

# Modelagem de Casos de Uso com UML na Prática

A modelagem de **casos de uso** utilizando UML é uma técnica fundamental na engenharia de software para representar **requisitos funcionais** de forma visual e intuitiva. Este relatório aborda os conceitos básicos de atores e casos de uso, descreve os elementos e notações principais do diagrama de casos de uso, e explora como essa modelagem contribui para melhorar a comunicação entre a equipe de desenvolvimento e o cliente.

## Conceito de Atores e Casos de Uso

Antes de construir o diagrama em si, é importante entender o que são *atores* e *casos de uso* no contexto de UML:

- **Ator:** Entidade externa ao sistema que interage com ele, assumindo um determinado papel. Um ator pode ser uma pessoa (usuário humano), uma organização ou mesmo outro sistema que se comunica com o sistema em desenvolvimento <sup>1</sup>. Em UML, cada ator representa um papel distinto e tipicamente inicia ações ou eventos no sistema, recebendo as respostas correspondentes do mesmo <sup>2</sup>.
- **Caso de Uso:** Define uma funcionalidade ou serviço específico oferecido pelo sistema que atende a uma meta do usuário. Em essência, um caso de uso descreve uma sequência de interações entre o ator e o sistema para completar um processo que tenha um resultado de valor para o ator <sup>3</sup> <sup>4</sup>. Ou seja, cada caso de uso encapsula um objetivo do usuário e o conjunto de passos necessários para atingi-lo via sistema.

Em resumo, os atores representam *quem* interage com o software, enquanto os casos de uso representam *o quê* o sistema faz em resposta, do ponto de vista de quem está usando. Por exemplo, em um sistema de e-commerce, um ator poderia ser o **Cliente** e um de seus casos de uso seria **"Realizar Compra"**, uma funcionalidade que engloba desde selecionar produtos até efetuar o pagamento.

## Diagrama de Casos de Uso: Elementos e Notações

O **Diagrama de Casos de Uso** é uma representação visual que mostra como os atores se relacionam com as funcionalidades do sistema. Ele apresenta uma visão de alto nível do comportamento do sistema **do ponto de vista do usuário**, descrevendo *o que* o sistema faz (suas principais funções) e **como interage com os usuários**, mas **não detalha como** essas funcionalidades são implementadas internamente <sup>5</sup>. Por isso, é uma ferramenta valiosa nas primeiras etapas do projeto para capturar **escopo e requisitos** de forma simples e compartilhável. Geralmente, os diagramas de caso de uso são elaborados no início do projeto e servem como referência durante todo o desenvolvimento <sup>6</sup>.

Os principais elementos e símbolos em um diagrama de caso de uso incluem atores, casos de uso e o sistema em si, além dos relacionamentos entre esses elementos:

- **Ator:** representado por um *boneco de palito* (figura humana estilizada). Cada ator fica **fora** dos limites do sistema, indicando que ele está externo ao software. O nome do ator (por exemplo,

"Cliente", "Administrador", "Sistema Externo") identifica o papel que ele exerce. Um mesmo ator pode interagir com vários casos de uso diferentes.

- **Caso de uso:** representado por uma **elipse (oval)**, geralmente contendo uma breve descrição textual da funcionalidade (de preferência um verbo + substantivo, por exemplo "Emitir Relatório", "Fazer Login" <sup>7</sup>). Os casos de uso ficam **dentro** dos limites do sistema, indicando que são funcionalidades oferecidas pelo software. Cada caso de uso deve produzir um resultado observável e útil para o ator associado <sup>3</sup>.
- **Sistema:** normalmente indicado por um **retângulo** que delimita o escopo do sistema no diagrama. Tudo que está dentro do retângulo faz parte do sistema; elementos (atores ou casos de uso) posicionados fora dele não pertencem ao escopo do sistema modelado <sup>8</sup>. O retângulo costuma ter o nome do sistema escrito no topo para deixar claro qual sistema está sendo representado.
- **Associação (comunicação):** linha que conecta um ator a um caso de uso, representando que o ator participa ou interage com aquele caso de uso. Essa ligação indica que há troca de informação entre o ator e o caso de uso (por exemplo, o ator inicia o caso de uso ou o caso de uso fornece algum resultado ao ator) <sup>9</sup>. Em suma, a linha de associação mostra quais atores estão ligados a quais funcionalidades.

Além das associações simples entre atores e casos de uso, a UML define **relacionamentos especiais** entre casos de uso para capturar dependências, reutilização de passos e variações de comportamento. Os três principais relacionamentos desse tipo são: **Inclusão**, **Extensão** e **Generalização** <sup>10</sup>:

- **Inclusão** (`<<include>>`): Indica que um caso de uso *incorpora* explicitamente o comportamento de outro caso de uso. É usada para fatorar trechos comuns ou obrigatórios de fluxo de eventos que são compartilhados entre múltiplos casos de uso. Quando um caso de uso A inclui B, significa que A em algum ponto executa o caso de uso B. Por exemplo, no caso de uso "Comprar Item", o passo de pagamento poderia ser extraído como um caso de uso separado "Efetuar Pagamento", que o caso de uso principal *inclui* durante sua execução <sup>11</sup>. O relacionamento de inclusão é representado por uma seta tracejada com o rótulo `<<include>>` apontando para o caso de uso incluído <sup>12</sup>.
- **Extensão** (`<<extend>>`): Indica que um caso de uso *estende* o comportamento de outro caso de uso principal, geralmente para representar uma **variação opcional** ou cenário alternativo. O caso de uso base funciona por si só, mas em determinadas condições extras ele poderá acionar o caso de uso de extensão. Em outras palavras, a extensão adiciona passos adicionais ao fluxo básico somente quando certa condição é atendida <sup>13</sup>. Por exemplo, poderíamos ter o caso base "Realizar Venda" e um caso que o estende "Aplicar Desconto Promocional", que ocorre apenas se houver um cupom de desconto válido. Assim como include, a extensão é denotada por uma seta tracejada rotulada `<<extend>>` ligada do caso de uso opcional ao caso base <sup>14</sup>.
- **Generalização:** Representa uma relação *herança/especialização* entre casos de uso ou entre atores. No caso de **casos de uso**, significa que um caso de uso genérico possui versões especializadas que herdam seu comportamento básico e podem acrescentar ou modificar algo. O caso de uso geral abstrai as partes comuns, e os casos de uso específicos detalham particularidades <sup>15</sup>. Por exemplo, poderíamos ter um caso de uso geral "Pagamento" com dois casos especializados "Pagamento com Cartão" e "Pagamento em Dinheiro", cada um adicionando detalhes próprios, mas ambos compartilhando os passos básicos do pagamento. Já

a **generalização de atores** indica que um ator mais específico herda o papel de um ator mais geral <sup>16</sup>. Por exemplo, um ator "**Administrador**" pode ser uma especialização de "**Usuário**" – o administrador consegue realizar todos os casos de uso que um usuário comum realiza (herdando esse papel) e possui casos de uso adicionais exclusivos.

*Exemplo de diagrama de caso de uso.* O diagrama acima ilustra um sistema simplificado de **reserva de passagens de trem**. Nele, vemos múltiplos atores externos (por exemplo, o *Passageiro* e um *Sistema de Pagamento* externo) interagindo com casos de uso do sistema de reservas. O ator *Passageiro* está associado a casos de uso como **Reservar Passagem** e **Cancelar Passagem**, representando funcionalidades que ele pode acionar. Repare que o caso de uso **Reservar Passagem** inclui sub-funções auxiliares, como **Preencher Dados** do passageiro e **Efetuar Pagamento**, através de relacionamentos `<<include>>` (ou seja, essas etapas fazem parte do fluxo sempre que uma reserva é realizada). Já o cancelamento de passagem aciona o reembolso ao cliente, modelado como o caso de uso **Reembolsar Valor** – uma funcionalidade que é **incluída** no fluxo de cancelar passagem quando aplicável. Esse exemplo demonstra, na prática, o uso de vários atores e diferentes relacionamentos (associações e inclusões) para representar requisitos do mundo real em um diagrama de casos de uso UML.

## Comunicação entre Equipe e Cliente

Um dos maiores benefícios de modelar casos de uso é servir como uma **ponte de comunicação** entre os clientes (ou usuários finais) e a equipe técnica de desenvolvimento. A natureza narrativa e visual dos casos de uso ajuda a garantir que todos tenham **entendimento alinhado** sobre o que o sistema deve fazer. De fato, a construção dos casos de uso normalmente ocorre de forma iterativa, envolvendo discussões entre o cliente e os desenvolvedores até se chegar a uma especificação de sistema da qual **todos estão de acordo** <sup>17</sup>.

Por utilizarem uma linguagem simples e focada no domínio do usuário (evitando detalhes técnicos), os casos de uso podem ser facilmente compreendidos por stakeholders não técnicos. Isso significa que o **cliente consegue ler e entender** os diagramas e descrições de casos de uso sem necessidade de formação em TI <sup>18</sup>. Assim, ele pode validar se as funcionalidades capturadas correspondem realmente aos requisitos desejados. Essa validação antecipada, em conjunto com a equipe, ajuda a evitar divergências de interpretação e reduz falhas de entendimento no levantamento de requisitos.

Em outras palavras, a modelagem de casos de uso funciona como um **contrato** entre os usuários/clientes e a equipe de desenvolvimento sobre as funcionalidades do sistema <sup>19</sup>. Cada caso de uso acordado representa um serviço que o sistema deve fornecer, e todos (tanto cliente quanto desenvolvedores) têm uma referência visual e textual do escopo funcional combinado. Dessa forma, além de documentar requisitos, os casos de uso fomentam uma comunicação clara: o cliente confirma que a equipe entendeu corretamente suas necessidades, e a equipe confirma junto ao cliente que as funcionalidades modeladas cobrem os cenários esperados. Este alinhamento contínuo facilita a validação de funcionalidades ao longo do projeto e aumenta a chance de sucesso do produto final, atendendo às expectativas do cliente.

**Referências:** As informações apresentadas neste relatório foram compiladas de materiais confiáveis de Engenharia de Software e UML, incluindo documentações da IBM <sup>20</sup> <sup>6</sup>, tutoriais especializados (Lucidchart <sup>21</sup> <sup>9</sup>) e literatura acadêmica e de treinamento na área <sup>4</sup> <sup>7</sup> <sup>11</sup>, entre outras fontes citadas ao longo do texto. Essas referências cobrem definições formais de casos de uso e atores, recomendações de notação UML e boas práticas de comunicação de requisitos. Cada citação indicada (por exemplo, <sup>4</sup>) corresponde à localização da informação na fonte consultada, permitindo verificar e aprofundar cada tópico conforme necessário.

---

1 8 9 10 21 Diagrama de caso de uso UML: O que é, como fazer e exemplos | Lucidchart  
<https://www.lucidchart.com/pages/pt/diagrama-de-caso-de-uso-uml>

2 4 7 11 12 13 15 16 17 19 Grátis: Modelando Sistemas em UML - Material Claro e Objetivo em PDF para Estudo Rápido  
<https://www.passeidireto.com/arquivo/35755194/modelando-sistemas-em-uml>

3 6 20 Diagramas de Caso de Uso na Modelagem UML  
<https://www.ibm.com/docs/pt-br/rsm/7.5.0?topic=diagrams-use-case>

5 14 18 O que é UML e Diagramas de Caso de Uso: Introdução Prática à UML  
<https://fabiojoaoanastacio.wixsite.com/minhasaventuras/post/o-que-%C3%A9-uml-e-diagramas-de-caso-de-uso-introdu%C3%A7%C3%A3o-pr%C3%A1tica-%C3%A0-uml>