



Visualização Arquitetural

Trabalho Prático 11

Christian Marlon Souza Couto
Luana Almeida Martins

Objetivo

- Contagem e exibição de dependências;
- Detecção de dependências cíclicas;
- Detecção e exibição de desvios em relação à arquitetura.

Solução

- Fornecer regras de conformidade;
- Criar uma matriz de dependências entre pacotes;
- Relatar tipos de dependências não permitidos conforme as regras estabelecidas.

Terminologia

- CAN - indica uma dependência permitida;
- CANNOT - indica uma dependência não permitida;
- MUST - indica uma dependência obrigatória.

Tipos de dependências

- Declaração de atributos;
- Criação de objetos;
- Extensão de classes;
- Implementação de interfaces.

Terminologia de cores

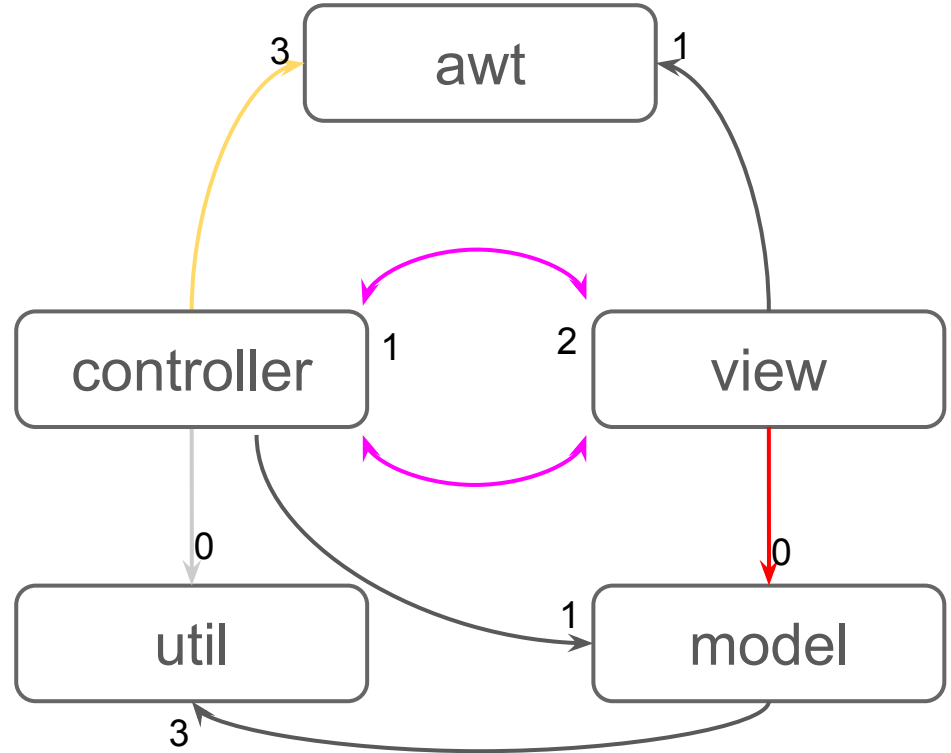
- Cinza - dependência permitida, porém não existente (CAN);
- Amarelo - indica uma dependência não permitida, porém existente (CANNOT);
- Vermelho - indica uma dependência obrigatória, porém não cumprida (MUST);
- Rosa - indica dependências cíclicas entre pacotes.

Exemplo

view **MUST** model

controller **CANNOT** awt

controller CAN util



Exemplo

	controller	model	util	view	awt
controller	-	1	0	2	3
model	0	-	3	0	0
util	0	0	-	0	0
view	1	0	0	-	1
awt	0	0	0	0	-

Implementação

