

**UNIVERSIDADE ESTADUAL DA PARAÍBA**

**CAMPUS DE CAMPINA GRANDE**

**CENTRO DE CIÊNCIAS E TECNOLOGIA**

**CURSO DE CIÊNCIA DA COMPUTAÇÃO**

**THOMAS FREIRE**

**ZAYON TAVARES**

**JOÃO VICTOR BATISTA**

# **Relatório de Testes e Cobertura de Código**

**CAMPINA GRANDE - PB**

**2025**

## **1. Introdução**

Este relatório apresenta os testes realizados e a análise da cobertura de código para o projeto desenvolvido em Java. Os testes foram implementados utilizando **JUnit** e **Mockito**, enquanto a cobertura de código foi medida por meio da ferramenta integrada do **IntelliJ IDEA**. O objetivo deste relatório é fornecer uma visão detalhada sobre a eficácia dos testes implementados e apontar possíveis melhorias.

## **2. Ferramentas Utilizadas**

* **Linguagem**: Java
* **Framework de Testes**: JUnit, Mockito
* **Ferramenta de Cobertura**: IntelliJ IDEA (integrada)

## **3. Metodologia**

Os testes foram organizados em diferentes classes e camadas do sistema, abrangendo as seguintes categorias:

* **DAO (Data Access Object)**: Testes relacionados à persistência de dados.
* **Interface**: Testes de comunicação entre o usuário e o sistema.
* **Sistema**: Testes da lógica de negócio.

Para medir a cobertura, foram considerados os seguintes critérios:

* **Cobertura de métodos**: Quantidade de métodos testados em relação ao total.
* **Cobertura de linhas**: Percentual de código executado durante os testes.

## **4. Resultados dos Testes**

A seguir, apresentamos os resultados dos testes executados:

| **Módulo** | **Testes Executados** | **Testes Passaram** | **Testes Falharam** |
| --- | --- | --- | --- |
| ClienteInterface | 3 | 3 | 0 |
| LojaInterface | 5 | 4 | 1 |
| LojaSystem | 3 | 3 | 0 |
| CarrinhoCompras | 6 | 6 | 0 |
| ProdutoValidation | 1 | 1 | 0 |
| ProdutoDAO | 9 | 9 | 0 |
| ProdutoSystem | 6 | 6 | 0 |
|  |  |  |  |

### 

**4.1 Análise dos Resultados**

* **CarrinhoCompras, ProdutoValidation, ProdutoDAO e ProdutoSystem** apresentaram 100% de sucesso, o que indica boa cobertura e testes bem implementados.
* **ClienteInterface e LojaSystem,** que anteriormente apresentavam falhas, agora foram corrigidos e estão com todos os testes passando.
* **LojaInterface** ainda possui 1 teste falho (Este erro ocorreu porque uma parte do código foi alterado e faltou ajustar o teste, mas antes ele estava passando).

## **5. Cobertura de Código**

A ferramenta do IntelliJ IDEA foi utilizada para medir a cobertura de código, fornecendo insights sobre as áreas do sistema que não estão sendo devidamente testadas. Com base nos testes:

* A camada de persistência (DAO) apresenta cobertura excelente, com 100% de sucesso nos testes.
* As camadas Sistema e Interface mostram boa cobertura geral, com exceção de um ponto de falha em LojaInterface.
* As funcionalidades de Carrinho de Compras e Produto estão completamente testadas e cobertas, o que garante confiança em funcionalidades essenciais como compra e manipulação de produtos.

## **6. Conclusão**

Os testes realizados demonstram que a camada de persistência de dados (**DAO**) está bem testada e funcionando corretamente. Houve evolução positiva nos módulos de sistema e interface, com destaque para a correção dos testes que anteriormente falharam.

### **6.1 Recomendações para próxima sprint**

* Investigar e corrigir a falha remanescente no módulo **LojaInterface**.
* Ampliar o uso de **Mockito** para isolar dependências e testar comportamentos específicos.
* Manter a prática de testes contínuos para assegurar a estabilidade do sistema em futuras alterações.
* Realizar testes de integração para validar a interação entre os componentes, especialmente nos fluxos de compra e cadastro.

A implementação dessas recomendações contribuirá significativamente para o aumento da confiabilidade, manutenibilidade e qualidade geral do projeto.