

# **Introdução aos Testes de Software e Automação com ênfase em qualidade e eficiência**



Alunos: João Antonio, João Vítor Póvoa, João Pedro Paulino, Frederico Gomes e Sebastião Nogueira



# 1 Tipos de Testes de Software

# Exploração aprofundada sobre testes unitários em aplicações modernas

## Importância dos Testes Unitários

Testes unitários garantem a funcionalidade correta de cada parte do código, ajudando desenvolvedores a manterem a qualidade do software.

## Integração com Ferramentas de Teste

Ferramentas como JEST permitem a integração contínua, aumentando a eficiência do desenvolvimento e a confiabilidade do software.

## Automação no Processo de Testes

Automatizar testes reduz o esforço manual e aumenta a cobertura, sendo essencial para práticas de integração contínua.

## Desafios em Testes Unitários

Apesar dos benefícios, testes unitários podem ser complexos e exigem disciplina para escrever e manter, especialmente em projetos grandes.

# ■ Integração de módulos e sua importância em testes de software

## ■ Verificação de Interações entre Módulos

Os testes de integração garantem que as interações entre diferentes módulos do software funcionem como esperado, minimizando erros.



# ■ Teste E2E: Simulação de comportamento do usuário em interfaces

## Importância do Teste E2E

Testes de ponta a ponta validam a experiência do usuário final, garantindo que todas as funcionalidades funcionem em conjunto.

## Ferramentas para Teste E2E

Cypress é uma ferramenta poderosa para testes E2E, permitindo simulações precisas de ações do usuário em navegadores.

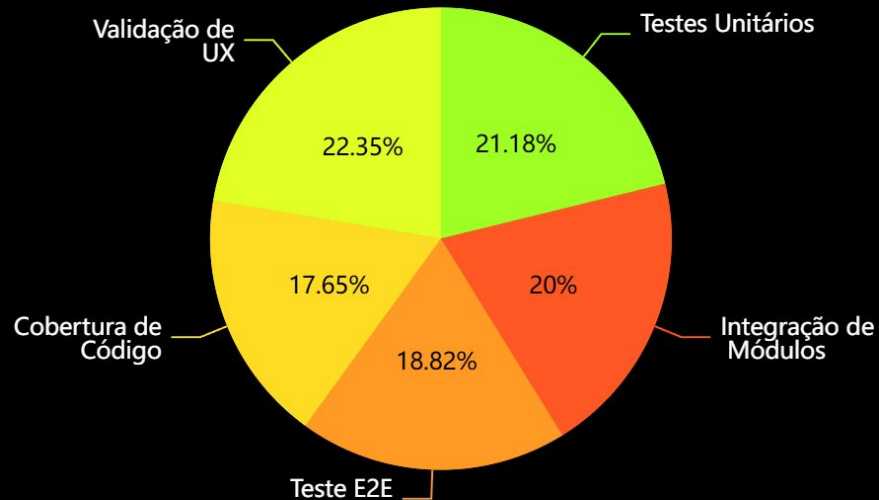
## Exemplo de Teste E2E com Cypress

A simulação de login e interações com a interface garante que o fluxo do usuário esteja correto e livre de erros.

## Desafios do Teste E2E

Os testes E2E podem ser mais lentos e requerem um ambiente de teste estável para simular condições reais de uso.

# ■ Análise de Testes em Aplicações Modernas



Importância dos Testes em Software(porcentagem)





## **2 Ferramentas de Teste Utilizadas**



# JEST: Ferramenta essencial para testes unitários em projetos React



## Desenvolvimento Rápido com JEST

A ferramenta facilita a criação de testes unitários rápidos e eficientes, ideal para desenvolvedores que buscam agilidade em projetos React.



## Integração com React

JEST oferece integração perfeita com componentes React, permitindo simulações eficazes de comportamento.



## Cobertura de Código Aumentada

A metodologia de cobertura de código do JEST ajuda a identificar partes do código que não estão sendo testadas, melhorando assim a qualidade do software.



## Execução dos Testes Simplificada

A execução dos testes é feita de forma simples através de comandos npm, otimizando o fluxo de trabalho do desenvolvedor.



# ■ Cypress: A solução perfeita para testes E2E visuais e interativos



## Testes E2E Visualmente Interativos

Cypress é ideal para testes de ponta a ponta, permitindo que desenvolvedores simulem ações de usuários em cenários reais.



## Geração Automática de Vídeos e Screenshots

A ferramenta gera vídeos e screenshots automaticamente, facilitando a análise de falhas durante os testes.



## Documentação Acessível e Abrangente

Cypress possui extensa documentação, tornando o aprendizado e a utilização da ferramenta mais acessíveis para novos usuários.

# Comparativo entre JEST e Cypress no contexto de automação de testes

## Testabilidade e Cobertura

Ambas as ferramentas oferecem soluções únicas para testes, cada uma focando em aspectos diferentes da testabilidade do software.

## Desempenho e Curva de Aprendizado

JEST é mais fácil de aprender e implementar, enquanto Cypress requer um pouco mais de tempo para dominar, devido à sua complexidade.

## Contexto de Uso e Aplicações

JEST é indicado para testes unitários e integração, enquanto Cypress é mais apropriado para testes de interface e E2E.

## Considerações Finais e Escolha da Ferramenta

A escolha entre JEST e Cypress depende das necessidades específicas do projeto e do foco em testes unitários ou de interface.



**Obrigado**