

Resumo Arquitetura de Software - pt III

Diagramas UML Especificando a Arquitetura de Software

Especificando arquitetura

- Forma de comunicar a arquitetura através de modelos e diagramas

Especificação da Arquitetura

- Modelagem estrutural “avançada”
 - Diagrama de Pacotes
 - Diagrama de Componentes
 - Modelagem de comportamento
 - Diagrama de Casos de Uso
 - Modelagem da Arquitetura
 - Diagrama de Implantação
 - Padrões e Frameworks
-

Diagrama de pacotes?

- Diagramas estruturais usados para mostrar , em pacotes a organização e disposição de vários elementos do modelo
- Auxilia a gerencia das classes, interfaces, componentes...
- Organização de elementos de modelagem em grupos

Pacotes?

- Agrupam elementos proximos
- Hierarquicamente organizados
- Armazenar codigos e outros elementos

Benefícios dos diagramas de pacotes

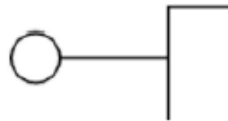
- O diagrama fornece uma visão clara da estrutura hierárquica dos variados elementos UML dentro de um determinado sistema
- Esses diagramas podem simplificar diagramas de classes complexos, criando elementos visuais organizados.
- Eles oferecem uma ótima visibilidade geral de projetos e sistemas de grande escala.
- Os elementos visuais podem ser atualizados com facilidade conforme a evolução dos sistemas e projetos.

- Dependências de importação e acesso entre pacotes , classes....
 - Principais dependências: **acesso e importação**
 - Um pacote bem estruturado é:
 - Coeso → elementos claramente relacionados
 - Fracamente acoplado → exportando apenas os elementos que os outros pacotes precisam enxergar
 - Não contém muitos aninhamentos
 - Tem um conjunto equilibrado de conteúdos
-

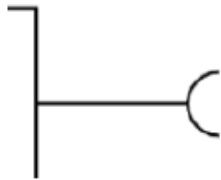
Diagrama de componentes

Diagrama de componentes

- Parte logica e substituível de um sistema ao qual se adapta e fornece a realização de um conjunto de interfaces
- Abstrações com interfaces bem definidas
- **Interface**: coleção de operações que especifica um serviço fornecido ou solicitado de uma classe ou componentes
- **Porta**: ponto específico de um componente que aceita mensagens
- **Estrutura interna**: implementação de um componente por meio de um conjunto de partes que são conectadas de uma determinada maneira
- **Parte**: especificação de um papel que compõe a implementação de um componente
- **Conector**: relacionamento de comunicação entre duas partes ou portas no contexto de um componente



Interface fornecida: uma linha reta a partir da caixa de componentes e com um círculo anexado. Esse símbolo representa a interface na qual um componente produz informações usadas pela interface necessária de outro componente.



Interface necessária: uma linha reta da caixa de componentes e com um semicírculo anexado (também representado como uma seta tracejada com uma seta aberta). Esse símbolo representa a interface na qual um componente requer informações para executar a função correta.

