

Aonde você quer chegar?  
**Vai com a**



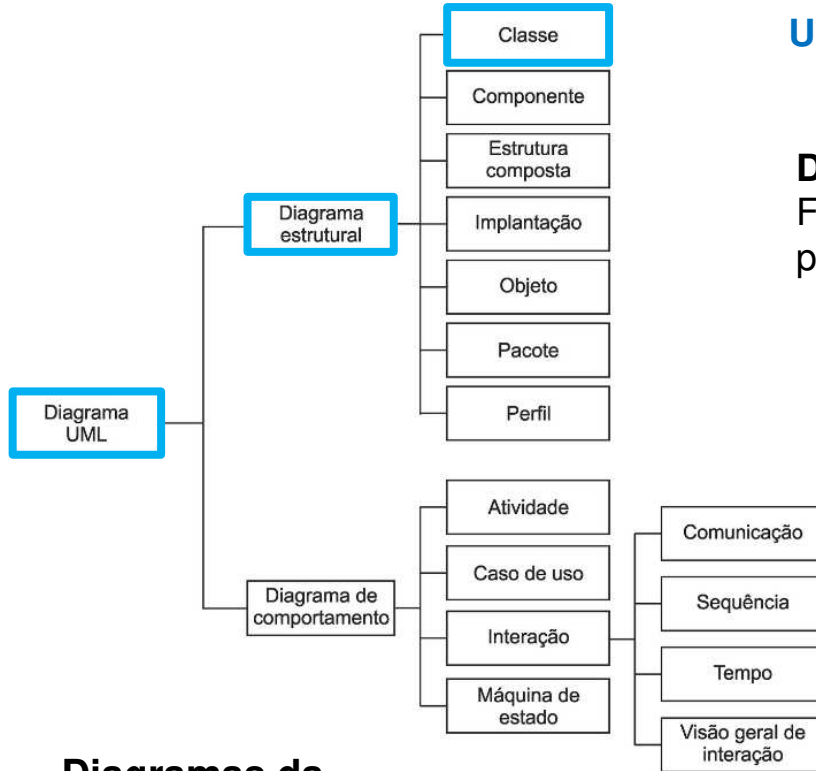


Disciplina: Análise e Projeto OO  
Prof. Maurício P. de Freitas MSc.

**Aula 04 – 14/03/2024**  
**Diagrama de Classe**







## UML – Unified Modeling Language

### Diagrama de Classe:

Faz parte do conjunto de diagramas estruturais propostos pela UML (Unified Modeling Language).

### Tipos de diagramas:

**Diagramas Estruturais:** descrevem os elementos estruturais que compõe o sistema;

**Diagramas de comportamento:** descrevem o comportamento dos elementos e suas interações.

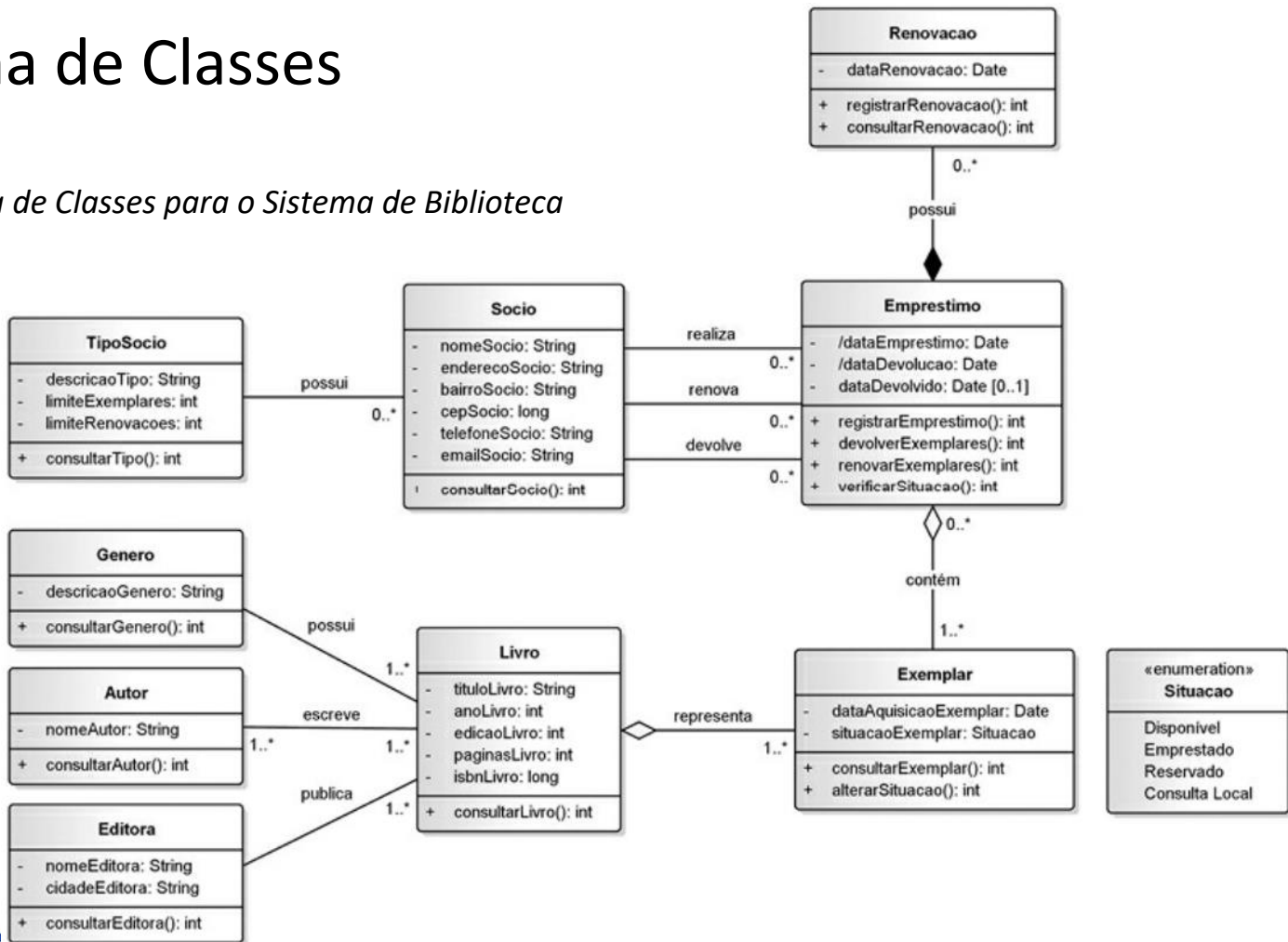
Diagramas da  
UML

# Diagrama de Classes

- Seu principal enfoque está em permitir a visualização das **classes que comporão o sistema** com seus respectivos atributos e métodos, bem como em demonstrar **como as classes do diagrama se relacionam**, complementam e transmitem informações entre si.

# Diagrama de Classes

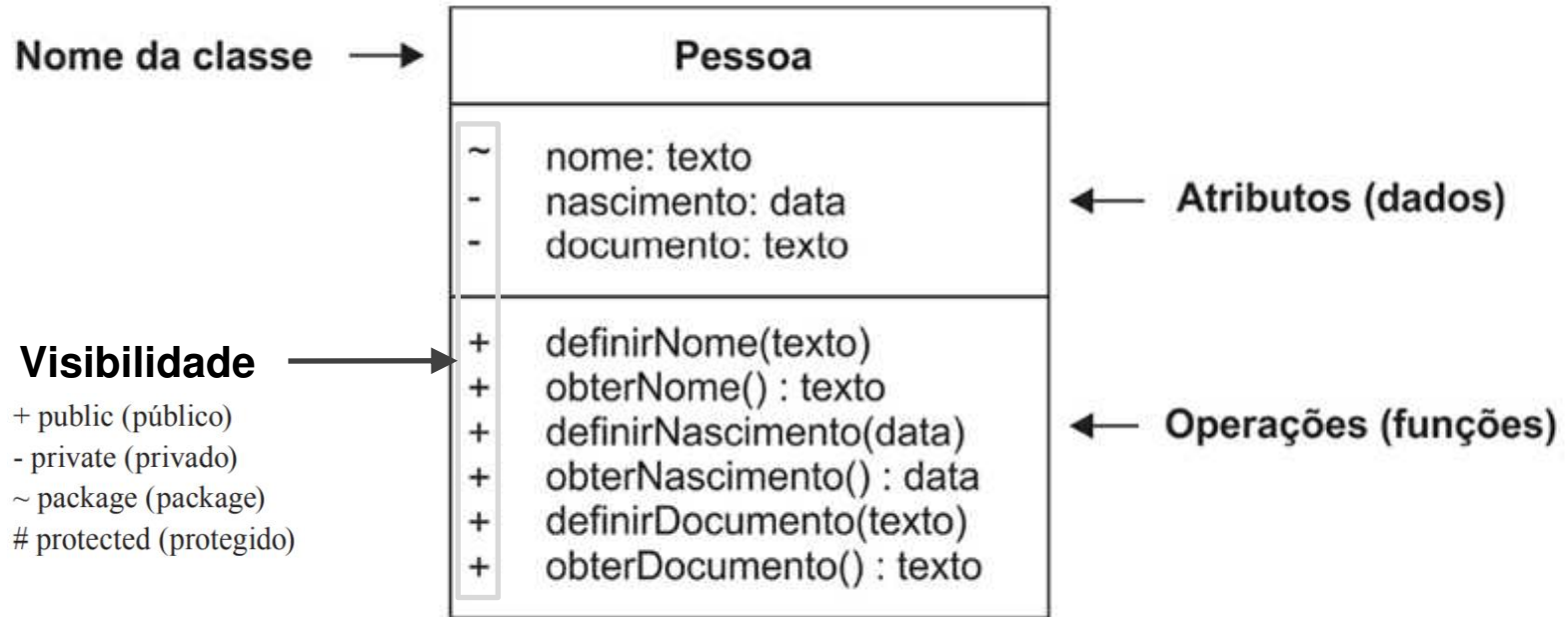
*Diagrama de Classes para o Sistema de Biblioteca*



# Classes

- ❑ São uma descrição de um grupo de objetos com papéis semelhantes no sistema.
  - ❑ A partir das classes, podemos criar os objetos (instâncias).
- ❑ Os objetos derivam de:
  - ❑ Coisas: objetos tangíveis, do mundo real, ...
  - ❑ Papéis/funções: classes de atores em sistemas → estudantes, gerentes, enfermeiros, ...
  - ❑ Eventos: admissão, inscrição, matrícula, ...
  - ❑ Interações: reuniões, tutoriais, ...

# Classes



# Classes

**Visibilidade:** indicar o nível de acessibilidade de um determinado atributo ou método.

Pessoa	
~	nome: texto
-	nascimento: data
-	documento: texto
+	definirNome(texto)
+	obterNome() : texto
+	definirNascimento(data)
+	obterNascimento() : data
+	definirDocumento(texto)
+	obterDocumento() : texto

**Privada (-):** significa que somente os objetos da classe detentora do atributo ou método poderão enxergá-lo.

**Protegida (#):** determina que, além dos objetos da classe detentora do atributo ou método, também os objetos de suas subclasses poderão ter acesso a este.

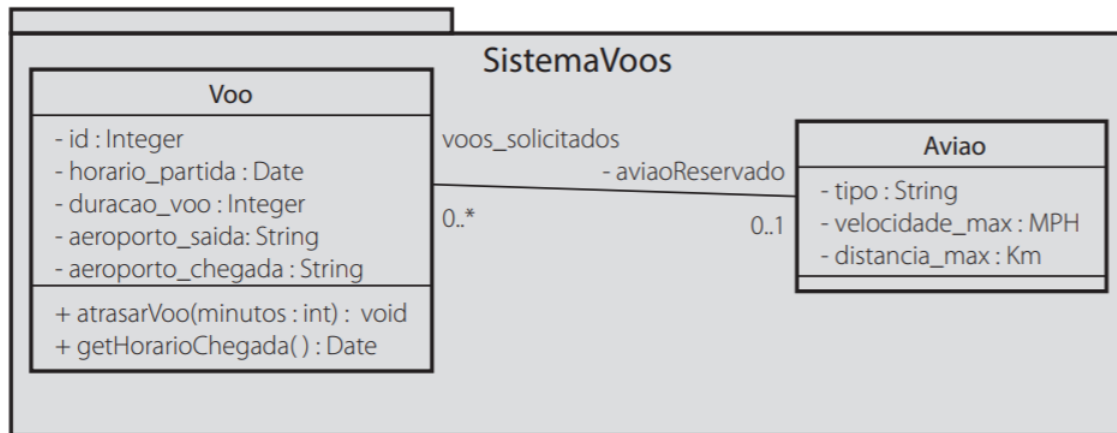
**Pública (+):** determina que o atributo ou método pode ser utilizado por qualquer objeto.

**Pacote (~):** determina que o atributo ou método é visível por qualquer objeto dentro do pacote.



# Pacotes

- Agrupar elementos que são semanticamente relacionados e podem ser alterados de forma unificada (módulos de um sistema).



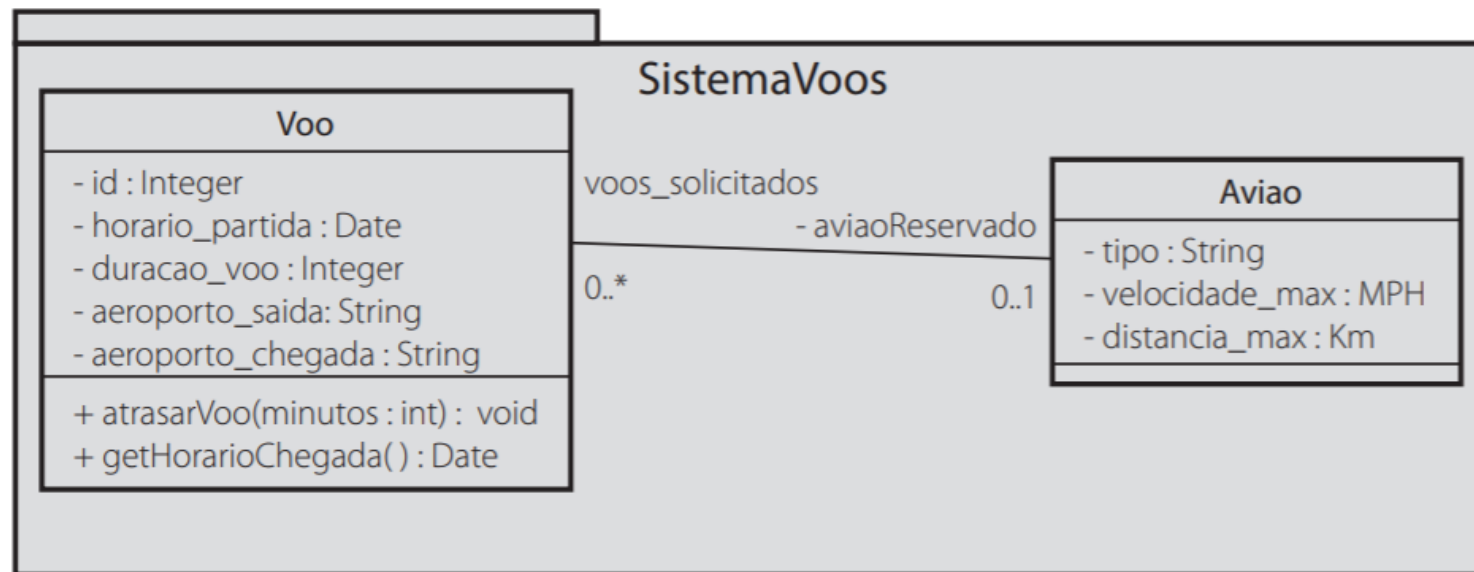
# Relacionamentos ou Associações

- ❑ **Os relacionamentos entre as classes, chamados associações,** permitem que as classes compartilhem informações entre si e colaborem para a execução dos processos executados pelo sistema.
- ❑ Determinam que as instâncias de uma classe estão, de alguma forma, ligadas às instâncias da outra classe.
- ❑ Multiplicidades/Cardinalidades:
  - ❑ 0..1 – 1..1 – 0..\* – 1..\* – 3..5 (valores específicos).

# Multiplicidades

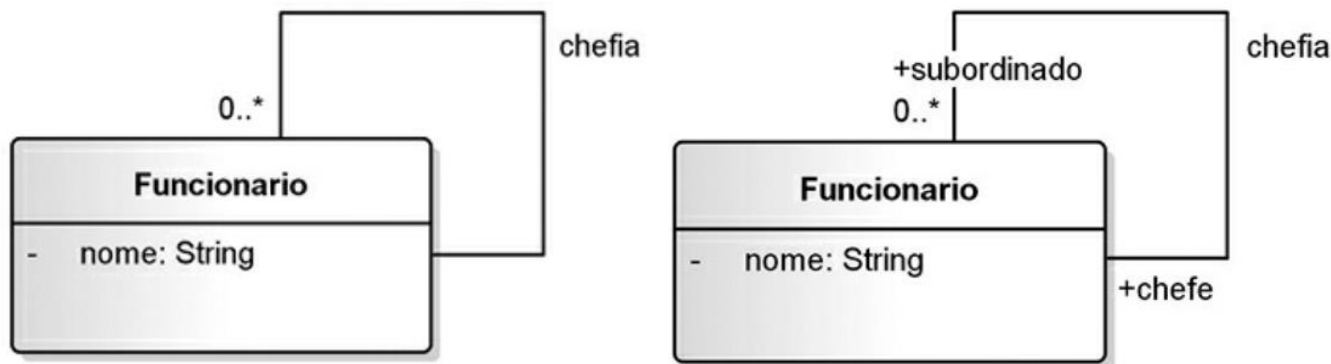
Multiplicidade	Significado
0..1	No mínimo, zero (nenhum) e, no máximo, um. Indica que os objetos das classes associadas não precisam obrigatoriamente estar relacionados, mas se houver relacionamento, indicará que apenas uma instância da classe relaciona-se com as instâncias da outra classe (ou da outra extremidade da associação, se esta for unária).
1..1	Um e somente um. Indica que apenas um objeto da classe relaciona-se com os objetos da outra classe.
0..*	No mínimo, nenhum e, no máximo, muitos. Indica que pode ou não haver instâncias da classe participando do relacionamento.
*	Muitos. Indica que muitos objetos da classe estão envolvidos na associação.
1..*	No mínimo, um e, no máximo, muitos. Indica que há pelo menos um objeto envolvido no relacionamento, podendo haver muitos objetos envolvidos.
3..5	No mínimo, três e, no máximo, cinco. Estabelece que existem pelo menos três instâncias envolvidas no relacionamento, mas podem ser quatro ou cinco as instâncias envolvidas, mas não mais do que isso.

# Multiplicidades



# Associação Unária (ou Reflexiva)

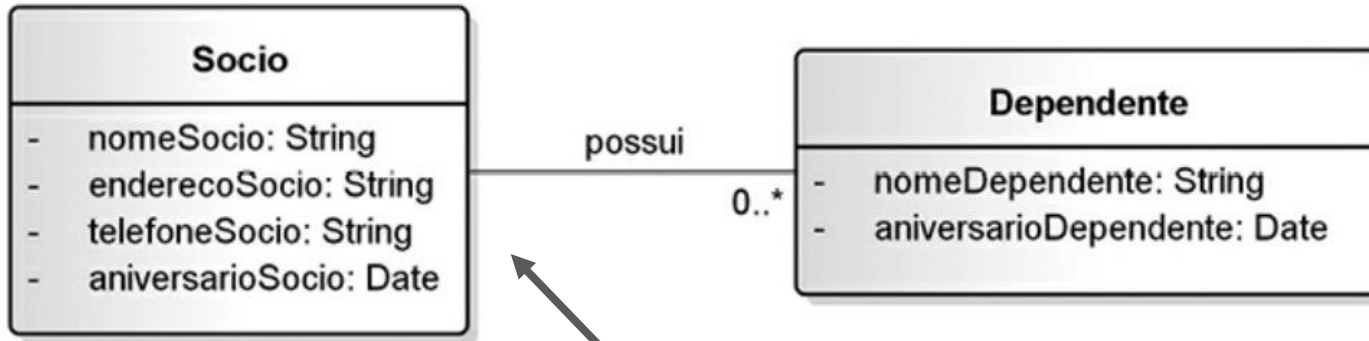
- ❑ Ocorre quando existe um relacionamento de um objeto de uma classe com objetos da mesma classe.





# Associação Binária

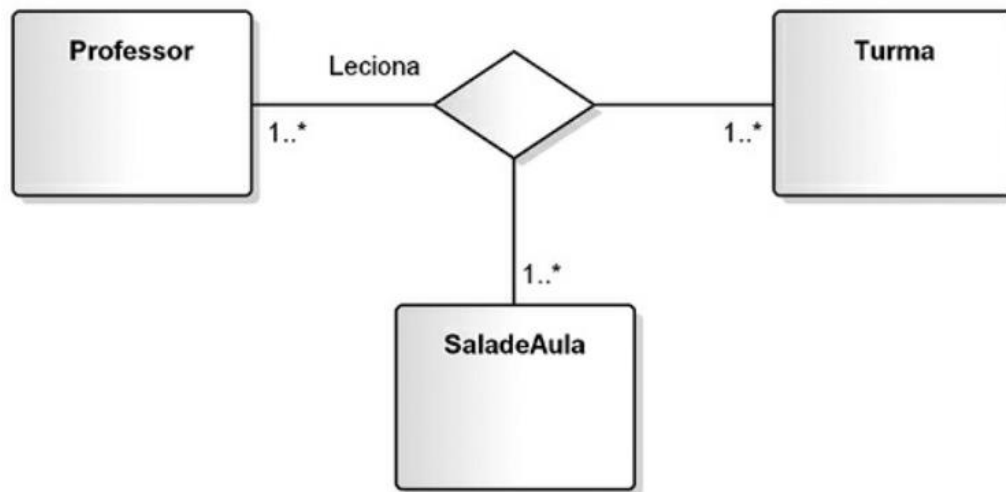
- ❑ Ocorrem quando são identificados relacionamentos entre objetos de duas classes distintas.



A omissão da  
cardinalidade indica-a  
como 1..1

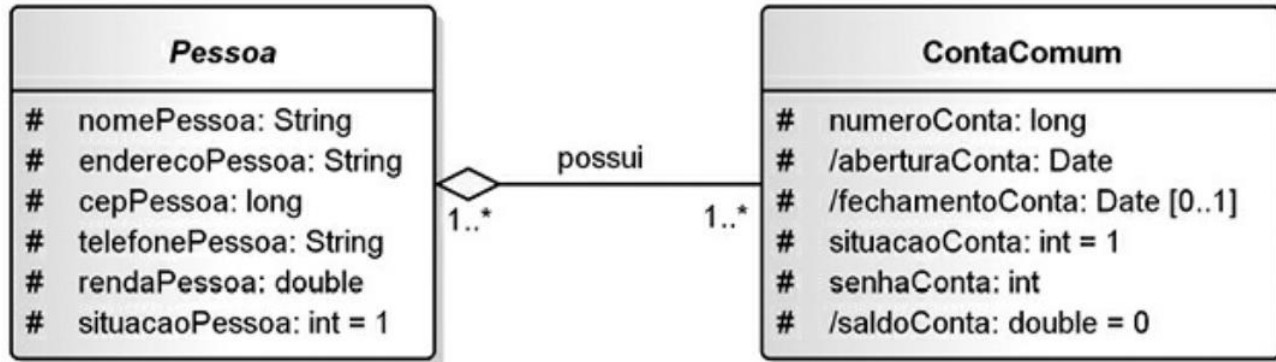
# Associação Ternária

- ❑ Associações ternárias ou n-árias conectam objetos de mais de duas classes. são úteis para demonstrar associações complexas, entretanto, é convém evitar a sua utilização.



# Agregação

- Objetivo demonstrar que as informações de um objeto (objeto-todo) são complementadas pelas informações contidas em um ou mais objetos no outro fim da associação (chamados objetos-parte).

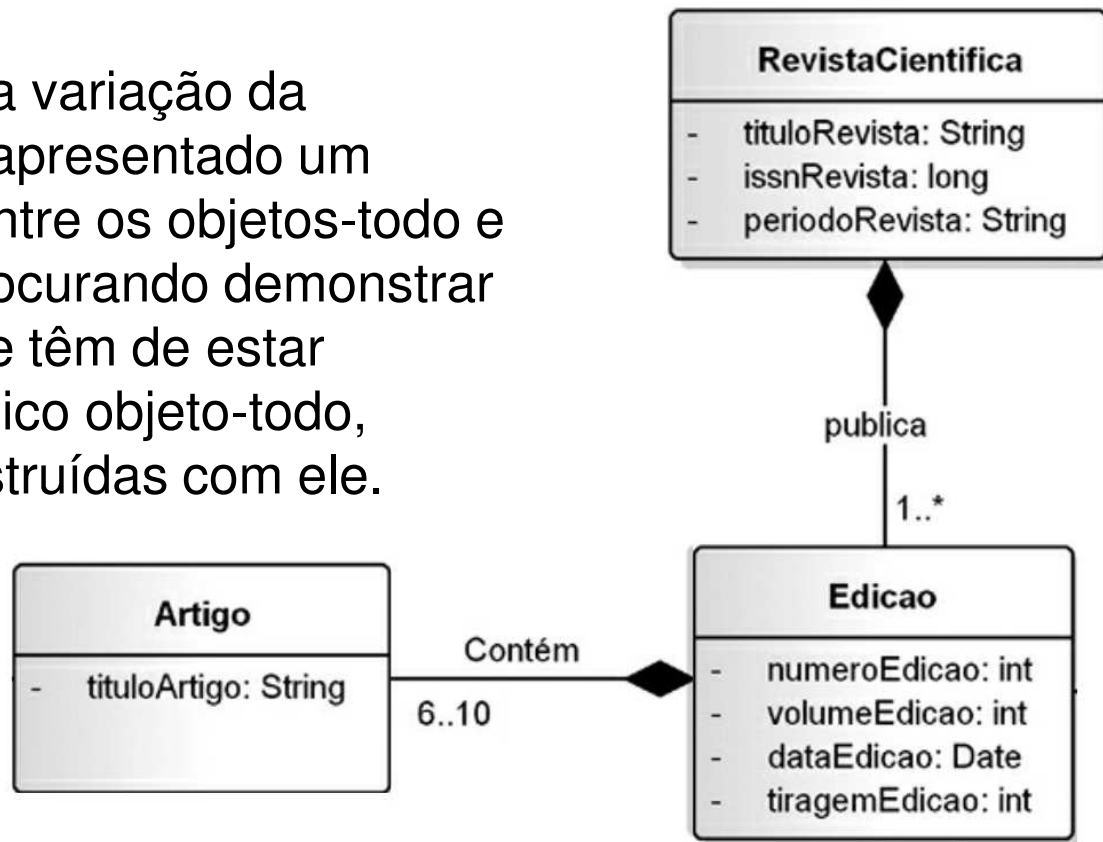


Objeto-todo

Objeto-parte

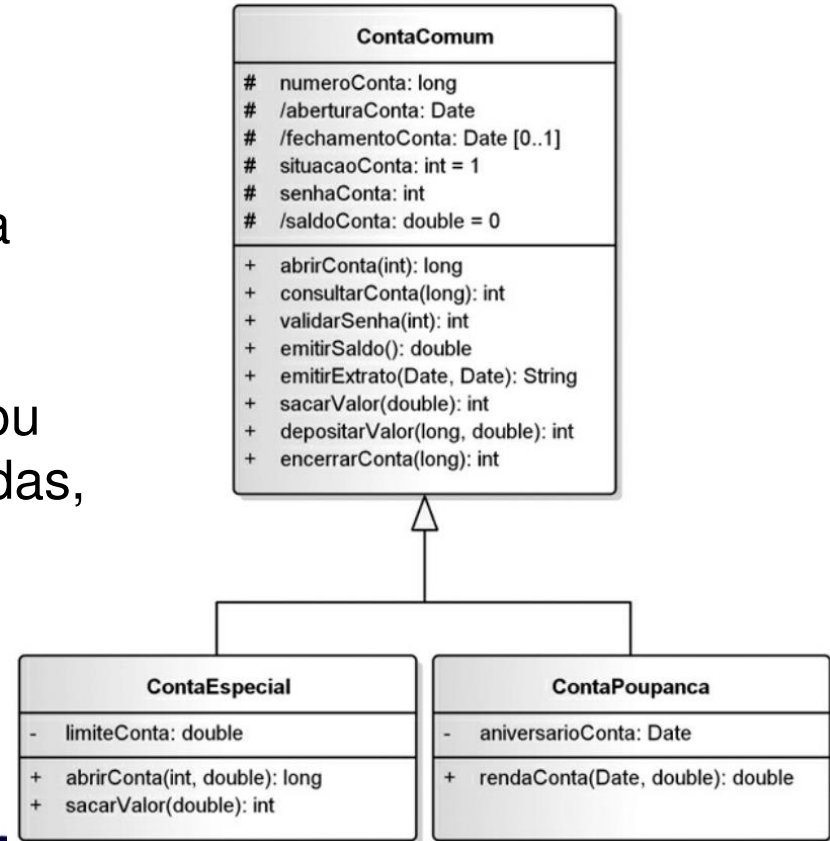
# Composição

- ❑ Constitui-se em uma variação da agregação, onde é apresentado um vínculo mais forte entre os objetos-todo e os objetos-parte, procurando demonstrar que os objetos-parte têm de estar associados a um único objeto-todo, sendo criadas e destruídas com ele.



# Generalização/Especialização

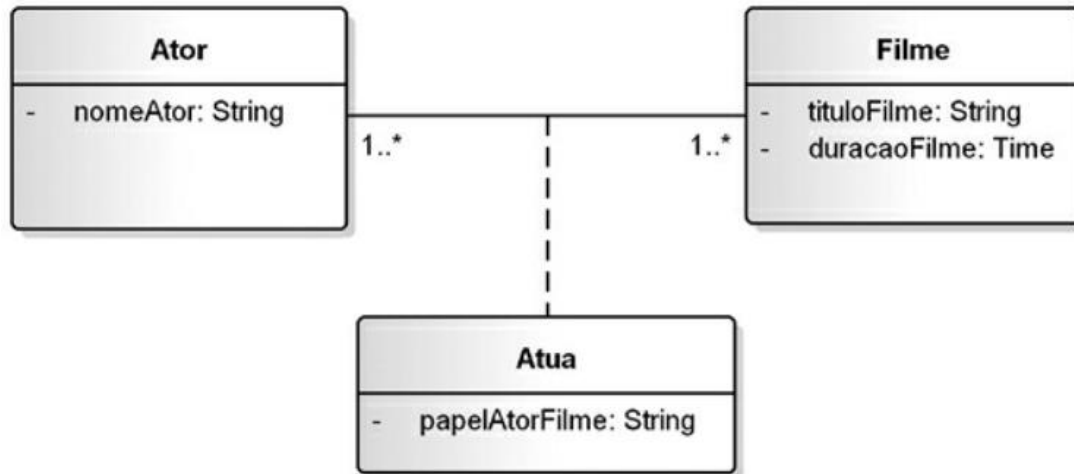
- ❑ O objetivo dessa associação é representar a ocorrência de herança entre as classes, identificando as classes-mãe (ou superclasses), chamadas gerais, e classes-filhas (ou subclasses), chamadas especializadas, demonstrando a hierarquia entre as classes e, possivelmente, métodos polimórficos nas classes especializadas.





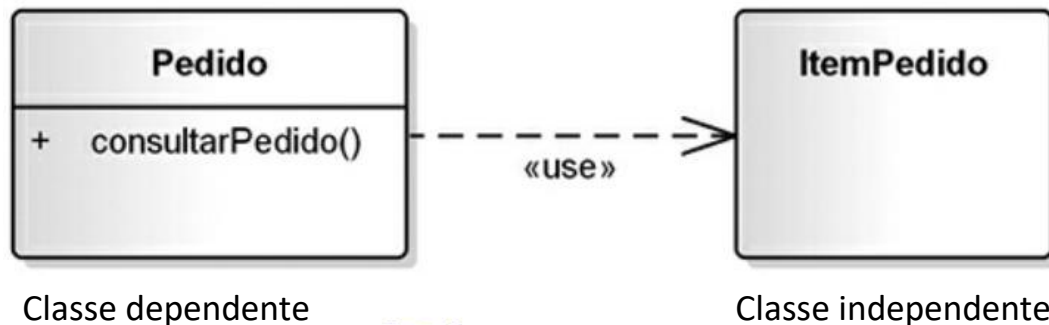
# Classe Associativa

- ❑ Classes associativas são necessárias nos casos em que existem atributos relacionados à associação que não podem ser armazenados por nenhuma das classes envolvidas.



# Dependência

- Uma relação de dependência é um tipo de associação que indica que uma mudança em uma classe (chamada de classe dependente) pode afetar outra classe (chamada de classe independente). A dependência é geralmente indicada por uma seta pontilhada com a etiqueta «use».

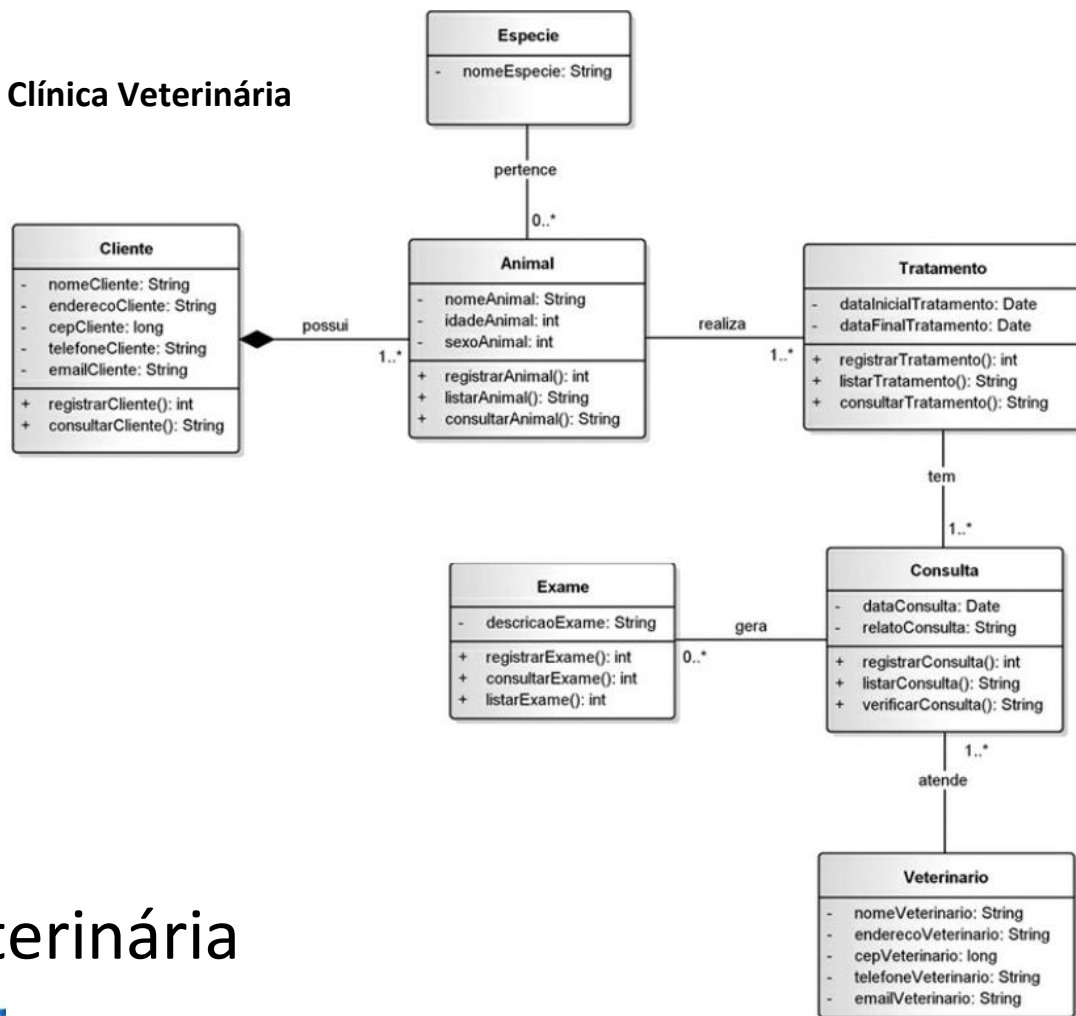


## Exemplo Clínica Veterinária: Identificar os possíveis atores.

- Os clientes primeiramente marcam consultas com a secretária, fornecendo informações pessoais e as dos animais que desejam tratar. Se o cliente ou o animal ainda não estiver cadastrado no sistema ou existir algum dado que precise ser atualizado, a secretária deverá atualizar o cadastro.
- Em cada sessão de tratamento (uma sessão equivale a uma consulta), o cliente deve informar os sintomas aparentes do animal, os quais devem ser registrados. O tratamento pode ser encerrado em apenas uma consulta, quando se tratar de algo simples, ou arrastar-se por muitas sessões, dependendo do diagnóstico do médico-veterinário.
- Durante a consulta, o veterinário pode marcar exames para o animal, a serem trazidos na sessão seguinte. O pedido dos exames e seus resultados devem ser registrados no histórico de tratamento do animal. Após cada sessão, o histórico da consulta deve ser atualizado.
- É responsabilidade da secretária manter atualizados os cadastros de clientes, animais, médicos e espécies.



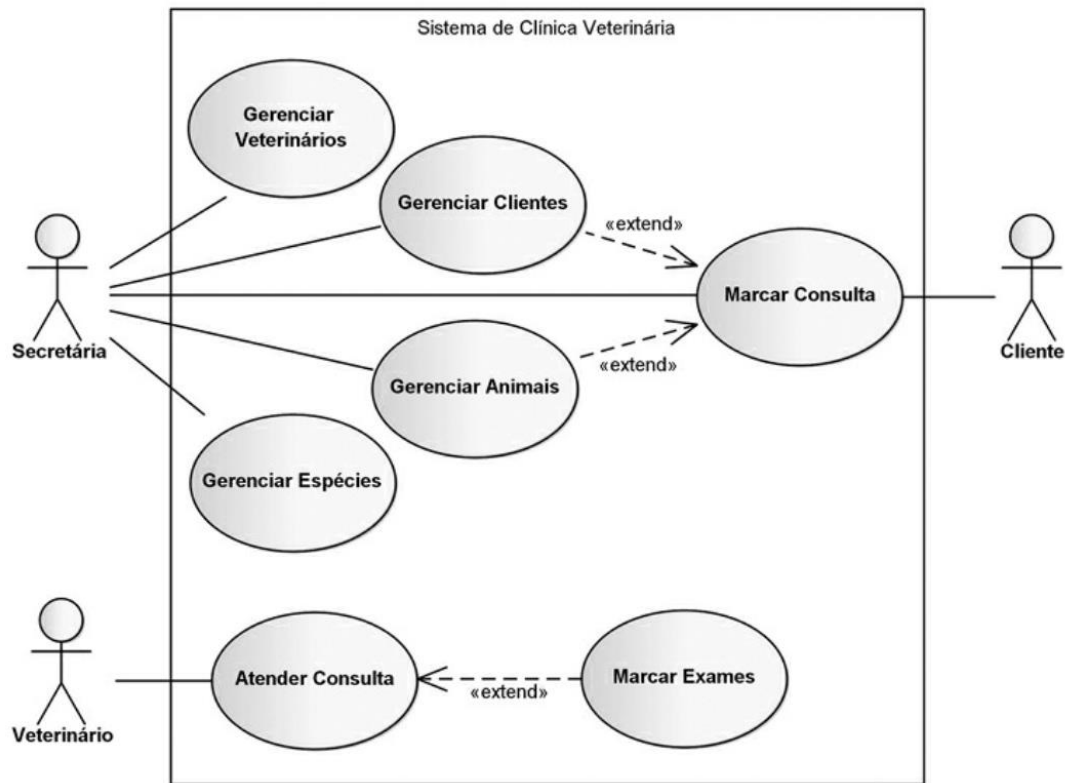
## Diagrama de Classe: Clínica Veterinária



Exemplo:  
Clínica Veterinária

# Diagrama de Caso de Uso

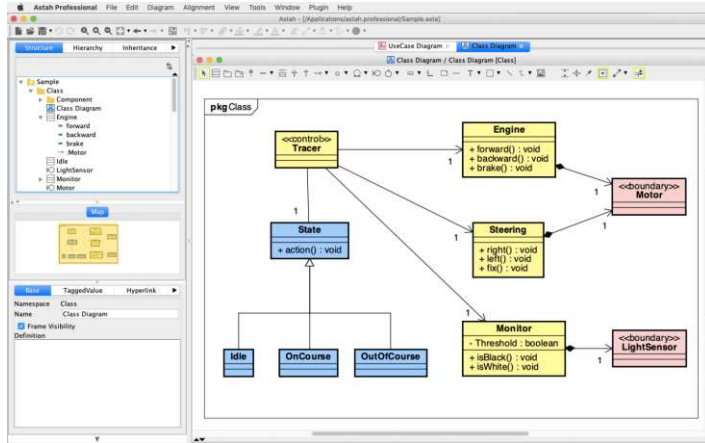
uc Modelo de Casos de Uso - Sistema de Clínica Veterinária



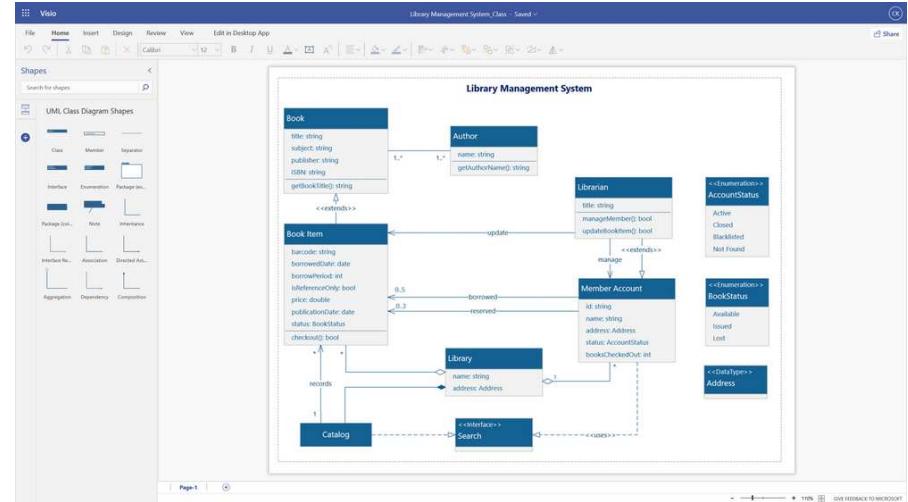


# UML - Ferramentas CASE

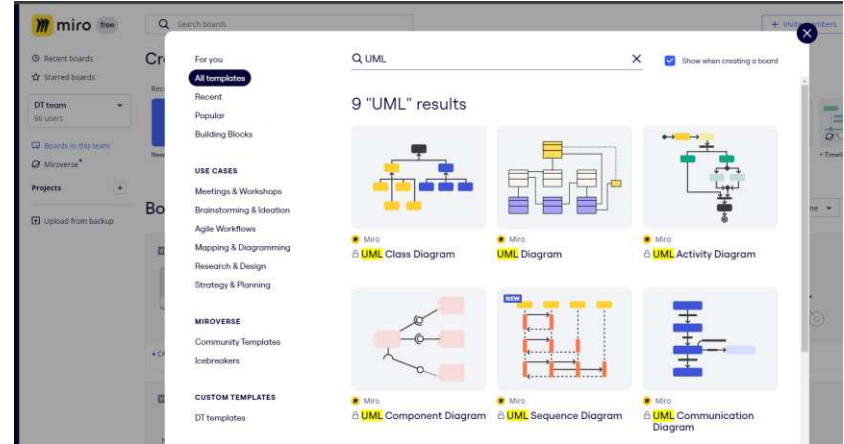
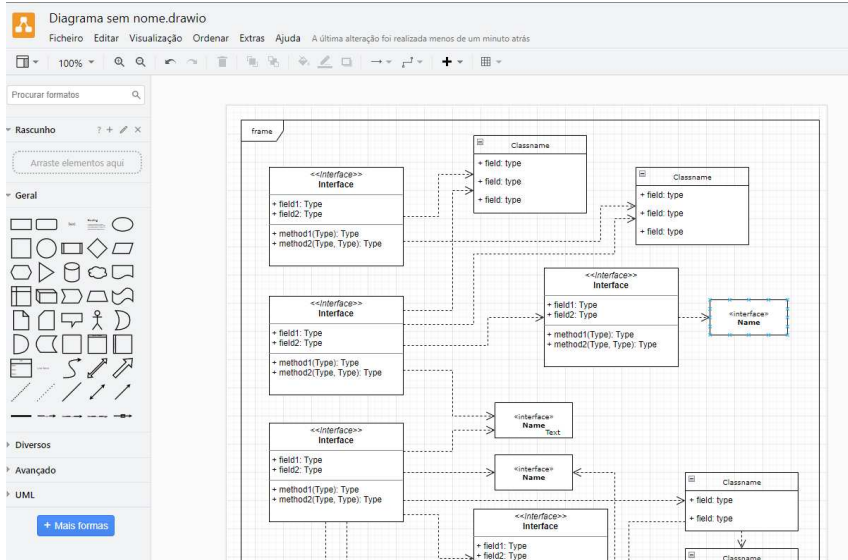
# UML - Ferramentas CASE



astah



# UML - Ferramentas CASE





“Sucesso é o  
acúmulo de  
pequenos esforços,  
repetidos dia e noite.”

Robert Collier







# UniCesumar

EDUCAÇÃO PRESENCIAL E A DISTÂNCIA