

Mentoria Seja Tester Academy

– Aula 3

Fundamentos de Teste - Parte 02



SUMÁRIO

Quais assuntos vamos aprender nessa aula?

- O que é teste de software?
- Para que serve o teste de software?
- Teste vs. Depuração: Qual a diferença?
- Por que os testes são necessários?
- Como os testes contribuem para o sucesso?
- Teste x Garantia da Qualidade (QA): Qual a diferença?
- Princípios de Teste
- Atividades de teste
- Processo de teste no contexto



O QUE É TESTWARE?

Durante os testes de um sistema, várias coisas são criadas: documentos, registros, dados e ferramentas. Todo esse material que é produzido como resultado do trabalho de teste recebe um nome técnico: testware.

Cada empresa pode organizar esse material de um jeito diferente, mas o importante é que tudo esteja bem cuidado, bem nomeado e guardado da forma certa. Isso evita confusão e garante que o material esteja correto, completo e útil no futuro.



TIPOS DE MATERIAIS CRIADOS DURANTE O PROCESSO DE TESTE:

1. Planejamento dos Testes:

Aqui são criados documentos que ajudam a organizar os testes antes que eles comecem:

- Plano de testes (um guia geral do que será feito)
- Cronograma (quando cada teste será feito)
- Registro de riscos (possíveis problemas e como evitá-los)
- Critérios de entrada e saída (quando o teste pode começar e quando pode ser encerrado)



O QUE É TESTE DE SOFTWARE?

Muita gente acha que testar software é apenas rodar o programa e ver se está certo ou errado. Mas o teste vai além disso! Ele faz parte de todo o processo de desenvolvimento e inclui atividades como planejamento, análise e verificação.

Outro erro comum é pensar que testar significa apenas conferir se o sistema segue as regras definidas. Na verdade, os testes também verificam se o software atende às necessidades reais dos usuários e do ambiente onde será usado.



O QUE É TESTE DE SOFTWARE?

Os testes podem ser feitos de duas formas principais:

Testes dinâmicos, onde o programa é executado para ver como se comporta.

Testes estáticos, que analisam o código e os documentos sem rodar o software.







Além disso, testar não é só apertar botões e esperar resultados. Exige planejamento, organização e pensamento crítico para garantir que o produto final seja confiável.

A norma internacional ISO/IEC/IEEE 29119-1 contém diretrizes sobre como os testes devem ser feitos corretamente.



PARA QUE SERVE O TESTE DE SOFTWARE?

Os testes têm vários objetivos, entre eles:

-  Revisar documentos como requisitos e código do sistema.
-  Encontrar erros e falhas antes que o usuário perceba.
-  Garantir que todas as partes importantes do sistema sejam testadas.
-  Reduzir o risco de problemas graves no software.
-  Confirmar se o sistema segue as regras definidas.
-  Checar se o programa está de acordo com contratos e leis.



PARA QUE SERVE O TESTE DE SOFTWARE?

- ✓ Ajudar os responsáveis pelo projeto a tomar decisões com base em informações confiáveis.
- ✓ Criar confiança de que o software funciona bem.
- ✓ Validar se o sistema atende às expectativas dos usuários.

Os objetivos podem mudar dependendo do tipo de sistema, do momento em que o teste é feito e das necessidades do negócio.



POR QUE OS TESTES SÃO NECESSÁRIOS?

- Testar um software é essencial para garantir que ele funcione corretamente e atenda às expectativas. Isso ajuda a evitar problemas que poderiam trazer prejuízos, atrasos e até falhas graves.
- Os testes fazem parte do controle de qualidade, ajudando a manter o equilíbrio entre tempo, custo e qualidade. Mas não é só a equipe de testes que pode contribuir: qualquer pessoa envolvida no projeto pode ajudar a identificar problemas e melhorar o produto.



COMO OS TESTES CONTRIBUEM PARA O SUCESSO?

- ✓ Ajudam a encontrar erros de forma mais barata – Quanto mais cedo os defeitos forem encontrados, mais fácil (e barato) será corrigi-los.
- ✓ Ajudam na tomada de decisões – Os testes mostram se o software está pronto para ser usado ou se precisa de mais ajustes antes de ser lançado.
- ✓ Dão voz aos usuários – Testadores garantem que o software seja feito pensando nas necessidades dos usuários, mesmo quando não há como envolver os próprios usuários no desenvolvimento.
- ✓ Atendem a regras e contratos – Em alguns casos, os testes são exigidos por leis, contratos ou normas de segurança.



TESTE X GARANTIA DA QUALIDADE (QA): QUAL A DIFERENÇA?

Muitas pessoas confundem "teste" com "garantia da qualidade" (QA), mas são coisas diferentes:

- Teste faz parte do controle de qualidade (QC), ou seja, é uma forma de verificar se o produto final tem qualidade suficiente.
- QA (Garantia da Qualidade) é um conjunto de práticas para melhorar os processos de desenvolvimento e evitar erros antes mesmo que eles aconteçam.

💡 Resumo rápido:

Teste (QC) = Corrige erros no produto.

QA = Melhora os processos para evitar erros no futuro.

Ambos são importantes e se complementam para garantir um software confiável.



7 PRINCÍPIOS DE TESTE

1 – O teste mostra a presença, não a ausência de defeitos

- ◆ Testar mostra que tem problema, mas não garante que está tudo certo. Mesmo que você teste e não ache nenhum erro, isso não quer dizer que o sistema está perfeito. Testes ajudam a encontrar defeitos, mas não conseguem provar que o sistema está 100% livre de problemas. É como revisar um texto e não achar nenhum erro, pode ainda ter algum que passou despercebido.

2 – Testes exaustivos são impossíveis

- ◆ Testar tudo é impossível. É muito difícil (quase impossível) testar todas as combinações e possibilidades de um sistema. Então, a gente precisa escolher o que é mais importante testar primeiro, com base nos riscos, no que é mais usado ou no que tem mais chance de dar problema.



7 PRINCÍPIOS DE TESTE

3 – Testes antecipados economizam tempo e dinheiro

- ◆ Começar os testes cedo economiza tempo e dinheiro. Se a gente encontra e corrige os problemas no início do projeto, eles não causam mais problemas lá na frente. Consertar um erro logo no começo sai muito mais barato do que deixar pra depois. É como notar que a receita do bolo tá errada antes de começar a cozinhar, bem melhor do que descobrir só depois que o bolo queimou!

4 – Os defeitos se agrupam

- ◆ A maioria dos erros está concentrada em poucos lugares

Normalmente, uma pequena parte do sistema tem a maior parte dos problemas.

Isso é comum e esperado.

Então, faz sentido prestar atenção especial nessas partes que costumam “dar mais trabalho”.



7 PRINCÍPIOS DE TESTE

5 – Os testes se degradam

- ◆ Testes repetidos perdem a força. Se você sempre faz os mesmos testes, com o tempo eles deixam de ser tão úteis pra achar defeitos novos.

Pra evitar isso, é importante atualizar os testes, mudar os dados ou criar testes diferentes.

Mas em alguns casos, repetir os testes pode sim ser bom, como nos testes automáticos que garantem que o sistema não quebrou depois de uma mudança.

6 – Os testes dependem do contexto

- ◆ O jeito de testar depende do projeto. Não existe uma forma única de testar que sirva pra todos os sistemas. Cada projeto tem suas particularidades, e os testes precisam se adaptar ao tipo de software, ao prazo, aos riscos, à equipe etc. É como escolher roupa: depende da ocasião, do clima e do gosto.



7 PRINCÍPIOS DE TESTE

7 – Falácia da ausência de defeitos.

- ◆ Um sistema sem erros ainda pode ser ruim. Mesmo que o sistema não tenha nenhum erro técnico, ele pode não servir pra nada se não atender ao que o usuário precisa.

Ou seja, é possível ter um sistema todo certinho, mas que ninguém quer usar porque não resolve o problema real.

Por isso, além de verificar se tudo funciona, é essencial validar se o sistema faz sentido pro usuário e pro negócio.



ATIVIDADES DE TESTE

Mesmo que o jeito de testar mude dependendo da situação, existem algumas etapas que quase sempre fazem parte de um bom processo de testes. Seguir essas etapas ajuda a garantir que o teste realmente funcione e traga os resultados esperados. Esse conjunto de etapas é chamado de processo de teste. Ele pode ser adaptado de acordo com o tipo de sistema que está sendo testado, os objetivos da empresa, o tempo disponível, entre outros fatores.

Existe uma norma internacional chamada ISO/IEC/IEEE 29119-2 que traz mais detalhes sobre como esse processo de testes deve funcionar.



ATIVIDADES E TAREFAS DE TESTE

Testar um sistema envolve várias etapas importantes. Essas etapas nem sempre seguem uma ordem certinha, como se fosse uma receita de bolo. Às vezes, elas acontecem ao mesmo tempo ou são repetidas conforme o projeto vai avançando. Aqui estão as principais:

1 – Planejamento do Teste: É o momento de pensar no que se quer com o teste e escolher a melhor forma de chegar lá, levando em conta o tempo, o orçamento e as necessidades do projeto.

2 – Monitoramento e Controle: Enquanto os testes estão acontecendo, é preciso acompanhar tudo de perto para ver se está indo como o planejado. Se algo sair do esperado, são tomadas decisões para ajustar o rumo e garantir que os objetivos do teste sejam alcançados.

3 – Análise do Teste: Aqui, a equipe estuda tudo o que será testado para entender melhor o sistema. A ideia é descobrir **o que deve ser testado**. Também se avalia o risco de problemas e se identifica onde é mais importante focar.



ATIVIDADES E TAREFAS DE TESTE

4 – Modelagem do Teste: Depois de saber o que testar, é hora de pensar em **como testar**. Nessa etapa, são criados os casos de teste (ou seja, os cenários que vão ser usados para testar o sistema). Também são definidos os dados que serão usados e tudo que será necessário para os testes funcionarem, como ferramentas e ambientes.

5 – Implementação do Teste: Aqui, o time coloca a mão na massa: monta os testes, prepara os dados e cria os roteiros (manuais ou automatizados) para que tudo possa ser executado. Também é nessa fase que se organiza a ordem dos testes e se monta o ambiente onde eles serão feitos.

6 – Execução do Teste: Agora sim é a hora de **rodar os testes**! Pode ser tudo feito manualmente ou com ajuda de ferramentas. Depois da execução, os resultados são comparados com o que era esperado. Se algo der errado, os problemas são analisados para entender o motivo.



ATIVIDADES E TAREFAS DE TESTE

7 – Conclusão do Teste: No fim do processo, tudo é revisado: o que foi feito, o que deu certo, o que deu errado, se ainda ficou algo pendente... Essas informações são registradas num relatório final, que é compartilhado com as pessoas interessadas. O ambiente de teste é encerrado, e tudo o que pode ser útil no futuro é guardado.



PROCESSO DE TESTE NO CONTEXTO

Os testes não acontecem sozinhos.

Os testes fazem parte de todo o processo de criação de um sistema dentro de uma empresa. Eles não são feitos de forma isolada. Pelo contrário, são planejados para ajudar a empresa a atingir seus objetivos de negócio, ou seja, o que os clientes ou donos do projeto realmente precisam. Por isso, a forma como os testes são feitos pode mudar bastante, dependendo da situação. Veja alguns exemplos do que pode influenciar nos testes:




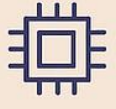

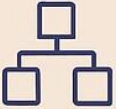


- **Quem está envolvido (stakeholders):** O que essas pessoas esperam do sistema? Elas ajudam no processo? Quais são os requisitos delas?
- **Quem faz parte da equipe de testes:** As pessoas têm experiência? Conhecem bem o projeto? Precisam de treinamento?
- **O tipo de sistema que está sendo testado:** É algo crítico, como um sistema de hospital ou banco? Existem regras de mercado ou leis a seguir?
- **Parte técnica:** Qual tecnologia foi usada? Como o sistema foi construído?
- **Limitações do projeto:** Quanto tempo temos para testar? Qual é o orçamento? Quantas pessoas estão disponíveis?
- **A forma como a empresa trabalha:** Quais são as regras internas? Como a equipe costuma trabalhar?
- **O processo de desenvolvimento:** O sistema é feito por etapas? Tudo é feito junto com o cliente (como no ágil)? Isso influencia como os testes acontecem.
- **Ferramentas disponíveis:** Existem ferramentas de teste? São fáceis de usar? Estão de acordo com as regras da empresa?



ESSES FATORES VÃO IMPACTAR VÁRIAS DECISÕES IMPORTANTES, COMO:

- Qual estratégia de teste será usada
- Quais técnicas vão ser aplicadas
- Se os testes serão manuais ou automatizados
- Quanto do sistema será testado
- O quanto será documentado
- E como os relatórios de teste serão feitos.

FATORES QUE INFLUENCIAM OS TESTES DE SOFTWARE

	Stakeholders	Necessidades e expectativas das partes interessadas.
	Equipe	Habilidades e experiência dos membros da equipe.
	Domínio do negocio	Criticidade e necessidades do objeto de teste.
	Aspectos tecnicos	Tipo de software e tecnologia utilizada.
	Restrições do projeto	Escopo, tempo e orçamento do projeto.
	Estrutura organizacional	Estrutura e politicas da organização.
	Ciclo de vida do desenvolvimento	Praticas de engenharia e desenvolvimento utilizadas.
	Ferramentas	Disponibilidade e usabilidade das ferramentas.



DÚVIDAS?
SUGESTÕES?
RECLAMAÇÃO?
ELOGIO?



VAMOS JUNTOS(A) NESSA!

