

Preparatorio CTFL

O que é Teste? [🔗](#)

O teste de software é o processo sistemático de identificar defeitos e garantir que o sistema funcione conforme esperado. Envolve o planejamento, análise, modelagem, implementação, execução e geração de relatórios, avaliando o progresso e os resultados.

Importância do Teste [🔗](#)

Sistemas com defeitos podem gerar:

- Prejuízos financeiros
- Perda de tempo e produtividade
- Danos à reputação da empresa
- Perda de confiança dos clientes
- Riscos à vida (em sistemas críticos)

Características dos Testes [🔗](#)

- Atividades para descobrir defeitos e avaliar a qualidade dos artefatos
- Verificação (requisitos) e validação (necessidades)
- Atividade essencialmente intelectual, apesar do uso de ferramentas
- Envolve pensamento crítico, analítico e sistêmico

Objetivos dos Testes [🔗](#)

- **Encontrar defeitos**
- **Evitar defeitos**
- **Construir confiança na qualidade**
- **Fornecer informações para tomada de decisão**
- **Reduzir riscos**
- **Verificar conformidade com requisitos**
- **Avaliar produtos de trabalho**

Os objetivos podem variar conforme o contexto, o nível de teste e o modelo de ciclo de vida adotado.

Teste e Depuração [🔗](#)

- **Teste:** Identifica falhas, confirma correções (teste de confirmação) e previne regressões.
- **Depuração (Debug):** Reproduz falhas, diagnostica e corrige defeitos.

Por que os testes são necessários? [🔗](#)

- Garantem qualidade dentro de escopo, tempo e orçamento
- Apoiam decisões ao longo do ciclo de vida do software
- Ajudam a cumprir requisitos legais, regulatórios e contratuais
- Identificam defeitos que podem ser corrigidos antes do impacto

Teste e Garantia de Qualidade (QA) [🔗](#)

- **Teste (Controle de Qualidade - QC):** Foca no produto, tem abordagem corretiva.

- **QA:** Foca no processo, com abordagem preventiva.
- Ambos usam os resultados dos testes para melhorar o produto e o processo.

Erro, Defeito e Falha [↗](#)

- **Erro:** Engano humano
 - **Defeito:** Erro que gera um bug no código ou documento
 - **Falha:** Comportamento incorreto causado pela execução de um defeito
 - As falhas podem ser causadas por ambiente, pressão, desconhecimento do domínio ou fatores emocionais
-

Sete Princípios do Teste [↗](#)

1. **O teste revela a presença, não a ausência de defeitos**
 2. **Testes exaustivos são impossíveis**
 3. **Testes antecipados economizam tempo e dinheiro**
 4. **Defeitos se concentram em áreas específicas**
 5. **Testes se tornam obsoletos com o tempo**
 6. **Testes dependem do contexto**
 7. **Ausência de defeitos não garante satisfação do usuário**
-

Processo de Teste [↗](#)

Não existe um modelo único; o processo deve ser adaptado ao contexto. As atividades principais incluem:

1. **Planejamento:** Define objetivos e abordagem.
 2. **Monitoramento e Controle:** Acompanha progresso e ajusta planos.
 3. **Análise:** Determina o que testar, com base em requisitos e riscos.
 4. **Modelagem:** Converte condições de teste em casos e scripts.
 5. **Implementação:** Prepara dados, ambiente e procedimentos de teste.
 6. **Execução:** Executa os testes, registra e analisa resultados.
 7. **Conclusão:** Encerra o ciclo, coleta lições aprendidas e reporta aos stakeholders.
-

Fatores que Influenciam o Teste [↗](#)

- Expectativas dos stakeholders
- Experiência e habilidades da equipe
- Criticidade do sistema
- Restrições técnicas e de projeto
- Modelo de desenvolvimento adotado
- Ferramentas disponíveis

Esses fatores impactam a estratégia, automação, cobertura, documentação e relatórios.

Testware [↗](#)

São os artefatos produzidos durante o teste (ex: planos, scripts, dados, relatórios). Cada organização os gerencia de forma diferente. A rastreabilidade entre testware e requisitos garante:

- Cobertura adequada
 - Análise de riscos
 - Impacto das mudanças
 - Clareza nos relatórios
-

Papéis no Teste [↗](#)

- **Gerente de Teste:** Lidera o processo e a equipe, planeja, monitora e conclui.
- **Testador:** Executa testes, automatiza quando necessário, documenta falhas e avalia características funcionais e não funcionais.

Papéis podem ser acumulados ou alternados conforme o contexto.

Habilidades Essenciais de Testadores [↗](#)

- Conhecimento técnico e de testes
- Atenção aos detalhes, curiosidade, pensamento crítico
- Boa comunicação e trabalho em equipe
- Compreensão do domínio de negócio

Testadores devem comunicar falhas de forma construtiva e lidar com resistência (ex: viés de confirmação).

Abordagem de Equipe Completa [↗](#)

Inspirada no Extreme Programming (XP), todos são responsáveis pela qualidade. Testadores colaboram com desenvolvedores e stakeholders para:

- Criar testes de aceitação
- Planejar automações
- Compartilhar conhecimento

Nem sempre é adequada, especialmente em sistemas críticos.

Independência nos Testes [↗](#)

Quanto mais independente o testador, mais imparcial o processo:

- **Sem independência:** Dev testa seu próprio código
- **Alguma independência:** Outro dev testa
- **Alta independência:** Testadores dedicados fora da equipe de desenvolvimento
- **Muito alta:** Testadores externos à organização

Vantagens:

- Menos vieses, maior imparcialidade
- Detecção de diferentes tipos de falhas

Desvantagens:

- Menor colaboração
- Possíveis atrasos
- Falta de informações do contexto