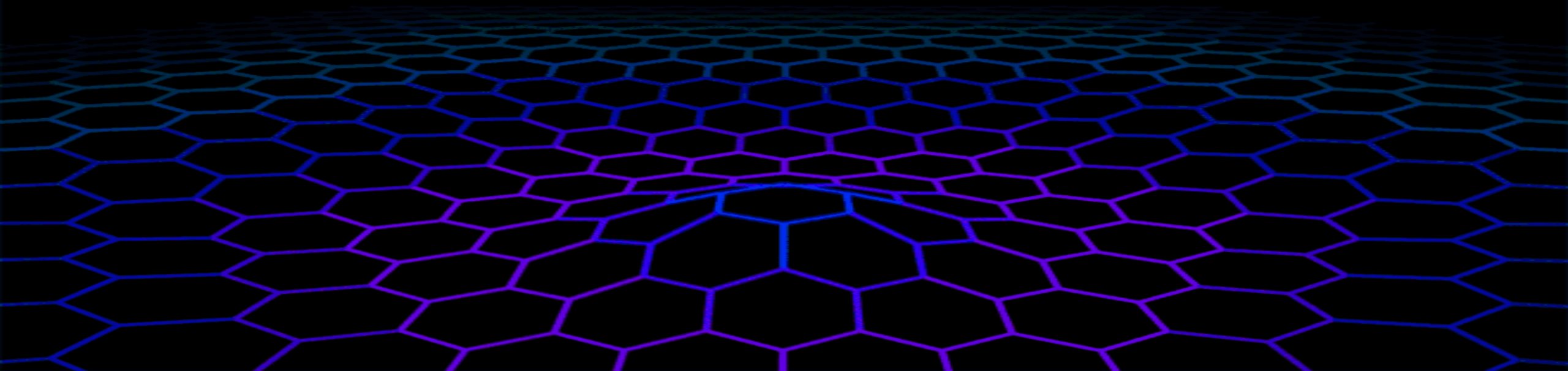


Engenharia de Software 1



Análise Orientada a Objetos

Análise Orientada a Objetos



No passado

O desenvolvimento de software adotava uma perspectiva de algoritmo;

Nessa visão, o principal bloco de construção do software é o procedimento (função).

Resultado: tínhamos sistemas instáveis, pois à medida que os requisitos se modificavam e o sistema crescia, ficava difícil fazer a manutenção dos mesmos;

Orientação a Objetos

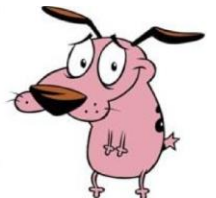
Cachorro
-tamanho: int -raca: String
*latir()



Introdução

- O paradigma de orientação a objetos foi criado por volta da década de 60;
- Apesar de antiga, ela não era muito utilizada;
- O pulo do gato foi a evolução do hardware e o aumento da complexidade dos sistemas;
- Desta forma era necessário um novo paradigma de desenvolvimento;

Cachorro
-tamanho: int
-raca: String
*latir()



Dúvida

O que nós vemos no mundo real?

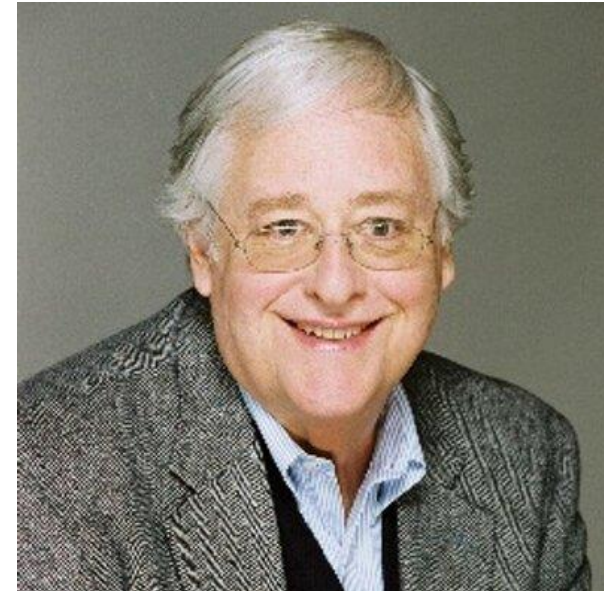
- O mundo real é composto por objetos;
- Cada objeto tem propriedades e comportamentos;

Por que não desenvolver programas que simulem no computador os objetos do mundo real com suas propriedades e comportamentos?



Edward Yourdon disse:

“Um sistema construído usando um método Orientado a Objetos é aquele cujos componentes são partes encapsuladas de dados e funções, que podem herdar atributos e comportamentos de outros componentes da mesma natureza, e cujos componentes comunicam-se entre si por meio de mensagens.”



O mundo hoje

O método orientado a objetos já mostrou o seu valor para a construção de sistemas em todos os tipos de problemas, abrangendo todos os graus de tamanho e complexidade;

Além disso, hoje em dia as principais linguagens de desenvolvimento e as principais ferramentas utilizadas são orientadas a objetos;

Análise Orientada a Objetos



Por onde começar?

Basicamente, modelar um sistema orientado a objetos não tem mistério algum. Sendo ou não orientado a objetos a ideia é levantar o quais as necessidades do sistema;

Porém, existem algumas linguagens visuais que foram criadas especificamente para modelarmos sistema orientados a objetos;



UML - Unified Modeling Language (Linguagem de Modelagem Unificada)

- A visualização;
- A especificação;
- A construção;
- E a documentação...

de sistemas orientados a objetos
é exatamente o objetivo da UML



Introdução

A UML (Unified Modeling Language) é uma linguagem visual utilizada para modelar sistemas computacionais por meio do paradigma de Orientação a Objetos;

Essa linguagem se tornou, nos últimos anos, a linguagem-padrão de modelagem de software adotada internacionalmente pela indústria de Engenharia de Software;



Introdução

Definir as características do software, como:

- Comportamento
- Estrutura lógica
- Processos
- Entre outros



O que é?

A UML **não** é uma linguagem de programação ela é uma linguagem de modelagem;

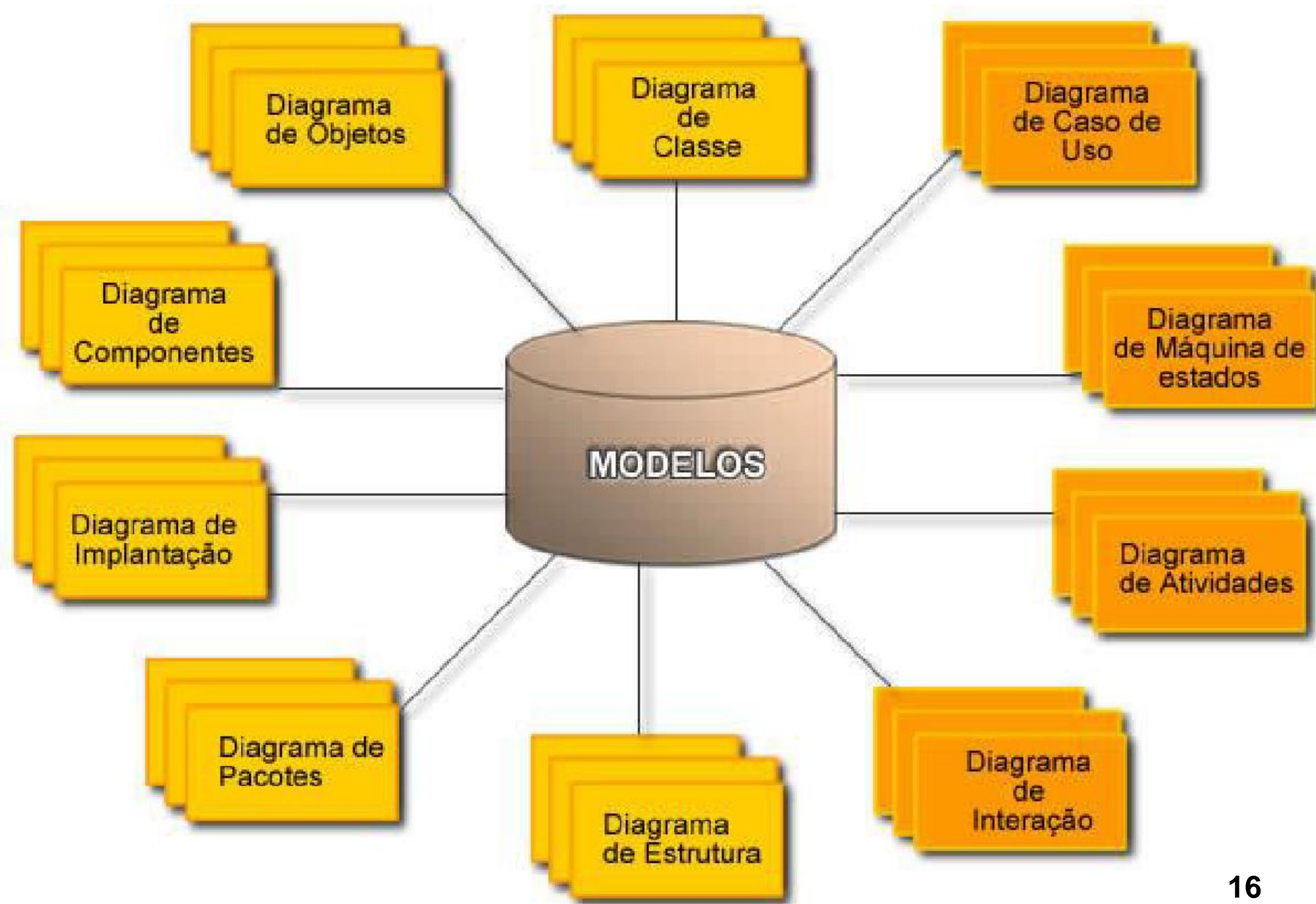
Podemos dividir os diagramas da UML em três grupos:

- Estruturais

- Comportamentais

- Interação





Diagramas estruturais

Diagrama de classes

Diagrama de objetos

Diagrama de componentes

Diagrama de implantação

Diagrama de pacotes

Diagrama de estrutura



Diagramas comportamentais

Diagrama de casos de uso

Diagrama de máquina de estados

Diagrama de atividades



Diagramas de interação

Diagrama de sequência

Diagrama de interatividade

Diagrama de colaboração

Diagrama de tempo



Como usá-los



Utilizando a UML

Geralmente para utilizarmos a UML começamos com os modelos comportamentais (ex. diagrama de casos de uso);

Após, criamos alguns modelos estruturais (ex. diagrama de classes);

E, por fim, criamos alguns modelos de interação (ex. diagrama de sequência).



Como podemos criar esses diagramas?



Ferramentas



Utilizando a UML na prática

A melhor forma de utilizar a UML é utilizando uma ferramenta para nos auxiliar;

Essas ferramentas geralmente são chamadas de Ferramentas CASE (Engenharia de Software Auxiliada por Computador);

São aplicativos que auxiliam os profissionais envolvidos na tarefa de produzir sistemas;

Utilizando a UML na prática

Existem muitas Ferramentas CASE para trabalharmos com a UML;

Vocês podem utilizar qualquer ferramenta: Astah, Lucidchart, Enterprise Architect, Microsoft Visio, Libre Office Draw, ...

Diagrama de Casos de Uso



Vamos pensar

Casos de uso podem servir para orientar como uma casa será arquitetada:



Objetivo dos casos de uso

O objetivo dos casos de uso é descrever a interação dos **atores** com o sistema;

Um **ator** é um item externo ao sistema (ex. cliente, fornecedor, aluno, etc.) que executa uma sequência de ações dentro do mesmo;

Uma vantagem importante da criação de casos de uso está no fato de criá-los sem especificar o modo como eles são implementados;



Dúvida

Como eu sei quais casos de uso meu sistema terá?

Através dos requisitos funcionais;

A grosso modo, cada requisito funcional é um caso de uso;

E quem são meus atores?

Os atores serão as entidades externas que irão interagir com o sistema;



Atores

São usuários e/ou outros meios externos que desenvolvem algum papel em relação ao sistema;

Os meios externos são hardwares e/ou softwares que, assim como os usuários, geram informações para o sistema ou necessitam de informações geradas a partir do sistema;



Na prática

Atores, muitas vezes, podem desempenhar mais de um papel no sistema;

Devemos pensar neles como **os papéis desempenhados** em vez de pensar como pessoas, cargos, máquinas;

Quais papéis um ator poderia desempenhar em um sistema?



Introdução

É o diagrama mais abstrato, flexível e informal da UML.

Normalmente, é utilizado no início da modelagem para identificar os requisitos do sistema.

Pode ser utilizado como base para criação de outros diagramas.

Usando uma linguagem simples, permite que qualquer pessoa compreenda o comportamento externo do sistema.



Introdução

Identifica os tipos de usuários que interagem com o sistema, os que eles assumem e as funções requisitadas;

Pode (e deve ser) apresentado durante reuniões iniciais com os clientes porque pode auxiliar na identificação de possíveis falhas;

Descreve os requerimentos funcionais do sistema de maneira consensual entre usuários e desenvolvedores de sistema.



Voltando...

Que atores farão parte de um sistema que simule uma casa e quais casos de uso teremos?



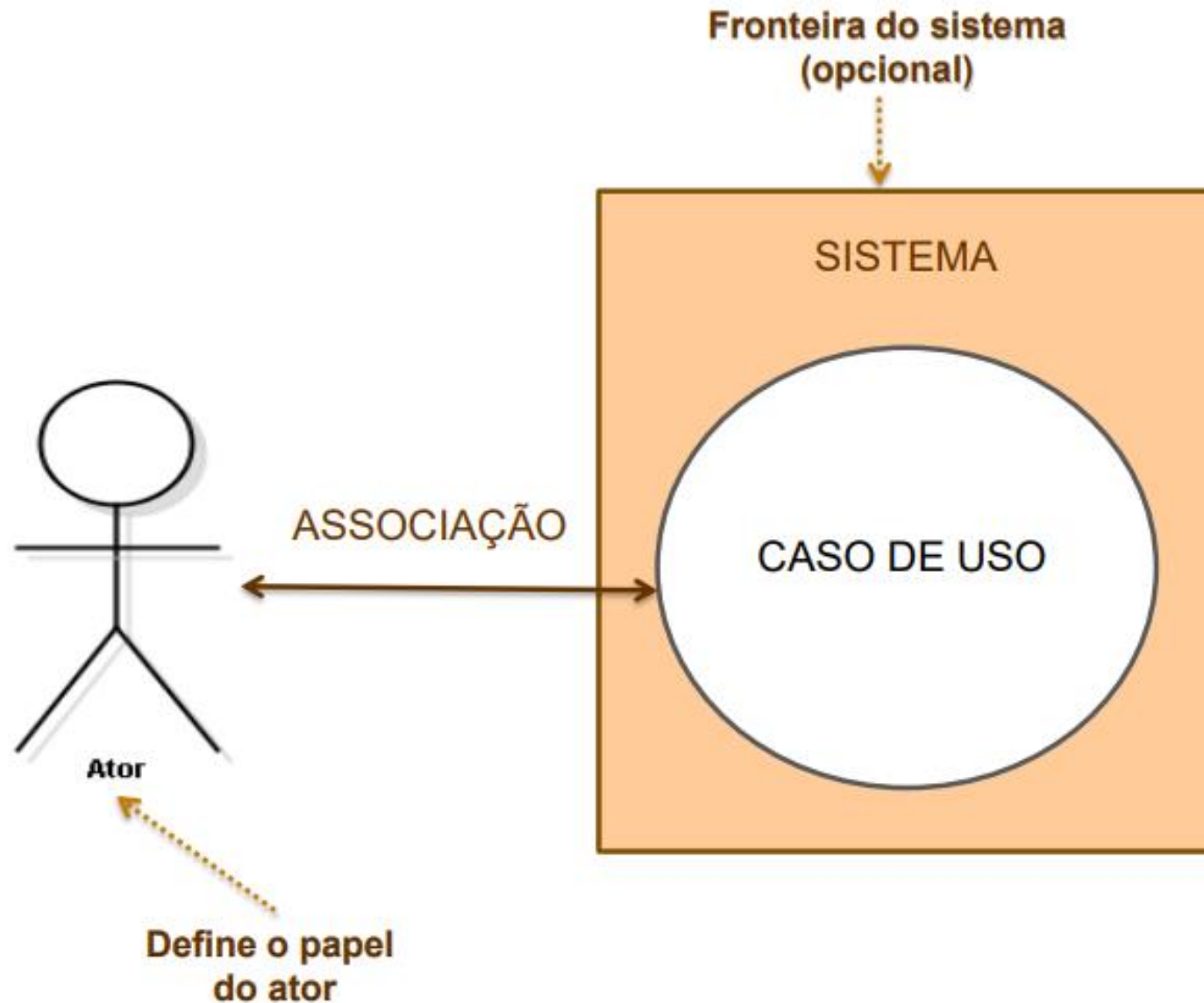
Elementos básicos

Atores

Caso de Uso

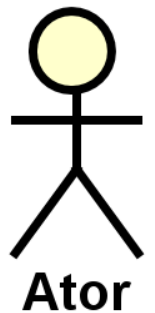
Associação

Sistema



Atores

- Representam os papéis desempenhados pelos diversos usuários que poderão utilizar ou interagir com os serviços e funções do sistema.
- Pode ser qualquer elemento externo que interaja com o sistema, inclusive um software ou hardware.
- Exemplos típicos: cliente, aluno, supervisor, professor, impressora fiscal, dispositivo de conexão de rede etc.
- Identificando atores de um sistema:
- Quem utilizará a principal funcionalidade do sistema?
- Quem (ou o que) tem interesse nos resultados do sistema?
- Quais dispositivos (hardware) são necessários?
- Com quais outros sistemas o sistema em foco irá interagir?



Casos de Uso

- Referem-se aos serviços, tarefas ou funções que podem ser utilizados pelos usuários do sistema;
- São usados para expressar e documentar os comportamentos das funções do sistema;
- Em geral, podemos associar um caso de uso a uma tela (ou página) de um sistema, apesar de isto não ser uma regra;
- Contém um texto descrevendo o serviço (iniciando-se com um verbo);
- Exemplos:
 - Cadastrar produto;
 - Gerar relatório de vendas;
 - Emitir NF-e.



Características dos Casos de Uso

É sempre iniciado por um ator

- Realizado em nome de um ator que, por sua vez, deve pedir direta ou indiretamente ao sistema tal realização.

Um caso de uso é completo

- Deve ser uma descrição completa de um determinado processo.

Deve prover um valor a um ator

- Como resposta à solicitação do ator, retorna um valor

Identificação de Casos de Uso

O trabalho diário do sistema pode ser simplificado ou tornado mais eficiente usando novas funções?

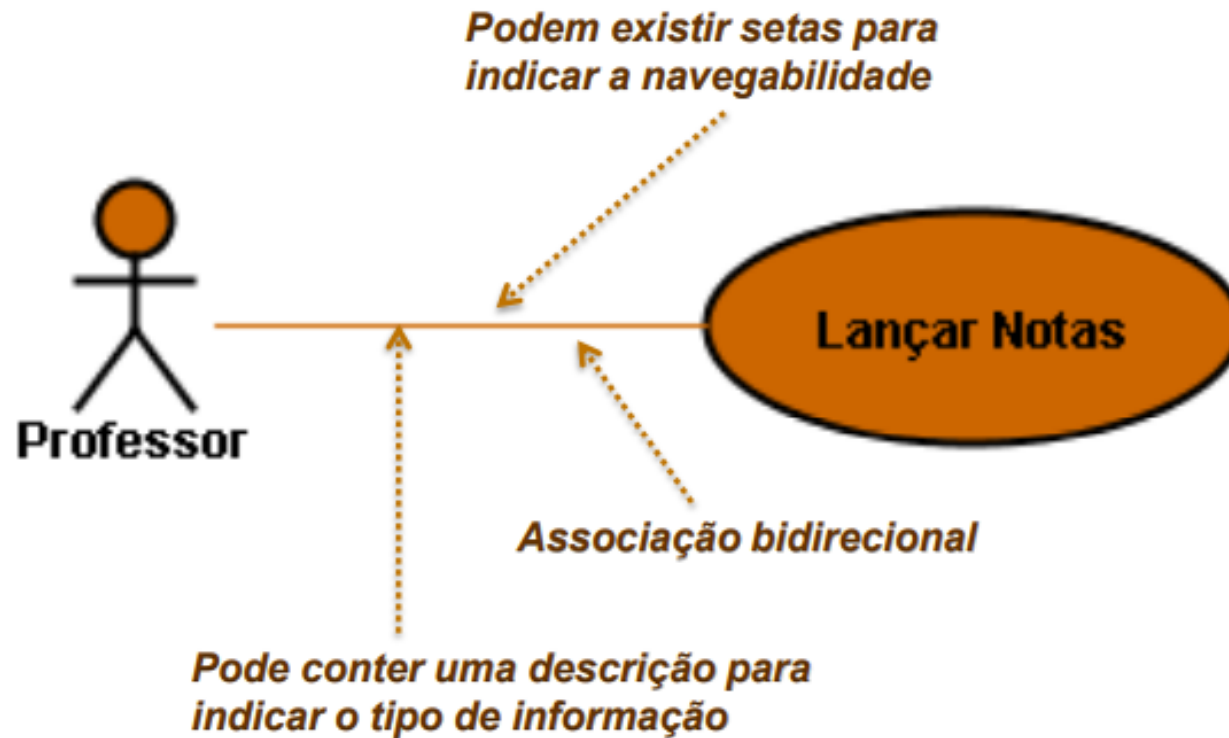
Quais as funções que o ator necessita do sistema?

O que o ator necessita fazer?

Quais são as entradas e as saídas, juntamente com sua origem e destino, que o sistema requer?

Associações

São representadas por uma linha que liga o ator ao caso de uso:



Generalização e Especialização

Forma de associação na qual existem dois ou mais casos de uso com características semelhantes;

Existem pequenas diferenças entre os casos de uso associados;

Também é possível com atores;

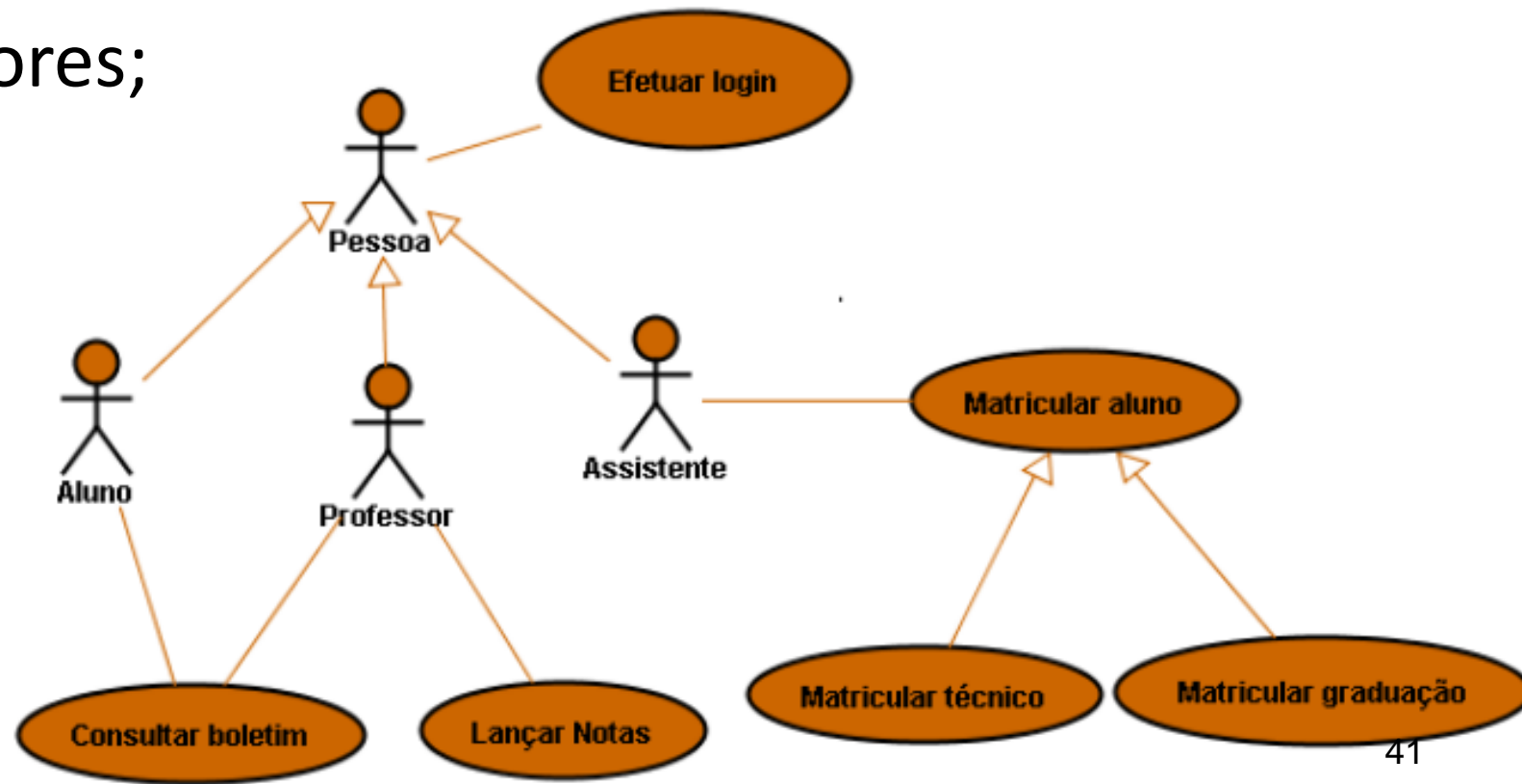


Diagrama de Casos de Uso



Na prática

O **diagrama de casos de uso** tem o objetivo de auxiliar a comunicação entre os analistas e o cliente;

A ideia do diagrama é fazer com que o cliente visualize as principais funcionalidades de seu sistema e **quem pode executar o que**;

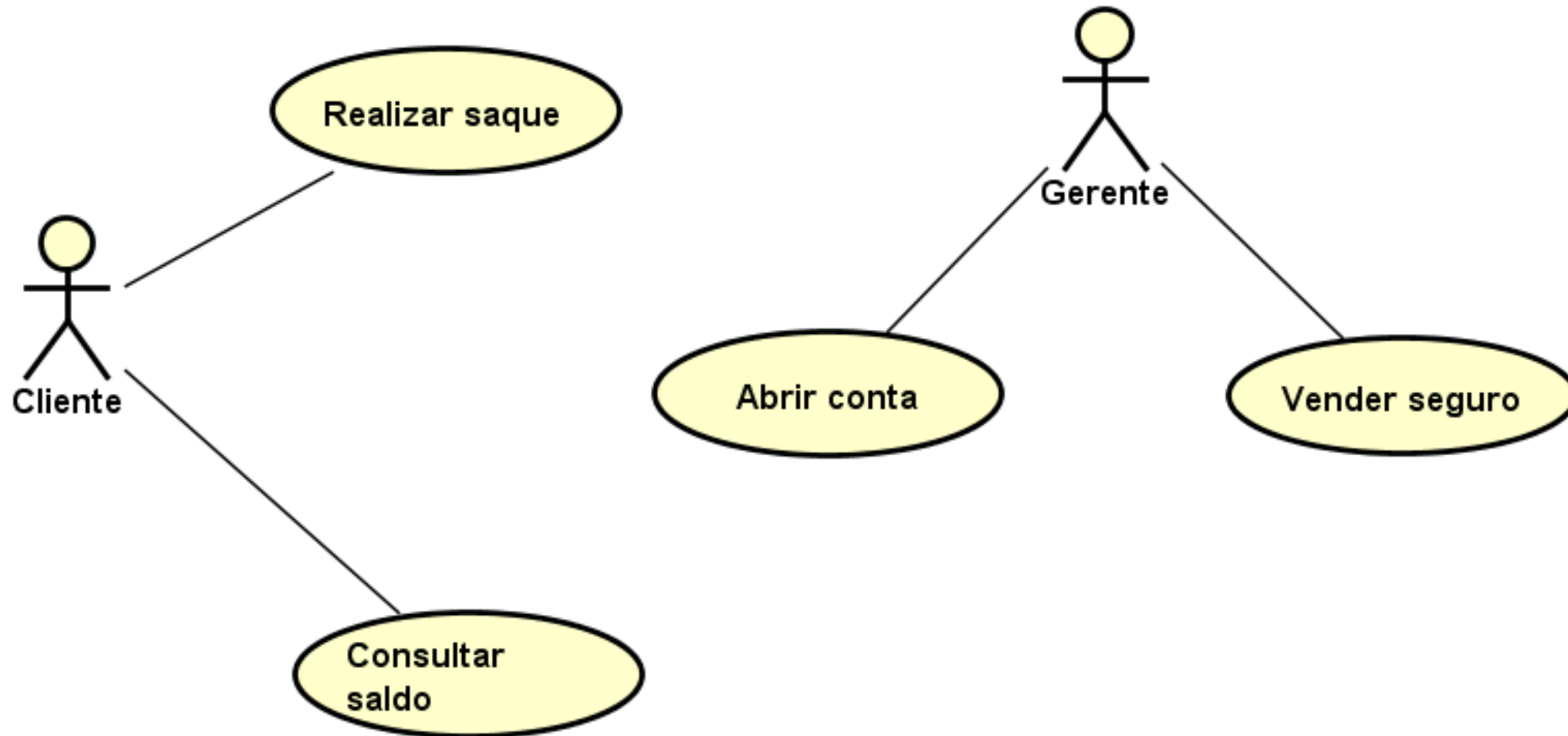
São compostos por:

- Atores

- Casos de uso

- Relação entre eles

Exemplo simples

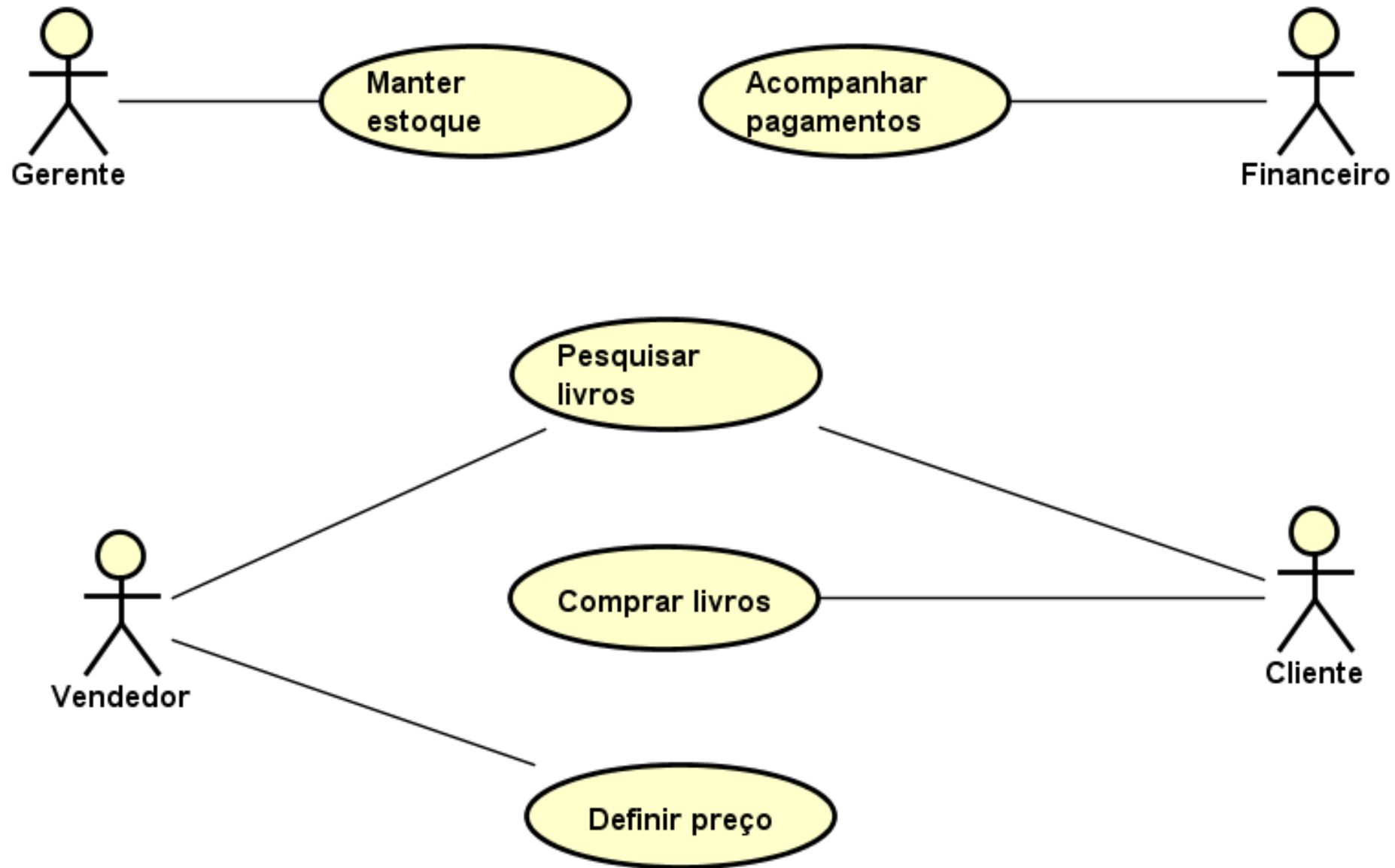


A ideia

Nesse diagrama representamos os atores que interagem com o sistema e os casos de uso (funções do sistema) que os atores podem executar;

A seguir, será apresentado um diagrama de casos de uso típico de uma sistema de venda de livros;

Exemplo

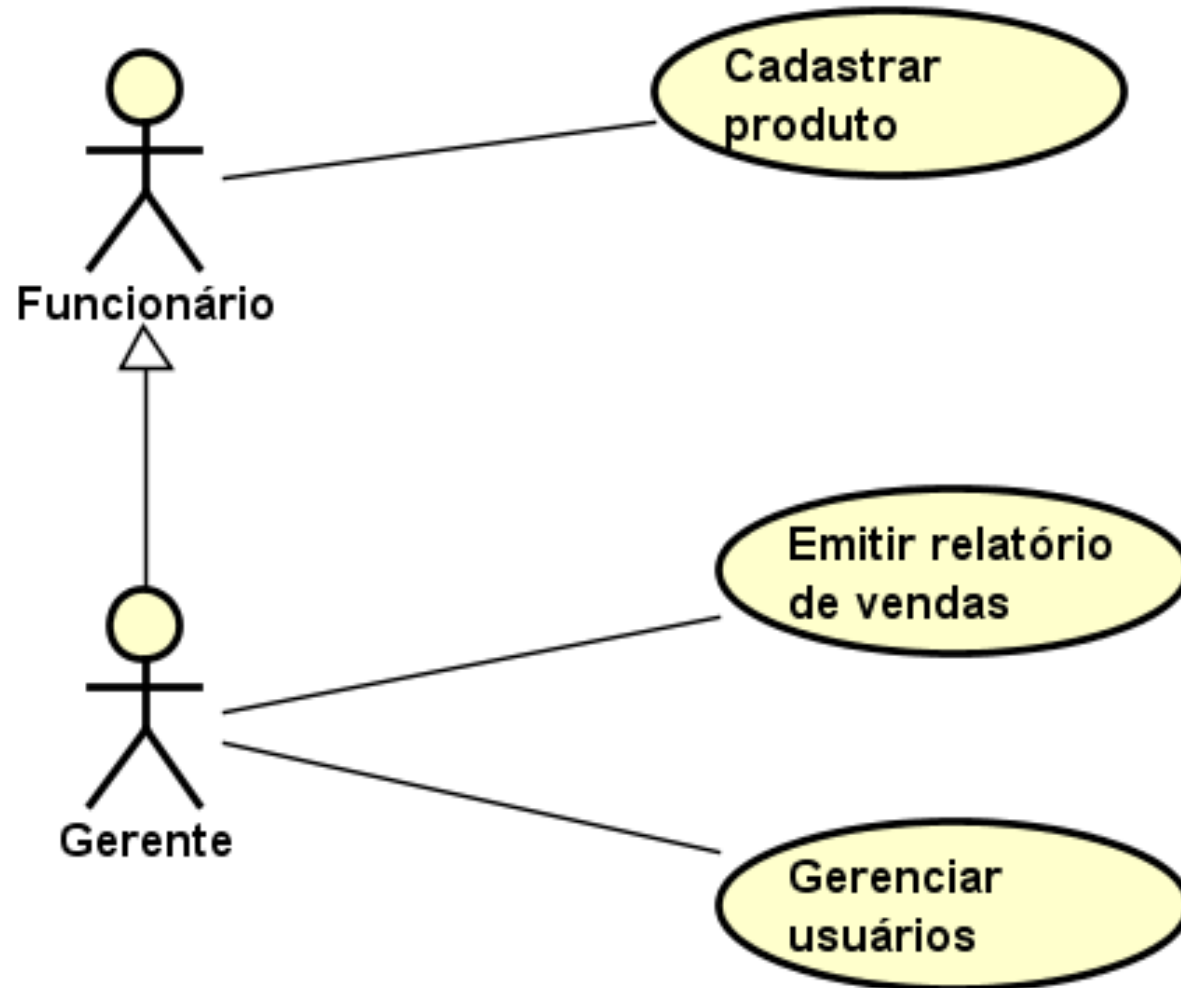


Orientação a Objetos

Assim como na orientação a objetos, também podemos ter relacionamento entre os atores;

A generalização (também conhecida como especialização ou herança) permite que um ator herde todas as características de um ator pai;

Generalização / Especialização



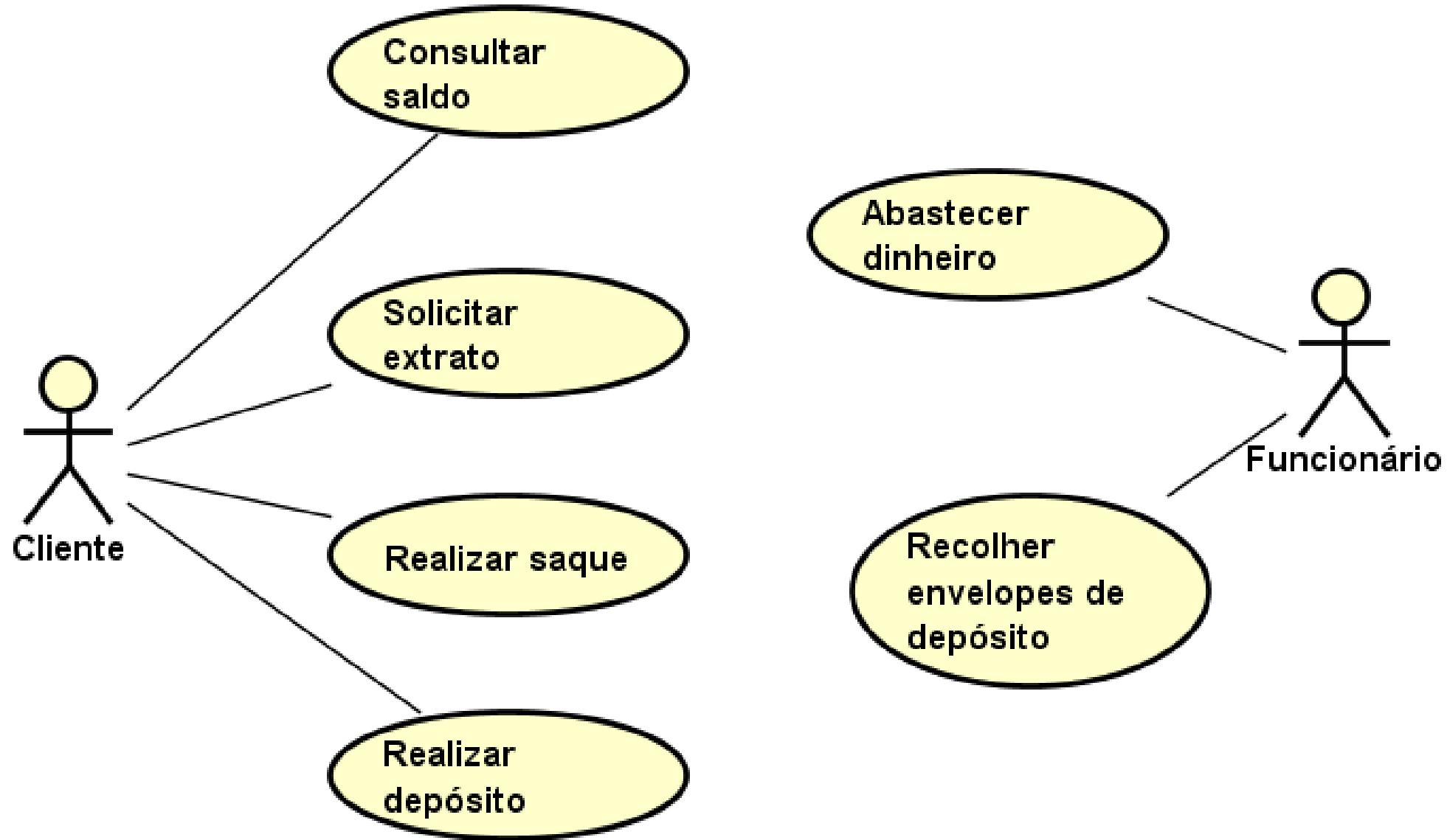
Modelagem de Casos de Uso

O diagrama de casos de uso exerce um papel importante na análise de sistemas:

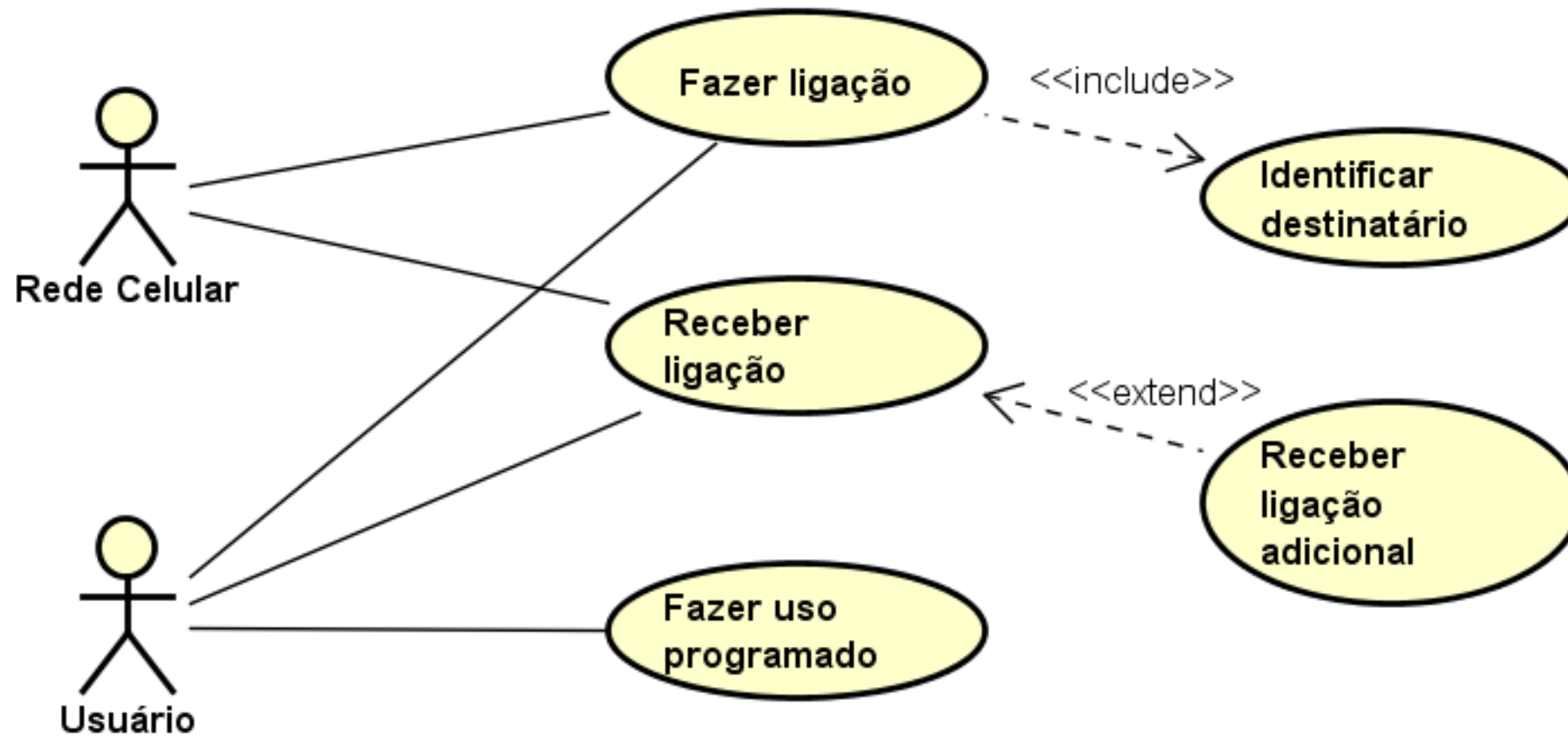
- É o principal diagrama para ser usado no diálogo com o usuário na descoberta e validação de requisitos;

- Os casos de uso constituem elementos que estruturam todas as etapas do processo de software.

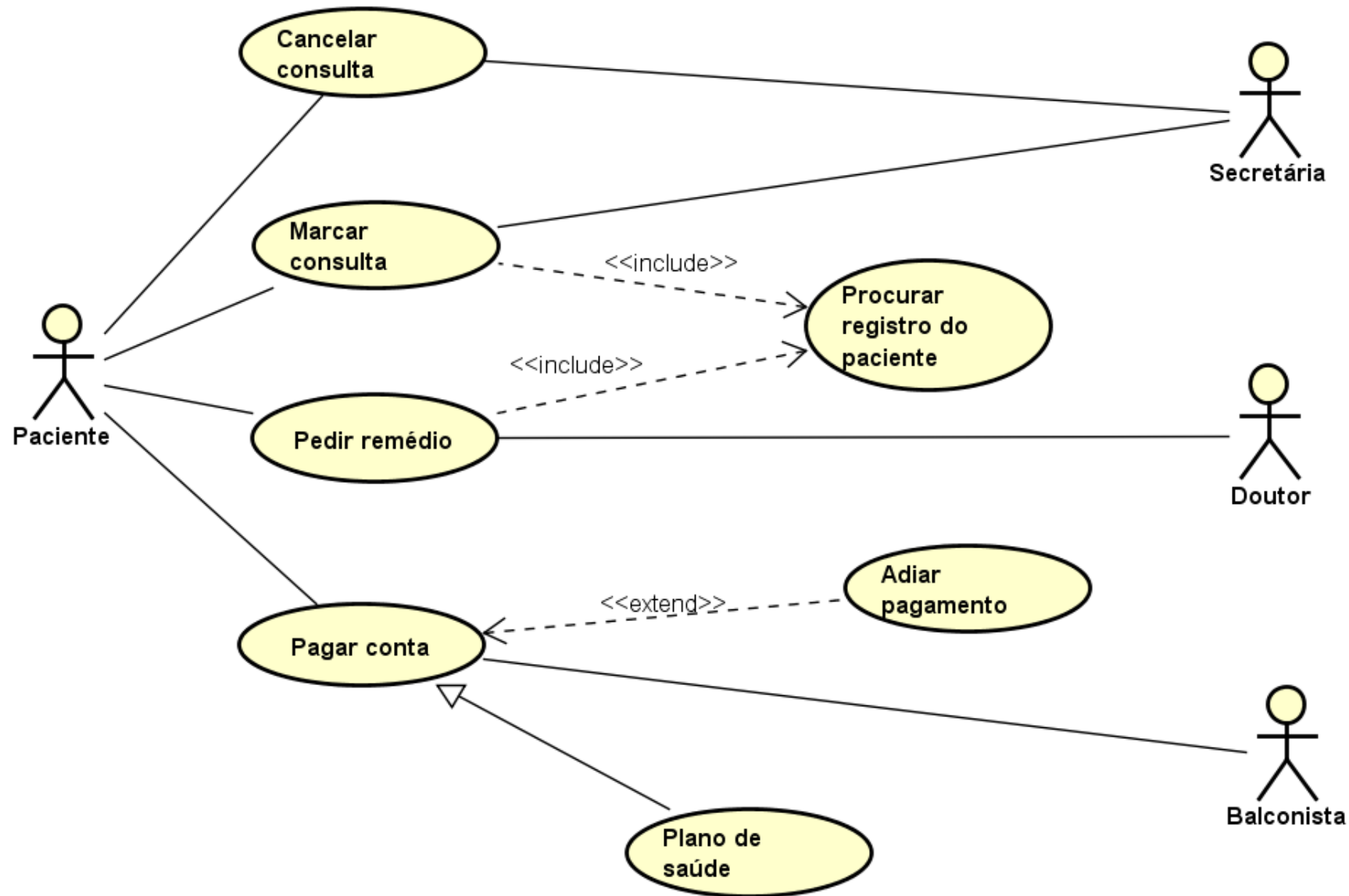
Exemplo



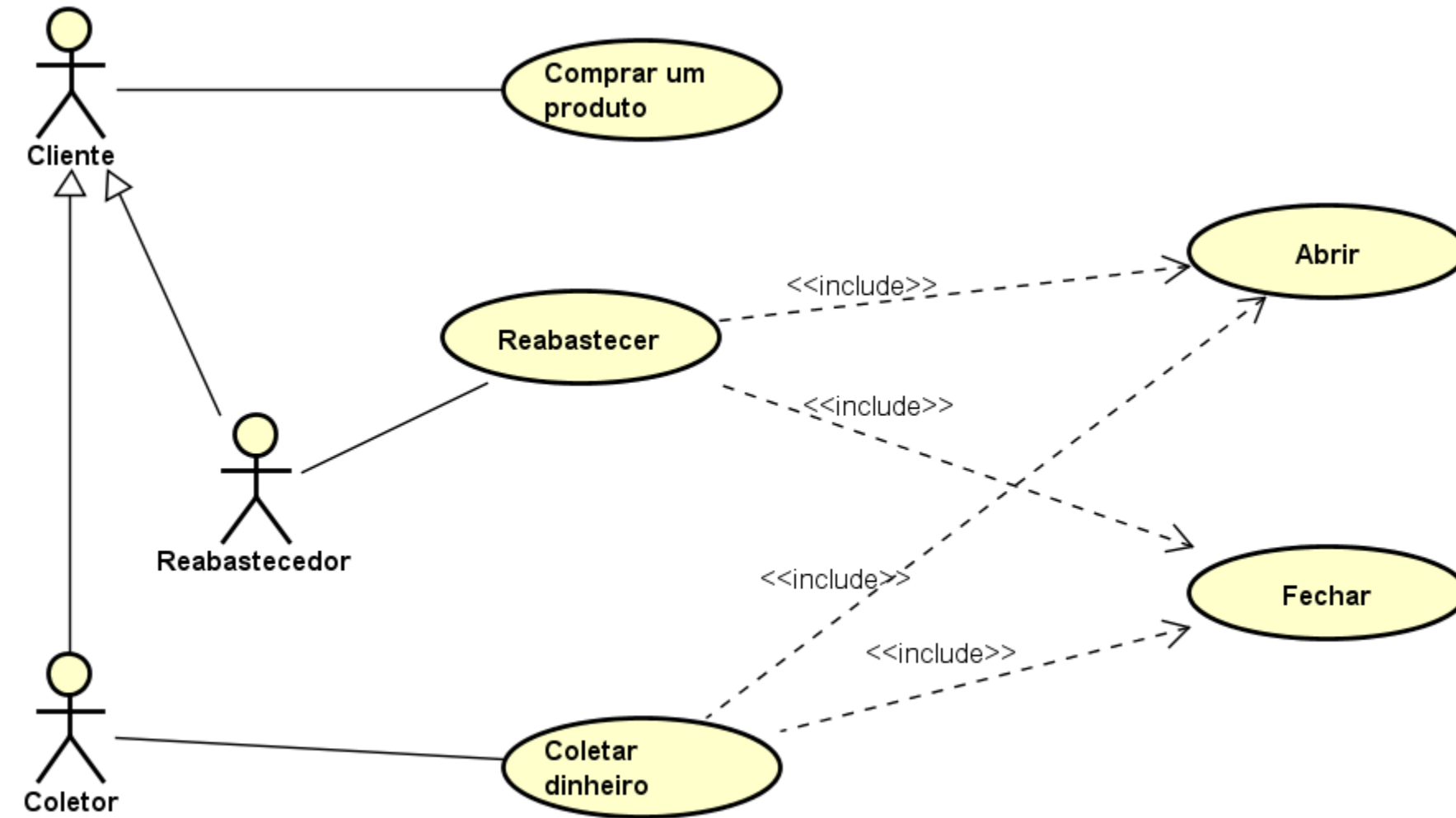
Exemplo – Telefone celular



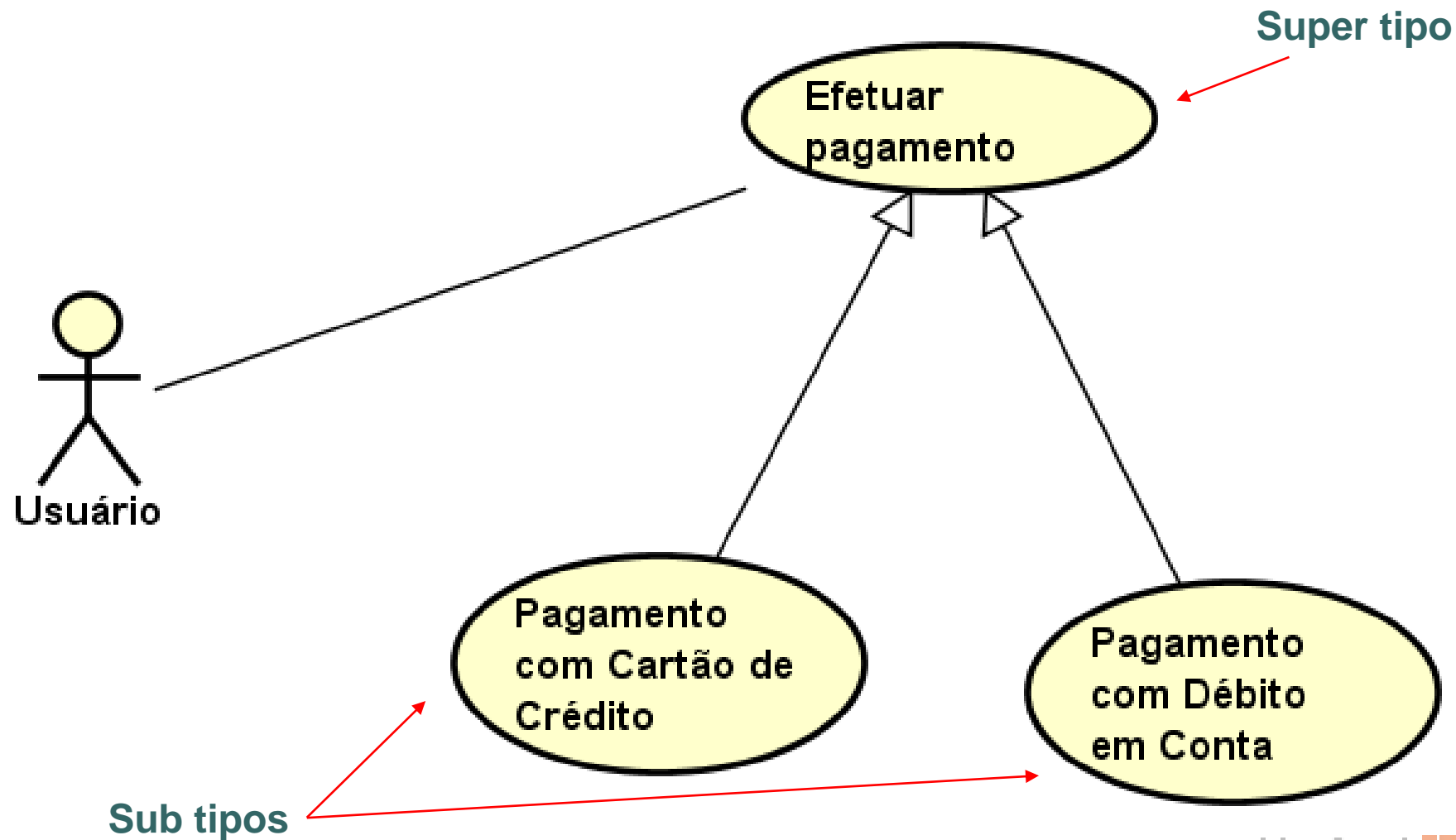
Exemplo – Clínica



Exemplo – Máquina SelfService



Exemplo – Generalização



Tipos possíveis de associação

	COMUNICAÇÃO	EXTENSÃO	INCLUSÃO	HERANÇA
Caso de Uso-Caso de Uso		X	X	X
Ator-Ator				X
Caso de Uso-Ator	X			

Exercícios

Elabore Histórias de Usuário, Casos de Teste e Diagramas de Casos de Uso para:

1. Biblioteca escolar.
2. Sistema de reserva de salas.
3. Sistema de entrega de pizzas.

**EU NÃO AGUENTO,
É MUITO TRABALHO!**



FIM