

```

(003, 2193, 5),
(004, 3567, 1),
(005, 2610, 3),
/*Selecione todos os funcionários que realizaram algum venda*/
select distinct Pnome, Minical, Unome
from funcionarios, vendas
where cpf=cpf_vendedor;

```

```

-- 2
/*Selecione todos os produtos que custam mais de R$20.00*/
select tipo_peça
from pecas
where valor>50.0;

```

```

/*Consultas aninhadas*/
-- 1

```

```

(006, 1123, 2),
(007, 9851, 2),
(007, 7430, 5),
(008, 7431, 1),
(009, 3059, 2),
(010, 7899, 1),
(011, 2793, 3);

```

Após a implementação, para testagem foram realizadas 2 consultas simples, 4 consultas aninhadas e complexas, uma visão do banco de dados e o desenvolvimento de um gatilho(trigger).

```

/*Consultas simples*/
-- 1
/*Selecione os nomes dos clientes que não compraram a peça de ID =2610 */
select nome
from clientes
where clientes.cpf_cliente not in(select clientes.cpf_cliente
                                from
                                clientes,vendas,produtos_vendidos
                                where
                                clientes.cpf_cliente=vendas.cpf_cliente and cod_venda=codigo_venda and
                                cod_produtos=2610);

-- 2

/*Para cada venda realiza determine o codigo da venda, código do produto e o tipo da
peça*/
select cod_venda, cod_produtos, tipo_peça
from vendas, produtos_vendidos, pecas
where codigo_venda=cod_venda and cod_produtos=Cod_produto
group by cod_venda, cod_produtos
order by (cod_venda);

-- 3

```



```

/*Para cada venda recupere nome do cliente que realizou a compra, tipo da peça que o
cliente comprou e o código da venda*/
select nome,tipo_peça, cod_venda
from clientes,vendas, pecas, produtos_vendidos
where clientes.cpf_cliente=vendas.cpf_cliente and cod_venda=codigo_venda and
cod_produtos=Cod_produto
order by (cod_venda);

```

-- 4

```

/*Selecione o nome dos clientes que não compraram com a vendedora 'Joice' */
select nome
from clientes
where nome not in (select nome
                    from clientes,funcionarios,vendas
                    where clientes.cpf_cliente=vendas.cpf_cliente and
cpf=cpf_vendedor and Pnome='Joice');

```

/*VISÃO*/

```

/*Escreva uma visão que contenha o nome da peça, o seu gênero, sua marca a sua
quantidade atual e a soma de quantidade de peças vendidas*/

```

```

create view A as
select tipo_peça,quant,sum(quantidade),genero_peca,marca
from vendas, produtos_vendidos,pecas
where cod_venda=codigo_venda and cod_produtos=Cod_produto
group by cod_produto;

```

```

select * from A;

```

/*TRIGGER*/

```

/*Crie um gatilho para diminuir a quantidade de peças (na tabela peças), toda vez em
que esta for vendida*/

```

```

delimiter $$
create trigger diminui_quant
after insert on produtos_vendidos
for each row
begin
update pecas set quant=quant - new.quantidade;
end $$
delimiter ;

```

2.4.0 - Carga horária por integrante

Atividade	Geilso	Rodrigo
Levantamento de requisitos	50%	50%
Modelagem	50%	50%
Projeto	50%	50%

Implementação e testes	50%	50%
------------------------	-----	-----

3.0 - Conclusão

Portanto, o que ELMASRI e NAVATHE (2010) afirmaram inicialmente na introdução do relatório (pág. 5), torna-se válido. Pois, com a proliferação de métodos de armazenamento digital, gerenciar grandes quantidades de informações começa a se tornar um problema. Logo, um sistema gerenciador de banco de dados serve, justamente, para que todas essas informações possam ser registradas e armazenadas de maneira segura, organizada e padronizada.

Ademais, testar sobre esses conceitos neste projeto foi enormemente vantajoso, pois pode-se colocar em prática o que foi passado e treinado em sala de aula, além de possibilitar a ampliação e o entendimento de um esquema SGBD, bem como, o contexto geral para que uma aplicação seja desenvolvida corretamente, desde o momento da etapa de levantamento de requisitos até a fase de implementação e testes. Logo, utilizar um SGBD não se trata apenas de armazenar informações, bem como diminuição de custos, ganho de tempo, aumento de ganhos e outros mais.

4.0-Bibliografia

ELMASRI, R.; NAVATHE, S. B.. Sistemas de Banco de Dados. 4a ed Pearson-Addison-Wesley, 2005.