

Diagramas UML

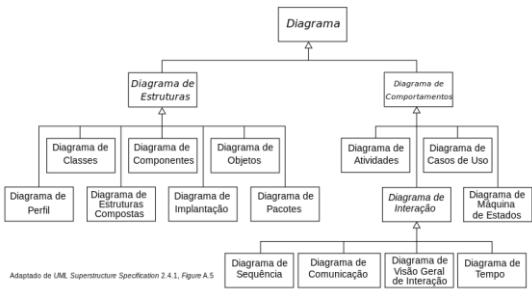
Disciplina: Engenharia de Software I
Prof. Me. Fernando Roberto Proença

Diagramas UML

- Um modelo é representado por um ou mais diagramas. Desta forma, um diagrama pode ser visto como uma visão dentro de um modelo.
- Um diagrama pode ser representado de várias formas, dependendo de quem irá interpretá-lo.

2

Conjunto de Diagramas UML



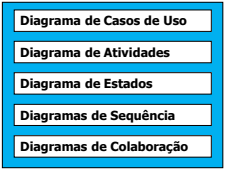
3

Principais Diagramas UML

Ponto de Vista Estático



Ponto de Vista Dinâmico



4

Diagrama de Casos de Uso

- Um caso de uso é uma **sequência de ações** realizada por um sistema que produz um **resultado** de valor observável para determinado ator
- Todos os casos de uso juntos devem descrever a **funcionalidade** completa do sistema (requisitos)
- **Atores** são fundamentais para a descoberta dos casos de uso.

5

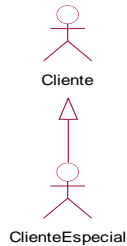
Diagrama de Casos de Uso

- **Cenários** para os requisitos
- Detalhamento dos requisitos para **implementação**
- **Interação** dos atores com o sistema.

6

Atores – Especialização

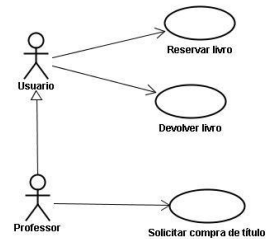
- É possível definir tipos gerais de atores e especializá-los usando o relacionamento de **especialização**.



7

Atores – Especialização

- Na generalização de ator o filho herda o papel que o pai pode desempenhar em um caso de uso.
- Facilita a leitura do modelo de caso de uso.



8

Diagrama de Classes

- Modelar o vocabulário de um sistema do ponto de vista do usuário/problema ou do desenvolvedor/solução
 - Ponto de vista do usuário/problema – na fase de captura e análise de requisitos, em paralelo com a identificação dos casos de uso;
 - Vocabulário do desenvolvedor/solução – na fase de projeto (*design*).
- Construído e refinado ao longo das várias fases do desenvolvimento do software.

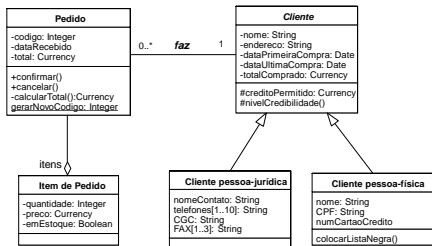
9

Diagrama de Classes

- Também serve para:
 - Especificar relacionamentos e colaborações;
 - Especificar esquemas lógicos de bases de dados;
 - Especificar visões (estrutura de dados de formulários, relatórios, etc).
- Modelos de objetos de domínio, negócio, análise e projeto.
- Base da implementação.

10

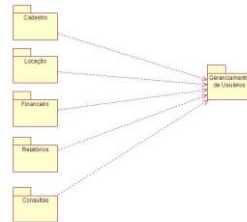
Diagrama de Classes



11

Diagrama de Pacotes

- Organizam elementos do sistema em grupos relacionados a fim de minimizar a dependência entre eles.



12

Diagrama de Objetos

- Oferece uma **visão estática** das instâncias (objetos) das classes
- Registrar os objetos e seus **relacionamentos** num determinado instante de tempo

Diagrama de Objetos

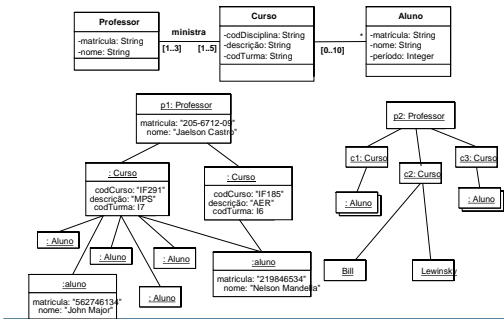


Diagrama de Objetos

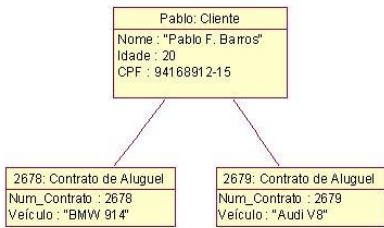


Diagrama de Objetos

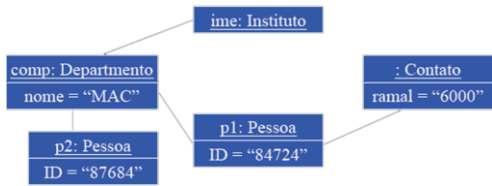


Diagrama de Atividades

- É um tipo **especial** de gráfico de **estados**.
- Exibe o **fluxo de atividades** no sistema
 - Atividade**: operação que resulta na mudança de um estado do sistema.
- Geralmente é utilizado para modelar uma **função** do sistema.

Diagrama de Atividades

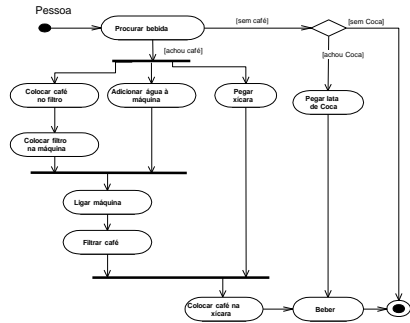
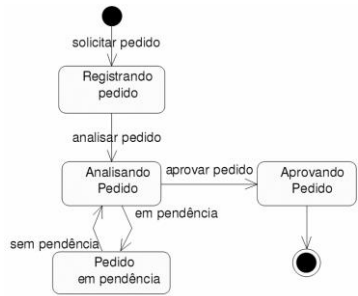


Diagrama de Estados

- Exibe os **estados** e as **transições** entre estes
- Modela uma **máquina de estados** (seqüência do ciclo de vida de um objeto)
- Especifica as **seqüências** de estados pelos quais um objeto passa durante seu tempo de vida em resposta a **eventos**.

19

Diagrama de Estados



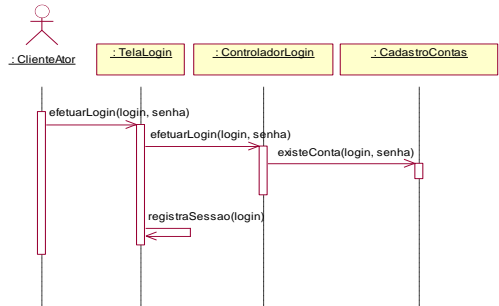
20

Diagrama de Sequência

- Exibe **objetos** e as **mensagens** trocadas entre eles ao longo do tempo
- Retrata a **ordenação temporal** das mensagens
- Descreve a maneira como os grupos de objetos colaboram em algum **comportamento** ao longo do tempo
 - registra o comportamento de **um único caso de uso**.

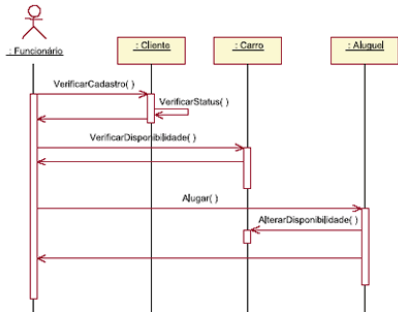
21

Diagrama de Sequência



22

Diagrama de Sequência



23

Diagrama de Colaboração

- Ilustra o fluxo de troca de mensagens entre objetos
- Mostram interação entre objetos **sem (muita) preocupação** com a **seqüência** das ações
- São adequados para especificar um **processo** na lógica do programa

24

Diagrama de Colaboração

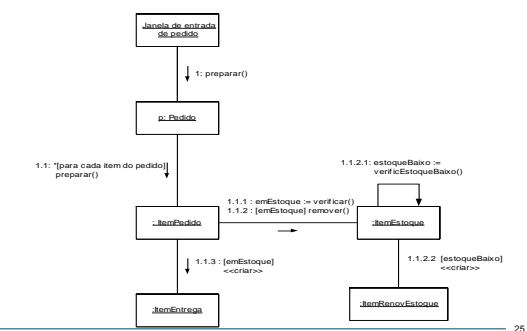


Diagrama de Colaboração

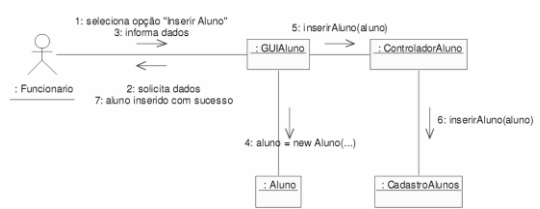


Diagrama de Componentes

- Exibe as **organizações** e as **dependências** de um conjunto de **componentes** de SW
- Retrata uma **visão estática** da **implementação** do **SOFTWARE**
- Relacionado com o diagrama de classes
 - **classes** e **interfaces** são mapeadas em **componentes**

Diagrama de Componentes

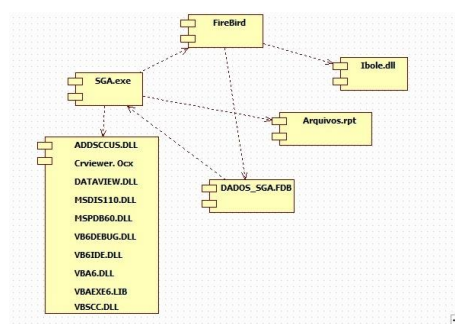


Diagrama de Componentes

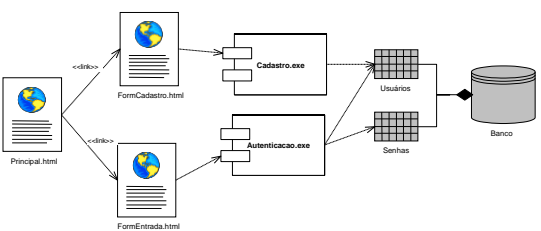
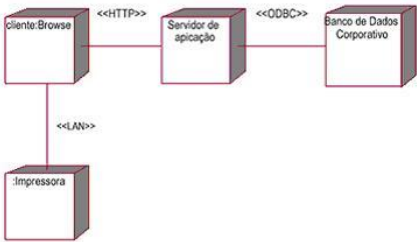


Diagrama de Implantação

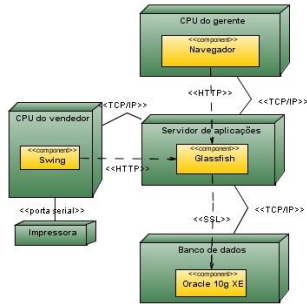
- Exibe a configuração dos nós de processamento e **dispositivos**
- Retrata uma **visão estática** da organização do **hardware**
- Relacionado com o diagrama de componentes
 - um **nó** inclui um ou mais **componentes**.

Diagrama de Implantação



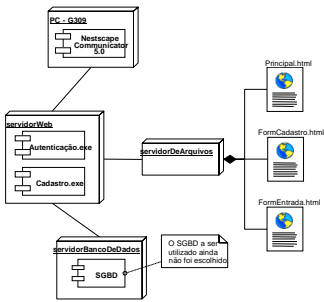
31

Diagrama de Implantação



32

Diagrama de Implantação



33

Dúvidas?

Prof. Me. Fernando Roberto Proença

fernando.proenca@uemg.br



34