

Aluno THOMAS EDUARDO SILVA E SOUZA

RA: 22095036-2

TRABALHO HENRIQUE

RESUMO VIDEO.

O vídeo "Como criar e desacoplar camadas usando SOLID", apresentado por Rodrigo Branas, aborda os princípios do SOLID como uma abordagem para o desenvolvimento de software mais modular e fácil de manter. Um resumo dos principais assuntos discutidos pode ser encontrado aqui:

O SOLID é um acrônimo para cinco princípios de design de software que se concentram em um aspecto específico da escrita de código limpo e de fácil manutenção. A seguir estão os princípios: Princípio da Responsabilidade Única, Princípio Aberto/Fechado, Princípio da Substituição de Liskov, Princípio da Segregação de Interfaces e Princípio da Inversão de Dependências.

De acordo com o princípio da única responsabilidade (SRP), uma classe deve ter apenas uma razão para mudar ou apenas uma responsabilidade para fazê-lo. Facilita a coesão e a manutenção do código.

O princípio de cooperação aberta/fechada (OCP): O princípio OCP diz que as classes devem estar fechadas para extensão, mas abertas para modificação. Isso pode ser feito com abstrações e interfaces, que permitem a inclusão de novos comportamentos sem alterar o código existente.

De acordo com o princípio de substituição de Liskov (LSP), é possível substituir objetos por instâncias de suas subclasses sem comprometer a integridade do programa. Isso garante que as classes derivadas possam ser utilizadas em vez de suas classes base sem erros.

O princípio de segregação de interfaces (ISP) defende que as interfaces devem ser personalizadas para cada cliente, evitando interfaces genéricas e muito grandes. Isso evita que as classes dependam de recursos que não são necessários.

O Princípio de Inversão de Dependência (DIP): O DIP diz que as dependências não devem ser usadas para implementações concretas, mas para abstrações. Isso torna o código mais flexível e fácil de testar, permitindo que as classes dependam de abstrações.

Desacoplamento de Camadas: É possível desacoplar as camadas de um sistema usando os princípios do SOLID. Isso torna cada camada independente e torna a manutenção mais fácil. Isso é feito por meio da utilização de injeção de dependência para fornecer implementações concretas e da criação de interfaces que definem contratos entre as camadas.