

Engenharia de Software I

Requisitos de Software:

- Engenharia de Requisitos

→ **Especificação de Requisitos**

→ **Diagrama de casos de uso**



**Elicitação/
Especificação
de requisitos**

**Manutenção
Evolução**



Plano de testes

Projeto de software



Codificação



Implantação



Testes de software

ID	REQUISITO FUNCIONAL	DESCRIÇÃO DO REQUISITO FUNCIONAL
RF01	Cadastrar usuário	<p>O sistema deve possibilitar cadastrar novos usuários.</p> <p>Deverá conter: E-mail (que será utilizado para realizar o login), Nome do usuário, celular e data de nascimento.</p> <p>Ao adicionar um novo usuário, automaticamente será criada uma conta com saldo zero.</p>
RF02	Efetuar Login	<p>Permitir que o usuário acesse o sistema através do seu e-mail (login) e senha.</p> <p>O sistema deverá armazenar um histórico de logs de acesso dos usuários ao sistema (usuário, data, hora de acesso, ip).</p>
RF03	Registrar recebimentos	<p>Registrar recebimentos de várias fontes de rendas e classificar por categoria do recebimento (salário, bolsa de estudos, estágios, auxílio familiar, etc).</p> <p>Deve ter opção de registrar se é um recebimento fixo mensal ou eventual.</p> <p>Deverá conter: nome da fonte do recebimento, data do recebimento, valor recebido e o tipo da categoria do recebimento.</p> <p>Após incluir o recebimento deverá somar ao saldo da conta do usuário.</p>
RF04	Categoria de recebimentos	<p>Permitir o usuário criar categorias (tags) de recebimentos para melhor organizar seus créditos. Campos: Código da categoria e Nome da categoria. (exemplo: salário, bolsa de estudos, estágios, auxílio familiar, etc)</p>

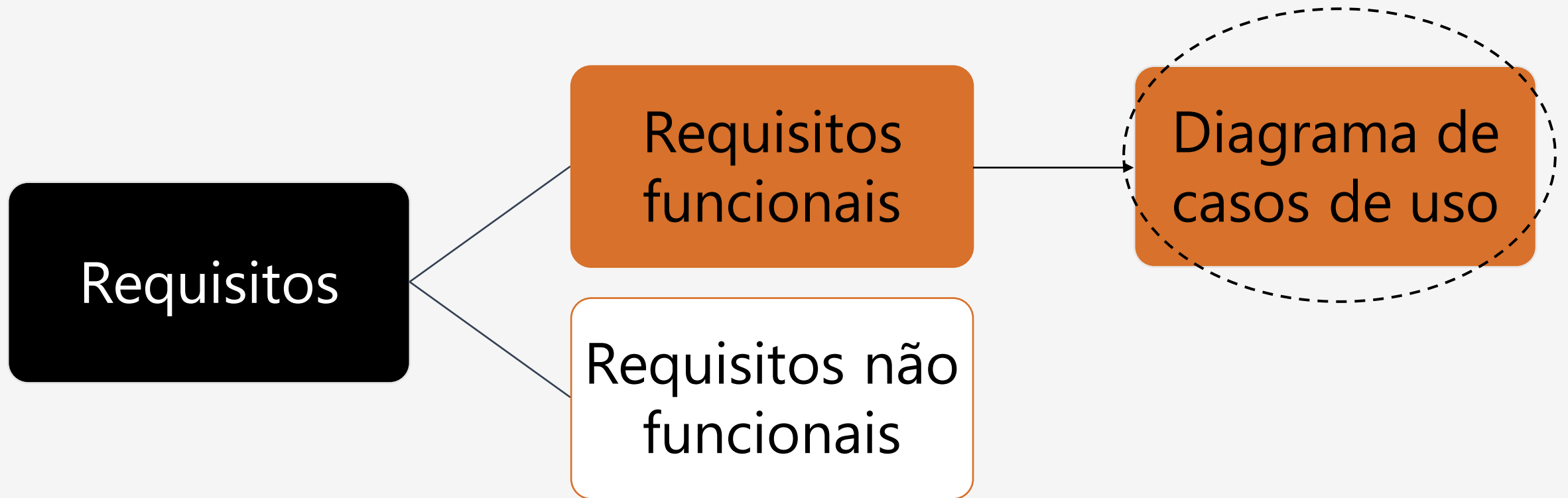


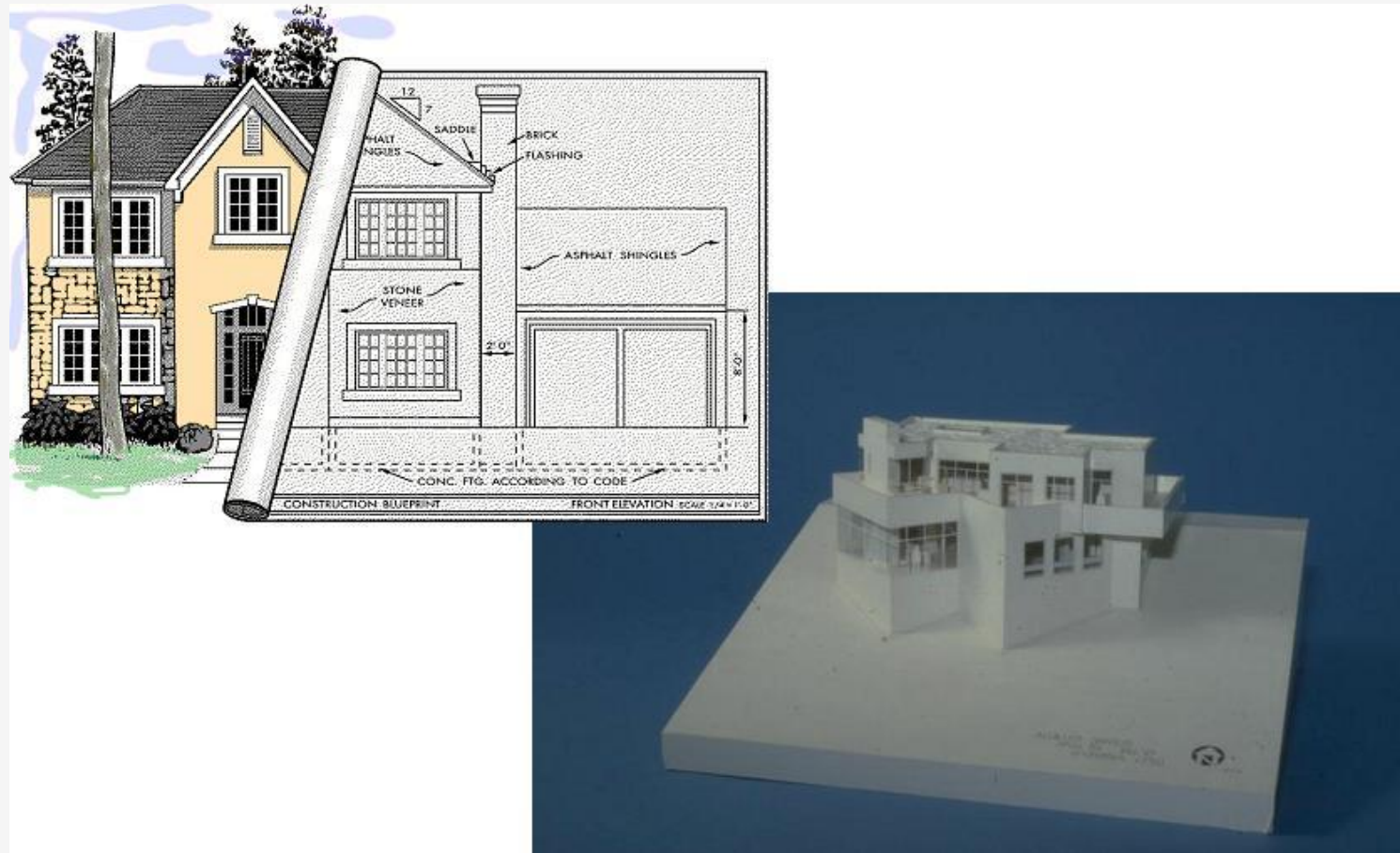
Diagrama de casos de uso

UML



- UML – Linguagem de Modelagem Unificada
- A UML é uma linguagem visual utilizada para **modelar softwares** baseados no paradigma de orientação a objetos.
- Tornou-se, nos últimos anos, a linguagem **padrão de modelagem adotada internacionalmente** pela indústria de engenharia de software.

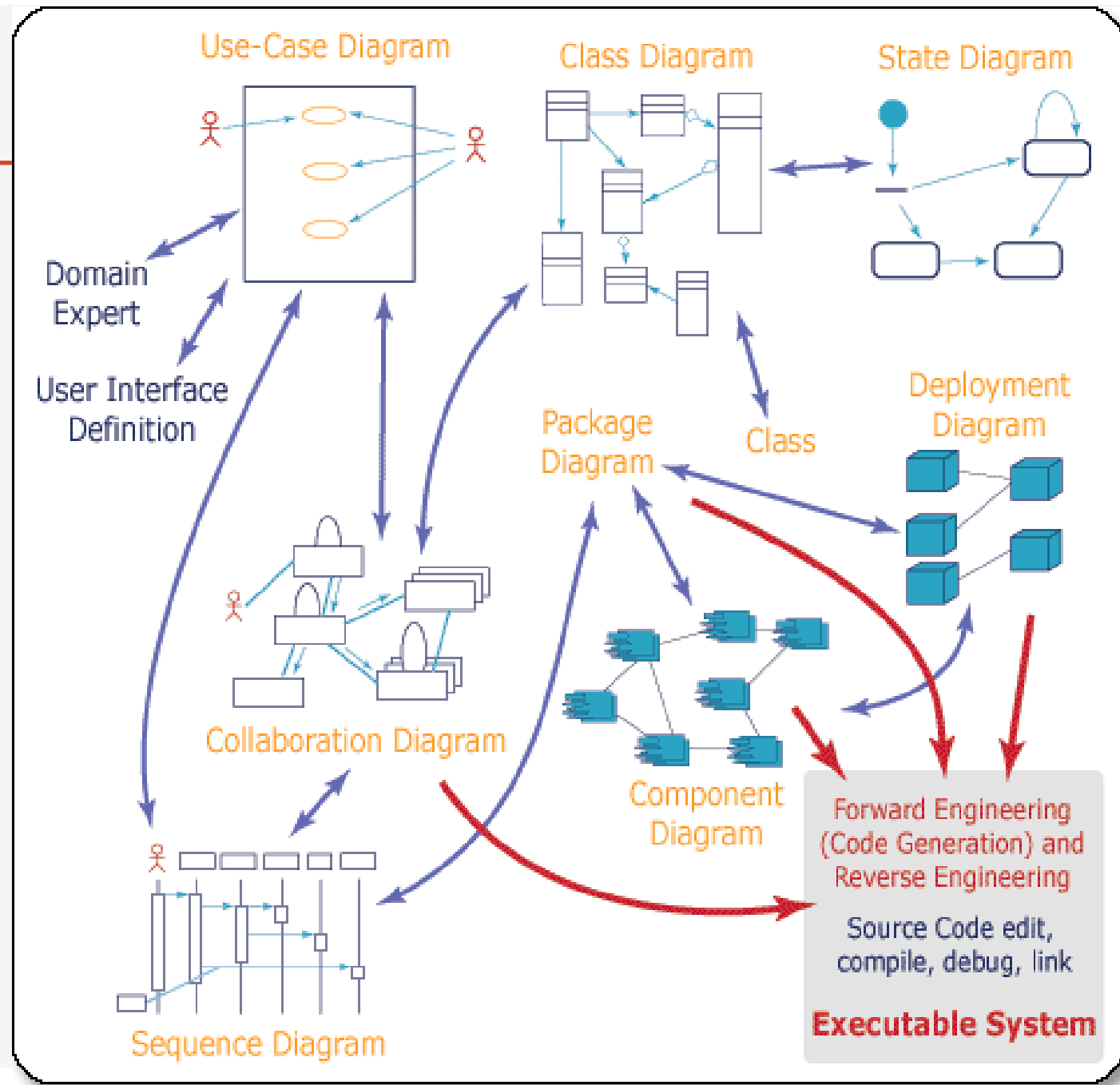
Representação do projeto de software de forma visual e padronizada, como ocorre na construção civil



Diagramas da UML

Modelagem visual:

necessidade de modelar o sistema seguindo uma notação padrão



UML

- A UML foi um passo importante em direção à padronização de desenvolvimento de software
- Surgiu em 1995 da união de três métodos de modelagem OO: Booch, OMT e OOSE
- Tornou-se padrão para modelagem e projetos de sistemas



Grady Booch
Método Booch



Ivar Jacobson
Método OMT (Rational)



James Rumbaugh
Método OOSE

UML

Linguagem consistente para:

- Visualização
- Especificação
- Documentação
- Construção

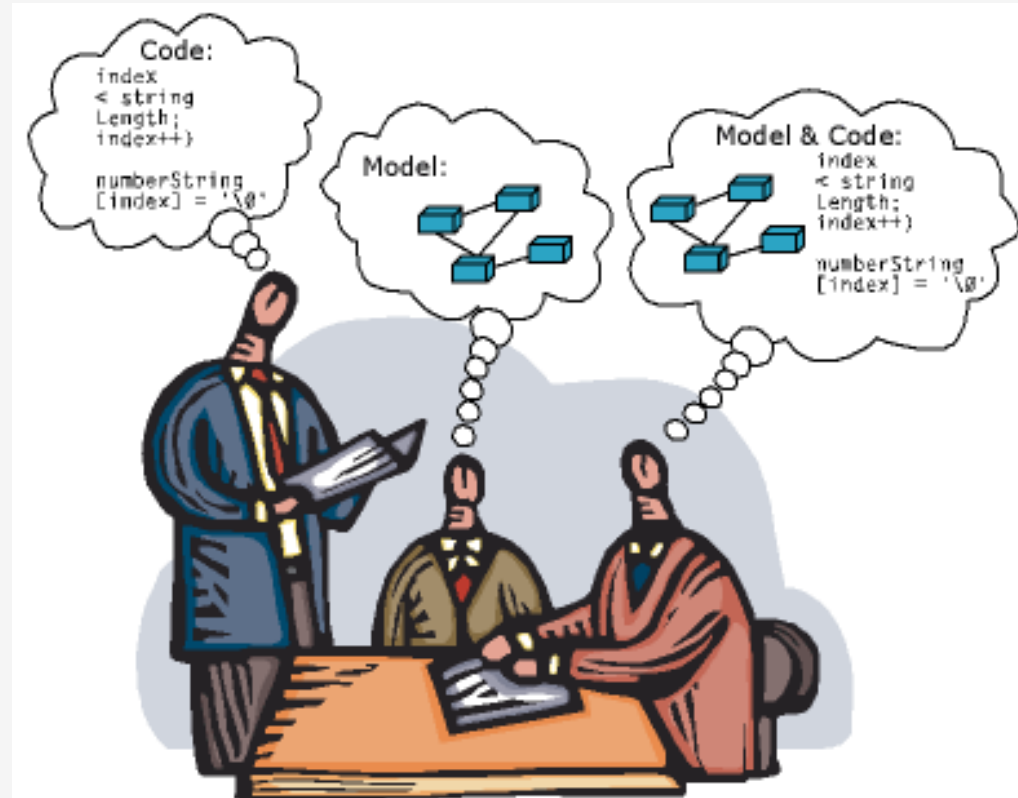


Diagrama de casos de uso

- Técnica **especificação de requisitos** baseada em cenários
- identificam os agentes em uma interação (atores) e descrevem a interação com o sistema (casos de uso).

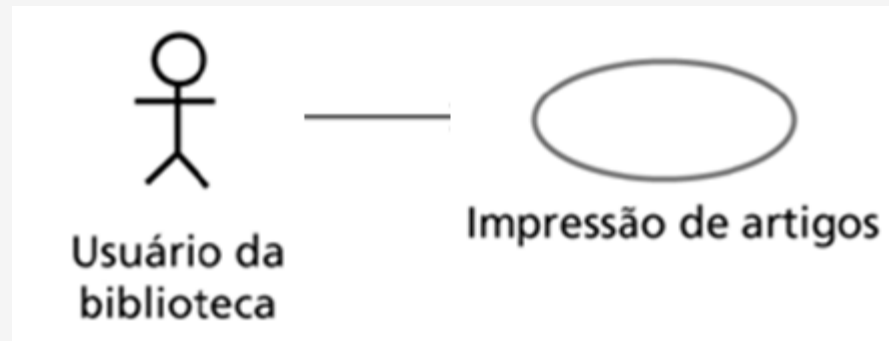
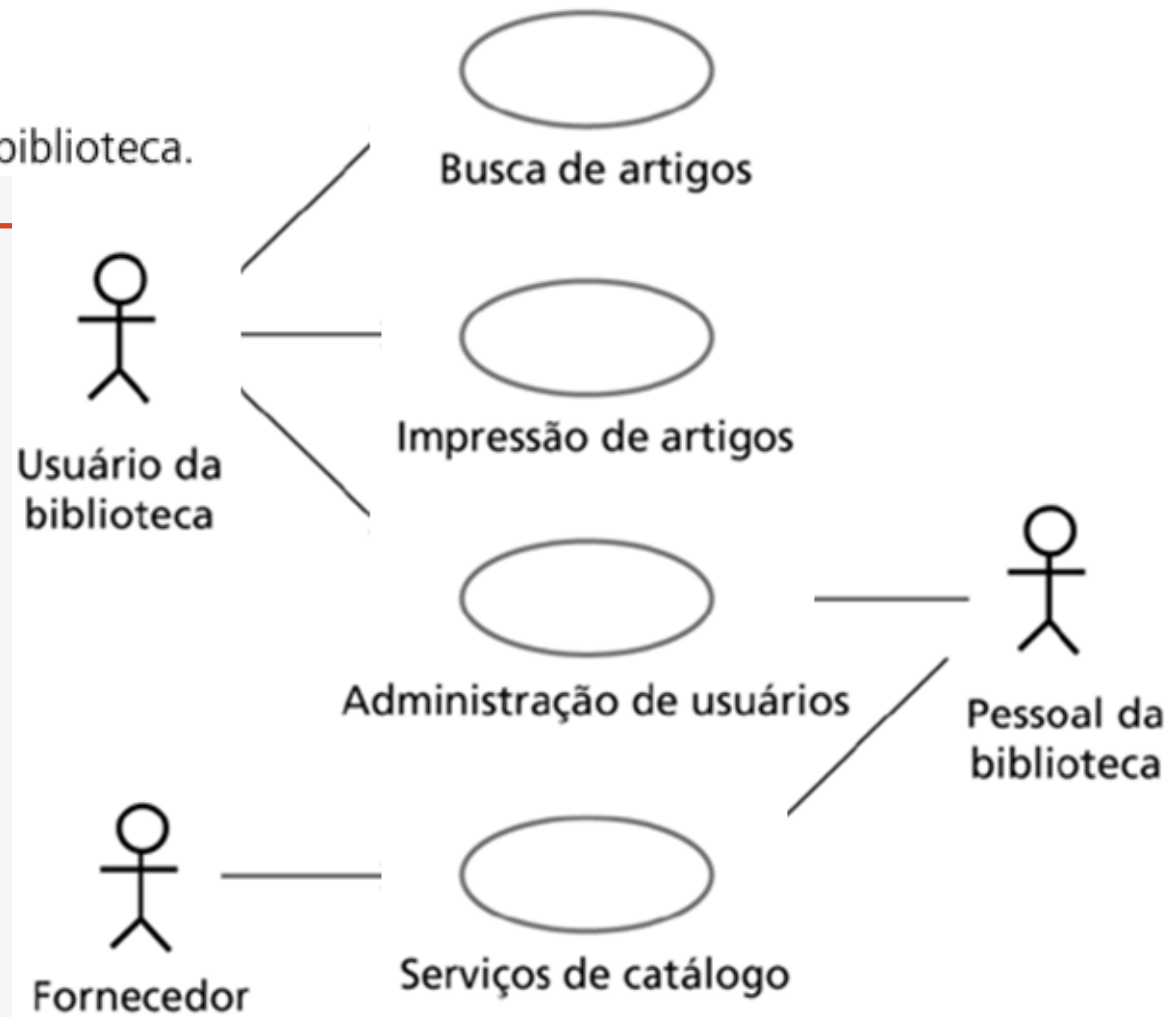


Figura 7.6

Casos de uso para o sistema de biblioteca.



Mostra os potenciais usuários do sistema (atores) e as funcionalidades que serão úteis para eles (casos de uso)

Casos de uso

- Um conjunto de casos de uso deve descrever todas as possíveis interações do ator com o sistema.
- Os diagramas de caso de uso são usados para especificar os requisitos funcionais do sistema.
- Permite que os requisitos fiquem acessíveis e compreensíveis a todos, por apresentarem uma visão externa sobre como os elementos que interagem com sistema.
- Modelagem dinâmica de sistema em alto nível de abstração

Principais elementos gráficos



→ Atores



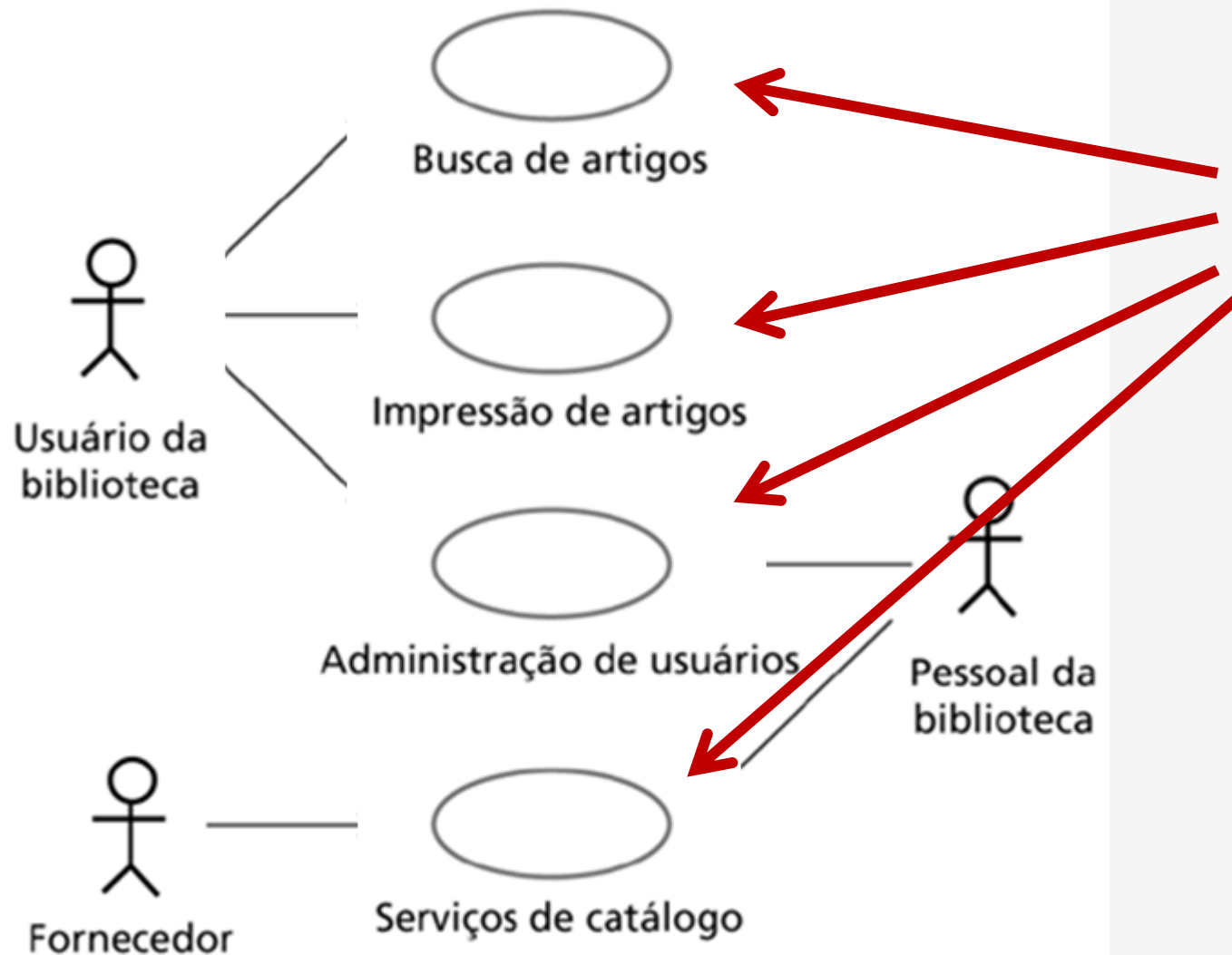
→ Casos de uso



→ Relacionamentos entre os elementos atores e casos de uso

Figura 7.6

Casos de uso para o sistema de biblioteca.



CASOS DE USO

Casos de uso

- Uma funcionalidade do software
- Atômica, completa (não uma fração)
- Externamente perceptível
- EX: cada uma das opções do menu de um caixa eletrônico de banco
 - *emissão de extrato de conta corrente*
 - *Realizar saque*
 - *Transferir dinheiro*

Mono sessão

- Um bom caso de uso deve ser mono sessão. Isso significa que ele deve iniciar e terminar sem ser interrompido.
- Por exemplo, o registro de uma encomenda de livros é feito em uma única sessão de uso do sistema.

Dica

Pode-se pensar assim:

- Somente será um caso de uso um **processo completo**
- Um usuário iria ao computador, ligaria o sistema, executaria o processo e em seguida poderia desligar o computador porque o processo estaria completo.

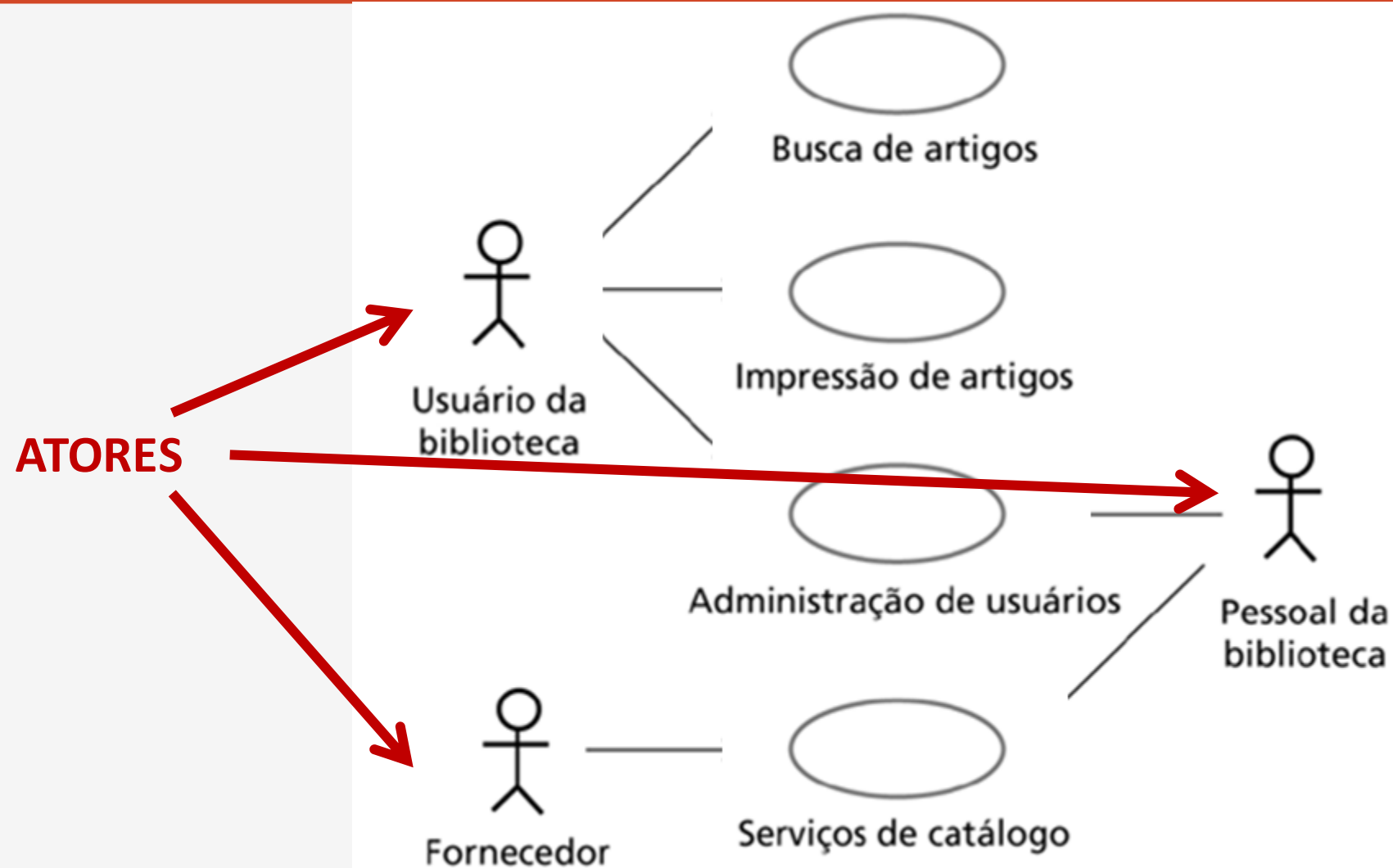
Ator

- É alguém ou algo que interage com o sistema, ou seja, é quem usa o sistema.
- Um ator pode ser um humano, outro sistema ou um software
 - *Vendedor (pessoa)*
 - *Impressora (hardware)*
 - *Sistema bancário (outro software)*



Figura 7.6

Casos de uso para o sistema de biblioteca.



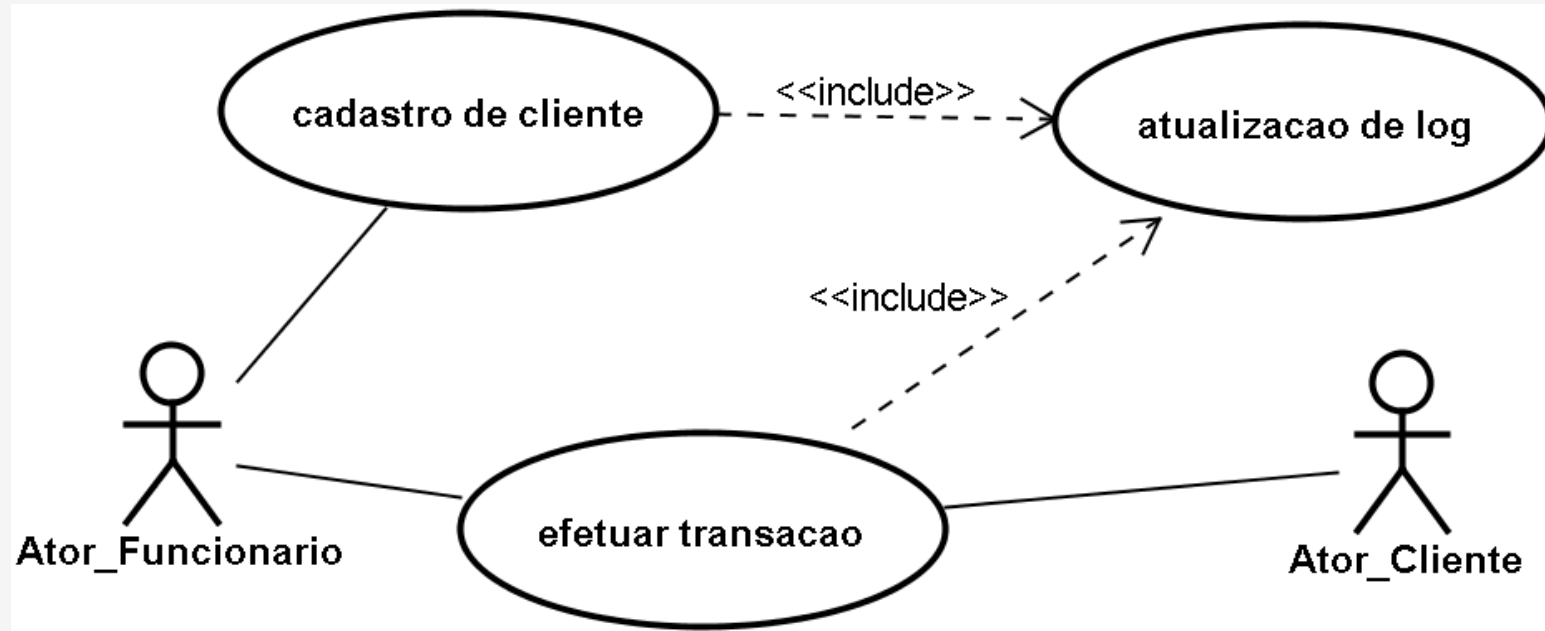
Associação entre ator e caso de uso

- Participação do ator na utilização de um caso de uso
 - Um ator sempre está associado a pelo menos um caso de uso
 - Um caso de uso sempre estará associado a pelo menos um ator

Associação entre casos de uso

Associação entre casos de uso

➔ Inclusão <<include>> (obrigatório)



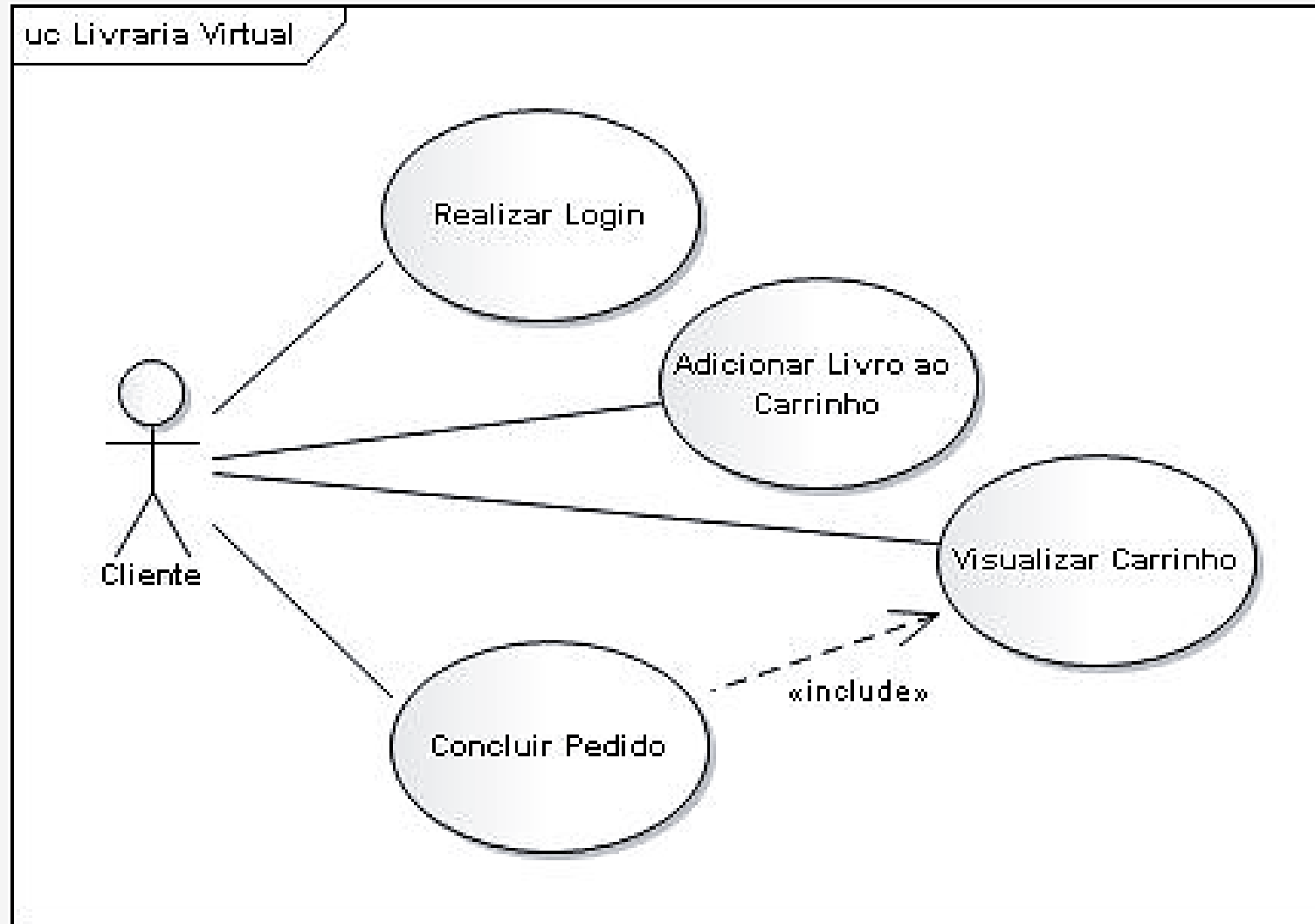
Explicação:

- O comportamento referente ao caso de uso *atualização de log* **está contido** tanto no caso de uso *cadastro de cliente*, quanto no *efetuar transação*

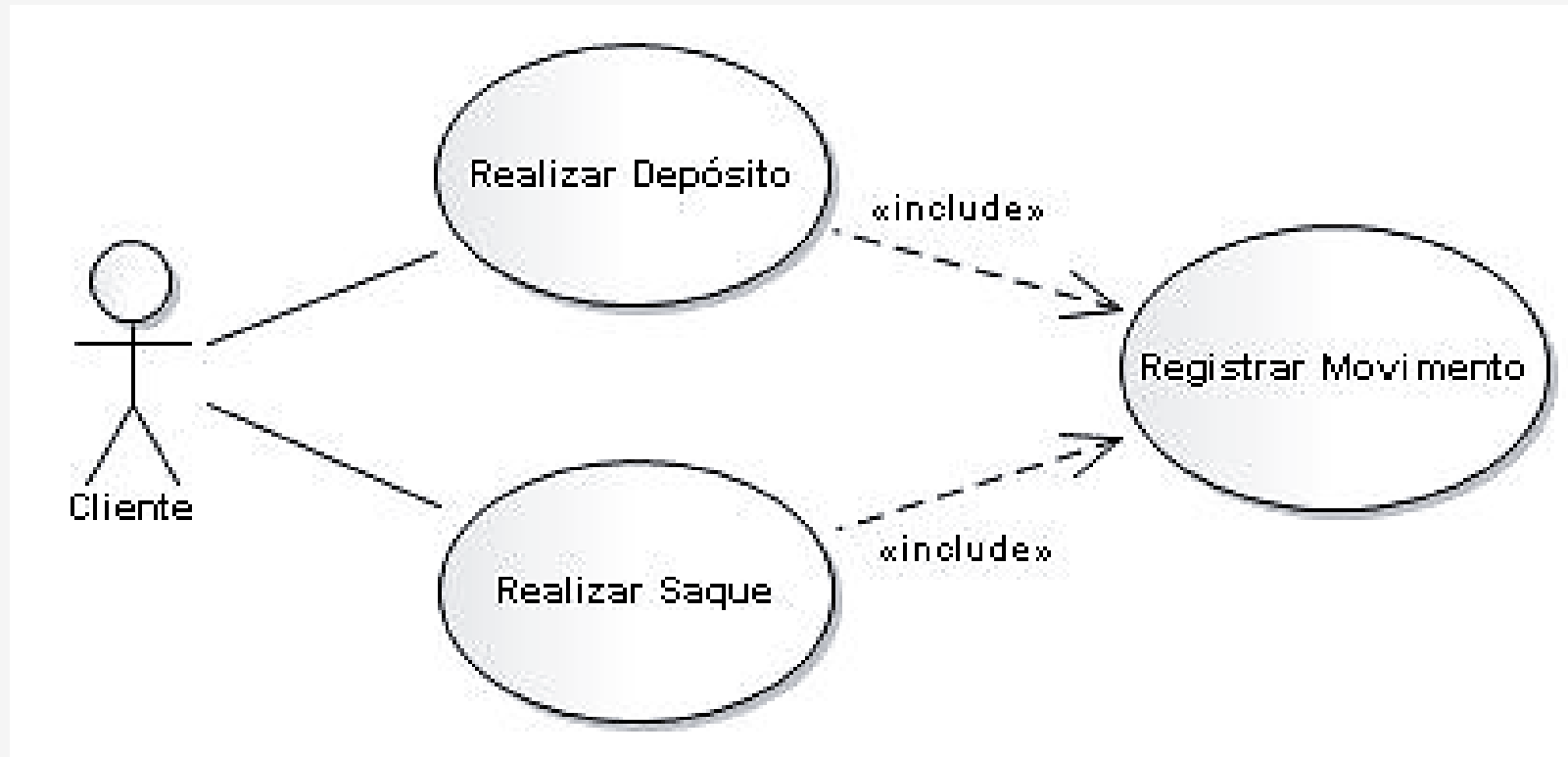
Associação de Inclusão <<include>>

- Estabelece que parte do comportamento de um caso de uso está definida em outro caso de uso
- Um caso de uso contém o comportamento definido em outro caso de uso

Associação de Inclusão <<include>>

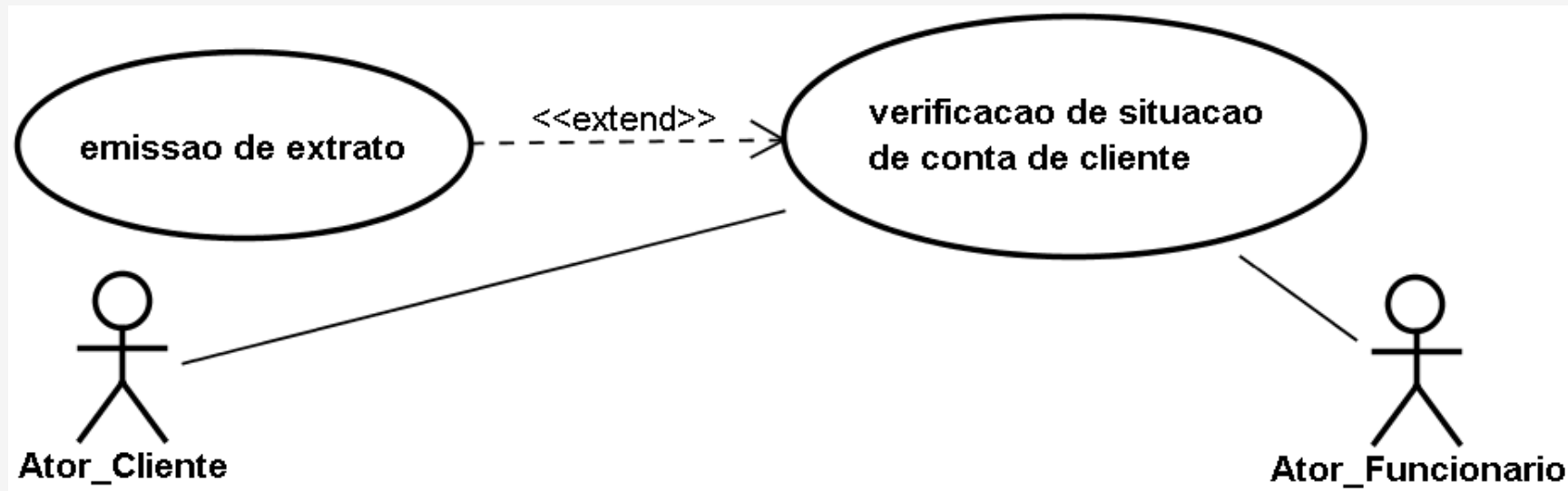


Associação de Inclusão <<include>>



Associação entre casos de uso

→ Extensão <<extend>> (opcional)



Explicação:

- *verificação de situação de conta de cliente tem seu comportamento estendido pelo caso de uso *emissão de extrato**
- Pode **ocorrer com ou sem** *emissão de extrato*

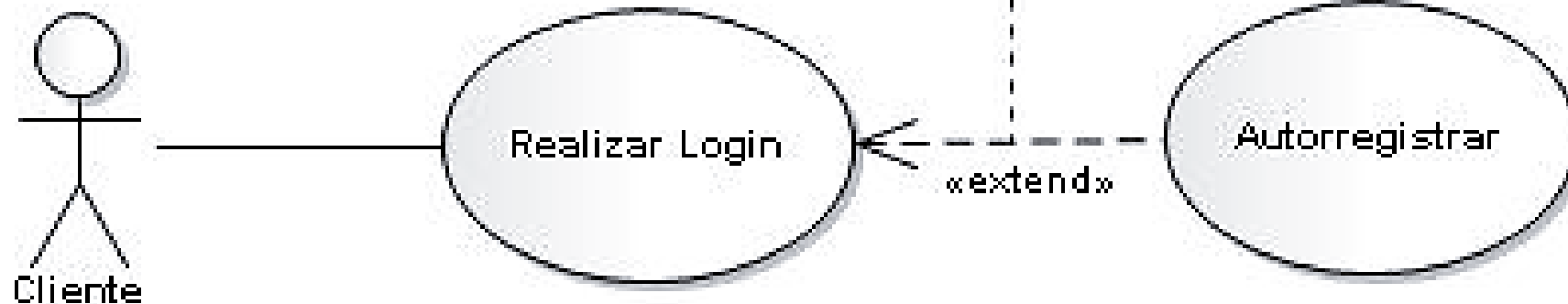
Associação de Extensão <<extend>>

- Estabelece uma relação em que um dos casos de uso tem seu **comportamento estendido** através do comportamento definido em outro caso de uso
- Pode ou não ocorrer

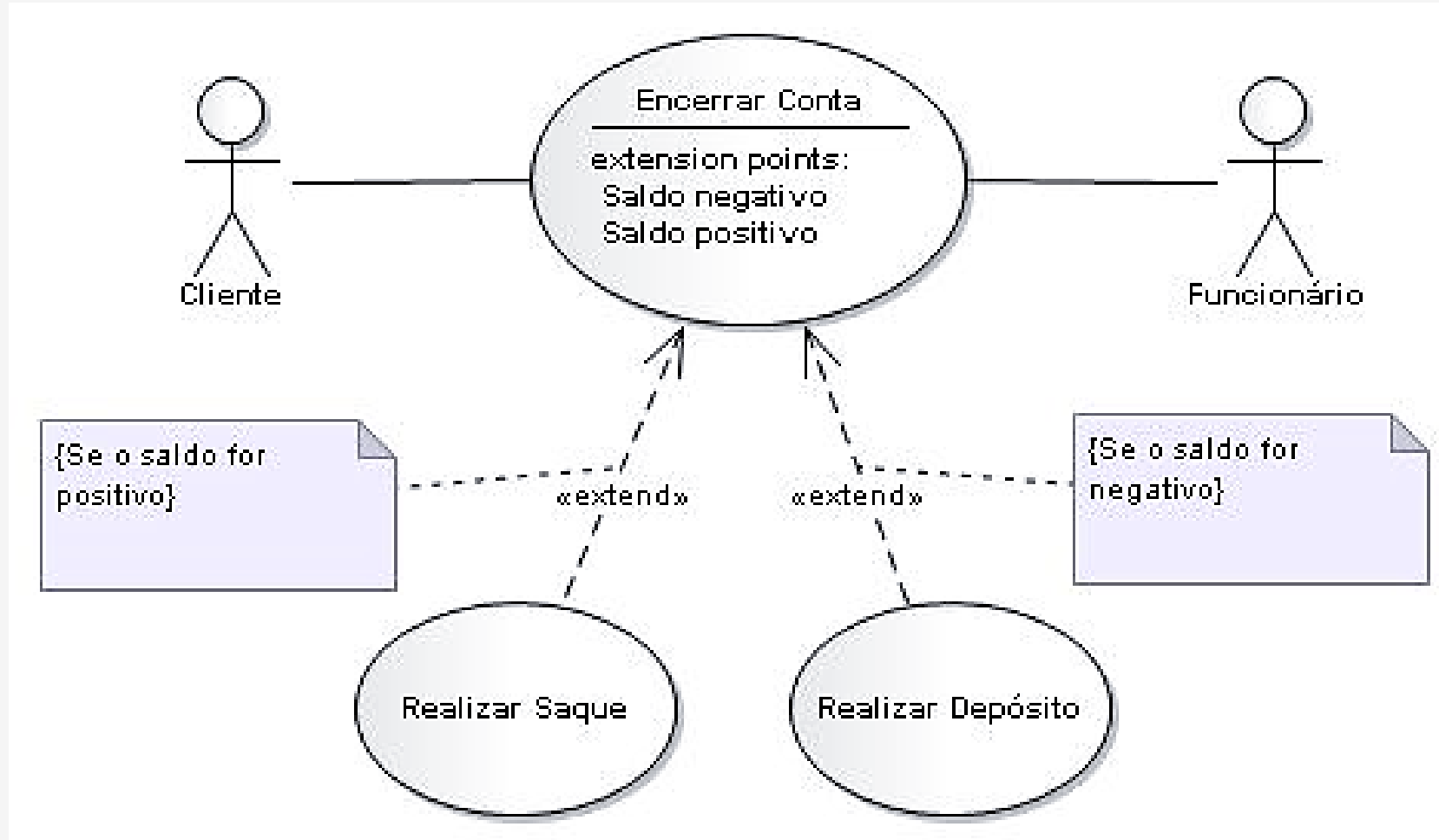
Associação de Extensão <<extend>>

Restrição: define em que situação/condição deve ser executado um determinado caso de comportamento estendido

{Caso o cliente ainda não esteja registrado}



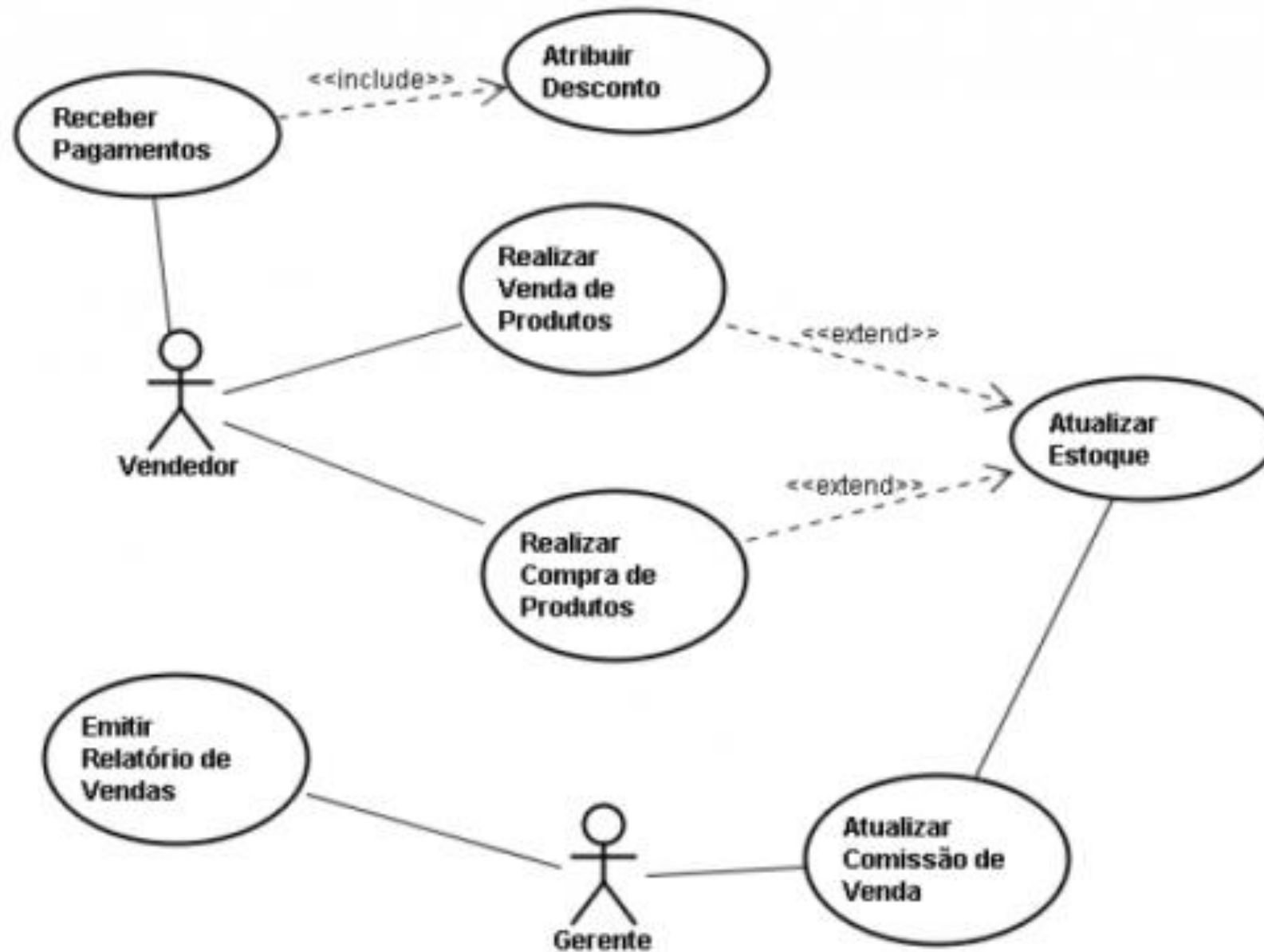
Associação de Extensão <<extend>>



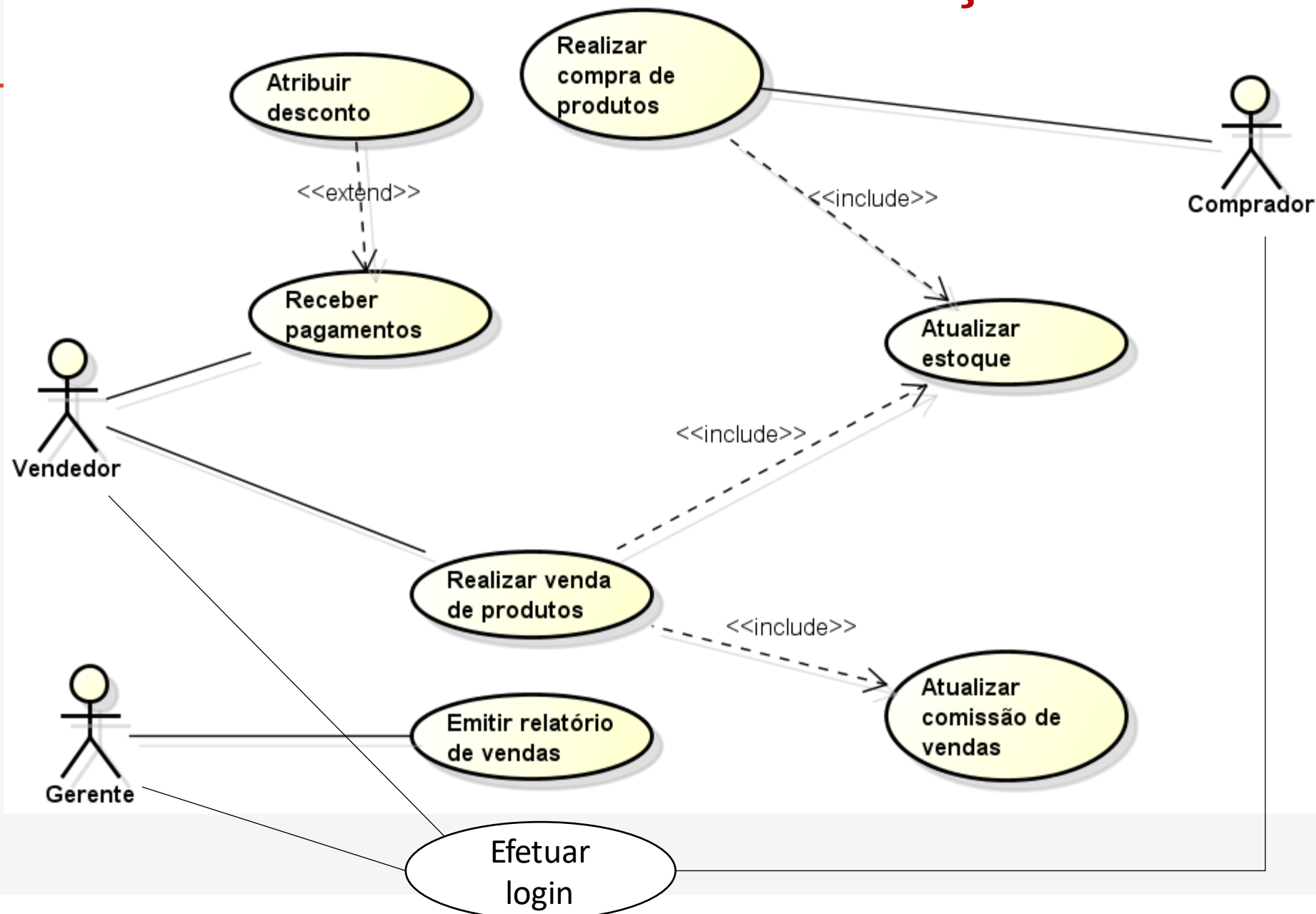
Diferença entre <<include>> e <<extend>>

- **Inclusão** denota **obrigatoriedade**
 - Quando o caso de uso principal ocorre, o caso de uso auxiliar também ocorrerá
- **Extensão** denota **opcionalidade**
 - Quando o caso de uso principal ocorre, o caso de uso auxiliar poderá ou não ocorrer

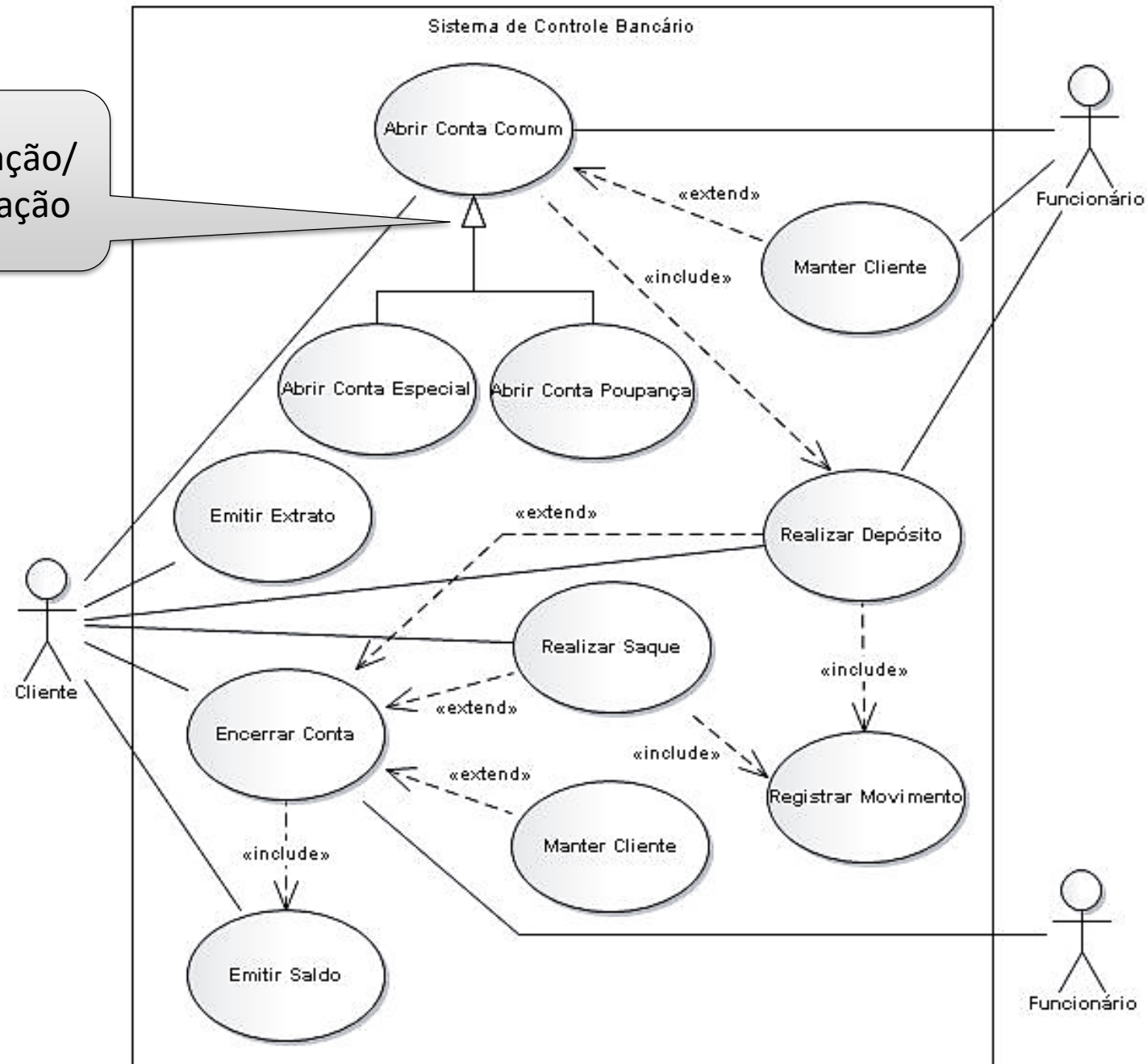
Exercício - Encontre os erros



Exercício - Encontre os erros - Solução

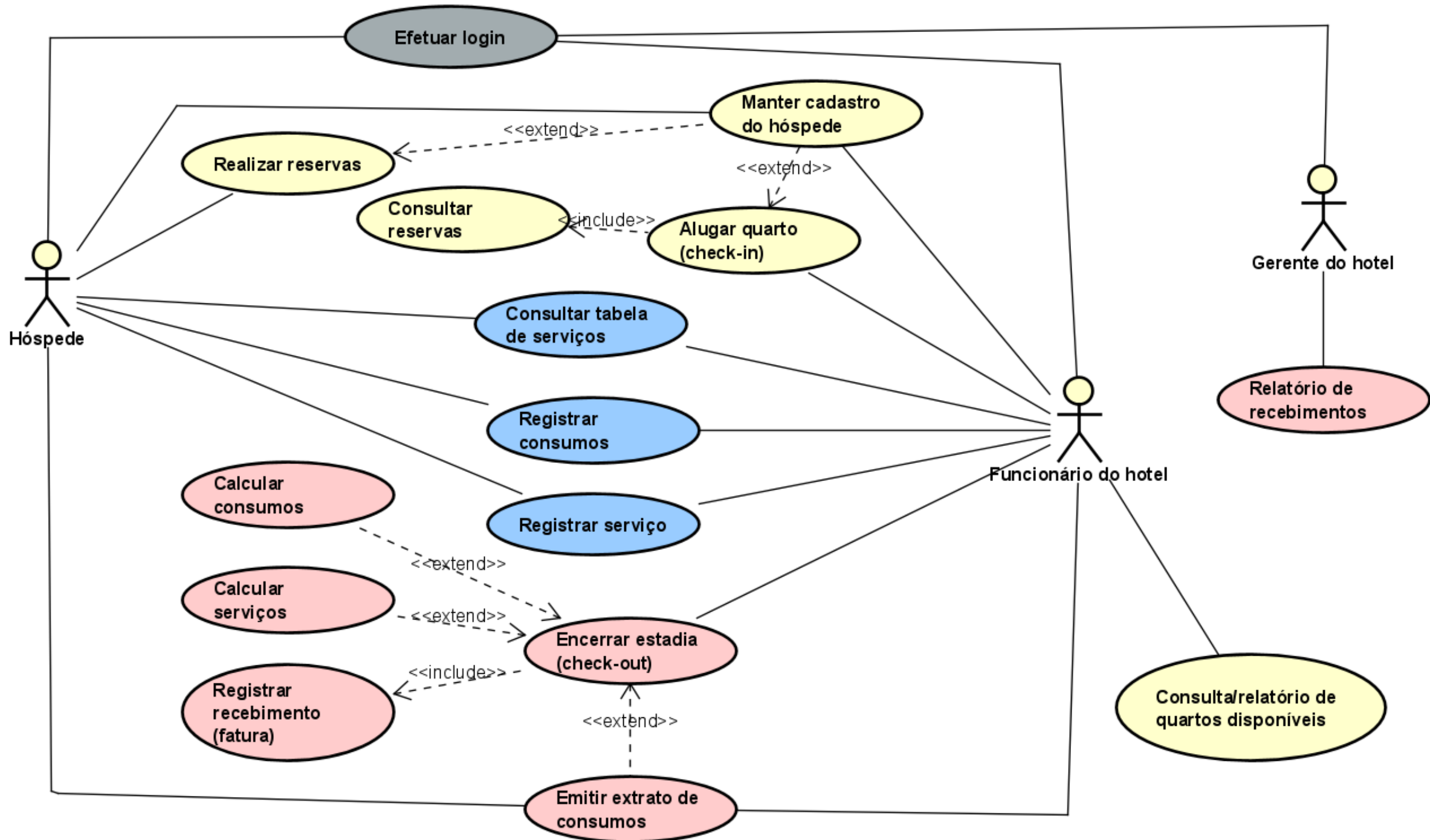


Generalização/
especialização



Exercício de exemplo – Hotel (caso parcial)

- Os hóspedes poderão realizar reservas de quartos via internet para garantir que terá um quarto disponível ao chegarem no hotel
- No momento que o hóspede chega no hotel deverá fazer check-in (alugar quarto)
- Caso seja a primeira vez que está se hospedando neste hotel ou seus dados tenham mudado, então seu cadastrado deve ser atualizado
- O hotel fornece diversos serviços como: estacionamento, lavar e/ou passar roupas, etc . Todos esses serviços serão cobrados na fatura final. O valor é pré-estabelecido pelo hotel através de uma tabela de serviços que contém a descrição dos serviços e o seu preço.
- O hóspede pode também consumir produtos contidos no frigobar ou no restaurante do hotel, que também serão cobrados pelo hotel na fatura final.
- Quando for encerrar sua estadia (fazer check-out), o hóspede deverá quitar (pagar) as diárias (número de dias de aluguel do quarto), serviços e produtos consumidos



Software para modelagem UML

- **Astah** UML versão for estudantes (free):
<https://astah.net/astah-users/students/>