



**UNifeob**  
| ESCOLA DE NEGÓCIOS



2024

# PROJETO INTEGRADO



UNIFEOB

CENTRO UNIVERSITÁRIO DA FUNDAÇÃO DE ENSINO  
OCTÁVIO BASTOS

ESCOLA DE NEGÓCIOS

**ANÁLISE E DESENVOLVIMENTO DE SISTEMAS  
CIÊNCIA DA COMPUTAÇÃO**

**PROJETO INTEGRADO**

**DESENVOLVIMENTO DE SOLUÇÕES CONSOLE  
INTEGRADAS PARA EDUCAÇÃO,  
SUSTENTABILIDADE, INCLUSÃO SOCIAL E  
EMPREENDEDORISMO**

**Localiza Rent a Car S.A.**

SÃO JOÃO DA BOA VISTA, SP

NOVEMBRO 2024

UNIFEOB

CENTRO UNIVERSITÁRIO DA FUNDAÇÃO DE ENSINO  
OCTÁVIO BASTOS

ESCOLA DE NEGÓCIOS

**ANÁLISE E DESENVOLVIMENTO DE SISTEMAS  
CIÊNCIA DA COMPUTAÇÃO**

**PROJETO INTEGRADO**

**DESENVOLVIMENTO DE SOLUÇÕES CONSOLE  
INTEGRADAS PARA EDUCAÇÃO,  
SUSTENTABILIDADE, INCLUSÃO SOCIAL E  
EMPREENDEDORISMO**

**Localiza Rent a Car S.A.**

MÓDULO MODELAGEM E DESENVOLVIMENTO DE SISTEMAS

Business Intelligence – Profª. Mariângela Martimbianco Santos

Programação Orientada a Objeto – Prof. Nivaldo de Andrade

Lógica de Programação – Prof. Marcelo Ciacco Almeida

Modelagem de Dados – Prof. Max Streicher Vallim

Projeto de Modelagem e Desenvolvimento de Sistemas – Profª. Mariângela M. Santos

Estudantes:

João Gabriel Patrocínio do Amaral, RA 24001089

Lucas Vigo Calió, RA 24000092

Mateus Oliveira Milane, RA 24000308

SÃO JOÃO DA BOA VISTA, SP  
NOVEMBRO 2024

# SUMÁRIO

1. INTRODUÇÃO	4
2. DESCRIÇÃO DA EMPRESA	5
3. PROJETO INTEGRADO	6
3.1 PROGRAMAÇÃO ORIENTADA A OBJETO	6
3.1.1 CLASSES E OBJETOS	6
3.1.2 ATRIBUTOS, MÉTODOS, ENCAPSULAMENTO, HERANÇA E POLIMORFISMO.	8
3.1.3 MÉTODOS ESTÁTICOS, PÚBLICOS E PRIVADOS	9
3.2 LÓGICA DE PROGRAMAÇÃO	10
3.2.1 CONCEITOS FUNDAMENTAIS DO DESENVOLVIMENTO DE SOFTWARE	10
3.2.2 DESENVOLVIMENTO DE APLICAÇÕES	11
3.2.3 IMPLEMENTAÇÃO E VALIDAÇÃO	12
3.3 MODELAGEM DE DADOS	14
3.3.1 MODELO CONCEITUAL	14
3.3.2 MODELO LÓGICO E FÍSICO	14
3.3.3 SQL	16
3.4 BUSINESS INTELLIGENCE	18
3.4.1 ORGANIZAÇÃO E IDENTIFICAÇÃO DAS INFORMAÇÕES	18
3.4.2 MANIPULAÇÃO E ANÁLISE DE DADOS	19
3.4.3 CRIAÇÃO DE MODELOS DE ANÁLISE DE DADOS	20
3.5 CONTEÚDO DA FORMAÇÃO PARA A VIDA: GERENCIANDO FINANÇAS	20
3.5.1 GERENCIANDO FINANÇAS	20
3.5.2 ESTUDANTES NA PRÁTICA	25
4. CONCLUSÃO	27
REFERÊNCIAS	28
ANEXOS	29

# 1. INTRODUÇÃO

O presente projeto visa desenvolver um sistema de gestão para uma locadora de carros, proporcionando uma solução eficiente e intuitiva para a administração de locações de veículos. Com a crescente demanda por serviços de transporte flexíveis e acessíveis, é essencial que a locadora possa gerenciar suas operações de forma eficaz, garantindo a satisfação dos clientes e a otimização dos recursos disponíveis.

De acordo com Stair e Reynolds (2010), sistemas de informação são essenciais para otimizar a gestão e a tomada de decisões em empresas, pois integram dados importantes e permitem maior controle sobre os processos operacionais e administrativos. O sistema permitirá o registro completo dos dados do veículo, incluindo marca, modelo, ano, categoria e condições de locação. Além disso, os dados do locatário serão registrados de forma segura, abrangendo informações como nome, documento de identificação, contato e histórico de locações. O projeto também contemplará a criação de diferentes tipos de contrato, adaptando-se às necessidades dos clientes, seja para locações de curto ou longo prazo.

Por meio de uma interface amigável, os usuários poderão inserir e consultar informações sobre o tempo de locação, tarifas aplicáveis e condições específicas de cada contrato, facilitando a tomada de decisões e o gerenciamento das operações. Este sistema não só aprimorará a experiência do cliente, mas também permitirá à locadora operar de maneira mais organizada e eficiente, contribuindo para seu crescimento sustentável no mercado.

## **2. DESCRIÇÃO DA EMPRESA**

A Localiza é uma das maiores e mais renomadas locadoras de veículos do Brasil, atuando no mercado desde sua fundação em 1973. A empresa é registrada sob a razão social Localiza Rent a Car S.A., com CNPJ 17.197.508/0001-80. Sua sede está localizada em Belo Horizonte, Minas Gerais, na Avenida Cristiano Machado, 1.400, bairro São Luiz.

A Localiza se destaca na atividade de locação de veículos, oferecendo uma ampla frota que abrange diferentes categorias, desde automóveis compactos até SUVs e vans. Com um compromisso contínuo com a excelência no atendimento ao cliente e inovação em seus serviços, a empresa também atua na área de gestão de frotas e no segmento de venda de veículos seminovos.

Com presença em diversas localidades do Brasil e em países da América Latina, a Localiza se consolidou como referência em mobilidade, proporcionando soluções práticas e acessíveis para clientes corporativos e particulares. Seu foco na qualidade do serviço e na satisfação do cliente a torna uma escolha confiável para quem busca locação de veículos com segurança e comodidade.

### **3. PROJETO INTEGRADO**

Nesta etapa do PI serão apresentados os conteúdos que cada unidade de estudo utilizará para realizar o projeto, assim como a forma que serão aplicados na empresa escolhida para a realização do projeto.

#### **3.1 PROGRAMAÇÃO ORIENTADA A OBJETO**

A Programação Orientada a Objetos (POO) é um paradigma de programação que permite a organização do código de maneira estruturada e modular, facilitando o desenvolvimento e a manutenção de software.

Por meio da POO, os estudantes aprendem a criar classes, que são moldes para a definição de objetos. Os objetos são instâncias dessas classes e representam entidades do mundo real, permitindo que o código seja mais intuitivo e alinhado com a lógica do domínio abordado (BUDD, 2005). A herança permite que uma classe herde características e comportamentos de outra, promovendo a reutilização de código e a criação de hierarquias lógicas. O polimorfismo, por sua vez, possibilita que diferentes classes sejam tratadas como instâncias de uma mesma classe base, permitindo a implementação de métodos que podem operar em objetos de diferentes tipos, aumentando a flexibilidade do código (LAFORE, 2002).

A abordagem da POO é especialmente relevante para o desenvolvimento de soluções que integram áreas como Business Intelligence e Modelagem de Dados. A POO permite o encapsulamento de dados e funcionalidades, garantindo que as informações sejam manipuladas de forma segura e eficiente. Isso é fundamental em aplicações que requerem análise de dados e operações complexas, já que a estrutura organizada favorece a integração de diferentes módulos e facilita a manutenção do software.

##### **3.1.1 CLASSES E OBJETOS**

No desenvolvimento do sistema de locadora de veículos, a utilização de classes e objetos foi fundamental para organizar e modularizar o código, facilitando a manutenção e a

escalabilidade da aplicação. A orientação a objetos permite modelar entidades do mundo real como clientes e veículos, encapsulando suas características e comportamentos em classes que interagem de maneira estruturada (PILONE, 2008).

A classe Cliente representa os usuários do sistema que podem alugar veículos. Cada objeto desta classe contém atributos como CPF, CNH e telefone, que identificam de forma única cada cliente. A classe também possui métodos que possibilitam a criação e manipulação dos dados dos clientes, como a validação de informações e a atualização de dados de contato. O uso dessa classe garante que o sistema consiga gerenciar os clientes de forma organizada, facilitando o controle sobre quem está autorizado a alugar veículos.

A classe Carro modela os veículos disponíveis para locação. Cada objeto Carro possui atributos como código, placa, e disponível, que indicam se o veículo está ou não alugado. Métodos como alugar() e devolver() são implementados nesta classe para mudar o estado do carro, permitindo que ele seja associado a um cliente durante o período de locação e liberado após a devolução. A estrutura da classe facilita o controle do inventário da locadora, garantindo que os carros possam ser gerenciados individualmente.

Para gerenciar as interações entre clientes e carros, foi criada a classe Locadora. Esta classe atua como um controlador, centralizando operações como cadastro de clientes e veículos, além dos processos de aluguel e devolução. A classe Locadora mantém listas de clientes e veículos, e implementa métodos para adicionar novos registros e realizar operações como alugar\_carro() e devolver\_carro(). A utilização desta classe permite que o sistema mantenha uma visão geral das operações, facilitando a gestão dos dados e das regras de negócios.

Cada uma dessas classes foi instanciada em objetos que representam os elementos reais da locadora, como clientes individuais e carros específicos. A criação de objetos a partir dessas classes permite que o sistema simule o comportamento de uma locadora de veículos, onde cada cliente e cada carro tem um papel específico nas operações diárias. Por exemplo, ao instanciar um objeto Cliente, é possível registrar um novo cliente na base de dados, e ao instanciar um objeto Carro, o sistema pode adicionar um novo veículo ao inventário.

O uso de orientação a objetos no desenvolvimento do sistema permite não apenas uma melhor organização do código, mas também facilita a expansão futura da aplicação, caso novas funcionalidades precisem ser adicionadas. A modularidade oferecida pelas classes e a clareza nas interações entre objetos garantem que o sistema possa evoluir de maneira controlada, atendendo às necessidades da locadora de veículos de forma eficiente e escalável.



### **3.1.2 ATRIBUTOS, MÉTODOS, ENCAPSULAMENTO, HERANÇA E POLIMORFISMO.**

No desenvolvimento do sistema de locadora de veículos, foram utilizados conceitos fundamentais de programação orientada a objetos, como atributos, métodos, encapsulamento, herança e polimorfismo, para estruturar a aplicação de maneira eficiente e organizada (PILONE, 2008).

Os atributos são características que definem o estado dos objetos. No sistema, cada classe possui atributos específicos: a classe Cliente contém cpf, cnh e telefone, que identificam e armazenam informações pessoais, enquanto a classe Carro possui codigo, placa e disponivel, que controlam a identificação e a disponibilidade dos veículos. Esses atributos são essenciais para representar as informações de cada entidade e possibilitar o controle das operações de locação.

Os métodos são funções dentro das classes que definem os comportamentos dos objetos. No projeto, cada classe possui métodos que manipulam seus atributos e realizam ações específicas. Por exemplo, a classe Cliente possui métodos como atualizar\_telefone(), que permite a modificação dos dados de contato, enquanto a classe Carro inclui métodos como alugar() e devolver(), que alteram o estado do veículo. Já a classe Locadora centraliza operações como adicionar\_cliente() e alugar\_carro(), gerenciando o fluxo de informações e garantindo que as operações sigam as regras de negócio estabelecidas.

O encapsulamento protege os dados das classes, restringindo o acesso direto aos atributos e promovendo a manipulação controlada por meio de métodos específicos. No sistema, muitos atributos são privados, sendo acessados apenas por métodos de leitura e escrita definidos nas classes, como get\_cpf() e set\_cnh() na classe Cliente. Isso assegura a integridade dos dados, evitando modificações indevidas e garantindo que as operações sejam realizadas de acordo com as regras definidas (LAFORE, 2002).

A herança permite a criação de novas classes a partir de uma classe base, facilitando a reutilização de código e a especialização de comportamentos. No projeto, a classe Carro pode ser uma superclasse para outras classes como CarroLuxo e CarroEconomico, que herdam seus atributos e métodos, mas podem implementar funcionalidades adicionais específicas. Isso possibilita a criação de novos tipos de veículos com características próprias, sem a necessidade de duplicar código, promovendo uma estrutura flexível e expansível.

O polimorfismo permite que objetos de diferentes classes derivadas de uma mesma superclasse respondem de maneiras distintas a chamadas de métodos comuns. No sistema,

isso é utilizado para que diferentes tipos de carros possam ser tratados como objetos da classe Carro, mas com comportamentos específicos. Métodos como `calcular_tarifa()` podem ser sobrescritos em subclasses para calcular diferentes tarifas conforme o tipo de veículo, permitindo que o sistema trate carros de luxo e econômicos de maneira adequada, mesmo usando uma interface comum.

Dessa forma, o uso desses conceitos de orientação a objetos no sistema da locadora de veículos possibilitou uma estrutura clara e eficiente. Os atributos e métodos permitiram modelar as entidades e suas operações, o encapsulamento garantiu a proteção dos dados, e a herança e o polimorfismo promoveram a flexibilidade e a reutilização de código. Isso resultou em uma aplicação organizada e preparada para futuras expansões.

### **3.1.3 MÉTODOS ESTÁTICOS, PÚBLICOS E PRIVADOS**

No sistema de locadora de veículos desenvolvido, a definição de métodos estáticos, públicos e privados foi fundamental para organizar o comportamento das classes e controlar o acesso às suas funcionalidades, seguindo boas práticas de programação orientada a objetos (PILONE, 2008).

Os métodos estáticos são aqueles que pertencem à classe em si, e não a instâncias específicas dessa classe. Eles podem ser chamados diretamente através do nome da classe, sem a necessidade de criação de um objeto. No projeto, métodos estáticos foram utilizados, por exemplo, para validações de dados, como a verificação de CPF e CNH. Esses métodos são úteis para operações que não dependem do estado individual dos objetos e são comumente utilizados para executar tarefas auxiliares ou utilitárias. No caso do sistema, o uso de um método estático para validar o CPF garante que o formato do dado seja verificado antes mesmo de um cliente ser instanciado.

Os métodos públicos são aqueles que podem ser acessados de qualquer parte do programa, permitindo a interação com as instâncias das classes. No projeto da locadora, a maioria dos métodos que realizam operações de negócios, como `alugar_carro()` e `devolver_carro()` na classe `Locadora`, são métodos públicos. Isso permite que esses métodos sejam chamados a partir de outras partes do sistema, possibilitando que os clientes realizem operações de locação e devolução de veículos de maneira controlada e segura. Métodos públicos são essenciais para definir a interface pública de uma classe, permitindo que outras partes do programa interajam com seus objetos de forma pré-determinada e adequada.

Os métodos privados são aqueles que só podem ser acessados dentro da própria classe, sendo invisíveis para o restante do programa. Eles são utilizados para operações internas que não devem ser acessadas diretamente por outros objetos. No sistema da locadora, métodos privados foram utilizados para manipular diretamente atributos sensíveis ou realizar operações internas de verificação e cálculo. Por exemplo, na classe Carro, um método privado pode ser utilizado para alterar o estado de disponibilidade do veículo de maneira controlada, sendo chamado apenas por métodos públicos como alugar() e devolver(). O uso de métodos privados assegura que a lógica interna da classe seja protegida, evitando modificações indevidas e garantindo que as regras de negócio sejam sempre respeitadas.

A utilização de métodos com diferentes níveis de acesso – estáticos, públicos e privados – permitiu a implementação de um sistema modular e seguro. Métodos estáticos foram utilizados para funções auxiliares e validações gerais, métodos públicos definiram a interface das operações permitidas nas classes, e métodos privados garantiram que as manipulações internas fossem realizadas de forma segura e consistente. Essa estrutura permitiu que o sistema funcionasse de maneira organizada e eficiente, protegendo a integridade dos dados e a lógica de negócios da aplicação.

## **3.2 LÓGICA DE PROGRAMAÇÃO**

A lógica de programação é fundamental no início do desenvolvimento de sistemas, pois estabelece a base para a construção de soluções computacionais eficientes. Os conceitos fundamentais incluem algoritmos, que são sequências de passos para resolver problemas; variáveis, que armazenam dados temporários e podem mudar durante a execução do programa; e tipos de dados, que em Python incluem inteiros, floats, strings e booleanos (LUTZ, 2013). Além disso, as funções permitem a modularização do código, facilitando a reutilização e manutenção do mesmo.

### **3.2.1 CONCEITOS FUNDAMENTAIS DO DESENVOLVIMENTO DE SOFTWARE**

A lógica de programação refere-se ao processo de resolução de problemas por meio de algoritmos, que são sequências de instruções organizadas para alcançar um objetivo. No caso do projeto de locadora de veículos, o algoritmo principal envolve funcionalidades como a

adição de clientes e carros, o aluguel e a devolução de veículos, e a listagem de carros disponíveis e alugados.

As variáveis, que armazenam informações necessárias para a execução do programa, desempenham um papel fundamental. No exemplo, são utilizadas para guardar dados como CPF, CNH, telefone dos clientes e os detalhes dos carros, como código e placa. Os tipos de dados utilizados variam, sendo strings, inteiros e booleanos, adequados conforme o tipo de informação que precisa ser armazenada.

As funções são essenciais para modularizar o código, tornando-o mais organizado e reutilizável. Funções como `adicionar_cliente()` e `alugar_carro()` ilustram essa organização, cada uma com responsabilidades específicas dentro do sistema. Além disso, estruturas condicionais são amplamente utilizadas para controlar o fluxo do programa, permitindo a tomada de decisões com base em verificações, como se um carro está disponível ou não.

Operadores também têm um papel central, como operadores de atribuição, comparação e lógicos, que permitem manipular e comparar dados. No exemplo, são utilizados em instruções que verificam se um carro está alugado e em atribuições de valores a variáveis. Estruturas de repetição, embora não utilizadas explicitamente no código, poderiam ser aplicadas para percorrer listas de dados de forma eficiente.

Portanto, esses conceitos fundamentais da lógica de programação, incluindo algoritmos, variáveis, tipos de dados, funções, estruturas condicionais e operadores, são essenciais para o desenvolvimento de sistemas como o da locadora de veículos, garantindo sua funcionalidade e organização.

### **3.2.2 DESENVOLVIMENTO DE APLICAÇÕES**

As regras de negócios de uma aplicação definem como os processos internos devem ocorrer para atender aos objetivos organizacionais. No projeto da locadora de veículos, essas regras determinam a lógica de cadastro, aluguel, devolução e consulta de veículos e clientes. Elas são implementadas por meio de funções que asseguram o funcionamento adequado da aplicação, seguindo um conjunto de critérios e condições específicas.

No sistema, o cadastro de clientes é uma regra fundamental, pois para alugar um veículo o cliente deve estar registrado com CPF, CNH e telefone. A função `adicionar_cliente()` cuida desse processo, garantindo que todas as informações necessárias sejam fornecidas e o cliente seja adicionado à base de dados.

Outro aspecto importante é o cadastro de veículos, feito pela função `adicionar_carro()`, que armazena os carros disponíveis para locação, identificando-os por código e placa. Cada carro cadastrado inicialmente está disponível para aluguel, o que prepara o sistema para gerenciar os veículos de forma eficiente.

A locação de veículos segue a regra de que um carro só pode ser alugado se estiver disponível e o cliente estiver cadastrado. A função `alugar_carro()` verifica essas condições antes de permitir que o carro seja associado a um cliente por um período de locação específico. Essa função evita conflitos, como tentar alugar um carro já alugado.

A devolução de veículos é tratada pela função `devolver_carro()`, que torna o carro novamente disponível após o término do período de locação. Esse processo mantém o controle sobre os veículos que estão ou não em uso, e é essencial para liberar os carros para novos alugueis.

A consulta de carros disponíveis é implementada na função `listar_carros_disponiveis()`, que retorna uma lista de todos os veículos que estão prontos para locação. Isso facilita o trabalho da equipe ao mostrar quais carros podem ser alugados. Similarmente, a consulta de carros alugados, realizada pela função `listar_carros_alugados()`, exibe os veículos que estão em uso, juntamente com as informações dos clientes que os alugaram.

Cada funcionalidade é organizada em módulos que compõem a estrutura do sistema, permitindo que os processos da locadora sejam gerenciados de forma eficiente. Esses módulos incluem o cadastro de clientes e veículos, o processo de aluguel e devolução, e as consultas de disponibilidade e uso dos veículos. A modularização do código facilita a manutenção e possibilita a expansão futura da aplicação.

Portanto, as regras de negócios e os módulos desenvolvidos asseguram o controle e a organização do sistema da locadora, permitindo que as operações sejam realizadas com base nas necessidades e objetivos da empresa. O sistema é eficiente e fácil de gerenciar, garantindo uma execução clara e precisa de cada funcionalidade.

### **3.2.3 IMPLEMENTAÇÃO E VALIDAÇÃO**

A implementação dos módulos do sistema de locadora de veículos é essencial para assegurar que as funcionalidades planejadas operem de maneira eficiente. A lógica do sistema foi organizada em módulos específicos, responsáveis por cada etapa do processo, como cadastro, aluguel, devolução e consulta de veículos e clientes, conforme estabelecido pelas regras de negócios (PRESSMAN, 2011).

O primeiro módulo desenvolvido foi o cadastro de clientes, realizado pela função `adicionar_cliente()`, que armazena dados obrigatórios como CPF, CNH e telefone. Esse módulo evita duplicidade no registro de clientes, uma regra fundamental para o correto funcionamento do sistema. Em seguida, foi implementado o cadastro de veículos, através da função `adicionar_carro()`, que insere carros na base de dados, identificando-os por código e placa, garantindo que estejam disponíveis para aluguel após o cadastro.

A locação de veículos é uma das funcionalidades centrais do sistema, implementada pela função `alugar_carro()`, que verifica se o cliente está cadastrado e se o veículo está disponível para aluguel. Essa função garante que o carro não seja alugado para mais de um cliente ao mesmo tempo. Após a locação, a devolução de veículos é realizada pela função `devolver_carro()`, que libera o carro para ser alugado novamente, controlando eficientemente a frota disponível.

Para fornecer visibilidade sobre o status dos veículos, foram desenvolvidos os módulos de consulta de carros disponíveis e carros alugados, através das funções `listar_carros_disponiveis()` e `listar_carros_alugados()`. Esses módulos permitem que os usuários verifiquem quais veículos estão prontos para aluguel ou quais estão atualmente em uso.

Após a implementação, foram realizados testes de unidade para garantir que cada função funcionasse corretamente de forma individual (BEIZER, 1990). Por exemplo, os testes no cadastro de clientes e veículos validaram a inserção correta de dados e a verificação de duplicidades. Em seguida, o módulo de locação foi testado para assegurar que apenas veículos disponíveis fossem alugados por clientes registrados. A função de devolução foi validada para garantir que o carro fosse novamente disponibilizado após a devolução.

Os testes de integração verificaram se os módulos funcionam adequadamente em conjunto, garantindo que o sistema operasse de maneira integrada, desde o cadastro até a devolução e consulta de veículos. Por fim, os testes de aceitação foram realizados com base nos requisitos do projeto, simulando cenários reais de operação de uma locadora. Esses testes confirmaram que o sistema atendia a todas as exigências funcionais e operacionais.

Concluindo, a implementação dos módulos e sua validação por meio de testes garantiu que o sistema de locadora de veículos funcionasse de acordo com os requisitos, proporcionando uma gestão eficiente dos processos de cadastro, aluguel e devolução de veículos.

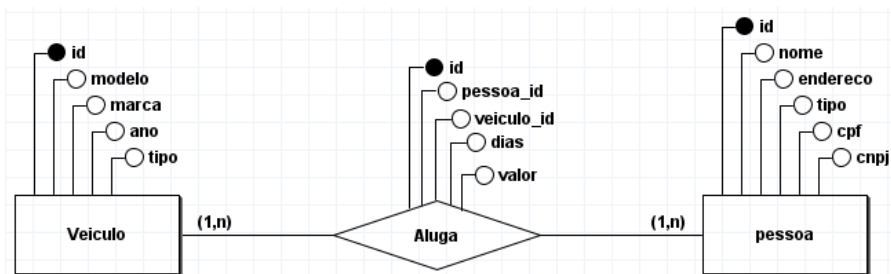
### 3.3 MODELAGEM DE DADOS

Modelagem de dados é o processo de criar uma representação visual das informações e seus relacionamentos dentro de um sistema. Segundo Teorey, Yang e Fry (2011), a modelagem de dados "ajuda a abstrair e organizar informações complexas, permitindo uma compreensão mais clara do sistema". Esse processo é essencial na criação de bancos de dados, pois facilita a análise, o design e a implementação das estruturas de dados necessárias, garantindo consistência e integridade nas informações (Harrington, 2016).

#### 3.3.1 MODELO CONCEITUAL

Este modelo apresenta a estrutura básica do banco de dados, destacando as principais entidades envolvidas (Pessoa, Veículo e Aluguel) e os relacionamentos entre elas. No modelo conceitual, cada entidade é descrita com suas propriedades essenciais, como id, nome, endereço, tipo, e documentos (CPF ou CNPJ) para Pessoa; modelo, marca, ano, e tipo para Veículo; e dias e valor para Aluguel, com as devidas chaves estrangeiras para Pessoa e Veículo.

**Imagem 1 - Modelo Conceitual**

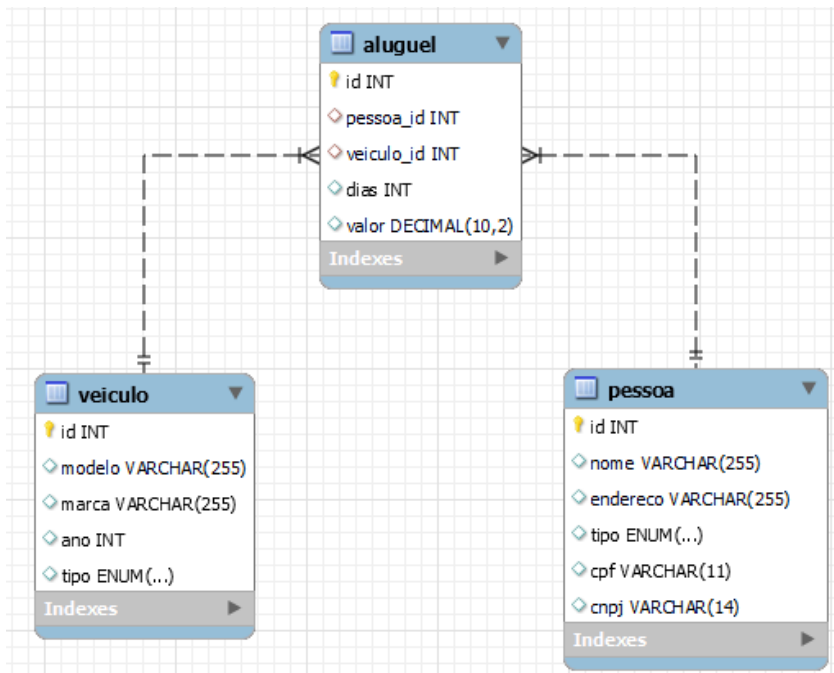


Fonte: Autores.

#### 3.3.2 MODELO LÓGICO E FÍSICO

Este modelo detalha as tabelas e os tipos de dados do banco de dados SQL. Ele inclui a criação do banco de dados "locadora" e as tabelas (Pessoa, Veículo e Aluguel), com seus atributos específicos, tipos de dados e relacionamentos (chaves primárias e estrangeiras). O Modelo Lógico implementa o armazenamento de informações da locadora com comandos SQL e insere dados exemplares em cada tabela, permitindo o gerenciamento das informações de clientes, veículos e aluguéis.

Imagem 2 - Modelo Lógico



Fonte: Autores.

O modelo físico apresentado no documento define a estrutura e implementação de um banco de dados relacional para uma locadora, utilizando SQL para criar as tabelas e definir os relacionamentos entre elas. Este modelo inclui as seguintes tabelas:

1. **Pessoa:** Armazena informações sobre as pessoas, com colunas para id, nome, endereço, tipo (pessoa física ou jurídica), cpf e cnj.
2. **Veículo:** Registra os veículos disponíveis para aluguel, com colunas para id, modelo, marca, ano e tipo (carro, moto ou caminhão).
3. **Aluguel:** Representa os alugueis realizados, com id, **pessoa\_id** (chave estrangeira para a tabela Pessoa), **veiculo\_id** (chave estrangeira para a tabela Veículo), dias de aluguel e valor.



### Imagem 3 - Modelo Físico

```

CREATE DATABASE locadora;

USE locadora;

CREATE TABLE Pessoa (
  id INT AUTO_INCREMENT PRIMARY KEY,
  nome VARCHAR(255),
  endereco VARCHAR(255),
  tipo ENUM('Fisica', 'Juridica'),
  cpf VARCHAR(11),
  cnpj VARCHAR(14)
);

CREATE TABLE Veiculo (
  id INT AUTO_INCREMENT PRIMARY KEY,
  modelo VARCHAR(255),
  marca VARCHAR(255),
  ano INT,
  tipo ENUM('CARRO', 'MOTO', 'CAMINHAO')
);

CREATE TABLE Aluguel (
  id INT AUTO_INCREMENT PRIMARY KEY,
  pessoa_id INT,
  veiculo_id INT,
  dias INT,
  valor DECIMAL(10, 2),
  FOREIGN KEY (pessoa_id) REFERENCES Pessoa(id),
  FOREIGN KEY (veiculo_id) REFERENCES Veiculo(id)
);

```

Fonte: Autores.

#### 3.3.3 SQL

Primeiramente, utiliza-se o comando `CREATE DATABASE locadora;` para criar o banco de dados e, em seguida, o comando `USE locadora;` para selecionar esse banco para as operações subsequentes.

Para organizar as informações das pessoas, é criada a tabela "Pessoa" por meio do comando `CREATE TABLE Pessoa`, que contém as colunas `id`, `nome`, `endereco`, `tipo`, `cpf` e `cnpj`. O campo `id` é uma chave primária (`PRIMARY KEY`) e é definido como `AUTO_INCREMENT`, assegurando que cada entrada seja única. A coluna `tipo` utiliza a estrutura `ENUM`, restringindo os valores a "Fisica" ou "Juridica".

Na sequência, é criada a tabela "Veiculo" com o comando `CREATE TABLE Veiculo`, que contém as colunas `id`, `modelo`, `marca`, `ano` e `tipo`. Da mesma forma, o campo `id` é uma chave primária com `AUTO_INCREMENT`. A coluna `tipo` utiliza `ENUM` para categorizar os veículos como "CARRO", "MOTO" ou "CAMINHAO".

O documento também inclui a criação da tabela "Aluguel", que organiza as transações de aluguel. Com o comando CREATE TABLE Aluguel, são definidas as colunas id, pessoa\_id, veiculo\_id, dias e valor, sendo id uma chave primária e pessoa\_id e veiculo\_id chaves estrangeiras, estabelecendo relacionamentos com as tabelas "Pessoa" e "Veiculo". A coluna valor é do tipo DECIMAL(10,2), permitindo precisão em valores monetários.

Para inserir dados iniciais, utiliza-se o comando INSERT INTO, adicionando registros nas tabelas. Na tabela "Pessoa", são incluídos três registros, representando duas pessoas físicas e uma jurídica. A tabela "Veiculo" recebe três registros que representam um carro, uma moto e um caminhão. Finalmente, na tabela "Aluguel" são inseridos três registros que associam pessoas e veículos, especificando os dias de aluguel e os valores cobrados.

Este conjunto de comandos SQL configura o banco de dados com uma estrutura básica para operação de uma locadora, de acordo com o modelo relacional, e fornece dados iniciais para simulação das operações.

#### Imagem 4 - Insert, Update, Delete e Select

```
-- Inserindo dados na tabela Pessoa
INSERT INTO Pessoa (nome, endereco, tipo, cpf, cnpj) VALUES
('João da Silva', 'Rua A, 123', 'Fisica', '12345678901', NULL),
('Maria Oliveira', 'Avenida B, 456', 'Fisica', '98765432100', NULL),
('Empresa X', 'Rua C, 789', 'Juridica', NULL, '12345678000195');

-- Inserindo dados na tabela Veiculo
INSERT INTO Veiculo (modelo, marca, ano, tipo) VALUES
('Fusca', 'Volkswagen', 1975, 'CARRO'),
('CB 500', 'Honda', 2020, 'MOTO'),
('Caminhão 1113', 'Mercedes-Benz', 1990, 'CAMINHAO');

-- Inserindo dados na tabela Aluguel
INSERT INTO Aluguel (pessoa_id, veiculo_id, dias, valor) VALUES
(1, 1, 5, 150.00),
(2, 2, 3, 200.00),
(3, 3, 7, 800.00);

-- DELETE
DELETE FROM Pessoa WHERE nome = 'João da Silva';
DELETE FROM Veiculo WHERE modelo = 'Fusca';
DELETE FROM Aluguel WHERE id = 1;

-- UPDATE
UPDATE Pessoa SET endereco = 'Rua Nova, 456' WHERE nome = 'Maria Oliveira';
UPDATE Aluguel SET valor = 180.00 WHERE pessoa_id = 2 AND veiculo_id = 2;
UPDATE Veiculo SET ano = 1985 WHERE modelo = 'Fusca';

-- SELECT
SELECT * FROM Pessoa;
SELECT * FROM Veiculo WHERE tipo = 'CARRO';
SELECT Pessoa.nome, Veiculo.modelo, Aluguel.dias, Aluguel.valor
FROM Aluguel
INNER JOIN Pessoa ON Aluguel.pessoa_id = Pessoa.id
INNER JOIN Veiculo ON Aluguel.veiculo_id = Veiculo.id;
SELECT * FROM Aluguel WHERE valor > 200.00;
```

### 3.4 BUSINESS INTELLIGENCE

Business Intelligence (BI) refere-se ao conjunto de práticas, tecnologias e ferramentas que possibilitam a transformação de dados brutos em informações significativas, fornecendo apoio à tomada de decisões informadas dentro das organizações. Sua importância reside na capacidade de gerar insights valiosos sobre o desempenho empresarial, identificar tendências de mercado e entender o comportamento dos consumidores. Tais informações possibilitam a criação de estratégias mais eficazes, a otimização de processos e, conseqüentemente, o alcance de uma vantagem competitiva no mercado (DUNHAM, 2009).

Dentro deste contexto, o Power BI é uma das ferramentas mais populares e robustas disponíveis no mercado para BI. Este software se destaca por sua interface intuitiva, pela capacidade de integrar-se com diversas fontes de dados e por suas funcionalidades avançadas de visualização (MICROSOFT, 2020). Além disso, o Power BI permite a criação de dashboards interativos e relatórios dinâmicos, facilitando a análise de dados em tempo real e proporcionando um ambiente propício para a tomada de decisões baseadas em evidências. Ao adotar o Power BI, as empresas não apenas potencializam a análise de dados, mas também promovem uma cultura organizacional focada em decisões fundamentadas em informações concretas (GILL, 2020).

#### 3.4.1 ORGANIZAÇÃO E IDENTIFICAÇÃO DAS INFORMAÇÕES

O dashboard desenvolvido no Power BI com base nos dados do sistema da locadora de veículos apresenta diversas informações relevantes para o monitoramento e análise do desempenho dos alugueis de veículos. Vamos detalhar cada seção exibida no painel:

##### 1. Tabela de Clientes e Alugueis

Localizada na parte esquerda, a tabela lista informações dos clientes, o tipo de cliente (Física ou Jurídica), o valor total do aluguel, e a quantidade de dias em que cada cliente manteve o veículo alugado. Isso permite uma visão detalhada dos principais clientes e do valor gerado por cada um.

##### 2. Mapa de Localização dos Clientes

O mapa, exibido no centro superior, mostra os locais de origem dos clientes, com pontos marcados na localização dos endereços registrados. Essa informação facilita a identificação da dispersão geográfica dos clientes e pode ajudar a definir estratégias regionais.

##### 3. Meta de Faturamento

No canto superior direito, há um gráfico em forma de indicador que indica o progresso em relação a uma meta de faturamento estabelecida em R\$ 33 mil. Neste caso, o valor total acumulado até o momento é de R\$ 13 mil, mostrando que a meta ainda não foi atingida, mas fornecendo uma visão de progresso.

#### **4. Valor Total de Alugueis**

Logo abaixo do gráfico de indicador, o painel destaca o valor total de faturamento dos alugueis, que é de R\$ 13 mil. Este valor resume o montante arrecadado pela locadora até o momento, somando todos os alugueis registrados.

#### **5. Valor por Tipo de Veículo**

Abaixo do mapa, o gráfico de barras exhibe a soma dos valores de aluguel segmentados por tipo de veículo: Carro, Caminhão e Moto. Neste caso, Carros geraram o maior faturamento (R\$ 8,4 mil), seguidos por Caminhões (R\$ 3,8 mil) e Motos (R\$ 0,8 mil). Esse tipo de análise auxilia na compreensão de quais categorias de veículos são mais lucrativas.

#### **6. Soma de Dias por Modelo**

No canto inferior direito, um gráfico de barras mostra a soma dos dias em que cada modelo foi alugado. Veículos como o Fusca e o Civic são os mais populares, com 17 e 15 dias alugados, respectivamente. Essa análise permite identificar os modelos mais requisitados e pode ajudar no planejamento de aquisição ou manutenção de veículos mais populares.

### **3.4.2 MANIPULAÇÃO E ANÁLISE DE DADOS**

Os dados apresentados referem-se a uma locadora de veículos, organizada em três tabelas principais: Pessoa, Veiculo e Aluguel. Cada tabela contém informações cruciais para a operação e análise do negócio.

A tabela Pessoa armazena os dados dos clientes, diferenciando entre pessoas físicas e jurídicas, e inclui informações como nome, endereço e documentos de identificação (CPF ou CNPJ). Isso permite que a locadora mantenha um registro claro dos seus clientes, facilitando a comunicação e a gestão de relacionamento.

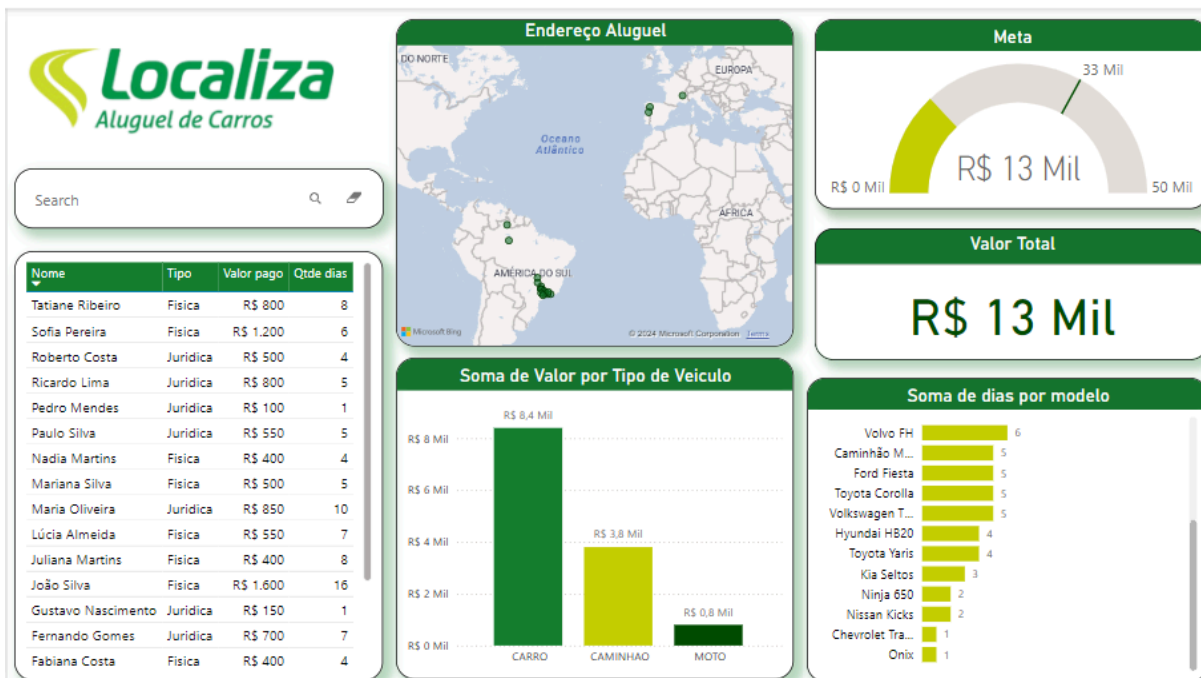
A tabela Veiculo contém informações sobre os veículos disponíveis para locação, incluindo modelo, marca, ano e tipo (carro, moto ou caminhão). Esses dados são essenciais para a gestão da frota, permitindo à locadora entender quais veículos estão disponíveis, suas características e a melhor adequação às necessidades dos clientes.

A tabela Aluguel relaciona os clientes com os veículos alugados, registrando detalhes como o número de dias de locação e o valor total do aluguel. Essas informações são fundamentais para acompanhar a performance financeira da locadora, permitindo o cálculo de

receitas, análise de frequência de aluguel por tipo de veículo e identificação de tendências de uso.

### 3.4.3 CRIAÇÃO DE MODELOS DE ANÁLISE DE DADOS

Imagem 5 - Power BI (Dashboard)



Fonte: Autores.

## 3.5 CONTEÚDO DA FORMAÇÃO PARA A VIDA: GERENCIANDO FINANÇAS

A Formação para a Vida é um dos eixos do Projeto Pedagógico de Formação por Competências da UNIFEQB.

Esta parte do projeto está diretamente relacionada com a extensão universitária, ou seja, o objetivo é que seja aplicável e que tenha real utilidade para a sociedade, de um modo geral.

### 3.5.1 GERENCIANDO FINANÇAS

Está disponível para os estudantes no Classroom, o tema “Gerenciando Finanças”.

Nesta parte do Projeto, os estudantes deverão realizar uma síntese dos 4 (quatro) tópicos deste tema, quais sejam:

- **Tópico 1: Introdução aos conceitos econômicos e financeiros básicos**

- 1. Dinheiro como meio de troca:**

- Facilita a troca de bens e serviços.

- 2. Relação entre Finanças, Contabilidade e Economia:**

- Finanças: Estudo de gestão de recursos financeiros em empresas e indivíduos.
- Contabilidade: Registro de transações e variações patrimoniais.
- Economia: Estudo da produção, circulação e consumo de bens e serviços.

- 3. Economia:**

- Microeconomia: Análise do mercado de produtos específicos e comportamento de compradores e vendedores.
- Macroeconomia: Estudo do funcionamento da economia como um todo e variáveis como produção e preços.

- 4. Contabilidade:**

- Representa a situação financeira das empresas em um determinado momento.
- Diferença entre regimes de caixa (movimentação de dinheiro) e competência (fato gerador).

- 5. Classificação dos Gastos:**

- Gasto: Sacrifício financeiro para alcançar um objetivo.
- Desembolso: Saída de recurso financeiro.

- 6. Exemplos Práticos:**

- Investimento: Aquisições para uso próprio (ex: computadores e móveis).
- Custo: Gastos com mercadorias que contribuem para a geração de receita.
- Despesa: Gastos operacionais (ex: aluguel).

- 7. Gestão de Custos Pessoais:**

- Aplicação de conceitos financeiros em finanças pessoais.
- Controle de receitas e despesas para evitar déficits.

- 8. Formas de Controlar Gastos:**

- Classificação de Despesas:

- Despesas com ocupação (aluguéis, contas).
- Despesas com serviços profissionais (advogados, contadores).
- Despesas diversas (entretenimento, viagens).
- Despesas com pessoal (salários, benefícios).

- 9. Decisões Financeiras:**

- Controle de gastos permite tomar decisões sobre investimentos.

- Fluxo de Caixa: Análise das entradas e saídas financeiras.
  - Fórmula: saldo inicial + entradas - saídas = saldo final.

## **10. Importância do Controle Financeiro:**

- Conhecer custos é essencial para a saúde financeira de empresas e indivíduos.
  - Demonstrações contábeis ajudam na tomada de decisões financeiras.
- **Tópico 2:** Entendendo o ambiente: independência financeira, o valor da minha riqueza e o registro do dia a dia

### **1. Independência financeira e gestão financeira:**

- Maximizar a riqueza do acionista é um princípio da administração financeira.
- Importância de gerenciar finanças pessoais como uma empresa.

### **2. Gestão financeira:**

- Busca liquidez, menores custos e otimização de resultados.
- Liquidez: valor monetário após pagamento de despesas.
- Necessidade de decisões sábias ao investir.

### **3. Fontes de rendimentos:**

- Geração de caixa para empresas: faturamento de bens ou serviços.
- Geração de renda para indivíduos: emprego ou empreendedorismo.
- Investimentos: utilizar o que sobra após despesas mensais.

### **4. Tipos de investimentos:**

- Imóveis: gerar renda de aluguel e valorização ao longo do tempo.
- Ações: aquisição de ações de empresas, retorno através de dividendos.
- Direitos autorais: renda pela venda de produções criativas.

### **5. Redução de custos:**

- Disciplina necessária para alcançar independência financeira.
- Controle de gastos é essencial.
- Necessidade de conhecimento sobre entradas e saídas mensais.

### **6. Estratégias de controle de gastos:**

- Evitar dívidas bancárias devido a altas taxas de juros.
- Renegociar com fornecedores para melhores taxas.
- Controlar gastos e otimizar processos (ex: digitalização).
- Verificar despesas com menor relevância.

### **7. Conceitos de investimento:**

- Vender ativos para reduzir gastos fixos.

- Investimentos em bens permanentes (ex: imóveis) e ativos financeiros (ex: CDBs, ações).

#### **8. Perfis de investidor:**

- Conservador: avesso a riscos, prefere investimentos seguros.
- Moderado: diversificação entre renda fixa e ações.
- Agressivo: maior risco em renda variável, acompanhamento constante do mercado.

#### **9. Demonstrações financeiras:**

- Importância de relatórios financeiros periódicos para análise.
- Empresas elaboram balanço patrimonial, demonstração de resultado e fluxo de caixa.
- Necessidade de controle financeiro pessoal, categorizando entradas e saídas.

- **Tópico 3:** Dívidas e juros compostos, opções de empréstimo e alternativas ao endividado

#### **1. Matemática Financeira:**

- Estudo do valor do dinheiro no tempo.
- Análise de fluxos de caixa: entradas (setas para cima) e saídas (setas para baixo).

#### **2. Juros Simples:**

- Cálculo baseado apenas no capital inicial.
- Fórmula:  $V F = V P + V P \times i \times n$  ou  $F V = V P \times ( 1 + i \times n )$   
 $FV=VP \times (1+i \times n)$ .
- Exemplo: R\$ 1.000 a 10% por 3 meses resulta em R\$ 1.300.

#### **3. Juros Compostos:**

- Juros calculados sobre o montante acumulado (juros sobre juros).
- Fórmula:  $V F = V P \times ( 1 + i ) ^ n$  ou  $F V = V P \times ( 1 + i ) ^ n$ .
- Exemplo: R\$ 1.000 a 10% por 3 meses resulta em R\$ 1.331.
- Usado amplamente em investimentos e empréstimos.

#### **4. Possibilidades de Crédito:**

- Análise de crédito: coleta de informações e avaliação de garantias.
- Fatores: histórico, patrimônio, capacidade de geração de caixa.
- Importância de comparar taxas de juros e modalidades de crédito.

#### **5. Organização Financeira:**



- Controle das finanças: entradas e saídas.
- Definição de prioridades e elaboração de um orçamento.
- Automação de pagamentos para evitar multas e juros.

#### **6. Educação Financeira:**

- Importância de entender conceitos financeiros (juros, inflação, matemática financeira).
- Ferramentas e planilhas orçamentárias disponíveis online.

#### **7. Planejamento Orçamentário:**

- Ciclo de planejamento e controle para alcançar metas financeiras.
- Relação entre planejamento financeiro e estratégico.

- **Tópico 4:** Estabelecer metas para a realização de seus sonhos e como envolver o grupo a que você pertence para atingir seus objetivos

#### **1. Estabelecimento de Metas e Sonhos:**

- Importância de um futuro financeiro sólido e uma aposentadoria tranquila.
- Necessidade de planejar ações no presente para alcançar um padrão de vida desejado.

#### **2. Gestão Financeira:**

- A gestão financeira pessoal é similar à gestão de uma empresa.
- Importância do planejamento inicial e controle contínuo das finanças.
- Uso de fluxo de caixa para prever entradas e saídas.
- Necessidade de uma reserva financeira para imprevistos.

#### **3. Sonhos e Projetos:**

- Sonhos variam em complexidade e custo; todos devem ser priorizados.
- Transformação de sonhos em projetos claros com prazos definidos.
- Exemplo prático: planejamento de uma viagem e suas variáveis financeiras.

#### **4. Mitos nas Finanças Pessoais:**

- Investimentos: Qualquer quantia pode ser investida; poupança é um tipo de investimento acessível.
- Cartão de crédito: Pode ser utilizado de forma responsável com controle de gastos.
- Gastar o que ganha: Imprevistos podem afetar a saúde financeira; recomenda-se reservar uma parte da renda.

#### **5. Atitudes para o Sucesso Financeiro:**

- Definição clara de objetivos e estratégias.
- Importância do planejamento e do controle financeiro diário.

#### **6. Opções para Aposentadoria:**

- Preparação financeira para a aposentadoria é crucial.
- Opções incluem fundos de aposentadoria oferecidos por empresas e previdência privada.
- Exemplo de poupança a longo prazo: a poupança mensal pode gerar um montante significativo ao se aposentar.

### **3.5.2 ESTUDANTES NA PRÁTICA**

Gerenciar melhor as finanças pessoais tem um impacto profundo na qualidade de vida, proporcionando maior segurança e tranquilidade. Segundo Cerbasi (2014), o controle adequado do orçamento permite a criação de uma reserva financeira para imprevistos, reduz o endividamento e facilita a realização de metas a longo prazo, como a compra de bens ou a aposentadoria. Além disso, o planejamento financeiro possibilita uma melhor gestão do consumo, evitando gastos desnecessários e criando hábitos de poupança e investimento.

Desse modo, o equilíbrio financeiro contribui para a melhoria do bem-estar emocional e aumenta a capacidade de lidar com desafios financeiros de maneira mais eficiente.

Este guia apresenta 6 estratégias essenciais para te auxiliar nessa jornada:

- 1. Crie um orçamento detalhado:** Faça um planejamento mensal das suas receitas e despesas. Liste todas as suas fontes de renda e os gastos fixos e variáveis. Isso te ajuda a visualizar para onde o dinheiro está indo e a identificar áreas onde você pode economizar.
- 2. Estabeleça metas financeiras claras:** Tenha objetivos financeiros, como pagar dívidas, fazer uma reserva de emergência ou investir para o futuro. Metas claras orientam suas decisões de gasto e poupança.
- 3. Pague-se primeiro:** Antes de gastar com outras coisas, reserve uma parte do seu dinheiro para poupança ou investimentos. Isso cria o hábito de economizar regularmente e evita que você gaste tudo.
- 4. Reduza despesas desnecessárias:** Analise seus gastos e veja onde pode cortar custos. Assinaturas que você quase não usa, compras por impulso e refeições fora de casa são áreas comuns para ajustes.

5. **Invista no seu conhecimento financeiro:** Aprender sobre finanças pessoais, investimentos e planejamento financeiro pode te ajudar a tomar decisões mais informadas. Ler livros, assistir vídeos ou consultar um especialista são boas opções.
6. **Crie um fundo de emergência:** Reserve uma quantia equivalente a, no mínimo, três a seis meses de suas despesas mensais em uma conta de fácil acesso. Esse fundo te protege em situações inesperadas, como perda de emprego ou emergências médicas, evitando que você precise recorrer a empréstimos ou ao uso do cartão de crédito.

Assim, ao implementar essas estratégias, estaremos aptos a ter um controle melhor sobre nosso dinheiro, tendo uma vida mais balanceada e melhor controlada.

**Imagem 8 - Banner Formação para Vida**

**COMO GERENCIAR MELHOR SUAS FINANÇAS**

**Invista no seu conhecimento financeiro**

Aprender sobre finanças pessoais, investimentos e planejamento financeiro pode te ajudar a tomar decisões mais informadas. Ler livros, assistir vídeos ou consultar um especialista são boas opções.

**Crie um orçamento detalhado**

Faça um planejamento mensal das suas receitas e despesas. Liste todas as suas fontes de renda e os gastos fixos e variáveis. Isso te ajuda a visualizar para onde o dinheiro está indo e a identificar áreas onde você pode economizar.

**Reduza despesas desnecessárias**

Analise seus gastos e veja onde pode cortar custos. Assinaturas que você quase não usa, compras por impulso e refeições fora de casa são áreas comuns para ajustes.

**Estabeleça metas financeiras claras**

Tenha objetivos financeiros, como pagar dívidas, fazer uma reserva de emergência ou investir para o futuro. Metas claras orientam suas decisões de gasto e poupança.

**Pague-se primeiro**

Antes de gastar com outras coisas, reserve uma parte do seu dinheiro para poupança ou investimentos. Isso cria o hábito de economizar regularmente e evita que você gaste tudo.

**Crie um fundo de emergência**

Reserve uma quantia equivalente a, no mínimo, três a seis meses de suas despesas mensais em uma conta de fácil acesso. Esse fundo te protege em situações inesperadas, como perda de emprego ou emergências médicas, evitando que você precise recorrer a empréstimos ou ao uso do cartão de crédito.

## 4. CONCLUSÃO

A conclusão deste projeto de desenvolvimento de um sistema de gestão para uma locadora de carros ressalta a importância da implementação de uma solução integrada que atenda às demandas contemporâneas do setor. O sistema foi projetado para otimizar a administração de locações, promovendo uma experiência do cliente mais satisfatória e eficiente.

Os principais pontos abordados incluem o registro detalhado de veículos e locatários, assegurando um controle rigoroso sobre a disponibilidade e o histórico de locações, o que é crucial para a gestão de recursos. A criação de diferentes tipos de contratos oferece adaptabilidade às necessidades dos clientes, abrangendo locações de curto e longo prazo. Além disso, a interface amigável facilita o uso por parte dos funcionários, reduzindo a curva de aprendizado e aumentando a eficiência operacional. A disponibilização de informações em tempo real sobre tarifas e condições de locação permite decisões mais informadas e rápidas, melhorando o atendimento ao cliente.

Entretanto, algumas dificuldades foram encontradas durante o desenvolvimento do projeto. Primeiramente, a conexão foi prejudicada por configurações inadequadas. Além disso, a busca de informações acabou sendo comprometida por consultas ineficientes, falta de índices adequados e dados inconsistentes, resultando em respostas imprecisas. A gestão de erros é outro ponto crítico, pois a ausência de tratamento adequado de exceções e logs insuficientes dificultaram a identificação de problemas. Por fim, a experiência do usuário pode ser afetada por uma interface não intuitiva.

Em suma, este projeto não apenas oferece uma solução tecnológica robusta para a locadora de carros, mas também serve como um modelo de inovação no setor de locação, alinhando-se às tendências de mercado e às expectativas dos consumidores. Com a implementação deste sistema, espera-se que a locadora alcance maior eficiência, reduza custos operacionais e, conseqüentemente, se posicione de maneira competitiva no mercado. A continuidade na avaliação e no aprimoramento do sistema será crucial para sua efetividade a longo prazo.

## REFERÊNCIAS

- BEIZER, Boris. Software Testing Techniques. 2. ed. New York: Van Nostrand Reinhold, 1990.
- BUDD, T. Understanding Object-Oriented Programming with Java. 2. ed. Boston: Addison-Wesley, 2005.
- CERBASI, Gustavo. Adeus, aposentadoria. São Paulo: Sextante, 2014.
- DUNHAM, M. H. Data mining: introductory and advanced topics. Pearson Education, 2009.
- GILL, S. Power BI: Dashboards, Reports & DAX. 2. ed. Wiley, 2020.
- HARRINGTON, John L. Relational Database Design Clearly Explained. 3. ed. Morgan Kaufmann, 2016.
- LAFORE, Robert. Programação Orientada a Objetos em Java. 3. ed. São Paulo: Bookman, 2002.
- LAFORE, R. Object-Oriented Programming in C++. 4. ed. Englewood Cliffs: Sams Publishing, 2002.
- LUTZ, M. Learning Python. 5. ed. Sebastopol: O'Reilly Media, 2013.
- MICROSOFT. Power BI Documentation. 2020. Disponível em: <https://docs.microsoft.com/en-us/power-bi/>. Acesso em: 10 nov. 2024.
- PILONE, Dan. UML 2.0 in a Nutshell. 1st ed. Sebastopol, CA: O'Reilly Media, 2008.
- PRESSMAN, Roger S. Engenharia de Software. 7. ed. Porto Alegre: McGraw-Hill, 2011.
- STAIR, Ralph M.; REYNOLDS, George W. Princípios de Sistemas de Informação: uma abordagem gerencial. 6. ed. São Paulo: Cengage Learning, 2010.
- TEOREY, Toby J.; YANG, Fang; FRY, Jim. Database Modeling and Design: Logical Design. 4. ed. Morgan Kaufmann, 2011.

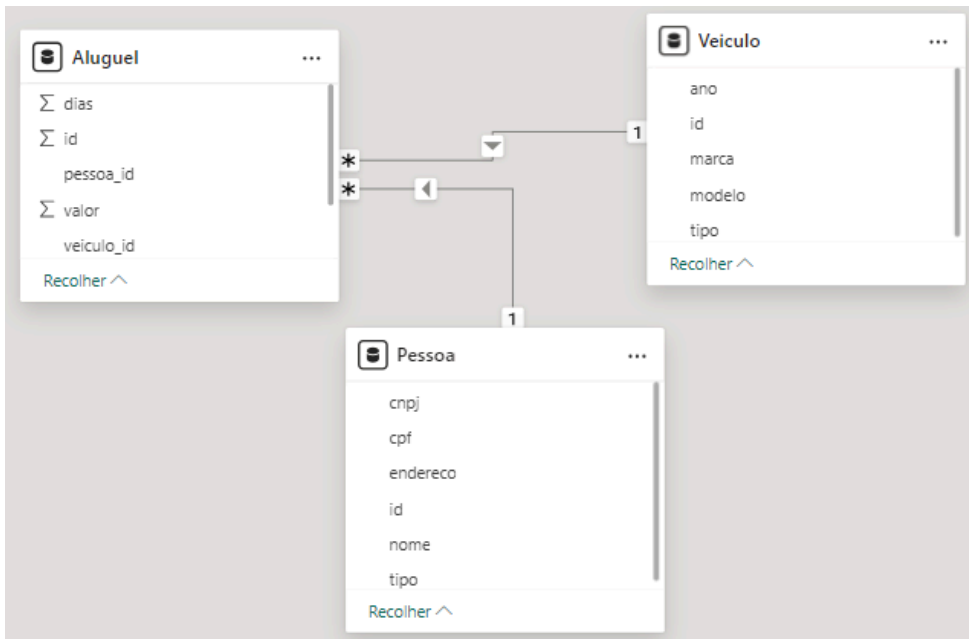
# ANEXOS

**Imagem 6 - Tabela de dados**

Aluguel						Pessoa						Veiculo					
id	pessoa_id	veiculo_id	dias	valor		id	nome	endereco	tipo	cpf	cnpj		id	modelo	marca	ano	tipo
1	1	1	5	RS 500		1	João Silva	Rua A	Fisica	12345678901		1	Civic	Honda	2020	CARRO	
2	2	2	3	RS 150		2	Maria Oliveira	Avenida B	Juridica		12345678000195	2	CB 500	Honda	2019	MOTO	
3	3	3	7	RS 1.400		3	Carlos Santos	Rua C	Fisica	23456789012		3	Scania	Road 500	2021	CAMINHÃO	
4	1	4	10	RS 1.000		4	Ana Souza	Avenida D	Juridica		23456789000185	4	Fusca	Volkswagen	1975	CARRO	
5	5	5	4	RS 400		5	Lúcia Almeida	Rua E	Fisica	34567890123		5	Toyota Corolla	Toyota	2022	CARRO	
6	6	6	2	RS 100		6	Roberto Costa	Avenida F	Juridica	34567890000176		6	Ninjab 650	Kawasaki	2021	MOTO	
7	7	7	6	RS 1.200		7	Sofia Pereira	Rua G	Fisica	45678901234		7	Volvo FH	Volvo	2020	CAMINHÃO	
8	8	8	1	RS 100		8	Pedro Mendes	Avenida H	Juridica		45678989000167	8	Onix	Chevrolet	2023	CARRO	
9	9	9	8	RS 400		9	Juliana Martins	Rua I	Fisica	56789012345		9	Yamaha MT-09	Yamaha	2021	MOTO	
10	10	10	5	RS 800		10	Ricardo Lima	Avenida J	Juridica		56789078000158	10	Caminhão MB 1933	Mercedes-Benz	2019	CAMINHÃO	
11	5	2	3	RS 150		11	Mariana Silva	Rua K	Fisica	67890123456		11	Ford Fiesta	Ford	2021	CARRO	
12	6	3	2	RS 400		12	André Almeida	Avenida L	Juridica		67890167000189	12	Kia Seltos	Kia	2022	CARRO	
13	1	5	1	RS 100		13	Fabiana Costa	Rua M	Fisica	78901234567		13	Hyundai HB20	Hyundai	2023	CARRO	
14	2	4	7	RS 700		14	Cleber Ferreira	Avenida N	Juridica		78901256000190	14	Nissan Kicks	Nissan	2022	CARRO	
15	3	1	10	RS 1.000		15	Elena Rocha	Rua O	Fisica	89012345678		15	Renault Kiwid	Renault	2021	CARRO	
16	11	11	5	RS 500		16	Gustavo Nascimento	Avenida P	Juridica		89012347000181	16	Chevrolet Tracker	Chevrolet	2022	CARRO	
17	12	12	3	RS 300		17	Tatiane Ribeiro	Rua Q	Fisica	90123456789		17	Fiat Argo	Fiat	2021	CARRO	
18	13	13	4	RS 400		18	Paulo Silva	Avenida R	Juridica		90123458000192	18	Volkswagen T-Cross	Volkswagen	2022	CARRO	
19	14	14	2	RS 200		19	Nadia Martins	Rua S	Fisica	01234567890		19	Toyota Yaris	Toyota	2023	CARRO	
20	15	15	6	RS 600		20	Fernando Gomes	Avenida T	Juridica		01234569000193	20	Peugeot 208	Peugeot	2021	CARRO	
21	16	16	1	RS 150													

Fonte: Autores.

**Imagem 7 - Relacionamento entre as tabelas**



Fonte: Autores.