

UNIP
UNIVERSIDADE PAULISTA

Linguagem de Programação Orientada a Objeto

LPOO - 00.3 - UML - Diagrama Classe

Prof. Msc Wanderley Gonçalves Freitas
Wanderley.unip@gmail.com
www.professor.wanderley.nom.br

Prof: Msc Wanderley Gonçalves Freitas LPOO - 00.3 - UML - Diagrama Classe 1 de 44

UNIP
UNIVERSIDADE PAULISTA

Classe

- Classes são o elemento mais importante de qualquer sistema orientado a objetos
- Uma classe é uma descrição de um conjunto de objetos com os mesmos atributos, relacionamentos, operações e semântica
- Classes são usadas para capturar o vocabulário de um sistema
- Classes são abstrações de elementos do domínio do problema, como “Cliente”, “Banco”, “Conta”

Prof: Msc Wanderley Gonçalves Freitas LPOO - 00.3 - UML - Diagrama Classe 2 de 44

UNIP
UNIVERSIDADE PAULISTA

Classe - Notação básica

Atributos (opcional)

Cliente	
- CPF : long	Nome (obrigatório)
- Nome : char	
- Endereço : char	
+ Consultar(CPF : long) : int	Operações (opcional)

Prof: Msc Wanderley Gonçalves Freitas LPOO - 00.3 - UML - Diagrama Classe 3 de 44

UNIP
UNIVERSIDADE PAULISTA

Atributos

- Um atributo representa alguma propriedade do que está sendo modelado, que é compartilhada por todos os objetos da classe
- Os atributos descrevem os dados contidos nas instâncias de uma classe
- Os atributos descrevem as características dos objetos
- Em um momento dado, um objeto de uma classe conterá valores para todos os atributos descritos na sua classe

Prof: Msc Wanderley Gonçalves Freitas LPOO - 00.3 - UML - Diagrama Classe 4 de 44

UNIP
UNIVERSIDADE PAULISTA

Atributos - Notação

- Atributos podem ter seus tipos (ou classes) especificados e terem valores padrão definidos

Parede	
altura : int	
largura : real	
espessura : real	
viga : boolean = false	

Prof: Msc Wanderley Gonçalves Freitas LPOO - 00.3 - UML - Diagrama Classe 5 de 44

UNIP
UNIVERSIDADE PAULISTA

Operações

- Operações são o meio de alterar os valores dos atributos
- Uma operação é uma abstração de alguma coisa que se pode fazer com um objeto e que é compartilhada por todos os objetos da classe
- Um classe pode ter qualquer número de operações, inclusive nenhuma

Prof: Msc Wanderley Gonçalves Freitas LPOO - 00.3 - UML - Diagrama Classe 6 de 44

Operações - Notação

- Como para os atributos, você pode especificar uma operação apenas com seu nome

Retângulo

mover()
aumentar()
diminuir()

- Você pode também especificar a assinatura da operação: seus parâmetros, o tipo desses parâmetros e o tipo de retorno

Prof: Msc Wanderley Gonçalves Freitas LPOO - 00.3 - UML - Diagrama Classe 7 de 44

Visibilidade

- Você pode usar marcações de acesso para especificar o tipo de acesso permitido aos atributos e operações
- Classificador pode ser classes, interfaces, componentes, nós, use cases, subsistemas
 - + público: todos os classificadores podem usar
 - # protegido: qualquer descendente do classificador poderá usar
 - privado: somente o próprio classificador poderá usar

Cliente

- CPF : long
+ Nome : char
- Endereço : char
+ Consultar(CPF : long) : int

Prof: Msc Wanderley Gonçalves Freitas LPOO - 00.3 - UML - Diagrama Classe 8 de 44

Relacionamentos

- Poucas classes vivem sozinhas
- Tipos de relacionamentos especialmente importantes na modelagem orientada a objetos:
 - Associações
 - Agregação
 - Composição
 - Dependências
 - Generalizações
 - Realização

Prof: Msc Wanderley Gonçalves Freitas LPOO - 00.3 - UML - Diagrama Classe 9 de 44

Relacionamentos

- Os relacionamentos ligam as classes/objetos entre si criando relações lógicas entre estas entidades.
- Os relacionamentos podem ser dos seguintes tipos:
 - Associação** - especifica que objetos de um elemento (classe) estão conectados a objetos de outros elementos
 - Agregação** - relacionamento fraco do tipo “é parte de”. É um tipo especial de associação
 - Composição** - relacionamento forte do tipo “é parte de”. A composição entre um elemento (o “todo”) e outros elementos (“as partes”) indica que as partes só existem em função do “todo”.

Prof: Msc Wanderley Gonçalves Freitas LPOO - 00.3 - UML - Diagrama Classe 10 de 44

Relacionamentos

- Dependência** - relacionamento de uso, no qual uma mudança na especificação de um elemento pode alterar a especificação do elemento dependente
- Generalização (herança)** - relacionamento entre descrições mais gerais e descrições mais específicas, com mais detalhes sobre alguns dos elementos gerais
- Realização** - relacionamento entre uma interface e o elemento que a implementa

Prof: Msc Wanderley Gonçalves Freitas LPOO - 00.3 - UML - Diagrama Classe 11 de 44

Relacionamentos - Notação

Associação
Sem/com navegação

Dependência

Agregação

Generalização

Composição

Realização

Prof: Msc Wanderley Gonçalves Freitas LPOO - 00.3 - UML - Diagrama Classe 12 de 44

Associação

- A associação é um relacionamento estrutural que especifica que objetos de um elemento estão conectados a objetos de outro elemento

```

classDiagram
    class Companhia
    class Funcionario
    Companhia -- "Funcionario" : emprega
    
```

Prof: Msc Wanderley Gonçalves Freitas LPOO - 00.3 - UML - Diagrama Classe 13 de 44

Multiplicidade

Multiplicidade	Significado
0..1	No mínimo zero e no máximo um. Indica que os objetos das classes não precisam obrigatoriamente estar relacionados mas, se houver, apenas uma instância da classe se relaciona.
1..1	Um e somente um. Indica que apenas um objeto da classe se relaciona com a outra classe.
0..*	No mínimo nenhum e no máximo muitos. Indica que pode ou não haver instâncias da classe participante do relacionamento.
*	Muitos. Indica que muitos objetos da classe estão envolvidos.
1..*	No mínimo um e no máximo muitos. Indica que há pelo menos um objeto envolvido no relacionamento, podendo haver muitos.
3..5	No mínimo três e no máximo cinco.

- É a cardinalidade de uma associação

Prof: Msc Wanderley Gonçalves Freitas LPOO - 00.3 - UML - Diagrama Classe 14 de 44

Associação Unária

- Quando há um relacionamento de uma classe para consigo própria

```

classDiagram
    class Funcionario
    Funcionario -- "1..*" Funcionario : gerencia
    
```

Prof: Msc Wanderley Gonçalves Freitas LPOO - 00.3 - UML - Diagrama Classe 15 de 44

Associação Binária

- Quando há duas classes envolvidas na associação de forma direta de uma para a outra

```

classDiagram
    class Cliente
    class Pedido
    Cliente -- "0..*" Pedido : faz
    
```

Prof: Msc Wanderley Gonçalves Freitas LPOO - 00.3 - UML - Diagrama Classe 16 de 44

ASSOCIAÇÃO TERNÁRIA

- Associações que conectam mais de duas classes
- São representadas por um losângulo para onde convergem todas as ligações de associação
- (No Astah representado por um círculo-"interface")
- Úteis para demonstrar associações complexas
- Devem ser evitadas – difíceis de interpretar

Prof: Msc Wanderley Gonçalves Freitas LPOO - 00.3 - UML - Diagrama Classe 17 de 44

ASSOCIAÇÃO TERNÁRIA

```

classDiagram
    class Professor
    class Turma
    class SalasdeAula
    Professor -- Turma : - Leciona
    Professor -- Turma : - possui2
    Professor -- SalasdeAula : Utiliza
    
```

Prof: Msc Wanderley Gonçalves Freitas LPOO - 00.3 - UML - Diagrama Classe 18 de 44

Associação: Navegabilidade

Em geral a navegação entre as classes de uma associação é bi-direcional
 Porém é possível limitá-la a apenas uma direção

```

classDiagram
    class Usuário
    class Senha
    Usuário --> Senha
    
```

Prof: Msc Wanderley Gonçalves Freitas LPOO - 00.3 - UML - Diagrama Classe 19 de 44

Associação: papéis

- Papéis: um dos lados da associação
- Nomes de papéis são necessários para associação entre dois objetos da mesma classe

```

classDiagram
    class Companhia
    class Empregado
    Companhia "1" -- "1..*" Empregado : +empregador
    Empregado "1" -- "0..*" Empregado : +chefe, +subordinado
    
```

Prof: Msc Wanderley Gonçalves Freitas LPOO - 00.3 - UML - Diagrama Classe 20 de 44

Associação com Atributos

- Modela as propriedades associadas com uma associação
- As propriedades devem ser representadas por uma classe

```

classDiagram
    class Companhia
    class Empregado
    class Trabalho {
        descrição
        salário
    }
    Companhia "*" -- "1..*" Empregado
    Companhia -- Trabalho
    Empregado -- Trabalho
    
```

Prof: Msc Wanderley Gonçalves Freitas LPOO - 00.3 - UML - Diagrama Classe 21 de 44

Agregação

- Uma forma especial de associação entre o todo e suas partes, no qual o todo é composto de partes
- Não impõe que a vida das "Partes" esteja relacionado com a vida do "Todo"

```

classDiagram
    class Pedido {
        -NroPedido : long
        -DiaPedido : Date
        -DataEntPedido : Date
    }
    class ItensPedido {
        - Qtd Itens : double
        - ValUnitItem : double
    }
    class Produto {
        - DescProd : char
        - QtdProd : int
        - ValProd : double
    }
    Pedido o-- "1..*" ItensPedido : Possui
    ItensPedido o-- "1..*" Produto : Compo
    
```

Prof: Msc Wanderley Gonçalves Freitas LPOO - 00.3 - UML - Diagrama Classe 22 de 44

Agregação

```

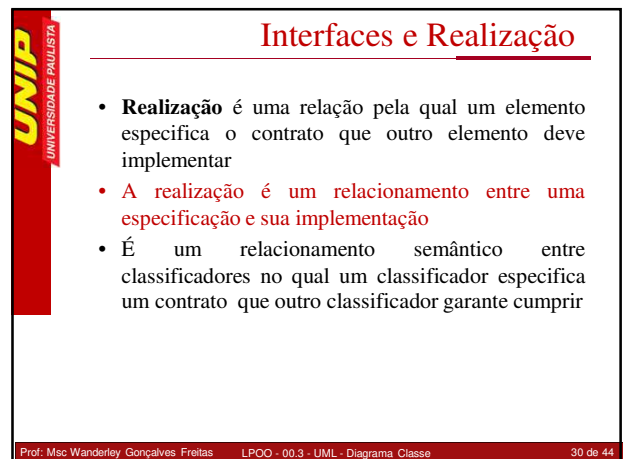
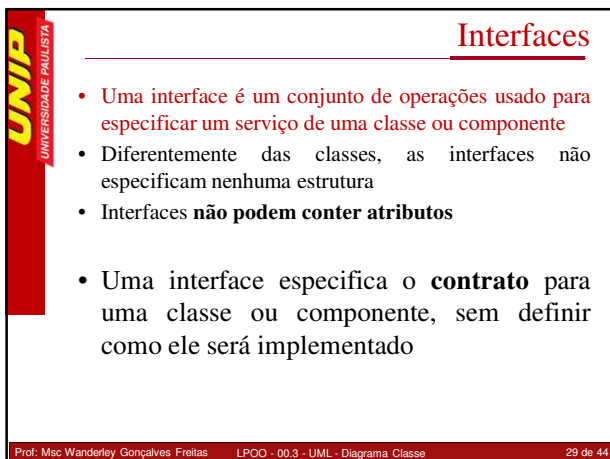
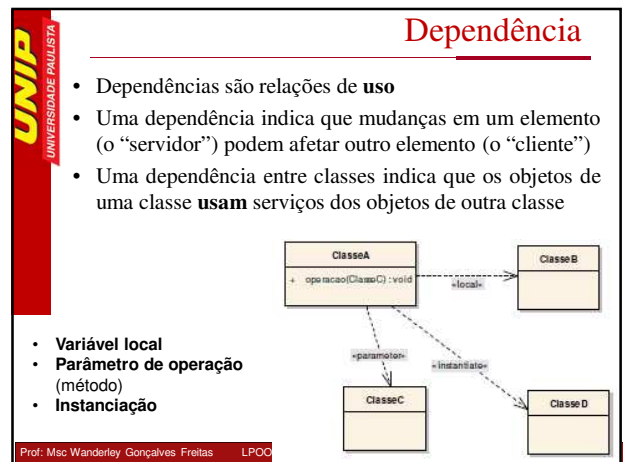
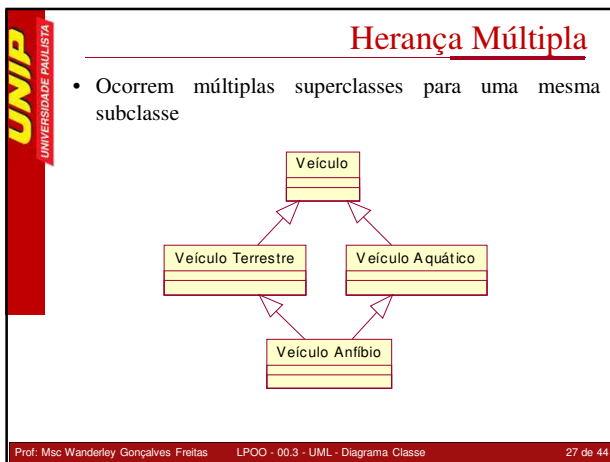
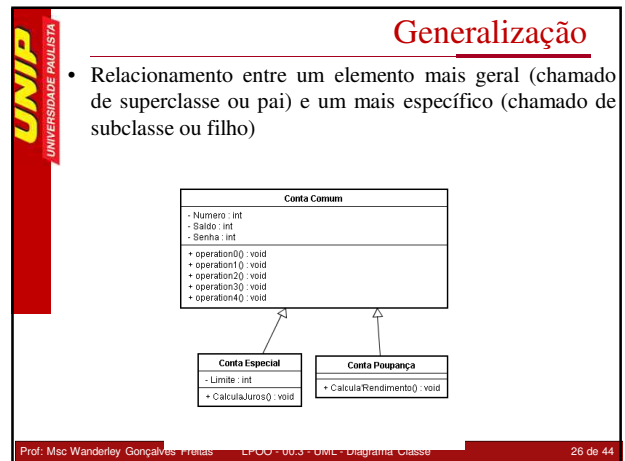
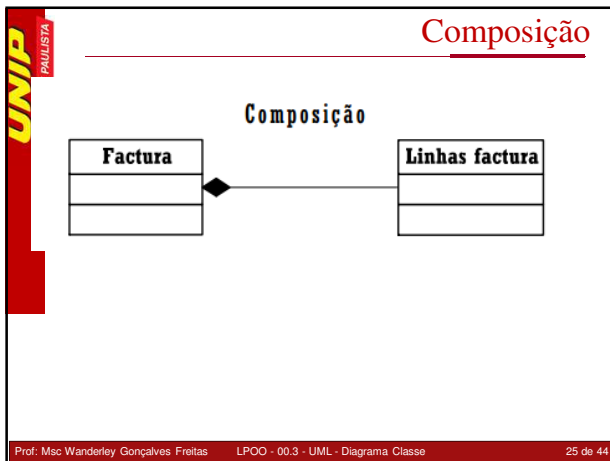
classDiagram
    class Carro
    class Motor
    class Pneu
    class Partes {
        ...
    }
    Carro o-- Motor
    Carro o-- Pneu
    Carro o-- Partes
    
```

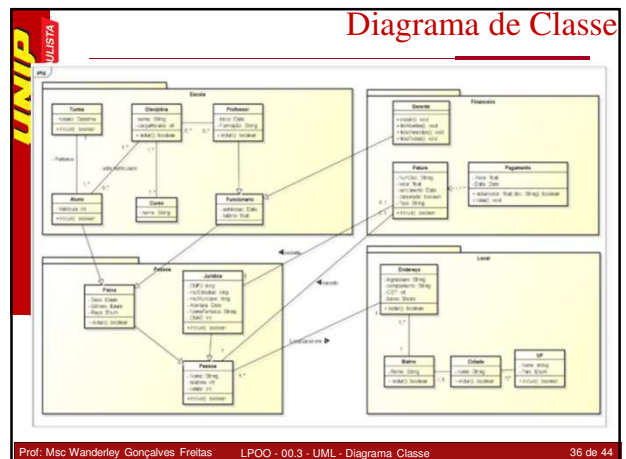
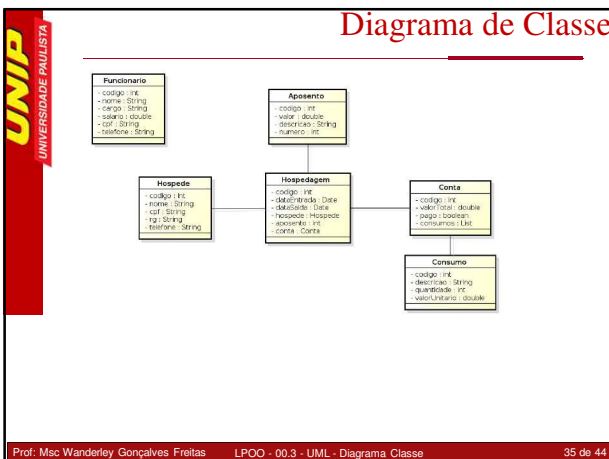
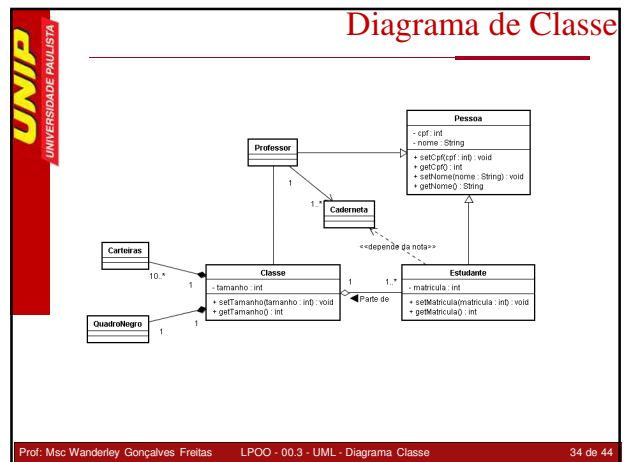
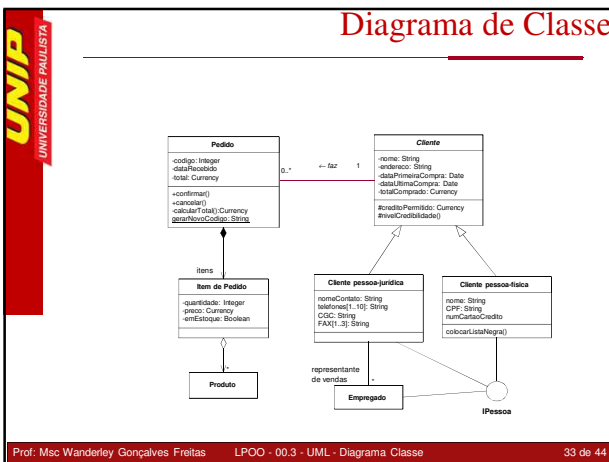
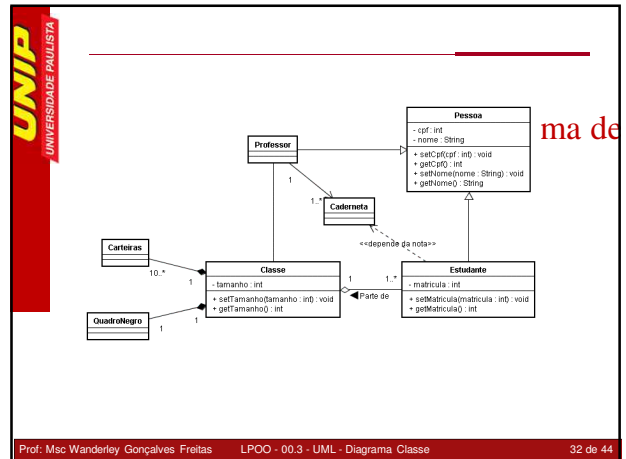
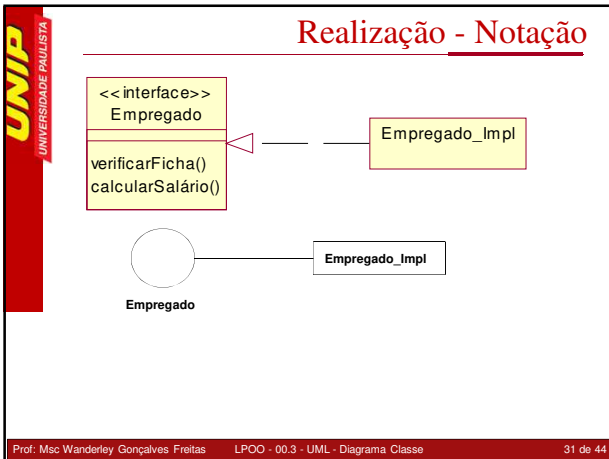
Prof: Msc Wanderley Gonçalves Freitas LPOO - 00.3 - UML - Diagrama Classe 23 de 44

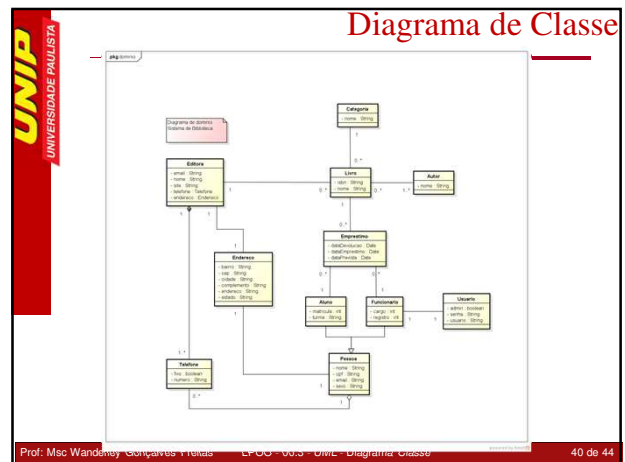
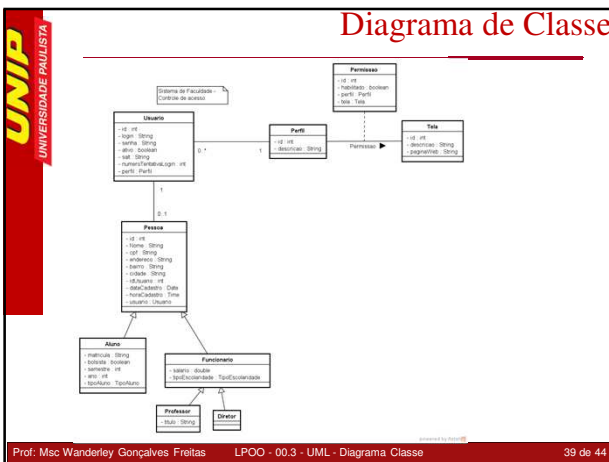
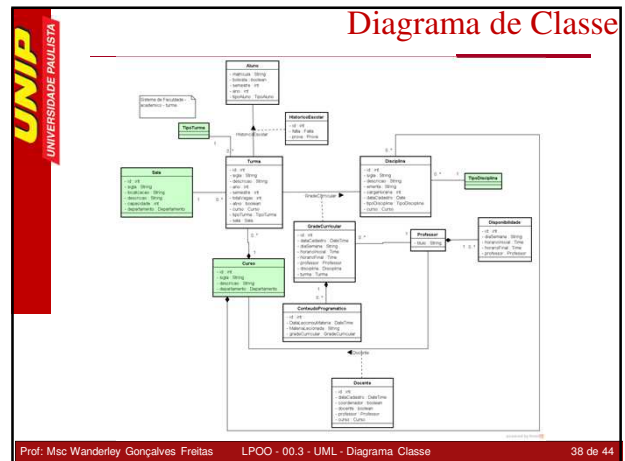
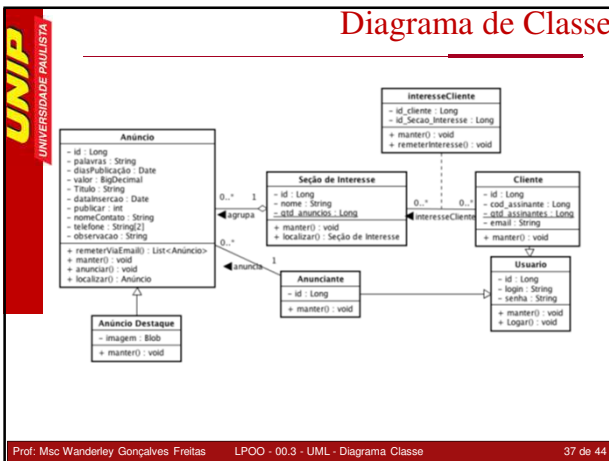
Composição

- Uma forma mais forte de agregação
- Há uma coincidência da vida das partes
- Uma vez criada a parte ela irá viver e morrer com ele
- O "Todo" é responsável pelo gerenciamento da criação e destruição das partes

Prof: Msc Wanderley Gonçalves Freitas LPOO - 00.3 - UML - Diagrama Classe 24 de 44







Aplicação Prática

- Aplicação Bibliotecária
- Aplicação de Controle Acadêmico
- Aplicação de Vídeo Locadora

PARTE DA APLICAÇÃO - GRUPO

Prof: Msc Wanderley Gonçalves Freitas LPOO - 00.3 - UML - Diagrama Classe 41 de 44

Referências Bibliográficas

- Pressman, R. S. **Software Engineering – A Practitioner’s Approach**. 6th Edition McGraw Hill (Higher Education). 2005. ISBN 0-07-285318-2.
- RUMBAUGH, J.; JACOBSON, I.; BOOCH, G. **The Unified Software Development Process**. 1st Edition. Addison-Wesley. 2005. ISBN 0-201-57169-2.
- RUMBAUGH, J.; JACOBSON, I.; BOOCH, G. **The Unified Modeling Language Reference Manual**. 2nd Edition. Addison-Wesley. 2005. ISBN 0-321-26797-4.
- Bezerra, E. **Princípios de Análise e Projeto de Sistemas com UML**. Editora Campus. 2003. ISBN 85-352-1032-6.

Outras bibliografias:

- GAMMA, E.; HELM, R.; JOHNSON, R.; VLISSIDES, J. **Design Patterns: Elements of Reusable Object-oriented Software**. Addison Wesley Longman, 1995.
- LARGMAN, G. **Applying UML and Patterns – An Introduction to Object-Oriented Analysis and Design**. Prentice Hall. 1998. ISBN 0-13-748880-7.

Prof: Msc Wanderley Gonçalves Freitas LPOO - 00.3 - UML - Diagrama Classe 42 de 44

P PAULISTA

Perguntas



Wanderley.unip@gmail.com
www.professor.wanderley.nom.br

Prof: Msc Wanderley Gonçalves Freitas LPOO - 00.3 - UML - Diagrama Classe 43 de 44

UNIP UNIVERSIDADE PAULISTA

Obrigado

Wanderley

Wanderley.unip@gmail.com
www.professor.wanderley.nom.br

Prof: Msc Wanderley Gonçalves Freitas LPOO - 00.3 - UML - Diagrama Classe 44 de 44