**Fundamentos Do Teste De Software - Falha, Defeito E Erro**

Até um tempinho atrás, metade dos anos 90, poucas empresas utilizava softwares o que atualmente é raro! Softwares estão presentes em quase todos os lugares, na padaria de bairro até grandes e poderosos estabelecimentos como aeroportos, bancos. E todos nós já passamos por algum tipo de falha de software como insdisponibilidade de software e sites, lentidão, etc.

**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**

Todos os produtos de desenvolvimento de software, desde a especificação e requerimentos até a documentação geral, são escritos por pessoas, e as pessoas são propensas a cometer erros, independentemente de suas experiências e qualificações. Podemos listar outros itens como:

* 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

**Você sabe a diferença entre erro, defeito e falha?**

**Erro**: é uma ação humana que produz um resultado incorreto (e pode ser cometido em qualquer fase do desenvolvimento).  
**Defeito**: é a manifestação de um erro no software, também conhecido como ***bug*** e se executado, o defeito pode causar uma falha - É o resultado do erro cometido.  
**Falha**: diferença indesejável entre o observado e o esperado (defeito encontrado)

Um software pode conter defeitos mas mesmo assim nunca falhar.  
Falha é um evento.  
Defeito é um estado do software causado por um erro.

[](https://backefront.com.br/posts/Figura1_ErroFalhaDefeito.jpg)Diferença entre os 3 termos apresentados

**Um exemplo clássico**  
Você conhece o caso da espaçonave [Mercury](https://pt.wikipedia.org/wiki/Mercury-Atlas_1)?  
Vamos lá, o programa de computador a bordo da espaçonave continha a declaração escrita na linguagem de programação FORTRAN:  
**Código escrito**: DO 100i = 1.10  
**Código esperado**: DO 100i = 1,10

Basicamente, por causa de uma vírgula, a espaçonave explodiu!

**Como o teste pode nos ajudar a construir sofwares melhores?**

Testes rigorosos em documentos e no próprio software podem reduzir os riscos de problemas acontecerem em produção.  
**Menos defeitos = menor chance do software falhar**  
Testes propiciam o aumento de qualidade do software à medida que defeitos encontrados são corrigidos.  
Ajuda a aumentar receitas das empresas desenvolvedoras de software (redução de retrabalho devido à correção de *bugs*)

**E como o teste consegue nos auxiliar nessas três questões?**

Encontrando e corrigindo defeitos antes dos software ser liberado para uso operacional!

No próximo post vamos comentar sobre Qualidade & Teste e também Quanto testar?