

DESACOPLANDO CAMADAS

A realização de uma arquitetura limpa se dá pela expressão do uso de princípios, como o SOLID, visando a “independência” do código de modificações externas. Dito de outra maneira, o software, então em camadas, pode tê-las separadas a fim de maior adaptabilidade e manutenção.

Quanto vídeo, inicia-se por um breve panorâmica, o qual tangencia o QDD em contrapartida ao CDD. Para o sentido didático que almeja o vídeo, criou a primeira rota, para o que leva a criação de um teste de integração. Isto é, para casos em que a camada não está bem definida, utiliza-se um teste de integração.

Destacou o que é QDD: dentre seus fundamentos estão o GIVEN, WHEN, THEN - ARRANGE, ACT, ASSERT.

Na prática: tem-se um cenário dado e uma ação é tomada e então, valida-se o resultado.

Uma tendência na tecnologia que motivou a separação de camadas foram aplicações que, na mesma camada, tínhamos a *view*, tratava-se da regra e persistem-se dados no banco. O que engessa o código para modificações e testes, uma vez que há dependências internas que não podem ser unitariamente acessadas.

Uma maneira de quebrar o Princípio de Responsabilidade Única, segue-se, são questões relativas ao banco de dados. Ora, o banco tem relação direta, em suma, pois mudanças no banco levariam a mudanças no código em como os dados são persistidos. Da forma como estava sendo feita, havia um acoplamento direto do banco com como as requisições eram feitas.

Uma solução para o bem desenvolvimento da aplicação, a qual conjuga testes, é através de desacoplamento. Separando os casos de uso, para o exemplo do vídeo, permite que separemos a camada do banco da camada da regra.

Através da compreensão de qual entidade engloba qual, é possível encontrar dependências entre as entidades e separar a lógica da geração da dependência dentro da classe que ela depende. Isso significa identificar as regras de aplicação independentes.

Um exemplo é a separação em interfaces da conexão do banco (que é um fato comum). Abstrai-se os atributos fundamentais para a conexão do banco, bem como seus métodos, assim garantindo maior segurança ao código, caso alterações do banco de dados utilizado.