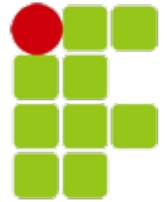


INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
BAIANO CAMPUS GUANAMBI

ANÁLISE DE SISTEMAS

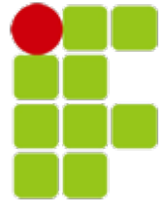
Diagrama de Classes

Conceitos Básicos



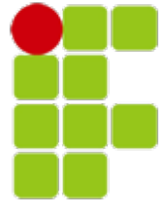
- O caso de uso fornece uma perspectiva do sistema de um ponto de vista externo (do ator)
- Internamente os objetos colaboram para atender às funcionalidades do sistema
- Demonstra a estrutura estática dessa colaboração, mostra as classes de um sistema, seus atributos e operações, e como as classes se relacionam.
- O diagrama de objetos (que pode ser visto como uma instanciamento do diagrama de classes) também representa a estrutura estática

Perspectivas de um Diagrama de Classes



- O **modelo conceitual** (análise) representa as classes no domínio do negócio em questão. Não leva em consideração restrições inerentes à tecnologia a ser utilizada na solução de um problema.
- O **modelo de classes de especificação** (projeto) é obtido através da adição de detalhes ao modelo anterior conforme a solução de software escolhida.
- O **modelo de classes de implementação** corresponde à implementação das classes em alguma linguagem de programação.

Definição de Classes



Uma classe de objetos, ou simplesmente classe, descreve :

- Um conjunto de objetos com propriedades semelhantes (atributos);
- O mesmo comportamento (operações);
- Os mesmos relacionamentos; e
- A mesma semântica.

São os blocos de construção mais importantes de qualquer sistema orientado a objetos;

Conceitos Básicos

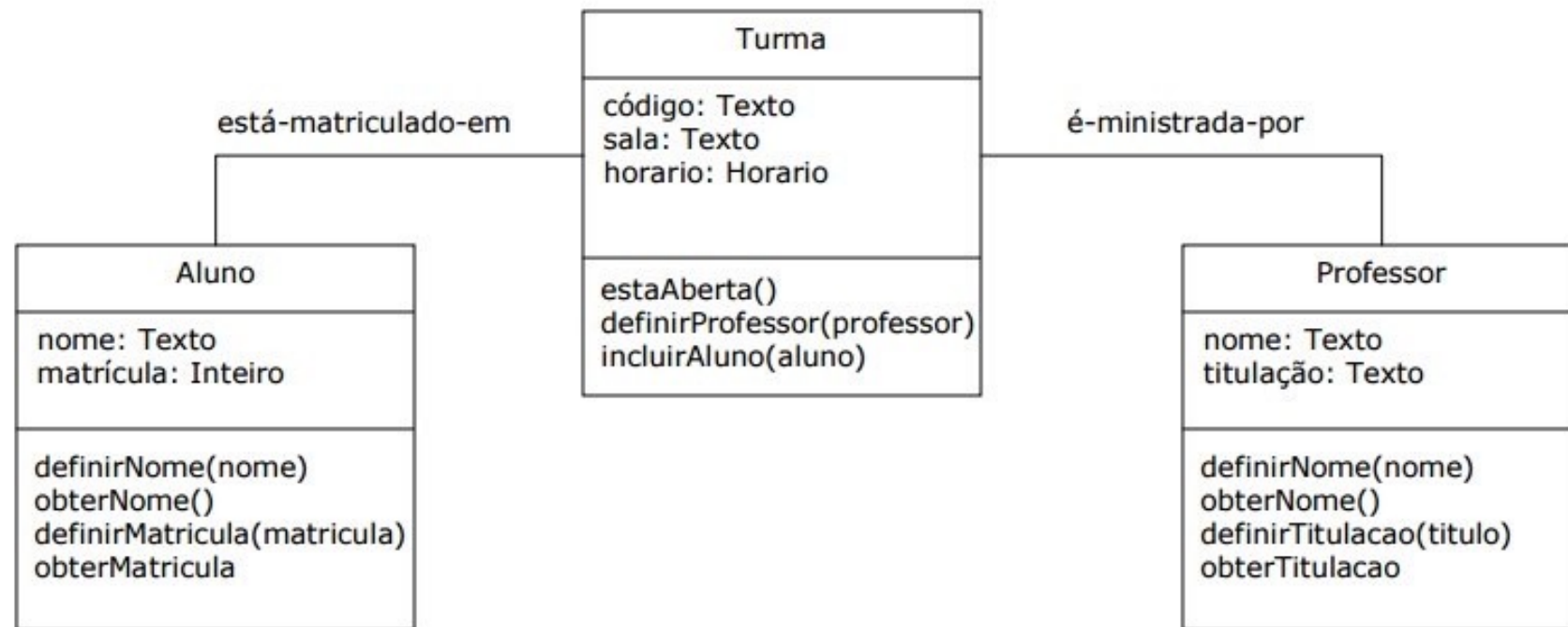
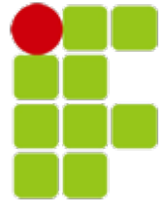
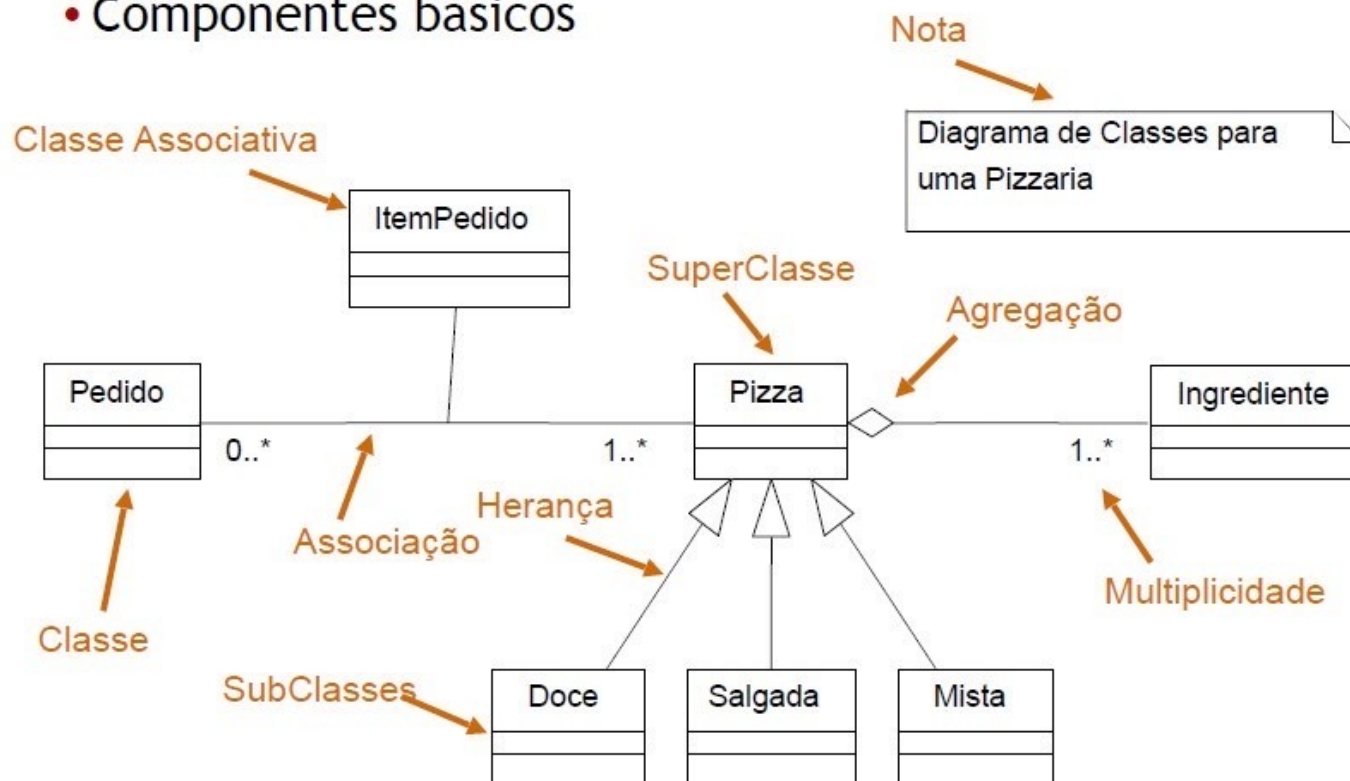


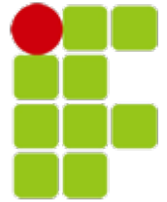
Diagrama de Classes



- Componentes básicos



Conceitos Básicos: Classes



- Representada através de uma “caixa” com no máximo três compartimentos exibidos (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.

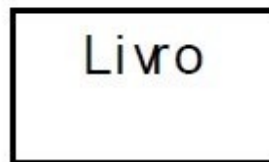
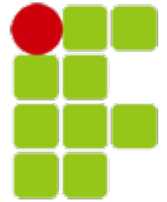
Nome da Classe

Nome da Classe
lista de atributos

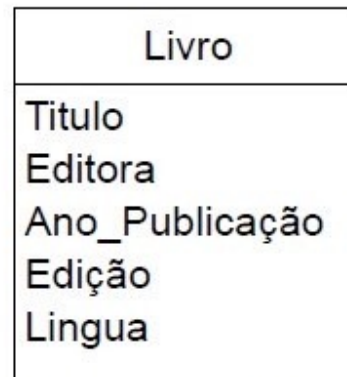
Nome da Classe
lista de operações

Nome da Classe
lista de atributos
lista de operações

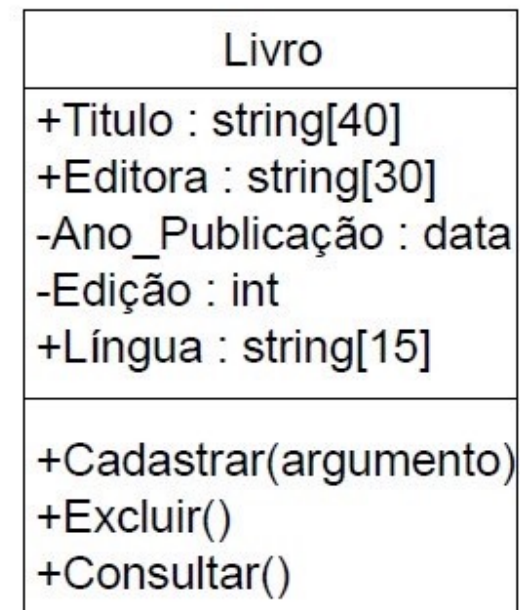
Conceitos Básicos: Classes



Classe com detalhes
suprimidos

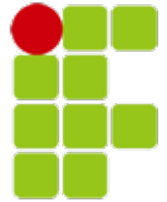


Classe com detalhes
a nível de análise



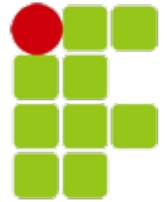
Classe com detalhes
a nível de implementação

Conceitos Básicos: Atributos



- Representam uma propriedade do objeto que está sendo modelado;
- Cada atributo possui um valor para cada instância de um objeto;
- Cada nome de atributo deve ser único dentro de uma classe;
- Podem ser representados apenas por seu nome, ou especificando seu tipo e valor inicial;

Conceitos Básicos: Atributos



- Representam o conjunto de características (estado) dos objetos daquela classe

- Visibilidade:

 - + público: visível para qualquer classe

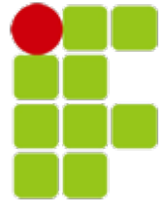
 - # protegido: visível somente para classes derivadas

 - privado: visível somente para classe

Exemplo:

 - + nome : String

Conceitos Básicos: Atributos



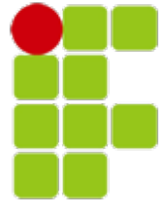
Livro
Titulo
Editora
Ano_Publicação
Edição
Lingua

Atributos Simples

Livro
+Titulo : string[40] +Editora : string[30] -Ano_Publicação : data -Edição : int +Língua : string[15]
+Cadastrar(argumento) +Excluir() +Consultar()

Atributos Completos

Conceitos Básicos: Métodos

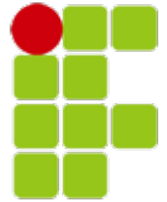


- Representam o conjunto de operações (comportamento) que a classe fornece
- Visibilidade:
 - + público: visível para qualquer classe
 - # protegido: visível somente para classes derivadas
 - privado: visível somente para classe

Exemplo:

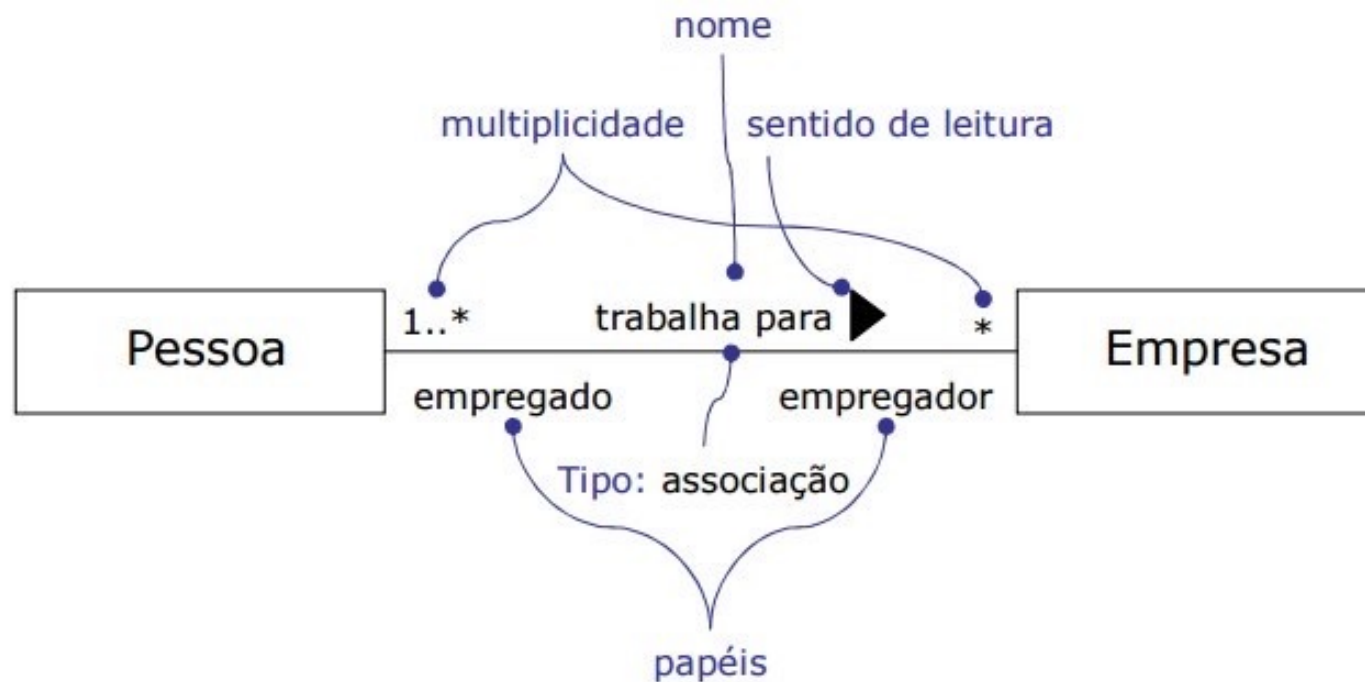
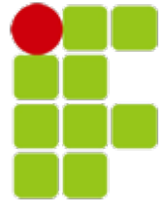
- `getNome() : String`

Classes: Relacionamentos



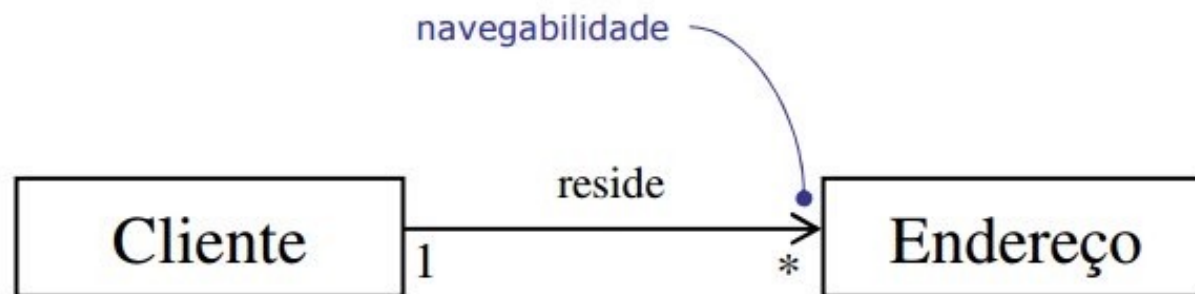
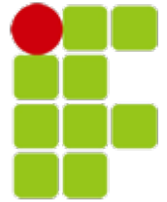
- Os relacionamentos possuem:
 - Nome: descrição dada ao relacionamento (faz, tem, possui,...)
 - Sentido de leitura
 - Navegabilidade: indicada por uma seta no fim do relacionamento
 - Multiplicidade: 0..1, 0..*, 1, 1..*, 2, 3..7
 - Tipo: associação (agregação, composição), generalização e dependência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

Classes: Relacionamentos



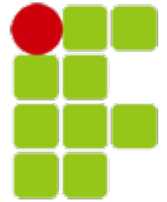
E a navegabilidade?

Classes: Relacionamentos



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

Classes: Relacionamentos

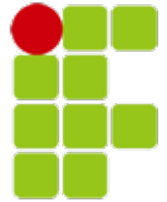


Relacionamentos: **Associação**

- 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.



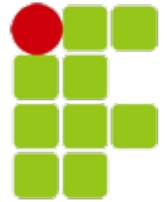
Classes: Relacionamentos



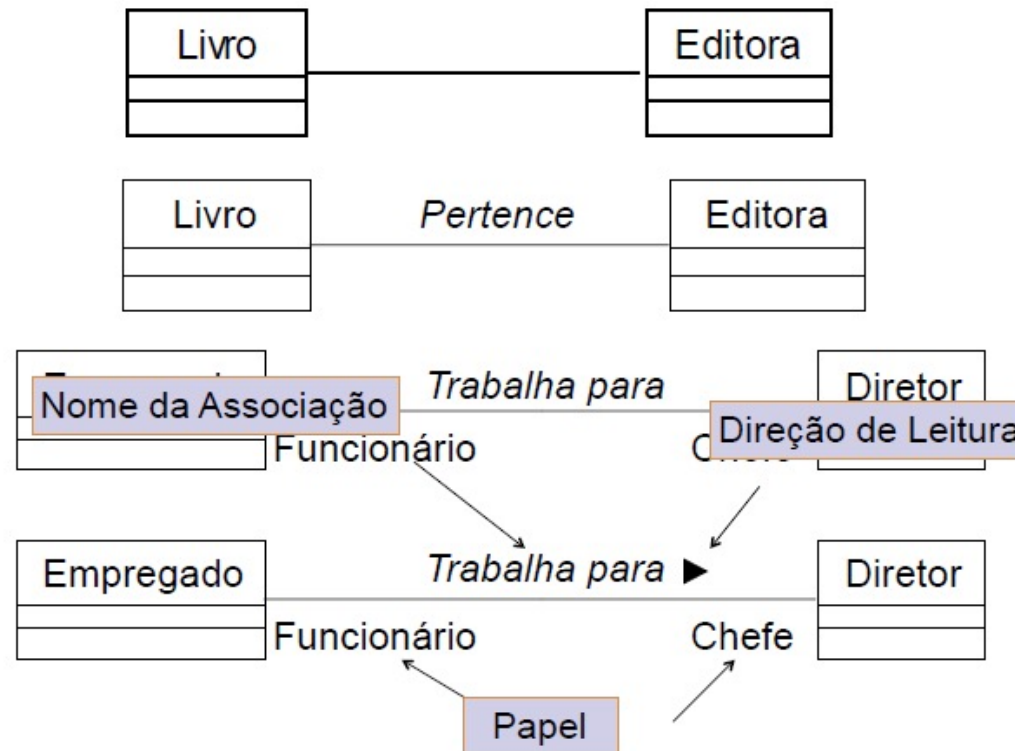
Relacionamentos: **Associação**

- Associações Normais (binárias)
- Tipo mais comum de associação, sendo uma conexão entre duas classes;
- Uma associação pode possuir um nome, que normalmente é um verbo (substantivos são aceitos);
- Cada terminação da associação pode possuir um papel, mostrando como uma classe é vista pela outra;

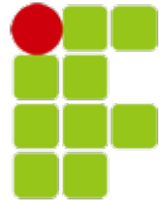
Classes: Relacionamentos



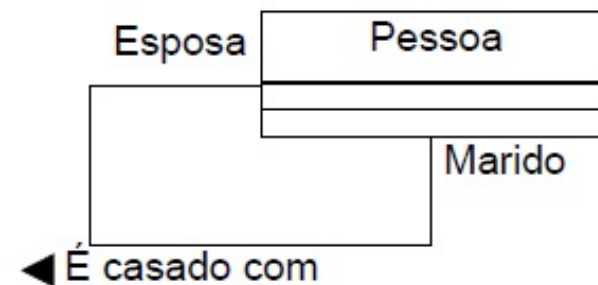
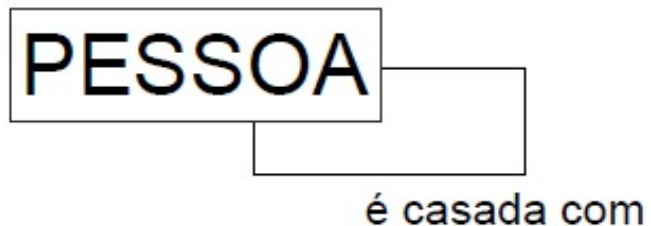
- Associações Normais



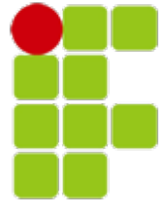
Classes: Relacionamentos



- Associações Binária Recursiva (reflexiva, autoassociação)
- Associação entre dois objetos, sendo que os mesmos pertencem a mesma classe;



Relacionamentos: Associação



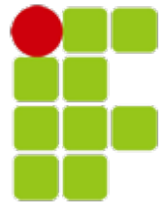
A multiplicidade especifica quantos objetos de uma classe relacionam-se a um único objeto de classe associada;

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)

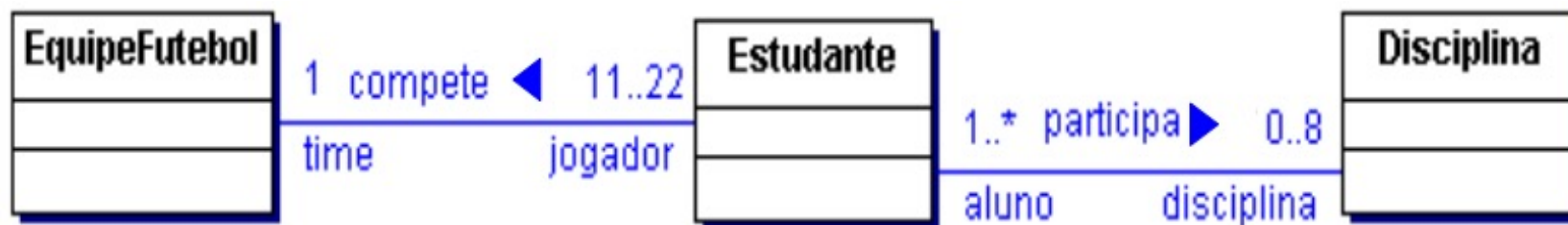


Classes: Relacionamentos

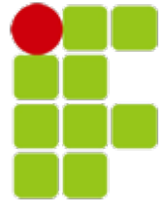


Relacionamentos: **Associação**

- 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

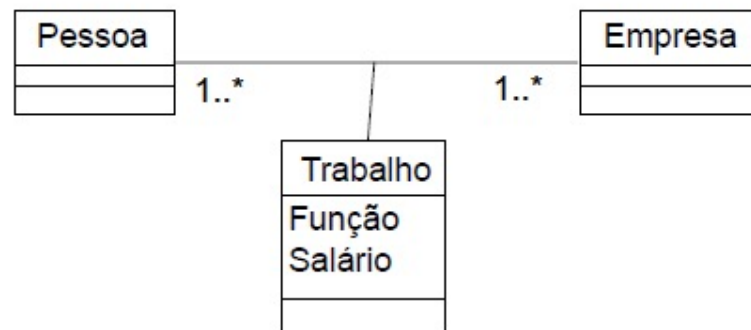


Classes: Relacionamentos

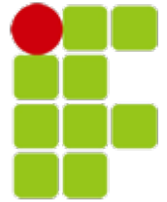


Classe Associativa

- Uma classe associativa é uma associação com propriedades de classe;
- Este tipo de classe surge quando duas ou mais classes estão associadas e é necessário manter informações sobre a associação;

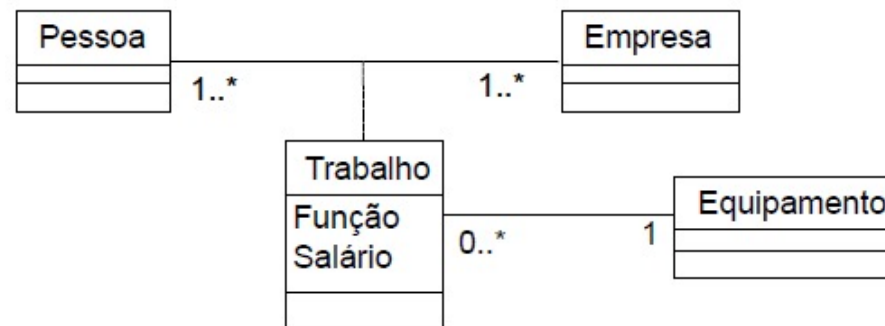


Classes: Relacionamentos

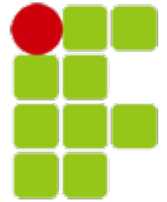


Classe Associativa

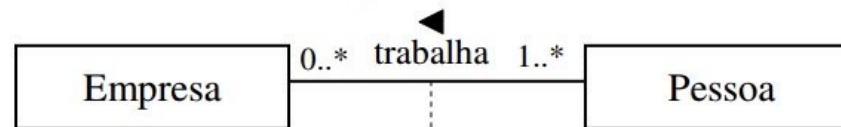
- Não se deve nomear a linha de associação de uma classe associativa;
- Somente o nome escolhido para a classe associativa deve ser suficiente para expressar o significado desejado;
- Uma classe associativa pode participar de outros relacionamentos;



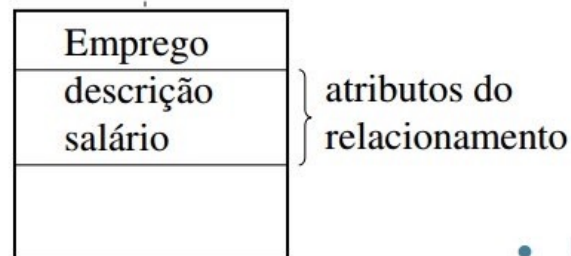
Classes: Relacionamentos



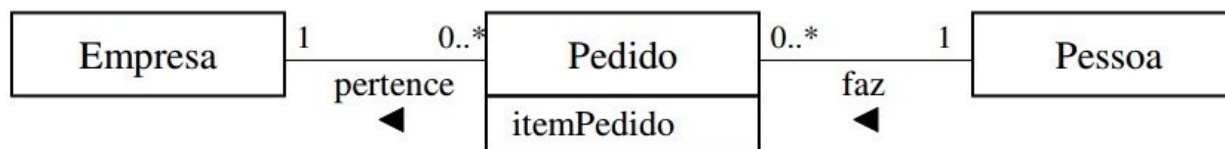
- Classe de associação



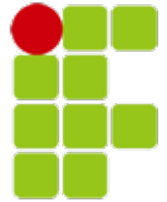
- Não existe uma pessoa com dois empregos na mesma empresa



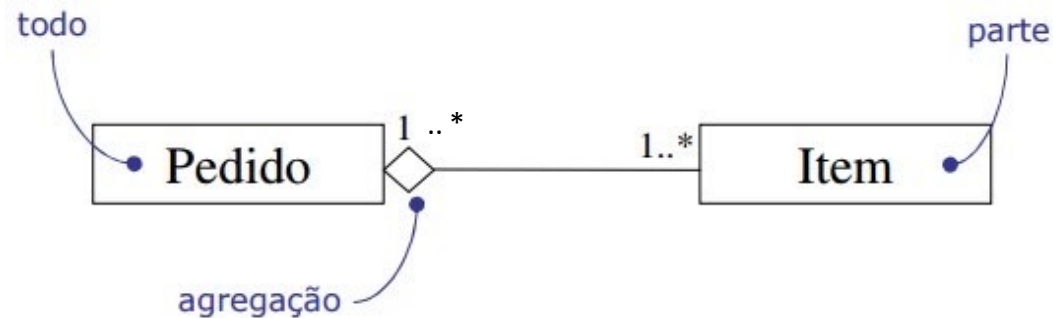
- Uma pessoa pode fazer mais de um pedido na mesma empresa



Relacionamentos: Agregação

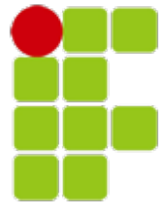


- É um tipo especial de associação
- Indica que uma das classes do relacionamento é uma parte de outra classe – relação *todo-parte*;



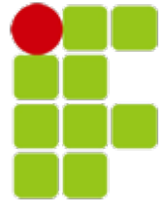
- um objeto “parte” pode fazer parte de vários objetos “todo”

Relacionamentos: Agregação

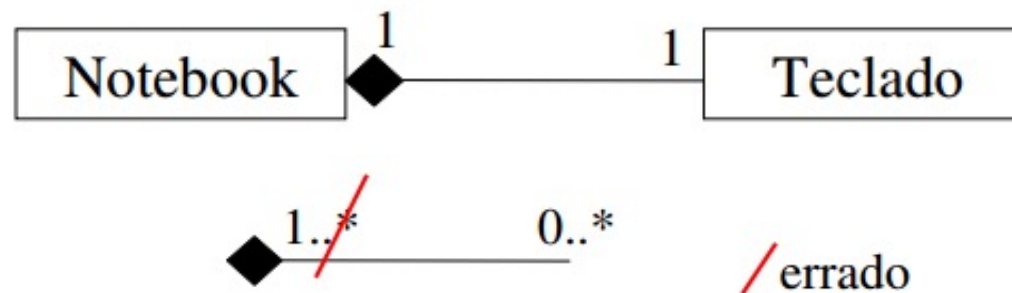


- A diferença entre associação e agregação é puramente semântica - em uma agregação um objeto está contido no outro;
- Onde se puder utilizar uma agregação, pode-se usar também uma associação;
- Palavras Chave: “*consiste em*”, “*é parte de*”;
- Graficamente representada por uma linha conectando as classes relacionadas, com um diamante (losango) branco perto da classe que representa o todo.

Relacionamentos: Composição

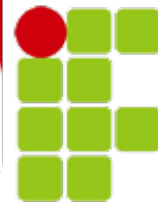


- É uma variante semanticamente mais “forte” da agregação.
- Os objetos “parte” só podem pertencer a um único objeto “todo” e têm o seu tempo de vida coincidente com o dele.

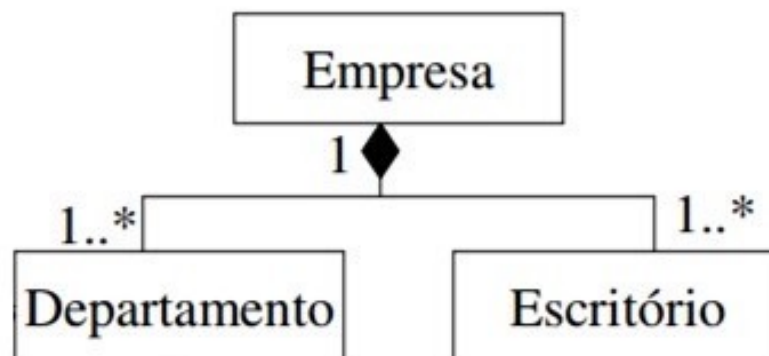


- Quando o “todo” *morre* todas as suas “partes” também *morrem*

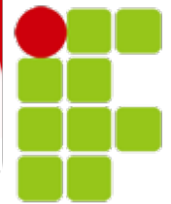
Relacionamentos: Composição



Ex:



Agregação e Composição



■ Agregação

- A multiplicidade do “todo” é maior que 1
- As partes não morrem obrigatoriamente com o todo
- Uma parte pode estar em mais de um “todo”
- As partes são semelhantes

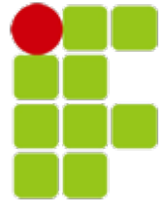


■ Composição

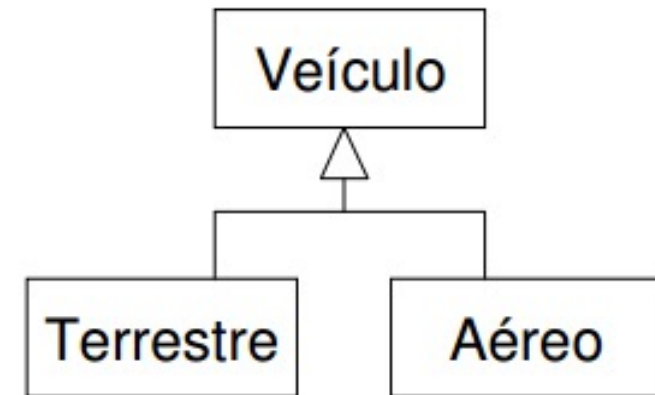
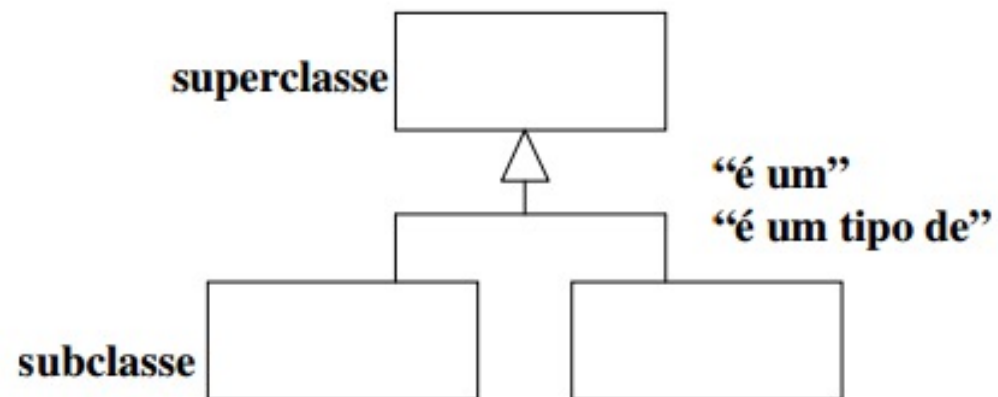
- A multiplicidade do “todo” é sempre 1
- As partes vivem e morrem com o todo
- Uma parte não pode estar em mais de um “todo”
- Em geral, as partes não são semelhantes



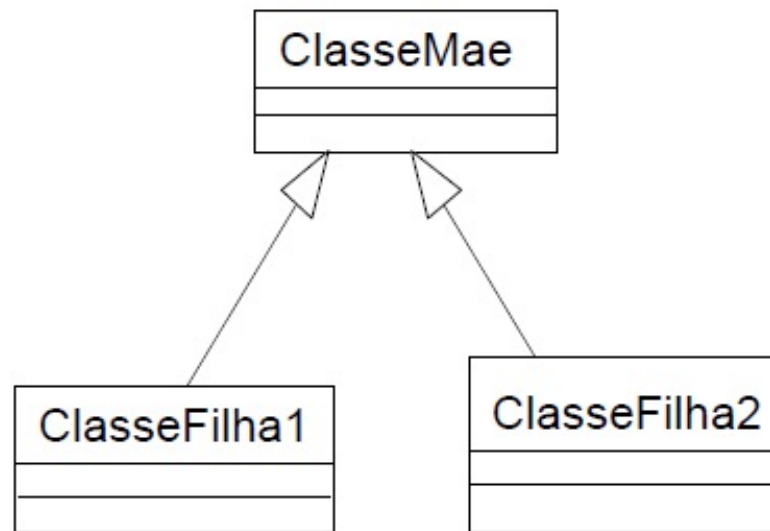
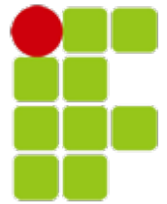
Relacionamentos: Generalização



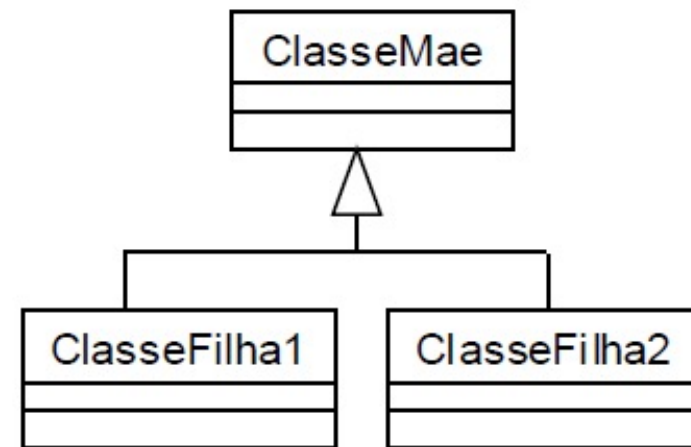
- É um relacionamento entre itens gerais (superclasses) e itens mais específicos (subclasses)



Relacionamentos: Generalização

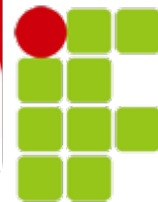


Estilo com setas separadas

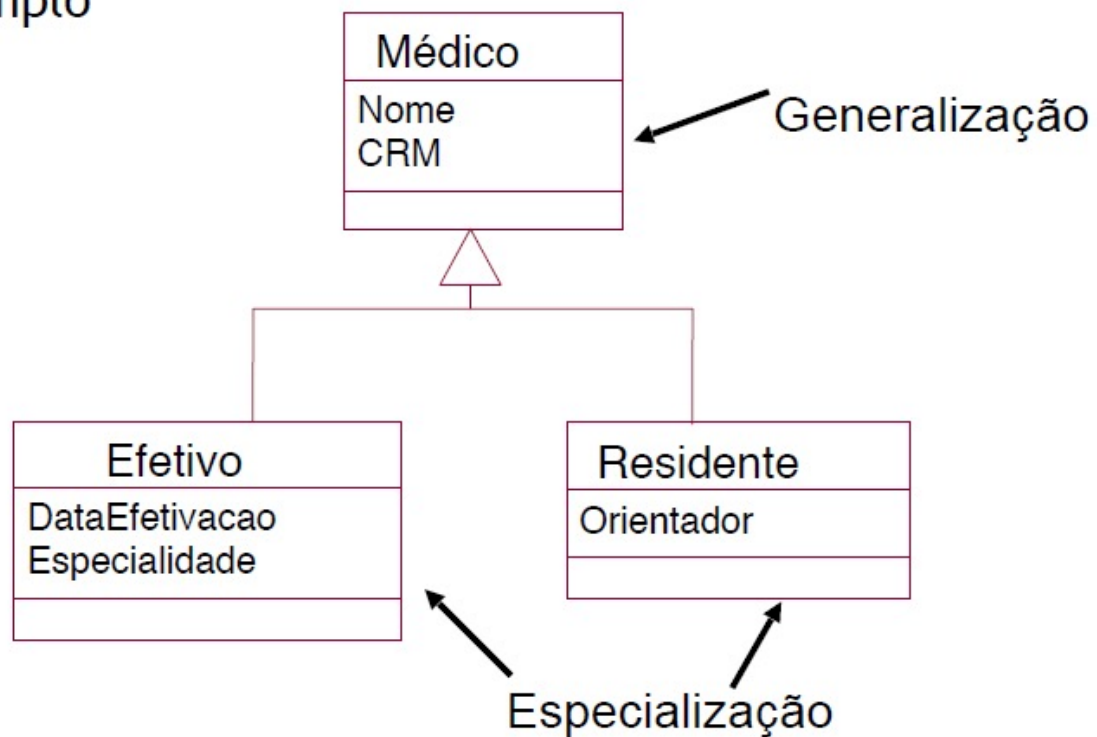


Estilo com setas Agrupadas

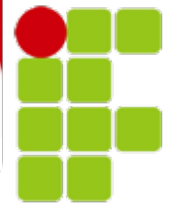
Relacionamentos: Generalização



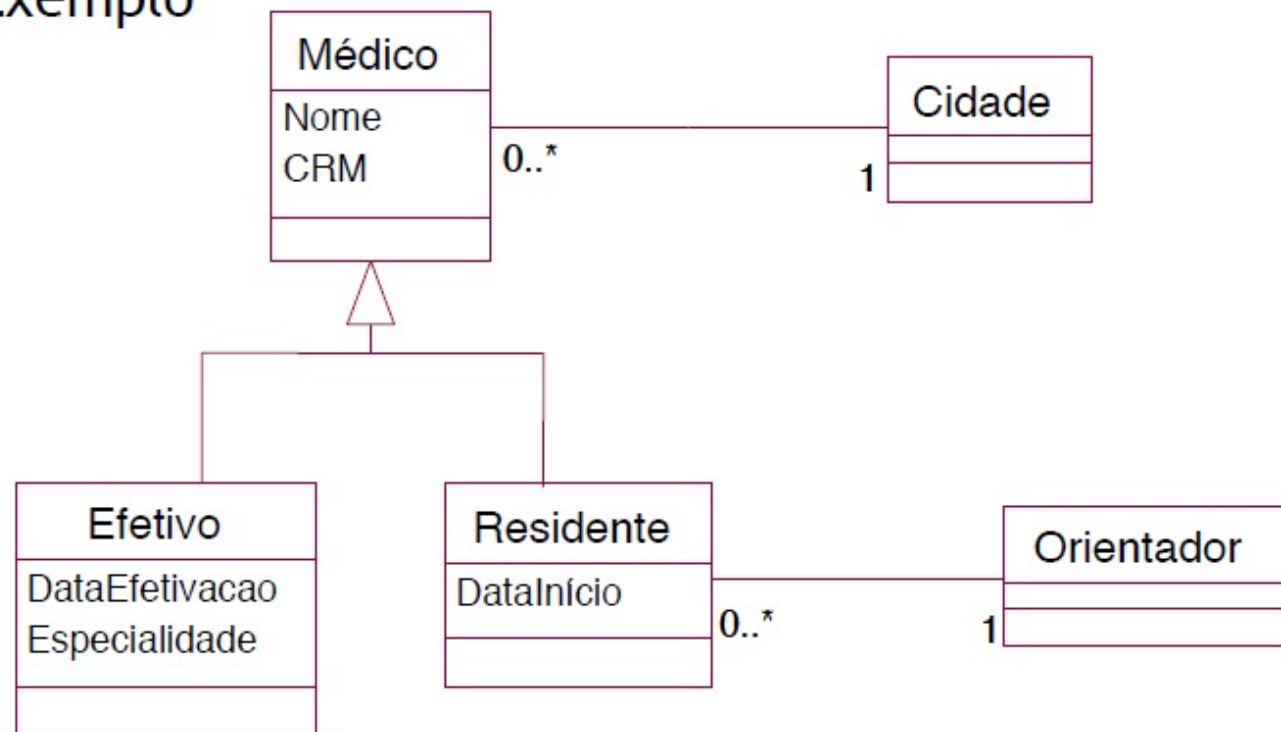
- Exemplo



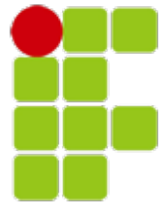
Relacionamentos: Generalização



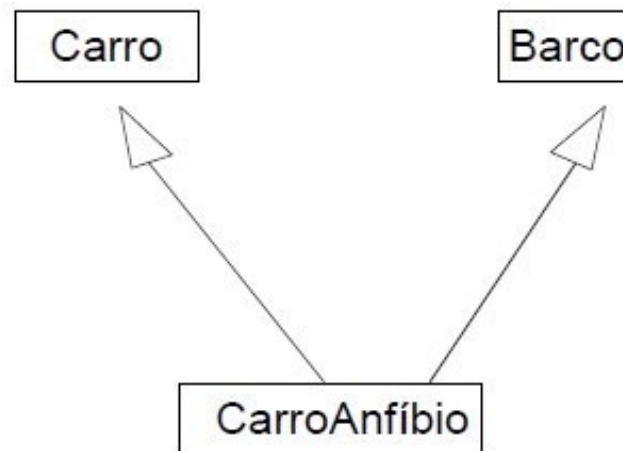
- Exemplo



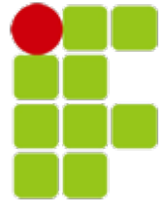
Relacionamentos: Generalização



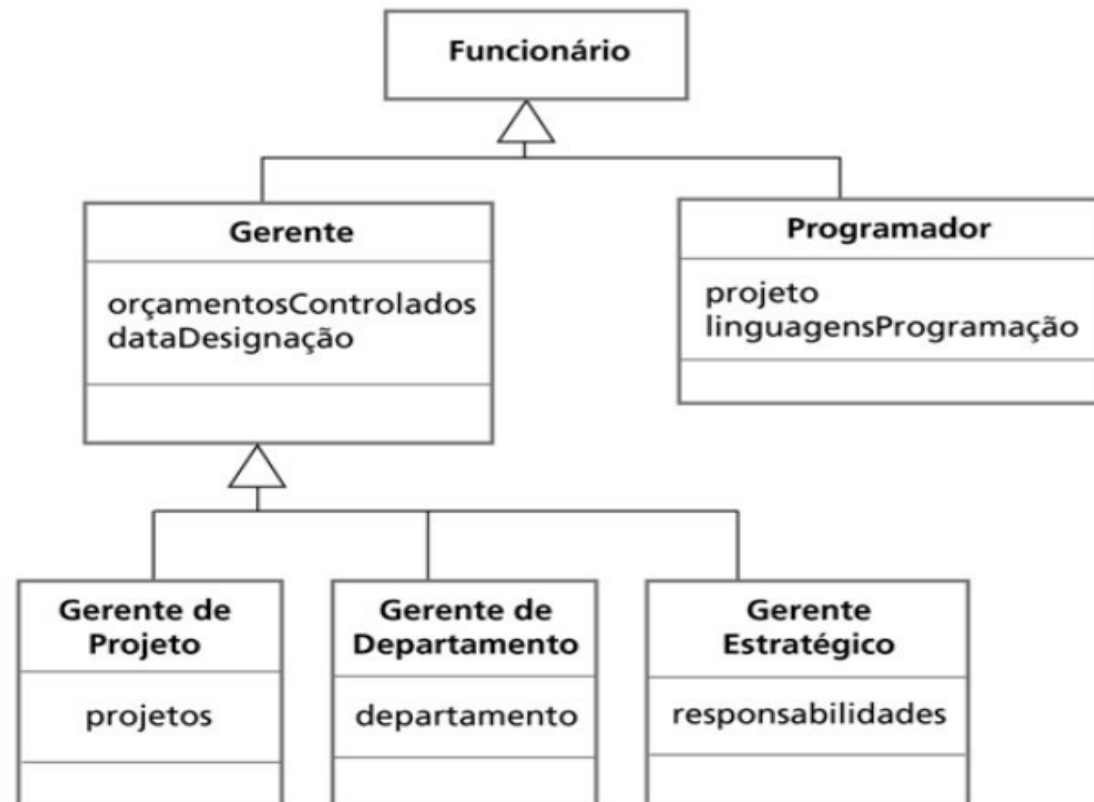
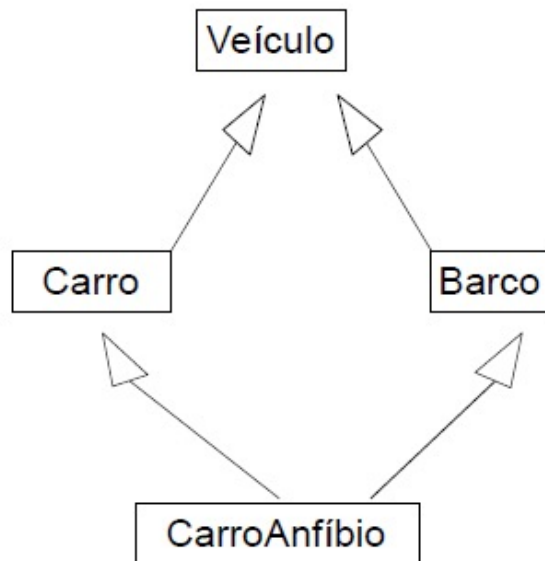
- Uma classe pode ter mais de uma superclasse (herança múltipla);



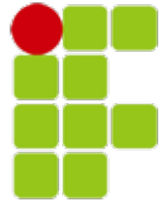
Relacionamentos: Generalização



Níveis de especialização



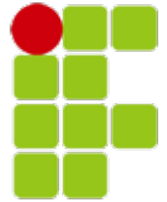
Exercício



Construa um diagrama de classes para os seguintes objetos. Se quiser, pode incluir novas classes. Utilize herança, agregação e demais conceitos.

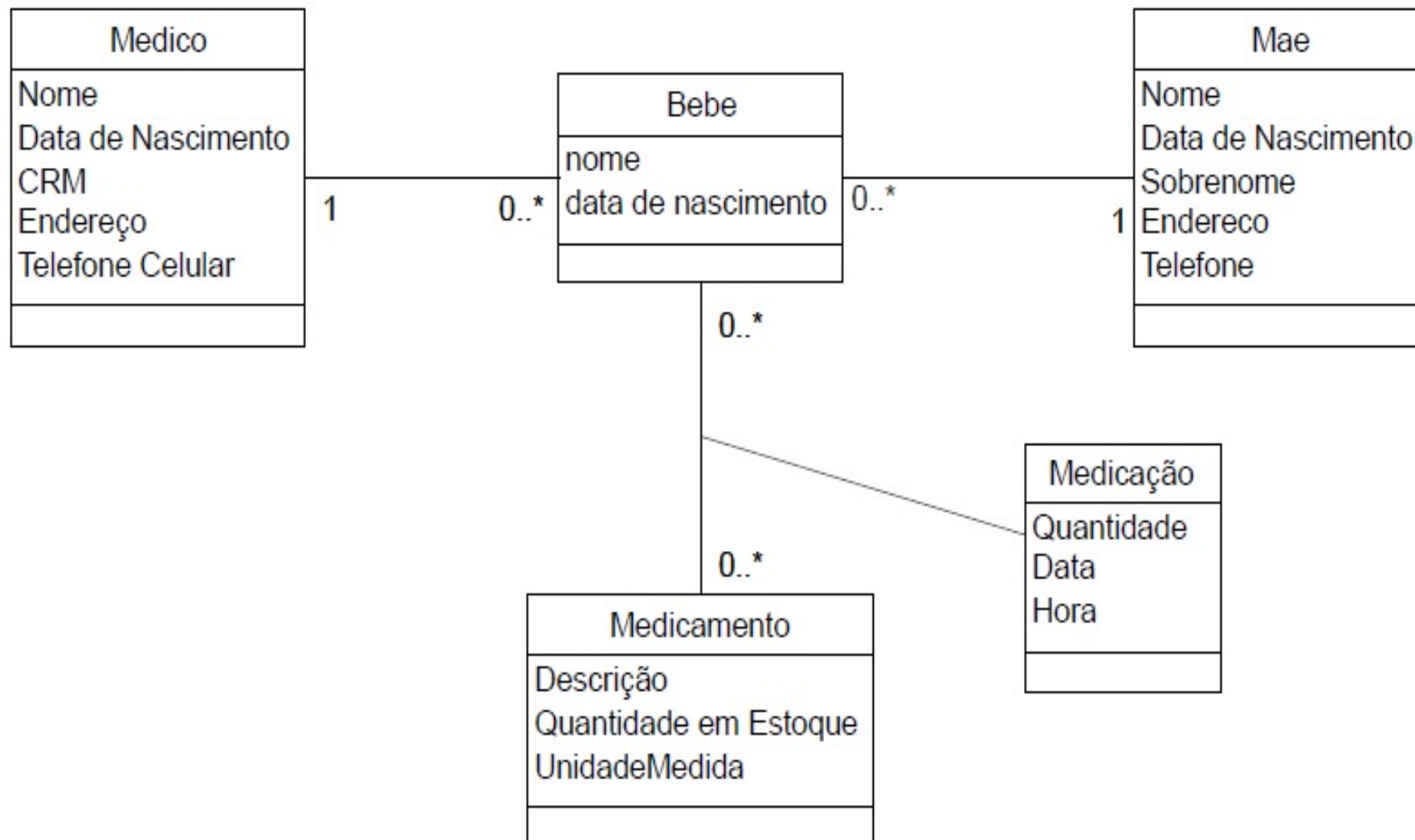
- Escola, computador, playground, livro, porta, laboratório, salas, biblioteca, régua, alunos, giz, balanço, revista, janela, sala de aula, professor, carteira, sala dos professores, cadeira, teclado, monitor, escorregador, bibliotecário, quadro-negro, secretária, boletim
- Casa, cozinha, sofá, banheiro, panela, armário, lâmpada, geladeira, janela, sala, cama, pia, sala de estar, fogão, tampa da panela, sistema de alarme, abajur, sala de jantar, porta, mesa, sistema de incêndio, pão, leite, sabonete, cabo da panela

Exemplo Berçário

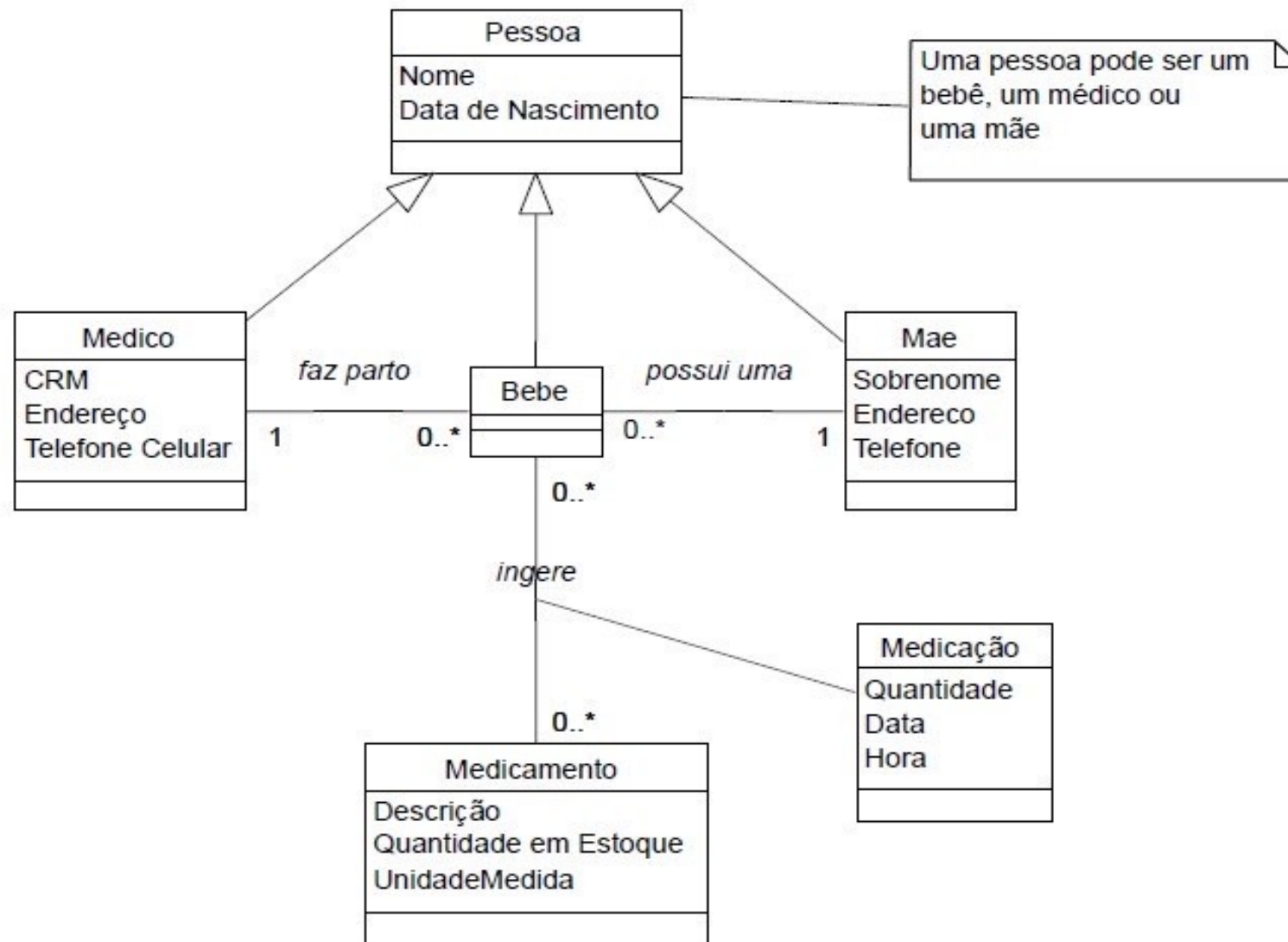
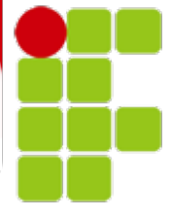


- Um berçário deseja controlar suas tarefas. Para isso deseja manter um cadastro para os bebês, constando **nome, médico que fez seu parto, sua mãe e data de nascimento**. Para os médicos é necessário saber: **o nome, data de nascimento, CRM, endereço e telefone celular**. Para as mães dos bebês é necessário manter informações como: **nome, sobrenome, data de nascimento, endereço e telefone**. Ainda, é necessário manter um controle dos medicamentos ingeridos pelos bebês no berçário. Sobre os medicamentos é necessário manter: **descrição, quantidade em estoque e unidade de medida**. Um bebê pode tomar vários medicamentos, assim como um medicamento pode ser dado para vários bebês. Quando um bebê toma uma medicação ainda é importante saber a **quantidade, o dia e a hora do medicamento**.

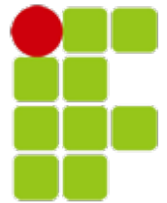
Exemplo Berçário



Exemplo Berçário



Exemplo Condomínio



Uma entrevista com o gerente da administradora resultou nas seguintes informações:

A administradora administra condomínios formados por unidades condominiais (apartamentos, casas, salas etc).

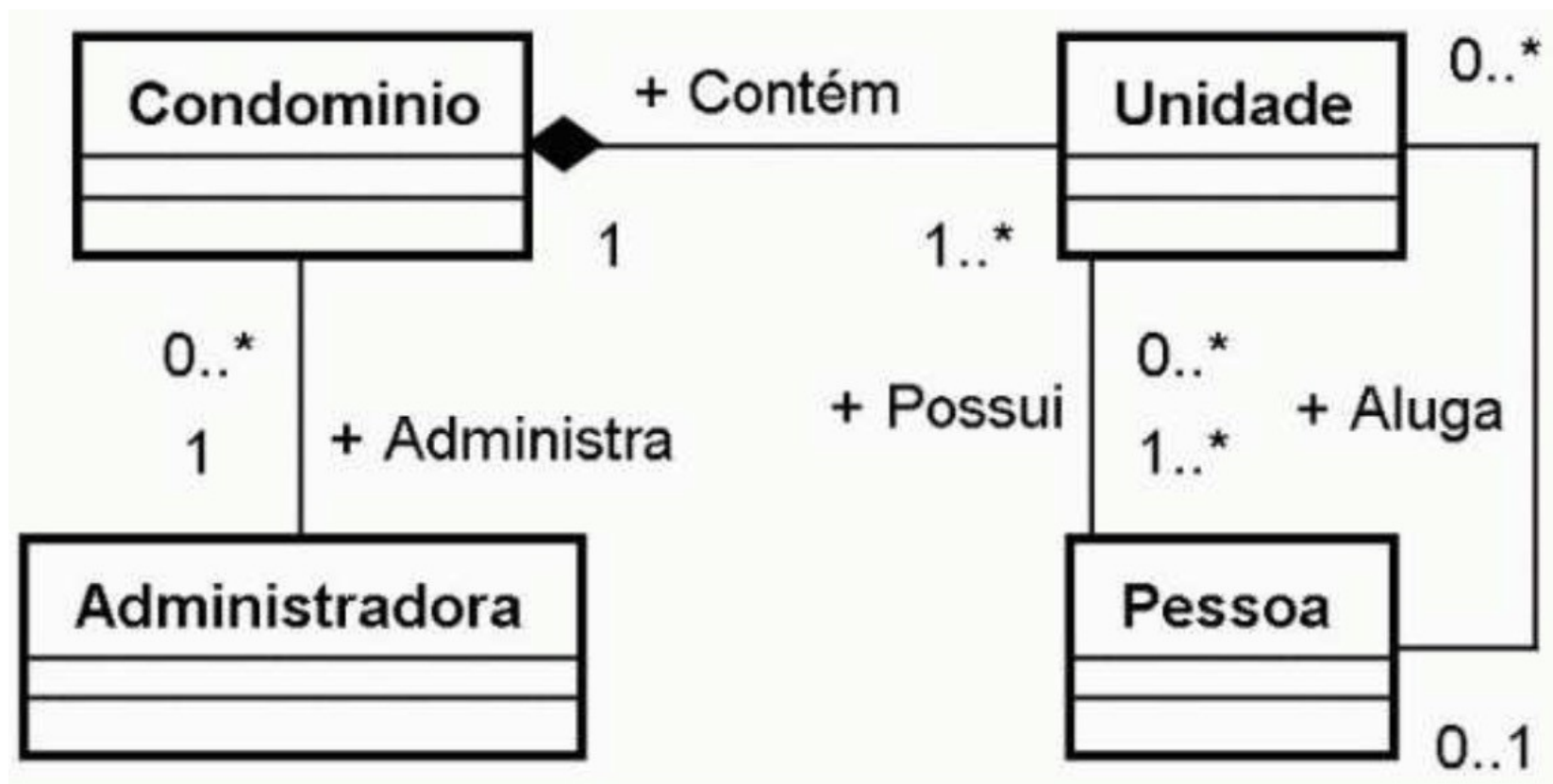
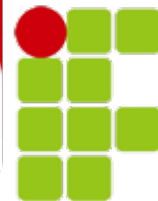
Cada unidade condominial é de propriedade de uma ou mais pessoas.

Uma pessoa pode possuir diversas unidades.

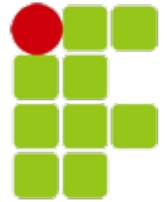
Cada unidade pode estar alugada para no máximo uma pessoa.

Uma pessoa pode alugar diversas unidades.

Exemplo Condomínio

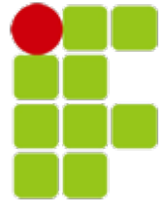


Exemplo Sistema de Controle Acadêmico



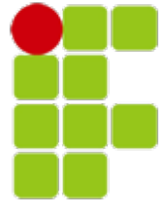
- **RF01** – o sistema deve permitir à secretaria cadastrar cursos contendo código, descrição e coordenador.
- **RF02** – o sistema deve permitir à secretaria cadastrar disciplinas de cursos, contendo código, descrição, carga horária, ementa, bibliografia e pré-requisitos.
- **RF03** – o sistema deve permitir à secretaria cadastrar alunos, contendo matrícula, nome, endereço, telefone e curso para o qual foi aprovado.
- **RF04** – o sistema deve permitir ao departamento de recursos humanos (RH) cadastrar professores, contendo nome, endereço, telefone e titulação máxima (graduação, especialização, mestrado, doutorado) e cursos que esteja vinculado.

Exemplo Sistema de Controle Acadêmico



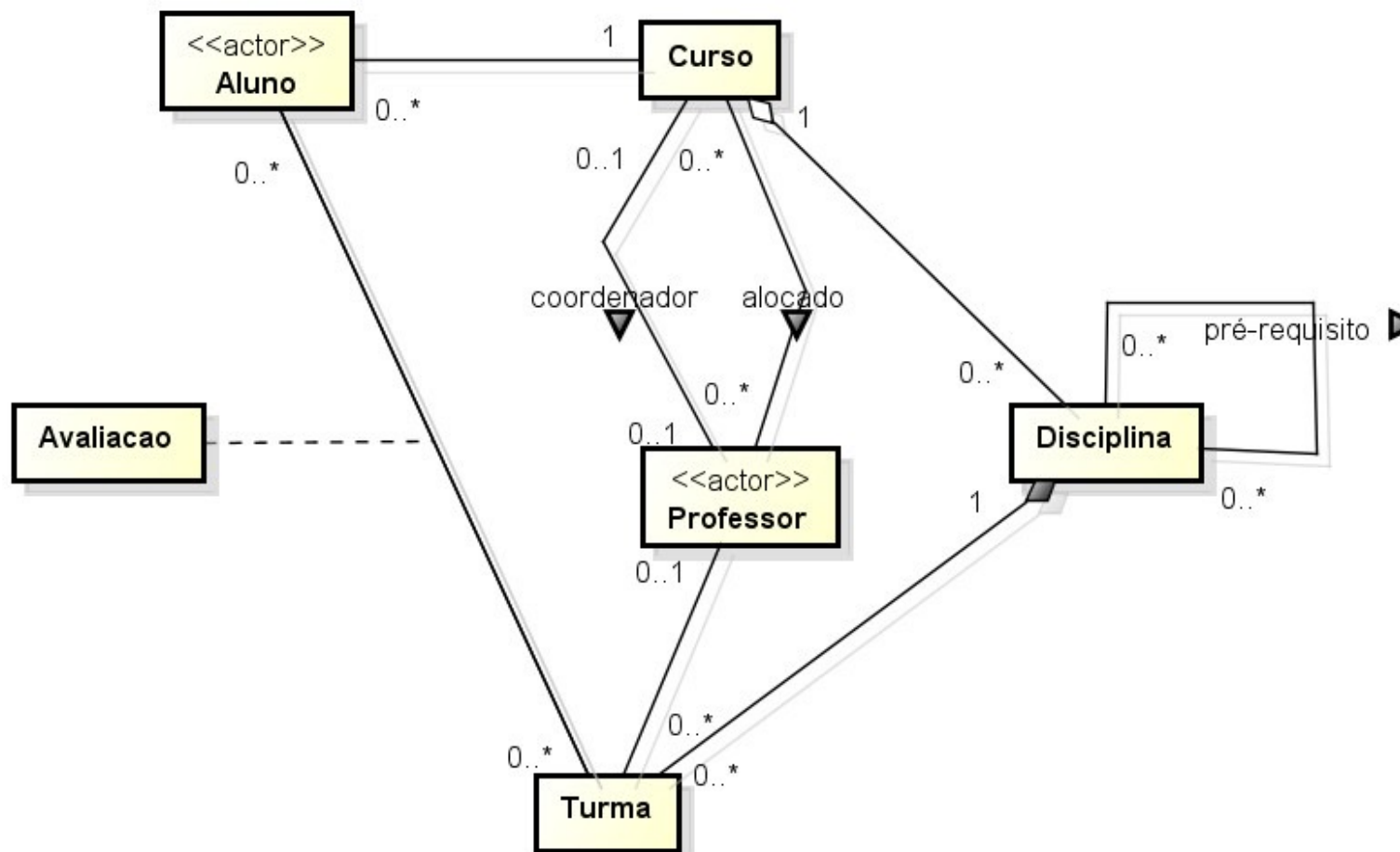
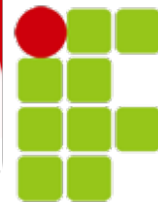
- **RF05** – o sistema deve permitir à secretaria abrir turmas de disciplinas de cursos, informando ano e semestre, dias da semana e horários de realização.
- **RF06** – o sistema deve permitir aos coordenadores de curso alocar professores a determinadas turmas.
- **RF07** – o sistema deve permitir à secretaria matricular alunos em turmas.
- **RF08** – o sistema deve permitir aos professores lançar avaliações (duas notas parciais, nota da prova final e frequência) dos alunos das turmas que estejam sob sua responsabilidade.
- **RF09** – o sistema deve permitir aos alunos consultar suas avaliações.

Exemplo Sistema de Controle Acadêmico

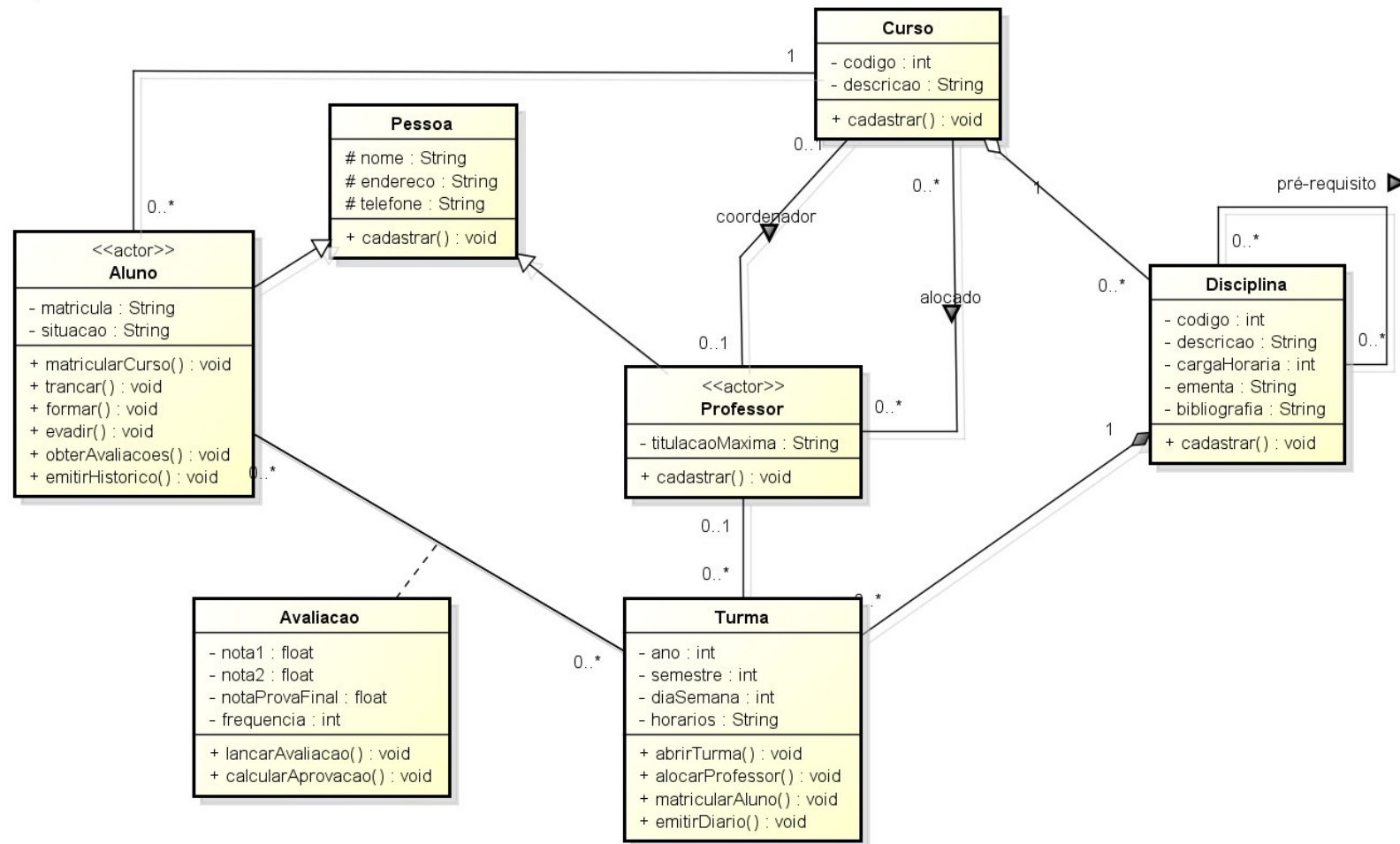
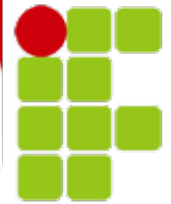


- **RF10** – o sistema deve permitir à secretaria emitir diários de classe das turmas.
- **RF11** – o sistema deve permitir à secretaria emitir históricos escolares dos alunos.
- **RF12** – o sistema deve efetuar o cálculo da aprovação de alunos em turmas, sendo que, para ser aprovado, deve-se ter frequência mínima de 75%. Além disso, para aprovação sem prova final, a média das notas parciais deve ser maior ou igual a 70. para reprovação direta, esta média deve ser menor que 30. médias entre 30 (inclusive) e 70 (exclusive) colocam o aluno em prova final. Se a média da prova final com a média anterior for menor que 50, o aluno está reprovado, caso contrário, aprovado.
- **RF13** – o sistema deve controlar a situação de um aluno, podendo estar matriculado, trancado, formado ou evadido.

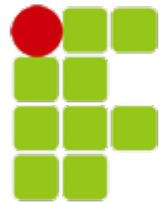
Exemplo Sistema de Controle Acadêmico



Exemplo Sistema de Controle Acadêmico



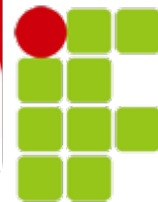
Exemplo Sistema de Gestão de Dados



Produtos Virtuais

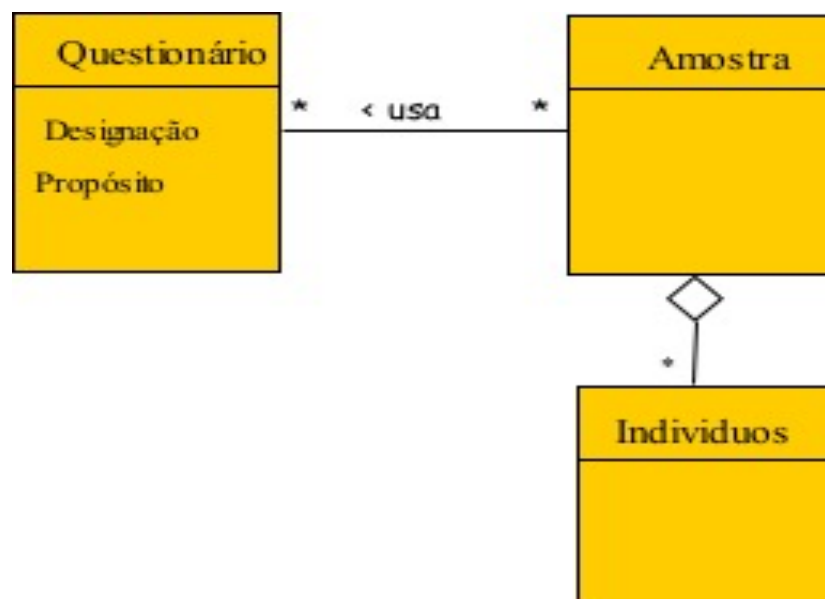
A empresa Produtos Virtuais, Lda. está a implementar um novo sistema de apoio aos estudos de mercado. Os técnicos de marketing produzem questionários, tendo a possibilidade de ao mesmo tempo registar as questões associados a cada estudo de mercado. Por sua vez, os estudos de mercados vão ser respondidos por indivíduos que compõem uma amostra. Os indivíduos respondem aos questionários on-line através da Internet ou então em papel. Neste último caso, as respostas são de seguida introduzidas no sistema por um técnico de marketing. Interessa registar informação relativa ao questionário, nomeadamente a designação e o propósito. O questionário é aplicado a uma ou várias amostras, compostas por indivíduos. Cada questionário é constituído por uma série de questões. Interessa armazenar o texto a perguntar ao indivíduo que responde ao questionário. As questões podem ser abertas ou fechadas. Se forem fechadas devem indicar a escala (limite inferior e limite superior). Relativamente ao indivíduo interessa armazenar a localidade, morada, sexo, idade, habilitações académicas, rendimento anual, profissão. Cada indivíduo pode responder a vários questionários e fazer parte de várias amostras. O indivíduo responde a questões, devendo ser introduzido um número no caso de ser uma questão fechada, ou o texto da resposta no caso de ser uma pergunta aberta

Exemplo Sistema de Gestão de Dados

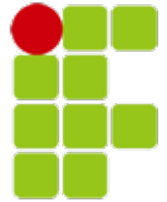


Solução possível:

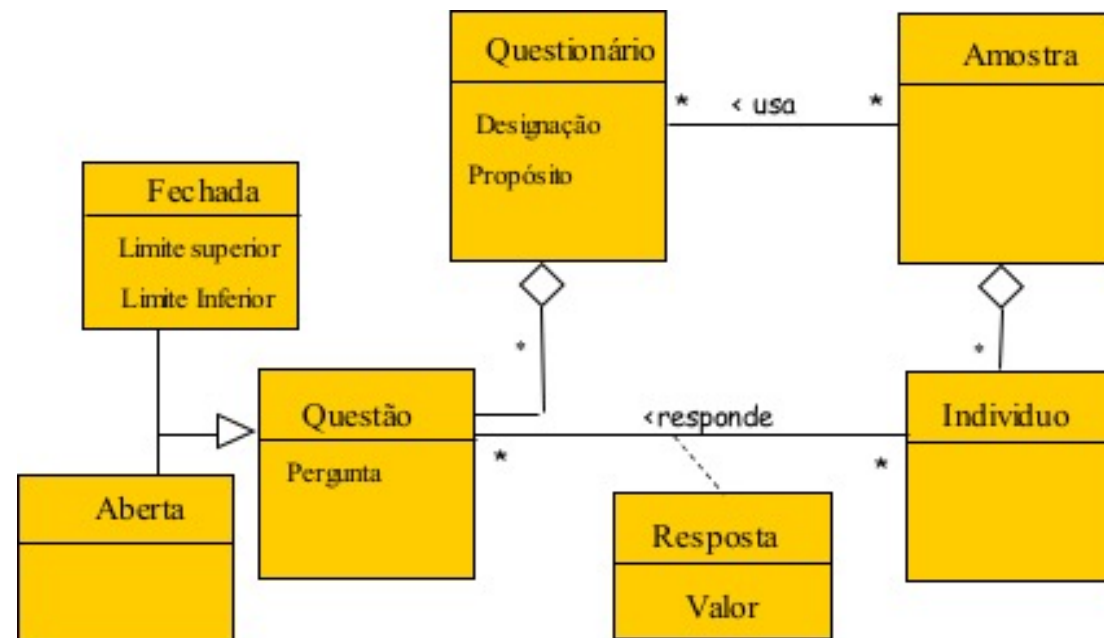
Interessa registar informação relativa ao questionário, nomeadamente a designação e o propósito. O questionário é aplicado a uma ou várias amostras, compostas por indivíduos



Exemplo Sistema de Gestão de Dados



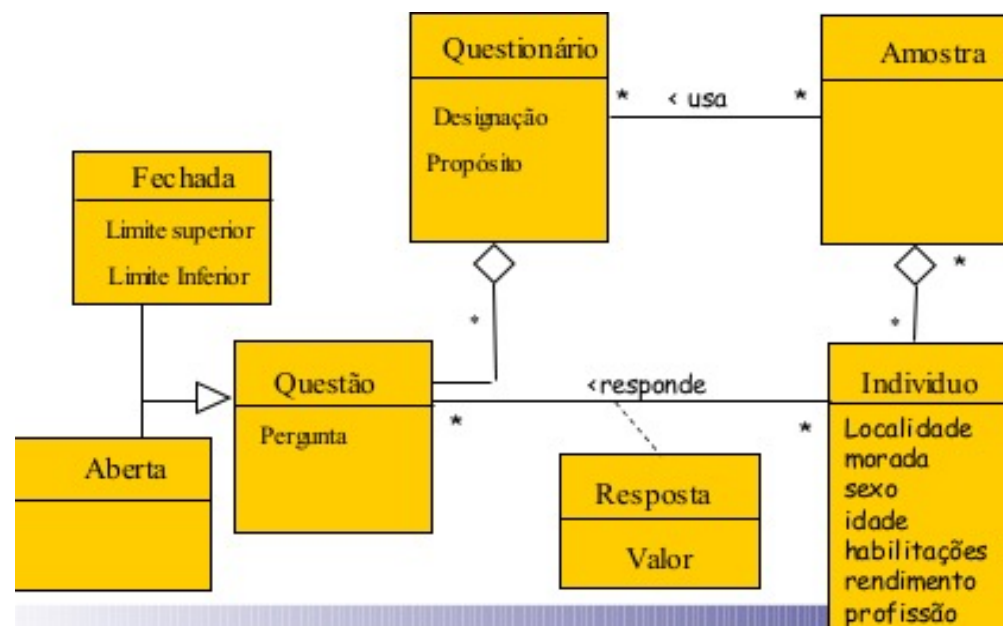
Cada questionário é constituído por uma série de questões. Interessa armazenar o texto a perguntar ao indivíduo que responde ao questionário. As questões podem ser abertas ou fechadas. Se forem fechadas devem indicar a escala (limite inferior e limite superior).



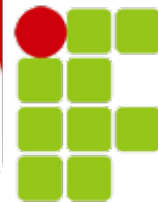
Exemplo Sistema de Gestão de Dados



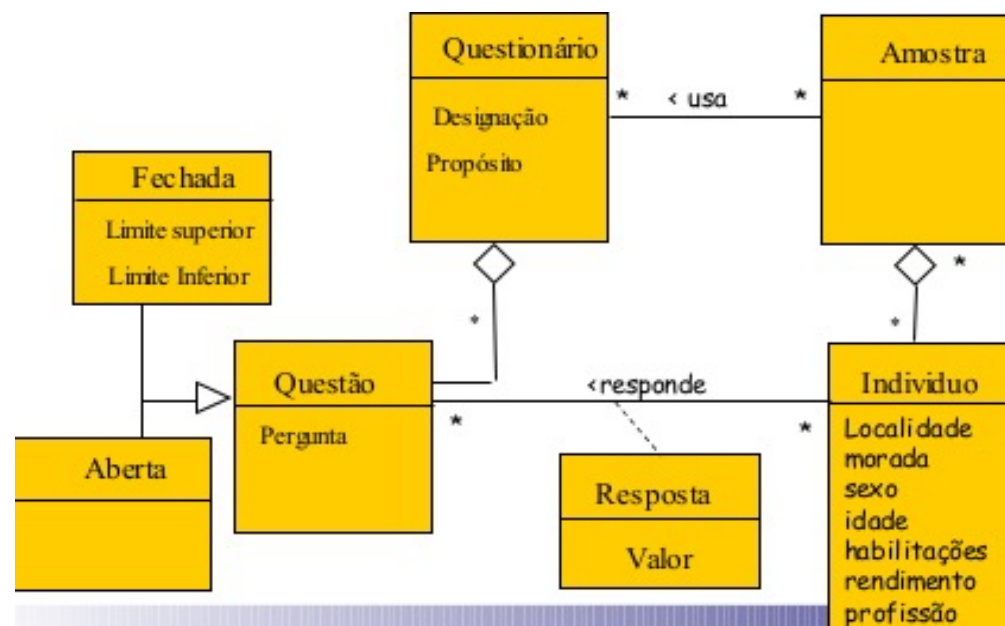
O indivíduo responde a questões, devendo ser introduzido um número no caso de ser uma questão fechada, ou o texto da resposta no caso de ser uma pergunta aberta



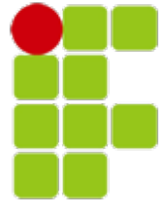
Exemplo Sistema de Gestão de Dados



O indivíduo responde a questões, devendo ser introduzido um número no caso de ser uma questão fechada, ou o texto da resposta no caso de ser uma pergunta aberta



Exemplo Sistema de Gestão de Dados



Revista On-Line

Pretende-se desenvolver um sistema que permite apoiar a gestão de revistas On-line.

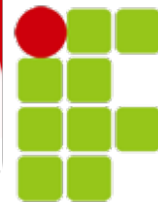
O sistema será utilizado por redatores, ilustrados e fotógrafos, que introduzem diretamente os seus conteúdos no sistema

O sistema é ainda gerido por um administrador de sistemas que cria os perfis de utilizador, atribuindo o login e password. Cada revista tem um título e é constituída por diversos números. Cada número tem uma data e ano. Cada número é constituído por artigos. O artigo tem um título, sendo constituído por caixas, esta tem uma posição no número, indicado pelas páginas, coordenadas e dimensão. As caixas podem ainda ser imagens ou de texto.

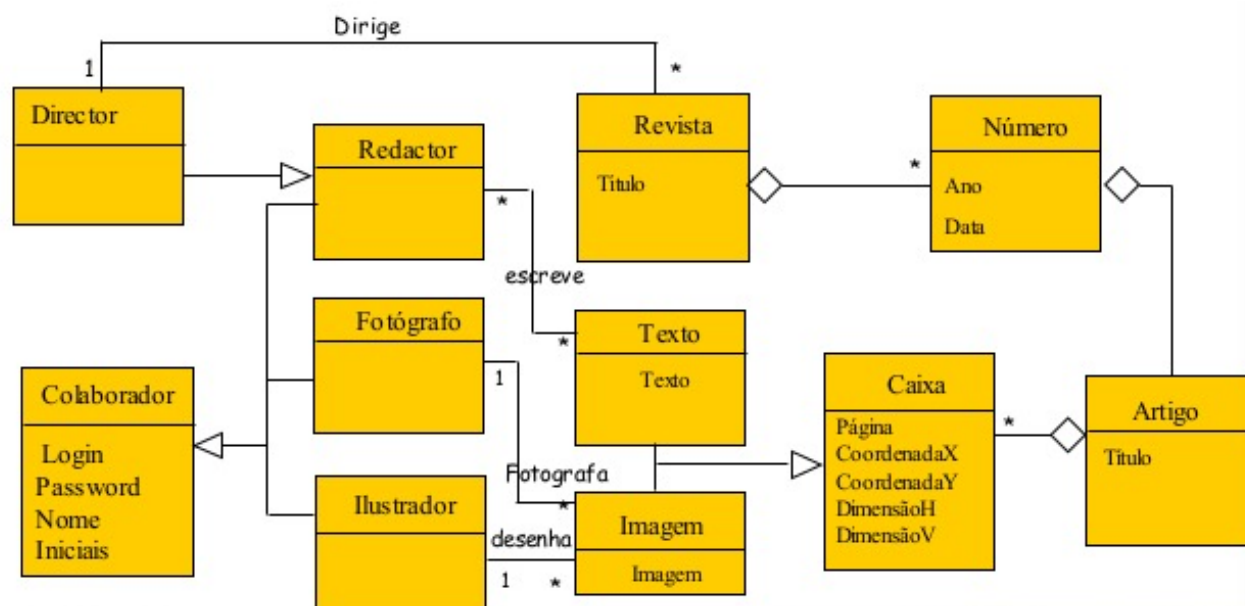
As imagens são da autoria de ilustradores ou de fotógrafos, enquanto que os textos são de autoria de redator. Todos eles são colaboradores, sendo necessário guardar o nome, iniciais.

Uma revista tem ainda um diretor. Este é um redator.

Exemplo Sistema de Gestão de Dados



Possível solução



Dúvidas?

