

Teste de Software

INTRODUÇÃO À QUALIDADE DE SOFTWARE



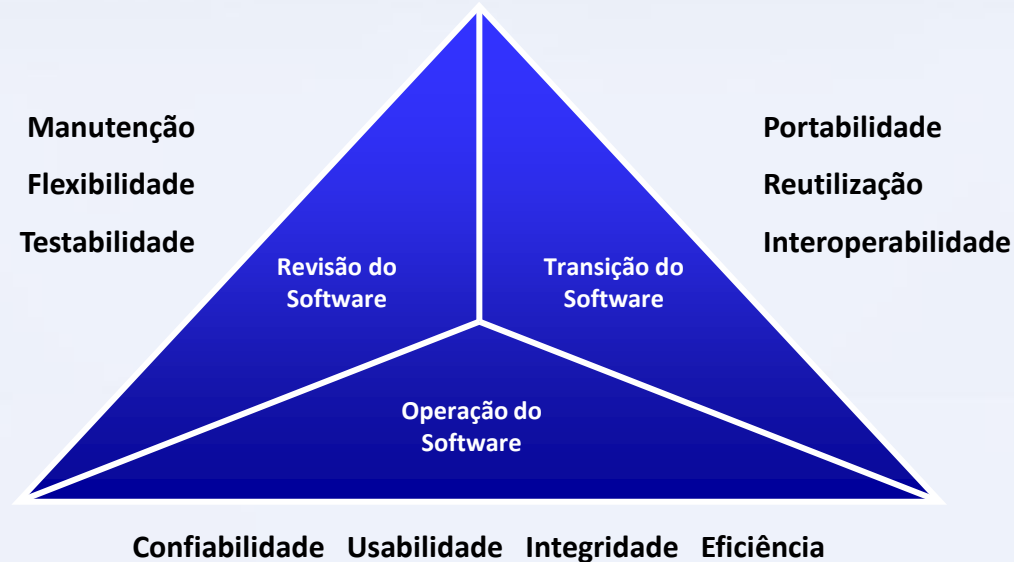
COTI Informática
Escola de Nerds



1. O QUE É QUALIDADE DE SOFTWARE?

1. O QUE É QUALIDADE DE SOFTWARE?

O objetivo principal do processo de Testes é fornecer informações sobre a **qualidade** de um software. A qualidade de um software é medida a partir do grau em que o mesmo atende aos requisitos inicialmente estabelecidos.



Dentre os fatores que afetam a qualidade de um software, podemos citar:

- **Confiabilidade:** Ele se comporta com precisão o tempo todo?
- **Eficiência:** Ele rodará em meu hardware tão bem quanto possível?
- **Usabilidade:** Ele foi projetado para o usuário?
- **Flexibilidade:** Posso mudá-lo?
- **Portabilidade:** Serei capaz de usá-lo em outra máquina?
- **Reusabilidade:** Serei capaz de reutilizar parte do software?
- **Interoperabilidade:** Serei capaz de compor uma interface com outro sistema?

1. O QUE É QUALIDADE DE SOFTWARE?

Terminologias

Defeito é uma imperfeição de um produto. O defeito faz parte do produto e, em geral, refere-se a algo que está implementado no código de maneira incorreta.

A palavra “defeito” não significa apenas um problema que faz um programa não funcionar. Um software defeituoso “não funciona como deve”.

O uso do termo **bug** para descrever defeitos inexplicáveis foi parte do jargão da engenharia por várias décadas; pode originalmente ter sido usado na engenharia mecânica para descrever maus funcionamentos mecânicos. Diz-se que o termo foi criado por Thomas Edison quando um inseto causou problemas de leitura em seu fonógrafo em 1878, mas pode ser que o termo seja mais antigo.

Falha é o resultado errado provocado por um defeito ou condição inesperada. Os defeitos podem existir, mas nem sempre ser visíveis, sua descoberta geralmente é exposta através de falhas apresentadas pelo software.

São exemplos de situações de ocorrências de defeitos ou falhas:

- O Software não faz algo que a especificação do produto diz que faz;
- O Software faz algo que na especificação diz que não é para fazer;
- O Software faz algo que a especificação do produto não menciona;

1. O QUE É QUALIDADE DE SOFTWARE?

Verificação e Validação

O objetivo do processo de **Verificação** é provar que produto de software está sendo construído da forma correta, avaliando se o que foi planejado foi realmente realizado. Visa verificar se os requisitos inicialmente estabelecidos foram implementados.

O processo de **Validação** visa avaliar se que foi entregue atende às expectativas do cliente, ou seja, se os requisitos foram implementados de maneira a melhor atender às necessidades do cliente do produto. Avaliar se o sistema é realmente aquilo que o cliente pagou para ter.

Verificação

Fizemos o software
corretamente?

Validação

Fizemos o software
correto?



2. QUALIDADE DO PRODUTO DE SOFTWARE.

2. QUALIDADE DO PRODUTO DE SOFTWARE.

Alguém que esteja determinado a produzir um excelente produto enfrenta dois problemas.

Como saber quando o produto é bom o suficiente?

Se o produto ainda não for bom o suficiente, como garantir que os envolvidos saibam disso?

A resposta à primeira pergunta permite liberar o release do produto. A resposta à segunda pergunta ajuda a evitar o release de um produto insatisfatório.

Paradigmas de Bom o Suficiente

A maioria dos negócios pratica alguma forma de raciocínio do tipo bom o suficiente em relação a seus produtos. Os únicos que não fazem isso são aqueles que acreditam terem atingido a perfeição, pois não conseguem ver como seus produtos podem ser melhorados.

Aqui estão alguns modelos do tipo bom o suficiente que já foram experimentados. Alguns deles são mais eficientes que outros, dependendo da situação:

Não Tão Ruim ("ainda não estamos liquidados")

Nossa qualidade só precisa ser boa o suficiente para que possamos permanecer no negócio. Fazê-lo bom o suficiente para não acabarmos sendo processados.

Infalibilidade Positiva ("tudo que fazemos é bom")

Nossa empresa é a melhor do mundo. Como somos bons, tudo que fazemos é automaticamente bom. Pense apenas no sucesso. Não pense no fracasso, porque o pensamento "negativo" atrai má qualidade.

2. QUALIDADE DO PRODUTO DE SOFTWARE.

Exaustão Justificada ("perfeição ou fiasco")

Nenhum produto é bom o suficiente; o esforço é que conta. E apenas a exaustão completa representará um nível de esforço bom o suficiente. Não temos de nos preocupar com as questões comerciais. Faremos todo o possível para tornar o produto perfeito. Como nunca vamos deixar de tentar melhorá-lo, alguém terá de vir até nós e implorar por ele, se o quiser. Assim, a responsabilidade por qualquer problema de qualidade será deles, e não nossa.

O Cliente está Sempre Certo ("parece que os clientes gostaram do produto")

Se gostaram, é porque deve ser bom o suficiente. É claro que você não pode agradar a todo mundo ao mesmo tempo. E se um cliente atual ou em potencial não gostar do produto, cabe a eles nos informar. Não podemos ler suas mentes.

Processo Definido ("nós seguimos um Processo Eficiente")

A qualidade é o resultado do processo que usamos para criar o produto. Definimos nosso processo e entendemos que é ele é eficiente. Portanto, se o seguirmos, o resultado inevitavelmente será um produto bom o suficiente.

Requisitos Estáticos ("nós satisfazemos aos Requisitos")

Definimos qualidade em termos de metas objetivas, quantificáveis e que não sejam polêmicas. Se atingirmos essas metas e tivermos um produto bom o suficiente, qualquer outra meta subjetiva, não-quantificável ou polêmica poderá ser sugerida.

Compromisso ("nós cumprimos nossas promessas")

A qualidade é definida por contrato. Prometemos realizar determinadas ações e atingir certas metas. Se cumprirmos nosso contrato, ela será boa o suficiente.

2. QUALIDADE DO PRODUTO DE SOFTWARE.

Defesa ("nós nos esforçamos ao máximo")

Defendemos a excelência. Durante todo o projeto, procuramos maneiras de evitar problemas, além de localizar e corrigir aqueles que não podemos evitar. Se trabalharmos com afinco rumo à excelência, isso será bom o suficiente.

Compensação Dinâmica("pesamos vários fatores")

Com relação à nossa missão e à situação em questão, um produto é bom o suficiente quando não tem problemas críticos, quando tem vantagens suficientes e elas compensam esses problemas, e quando é mais prejudicial que benéfico continuar a melhorá-lo.

A alta qualidade é necessariamente mais cara?

Dependendo de vários fatores (como processo, habilidade, tecnologia, ferramentas, ambiente e cultura), você pode conseguir produzir um produto de muito melhor qualidade pelo mesmo custo do que seria possível de outro modo. Um produto com maior capacidade de teste e manutenção custará menos para ser melhorado; outros custos são especificamente associados à má qualidade, como custos de suporte e custos para o cliente.

O custo da qualidade é uma questão complexa, o que torna difícil fazer generalizações amplas. Contudo, sempre é possível gastar mais tempo realizando testes muito mais abrangentes, efetuando um tratamento de erros mais adequado e corrigindo ou reescrevendo cada parte do produto. Por mais eficiente que você seja, isso terá um preço. E se você não consegue pensar em mais melhorias, é provável que tenha atingido o limite de sua imaginação, e não de qualidade.



3. CONCLUSÃO

Há várias medidas que podem ser usadas para obter o senso de qualidade, mesmo que você não possa medi-la de forma completa e objetiva. Mesmo assim, a questão do que é uma qualidade boa o suficiente requer um julgamento sofisticado.

Você não pode escapar do fato de que, no final, as pessoas precisam avaliar e fazer um julgamento. Para um produto simples, esse julgamento pode ser fácil. Para um produto complexo com grandes limitações, é muito difícil.