



Invista em você! Saiba como a DevMedia pode ajudar sua carreira. ▶

O que é UML e Diagramas de Caso de Uso: Introdução Prática à UML

Veja neste artigo um estudo prático sobre UML e uma introdução a um de seus principais diagramas, o diagrama de Casos de Uso.

Artigos > Engenharia... > O que é UML e Diagramas de Caso de Uso: Introdução Prática ...

UML é um acrônimo para a expressão Unified Modeling Language. Pela definição de seu nome, vemos que a UML é uma linguagem que define uma série de artefatos que nos ajuda na tarefa de modelar e documentar os sistemas orientados a objetos que desenvolvemos.

Ela possui nove tipos de diagramas que são usados para documentar e modelar diversos aspectos dos sistemas.

A maioria dos problemas encontrados em sistemas orientados a objetos tem sua origem na construção do modelo, no desenho do sistema. Muitas vezes as empresas e profissionais não dão muita ênfase à essa fase do projeto, e acabam cometendo diversos erros de análise e modelagem. Isso quando há modelagem, pois nós profissionais da área sabemos que muitas vezes o projeto começa já na fase de codificação.



Relacionado: [Curso de UML](#)

Diagrama de Casos 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.

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



- 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

Vamos criar um cenário de exemplo para vermos a notação de um diagrama de caso de uso:

“A clínica médica Saúde Perfeita precisa de um sistema de agendamento de consultas e exames. Um paciente entra em contato com a clínica para marcar consultas visando realizar um check-up anual com seu médico de preferência. A recepcionista procura data e hora disponível mais próxima na agenda do médico e marca as consultas. Posteriormente o paciente realiza a consulta, e nela o médico pode prescrever medicações e exames, caso necessário”.

Saiba mais: [UML](#)

Com esse cenário simples podemos começar a criar nosso diagrama.

Inicialmente vamos definir nossos atores:

1. Paciente
2. Secretária
3. Médico

Agora vamos definir algumas ações de cada usuário:



- Solicita Cancelamento de Consulta

2. Secretária



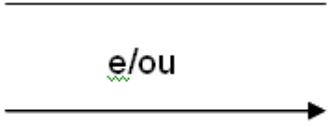
- Consulta Agenda
- Marca Consulta
- Cancela Consulta

3. Médico

- Realiza Consulta
- Prescreve Medicação
- Solicita Realização de exames

Bom, agora já temos uma relação de atores e ações relacionadas a esses atores. Poderíamos criar um documento textual (como foi feito acima), para registrar nossos atores e funcionalidades. Mas o leitor não concorda que uma imagem vale mais que mil palavras? Pois bem, podemos expressar tudo o que definimos em um desenho simples utilizando os padrões da UML para documentação de casos de uso.

No quadro abaixo segue a definição de algumas figuras do diagrama:

Ator	Caso de Uso	Comunicação
		

No mercado existem diversos tipos de ferramentas case que auxiliam na construção de diagramas. o leitor fique a vontade de utilizar a ferramenta de sua



Podemos agora construir o diagrama:



Como podemos observar esse diagrama composto por desenhos simples descrevem de maneira bem objetiva o que textualmente poderia ficar extenso. Nele vemos as funcionalidades do sistema e as interações dos usuários com elas.

Para melhorar um pouco mais esse diagrama vamos ver o conceito de `<>`.

Include e extend são relações entre os casos 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 <> .

Sabendo disso podemos modificar o diagrama inserindo um novo caso de uso “Consultar Agenda”, que será utilizado no caso de uso “Marca Consulta”. Pois a secretária, antes de marcar precisa verificar a disponibilidade da agenda do médico certo?





1. **Definição de Requisitos** - Novos casos de usos geralmente geram novos requisitos conforme o sistema vai sendo analisado e modelado;
2. **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;
3. **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.

Com isso chegamos ao fim desta parte do nosso artigo. Espero que tenham gostado. Por favos peço que deixem seus comentários para que possamos melhorar a qualidade de nossos artigos.

Relacionado: [Diagrama de casos de uso na prática](#)

Links

- Aprenda mais sobre o [Diagrama de Casos de Uso](#)
- Saiba mais sobre [Modelagem e UML](#)

Links Úteis

- [Curso de Python](#) - Neste curso de Python você aprenderá: - Operadores, estruturas condicionais, estruturas de repetição - CGI (criação de programas no lado do servidor web) - Programação orientada a objetos - Interfaces gráficas - tratamento de exceção - criação de jogos - E muitos mais.

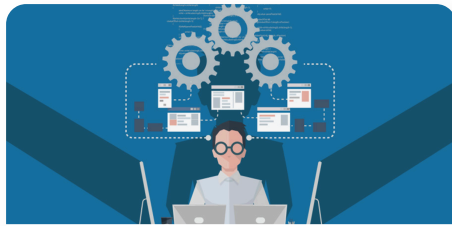


faremos uso de diversos assets disponíveis tanto na Unity Asset Store como no próprio Unity 3D.

- [Curso de Django: Um Poderoso Framework do Python](#) - Neste curso conheceremos o Django, um poderoso framework web escrito em Python que tem chamado muita atenção da comunidade por sua facilidade de implementação, sua organização de código e suas funcionalidades.

Saiba mais sobre UML ;)

- [Modelagem de software com UML](#) - Neste artigo falaremos sobre a importância da modelagem, as atividades gerais do desenvolvimento de software e como utilizar a linguagem UML em cada uma dessas atividades.
- [Curso de UML: Introdução Completa a UML](#) - Este curso de UML nos fornece uma boa bagagem para a pessoa que está começando com análise ou quer aprender um pouco mais sobre documentações. Fornece conhecimentos básicos e intermediários. É o início de tudo.
- [Requisitos, Modelagem e UML](#) - Neste guia você encontrará o conteúdo que precisa para saber como elicitar requisitos, gerenciá-los e modelar o software com as principais técnicas do mercado. Abaixo, confira os posts que te auxiliarão ao longo desse aprendizado.
- [UML Tutorial](#) - A UML (Unified Modeling Language ou Linguagem de Modelagem Unificada) é uma linguagem visual utilizada para modelar sistemas computacionais por meio do paradigma de orientação a objetos – OO. Essa linguagem tornou-se, nos últimos anos, a linguagem padrão de modelagem de software adotada internacionalmente pela indústria de desenvolvimento de software.



Teste de Acessibilidade de
Software



Boas práticas em

TDD

Boas Práticas em TDD



Por **Leandro**

Em 2012

<Formação completa
Programador **FullStack**>



- ✓ +10 mil exercícios gamificados
- ✓ +50 projetos reais
- ✓ Comunidade com + 200 mil alunos
- ✓ Suporte 365 dias do ano
- ✓ 12 meses de acesso

Comece agora

Perguntas Frequentes

Quem somos?

Por que a programação se tornou a profissão mais promissora da atualidade?

Como faço para começar a estudar?

Em quanto tempo de estudo vou me tornar um programador?



O que eu irei aprender estudando pela DevMedia?

Principais diferenciais da DevMedia

Qual o investimento financeiro que preciso fazer para me tornar um programador?

Como funciona a forma de pagamento da DevMedia?

Nossos casos de sucesso

Leonardo Carlos



Eu sabia pouquíssimas coisas de programação antes de começar a estudar com vocês, fui me especializando em várias áreas e ferramentas que tinham na plataforma, e com essa bagagem **consegui um estágio logo no início do meu primeiro período na faculdade.**

Lucas Rodrigues



Estudo aqui na Dev desde o meio do ano passado! Nesse período a Dev me ajudou a crescer muito aqui no trampo.

Fui o primeiro desenvolvedor contratado pela minha empresa. Hoje eu lidero um time de desenvolvimento!

Minha meta é continuar estudando e praticando para ser um Full-Stack Dev!

Heráclito Júnior



**Julio Cablen**

Nossa! Plataforma maravilhosa. To amando o curso de desenvolvimento front-end, tinha coisas que eu ainda não tinha visto. **A didática é do jeito que qualquer pessoa consegue aprender.** Sério, to apaixonado, adorando demais.

Joelberth Sena

Adquiri o curso de vocês e logo percebi que são os melhores do Brasil. É um passo a passo incrível. **Só não aprende quem não quer. Foi o melhor investimento da minha vida!**

Felipe Nunes

Foi um dos melhores investimentos que já fiz na vida e tenho aprendido bastante com a plataforma. Vocês estão fazendo parte da minha jornada nesse mundo da programação, **irei assinar meu contrato como programador graças a plataforma.**

Wanderson Oliveira

Comprei a assinatura tem uma semana, aprendi mais do que 4 meses estudando outros cursos. Exercícios práticos que não tem como não aprender, estão de parabéns!

José Lucas

Obrigado DevMedia, nunca presenciei **uma plataforma de ensino tão presente na vida acadêmica de seus alunos,** parabéns!



Aprender React na plataforma da DevMedia na cerca de 1 ano e meio... **Hoje estou há 1 ano empregado** trabalhando 100% com React!

Adauto Junior



Já fiz alguns cursos na área e **nenhum é tão bom quanto o de vocês**. Estou aprendendo muito, muito obrigado por existirem. Estão de parabéns... Espero um dia conseguir um emprego na área.

Ver todos os casos de sucesso

Menu

Assine agora

Quem somos

FAQ - Fale conosco

Plano para Instituição de ensino

Assinatura para empresas

Política de privacidade

Termos de uso

Política de estorno

DevMedia: 08.401.613/0001-42

Rua Victor Civita, 66 - Salas 306, 307 e 308 -

Jacarepaguá

Rio de Janeiro - RJ, 22775-044



Hospedagem web por Porta 80 Web Hosting.

Tecnologia:

HTML CSS Algoritmo Javascript React React Native Node.js SQL MySQL UML Scrum
Levantamento de Requisitos Padrão de Projeto Teste de Software C# Delphi Dart Java Kotlin
PHP Python TypeScript Angular Vue.js Django Laravel Spring .NET Flutter Modelagem de
Dados Oracle REST PostgreSQL SQL Server MVC Orientação a Objeto Docker Git Scrum

Cursos:

**Artigos:**

Front-End Javascript Iniciantes Angular Dart Engenharia Mobile Node.js Python React Native
Vue.js Android Banco de Dados Delphi Flutter Java Kotlin .Net PHP React Spring
Gratuitos

DevCast:

HTML e CSS Javascript Angular Engenharia Mobile Node.js Python React Native Android
Banco de Dados Delphi Flutter Java Automação .Net PHP React Spring Gratuitos Canal
Mais

Guia:

Fundamentos .NET PHP Python Java Delphi HTML e CSS JavaScript Node React Native
Flutter Banco de Dados Mobile Spring Arquitetura Automação Engenharia + Assuntos