

# Técnicas e níveis de teste de software

# Recapitulando

- \* Sabemos o que são testes de software
- \* Também sabemos para que utilizamos testes de software
- \* Mas ainda não temos muita idéia de como realizar este processo

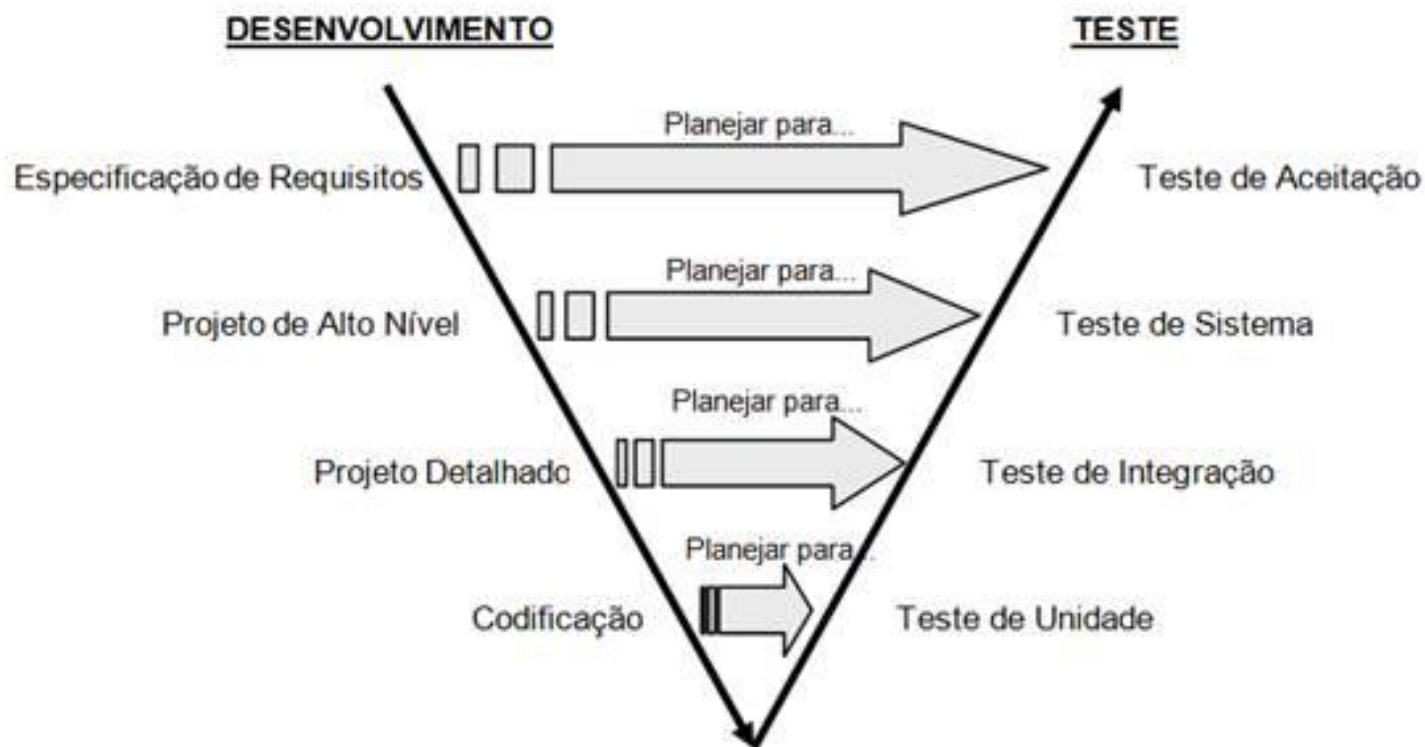
# Testes e processos de software

- Testes podem ser utilizados em qualquer processo de software
- Cada processo define a forma como os testes serão utilizados
- Além de oferecer diferentes níveis de importância aos testes

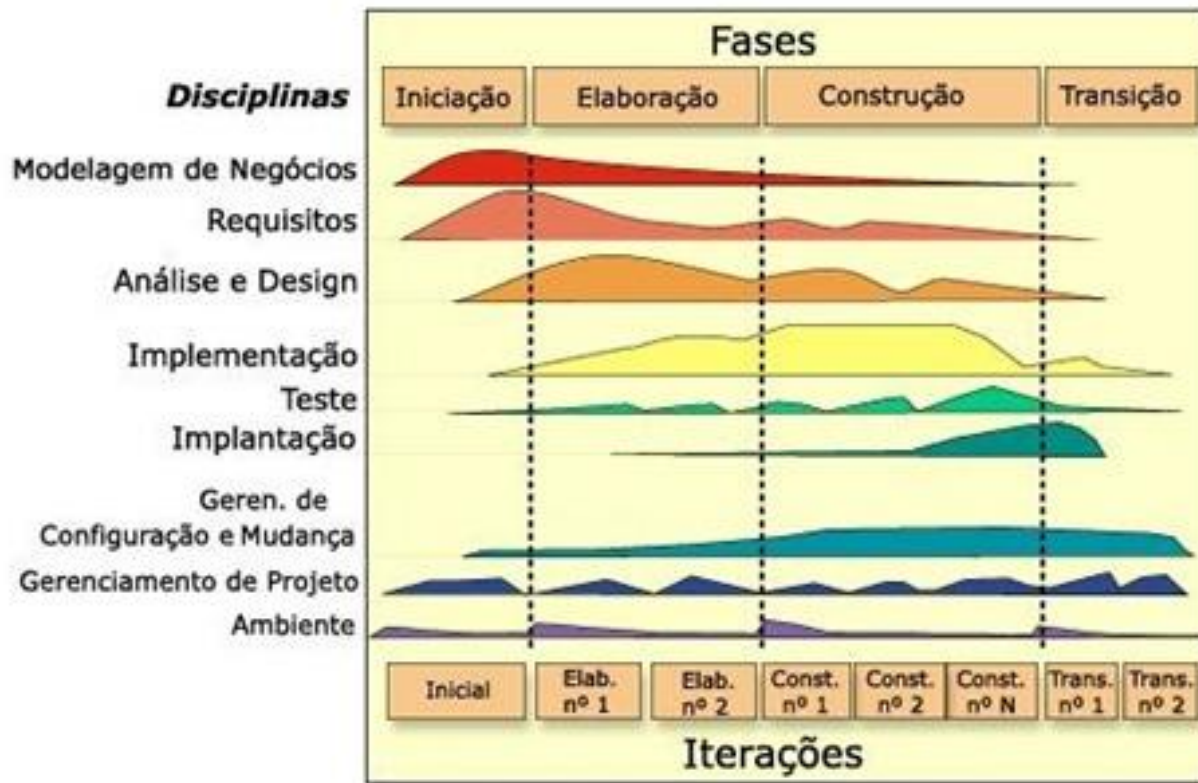
# Modelo cascata



# Modelo v



# Rational unified process (rup)



# Testes e RUP

- São realizados em várias fases do projeto, minimizando a probabilidade de serem encontrados no futuro
- Há um maior planejamento dos testes, o que trás uma maior segurança ao nosso software
- Designa uma equipe para tratar unicamente de testes

# Princípios do teste de software

1. Teste demonstra apenas a presença dos erros
2. Teste exaustivo é impossível
3. Agrupamento de defeitos (Regra do 80 x 20)
4. Testar depende do contexto



# Princípios do teste de software

1. Saber o que o cliente **espera** como **resultado**
2. Um programador nunca deve testar seu próprio programa
3. Uma equipe de software não deve testar os softwares que produz
4. Inspeccionar bem os resultados dos testes
5. Devemos testar nosso software considerando entradas válidas e inválidas

# Princípios do teste de software

1. Devemos verificar se o software não faz o que deveria fazer e se faz o que deveria fazer
2. Devemos testar partes de nosso software que irão se manter durante todo o software
3. Não planeje testes considerando que não encontrará erros
4. A probabilidade de mais erros serem encontrados é igual a quantidade de erros encontrados
5. Testar é uma atividade que requer criatividade e intelectualidade

# Papeis e responsabilidades

- \* Gerente de testes
- \* Analisa de teste
- \* Testador

# O Que testamos em um software?

- \* Funcionalidades
- \* Segurança
- \* “Velocidade”
- \* Usabilidade

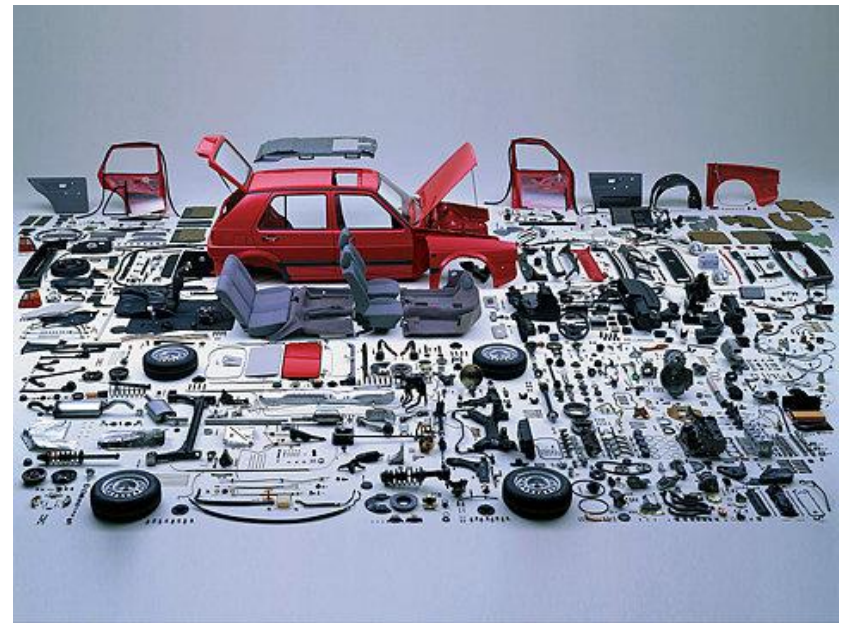
# Como testamos

- \* Usando o sistema como um usuário final
- \* Durante o processo de programação

# Técnicas de teste de software



**Testes caixa preta**



**Testes caixa branca**

# Teste caixa preta

- \* A forma como o software foi implementado não nos interessa
- \* Estamos interessados em identificar:
  - \* Requisitos funcionais implementados incorretamente
  - \* Erros de interface
  - \* Erros de desempenho
- \* Pode ser realizado por nosso cliente ou pelos desenvolvedores

# Exemplo

1. Para um sistema bancário, a operação de saque produz diferentes respostas de acordo com o valor a ser sacado e o saldo em conta
2. Sabendo que o saldo existente é de R\$ 2.000 e que há limites diários de R\$ 1.000
3. Tentaremos realizar saques de R\$ 999, R\$ 1.000 e R\$ 1.001
4. Iremos observar se o sistema a) autoriza o saque, b) autoriza o saque e c) rejeita o saque



# Teste caixa branca

- \* O código utilizado no desenvolvimento do software será analisado
- \* Estamos interessados em avaliar:
  - \* Se as condicionais são válidas
  - \* Testes de caminho lógico
  - \* Verificar se funções/métodos retornam os resultados esperados
- \* Deve ser executado pelo programador
  - \* Trás uma confiança sobre o código utilizado

# Diferenças entre o teste caixa branca e caixa preta

- \* No teste caixa preta verificamos se as funcionalidades estão implementadas conforme solicitado pelo cliente
- \* No teste caixa branca o código-fonte utilizado para implementar as funcionalidades do sistema é analisado

# Níveis de teste de software

- \* Testes também são classificados quanto a parte do software que está sendo testada
- \* A classificação dada a estas partes é a seguinte:
  - \* Teste de aceitação
  - \* Teste de sistema
  - \* Teste de integração
  - \* Teste unitário

# Níveis de teste de software

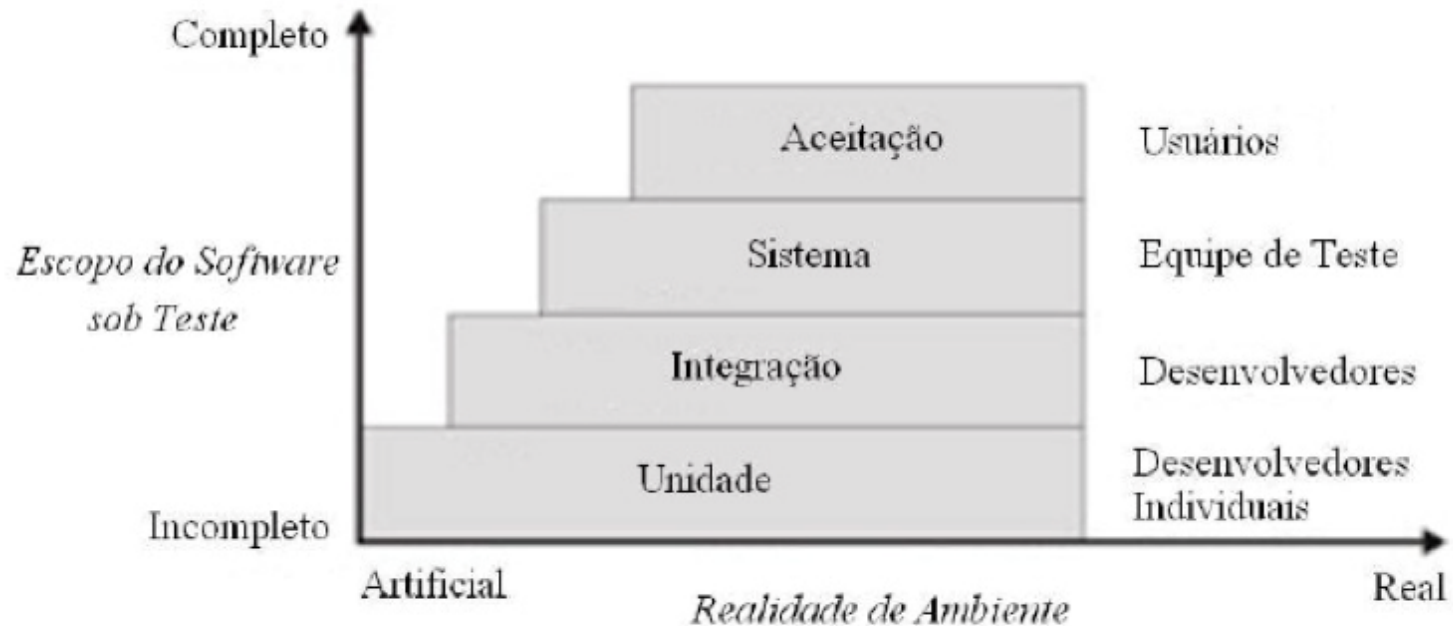
- \* Aceitação
- \* Sistema
- \* Integração
- \* Unitário



Teste caixa preta

Teste caixa branca

# Níveis de teste de software



**Figura 5 – Níveis de Teste versus Realidade de Ambiente.**

# Tipos de teste de software

- \* Teste funcional
- \* Teste de interface
- \* Teste de performance
- \* Teste de regressão
- \* Teste de configuração
- \* Teste alfa e beta

# Tipos de teste de software

- \* Teste de aceitação do usuário
- \* Teste de instalação
- \* Teste de segurança
- \* Teste de carga
- \* Teste de stress

# Ciclo de vida dos testes





# Custo da qualidade de software

