

# Trabalho de Teste de Software

**Trabalho avaliativo (1,0).**

**Cada resposta deve conter no mínimo 10 linhas (Exceto, atividade pratica).**

**Data de entrega 17/09/2024 até 23:59:59.**

**Cada pergunta deve ter referência do site acessado, o link e o dia do acesso.**

1. O que são testes de software e por que são fundamentais no ciclo de desenvolvimento de software?
2. Explique a diferença entre verificação e validação no contexto de testes.
3. Quais são os principais objetivos dos testes de software?
4. Cite três mitos comuns sobre testes de software e explique por que são falsos.
5. Qual a diferença entre erros, falhas e defeitos em um software?
6. Explique o que são testes funcionais e dê um exemplo prático.
7. O que são testes não funcionais? Cite dois exemplos e suas finalidades.
8. Qual é a diferença entre testes de caixa preta e testes de caixa branca?
9. Defina e explique os testes de regressão e quando eles devem ser aplicados.
10. Como os testes exploratórios diferem dos testes tradicionais? Quando são mais eficazes?
11. Quais são as etapas do ciclo de vida do teste de software?
12. O que deve ser incluído no plano de testes de um software?
13. Como um caso de teste é estruturado? Quais são os principais componentes?
14. Explique a importância da rastreabilidade entre requisitos e casos de teste.
15. Por que é importante realizar testes de regressão após correções de bugs?
16. O que são testes de integração e como eles diferem dos testes unitários?
17. Cite e explique dois desafios encontrados ao executar testes de integração.
18. Em que situação é adequado usar mocks ou stubs durante os testes de integração?
19. Qual é a importância de testar as interfaces entre diferentes módulos de software?
20. Como garantir que um teste de integração cubra todas as dependências entre componentes?

21. Quais são as principais vantagens e desvantagens da automação de testes?
22. Em que situação a automação de testes pode ser prejudicial ou ineficaz?
23. O que é um teste automatizado de regressão e como ele difere dos testes manuais?
24. Explique o conceito de CI/CD (Integração Contínua e Entrega Contínua) e sua relação com a automação de testes.
25. Cite duas ferramentas populares para automação de testes e suas principais características.
26. Qual é o propósito dos testes de aceitação e quem normalmente os executa?
27. Quais são os critérios de aceitação e como eles são definidos em um projeto?
28. Explique a diferença entre testes de aceitação e testes funcionais.
29. Por que os testes de aceitação são cruciais em metodologias ágeis?
30. Dê um exemplo prático de um caso de teste de aceitação e como ele seria validado.
31. O que são testes de carga e por que são importantes?
32. Explique como os testes de estresse são conduzidos e sua finalidade.
33. Como medir o desempenho de um software? Cite duas métricas importantes.
34. Quais são os principais desafios ao realizar testes de desempenho em sistemas distribuídos?
35. Como a escalabilidade de um sistema pode ser avaliada durante os testes?
36. Explique o que são testes unitários e qual é seu papel no desenvolvimento de software.
37. O que são mocks e stubs, e como eles são utilizados em testes unitários com JUnit?
38. Crie um exemplo de código Java que execute um teste unitário simples utilizando o framework JUnit.
39. Qual a importância da anotação `@BeforeEach` em testes unitários no JUnit? Dê um exemplo de uso.
40. Como o conceito de TDD (Test-Driven Development) se relaciona com os testes unitários? Explique o processo.