

# Módulo II

Requisitos Arquiteturais e  
Modelagem Arquitetural

Prof. Dr. João Paulo Aramuni

# Introdução à UML



- ❑ 6.2 – Diagramas de Caso de Uso.

## Caso de Uso

Esse diagrama documenta o que o sistema faz do ponto de vista do usuário.

Em outras palavras, ele descreve as principais funcionalidades do sistema e a interação dessas funcionalidades com os usuários do mesmo sistema.

Nesse diagrama não nos aprofundamos em detalhes técnicos que dizem como o sistema faz.

## Caso de Uso



Este artefato é comumente derivado da especificação de requisitos, que por sua vez não faz parte da UML.

Pode ser utilizado também para criar o documento de requisitos.

## Caso de Uso

O diagrama de caso de uso não oferece muitos detalhes — não espere, por exemplo, que ele mostre a ordem em que os passos são executados. Em vez disso, um diagrama de caso de uso adequado dá uma visão geral do relacionamento entre casos de uso, atores e sistemas.

Recomenda-se usar o diagrama de caso de uso para complementar um caso de uso descrito em texto.

# Caso de Uso



Diagramas de Casos de Uso são compostos basicamente por quatro partes:

- ❑ Cenário: Sequência de eventos que acontecem quando um usuário interage com o sistema.
- ❑ Ator: Usuário do sistema, ou melhor, um tipo de usuário.
- ❑ Use Case: É uma tarefa ou uma funcionalidade realizada pelo ator (usuário).
- ❑ Comunicação: é o que liga um ator com um caso de uso.

# Caso de Uso



Exemplo:



Fonte: Devmedia: Guia de UML

## Caso de Uso



Um ator é uma entidade externa ao sistema que de alguma forma participa de um caso de uso.

Um ator pode ser um ser humano, máquinas, dispositivos, ou outros sistemas. Atores típicos são cliente, usuário, gerente, computador, impressora, etc.



## Caso de Uso



Os atores representam um papel e iniciam um caso de uso que após executado, retorna um valor para o ator.

Um caso de uso especifica um serviço que será executado ao usuário e é composto por um ou mais cenários. Um cenário é uma narrativa de uma parte do comportamento global do sistema.

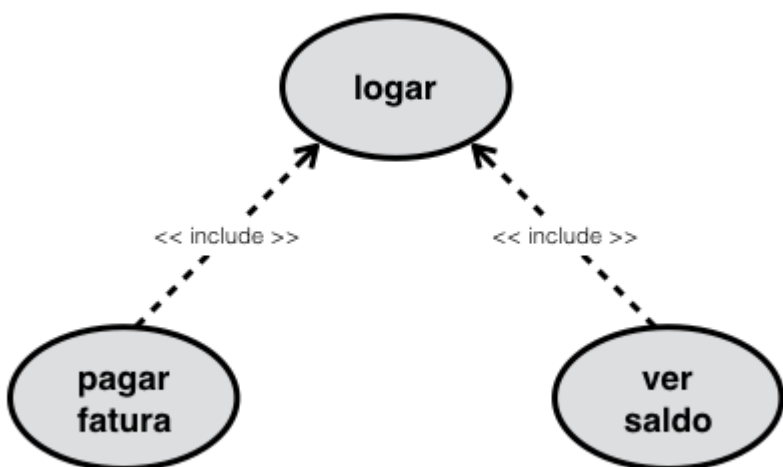
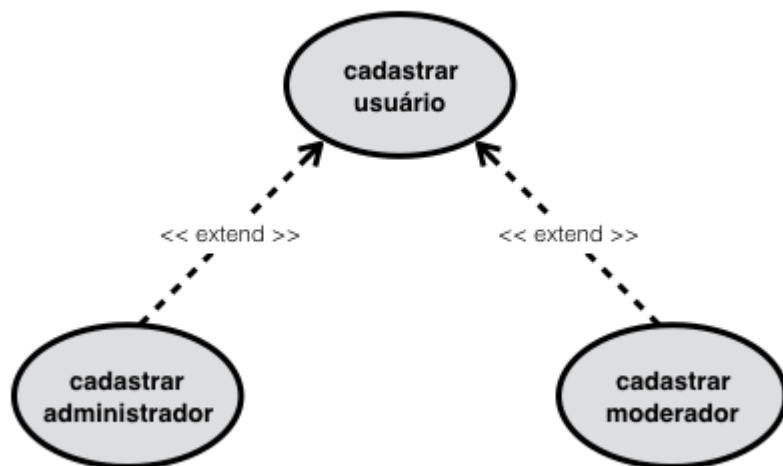
# Caso de Uso



Para melhorar nossos diagramas, podemos utilizar relacionamentos.

**Include** e **extend** são relações entre os casos de uso.

- ☐ Include: seria a relação de um caso de uso que para ter sua funcionalidade executada precisa chamar outro caso de uso.
- ☐ Extend: Esta relação significa que o caso de uso extendido vai funcionar exatamente como o caso de uso base só que alguns passos novos inseridos no caso de uso extendido.
- ☐ Tanto um como o outro, são notados como setas tracejadas com o texto <> ou <>.



Diagramas de caso de uso  
<<extend>> vs <<include>>

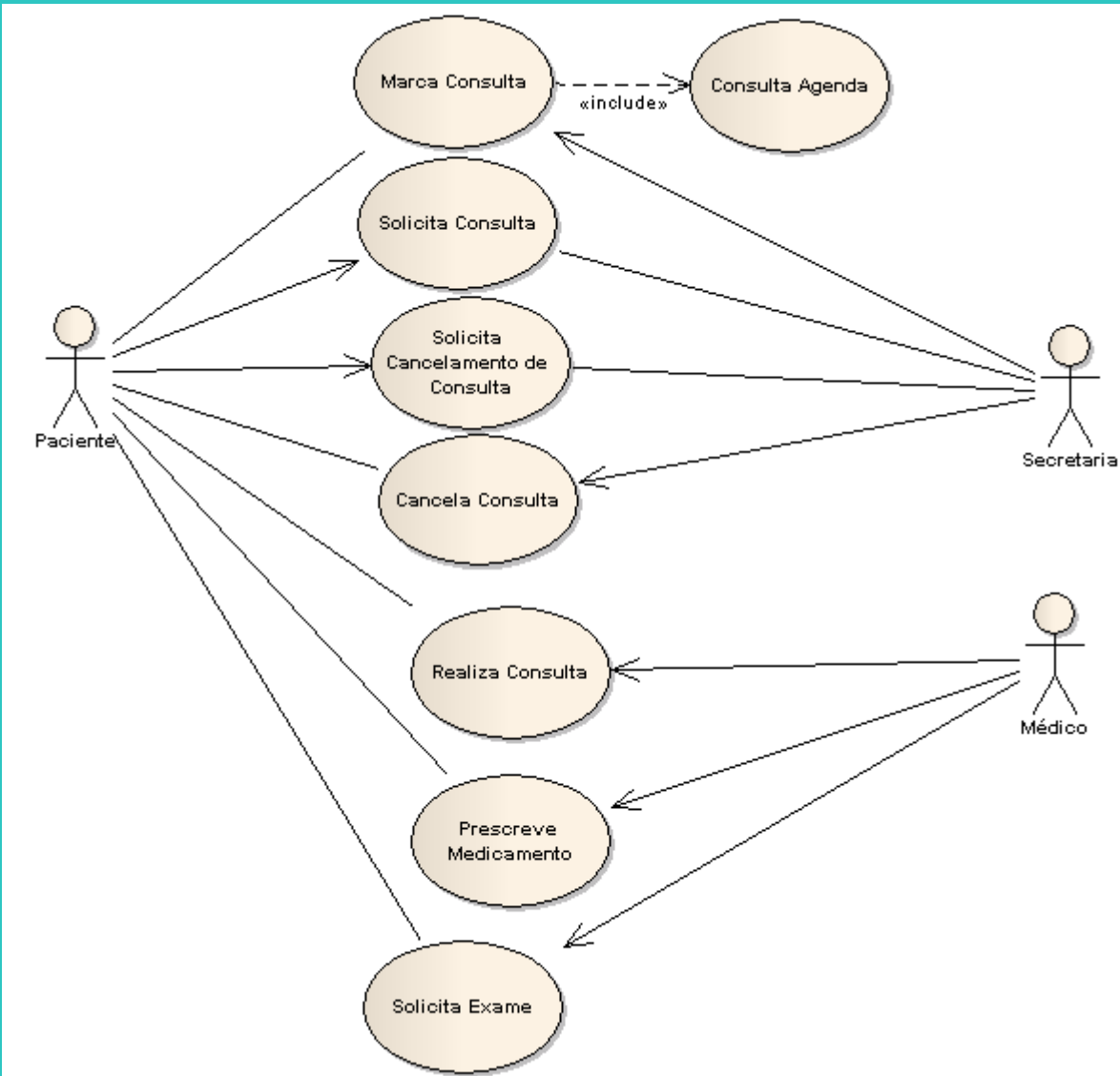


Diagrama de caso de uso

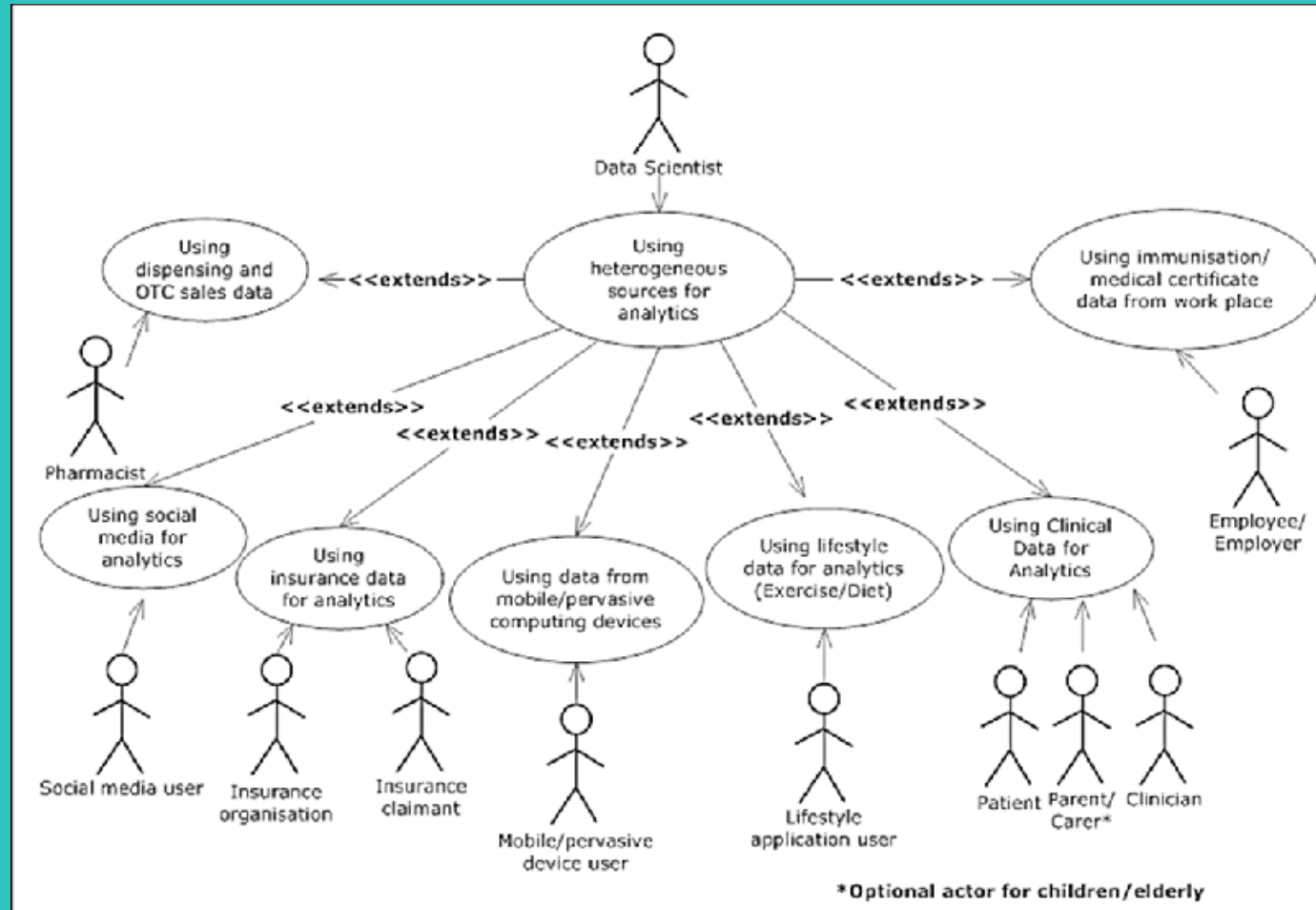


Diagrama de caso de uso

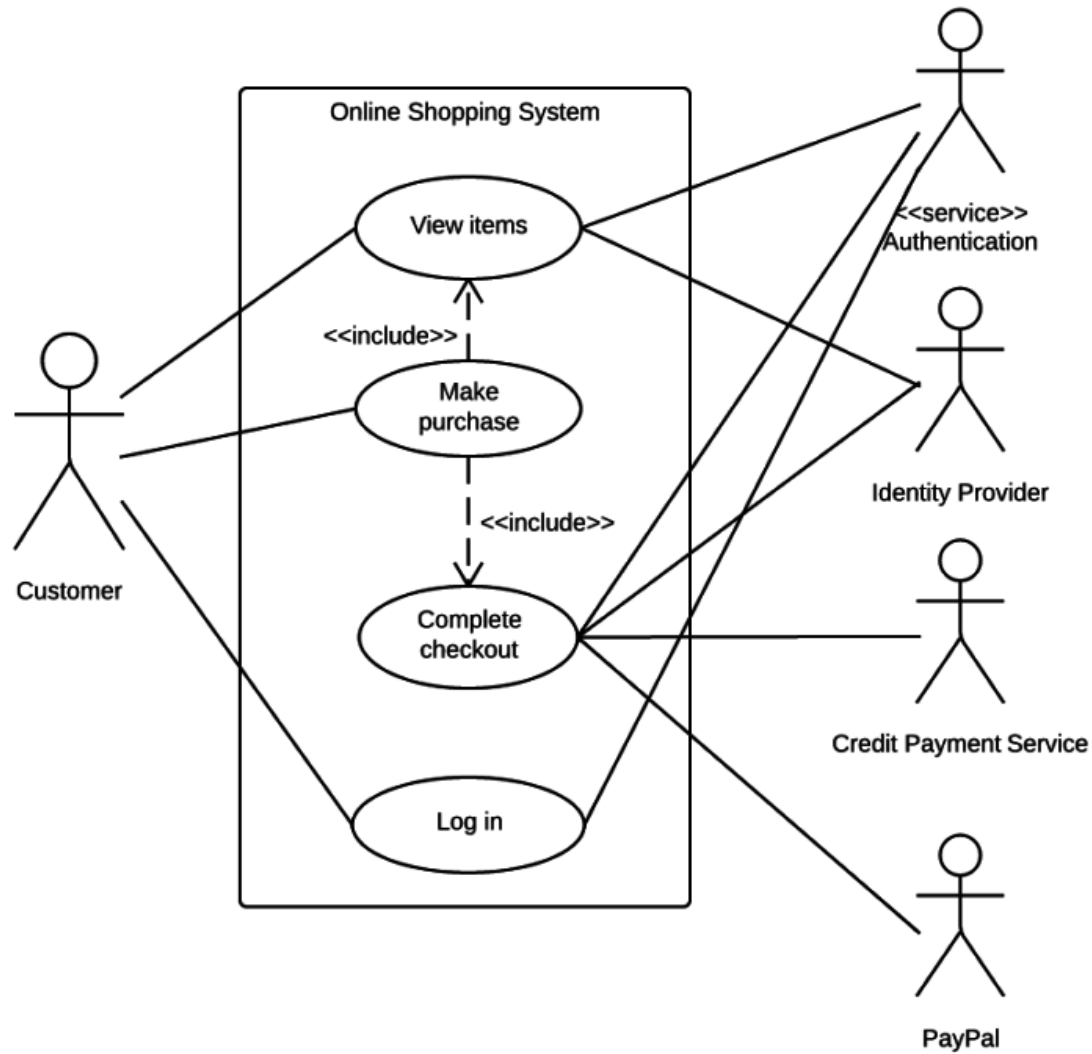


Diagrama de caso de uso

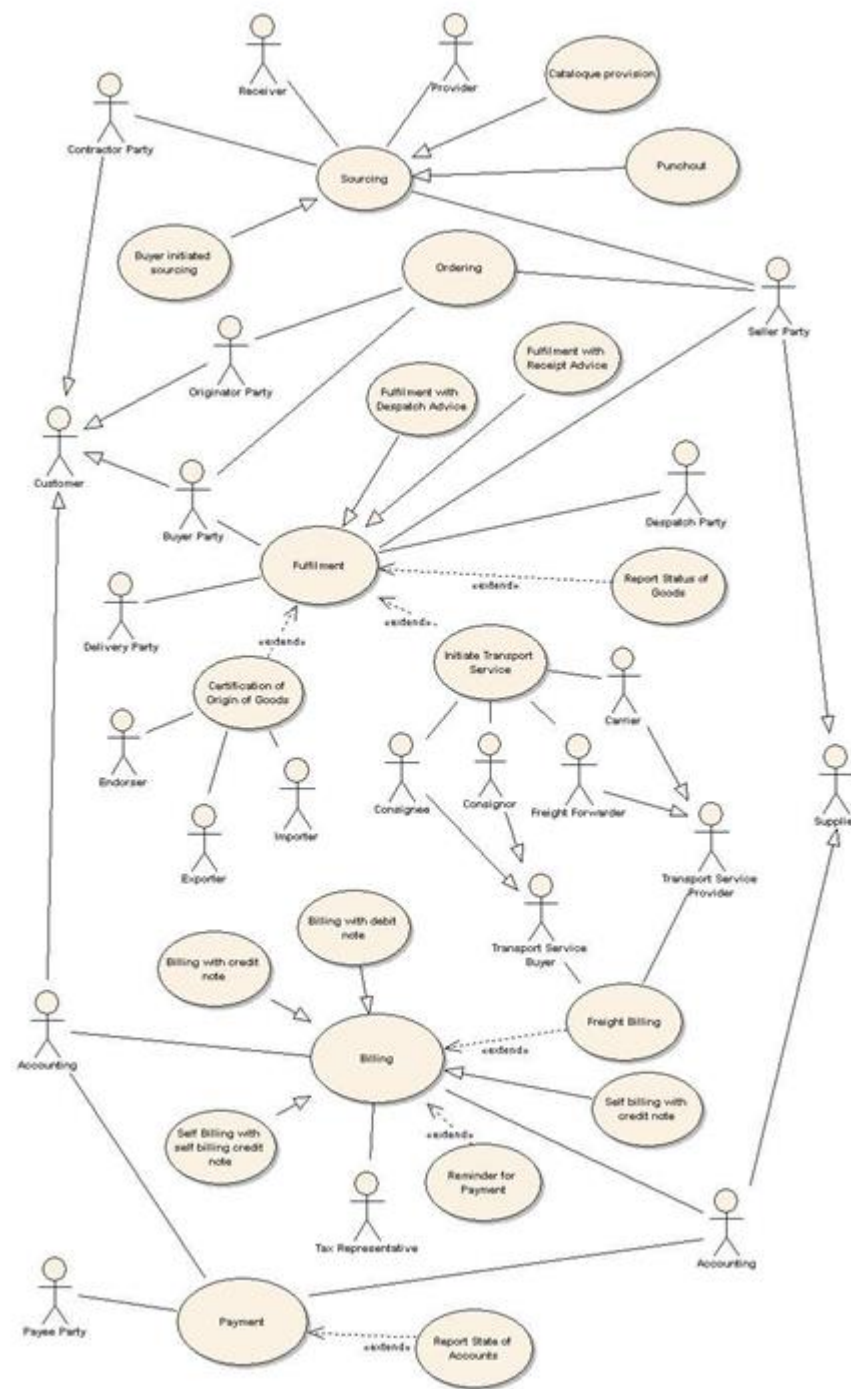


Diagrama de caso de uso

## Caso de Uso



Um bom diagrama de caso de uso ajuda sua equipe a representar e discutir:

- ☐ Cenários em que o sistema ou aplicativo interage com pessoas, organizações ou sistemas externos.
- ☐ Metas que o sistema ou aplicativo ajuda essas entidades (conhecidas como atores) a atingir.
- ☐ O escopo do sistema.




## Caso de Uso



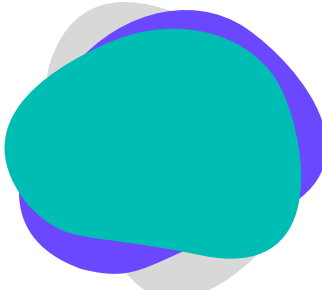

Com os diagramas de caso de uso podemos trabalhar em três áreas muito importantes nos projetos:

- ❑ **Definição de Requisitos** - Novos casos de usos geralmente geram novos requisitos conforme o sistema vai sendo analisado e modelado;
- ❑ **Comunicação com os Clientes** - Pela sua simplicidade, sua compreensão não exige conhecimentos técnicos, portanto o cliente pode entender muito bem esse diagrama, que auxilia o pessoal técnico na comunicação com clientes;
- ❑ **Geração de Casos de Teste** - A junção de todos os cenários para um caso de uso pode sugerir uma bateria de testes para cada cenário.

A decorative abstract shape composed of overlapping teal and purple rounded forms, located to the left of the first text block.

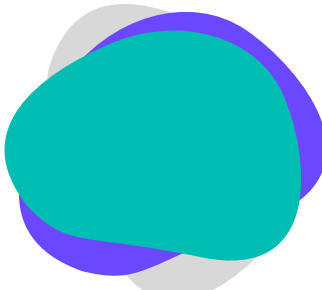
UML para representar as metas de interações entre sistemas e usuários.

UML para definir e organizar requisitos funcionais no sistema.

A decorative abstract shape composed of overlapping teal and purple rounded forms, located to the right of the second text block.A decorative abstract shape composed of overlapping teal and purple rounded forms, located to the left of the third text block.

UML para especificar o contexto e os requisitos do sistema.

UML para modelar o fluxo básico de eventos no caso de uso.

A decorative abstract shape composed of overlapping teal and purple rounded forms, located to the right of the fourth text block.

## Referências



- ❑ PRESSMAN, R. S.; Software Engineering: A Practitioner's Approach, 7 ed., McGraw Hill, 2010.
- ❑ SOMMERVILLE, I.; Software Engineering, 8. ed., Addison-Wesley, 2007.

**Obrigado!**

