



CENTRO UNIVERSITÁRIO UNIRUY
DEPARTAMENTO DE TI

ÍCARO LIMA, RUAN MÜLLER, ALAN GABRIEL, LUAN MASAO

Sistema de Gestão de Locadora de Veículos – S.G.Loc.Ve

SALVADOR

2023



CENTRO UNIVERSITÁRIO UNIRUY
DEPARTAMENTO DE TI

ÍCARO LIMA, RUAN MÜLLER, ALAN SANTOS, LUAN MASAO

Sistema de Gestão de Locadora de Veículos – S.G.Loc.Ve

Trabalho avaliativo semestral
apresentado a disciplina de Banco de
Dados, pertencente ao Departamento de
TI do Centro Universitário Ruy Barbosa,
como parte dos requisitos para obtenção
de aprovação na disciplina.

Professor: Heleno Cardoso Filho

SALVADOR

2023

SÚMARIO

| | |
|--|-----------|
| 1. Introdução..... | 1 |
| 2. Descrição do Negócio..... | 2 |
| 2.1 Funcionalidades do Banco de Dados | 2 |
| 3. Modelagem de Dados..... | 3 |
| 3.1 Modelo Entidade-Relacionamento: Modelo Conceitual..... | 3 |
| 3.1.1 Interpretação do DER..... | 4 |
| 3.2 Definição das Tabelas: Modelo Lógico..... | 5 |
| 3.3 Implementação do Banco de Dados: Modelo Físico..... | 7 |
| 4. Ferramentas Utilizadas..... | 18 |
| 5. Considerações Finais..... | 18 |

1. Introdução

Os sistemas de banco de dados são mais do que apenas recipientes de informações; eles são a espinha dorsal de aplicativos da web, sistemas de gerenciamento de informações de saúde, sistemas de suporte a decisões, sistemas de gerenciamento de recursos humanos e muito mais. Compreender como os bancos de dados funcionam e como projetá-los eficientemente é essencial para profissionais de tecnologia da informação e aqueles que buscam tirar o máximo proveito dos dados.

Este trabalho visa proporcionar uma visão abrangente do mundo dos bancos de dados e equipar o leitor com o conhecimento necessário para compreender, projetar e gerenciar sistemas de banco de dados eficazes. À medida que mergulhamos nesses tópicos, você descobrirá como os bancos de dados desempenham um papel crucial na nossa sociedade digital e como eles continuam a evoluir para enfrentar os desafios do futuro.

2. Descrição do Negócio

Esse projeto tem como principal objetivo a criação de um banco de dados para guardar informações dos Clientes, Funcionários e Veículos com suas respectivas Marcas e Modelos.

Tendo por finalidade controlar a reserva dos carros por meio dos contratos de locação onde servem como principal base para o controle do sistema por meio dos funcionários responsáveis da entrada e saída de veículos na locadora.

Sendo assim, torna-se primordial a presença de um banco de dados, que oferece, comodidade, desempenho e integridade das informações no decorrer das atividades que a empresa terá.

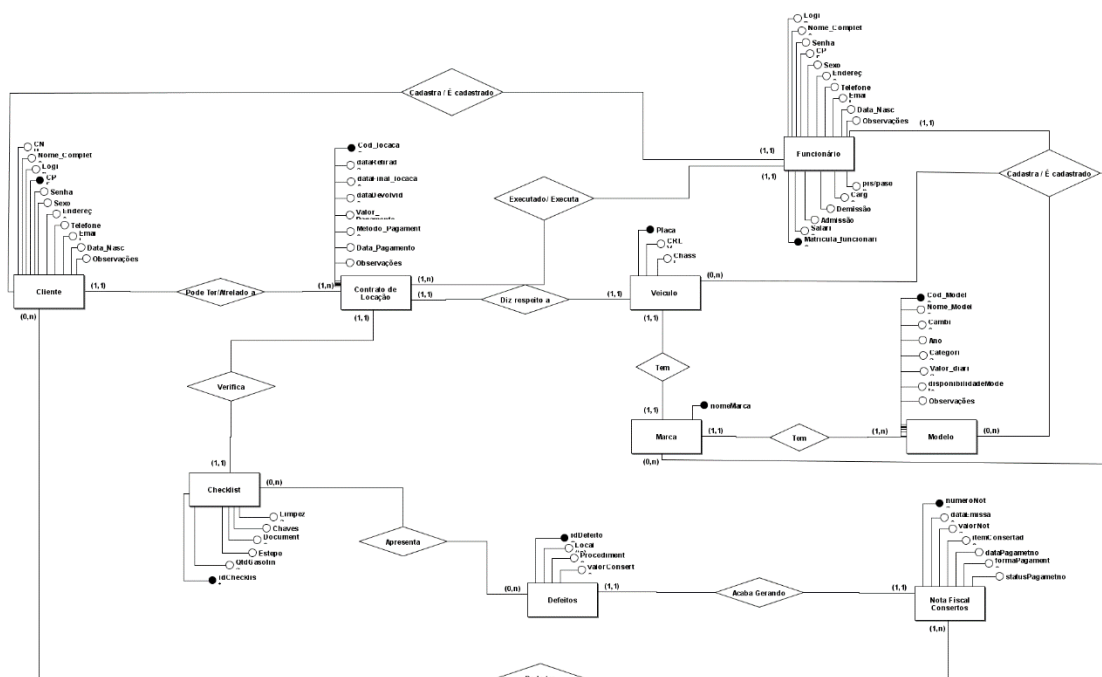
2.1 Funcionalidades do Banco de Dados

- 1) O Sistema Cadastra, Modifica, Exclui e Busca Informações dos Funcionários;
- 2) O Sistema Pede Login e Senha dos Funcionários;
- 3) O Sistema Cadastra, Modifica, Exclui e Busca Informações dos Clientes;
- 4) O Sistema Pede Login e Senha dos Clientes;
- 5) O Sistema Cadastra, Modifica, Exclui e Busca Informações dos Veículos, juntamente com as Marcas e modelos;
- 6) O Sistema informa a disponibilidade do Modelo do veículo;
- 7) O Sistema retém informações no Contrato de locação do veículo que o cliente reserva e também os dados do pagamento;
- 8) O Sistema Informa Defeitos encontrados, conforme a realização do Checklist na devolução no término de locação do veículo;
- 9) O Sistema Guarda e Consulta Informações da Nota Fiscal gerada para pagamento, referente ao defeito do veículo;

3. Modelagem de Dados

3.1 Modelagem Entidade-Relacionamento: Modelo Conceitual

A partir das funcionalidades do banco de dados foi elaborado o modelo E.R. O modelo conceitual ou Diagrama Entidade-Relacionamento (DER), define as entidades e requisitos do banco de dados, e de que maneira se relacionam.



3.1.1 Interpretação do DER

Entidades Principais:

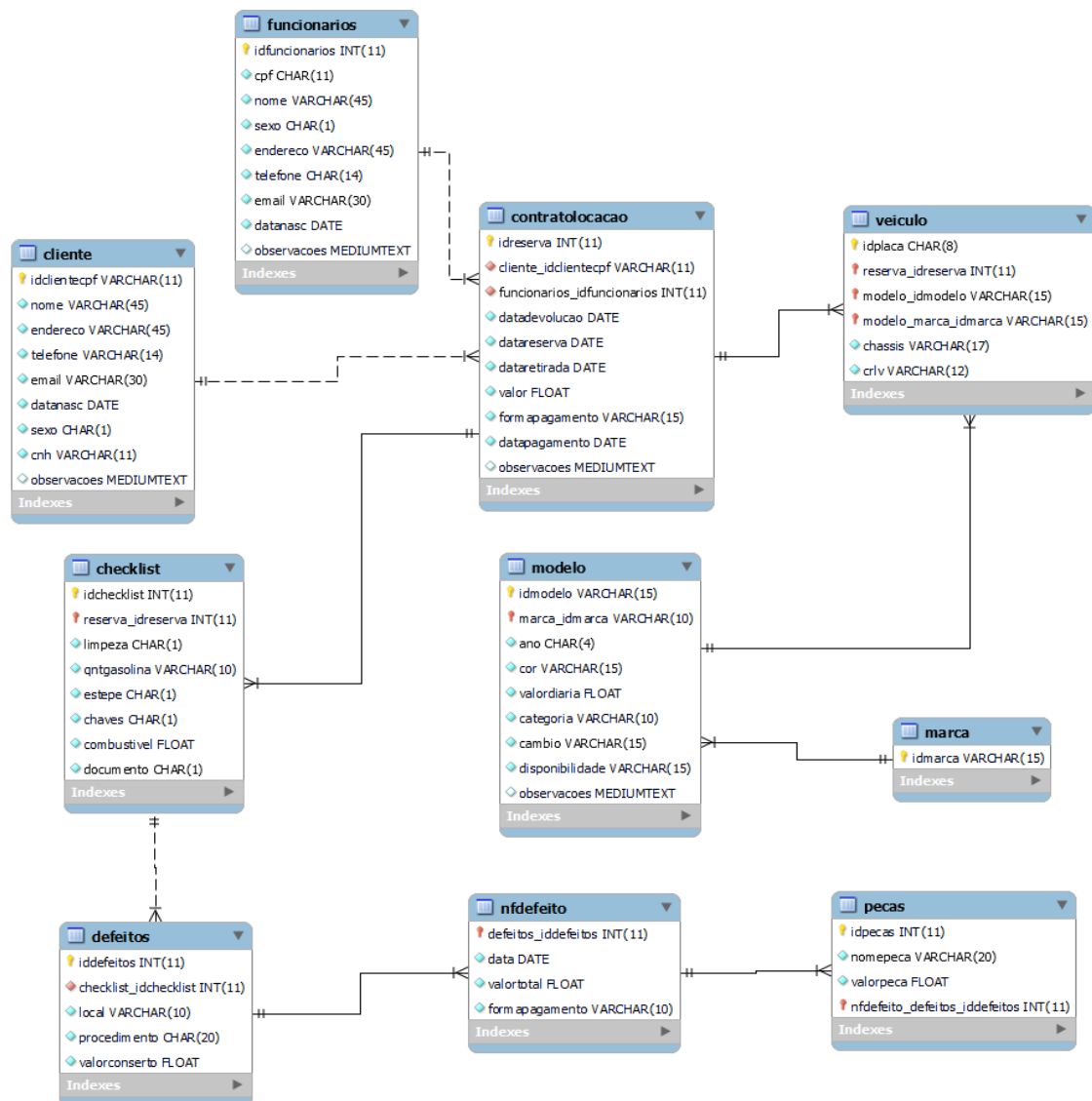
- Cliente
- Funcionários
- Veículo
- Marca
- Modelo
- ContratoLocacao
- Checklist
- Defeitos
- NFDefeito
- Peças

Relacionamentos:

- **Um cliente pode fazer vários contratos de locação** (relação 1 para N entre Cliente e ContratoLocacao).
- **Um funcionário pode estar associado a vários contratos de locação** (relação 1 para N entre Funcionários e ContratoLocacao).
- **Um veículo pode estar associado a vários contratos de locação** (relação 1 para N entre Veículo e ContratoLocacao).
- **Um veículo pertence a um modelo de carro** (relação 1 para 1 entre Veículo e Modelo).
- **Um modelo de carro pertence a uma marca** (relação 1 para 1 entre Modelo e Marca).
- **Um contrato de locação pode ter vários checklists** (relação 1 para N entre ContratoLocacao e Checklist).
- **Um checklist pode estar associado a vários defeitos** (relação 1 para N entre Checklist e Defeitos).
- **Um defeito pode estar relacionado a uma nota fiscal de defeito** (relação 1 para 1 entre Defeitos e NFDefeito).
- **Uma nota fiscal de defeito pode ter várias peças relacionadas** (relação 1 para N entre NFDefeito e Peças).

3.2 Definição das Tabelas: Modelo Lógico

O modelo lógico relacional define quais os nomes das tabelas e os nomes das colunas que compõem essas tabelas, tal como o tipo de dado que cada coluna vai receber, e a cardinalidade entre as tabelas no BD.



3.2.1 Chaves das tabelas

Primárias:

Cliente: idclientecpf

Funcionários: idfuncionarios

Veículo: idplaca

Marca: idmarca

Modelo: idmodelo, marca_idmarca

ContratoLocacao: idreserva

Checklist: idchecklist, reserva_idreserva

Defeitos: iddefeitos

NFDefeito: defeitos_iddefeitos

Pecas: idpecas, nfdefeito_defeitos_iddefeitos

Secundárias:

A tabela **ContratoLocacao** possui chaves estrangeiras para **Cliente** e **Funcionários**.

A tabela **Veículo** possui chaves estrangeiras para **ContratoLocacao** e **Modelo**.

A tabela **Checklist** possui chave estrangeira para **ContratoLocacao**.

A tabela **Defeitos** possui chave estrangeira para **Checklist**.

A tabela **NFDefeito** possui chave estrangeira para **Defeitos**.

A tabela **Pecas** possui chave estrangeira para **NFDefeito**.

3.3 Implementação do Banco de Dados: Modelo Físico

O modelo físico descreve como os dados estão realmente armazenados. São descritas a estrutura física de armazenamento do banco de dados, sua organização de arquivos e seus métodos de acesso.

O modelo de Banco de Dados é enriquecido com detalhes que influenciam no desempenho do Banco de Dados, mas não interferem em sua funcionalidade.

3.3.1 SQL

Utilizando o software MySQL Workbench, foram criadas e relacionadas as tabelas do Banco de Dados na Linguagem de Consulta Estruturada (SQL).

-- Desativar verificação de chaves únicas e chaves estrangeiras temporariamente

```
SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS,
UNIQUE_CHECKS=0;
SET
@OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_C
HECKS, FOREIGN_KEY_CHECKS=0;
SET @OLD_SQL_MODE=@@SQL_MODE,
SQL_MODE='ONLY_FULL_GROUP_BY,STRICT_TRANS_
TABLES,NO_ZERO_IN_DATE,NO_ZERO_DATE,ERROR_
FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION'
;
```

-- Criar o banco de dados 'locadora' e usá-lo

```
CREATE DATABASE `locadora` DEFAULT CHARACTER  
SET utf8;  
USE `locadora`;
```

--Cria a Tabela 'cliente'

```
CREATE TABLE `cliente` (  
  `idclientecpf` VARCHAR(11) NOT NULL,  
  `nome` VARCHAR(45) NOT NULL,  
  `endereco` VARCHAR(45) NOT NULL,  
  `telefone` VARCHAR(14) NOT NULL,  
  `email` VARCHAR(30) NOT NULL,  
  `datanasc` DATE NOT NULL,  
  `sexo` CHAR(1) NOT NULL,  
  `cnh` VARCHAR(11) NOT NULL,  
  `observacoes` MEDIUMTEXT NULL,  
  PRIMARY KEY (`idclientecpf`)  
) ENGINE = InnoDB;
```

-- Cria a Tabela 'funcionarios'

```
CREATE TABLE `funcionarios` (  
  `idfuncionarios` INT NOT NULL,  
  `cpf` CHAR(11) NOT NULL,  
  `nome` VARCHAR(45) NOT NULL,  
  `sexo` CHAR(1) NOT NULL,  
  `endereco` VARCHAR(45) NOT NULL,  
  `telefone` CHAR(14) NOT NULL,  
  `email` VARCHAR(30) NOT NULL,  
  `datanasc` DATE NOT NULL,  
  `observacoes` MEDIUMTEXT NULL,  
  PRIMARY KEY (`idfuncionarios`)  
) ENGINE = InnoDB;
```

--Cria a Tabela 'contratolocacao'

```
CREATE TABLE `contratolocacao` (  
  `idreserva` INT NOT NULL,  
  `cliente_idclientecpf` VARCHAR(11) NOT NULL,  
  `funcionarios_idfuncionarios` INT NOT NULL,  
  `datadevolucao` DATE NOT NULL,  
  `datareserva` DATE NOT NULL,  
  `dataretirada` DATE NOT NULL,  
  `valor` FLOAT NOT NULL,  
  `formapagamento` VARCHAR(15) NOT NULL,  
  `datapagamento` DATE NOT NULL,  
  `observacoes` MEDIUMTEXT NULL,  
  PRIMARY KEY (`idreserva`),  
  INDEX `fk_reserva_cliente1_idx` (`cliente_idclientecpf`),  
  INDEX `fk_contratolocacao_funcionarios1_idx`  
  (`funcionarios_idfuncionarios`),  
  CONSTRAINT `fk_reserva_cliente1`  
    FOREIGN KEY (`cliente_idclientecpf`)  
    REFERENCES `cliente` (`idclientecpf`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION,  
  CONSTRAINT `fk_contratolocacao_funcionarios1`  
    FOREIGN KEY (`funcionarios_idfuncionarios`)  
    REFERENCES `funcionarios` (`idfuncionarios`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION  
) ENGINE = InnoDB;
```

-- Tabela 'marca'

```
CREATE TABLE `marca` (  
  `idmarca` VARCHAR(15) NOT NULL,  
  PRIMARY KEY (`idmarca`)  
) ENGINE = InnoDB;
```

--Cria a Tabela 'modelo'

```
CREATE TABLE `modelo` (  
  `idmodelo` VARCHAR(15) NOT NULL,  
  `marca_idmarca` VARCHAR(10) NOT NULL,  
  `ano` CHAR(4) NOT NULL,  
  `cor` VARCHAR(15) NOT NULL,  
  `valordiaria` FLOAT NOT NULL,  
  `categoria` VARCHAR(10) NOT NULL,  
  `cambio` VARCHAR(15) NOT NULL,  
  `disponibilidade` VARCHAR(15) NOT NULL,  
  `observacoes` MEDIUMTEXT NULL,  
  PRIMARY KEY (`idmodelo`, `marca_idmarca`),  
  INDEX `fk_modelo_marca1_idx` (`marca_idmarca`),  
  CONSTRAINT `fk_modelo_marca1`  
    FOREIGN KEY (`marca_idmarca`)  
    REFERENCES `marca` (`idmarca`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION  
) ENGINE = InnoDB;
```

--Cria a Tabela 'veiculo'

```
CREATE TABLE `veiculo` (  
  `idplaca` CHAR(8) NOT NULL,  
  `reserva_idreserva` INT NOT NULL,  
  `modelo_idmodelo` VARCHAR(15) NOT NULL,  
  `modelo_marca_idmarca` VARCHAR(15) NOT NULL,  
  `chassis` VARCHAR(17) NOT NULL,  
  `crlv` VARCHAR(12) NOT NULL,  
  PRIMARY KEY (`idplaca`, `reserva_idreserva`,  
  `modelo_idmodelo`, `modelo_marca_idmarca`),  
  INDEX `fk_veiculo_reserva_idx` (`reserva_idreserva`),  
  INDEX `fk_veiculo_modelo1_idx` (`modelo_idmodelo`,  
  `modelo_marca_idmarca`),  
  CONSTRAINT `fk_veiculo_reserva`
```

```

FOREIGN KEY (`reserva_idreserva`)
  REFERENCES `contratolocacao` (`idreserva`)
  ON DELETE NO ACTION
  ON UPDATE NO ACTION,
CONSTRAINT `fk_veiculo_modelo1`
  FOREIGN KEY (`modelo_idmodelo`,
`modelo_marca_idmarca`)
  REFERENCES `modelo` (`idmodelo`, `marca_idmarca`)
  ON DELETE NO ACTION
  ON UPDATE NO ACTION
) ENGINE = InnoDB;

```

--Cria a Tabela 'checklist'

```

CREATE TABLE `checklist` (
  `idchecklist` INT NOT NULL,
  `reserva_idreserva` INT NOT NULL,
  `limpeza` CHAR(1) NOT NULL,
  `qntgasolina` VARCHAR(10) NOT NULL,
  `estepe` CHAR(1) NOT NULL,
  `chaves` CHAR(1) NOT NULL,
  `combustivel` FLOAT NOT NULL,
  `documento` CHAR(1) NOT NULL,
  PRIMARY KEY (`idchecklist`, `reserva_idreserva`),
  INDEX `fk_checklist_reserva1_idx` (`reserva_idreserva`),
  CONSTRAINT `fk_checklist_reserva1`
    FOREIGN KEY (`reserva_idreserva`)
      REFERENCES `contratolocacao` (`idreserva`)
      ON DELETE NO ACTION
      ON UPDATE NO ACTION
) ENGINE = InnoDB;

```

--Cria a Tabela 'defeitos'

```
CREATE TABLE `defeitos` (  
  `iddefeitos` INT NOT NULL,  
  `checklist_idchecklist` INT NOT NULL,  
  `local` VARCHAR(10) NOT NULL,  
  `procedimento` CHAR(20) NOT NULL,  
  `valorconserto` FLOAT NOT NULL,  
  INDEX `fk_defeitos_checklist1_idx` (`checklist_idchecklist`),  
  PRIMARY KEY (`iddefeitos`),  
  CONSTRAINT `fk_defeitos_checklist1`  
    FOREIGN KEY (`checklist_idchecklist`)  
    REFERENCES `checklist` (`idchecklist`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION  
) ENGINE = InnoDB;
```

--Cria a Tabela 'nfdefeitos'

```
CREATE TABLE `nfdefeito` (  
  `defeitos_iddefeitos` int NOT NULL,  
  `data` date NOT NULL,  
  `valortotal` float NOT NULL,  
  `formapagamento` varchar(10) NOT NULL,  
  PRIMARY KEY (`defeitos_iddefeitos`),  
  CONSTRAINT `fk_nfdefeito_defeitos1` FOREIGN KEY  
    (`defeitos_iddefeitos`) REFERENCES `defeitos` (`iddefeitos`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb3;
```

--Cria a Tabela 'pecas'

```
CREATE TABLE `pecas` (  
  `idpecas` int NOT NULL,  
  `nomepeca` varchar(20) NOT NULL,  
  `valorpeca` float NOT NULL,  
  `nfdefeito_defeitos_iddefeitos` int NOT NULL,
```

```

PRIMARY KEY (`idpecas`,`nfdefeito_defeitos_iddefeitos`),
  KEY `fk_itens_nfdefeito1_idx`
  (`nfdefeito_defeitos_iddefeitos`),
  CONSTRAINT `fk_itens_nfdefeito1` FOREIGN KEY
  (`nfdefeito_defeitos_iddefeitos`) REFERENCES `nfdefeito`
  (`defeitos_iddefeitos`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb3;

```

-- Restaurar as configurações originais

```

SET SQL_MODE=@OLD_SQL_MODE;
SET
FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;
SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;

```

-- Inserir dados na tabela 'cliente'

```

INSERT INTO cliente (idclientecpf, nome, endereco, telefone,
email, datanasc, sexo, cnh, observacoes)
VALUES ('12345678901', 'João da Silva', 'Rua A, 123', '111-
1111', 'joao@email.com', '1980-01-15', 'M', 'A1234567',
'Nenhuma observação');

```

```

INSERT INTO cliente (idclientecpf, nome, endereco, telefone,
email, datanasc, sexo, cnh, observacoes)
VALUES ('98765432109', 'Maria Oliveira', 'Av. B, 456', '222-
2222', 'maria@email.com', '1990-05-20', 'F', 'B7654321',
'Observações adicionais');

```

-- Inserir dados na tabela 'funcionarios'

```

INSERT INTO funcionarios (idfuncionarios, cpf, nome, sexo,
endereco, telefone, email, datanasc, observacoes)
VALUES (1, '11111111111', 'Carlos Santos', 'M', 'Rua X, 789',

```



```
'333-3333', 'carlos@email.com', '1985-07-10', 'Nenhuma
observação');
```

```
INSERT INTO funcionarios (idfuncionarios, cpf, nome, sexo,
endereco, telefone, email, datanasc, observacoes)
VALUES (2, '22222222222', 'Ana Pereira', 'F', 'Av. Y, 456',
'444-4444', 'ana@email.com', '1990-03-25', 'Observações
adicionais');
```

-- Inserir dados na tabela 'contratolocacao'

```
INSERT INTO contratolocacao (idreserva, cliente_idclientecpf,
funcionarios_idfuncionarios, datadevolucao, datareserva,
dataretirada, valor, formapagamento, datapagamento,
observacoes)
VALUES (1, '12345678901', 1, '2023-11-15', '2023-11-10',
'2023-11-10', 250.0, 'Cartão', '2023-11-10', 'Nenhuma
observação');
```

```
INSERT INTO contratolocacao (idreserva, cliente_idclientecpf,
funcionarios_idfuncionarios, datadevolucao, datareserva,
dataretirada, valor, formapagamento, datapagamento,
observacoes)
VALUES (2, '98765432109', 2, '2023-11-20', '2023-11-15',
'2023-11-15', 200.0, 'Dinheiro', '2023-11-15', 'Observações
adicionais');
```

-- Inserir dados na tabela 'marca'

```
INSERT INTO marca (idmarca)
VALUES ('Toyota');
```

```
INSERT INTO marca (idmarca)
VALUES ('Honda');
```

-- Inserir dados na tabela 'modelo'

```
INSERT INTO modelo (idmodelo, marca_idmarca, ano, cor,
valordiaria, categoria, cambio, disponibilidade, observacoes)
VALUES ('Corolla', 'Toyota', '2022', 'Prata', 100.0, 'Sedan',
'Automático', 'Disponível', 'Nenhuma observação');
```

```
INSERT INTO modelo (idmodelo, marca_idmarca, ano, cor,
valordiaria, categoria, cambio, disponibilidade, observacoes)
VALUES ('Civic', 'Honda', '2022', 'Branco', 95.0, 'Sedan',
'Automático', 'Disponível', 'Observações adicionais');
```

-- Inserir dados na tabela 'veiculo'

```
INSERT INTO veiculo (idplaca, reserva_idreserva,
modelo_idmodelo, modelo_marca_idmarca, chasis, crlv)
VALUES ('ABC1234', 1, 'Corolla', 'Toyota',
'12345678901234567', 'CRLV2023');
```

```
INSERT INTO veiculo (idplaca, reserva_idreserva,
modelo_idmodelo, modelo_marca_idmarca, chasis, crlv)
VALUES ('XYZ5678', 2, 'Civic', 'Honda',
'98765432109876543', 'CRLV2023');
```

-- Inserir dados na tabela 'checklist'

```
INSERT INTO checklist (idchecklist, reserva_idreserva,
limpeza, qntgasolina, estepe, chaves, combustivel, documento)
VALUES (1, 1, 'S', 'Full', 'S', 'S', 100.0, 'S');
```

```
INSERT INTO checklist (idchecklist, reserva_idreserva,
limpeza, qntgasolina, estepe, chaves, combustivel, documento)
VALUES (2, 2, 'S', '3/4', 'S', 'S', 75.0, 'S');
```

-- Inserir dados na tabela 'defeitos'

```
INSERT INTO defeitos (iddefeitos, checklist_idchecklist, local,  
procedimento, valorconserto)  
VALUES (1, 1, 'Pneu', 'Troca do pneu', 50.0);
```

```
INSERT INTO defeitos (iddefeitos, checklist_idchecklist, local,  
procedimento, valorconserto)  
VALUES (2, 1, 'Para-brisa', 'Reparo no para-brisa', 30.0);
```

```
-- Inserir dados na tabela 'nfdefeito'
```

```
INSERT INTO nfdefeito (defeitos_iddefeitos, data, valortotal,  
formapagamento)  
VALUES (1, '2023-11-15', 50.0, 'Cartão');
```

```
INSERT INTO nfdefeito (defeitos_iddefeitos, data, valortotal,  
formapagamento)  
VALUES (2, '2023-11-15', 30.0, 'Cartão');
```

```
-- Inserir dados na tabela 'pecas'
```

```
INSERT INTO pecas (idpecas, nomepeca, valorpeca,  
nfdefeito_defeitos_iddefeitos)  
VALUES (1, 'Pneu Novo', 100.0, 1);
```

```
INSERT INTO pecas (idpecas, nomepeca, valorpeca,  
nfdefeito_defeitos_iddefeitos)  
VALUES (2, 'Vidro para-brisas', 50.0, 2);
```

```
-- Selecciona o Banco de Dados `locadora`
```

```
USE locadora;
```

```
-- Selecciona todos os clientes
```

```
SELECT * FROM cliente;
```

-- Selecciona todos os funcionarios

SELECT * FROM funcionarios;

-- Seleccionar todas as reservas de locação com detalhes de clientes e funcionários:

**SELECT contratolocacao.*, cliente.nome AS cliente,
funcionarios.nome AS funcionario
FROM contratolocacao
JOIN cliente ON contratolocacao.cliente_idclientecpf =
cliente.idclientecpf
JOIN funcionarios ON
contratolocacao.funcionarios_idfuncionarios =
funcionarios.idfuncionarios;**

-- Seleccionar todos os modelos de carros e suas marcas:

**SELECT modelo.*, marca.idmarca
FROM modelo
JOIN marca ON modelo.marca_idmarca = marca.idmarca
LIMIT 0, 50000;**

-- Seleccionar todos os veículos e seus detalhes:

**SELECT veiculo.*, modelo.idmodelo, modelo.marca_idmarca,
modelo.ano, modelo.cor
FROM veiculo
JOIN modelo ON veiculo.modelo_idmodelo = modelo.idmodelo
AND veiculo.modelo_marca_idmarca =
modelo.marca_idmarca;**

-- Seleccionar as reservas de locação com checklist e defeitos associados:

**SELECT checklist.*, defeitos.*
FROM checklist**

```
LEFT JOIN defeitos ON checklist.idchecklist =  
defeitos.checklist_idchecklist;
```

```
-- Selecionar as reservas de locação com checklist, defeitos,  
notas fiscais e pecas associados:
```

```
SELECT checklist.*, defeitos.*, nfdefeito.*, pecas.*  
FROM checklist  
LEFT JOIN defeitos ON checklist.idchecklist =  
defeitos.checklist_idchecklist  
LEFT JOIN nfdefeito ON defeitos.iddefeitos =  
nfdefeito.defeitos_iddefeitos  
LEFT JOIN pecas ON defeitos.iddefeitos =  
pecas.nfdefeito_defeitos_iddefeitos  
LIMIT 0, 1000;
```

```
-- Selecciona o Banco de Dados `locadora`
```

```
USE locadora;
```

```
-- Excluir um veículo com base na placa:
```

```
DELETE FROM veiculo WHERE idplaca = 'ABC1234';
```

```
-- Excluir um modelo de carro com base no ID de modelo e ID  
de marca:
```

```
DELETE FROM modelo WHERE idmodelo = 'Corolla' AND  
marca_idmarca = 'Toyota';
```

```
-- Excluir um contrato de locação com base no ID de reserva:
```

```
DELETE FROM contratolocacao WHERE idreserva = 1;
```

```
-- Excluir um funcionário com base no ID de funcionário:
```

```
DELETE FROM funcionarios WHERE idfuncionarios = 1;
```

-- Excluir um cliente com base no CPF:

DELETE FROM cliente WHERE idclientecpf = '12345678901';

4. Ferramentas Utilizadas

Durante a elaboração desse trabalho, foram utilizadas as seguintes ferramentas:

- **BRModelo**, para a criação do modelo conceitual do BD;
- **MySQL**, para a criação do diagrama de dados, modelo lógico e físico do BD
- **Youtube**, para hospedagem do vídeo de apresentação do BD, segue link - <https://youtu.be/AelfvSb7-9A?si=2mXVV9nuqibeYc-E>

5. Considerações Finais

Neste trabalho, exploramos a criação e modelagem de um banco de dados para uma locadora de veículos. Durante o processo, focamos em várias etapas cruciais, incluindo a definição do esquema de banco de dados, a criação de tabelas, a configuração de chaves primárias e estrangeiras, e a inserção de dados de exemplo.

Ao longo deste projeto, aprendemos a importância da organização de dados para a eficiência e integridade do sistema. A modelagem do banco de dados é uma parte crítica do desenvolvimento de aplicativos e sistemas que envolvem o armazenamento e recuperação de informações.

Além disso, abordamos questões relacionadas à integridade referencial, garantindo que as relações entre as tabelas sejam mantidas para evitar dados inconsistentes e conflitantes. Compreendemos que a manutenção da integridade dos dados é essencial para garantir a qualidade e confiabilidade das informações armazenadas no banco de dados.

