

Módulo II

Requisitos Arquiteturais e
Modelagem Arquitetural

Prof. Dr. João Paulo Aramuni

Introdução à UML



- ❑ 6.1 – O que é UML.

UML

No final dos anos 80 e início dos anos 90, tínhamos muitos conflitos de definições e nomenclaturas na área de modelagem. A escolha para utilização de um determinado padrão era definido mais pelo “gosto” pessoal do que por fatores técnicos oferecidos.

Então, os três mais respeitados nomes nesse campo, cada qual com seu conceito e implementação de modelo, Ivar Jacobson (OOSE – Object Oriented Software Engineering), Grady Booch (The Booch Method) and James Rumbaugh (OMT – Object Modeling Technique) decidiram por fim aos debates e trabalhar juntos na definição de um modelo único que veio a ser a UML – Unified Modeling Language.

UML



A UML permite que você “desenhe” uma “planta” do seu sistema.

Esta “planta” garante, em todas as fases do projeto, seja na definição, desenvolvimento, homologação, distribuição, utilização e manutenção do mesmo, uma maior clareza e objetividade para execução de cada ação, e, com certeza, quanto maior a solução, maior a necessidade de um projeto definido adequadamente.

UML



Desta forma, a UML é uma linguagem padrão para visualização, especificação, construção e documentação de um aplicativo ou projeto de software, e objetiva aumentar a produtividade, otimizar as etapas que envolvem o desenvolvimento de um sistema, aumentando assim a qualidade do produto a ser implementado.

UML



Ela independe da ferramenta em que o aplicativo será desenvolvido.

A ideia é prover uma visão lógica de todo o processo de forma a facilitar a implementação física do mesmo.

UML



A UML disponibiliza, através de conceitos, objetos, símbolos e diagramas, uma forma simples, mas objetiva e funcional, de documentação e entendimento de um sistema.

Você pode utilizar os diagramas e arquivos que compõem um modelo UML para o desenvolvimento, apresentação, treinamento e manutenção durante todo o ciclo de vida da sua aplicação.

UML

Ela é mais completa que outras metodologias empregadas para a modelagem de dados pois, tem em seu conjunto todos os recursos necessários para suprir as necessidade de todas as etapas que compõe um projeto, desde a definição, implementação, criação do modelo de banco de dados, distribuição, enfim, proporcionando sem qualquer outra ferramenta ou metodologia adicional, um total controle do projeto.

UML



A UML implementa uma modelagem com uma visão orientada a objetos. Através dela podemos definir as classes que compõe a nossa solução, seu atributos, métodos e como elas interagem entre si.

Apesar da UML ter como base a orientação a objetos, não significa que a ferramenta e a linguagem utilizada para a implementação do modelo seja também orientada a objetos, embora seja recomendável.

UML



Os diagramas têm como objetivo representar, através de um conjunto de elementos, como o sistema irá funcionar e como cada peça do sistema irá trabalhar e interagir com as outras.

Outra vantagem vem da facilidade de leitura dos diagramas que compõe a UML, além da facilidade de confeccioná-los, pois existem inúmeras ferramentas para modelagem de dados orientados a objetos (ferramentas Case).

Além dos diagramas citados a UML disponibiliza outros diagramas, dentre os quais podemos citar:

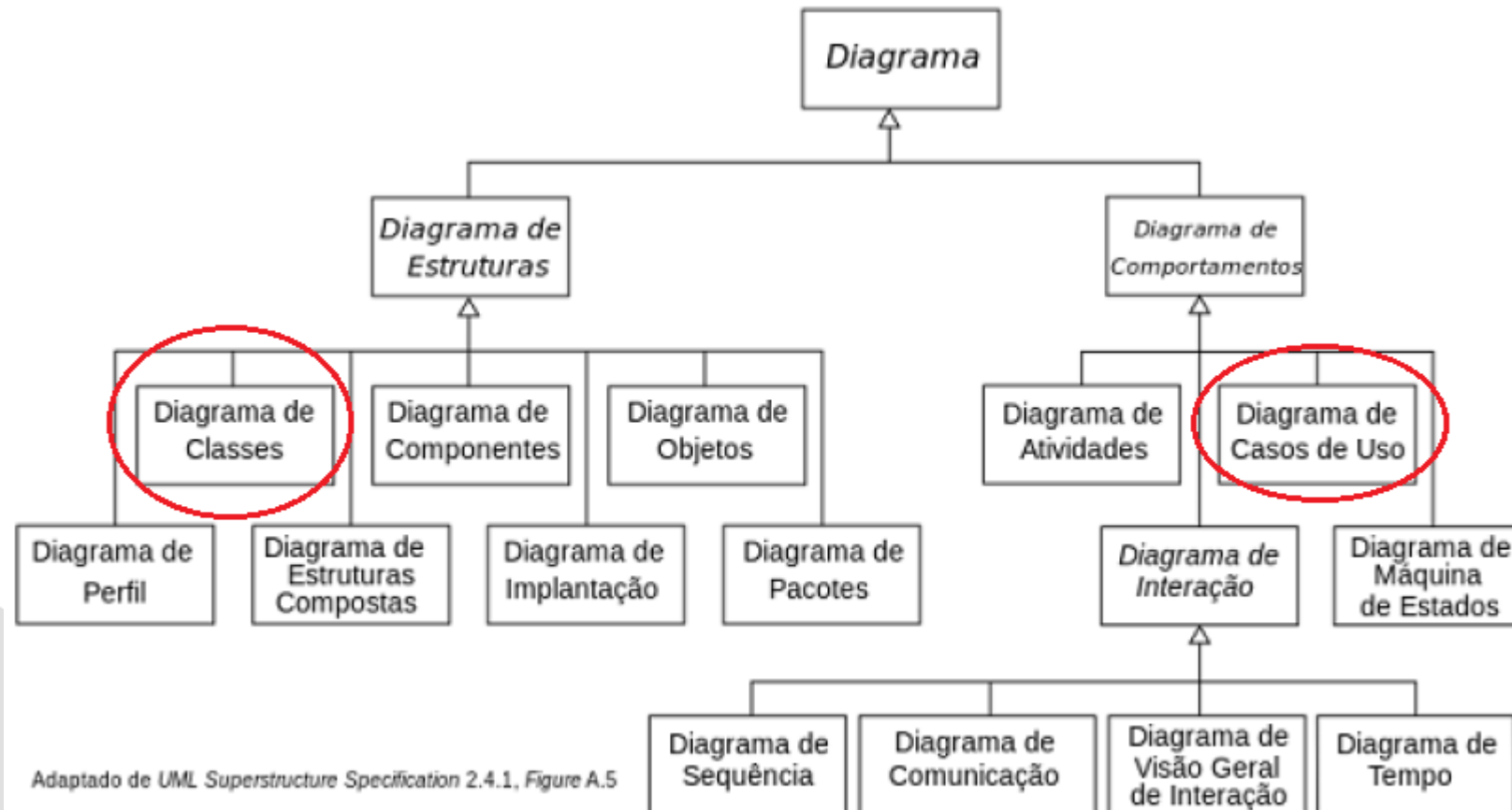
- ☐ Diagrama de Objetos,
- ☐ Diagrama de Sequência,
- ☐ Diagrama de Colaboração,
- ☐ Diagrama de Estado,
- ☐ Diagrama de Atividade e
- ☐ Diagrama de Componentes.



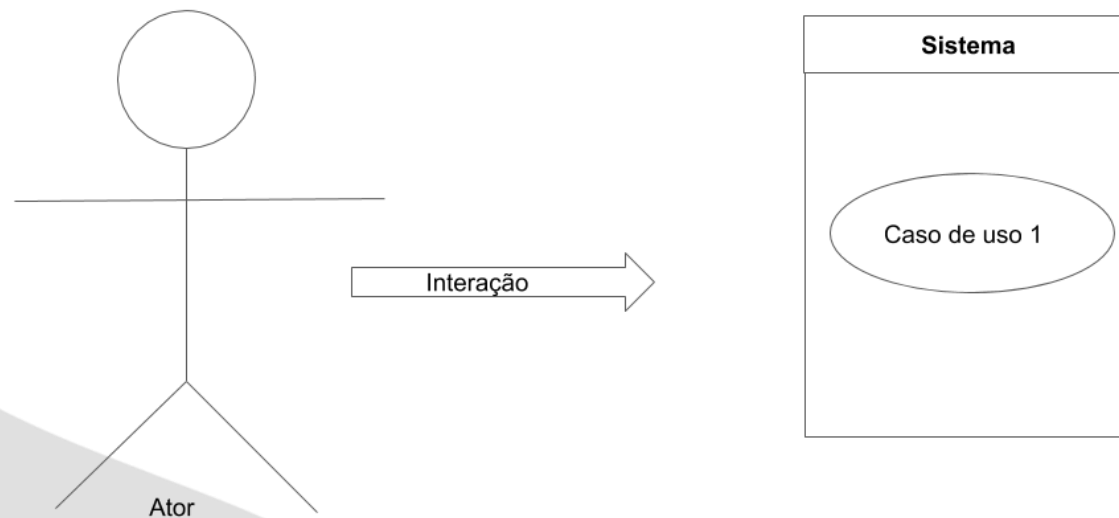
UML



Dado que estão entre os diagramas mais utilizados no mercado, vamos conhecer agora os elementos básicos de um diagrama de caso de uso e de um diagrama de classe.



Elementos básicos de um diagrama de **Caso de Uso**:

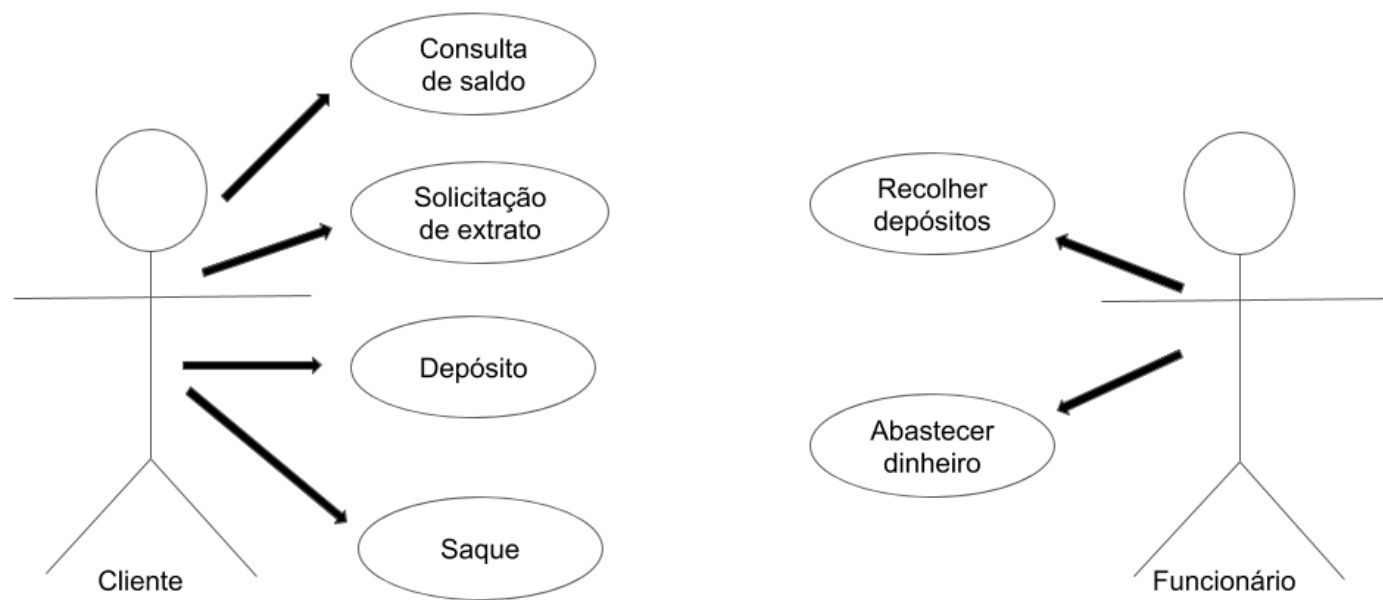


Elementos básicos de um diagrama de **Classe**:

Nome da classe
Atributo -nome do atributo - tipo de dado -nome do atributo - tipo de dado = valor inicial
Operação +Operação (lista de argumentos): lista do resultado

UML

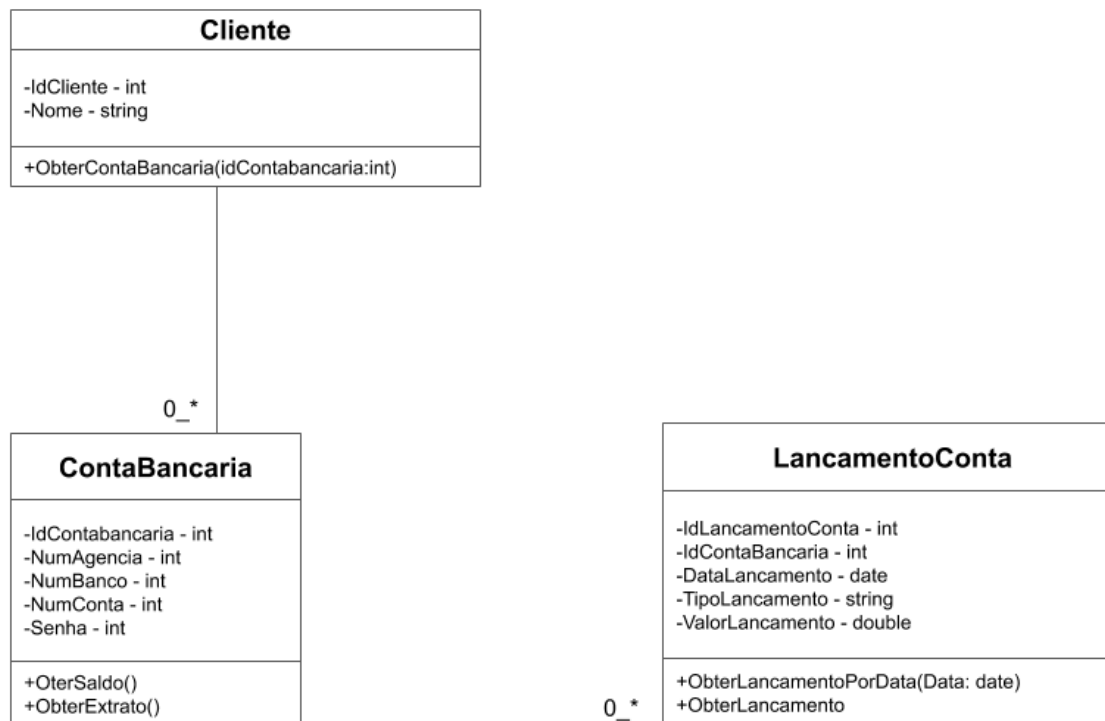
Exemplo de diagrama de caso de uso:



UML



Exemplo de diagrama de classe:



UML



Veremos detalhes da construção de casos de uso no próximo tópico
“Diagramas de Caso de Uso”.

UML



A UML se mostra como parte essencial no “ciclo de vida” de uma aplicação.

Foi mostrado, utilizando apenas dois diagramas, toda a funcionalidade operacional de um sistema bem como a definição de elementos internos referentes ao desenvolvimento do mesmo.

UML



Esses dois diagramas fazem parte da documentação do sistema, e podem ser utilizados para uma apresentação da solução para o requisitante antes da implementação da mesma, visualização dos processos que o sistema irá disponibilizar, definição de elementos inerentes ao desenvolvimento como estrutura de telas, procedimentos operacionais, referência para criação de objetos de persistência em um banco de dados, etc.

Diagrama	Função
Diagrama de Classes	Permite a visualização de um conjunto de classes, detalhando atributos e operações (métodos) presentes nesta última, assim como prováveis relacionamentos entre essas estruturas. Este tipo de representação pode incluir ainda definições de interfaces.
Diagrama de Componentes	Apresenta diferentes componentes de um sistema, além de possíveis dependências entre tais elementos. A ideia de componente refere-se a uma parte (ou até mesmo um módulo) de uma aplicação, englobando assim uma série de outras estruturas relacionadas (como classes, interfaces etc.).
Diagrama de Pacotes	Descreve as dependências entre diferentes namespaces/pacotes que compõem uma aplicação. Dentro da plataforma .NET, um namespace costuma conter classes, interfaces e outros elementos, atuando como uma forma de agrupamento lógico destes elementos.
Diagrama de Objetos	Apresenta o estado de instâncias de objetos dentro de um sistema, levando em conta para isto um intervalo de tempo específico.
Diagrama de Estrutura Composta	Utilizado para demonstrar a estrutura interna de uma classe, incluindo referências que apontam para outras partes de um sistema.
Diagrama de Instalação	Empregado para demonstrar a estrutura de hardware adotada para a implantação de uma aplicação em um ambiente. Pode envolver dispositivos como servidores de aplicação, servidores de banco de dados, terminais de usuários etc.
Diagrama de Perfil	Possibilita a definição de novos elementos UML, permitindo assim estender os diagramas existentes com a inclusão de estruturas customizadas para uma determinada necessidade.



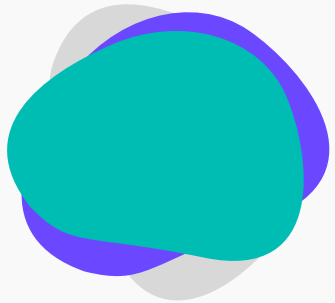
Diagramas Estruturais da UML

Diagrama	Função
Diagrama de Casos de Uso	Voltado à apresentação de funcionalidades e características de um sistema, assim como de que forma tais elementos se relacionam com usuários e entidades externas envolvidas num determinado processo.
Diagrama de Atividades	Contempla as diversas tarefas desempenhadas na execução de uma atividade, sendo utilizado geralmente na representação de processos dentro de uma empresa/organização.
Diagrama de Transição de Estados	Detalha os diferentes estados pelos quais pode passar um objeto, tomando por base a execução de um processo dentro do sistema que se está considerando.

Diagrama	Função
Diagrama de Sequência	Demonstra as interações entre diferentes objetos na execução de uma operação, destacando ainda a ordem em que tais ações acontecem num intervalo de tempo. A sequência em que as diversas operações são executadas ocorre na vertical, de cima para baixo.
Diagrama de Interatividade	Espécie de representação híbrida, com uma estrutura similar à de diagramas de atividade. O que diferencia este tipo de representação está justamente no fato do equivalente a uma atividade ser representada por outro diagrama, sendo o de sequência um exemplo de uso válido neste último caso.
Diagrama de Colaboração ou Comunicação	Similar aos diagramas de sequência, é também empregado na modelagem de interações entre vários objetos dentro de um determinado contexto. Este tipo de representação difere de um diagrama de sequência por não possuir uma estrutura rígida para demonstrar a comunicação entre objetos, ou seja, estes elementos podem ser dispostos na melhor ordem que se julgar necessária, sem a obrigatoriedade de exibir as diferentes operações na vertical uma após a outra.
Diagrama de Tempo	Corresponde a um tipo específico de diagrama de sequência, descrevendo mudanças de estado e interações entre objetos dentro de intervalos de tempo tomados como parâmetro.

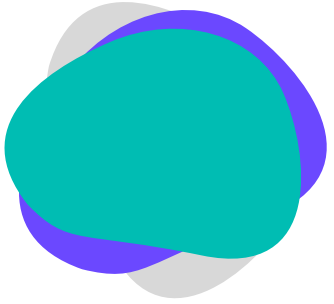



Diagramas de Interação da UML

A decorative abstract shape composed of overlapping teal and purple rounded rectangles with soft shadows.

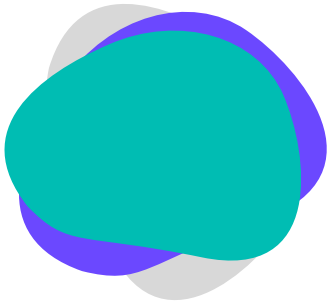
A UML foca na representação visual de diferentes elementos e aspectos de um software.

UML é uma linguagem que se presta à modelagem de estruturas que irão compor uma aplicação.

A decorative abstract shape composed of overlapping teal and purple rounded rectangles with soft shadows.A decorative abstract shape composed of overlapping teal and purple rounded rectangles with soft shadows.

Em termos práticos, a UML contempla uma série de notações para a construção de diagramas.

Sistemas construídos nas mais variadas linguagens podem se beneficiar das vantagens decorrentes do uso desta linguagem.

A decorative abstract shape composed of overlapping teal and purple rounded rectangles with soft shadows.

Referências



Básica:

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

Complementar:

- ❑ Revista SQL Magazine Edição 1.

Obrigado!

