



SENAI

REQUISITOS E MODELAGEM DE SOFTWARE



01 DIAGRAMA DE CLASSE

02 CONCEITOS FUNDAMENTAIS

03 TIPOS DE RELAÇÕES

04 ELEMENTOS

05 EXEMPLO

REQUISITOS DE SOFTWARE

REQUISITOS



1 – DIAGRAMAS DE CLASSE

- **Objetivo :** Representar as classes do sistema e as relações entre elas, fornecendo uma visão estruturada e abstrata do código.



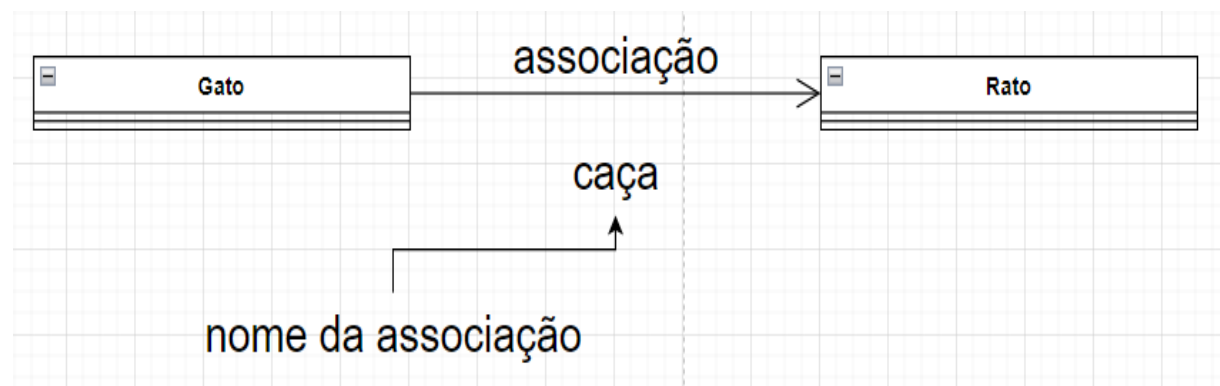
2 – CONCEITOS FUNDAMENTAIS

- **Classe:** Um modelo abstrato que define as características (atributos) e comportamentos (métodos) de um conjunto de objetos.
- **Objeto:** Uma instância de uma classe, representando uma entidade específica no sistema.
- **Atributo:** Uma propriedade de uma classe, que armazena dados sobre um objeto.
- **Método:** Uma operação que pode ser realizada por um objeto de uma classe, definindo seu comportamento.
- **Relação:** A ligação entre duas ou mais classes, mostrando como elas interagem.



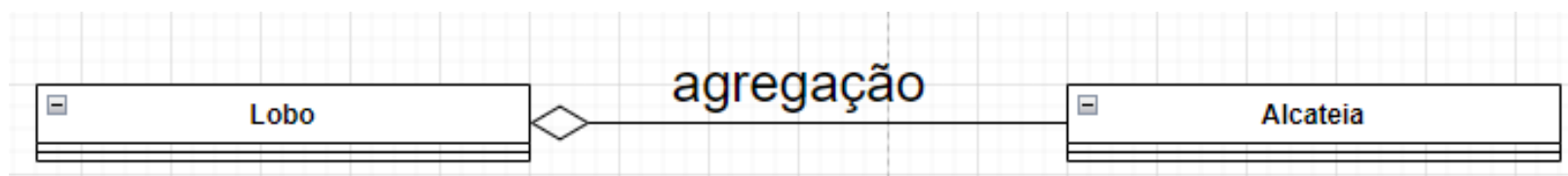
3 – TIPOS DE RELAÇÕES

- **Associação:**
 - Uma relação geral entre classes, indicando que objetos de uma classe podem interagir com objetos de outra, mas sem dependência entre elas.
 - A classe gato e a classe rato, gato caça rato, então existe uma relação entre elas, mas a classe gato não depende da classe rato e nem a classe rato depende da classe gato.



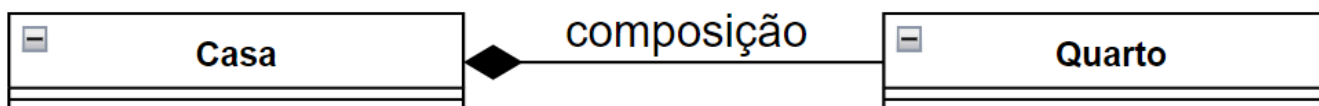
3 – TIPOS DE RELAÇÕES

- **Agregação:**
 - Uma relação "tem um", onde um objeto é composto por outros objetos.
 - A classe alcateia tem lobos, um lobo pode fazer parte da alcateia, mas ele não depende dessa alcateia para existir.



3 – TIPOS DE RELAÇÕES

- **Composição:**
 - Uma relação "tem um", mas mais forte que a agregação, onde a vida útil dos objetos compostos depende da classe composta.
 - Um quarto só existe dentro de uma casa. Se a casa for removida, o quarto também será removido.



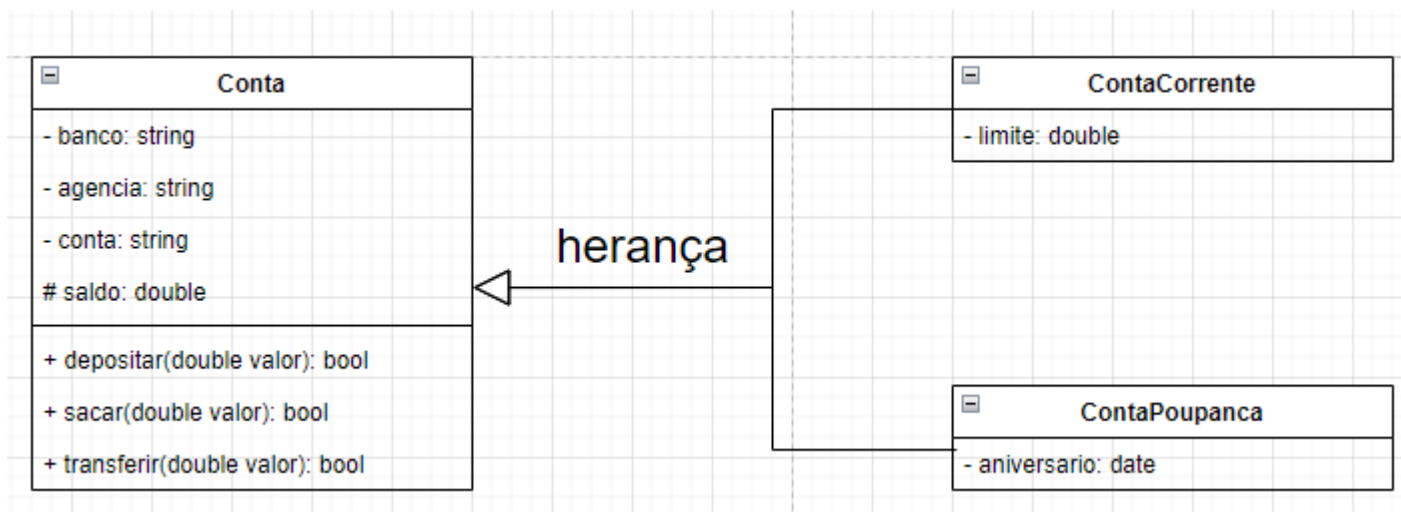
REQUISITOS DE SOFTWARE

REQUISITOS



3 – TIPOS DE RELAÇÕES

- **Herança:**
 - Uma relação "é um", onde uma classe herda características e comportamentos de outra classe.



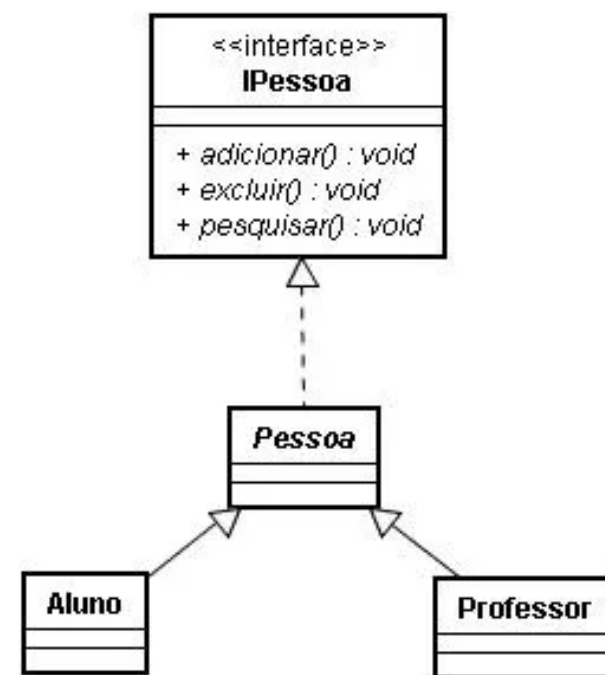
REQUISITOS DE SOFTWARE

REQUISITOS



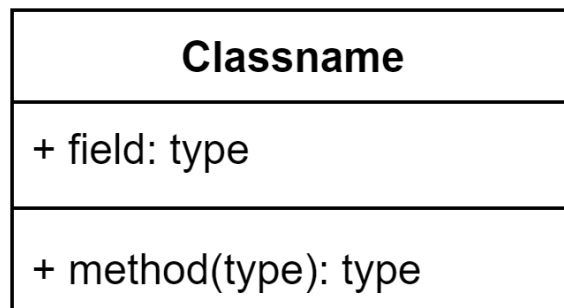
3 – TIPOS DE RELAÇÕES

- **Interface:**
 - Um contrato que define métodos, mas não implementações. As classes podem implementar interfaces, garantindo conformidade.
 - Classes “Aluno” e “Professor” herdam da classe “Pessoa” que por sua vez implementa a interface “IPessoa”.



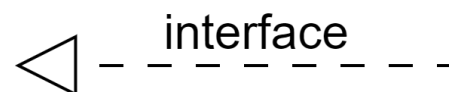
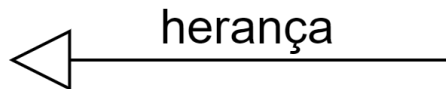
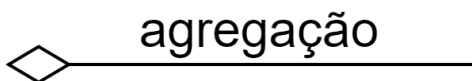
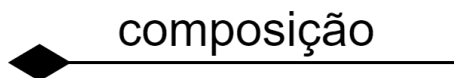
4 – ELEMENTOS DO DIAGRAMA DE CLASSE

- **Caixa de Classe:**
 - Representa uma classe, com seu nome, atributos e métodos.



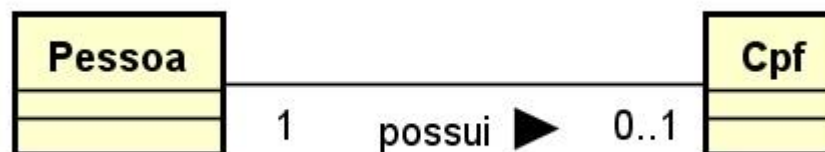
4 – ELEMENTOS DO DIAGRAMA DE CLASSE

- **Símbolo de relação:**
 - Linhas que conectam as caixas de classe, indicando o tipo de relacionamento.



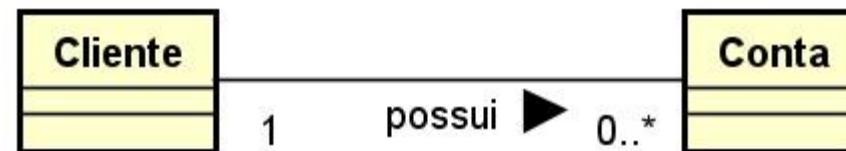
4 – ELEMENTOS DO DIAGRAMA DE CLASSE

- **Multiplicidade:**
 - Indica quantos objetos de uma classe podem se relacionar com um objeto de outra classe.
- **Um para um:**
 - Uma pessoa pode possuir 0 ou no máximo 1 Cpf e um Cpf pode ser somente de 1 pessoa.



4 – ELEMENTOS DO DIAGRAMA DE CLASSE

- Um para muitos:
 - Um cliente pode possuir 0 ou muitas contas e a Conta pode pertencer a somente 1 pessoa.



4 – ELEMENTOS DO DIAGRAMA DE CLASSE

- **Muitos para muitos:**
 - Um profissional pode trabalhar em zero ou muitos projetos e um projeto pode ter zero ou muitos profissionais trabalhando.



4 – ELEMENTOS DO DIAGRAMA DE CLASSE

- **Visibilidade:**
 - Indica o nível de acesso aos atributos e métodos:
 - **Público (+)**
 - Qualquer classe pode usar o método ou atributo.
 - **Privado (-)**
 - Apenas a própria classe pode usar o método ou atributo.
 - **Protegido (#).**
 - Apenas a subclasse(herança) ou superclasse podem usar o método ou atributo.



4 – ELEMENTOS DO DIAGRAMA DE CLASSE

- **Visibilidade:**
 - Indica o nível de acesso aos atributos e métodos:
 - **Público (+)**
 - Qualquer classe pode usar o método ou atributo.
 - **Privado (-)**
 - Apenas a própria classe pode usar o método ou atributo.
 - **Protegido (#).**
 - Apenas a subclasse(herança) ou superclasse podem usar o método ou atributo.



A nighttime photograph of a city street, likely in São Paulo, featuring tall buildings and light trails from moving vehicles. The Senai logo is overlaid on the left side.

SENAI

DEPARTAMENTO REGIONAL
DE SÃO PAULO

www.sp.senai.br