

ENGENHARIA DE SOFTWARE

Professor Me. Vladimir Geraseev Junior

***“UM MODELO DE DOMÍNIO NÃO É
UM DIAGRAMA ESPECÍFICO; ELE
É A IDEIA QUE O DIAGRAMA
PRETENDE TRANSMITIR.***

***Ele não é simplesmente o
conhecimento existente na cabeça
~~de um especialista em domínios;~~
ele é uma abstração rigorosamente
organizada e seletiva daquele
conhecimento.”***

Eric Evans

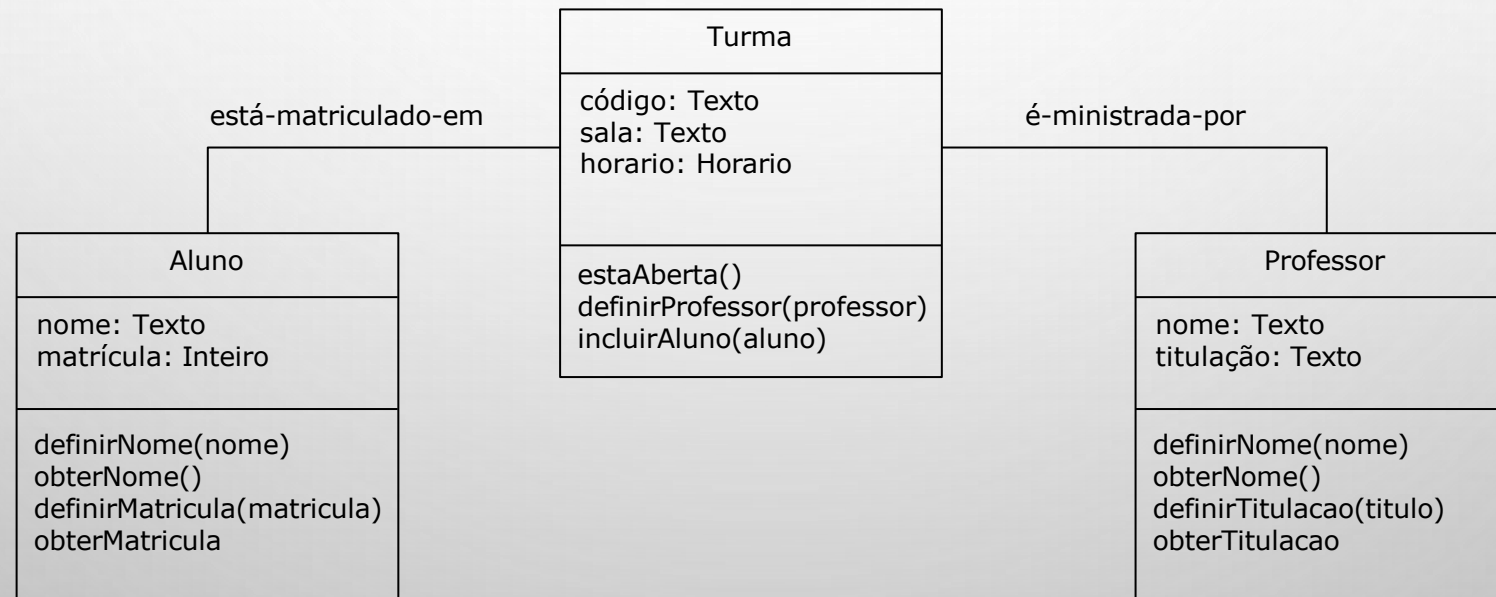
UML: DIAGRAMA DE CLASSES

UML – DIAGRAMA DE CLASSES

- Introdução – Diagrama de classes
- Elementos do diagrama de classes
- Exemplo: Sistema de matrícula

INTRODUÇÃO - DIAGRAMA DE CLASSES

- Mostra um conjunto de classes e seus relacionamentos.
- É o diagrama central da modelagem orientada a objetos.



ELEMENTOS – DIAGRAMA DE CLASSES

- ELEMENTOS DE UM DIAGRAMA DE CLASSES
 - CLASSES
 - RELACIONAMENTOS
 - ASSOCIAÇÃO
 - AGREGAÇÃO
 - COMPOSIÇÃO
 - GENERALIZAÇÃO
 - DEPENDÊNCIA

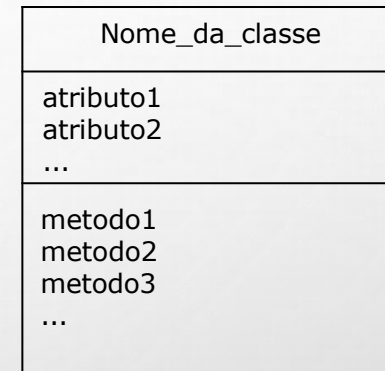
ELEMENTOS – DIAGRAMA DE CLASSES

- ELEMENTOS DE UM DIAGRAMA DE CLASSES
 - CLASSES
 - RELACIONAMENTOS
 - ASSOCIAÇÃO
 - AGREGAÇÃO
 - COMPOSIÇÃO
 - GENERALIZAÇÃO
 - DEPENDÊNCIA

ELEMENTOS – DIAGRAMA DE CLASSES

Classes

- Graficamente, as classes são representadas por retângulos incluindo nome, atributos e métodos.



- Devem receber nomes de acordo com o vocabulário do domínio do problema.
- É comum adotar um padrão para nomeá-las
Ex: todos os nomes de classes serão substantivos singulares com a primeira letra maiúscula

Classes

ELEMENTOS – DIAGRAMA DE CLASSES

- Atributos
 - Representam o conjunto de características (estado) dos objetos daquela classe
 - Visibilidade:
 - + público: visível em qualquer classe de qualquer pacote
 - # protegido: visível para classes do mesmo pacote
 - privado: visível somente para classe

Exemplo:

+ nome : String

Classes

ELEMENTOS – DIAGRAMA DE CLASSES

- Métodos
 - Representam o conjunto de operações (comportamento) que a classe fornece
 - Visibilidade:
 - + público: visível em qualquer classe de qualquer pacote
 - # protegido: visível para classes do mesmo pacote
 - privado: visível somente para classe

Exemplo:

- getNome() : String

ELEMENTOS – DIAGRAMA DE CLASSES

- ELEMENTOS DE UM DIAGRAMA DE CLASSES
 - CLASSES
 - RELACIONAMENTOS
 - ASSOCIAÇÃO
 - AGREGAÇÃO
 - COMPOSIÇÃO
 - GENERALIZAÇÃO
 - DEPENDÊNCIA

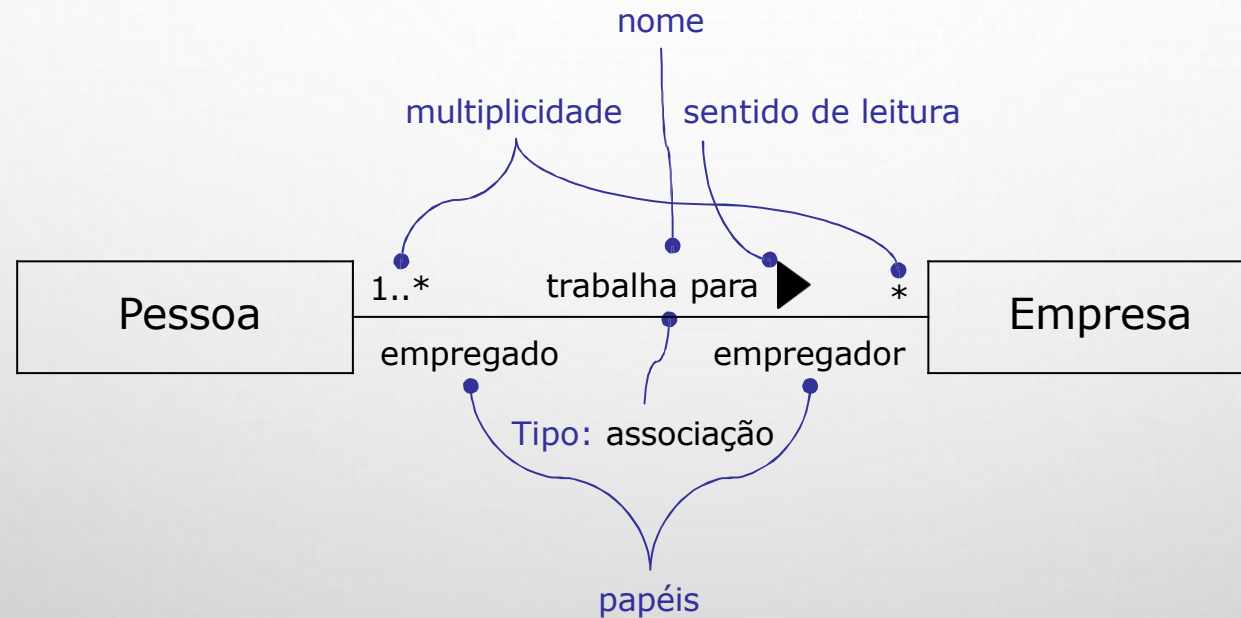
ELEMENTOS – DIAGRAMA DE CLASSES

Relacionamentos

- Os relacionamentos possuem:
 - **Nome:** descrição dada ao relacionamento (faz, tem, possui,...)
 - **Sentido de leitura**
 - **Navegabilidade:** indicada por uma seta no fim do relacionamento
 - **Multiplicidade:** 0..1, 0..*, 1, 1..*, 2, 3..7
 - **Tipo:** associação (agregação, composição), generalização e dependência
 - **Papéis:** desempenhados por classes em um relacionamento

ELEMENTOS – DIAGRAMA DE CLASSES

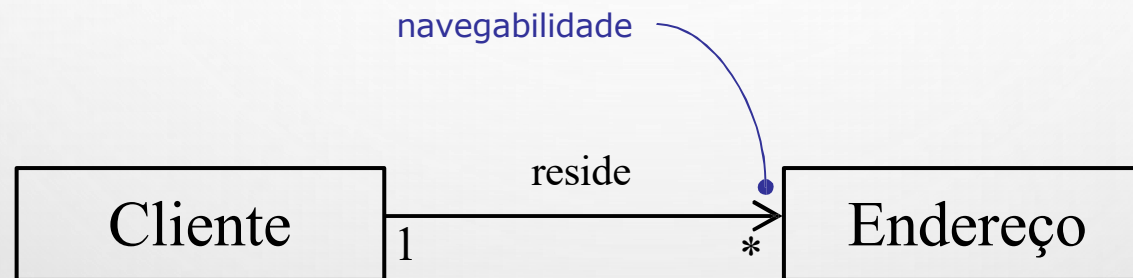
- Relacionamentos



E a navegabilidade?

ELEMENTOS – DIAGRAMA DE CLASSES

- Relacionamentos



- O cliente sabe quais são seus endereços, mas o endereço não sabe a quais clientes pertence

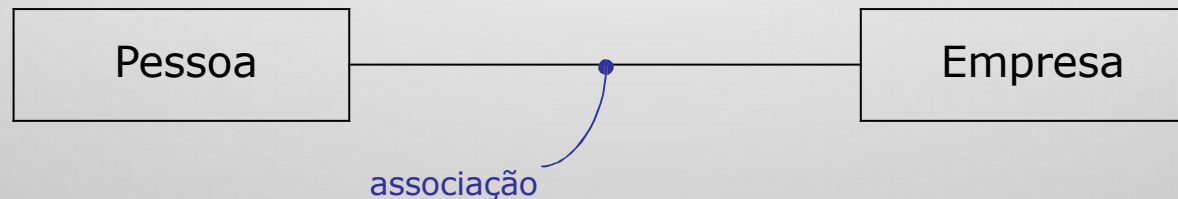
ELEMENTOS – DIAGRAMA DE CLASSES

- ELEMENTOS DE UM DIAGRAMA DE CLASSES
 - CLASSES
 - RELACIONAMENTOS
 - ASSOCIAÇÃO
 - AGREGAÇÃO
 - COMPOSIÇÃO
 - GENERALIZAÇÃO
 - DEPENDÊNCIA

ELEMENTOS – DIAGRAMA DE CLASSES

Relacionamentos: Associação

- Uma associação é um relacionamento estrutural que indica que os objetos de uma classe estão vinculados a objetos de outra classe.
- Uma associação é representada por uma linha sólida conectando duas classes.



ELEMENTOS – DIAGRAMA DE CLASSES

Relacionamentos: Associação

- Indicadores de multiplicidade:
 - 1 Exatamente um
 - 1..* Um ou mais
 - 0..* Zero ou mais (muitos)
 - * Zero ou mais (muitos)
 - 0..1 Zero ou um
 - m..n Faixa de valores (por exemplo: 4..7)

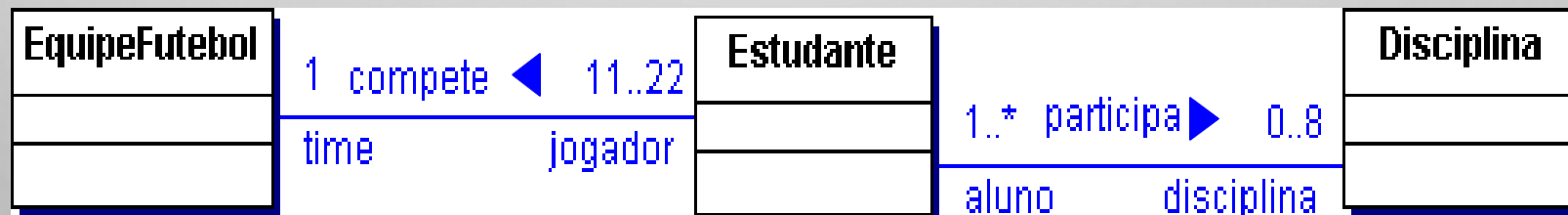


RELACIONAMENTOS: ASSOCIAÇÃO

Relacionamentos: Associação

Exemplo:

- Um Estudante pode ser
um **aluno** de uma Disciplina e
um **jogador** da Equipe de Futebol
- Cada Disciplina deve ser cursada por no mínimo 1 aluno
- Um aluno pode cursar de 0 até 8 disciplinas

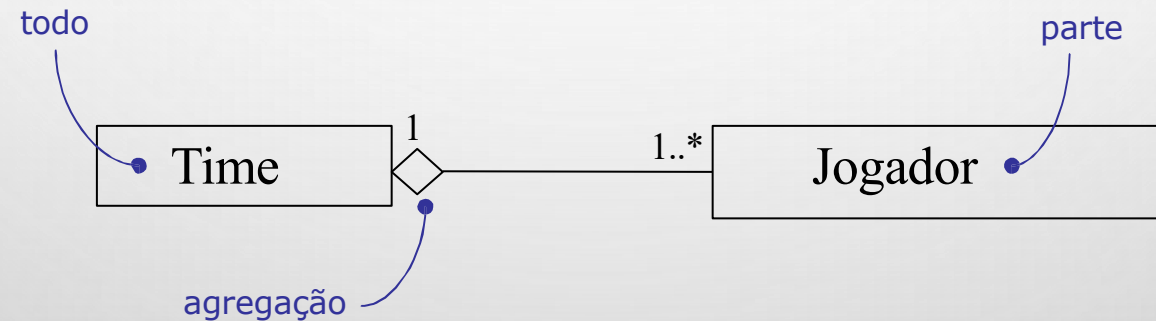


ELEMENTOS – DIAGRAMA DE CLASSES

- ELEMENTOS DE UM DIAGRAMA DE CLASSES
 - CLASSES
 - RELACIONAMENTOS
 - ASSOCIAÇÃO
 - AGREGAÇÃO
 - COMPOSIÇÃO
 - GENERALIZAÇÃO
 - DEPENDÊNCIA

ELEMENTOS – DIAGRAMA DE CLASSES

- Relacionamento: Agregação
 - É um tipo especial de associação
 - Utilizada para indicar “todo-parte”



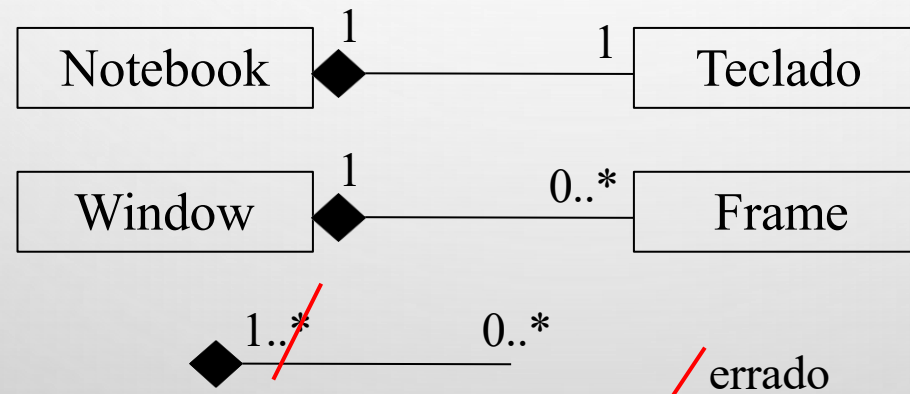
- um objeto “parte” pode fazer parte de vários objetos “todo”

ELEMENTOS – DIAGRAMA DE CLASSES

- ELEMENTOS DE UM DIAGRAMA DE CLASSES
 - CLASSES
 - RELACIONAMENTOS
 - ASSOCIAÇÃO
 - AGREGAÇÃO
 - COMPOSIÇÃO
 - GENERALIZAÇÃO
 - DEPENDÊNCIA

ELEMENTOS – DIAGRAMA DE CLASSES

- Relacionamento: Composição
 - É uma variante semanticamente mais “forte” da agregação
 - Os objetos “parte” só podem pertencer a um único objeto “todo” e têm o seu tempo de vida coincidente com o dele

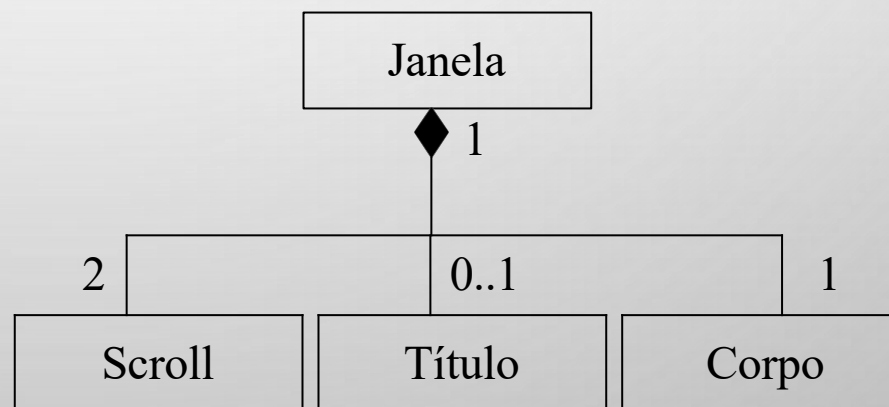
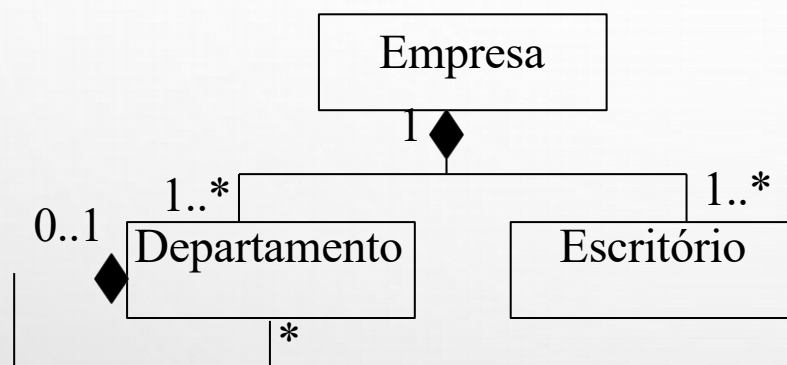


- Quando o “todo” *morre* todas as suas “partes” também *morrem*

ELEMENTOS – DIAGRAMA DE CLASSES

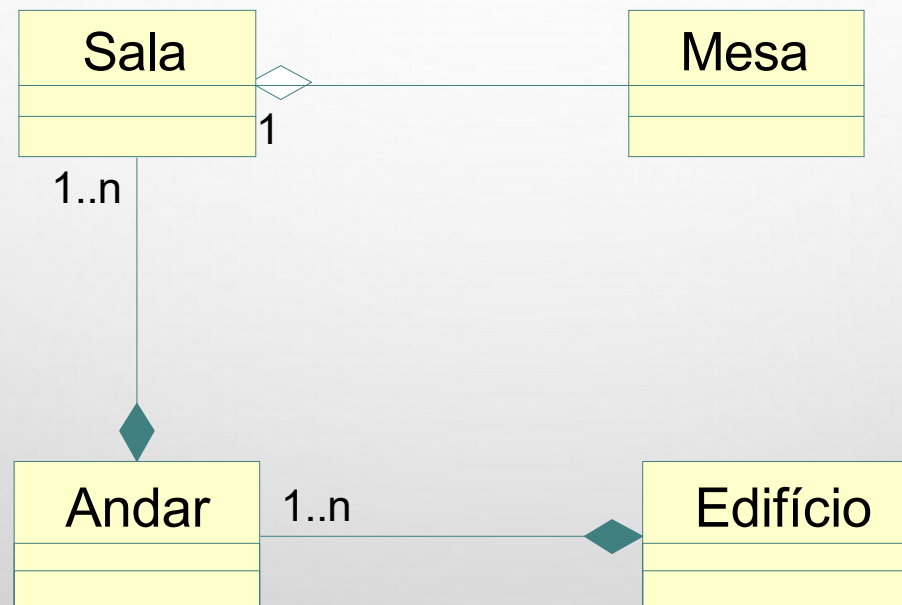
- Relacionamento: Composição

Ex:



ELEMENTOS – DIAGRAMA DE CLASSES

- Agregação X Composição



ELEMENTOS – DIAGRAMA DE CLASSES

- ELEMENTOS DE UM DIAGRAMA DE CLASSES
 - CLASSES
 - RELACIONAMENTOS
 - ASSOCIAÇÃO
 - AGREGAÇÃO
 - COMPOSIÇÃO
 - GENERALIZAÇÃO
 - DEPENDÊNCIA

ELEMENTOS – DIAGRAMA DE CLASSES

- Relacionamento: Generalização
 - É um relacionamento entre itens gerais (superclasses) e itens mais específicos (subclasses)



ELEMENTOS – DIAGRAMA DE CLASSES

- ELEMENTOS DE UM DIAGRAMA DE CLASSES
 - CLASSES
 - RELACIONAMENTOS
 - ASSOCIAÇÃO
 - AGREGAÇÃO
 - COMPOSIÇÃO
 - GENERALIZAÇÃO
 - DEPENDÊNCIA

ELEMENTOS – DIAGRAMA DE CLASSES

- Relacionamento: Dependência
 - Representa que a alteração de um objeto (o objeto independente) pode afetar outro objeto (o objeto dependente)

Ex:

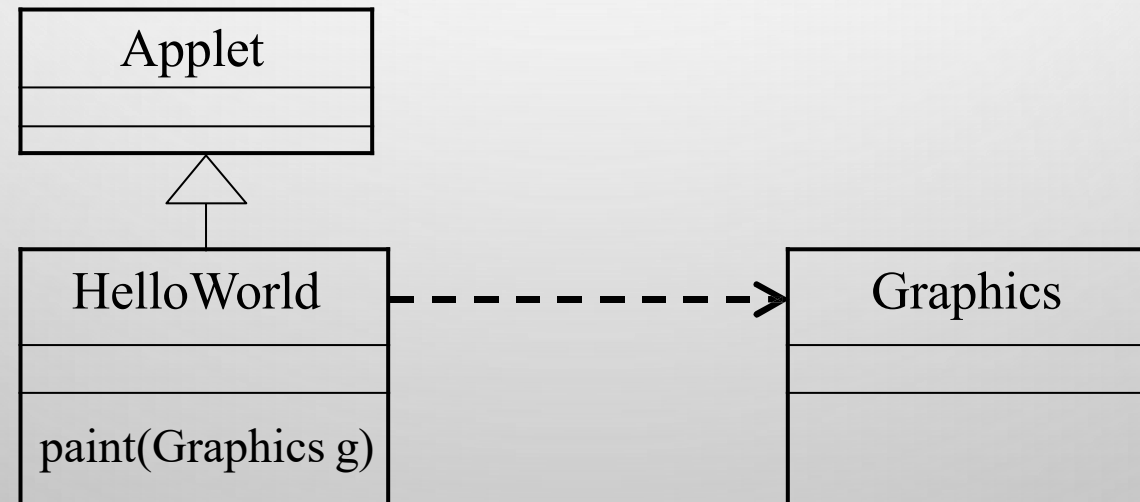


Obs:

- A classe cliente depende de algum serviço da classe fornecedor
- A mudança de estado do fornecedor afeta o objeto cliente
- A classe cliente não declara nos seus atributos um objeto do tipo fornecedor
- Fornecedor é recebido por parâmetro de método

ELEMENTOS – DIAGRAMA DE CLASSES

```
Import java.awt.Graphics;  
class HelloWorld extends java.applet.Applet  
{  
    public void paint (Graphics g)  
        g.drawString("Hello, world!", 10, 10);  
}
```

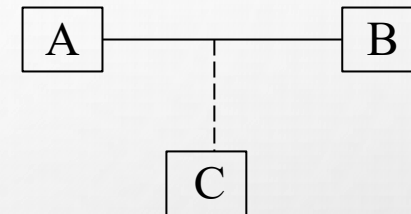


ELEMENTOS – DIAGRAMA DE CLASSES

- Classe de associação

- Usada quando uma associação entre duas classes contiver atributos da associação

- Atributos farão parte da classe de associação
- C existe para todo relacionamento de A com B



- C possui referência para A e para B

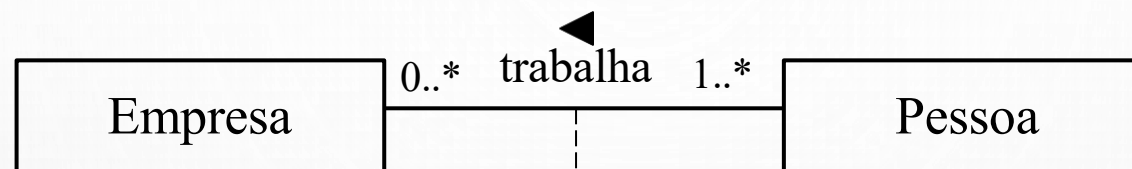
- Não existem dois objetos C e C' que referenciam a mesma tupla A,B, isto é, não existe $c(a,b,x)$ e $c'(a,b,x')$ onde a e b são objetos de A e B, respectivamente, e x e x' são valores de um atributo de C.

- $d(a,b,x)$ e $d(a,b,x')$ existem

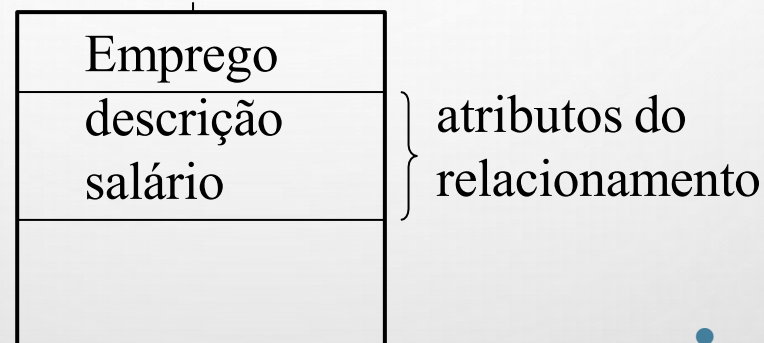


ELEMENTOS – DIAGRAMA DE CLASSES

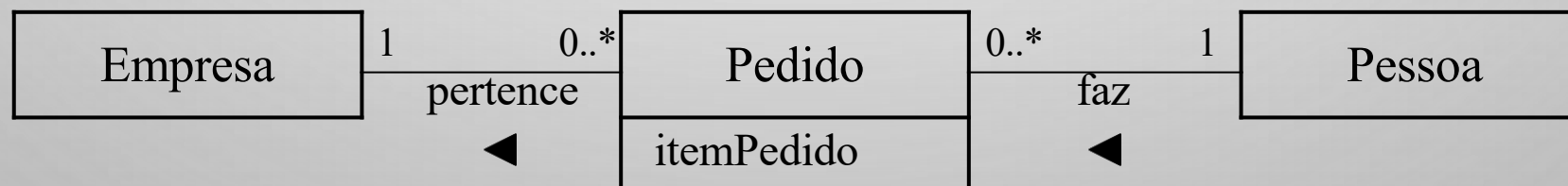
- Classe de associação



- Não existe uma pessoa com dois empregos na mesma empresa



- Uma pessoa pode fazer mais de um pedido na mesma empresa



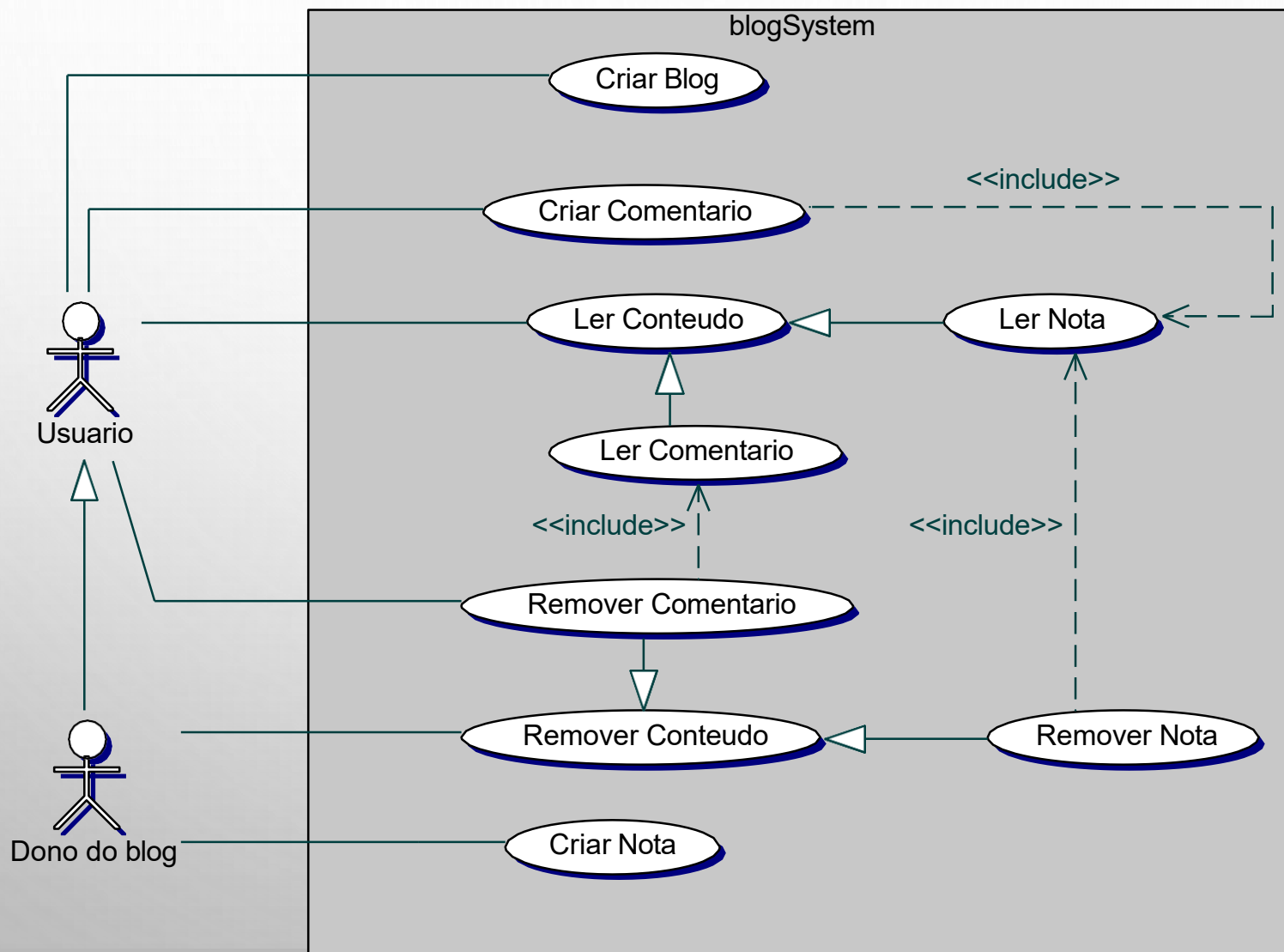
EXEMPLO: O BLOG

- Um *blog* tem um título e uma data de criação e além disso é um conjunto de conteúdos.
- Estes conteúdos (mensagens) podem ser notas ou comentários sobre as notas. Tanto notas quanto comentários têm características comuns como o texto e a data de sua criação.
- Todo usuário possui:
 - E-mail (deve ser único, ou seja, não há mais de um usuário com o mesmo e-mail)

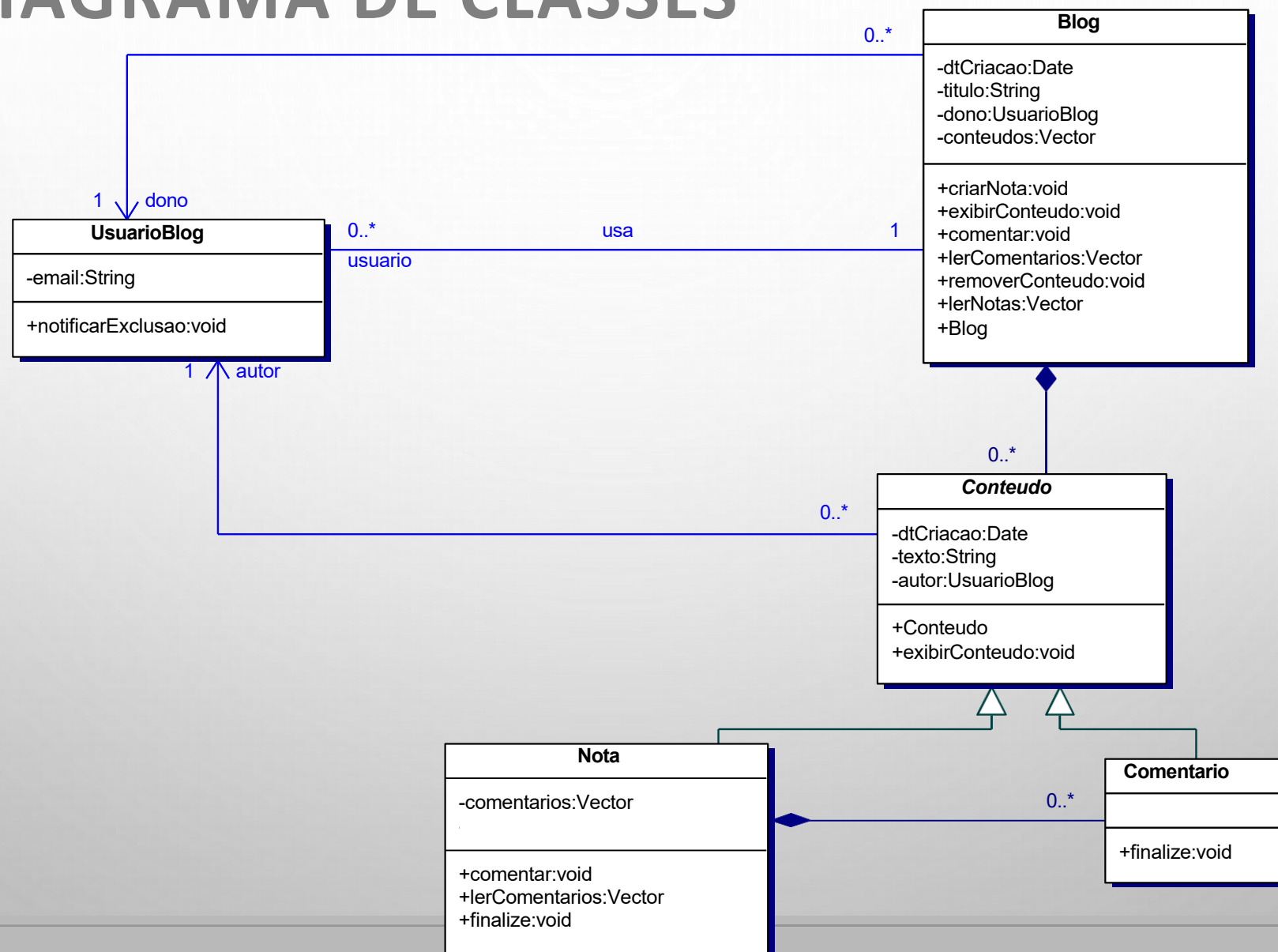
BLOG: O SISTEMA DEVE...

- Permitir a criação de blogs
- Permitir a utilização de blogs
 - Qualquer usuário pode ler conteúdos
 - Somente o dono do blog pode criar notas
 - Qualquer usuário pode criar comentários. Para criar um comentário o usuário precisa ler as notas.
 - Somente o dono do blog pode remover conteúdos. Para remover um conteúdo ele precisará ler o conteúdo. Caso ele remova um comentário, o autor do comentário deve ser notificado por e-mail.

BLOG: CASOS DE USO



BLOG: DIAGRAMA DE CLASSES



EXEMPLO: SISTEMA DE MATRÍCULA

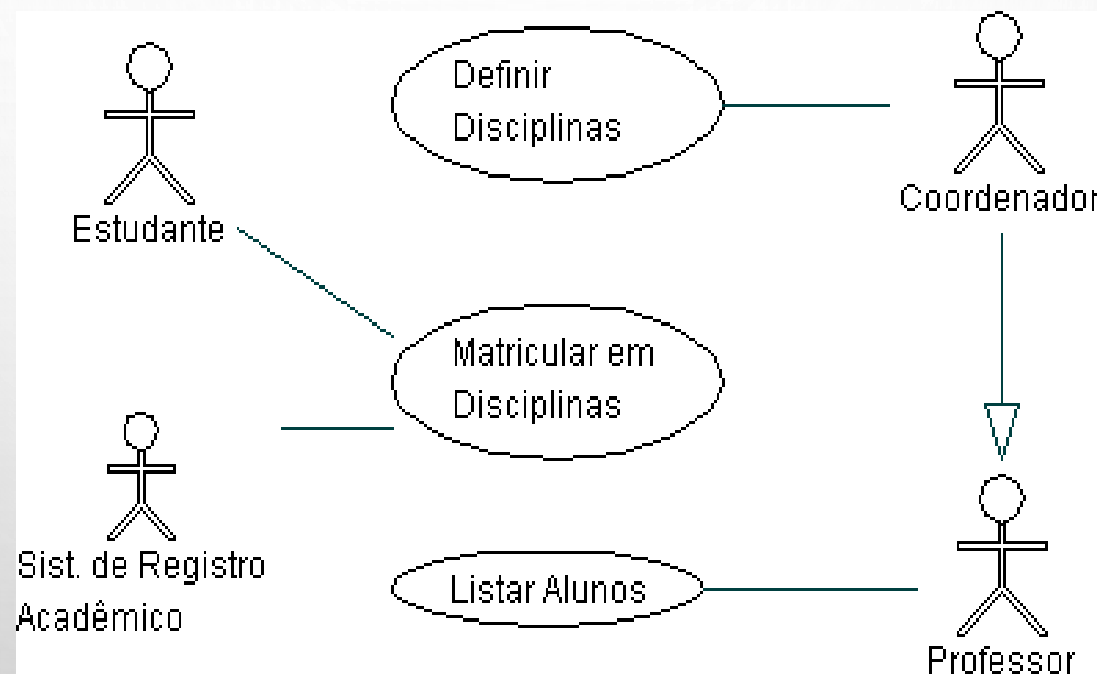
Descrição

A Universidade XYZ deseja informatizar seu sistema de matrículas:

- A universidade oferece vários cursos.
- O Coordenador de um curso define as disciplinas que serão oferecidas pelo seu curso num dado semestre.
- Várias disciplinas são oferecidas em um curso.
- Várias turmas podem ser abertas para uma mesma disciplina, porém o número de estudantes inscritos deve ser entre 3 e 10.
- Estudantes selecionam 4 disciplinas.
- Quando um estudante matricula-se para um semestre, o Sistema de Registro Acadêmico (SRA) é notificado.
- Após a matrícula, os estudantes podem, por um certo prazo, utilizar o sistema para adicionar ou remover disciplinas.
- Professores usam o sistema para obter a lista de alunos matriculados em suas disciplinas. O Coordenador também.
- Todos os usuários do sistema devem ser validados.

Diagrama de Casos de Uso

EXEMPLO: SISTEMA DE MATRÍCULA



EXEMPLO: SISTEMA DE MATRÍCULA

Descrição do Caso de Uso “Matricular em Disciplina”

- Esse caso de uso se inicia quando o Estudante de Curso inicia uma sessão no sistema e apresenta suas credenciais.
- O sistema verifica se a credencial é válida.
- O sistema solicita que o estudante realize sua matrícula, selecionando 4 disciplinas.
- O estudante preenche um formulário eletrônico de matrícula e o submete para uma análise de consistência.
- O sistema analisa as informações contidas no formulário.
 - Se as informações são consistentes, o estudante é incluído em turmas abertas de 4 disciplinas, iniciando pelas preferenciais.
 - Se as informações não são consistentes, o sistema informa o motivo da inconsistência e solicita que o formulário seja alterado.

Diagrama de Classes: identificando as classes

Professor

Coordenador

Estudante

Universidade

Disciplina

Turma

Curso

FormularioMatricula

AnalizadorMatricula

SistemaRegistroAcademico

ListaAlunos

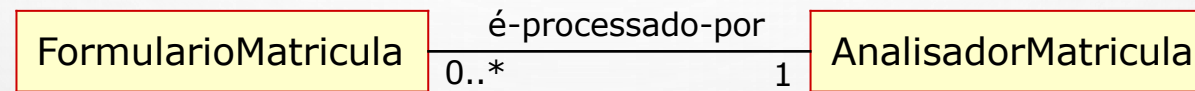
EXEMPLO: SISTEMA DE MATRÍCULA

Diagrama de Classes: identificando os relacionamentos

- Exemplos de candidatos a relacionamentos:
 - A é parte física ou lógica de B.
 - A está contido fisicamente ou logicamente em B.
 - A é uma descrição de B.
 - A é membro de B.
 - A é subunidade organizacional de B.
 - A usa ou gerencia B.
 - A se comunica/interage com B.
 - A está relacionado com uma transação B.
 - A é possuído por B.
 - A é um tipo de B.

Diagrama de Classes: identificando os relacionamentos

- O formulário de matrícula é processado por um analisador de matrícula



- O analisador de matrícula gerencia a disciplina

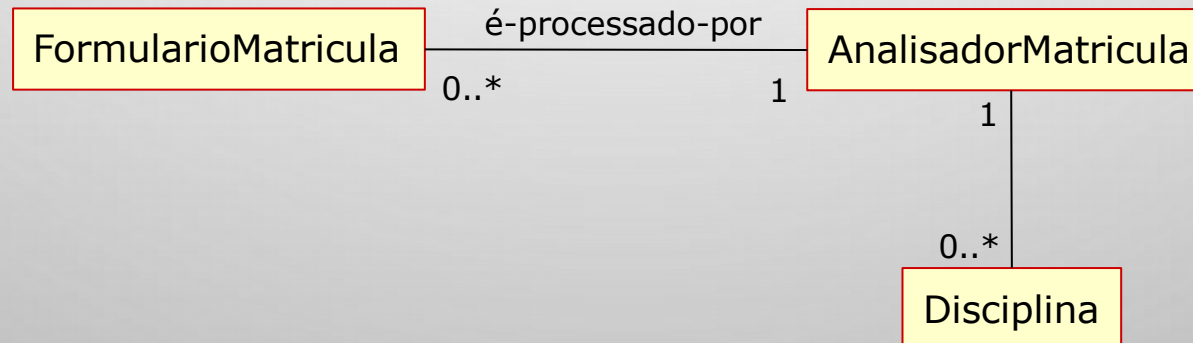


Diagrama de Classes

EXEMPLO: SISTEMA DE MATRÍCULA

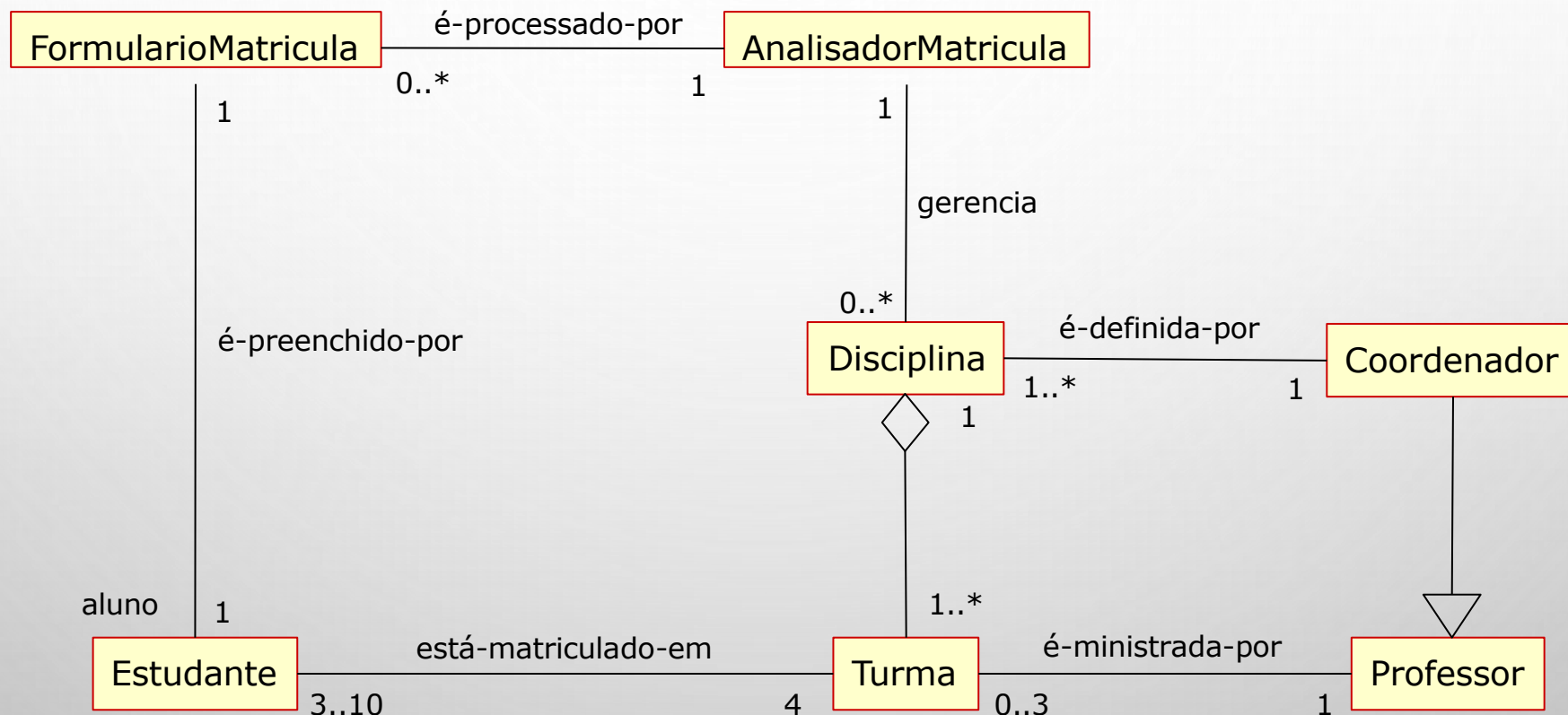
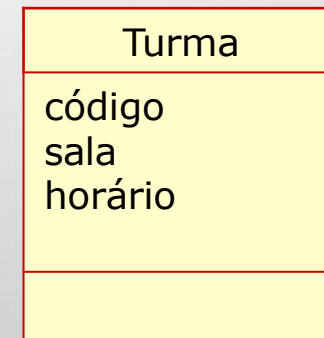


Diagrama de Classes: identificando os atributos

- Os atributos podem ser encontrados examinando-se as descrições dos casos de uso e também pelo conhecimento do domínio do problema.
- Cada turma oferecida possui um código, uma sala e um horário.



EXEMPLO: SISTEMA DE MATRÍCULA

Diagrama de Classes

EXEMPLO: SISTEMA DE MATRÍCULA

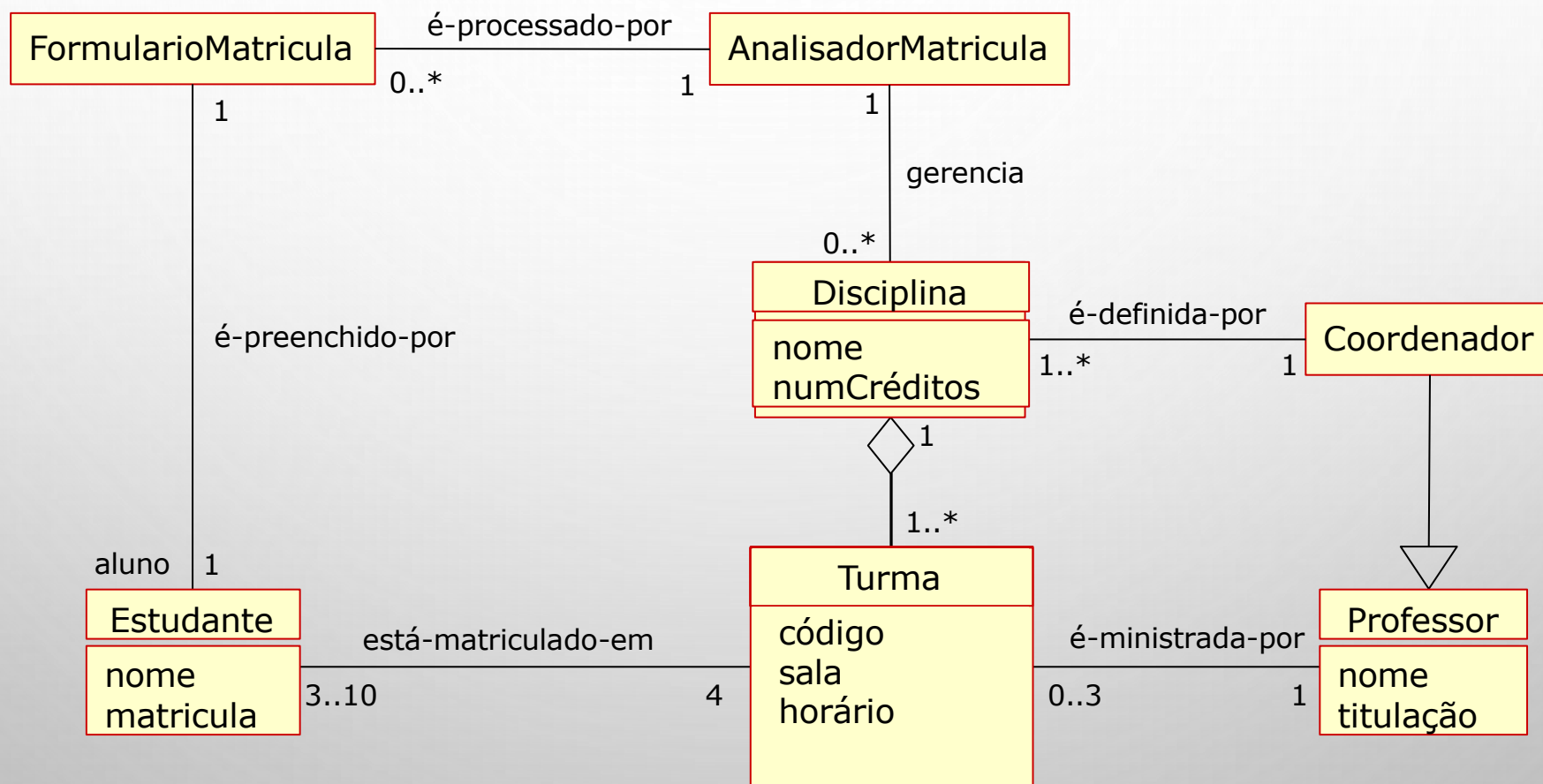
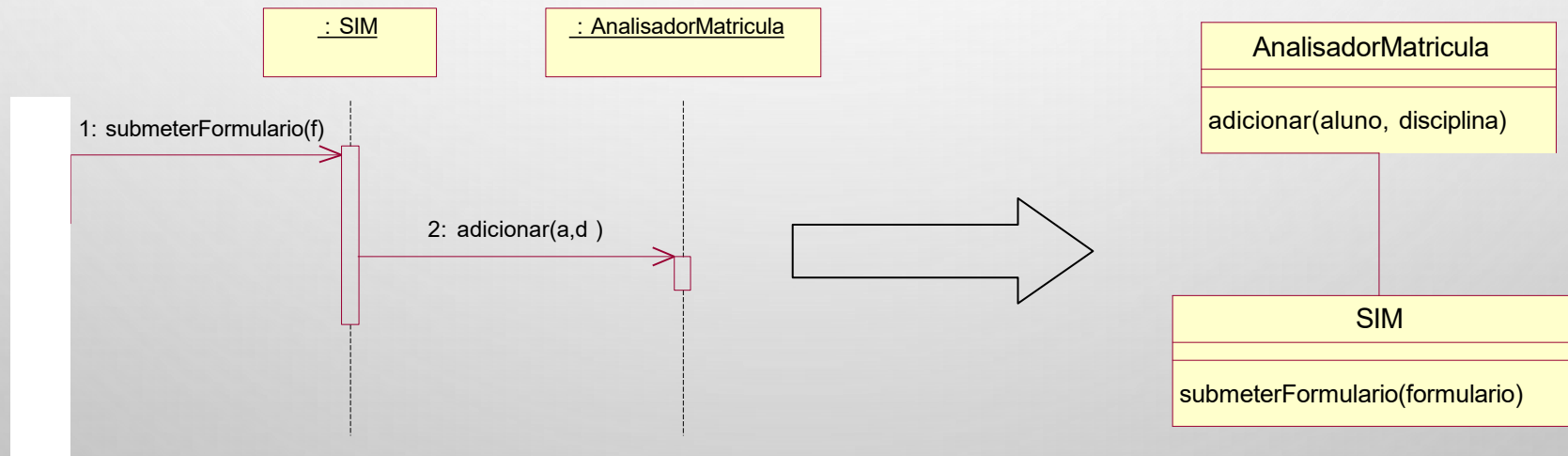


Diagrama de Classes: identificando os métodos



EXEMPLO: SISTEMA DE MATRÍCULA

Diagrama de Classes:

- E a navegabilidade?



```
public class Estudante {
    private String nome;
    private String matricula;
    ...
}
```

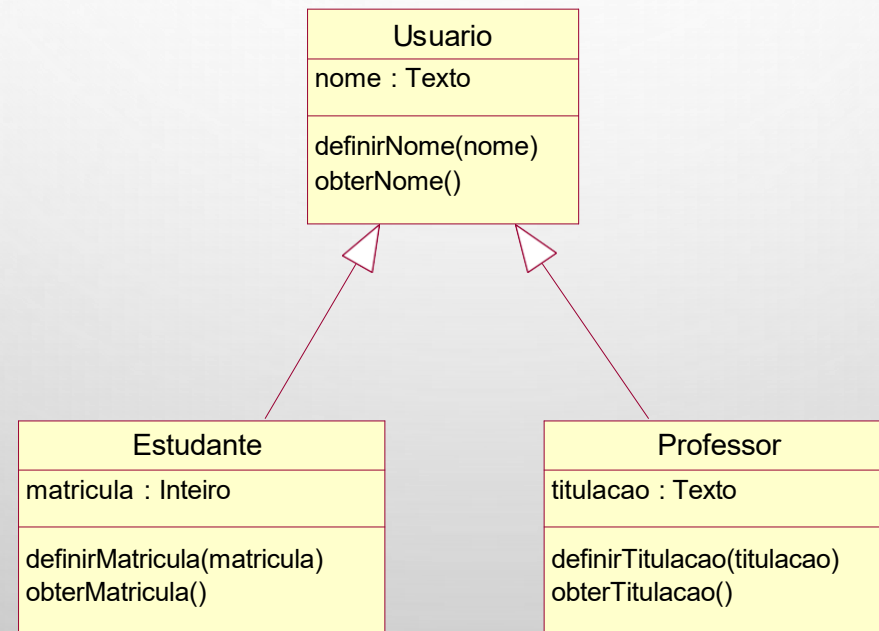
```
public class Turma {
    private String codigo;
    private String sala;
    private Estudante alunos[];
    ...
}
```

EXEMPLO: SISTEMA DE MATRÍCULA

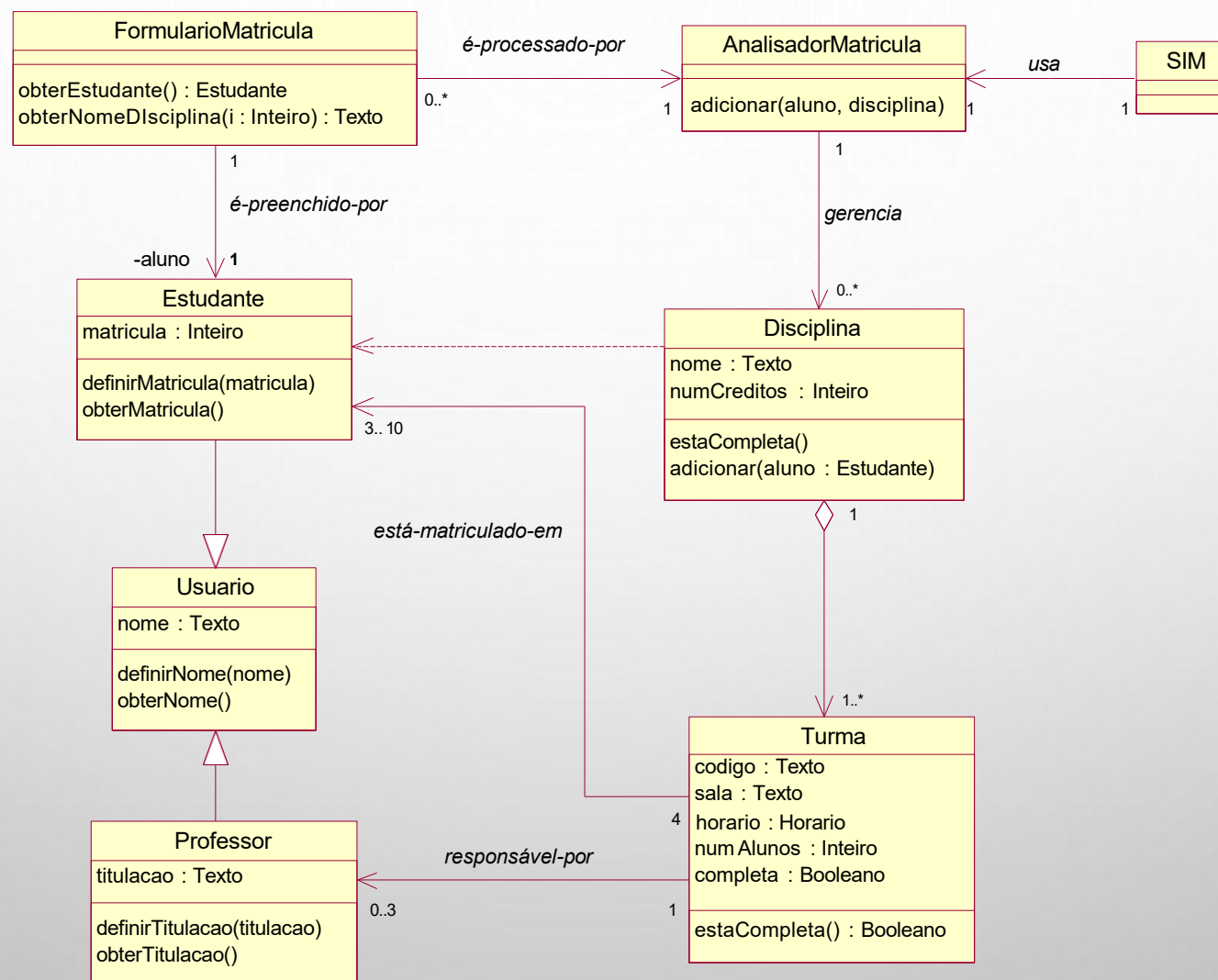
OBS: Turma não aparece como atributo de Estudante!

Diagrama de Classes:

- Acrescentando generalizações:
 - Atributos, operações e/ou relacionamentos comuns podem ser movidos para uma classe mais geral.



EXEMPLO: SISTEMA DE MATRÍCULA



EXEMPLO: SISTEMA DE MATRÍCULA

EXERCÍCIO

- ELABORE A DESCRIÇÃO DOS CASOS DE USO, DIAGRAMA DOS CASOS DE USO E DIAGRAMA DE CLASSES PARA UM SISTEMA DE PONTO DE VENDAS.
- PARA ELABORAR OS DIAGRAMAS UTILIZAR O SEGUINTE SITE:

<https://draw.io>

EXERCÍCIO

- O GERENTE DEVE FAZER LOGIN COM UM ID E SENHA PARA INICIAR E FINALIZAR O SISTEMA;
- O CAIXA (OPERADOR) DEVE FAZER LOGIN COM UM ID E SENHA PARA PODER UTILIZAR O SISTEMA;
- DESCREVER A OPERAÇÃO DE VENDA
- EXIBIR UMA TELA PARA CALCULAR O TOTAL DE VENDAS
- REDUZIR AS QUANTIDADES EM ESTOQUE QUANDO A VENDA É CONFIRMADA;