

PROJETO A3 – GESTÃO E QUALIDADE DE SOFTWARE

Tema: Refatoração de Código e Boas Práticas do Clean Code

Aluno: Kaique Alves

1. Código Original (Legado)

O código legado foi um pequeno sistema bancário em Java com interface Swing. O programa permitia criar conta, realizar depósitos, saques e verificar saldo. O código apresentava diversas más práticas de programação, o que o torna adequado para a aplicação de técnicas de Clean Code.

2. Deficiências Identificadas

- Variáveis com nomes não descritivos (a, b, c, etc.)
- Lógica de negócio e interface gráfica na mesma classe
- Falta de orientação a objetos e modularização
- Falta de tratamento de erros adequado
- Ausência de testes unitários
- Nenhum padrão de projeto utilizado

3. Refatoração e Justificativas

O código foi refatorado com base nos princípios de Clean Code e SOLID, adotando o padrão MVC. Foram criadas classes específicas para Cliente, ContaBancaria e BancoAppGUI. Os métodos receberam nomes claros, e as responsabilidades foram separadas. Além disso, foram implementados testes unitários simples para validar o comportamento do código.

4. Estrutura do Projeto Refatorado

```
BancoLimpo/  
  ■■■ src/  
    ■ ■■■ Cliente.java  
    ■ ■■■ ContaBancaria.java  
    ■ ■■■ BancoAppGUI.java  
    ■■■ testes/  
      ■■■ ContaBancariaTest.java
```

5. Resultados e Conclusão

Após a refatoração, o código tornou-se mais legível, modular e fácil de manter. A aplicação dos princípios SOLID e Clean Code reduziu a complexidade e aumentou a confiabilidade. A adição de testes unitários garantiu que as funcionalidades originais fossem preservadas. Este trabalho evidenciou a importância da refatoração e das boas práticas na manutenção de software.

6. Links do Repositório

- Repositório original: [ADICIONAR LINK DO GITHUB COM CÓDIGO LEGADO]
- Repositório refatorado: [ADICIONAR LINK DO GITHUB COM CÓDIGO LIMPO]
- Pasta de testes: [ADICIONAR LINK DO GITHUB COM TESTES UNITÁRIOS]