


**CONSEQUÊNCIAS DE UM
SOFTWARE DE MÁ QUALIDADE E
VANTAGENS DA REALIZAÇÃO DE
TESTES POR ESPECIALISTAS DE
TESTES**

QUALIDADE (VISÃO DO USUÁRIO)

O cliente valoriza que o software responda às suas necessidades.

1. As funções X estão disponíveis e são executadas eficientemente?
 2. Funciona corretamente em imprevistos, como, por exemplo, efetuar débito em uma conta com saldo negativo?
 3. O software é seguro, ou seja, evita que pessoas ou sistemas não autorizados tenham acesso aos dados para modificar?
 4. É fácil de usar ou requer muito treinamento?
 5. É fácil de integrar com outros sistemas existentes?
 6. Aceita trabalhar com arquivos de outros sistemas ou enviar dados para outros sistemas?
- 

ERRO

O Instituto de Engenheiros Eletricistas e Eletrônicos (IEEE) - padrão IEEE nº 610.12-1990

A diferença entre **um valor ou condição computada**, observada ou medida e o valor ou condição verdadeira, especificada ou **teoricamente correta**.

```
def soma(a, b):  
    return a * b
```

```
print( soma(1, 2) )
```



DEFEITO

Documento IEEE Std 1044-2009

Defeito (*defect*): uma imperfeição ou deficiência em um produto de trabalho em que o produto de trabalho não atende a seus requisitos ou especificações e precisa ser consertado ou substituído.

Um defeito (*fault*) é a manifestação de um erro em um software. Também é uma etapa, processo ou definição de dados incorreto que pode gerar uma falha.



FALHA

IEEE Std 1044-2009

Definição A: rescisão da capacidade de um produto executar uma função exigida ou sua incapacidade de executar dentro dos limites especificados anteriormente (adaptado da ISO/IEC 25000:2005).

Definição B: um evento no qual um sistema ou componente do sistema não executa uma função necessária dentro dos limites especificados (adaptado da ISO/IEC 24765:2009).

Um Software falha quando não cumpre seu propósito.



EXCEÇÃO

É um evento que causa a suspensão da execução normal de um programa. Segundo o IEEE Std 610.12-1990, isso inclui exceção de endereçamento, exceção de dados, exceção de operação, exceção de overflow, exceção de proteção e exceção de underflow.

```
def get_config(file_name):  
    with open(file_name, 'r') as config:  
        return config.readlines()
```

```
# O arquivo buscado foi renomeado por alguém para local.cfg  
get_config('local_config.cfg')
```

QUANDO FALHAS OCORREM

Softwares que não funcionam corretamente, podem levar a muitos problemas como **prejuízos financeiros, perda de reputação, risco em relação a integridade física das pessoas, multas contratuais...** E até consequências maiores como **lesões corporais e mortes** (softwares aeronáuticos por exemplo) e até desastres tecnológicos como um mal funcionamento de um sistema de controle de mísseis!



PORQUE FALHAS OCORREM?

- **Pressão** do tempo de entrega: acaba com que as atividades sejam feitas e entregues sem o devido cuidado
- Falhas de **comunicação** e falta de **requisitos** ou mudanças
- Pessoas com pouco ou **sem treinamento** em suas funções
- Processo de desenvolvimento **imaturo**



EXEMPLO CLÁSSICO

- Espaço-nave Mercury
- linguagem de programação FORTRAN
 - Código escrito: `DO 100i = 1.10`
 - Código esperado: `DO 100i = 1,10`
 - Resultado: Explosão

QUAL A IMPORTÂNCIA DE UM ESPECIALISTA?

O seu exame teria um desfecho melhor nas mãos de um clínico geral ou de um especialista?

Menos defeitos = menor chance do software falhar

