

EDSII – UML

# EDS – Diagramas UML

profa.Denise Neves



# UML

- **O que é uma linguagem de modelagem**

Uma linguagem de modelagem é uma notação gráfica para descrever projeto de software.

A linguagem inclui várias regras para distinguir entre desenhos corretos e incorretos.

Uma linguagem de modelagem não é igual a uma metodologia.

Uma metodologia diz a você como projetar o software. Em vez disso, uma linguagem de modelagem representa graficamente os produtos do projeto que você criará enquanto segue uma metodologia.

# UML

- **O que é**

A UML é uma linguagem visual para modelagem de sistemas através do uso de diagramas .

A UML é uma linguagem de modelagem padrão. A linguagem consiste em várias notações gráficas que você pode usar para descrever a arquitetura inteira de seu software.

# UML

- **UML** é uma abreviação para Unified Modeling Language (Linguagem de Modelagem Unificada). Cada uma destas palavras transmite um aspecto importante da UML.
- **Linguagem:** uma linguagem permite transmitir informação sobre a realidade. No sentido mais amplo, as linguagens não usam somente palavras escritas. A UML é uma linguagem para especificar, visualizar, construir e documentar artefatos de um sistema.

Artefatos são produtos de trabalho que são produzidos e usados dentro de um processo de desenvolvimento, incluindo documentação para comunicação entre as equipes que trabalham em um sistema e o próprio sistema físico.

# UML

- **Modelo:** um modelo é a representação de uma certa realidade.

Sem um modelo é muito difícil para os membros de uma equipe ter um entendimento comum dos requisitos e do próprio sistema. Por exemplo, avaliar o impacto das mudanças que ocorrem enquanto o sistema está sendo desenvolvido.

- **Unificada:** o termo “unificada” refere-se ao fato de que o OMG – Object Management Group, uma organização de padronização reconhecida internacionalmente, e a empresa Rational Software Corporation, criaram a UML com o objetivo de unir as melhores práticas de engenharia da indústria de tecnologia e de sistemas de informação. Essas práticas envolvem aplicação de técnicas que nos permitem ter mais sucesso no desenvolvimento de sistemas.

# UML

- **Histórico**

A UML 1.0 surgiu entre os anos de 1990 e 1997. James Rumbaugh e mais tarde Ivar Jacobson juntaram-se a Grady Booch na Rational Software Corporation para unificar seus enfoques. Devido ao esforço da unificação eles se tornaram conhecidos como os três amigos. Junto com várias outras organizações, a Rational Software Corporation formou o grupo de parceiros UML e estes submeteram a versão 1.0 da UML para a OMG.

# UML X Processo de desenvolvimento de sistemas

O processo de desenvolvimento de sistema envolve os seguintes tipos de atividades:

1. Atividades de requisitos para capturar características que definam o que o sistema deve fazer.
2. Atividades de análise para entender os requisitos.
3. Atividades de projeto para determinar como um sistema satisfará seus requisitos.
4. Atividades de implementação para construir um sistema.
5. Atividades de teste para verificar se um sistema satisfaz seus requisitos.
6. Atividades de implementação para tornar o sistema disponível para seus usuários.

# Os diagramas da UML

O objetivo é fornecer múltiplas visões do sistema a ser modelado, analisando-o sob diversos aspectos, procurando-se, assim, atingir a completitude da modelagem, permitindo que cada diagrama complete o outro.

- Diagrama de Casos de uso
- Diagrama de Classes
- Diagrama de Objetos
- Diagrama de Seqüência
- Diagrama de Comunicação/Colaboração
- Diagrama de Atividades
- Diagrama Máquina de Estados
- Diagrama de Pacotes
- Diagrama de Componentes
- Diagrama de Implantação



# UML

- O diagrama de objetos é também um diagrama importante na UML, mas não é um dos nove diagramas chave. Os dez diagramas são divididos em três categorias:
- **Estáticos:** um diagrama estático mostra a estrutura do sistema e as suas funcionalidades.  
Os diagramas estáticos são: classes, objetos e casos de uso.
- **Dinâmicos:** um diagrama dinâmico mostra as interações que o sistema suporta.  
Os diagramas dinâmicos detalham a interação entre os componentes estruturais definidos a partir dos diagramas estáticos. Assim, os diagramas dinâmicos fornecem uma representação mais clara do comportamento do sistema. Os diagramas dinâmicos são: colaboração, seqüência, atividades, estados.

# UML

- **Arquiteturais:** um diagrama arquitetural mostra a organização em seu sistema dos componentes executáveis.

Os diagramas arquiteturais distinguem a localização física de execução desses componentes e os nós de armazenamento com os quais eles podem interagir. Eles são produzidos no início da fase de desenvolvimento do sistema e são atualizados durante o projeto para indicar a arquitetura física pretendida. Os diagramas arquiteturais são: componentes, pacotes e implantação.

# UML

Ferramentas Case (computer-aided software engineering)

são softwares que, de alguma maneira, colaboram para a execução de uma ou mais atividades realizadas durante o processo de engenharia de software. Auxiliam o desenvolvedor na construção dos diagramas.

Abaixo são listadas algumas das ferramentas de diagramação UML mais usuais:

- Rational Rose
- ARGOUML
- Poseidon for UML
- DIA
- ASTAH
- Enterprise Architect