

***Mini-Imports***

Trabalho final do Projeto Integrador de Desenvolvimento Desktop

São José dos Campos

2023

Aluno:

Vinícius Luiz de Almeida Soares

**Sumário**

[Proposta 4](#_Toc126306607)

[Diagrama UML 5](#_Toc126306608)

[Diagrama Caso de Uso 6](#_Toc126306609)

[Colocar um texto 6](#_Toc126306610)

[Colocar uma imagem 6](#_Toc126306611)

[Diagrama de Classe 7](#_Toc126306612)

[Diagramas de Atividades 8](#_Toc126306613)

[Modelagem de Dados 9](#_Toc126306614)

[Exemplos de Códigos de Programação e Banco de Dados SQL-Server 10](#_Toc126306615)

# **Proposta**

Um sistema para atender aqueles que importam produtos e/ou compram produtos nacionais e assim facilitar no acesso e na administração dos mesmos.



**Diagrama UML**

UML (Unified Modeling Language) é uma linguagem de modelagem visual amplamente utilizada para representar sistemas de software, incluindo bancos de dados. É utilizada para descrever, modelar e documentar a estrutura de um sistema, ajudando a equipe de desenvolvimento a entender as necessidades do projeto, identificar relacionamentos entre entidades, e planejar a implementação de software.

Os diagramas UML incluem diferentes tipos de representações gráficas, como:

Diagrama de Classe: representa as classes e objetos que compõem um sistema, bem como suas relações e atributos.

* Diagrama de Caso de Uso: representa as interações entre os usuários e o sistema.
* Diagrama de Atividade: representa o fluxo de trabalho de uma aplicação.
* Diagrama de Sequência: representa a interação entre objetos em um sistema, mostrando a ordem em que as mensagens são trocadas.

Os diagramas UML são amplamente utilizados em todas as fases do desenvolvimento de software, desde a concepção até a manutenção, e são considerados uma ferramenta valiosa para ajudar a garantir a qualidade e a integridade do sistema.

No caso de uma transportadora internacional vamos fazer um diagrama uml dessa forma:

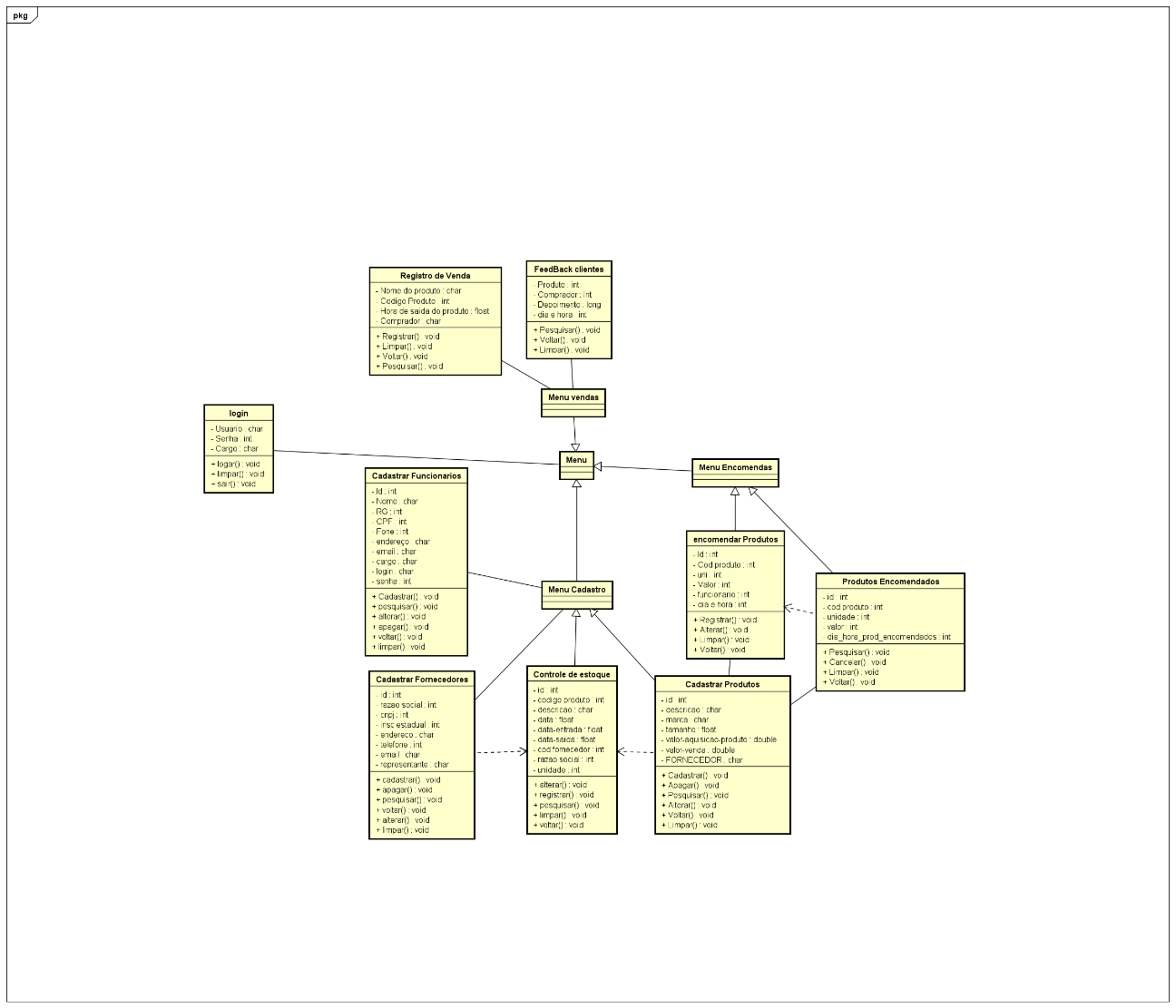
* Usar classes para representar entidades importantes do sistema, como "Empresa", "Veículo", "Rota", "Carga" e "Entrega".
* Usar relacionamentos entre as classes para representar as ligações entre as entidades. Por exemplo, a classe "Veículo" pode ter uma associação de um para muitos com a classe "Entrega".
* Usar herança para representar relações de tipo-subtipo. Por exemplo, a classe “Avião” e “Navio” pode herdar da classe "Veículo".
* Adicionar atributos às classes para representar informações relevantes, como "Nome da Empresa", "Tipo de Veículo", "Data de Entrega" e "Destino".

# **Diagrama Caso de Uso**

Diagrama

Descrição gerada automaticamente

# **Diagrama de Classe**

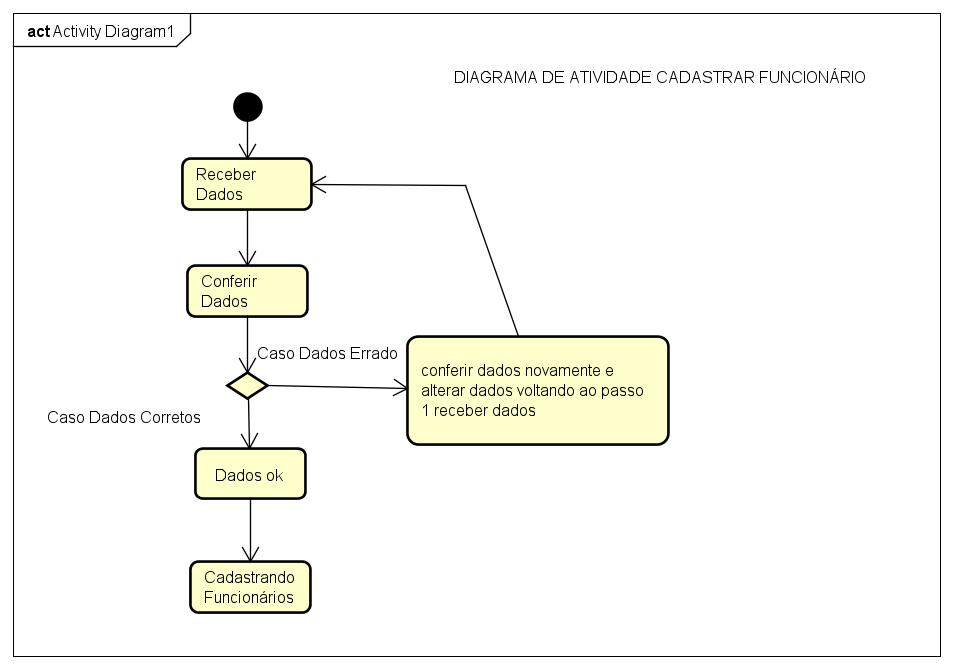


# Diagramas de Atividades

Diagrama

Descrição gerada automaticamente

Diagrama

Descrição gerada automaticamente

# Modelagem de Dados

Diagrama

Descrição gerada automaticamente

# Exemplos de Códigos de Programação e Banco de Dados SQL-Server

create database mini\_imports

go

use mini\_imports

go

create schema imports authorization dbo

go

create table imports.cadastrarFuncionarios (

cadFun\_id int primary key identity(1,1), --id

cadFun\_nm varchar(40) not null, --nome

cadFun\_rg int not null, --RG

cadFun\_fone int not null, --telefone

cadFun\_ender varchar(50) not null, --endereco

cadFun\_email varchar(50) not null, --email

cadFun\_cargo varchar(50) not null, --cargo

cadFun\_login varchar(50) unique, --login

cadFun\_senha varchar(50) --senha

)

go

TELAS

→

