

Qualidade de Software e Testes de Unidade

Bibliografia

- IBM. Minimizing code defects to improve software quality and lower development costs, 2008. Disponível em: <ftp://ftp.software.ibm.com/software/rational/info/do-more/RAW14109USEN.pdf>. Acesso em: 2018.
- PRESSMAN, R. S. **Engenharia de software**: Uma Abordagem Profissional. 7a. ed. Porto Alegre: Bookman, 2011.
- SHALLOWAY, A.; TROTT, J. R. **Explicando padrões de projeto**: uma nova perspectiva em projeto de orientado a objeto. Porto Alegre: Bookman, 2004.

Impactos de falhas de software

- **Falha em software deixou supermercados fechados**

QUA, 07 DE MARÇO DE 2012 17:00 ACESSOS: 433



Curtir



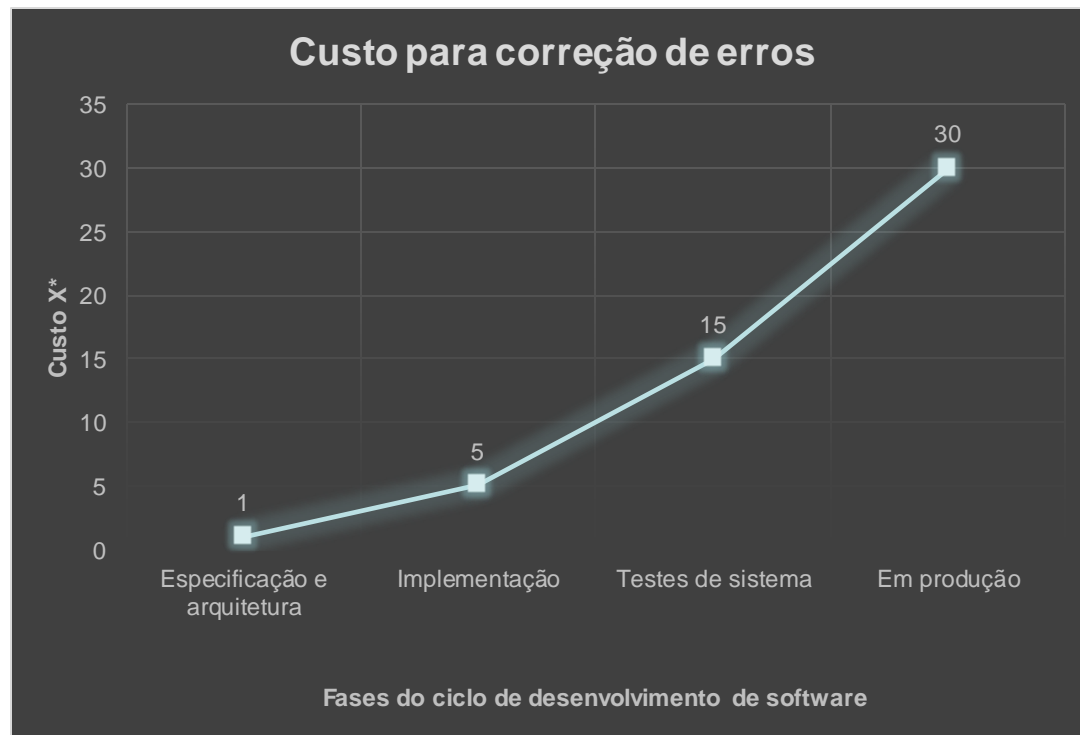
Seja o primeiro de seus amigos a curtir isso.

Um problema no software utilizado por alguns supermercados de Linhares causou transtorno nesta quarta-feira (07), quando simplesmente parou de funcionar. As lojas permaneceram fechadas até às 10h40.

De acordo com informações de funcionários destes estabelecimentos o software que é usado nos caixas dos supermercados deu uma “pane” e parou de funcionar.

- Máquina de terapia radiológica Therac-25

Custos de correção



Quanto mais cedo os erros forem detectados no ciclo de desenvolvimento, é possível reduzir de forma significativa o custo para desenvolver o software

Fonte: (IBM, 2008)

*X é uma unidade de medida de custo que pode ser expressa em termos de horas, ou reais, etc.

Qualidade

- Conceito de “qualidade” usado pela indústria:
 - “Um produto com qualidade é um produto que cumpre com sua especificação” (CROSBY, 1979)
- Teste de software
 - Teste é a execução do software de maneira controlada para avaliar se ele se comporta ou não conforme o especificado.
 - Utilizamos os testes para melhorar a qualidade de um software.

Tipos de Testes

- Testes de Sistema
 - Execução do sistema sob o ponto de vista do usuário, indo em busca de falhas em relação aos requisitos propostos.
Os testes são executados em condições próximas da realidade do usuário, isto é, ambiente de hardware, restrições de recursos, volume de dados, etc.
- Testes de Unidade (ou testes unitários)
 - Testam-se unidades, isto é, partes individuais (funcionalidades pequenas e específicas) de software.
 - Em OOP, a “unidade” é o método
 - Realizado por desenvolvedores, para verificar se o software faz o que deve fazer
 - Vamos construir testes de unidade para testar classes da camada de negócio

Plano de testes

- Testes precisam ser planejados
 - Construção de um “plano de testes”
- Um plano de testes é um documento para registrar formalmente o planejamento
- O plano de testes pode conter: propósito, identificação, itens a serem testados, critérios de aceite, documentos produzidos (como logs, relatórios, etc.), ambiente a ser testado.
- Um plano de testes contém um conjunto de “casos de testes”
 - Um caso de teste possui uma descrição de um teste particular
 - Cada caso possui um conjunto de dados de entrada diferente.

Exemplo de plano de testes

- Considerar a classe:

Calculadora
+ somar(num1 : int, num2 : int) : int + subtrair(num1 : int, num2 : int) : int + multiplicar(num1 : int, num2 : int) : int + dividir(num1 : int, num2 : int) : int

- Um plano de testes:

Plano de testes PL01 – Validar funcionamento da classe Calculadora			
Caso	Descrição	Entrada	Saída esperada
1	Validar soma de números inteiros	Somar números 10 e 75	Deve resultar em 85
2	Validar subtração de inteiros	Subtrair 35 de 100	Deve resultar em 65
3	Validar multiplicação de números inteiros	Multiplicar 5 por 8	Deve resultar em 40
4	Validar divisão de números inteiros	Dividir 10 por 2	Deve resultar em 5.