



# Testes de Software

Introdução à Testes de Software





# David Brandão

- SR Test Developer - SiDi
- Automated Testing Enthusiast
- Bachelor's degree in computer engineering - UniFBV
- Agile Test Specialist - CESAR School
- Ongoing Master's Degree in computer engineering - UPE



# Luiz Henrique

- Test Technician - SiDi
- Test Specialist - USP School
- Graduation in progress in ADS - Anhanguera

# Introdução à Testes de Software

- O que é teste de software?
- Por que precisamos testar?
- O que é qualidade?



<https://www.menti.com/alq3gyexpqgt>

# Introdução à Testes de Software

- O que é teste de software?
  - É um processo onde **avaliamos** e **verificamos** se um software faz o que deveria fazer.

# Introdução à Testes de Software

- Por que precisamos testar?
  - Garantir a qualidade do produto;
  - Melhorar a experiência do usuário;
  - Avaliar a segurança;
  - Validar a conformidade com requisitos;
  - Satisfazer às necessidades do cliente.

# Introdução à Testes de Software

- O que é um software com qualidade?
  - É um software atende às necessidades do cliente, levando em consideração alguns aspectos como:
    - Funcionalidade
    - Confiabilidade
    - Usabilidade
    - Eficiência
    - Segurança

# THE TESTING Manifesto



we value:

Testing  
throughout

**OVER**

testing at  
the end

Preventing  
bugs

**OVER**

finding  
bugs

Testing  
understanding

**OVER**

checking  
functionality

Building the  
best system

**OVER**

breaking the  
system

Team  
responsibility  
for quality

**OVER**

tester  
responsibility

[www.growingAgile.co.za](http://www.growingAgile.co.za)

@growingAgile



# Tipos de Teste - Funcionais

- **Unitário**
  - Código fonte
- **Integração**
  - Módulos e/ou serviços
- **Sistema (E2E)**
  - Comportamento do usuário
  - Ambiente completo
- **Aceitação**
  - Foco nos requisitos
- **Fumaça (Smoke)**
  - Funcionalidades básicas
- **Regressão**
  - Aplicação completa
- **Sanidade (Sanity)**
  - Áreas Restritas
- **Exploratório**
  - Rotinas não convencionais

# Tipos de Teste - Não Funcionais

- **Performance**

- Carga
- Capacidade
- Stress

- **Compatibilidade**

- Navegadores
- Sistemas Operacionais
- Dispositivos Mobile

- **Segurança**

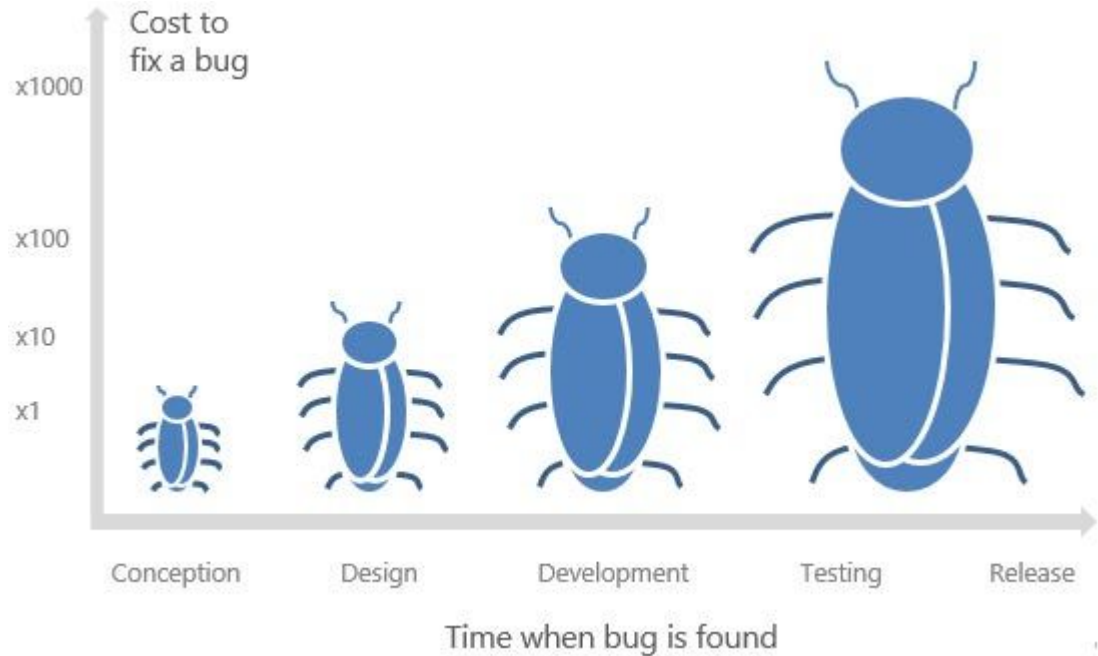
- Proteção de dados
- Controle de acesso
- Pentest

- **Usabilidade**

- Acessibilidade
- Interface
- UX (Experiência do usuário)

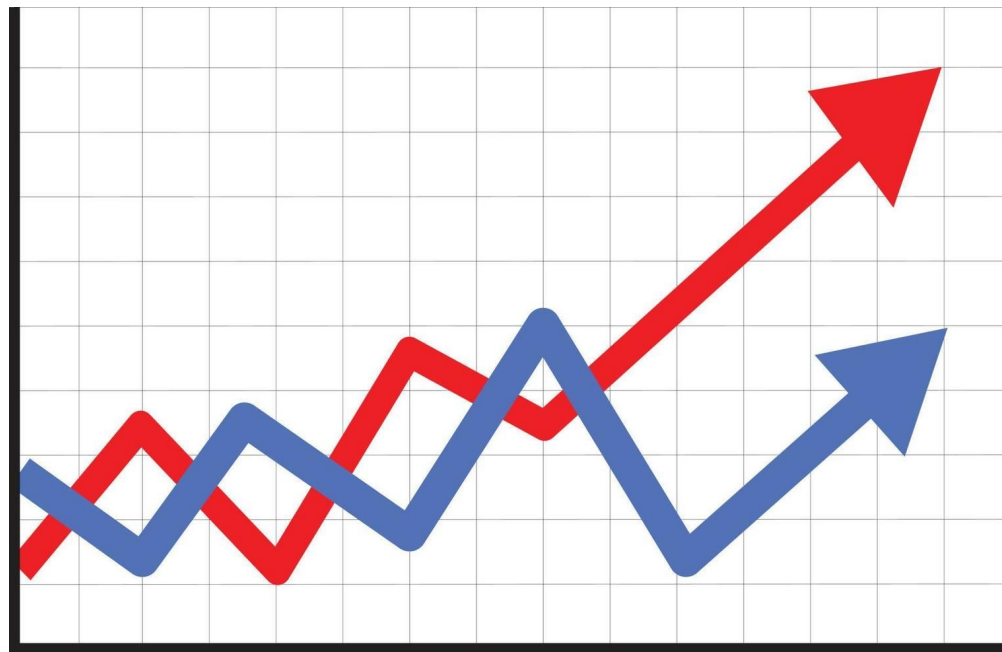
# Importância de Testes em Projetos de Software

- Garantia de Qualidade
- Detecção Precoce de Defeitos
- Confiança do Usuário
- Fácil manutenção
- Melhoria Contínua



# Tendências

- Automação de Testes
  - Crescimento do mercado
  - Inteligência artificial
- Shift-left Testing
- Cultura DevSecTestOps



# Desafios

- Fases iniciais do projeto bem definidas
- Visão da empresa sobre teste de software
- Falta de visão de qualidade no time de desenvolvimento
- Processos e ambientes



# Ciclo de Vida dos Testes

- Planejamento de Testes
  - Definição de estratégias e recursos necessários
- Design de Testes
  - Criação de casos de teste e identificação de dados de teste
- Execução de Testes
  - Implementação dos casos teste

# Ciclo de Vida dos Testes

- Avaliação dos resultados
  - Comparação dos resultados obtidos com os esperados
- Report de falhas
  - Documentação dos defeitos encontrados
- Reteste e regressão
  - Verificação de correções
  - Checagem de possíveis novas falhas introduzidas

# Metodologias de Desenvolvimento

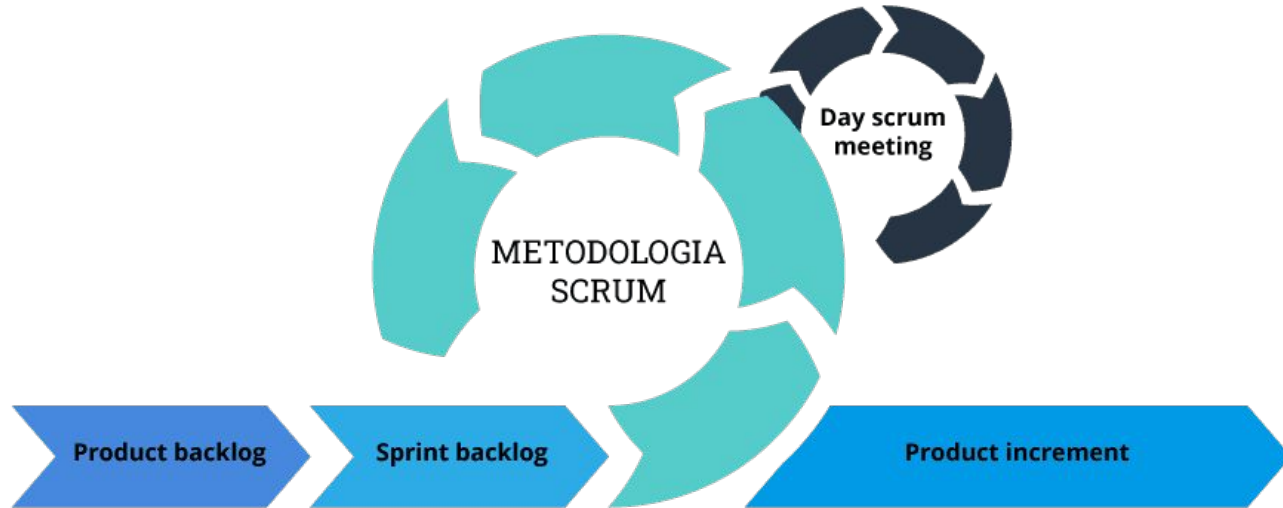
- Ágil
  - Valores:
    - Entrega contínua e incremental
    - Colaborações entre equipes
  - Estruturas:
    - Scrum: Sprints, Reuniões diárias, Revisões
    - Kanban: Fluxo de trabalho, Gestão de tarefas



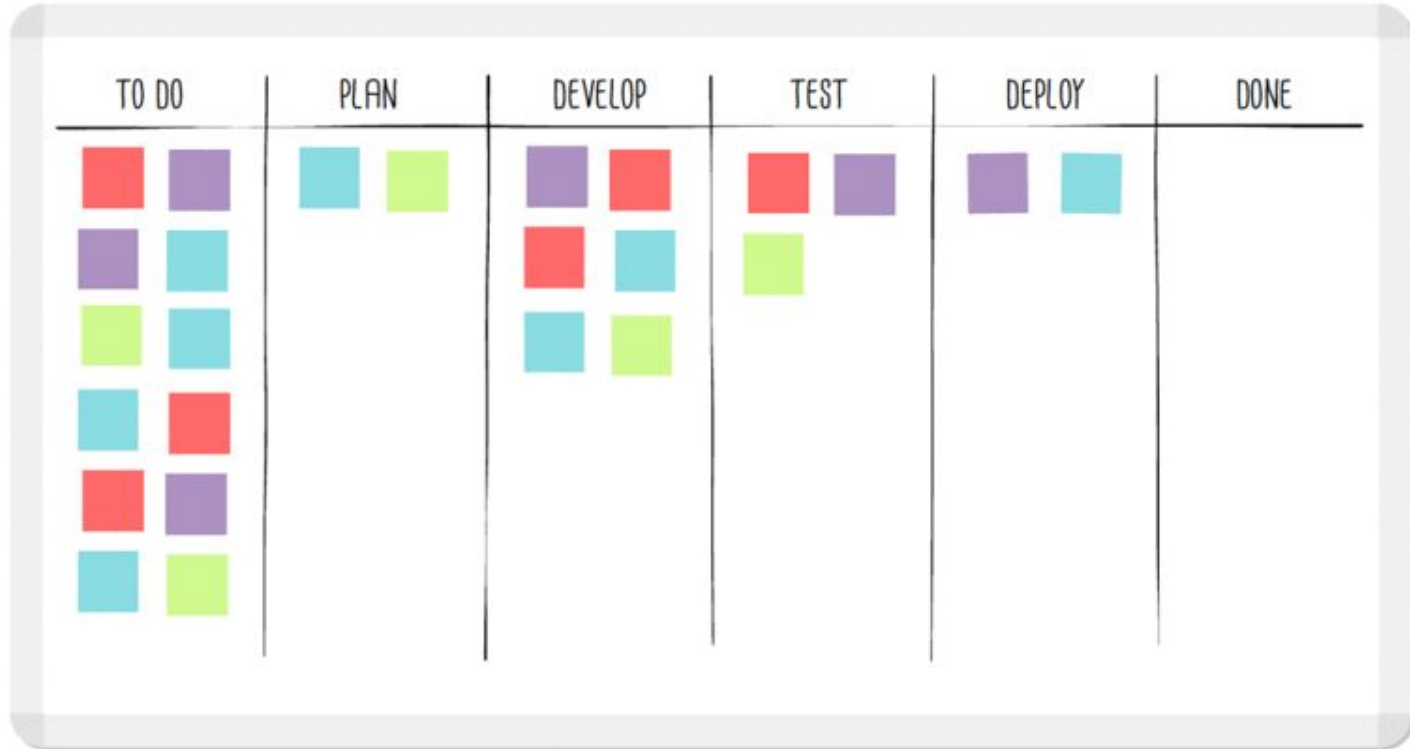
# Metodologias de Desenvolvimento

- Ágil
  - Ciclos de desenvolvimento iterativos
    - Planejamento
    - Desenvolvimento
    - Revisão
    - Iteração contínua

# Scrum



# Kanban



User Story



Defect



Task



Feature

# Metodologias de Desenvolvimento

- Cascata (Waterfall)
  - Etapas sequenciais
    - Requisitos
    - Design
    - Implementação
    - Verificação
    - Manutenção

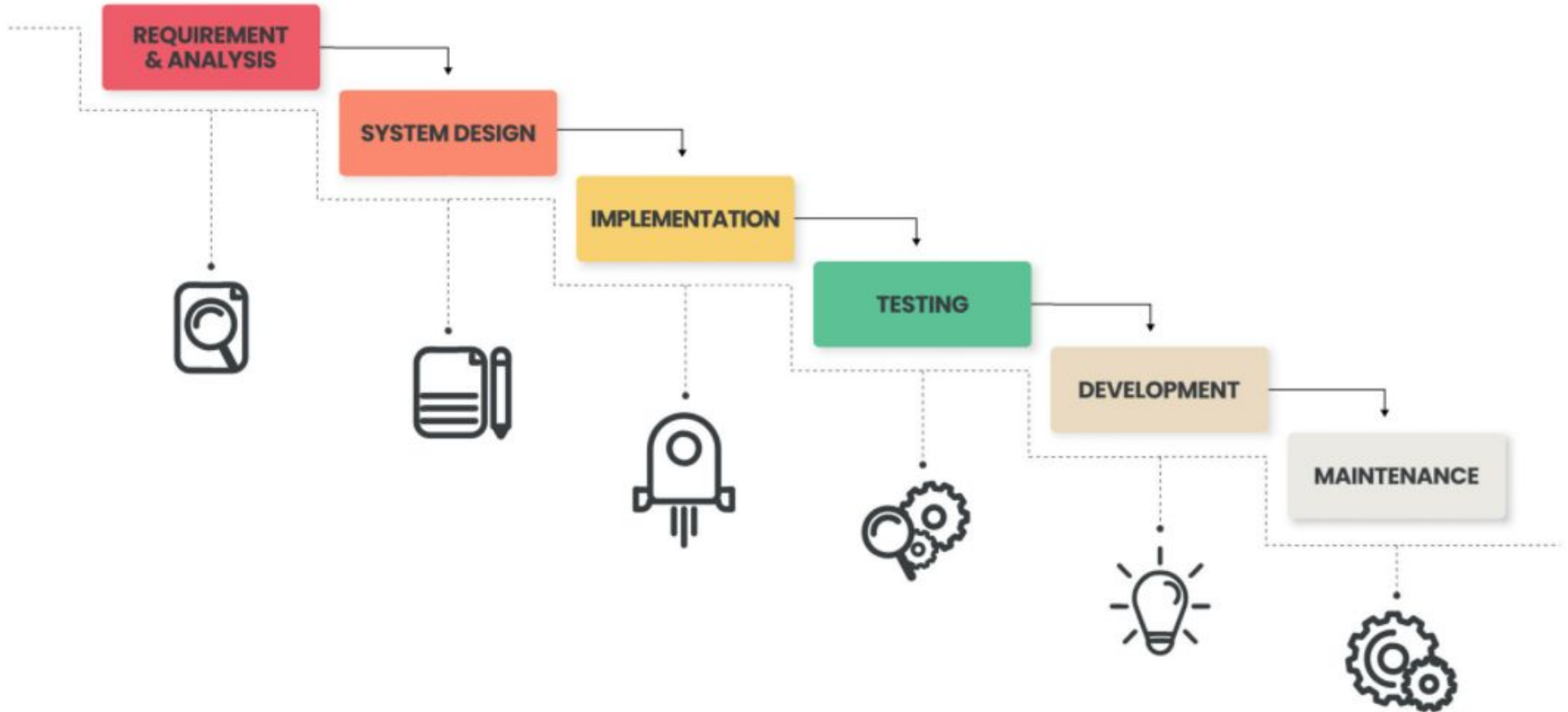
# Metodologias de Desenvolvimento

- Cascata (Waterfall)
  - Vantagens
    - Estrutura clara, fácil gerenciamento
  - Desvantagens
    - Difícil adaptação a mudanças
    - Etapas rígidas

# Metodologias de Desenvolvimento

- Ágil x Cascata
  - Ágil
    - Flexível, Adaptativo, Foco na colaboração
  - Cascata
    - Estrutura linear, Previsível, Foco em documentação

# Cascata



# Contato



<https://www.linkedin.com/in/dav-ferreira/>