



UNIVERSIDADE DO VALE DO ITAJAÍ

LUIZ GUSTAVO DE OLIVEIRA  
GABRIEL GARBARI DA SILVA  
JEFERSON FILOMENO

## **FÓRUM TEMÁTICO**

FLORIANÓPOLIS  
2024



LUIZ GUSTAVO DE OLIVEIRA  
GABRIEL GARBARI DA SILVA  
JEFERSON FILOMENO

Relatório apresentado para a disciplina Database for Apps,  
pelo curso de Análise e Desenvolvimento de Sistemas da  
UNIVALI, ministrado pelo Prof. Maurício Pasetto De  
Freitas.

FLORIANÓPOLIS  
2024

## Conteúdo

INTRODUÇÃO .....	4
OBJETIVO DO PROJETO .....	5
DIAGRAMA ENTIDADE-RELACIONAMENTO (DER).....	6
MODELO ENTIDADE-RELACIONAMETNO (MER) .....	11
CONCLUSÃO .....	13

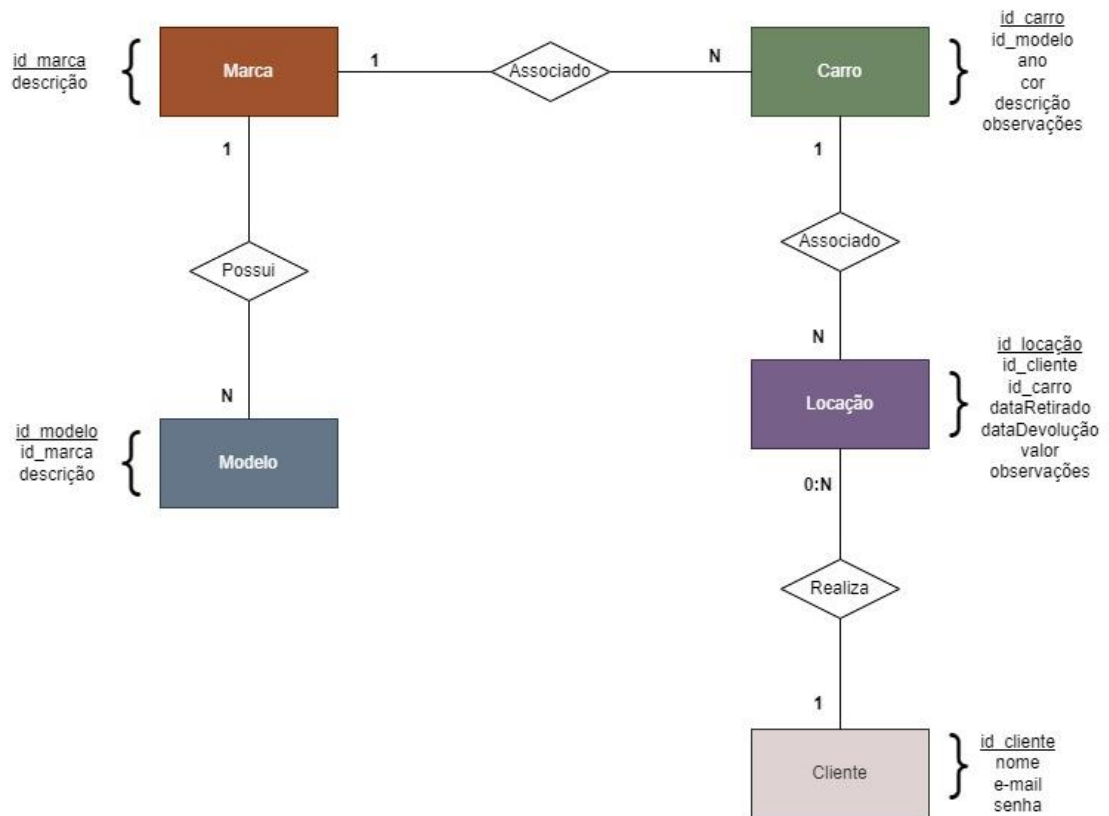
## INTRODUÇÃO

Este relatório descreve o desenvolvimento de um banco de dados relacional para uma locadora de carros, abordando o processo de modelagem desde o projeto conceitual até a implementação física no Sistema de Gerenciamento de Banco de Dados MySQL. O objetivo principal é proporcionar uma estrutura organizada e eficiente para o armazenamento e manipulação dos dados relacionados às operações de locação de veículos, incluindo o controle de marcas, modelos, carros, clientes e locações.

## OBJETIVO DO PROJETO

O objetivo do projeto é criar um sistema de banco de dados que suporte todas as funcionalidades necessárias para a gestão de uma locadora de carros. O sistema deve permitir o cadastro, edição, exclusão e listagem de marcas, modelos de carros, veículos, clientes e locações, garantindo integridade e consistência dos dados.

## DIAGRAMA ENTIDADE-RELACIONAMENTO (DER)



Nesta fase, foi criado um Diagrama Entidade-Relacionamento (DER) para identificar e descrever as entidades (Marca, Modelo, Carro, Cliente, Locação) e seus relacionamentos. Este diagrama fornece uma visão abstrata e de alto nível do banco de dados.

### Projeto Lógico

- **Entidade Marca:**

- Representa a marca dos carros;
- Atributos:
  - `codigo` (identificador único)
  - `descricao` (nome da marca)

- **Entidade Modelo:**

- Representa os modelos dos carros;
- Atributos:
  - `codigo` (identificador único)
  - `codigo_marca` (referência à marca)
  - `descricao` (nome do modelo)
- **Entidade Carro:**
  - Representa os carros disponíveis para locação;
  - Atributos:
    - `codigo` (identificador único)
    - `codigo_modelo` (referência ao modelo)
    - `ano` (ano de fabricação)
    - `cor` (cor do carro)
    - `descricao` (descrição do carro)
    - `observacoes` (observações adicionais)
- **Entidade Cliente:**
  - Representa os clientes da locadora;
  - Atributos:
    - `codigo` (identificador único)
    - `nome` (nome do cliente)
    - `email` (email do cliente)
    - `senha` (senha do cliente)
- **Entidade Locação:**
  - Representa as transações de locação de carros;
  - Atributos:
    - `codigo` (identificador único)
    - `codigo_carro` (referência ao carro)
    - `codigo_cliente` (referência ao cliente)
    - `data_retirada` (data de retirada do carro)
    - `data_devolucao` (data de devolução do carro)
    - `valor` (valor da locação)
    - `observacoes` (observações adicionais)

**Passo 1 : Criar o BD**

```
CREATE DATABASE IF NOT EXISTS locadora_carros;
```

**Passo 2 : Usar BD**

```
USE locadora_carros;
```

**Passo 3 : Criar as Tabelas**

```
CREATE TABLE IF NOT EXISTS Marca (  
    codigo INT PRIMARY KEY AUTO_INCREMENT,  
    descricao VARCHAR(100) NOT NULL  
);  
  
CREATE TABLE IF NOT EXISTS Modelo (  
    codigo INT PRIMARY KEY AUTO_INCREMENT,  
    codigo_marca INT,  
    descricao VARCHAR(100) NOT NULL,  
    FOREIGN KEY (codigo_marca) REFERENCES Marca(codigo)  
);  
  
CREATE TABLE IF NOT EXISTS Carro (  
    codigo INT PRIMARY KEY AUTO_INCREMENT,  
    codigo_modelo INT,  
    ano YEAR,  
    cor VARCHAR(20),  
    descricao VARCHAR(255),  
    observacoes TEXT,  
    FOREIGN KEY (codigo_modelo) REFERENCES Modelo(codigo)  
);  
  
CREATE TABLE IF NOT EXISTS Cliente (  
    codigo INT PRIMARY KEY AUTO_INCREMENT,  
    nome VARCHAR(100) NOT NULL,  
    email VARCHAR(100) NOT NULL,  
    senha VARCHAR(100) NOT NULL  
);  
  
CREATE TABLE IF NOT EXISTS Locacao (  
    codigo INT PRIMARY KEY AUTO_INCREMENT,  
    codigo_carro INT,
```





**UNIVALI**

codigo\_cliente INT,

data\_retirada DATE,

data\_devolucao DATE,

valor DECIMAL(10, 2),

observacoes TEXT,

**FOREIGN KEY** (codigo\_carro) **REFERENCES** Carro(codigo),

**FOREIGN KEY** (codigo\_cliente) **REFERENCES** Cliente(codigo)

);

Passo 4 : Inserir Dados

**--Marca**

**INSERT INTO** Marca (descricao) **VALUES** ('Chevrolet');

**INSERT INTO** Marca (descricao) **VALUES** ('Toyota');

**--Modelo**

**INSERT INTO** Modelo (codigo\_marca, descricao) **VALUES** (1, 'Onix');

**INSERT INTO** Modelo (codigo\_marca, descricao) **VALUES** (2, 'Corolla');

**--Carro**

**INSERT INTO** Carro (codigo\_modelo, ano, cor, descricao, observacoes) **VALUES** (1, 2021, 'Azul', 'Onix 2021', 'Novo');

**INSERT INTO** Carro (codigo\_modelo, ano, cor, descricao, observacoes) **VALUES** (2, 2020, 'Prata', 'Corolla 2020', 'Nenhuma');

**--Cliente**

**INSERT INTO** Cliente (nome, email, senha) **VALUES** ('Carlos Souza', 'carlos@gmail.com', 'senha123');

**INSERT INTO** Cliente (nome, email, senha) **VALUES** ('Maria Santos', 'maria@gmail.com', 'abcdef');

**--Locação**

**INSERT INTO** Locacao (codigo\_carro, codigo\_cliente, data\_retirada, data\_devolucao, valor, observacoes) **VALUES** (1, 1, '2024-06-01', '2024-06-07', 500.00, 'Sem observações');

**INSERT INTO** Locacao (codigo\_carro, codigo\_cliente, data\_retirada, data\_devolucao, valor, observacoes) **VALUES** (2, 2, '2024-06-05', '2024-06-10', 300.00, 'Nenhuma');

**--Verificar dados**

**SELECT \* FROM** Marca;

**SELECT \* FROM** Modelo;

**SELECT \* FROM** Carro;

**SELECT \* FROM** Cliente;



**UNIVALI**

**SELECT \* FROM** Locacao;

--Verificar relação entre tabelas (JOIN)

**SELECT** Modelo.descricao **AS** Modelo, Marca.descricao **AS** Marca

**FROM** Modelo

**JOIN** Marca **ON** Modelo.codigo\_marca = Marca.codigo;

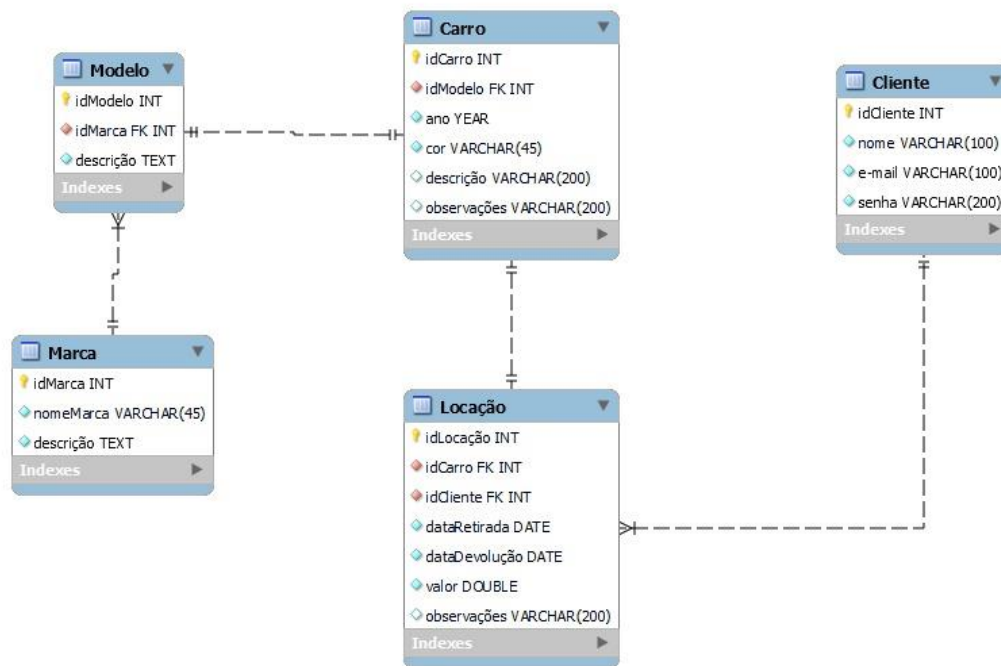
**SELECT** Locacao.codigo, Cliente.nome **AS** Cliente, Carro.descricao **AS** Carro,  
Locacao.data\_retirada, Locacao.data\_devolucao, Locacao.valor

**FROM** Locacao

**JOIN** Cliente **ON** Locacao.codigo\_cliente = Cliente.codigo

**JOIN** Carro **ON** Locacao.codigo\_carro = Carro.codigo;

## MODELO ENTIDADE-RELACIONAMENTO (MER)



MER é uma ferramenta essencial na fase de design de um banco de dados, ele é fundamental no processo de design de bancos de dados e serve como um intermediário entre a análise dos requisitos e a implementação física no Sistema de Gerenciamento de Banco de Dados (SGBD).

Fazendo com que possa ser proporcionado uma base sólida na criação de um sistema eficiente, foi definido de maneira clara e precisa as diferentes entidades envolvidas no sistema da locadora de carros como:

Entidades

‘Marca’, ‘Modelo’, ‘Carro’, ‘Cliente’ e ‘Locação’

Atributos

‘Codigo’, ‘Ano’, ‘Cor’.

Relacionamentos

**Modelo e Carro;**

**Cliente e locação.**

- Entidade `Marca`

- `codigo (PK)`
- `descricao`

- Entidade `Modelo`

- `codigo (PK)`
- `codigo_marca (FK)`
- `descricao`

- Entidade `Carro`

- `codigo (PK)`
- `codigo_modelo (FK)`
- `ano`
- `cor`
- `descricao`
- `observacoes`

- Entidade `Cliente`

- `codigo (PK)`
- `nome`
- `email`
- `senha`

- Entidade `Locacao`

- `codigo (PK)`
- `codigo_carro (FK)`
- `codigo_cliente (FK)`
- `data_retirada`
- `data_devolucao`
- `valor`
- `observacoes`

## CONCLUSÃO

O desenvolvimento deste projeto de banco de dados para uma locadora de carros ressaltou a importância de uma abordagem sistemática e bem estruturada na concepção e implementação de sistemas de informação. Desde a fase inicial de modelagem conceitual até a implementação física no SGBD MySQL, cada etapa foi essencial para garantir um sistema eficiente e funcional.

O uso do Diagrama Entidade-Relacionamento (DER) permitiu uma representação clara das entidades e seus relacionamentos, facilitando a transição para o modelo lógico e, posteriormente, para o modelo físico. Essa abordagem metodológica assegurou que o banco de dados fosse bem projetado, evitando redundâncias e assegurando a integridade dos dados.

A escolha do MySQL como SGBD foi fundamental para atender às necessidades da locadora, proporcionando um ambiente robusto e confiável para a gestão de dados. MySQL oferece ferramentas poderosas para a criação, manipulação e manutenção de bancos de dados, garantindo consistência e desempenho elevados.

Em conclusão, este projeto proporcionou uma experiência prática valiosa em modelagem e implementação de bancos de dados, consolidando conceitos teóricos e demonstrando a importância de um design cuidadoso e uma execução metódica. O sistema resultante atende plenamente às necessidades da locadora de carros, oferecendo uma base sólida para futuras expansões e melhorias.

## LINKS E REFERENCIAS

[https://github.com/xXOliveiraLuizXx/Locadora\\_carro\\_ScriptMySQL-Forum\\_Tematico-/blob/main/README.md](https://github.com/xXOliveiraLuizXx/Locadora_carro_ScriptMySQL-Forum_Tematico-/blob/main/README.md)