### ISAQUE NEVES SANT’ANA

### ANÁLISE ORIENTADA A OBJETOS

#### Relatório de Aula Prática

RIO DAS OSTRAS/RJ

2025

### ISAQUE NEVES SANT’ANA

### ANÁLISE ORIENTADA A OBJETOS

#### Relatório de Aula Prática

Relatório técnico apresentado como requisito parcial para obtenção de aprovação na disciplina Análise Orientada a Objetos, no Curso de Análise Orientada a Objetos, na Universidade Norte do Paraná.

Profª Vanessa Matias Leite

RIO DAS OSTRAS/RJ

2025

# RESUMO

Este relatório descreve o processo de desenvolvimento de um diagrama de classes para um sistema de locação de veículos, desde a análise de requisitos até a elaboração final do diagrama. Foram identificadas cinco classes principais: Automóvel, Cliente, Locação, Modelo e Marca. Cada classe foi detalhada com atributos específicos e métodos necessários para operações relacionadas à locação de veículos. As relações entre as classes foram definidas para refletir a complexidade e a interdependência entre os elementos do sistema, como os relacionamentos muitos-para-um entre carros e modelos/marcas, e muitos-para-muitos entre carros e clientes. O resultado é um diagrama de classes que oferece uma visão clara e organizada da estrutura do sistema, proporcionando o entendimento necessário para a implementação eficiente e eficaz do sistema de locação de veículos. Este relatório não apenas demonstra a aplicação prática dos conceitos de modelagem orientada a objetos, mas também contribui significativamente para a formação acadêmica e profissional dos envolvidos no projeto.

**SUMÁRIO**

[1. INTRODUÇÃO 1](#_Toc193755828)

[2. MÉTODOS 2](#_Toc193755829)

[2.1 Análise dos Requisitos 2](#_Toc193755830)

[2.2 Identificação das Classes 2](#_Toc193755831)

[2.3 Definição das Relações: 2](#_Toc193755832)

[2.4 Detalhamento dos Atributos e Métodos 3](#_Toc193755833)

[2.5 Diagramação 4](#_Toc193755834)

[3. RESULTADOS 5](#_Toc193755835)

[3.1 Classes 5](#_Toc193755836)

[3.2 Relacionamentos 7](#_Toc193755837)

[3.3 Métodos 7](#_Toc193755838)

[3.4 Atributos 7](#_Toc193755839)

[4. CONCLUSÃO 8](#_Toc193755840)

[REFERÊNCIAS 9](#_Toc193755841)

# INTRODUÇÃO

No contexto atual de desenvolvimento de software, a modelagem de sistemas se apresenta como uma etapa importante para garantir a eficiência das aplicações. Este relatório foca na criação de um diagrama de classes para um sistema de locação de veículos, utilizando a metodologia orientada a objetos. Através de uma análise detalhada dos requisitos, identificação de classes, definição de relações, e detalhamento dos atributos e métodos, este trabalho busca fornecer uma base sólida para o desenvolvimento e a implementação do sistema. O objetivo é criar uma representação estruturada que facilite tanto o entendimento quanto a manutenção do sistema proposto, utilizando o *Visual* *Paradigm* como ferramenta de suporte para a diagramação.

# MÉTODOS

## Análise dos Requisitos

Foi realizado um detalhamento de requisitos fornecidos, identificando as entidades principais do sistemas e suas relações. Cada uma possui seus atributos e relacionamentos específicos, conforme descrito nos requisitos fornecidos.

* + 1. **Automóveis:** Representa os automóveis disponíveis para locação. Esta entidade possui atributos como número da placa, cor, ano, tipo de combustível, número de portas, quilometragem, RENAVAM, chassi, valor de locação, modelo e marca.
    2. **Cliente:** Representa as pessoas que alugam os carros. Esta entidade contém informações como nome, endereço, número de telefone, e-mail, entre outros dados pessoais importantes.
    3. **Locação:** Representa a relação entre os carros e os clientes durante o período de locação. Esta entidade armazena informações sobre a data e hora de início da locação, data e hora de devolução, preço da locação, entre outros detalhes relacionados ao aluguel de veículos.
    4. **Marca**: Representa os modelos de carros disponíveis para locação. Cada modelo pode estar relacionado a muitos carros.
    5. **Modelo**: Representa as marcas dos carros disponíveis para locação. Cada marca pode referir-se a muitos modelos, mas cada modelo só tem uma marca específica.

## Identificação das Classes

Com base nos requisitos, foram identificadas as classes principais do sistema, “Automóvel”, “Cliente”, “Locação”, “Modelo”, “Marca”.

## Definição das Relações:

Foram estabelecidas as relações entre as classes, conforme especificado nos requisitos. Isso incluiu associações como:

* Um carro pertence a um modelo e uma marca (relacionamento muitos-para-um).
* Um modelo pode estar associado a muitos carros (relacionamento um-para- muitos).
* Uma marca pode estar associada a muitos modelos (relacionamento um-para-

muitos).

* Um carro pode ser alugado por muitos clientes e um cliente pode alugar muitos carros (relacionamento muitos-para-muitos).

## Detalhamento dos Atributos e Métodos

Cada classe foi detalhada com seus atributos específicos, como número da placa, cor, modelo, marca, etc., e métodos necessários para operações relacionadas à locação de veículos.

#### Automóvel:

* **Atributos**: Número da placa, cor, modelo, marca, ano, tipo de combustível, número de portas, quilometragem, RENAVAM, chassi, valor de locação.

#### Métodos:

* **getters/setters**: Métodos para acessar e modificar os atributos do carro.
* **isAlugado()**: Método para verificar se o carro está atualmente alugado.
* **cadastrarAutomovel()**: Método para cadastrar automóvel.
* **consultarAutomovel()**: Método para consultar automóvel.

#### Cliente:

* **Atributos**: Nome, endereço, número de telefone, e-mail.

#### Métodos:

* **getters/setters**: Métodos para acessar e modificar os atributos do cliente.
* **alugarCarro()**: Método para permitir que o cliente alugue um carro, registrando a locação.
* **devolverCarro()**: Método para permitir que o cliente devolva um carro, registrando

a devolução.

* **listarCarrosAlugados()**: Método para listar os carros atualmente alugados pelo cliente.

#### Locação:

* **Atributos**: Data e hora de início da locação, data e hora de devolução, cliente, automóvel, preço da locação.

#### Métodos:

* **getters/setters**: Métodos para acessar e modificar os atributos da locação.
* **registrarLocacao()**: Método para registrar a locação do carro, armazenando a data e hora.
* **registrarDevolucao()**: Método para registrar a devolução do carro, armazenando a data e hora.

#### Marca:

* **Atributos**: Nome da marca

#### Métodos:

* **getters/setters**: Métodos para acessar e modificar os atributos da marca.

#### Modelo:

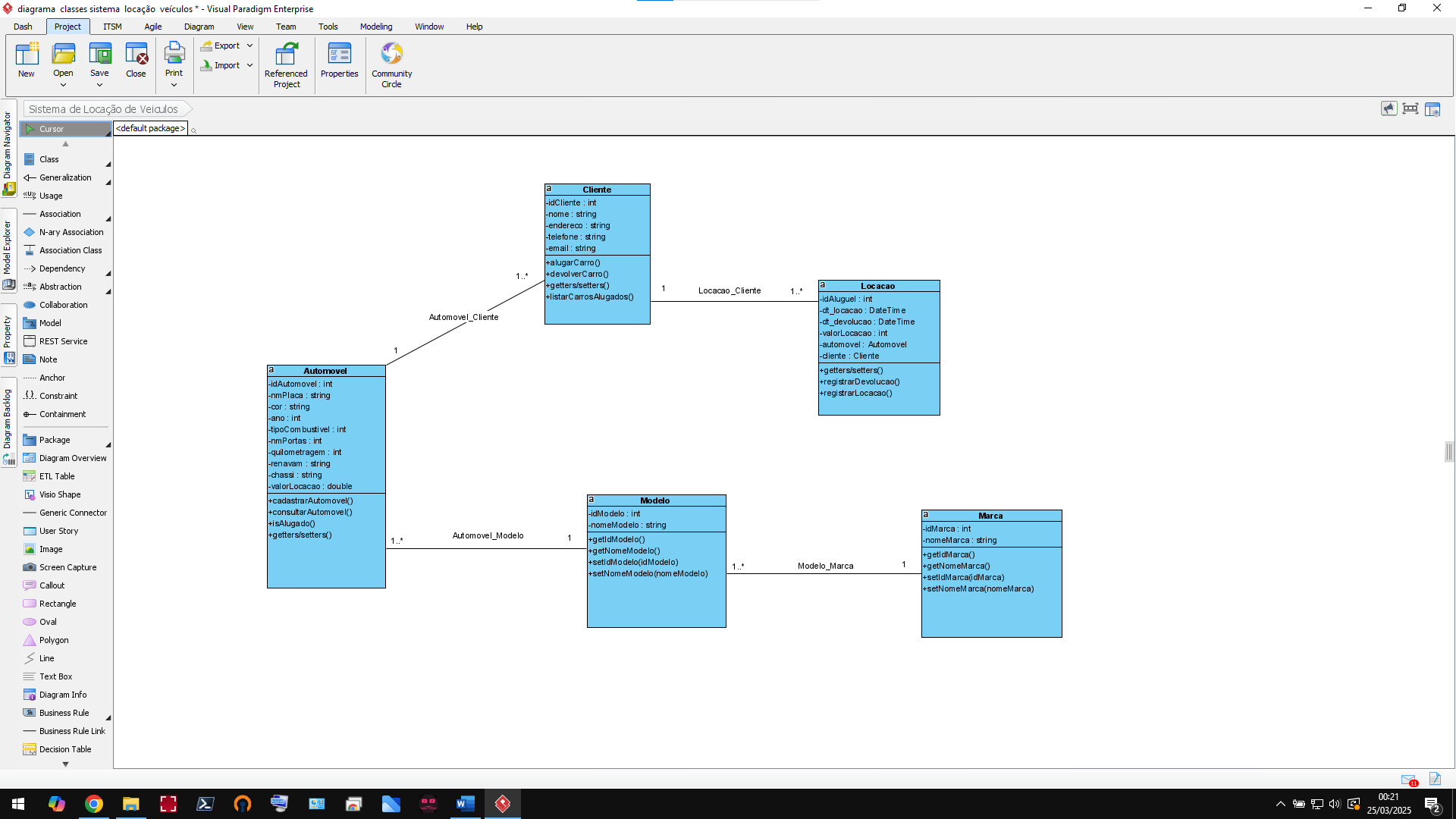
* **Atributos**: Nome do modelo

#### Métodos:

* **getters/setters**: Métodos para acessar e modificar os atributos do modelo.

## Diagramação

Utilizando uma ferramenta Visual Paradigm, o diagrama de classes foi elaborado, trazendo uma representação clara e organizada das classes e suas relações.

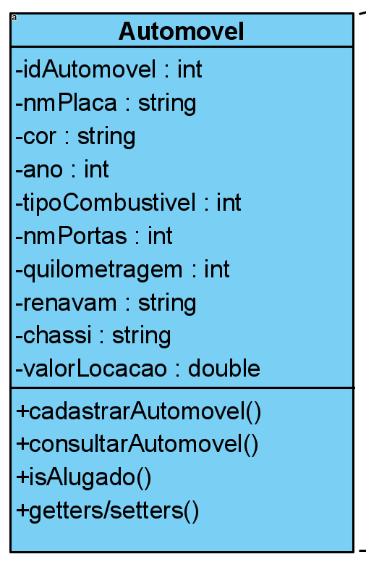


# RESULTADOS

O diagrama de classes proporciona uma visão geral da estrutura do sistema e das interações entre suas entidades, facilitando o entendimento e a implementação do software de locação de veículos. A seguir, são destacados os pontos-chave do diagrama:

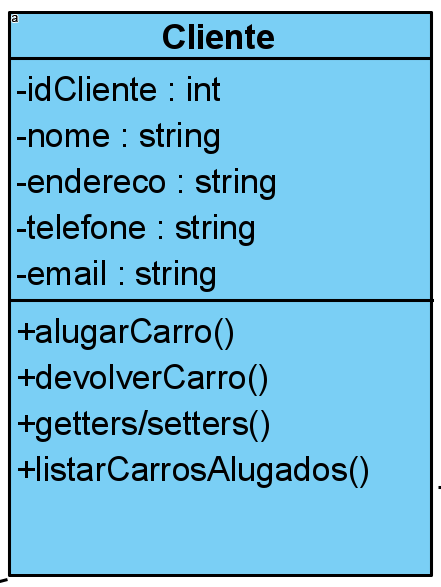
## Classes

#### Automóvel:

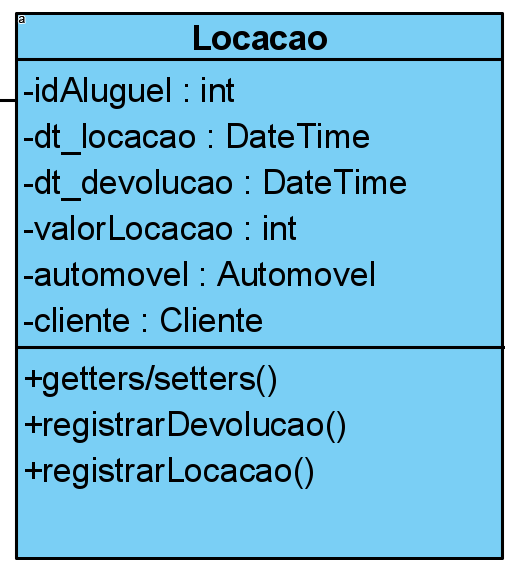


Representa os veículos disponíveis para locação, contendo atributos como número da placa, cor, ano, etc.

#### Cliente:



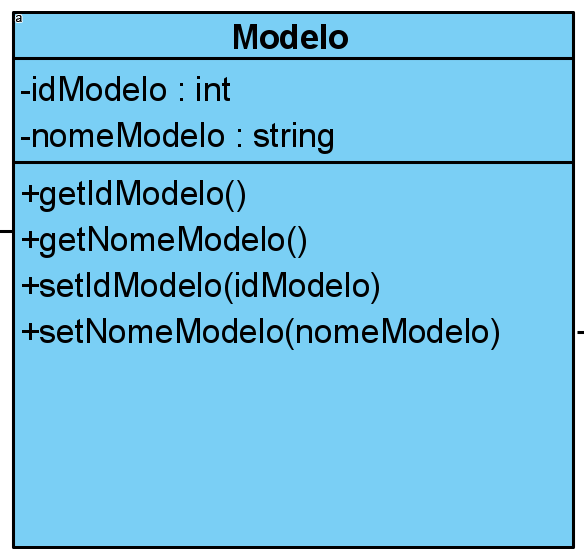
#### Locação:



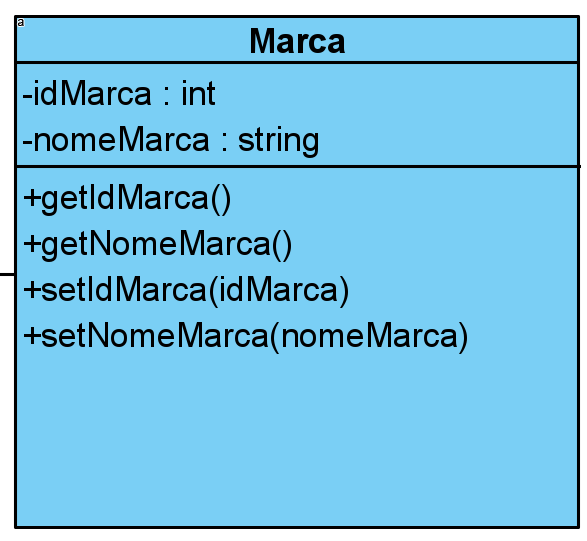
Representa os indivíduos ou empresas que alugam os carros, com informações como nome, endereço, etc.

Representa a relação entre carros e clientes durante o período de aluguel, armazenando informações sobre datas e preços.

#### Modelo:



#### Marca:



Representa os modelos de carros disponíveis para locação.

Representa as marcas dos carros disponíveis para locação.

## Relacionamentos

* + 1. Um carro pertence a um modelo e uma marca (relacionamento muitos-para- um).
    2. Um modelo pode estar associado a muitos carros (relacionamento um-para- muitos).
    3. Uma marca pode estar associada a muitos modelos (relacionamento um-para- muitos).
    4. Um carro pode ser alugado por muitos clientes e um cliente pode alugar muitos carros (relacionamento muitos-para-muitos).

## Métodos

* Cada classe foi detalhada com métodos específicos para operações relacionadas à locação de veículos, como registro de locação, registro de devolução, aluguel de carro, devolução de carro, entre outros.

## Atributos

* Foram definidos atributos relevantes para cada classe, garantindo que todas as informações necessárias para o gerenciamento do sistema estejam presentes.

# CONCLUSÃO

O desenvolvimento do diagrama de classes para o sistema de locação de veículos foi concluído com sucesso, proporcionando uma base sólida para a implementação do sistema. A representação estruturada das classes e suas relações facilitará o desenvolvimento e manutenção do sistema, garantindo sua eficiência. Este relatório demonstra o entendimento dos conceitos de modelagem de sistemas orientados a objetos e sua aplicação na prática, contribuindo para a formação acadêmica e profissional dos envolvidos.

# REFERÊNCIAS

**Visual Paradigm.** (s.d.). *Visual Paradigm*. Acessado em 24 de março de 2025, de <https://www.visual-paradigm.com>.

**Pereira, L. A. de M.** (2011). *Análise e Modelagem de Sistemas com a UML: Com Dicas e Exercícios Resolvidos*. 1ª edição. Rio de Janeiro: Edição do Autor.

**E.V. Online Learning.** (s.d.). *UML v2*. Recuperado de https://lms.ev.org.br/mpls/Custom/Cds/COURSES/3242-UMLv2/curso/modulo-1.html.