

# **Engenharia de Software II / Qualidade e Teste de Software**

## **Aula 03: Processo Básico de Teste de Software**

**Breno Lisi Romano**

**<http://sites.google.com/site/blromano>**

**Instituto Federal de São Paulo – IFSP São João da Boa Vista  
Bacharelado em Ciência da Computação – BCC (ENSC6)  
Tecnologia em Sistemas para Internet – TSI (QTSI6)**



**INSTITUTO FEDERAL DE  
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA  
SÃO PAULO  
Campus São João da Boa Vista**

# Revisão: Falhas vs. Defeito vs. Erro

- **Falha:**
  - Incapacidade do software de realizar a função requisitada
- **Defeito (Bug):**
  - Causa de uma falha: Um defeito em um sistema pode ocorrer devido a omissão de informações, definições de dados ou comandos/instruções incorretas dentre outros fatores
- **Erro:**
  - Pode resultar em defeito, se propagado até a saída
  - Pode ser causado por uma ação humana



# Revisão: Teste de Software

- Representam uma oportunidade de detectar **defeitos** antes do software ser entregue aos usuários
- A atividade de testes pode ser feita de forma **manual** e/ou **automática** e tem por objetivos:
  - **Detectar Erros** para **Eliminar os Defeitos** e **Evitar as Falhas**
  - **Produzir casos de teste** que tenham elevadas probabilidades de **revelar um defeito ainda não descoberto**, com uma quantidade mínima de tempo e esforço
  - **Comparar o resultado dos testes com os resultados esperados** → produzir uma indicação da qualidade e da confiabilidade do software. Quando há diferenças, inicia-se um processo de depuração para descobrir a causa



# Revisão: Estágios de Teste

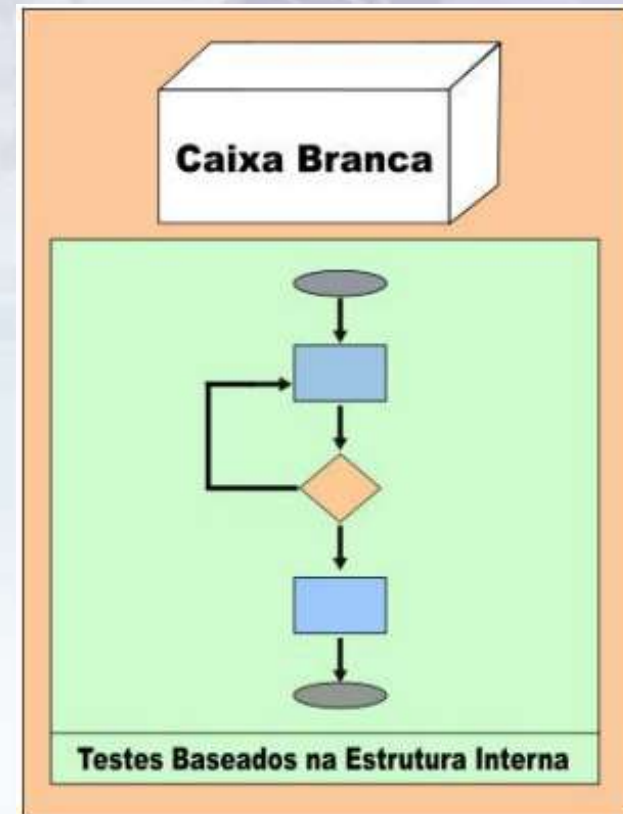
- Existem diferentes estágios de teste associados ao desenvolvimento de um produto de software:
  - Teste de Unidade
  - Teste de Integração
  - Teste de Sistemas
  - Teste de Aceitação (Homologação)

# Revisão: Abordagens de Teste

- **Abordagem Funcional (Caixa Preta):**



- **Abordagem Estrutural (Caixa Branca):**



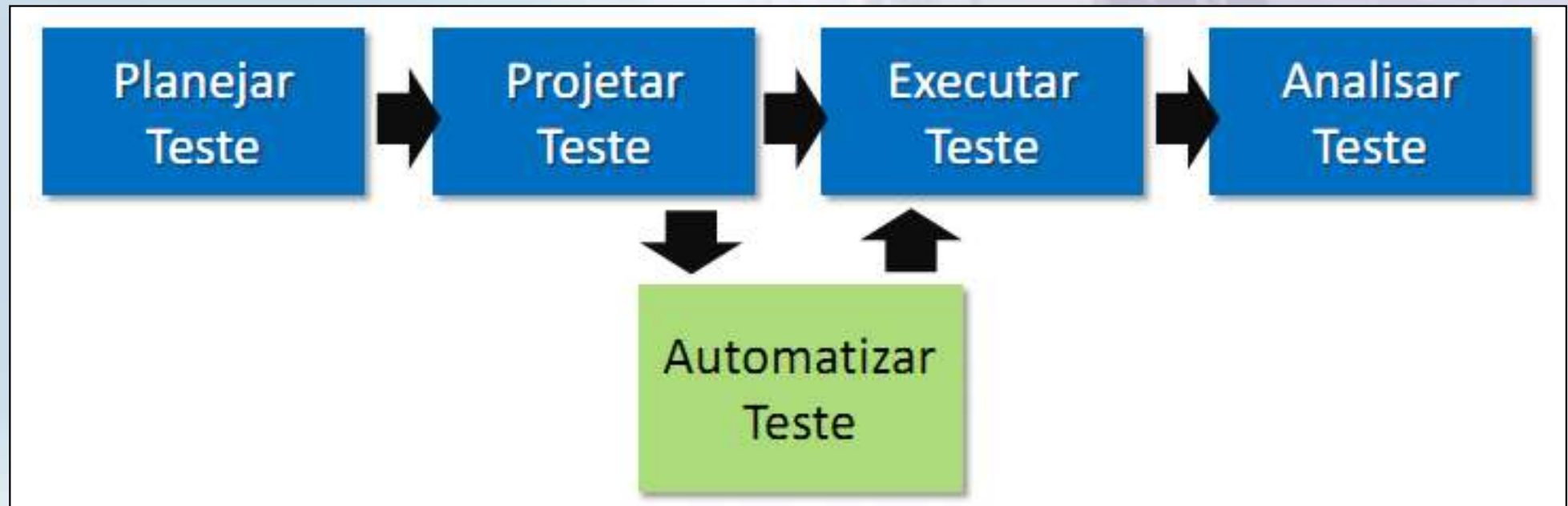


# Revisão: Tipos de Teste de Software

- Existem vários tipos de teste que podem ser executados nos diversos estágios de teste e utilizando as diferentes abordagens existentes:
  - **Teste de funcionalidade**
  - **Teste de recuperação de falhas**
  - **Teste de Segurança de acesso**
  - **Teste de Carga**
  - **Teste de Desempenho**
  - **Teste de Portabilidade**
  - **Teste de Regressão**



# Processo Básico de Teste de Software







# Planejar Testes (1)

**Identificar requisitos para testes**

O que será testado



Escopo dos testes





# Planejar Testes (2)

**Avaliar Riscos**



**Maximizar efetividade dos testes**

Pergunta a ser respondida: Os testes que estamos definindo são os melhores testes para encontrar novos defeitos ou continuamos encontrando os mesmos defeitos?



# Planejar Testes (3)

## Desenvolver Estratégia Testes

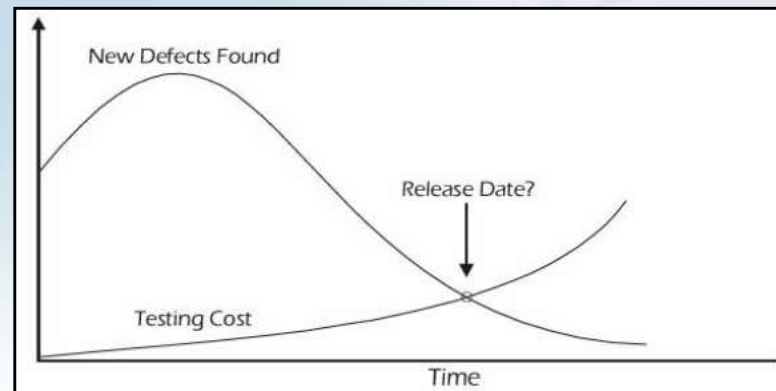
Estabelecer as técnicas e as ferramentas

Métodos de avaliação para determinar o final dos testes. Quando parar de testar??



# Planejar Testes (4)

- Uma das maneiras de saber quando deve-se dar por encerrado a etapa de testes é verificar os seguintes pontos (deve-se deixar bem claro no planejamento quando parar):
  - **Custo x Benefício:** O tempo médio entre defeitos encontrados é muito alto e gasta-se muito recurso para testar em relação a quantidade de defeitos identificados



- O número de bugs achados que estão fechados é maior do que o número de bugs que se esperava achar
- Todos os testes foram executados
- A porcentagem de cobertura da aplicação pelos testes já é o suficiente
- Todas as funcionalidades funcionam corretamente
- O sistema atende as métricas de confiabilidade definidas no Plano de Teste
- O número e severidade dos bugs caíram a um nível satisfatório



# Planejar Testes (5)

## Identificar Recursos

Recursos humanos, hardware, software de testes

## Criar Cronograma

Estimativa de esforço de teste e cronograma das atividades de teste



# Projetar Testes (1)

**Identificar** conjunto de **casos de teste** para cada  
*Build / Sprint*

**Identificar** procedimentos de teste que mostram  
como **casos de teste** são realizados



## Projetar Testes (2)

Identificar e descrever casos de teste

Identificar condições e dados de teste

Identificar resultados esperados



# Projetar Testes (3)

- **Exemplo RUIM:**
  - **Caso de Teste: Debitar Conta**
    - Verificar, em uma conta válida, o débito de uma quantia em reais da mesma
  - **Resultado Esperado:**
    - A conta escolhida deve apresentar um saldo que corresponde a quantia inicial – débito realizado
    - Uma mensagem mostrando o sucesso da operação deve ser apresentada





# Projetar Testes (4)

- **Exemplo BOM:**
  - **Caso de Teste: Debitar Conta**
    - A conta num. 22321-3 existe no sistema e está válida
    - O banco constatou que foi feita uma retirada no valor de R\$ 200,00 desta conta
    - A conta num. 22321-3 têm saldo de R\$650,00
  - **Resultado Esperado:**
    - A conta num. 22321-3 agora deve apresentar saldo de R\$450,00
    - Uma mensagem mostrando o sucesso da operação deve ser apresentada com os seguintes dizeres: “O Débito foi realizado com sucesso.”



# Projetar Testes (5)

## Caso de Teste: CT#01: Débito em Conta com Sucesso

Descrição do Caso de Teste:	Resultados Esperados:
Este caso de teste verifica o débito em uma conta válida e com saldo positivo, permitindo a retirada da quantia solicitada.	O débito de R\$200,00 na conta será permitido!  Novo Saldo da Conta: R\$450,00  Mensagem Apresentada ao Usuário ao Realizar o Débito: O Débito foi realizado com sucesso.
<b>Critério de Teste:</b> Funcional - Cenário	
<b>Entradas:</b>	
Número da Conta: 22321-3	
Saldo da Conta: R\$650,00	
Valor do Débito: R\$200,00	



# Automatizar Testes

Criar **scripts de teste reutilizáveis** que implementam e **executam casos de teste** e seus **procedimentos de teste**

**Excelente para os testes de regressão**



# Executar Testes

Executar Casos de Teste Manualmente ou Scripts  
Automatizados em componentes do software

Executar procedimentos de teste

**Verificar resultados dos testes executados  
contra os esperados**

**Identificar e registrar os erros**



# **Analisar Testes**

**Avaliar resultados de teste**

**Avaliar métricas de teste**

**Estabelecer ações de melhoria com base na  
avaliação**

# Engenharia de Software II / Qualidade e Teste de Software

## Aula 03: Processo Básico de Teste de Software

Breno Lisi Romano

**Dúvidas?**

<http://sites.google.com/site/blromano>

Instituto Federal de São Paulo – IFSP São João da Boa Vista

Bacharelado em Ciência da Computação – BCC (ENSC6)

Tecnologia em Sistemas para Internet – TSI (QTSI6)



INSTITUTO FEDERAL DE  
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA  
SÃO PAULO  
Campus São João da Boa Vista