Trabalho de Pesquisa e Apresentação

Objetivo: O objetivo deste trabalho é explorar e apresentar conceitos fundamentais de desenvolvimento de software, com foco em princípios de design de software e práticas ágeis. Cada grupo de alunos será responsável por pesquisar, analisar e apresentar um dos tópicos listados abaixo, discutindo suas aplicações práticas, benefícios e desafios.

- **Grupo 1: Princípios SOLID - Introdução e SRP**
    - Introduza os princípios SOLID e concentre-se no Single Responsibility Principle (SRP). Explique como ele contribui para a manutenção e flexibilidade do código.
- **Grupo 2: Princípios SOLID - OCP e LSP**
    - Explique o Open/Closed Principle (OCP) e o Liskov Substitution Principle (LSP). Dê exemplos práticos de como esses princípios são aplicados.
- **Grupo 3: Princípios SOLID - ISP e DIP**
    - Foque no Interface Segregation Principle (ISP) e no Dependency Inversion Principle (DIP). Mostre como esses princípios ajudam na criação de sistemas desacoplados.
- **Grupo 4: GRASP - Introdução e Creator**
    - Apresente os princípios GRASP e concentre-se no padrão Creator. Discuta como atribuir responsabilidades corretamente para criar um design eficiente.
- **Grupo 5: GRASP - Information Expert e Low Coupling**
    - Foco em Information Expert e Low Coupling. Explique como alocar responsabilidades e manter baixo acoplamento entre classes.
- **Grupo 6: GRASP - High Cohesion e Polymorphism**
    - Aborde High Cohesion e Polymorphism. Mostre a importância de manter alta coesão e usar polimorfismo para criar código mais flexível.
- **Grupo 7: Lei de Demeter**
    - Explique a Lei de Demeter e sua importância para o design de software. Dê exemplos de como reduzir o acoplamento entre objetos.
- **Grupo 8: Desenvolvimento Ágil - KISS e YAGNI**
    - Foque em KISS (Keep It Simple, Stupid) e YAGNI (You Aren't Gonna Need It). Mostre como esses princípios ajudam a manter o código simples e evitar funcionalidades desnecessárias.
- **Grupo 9: Desenvolvimento Ágil - DRY**
    - Concentre-se no princípio DRY (Don't Repeat Yourself). Discuta estratégias para evitar duplicação de código e promover a reutilização.

- **Grupo 10: DDD (Domain-Driven Design)**

    - Apresente o conceito de Domain-Driven Design (DDD). Explique como modelar o domínio e as práticas associadas a DDD.

Instruções:

1. **Pesquisa e Fundamentação Teórica**
    - Cada grupo deve fornecer uma base teórica sólida sobre o tópico, com definições, objetivos e importância no contexto de desenvolvimento de software.
2. **Exemplos Práticos**
    - Forneça exemplos claros e práticos que demonstrem a aplicação do conceito. Podem ser códigos de exemplo, casos de estudo ou cenários do mundo real.
3. **Benefícios e Desafios**
    - Discuta os benefícios de aplicar o conceito e os possíveis desafios enfrentados. Inclua vantagens e limitações.
4. **Comparação com Outras Abordagens**
    - Sempre que aplicável, compare o tópico com outras abordagens ou princípios para destacar suas vantagens e desvantagens.
5. **Demonstração/Apresentação**
    - Prepare uma apresentação clara e organizada, com slides e recursos visuais. Inclua uma demonstração prática se relevante.
6. **Perguntas e Respostas**
    - Esteja preparado para responder a perguntas da turma e discutir pontos adicionais relacionados ao tópico.
7. **Entrega**
    - Cada grupo deverá entregar um documento em PDF através do AVA com o conteúdo estudado. LEMBRE-SE DE COLOCAR O REFERENCIAL BIBLIOGRÁFICO
    - CHATGPT, GEMINI E outros NÃO são referências. Utilizem-nos como auxílio, mas busquem encontrar fontes confiáveis.