



Teste de Software

Prof. Dr. Alan Souza

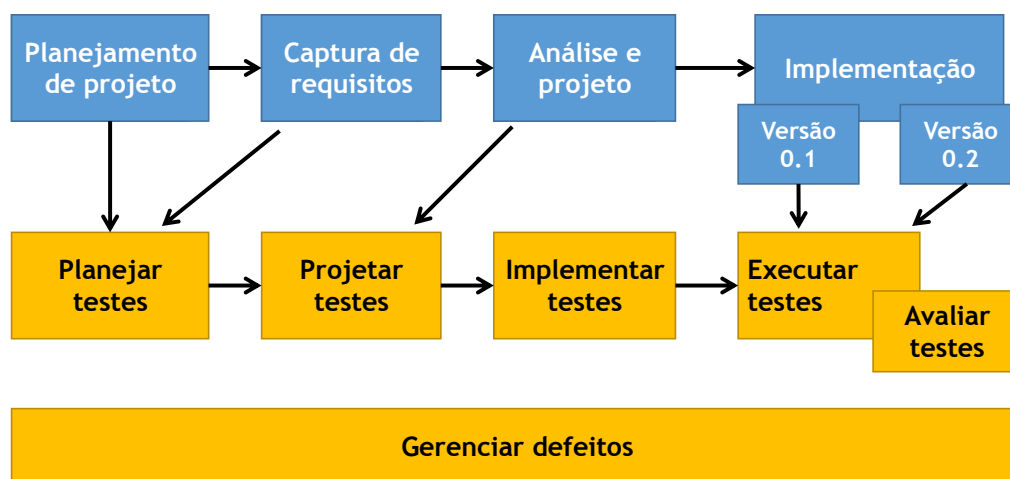
alan.souza@unama.br

2020

5. Processo de teste



- Quando começar a testar?



5. Processo de teste



- O processo de teste é definido como um processo separado, mas intimamente ligado, ao processo de desenvolvimento. Eles têm metas e medidas de sucesso diferentes.
- Por exemplo, quanto menor a taxa de defeitos (n° de casos de teste que falham dividido pelo total de casos de teste), mais bem sucedido é considerado o processo de desenvolvimento (McGregor e Sykes, 2001).
- Por outro lado, quanto maior a taxa de defeitos, considera-se mais bem sucedido o processo de teste (McGregor e Sykes, 2001).

5. Processo de teste



- Fases do processo de teste:
 1. Planejar testes
 2. Especificar/Projetar testes
 3. Executar testes
 4. Reportar testes

5. Processo de teste



5. Processo de teste



• Composição do plano de testes (*saída*):

1. Objetivo
2. Requisitos a serem testados
3. Estágios de Teste
4. Tipos de Teste
5. Abordagens de Teste
6. Critérios de parada/aceitação
7. Recursos
8. Matriz de Responsabilidade
9. Cronograma

5. Processo de teste



5. Processo de teste



• Composição do projeto de testes (*saída*):

1. Requisitos a serem testados(prioridade)
2. Identificador do caso de Teste
3. Requisitos Associados
4. Casos de Teste
5. Tipo de Teste
6. Pré-condição
7. Dados de entrada
8. Procedimento
9. Resultado esperado
10. Status do teste

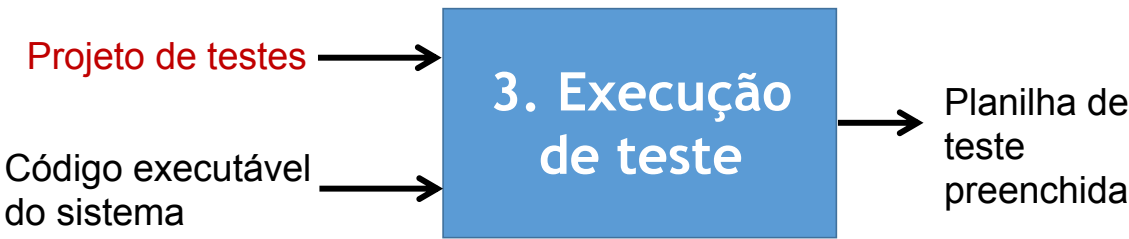
5. Processo de teste



• Exemplo de planilha de teste (saída):

| ID | Módulo | Descrição | Roteiro | Resultado esperado | ... |
|----|---------|-----------------------------------|--|---|-------------|
| 1 | Usuário | Realizar cadastro na loja virtual | 1) Preencher todos os campos do formulário com dados válidos ; 2) Clicar em "Cadastrar". | Inserir os dados no banco de dados e direcionar o usuário para a tela de boas vindas. | . . . |
| 2 | Usuário | Verificar se os dados são válidos | 1) Preencher todos os campos do formulário com dados inválidos ; 2) Clicar em "Cadastrar". | Não inserir dados no banco de dados e mostrar erros de validação na tela. | . . . |

5. Processo de teste



5. Processo de teste



• Exemplo de planilha de teste (saída):

| ... | Descrição | Roteiro | Resultado esperado | Teste feito |
|-----------------|-----------------------------------|--|---|--|
| | Realizar cadastro na loja virtual | 1) Preencher todos os campos do formulário com dados válidos ; 2) Clicar em “Cadastrar”. | Inserir os dados no banco de dados e direcionar o usuário para a tela de boas vindas. | Sim, Maria Silva, 18/03/2020 10:30 |
| | Verificar se os dados são válidos | 1) Preencher todos os campos do formulário com dados inválidos ; 2) Clicar em “Cadastrar”. | Não inserir dados no banco de dados e mostrar erros de validação na tela. | Sim, Maria Silva, 18/03/2020 13:30 |

5. Processo de teste



5. Processo de teste



- Exemplo de planilha de teste (saída):

| ID Teste | Responsável pela correção | Prioridade da correção | Descrição do erro | Teste corrigido |
|----------|---------------------------|------------------------|---|------------------------|
| 1 | José Souza | Alta | O usuário não é direcionado para a tela de boas vindas. | Sim, 19/03/2020 9:00. |
| 2 | José Souza | Baixa | O campo de CPF aceita qualquer número. | Sim, 19/03/2020 10:00. |
| 2 | José Souza | Baixa | Confirmação de senha não funciona. | Não. |



Teste de Software

Prof. Dr. Alan Souza

alan.souza@unama.br

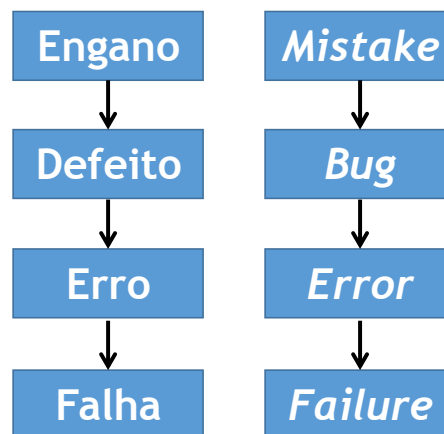
2020

6. Gerenciamento de bugs



Nomenclaturas (*relembrando*):

1. Um **engano** do programador
2. Introduziu um **defeito** no código
3. Quando o software foi executado, gerou-se um **erro**
4. E o software **falha**
5. Mas o erro e a falha só serão detectados se as entradas que as geram forem usadas.



6. Gerenciamento de bugs

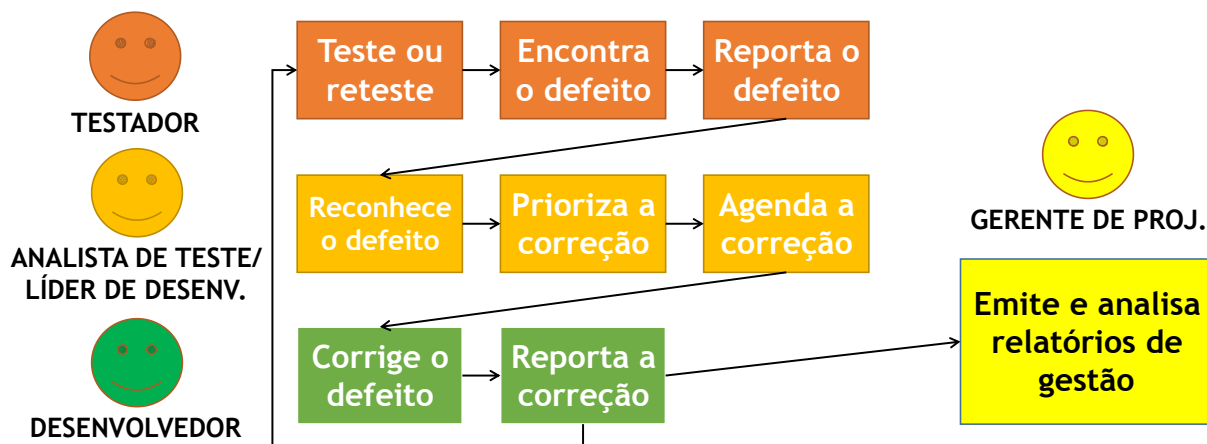


• Classificação de defeitos:

1. **Faltante**: defeito aparece pela uma falta completa ou parcial de um requisito;
2. **Errado**: defeito ocorre porque o requisito foi implementado incorretamente;
3. **Acréscimo**: defeito acontece quando funcionalidades que não foram especificadas são implementadas.

6. Gerenciamento de bugs

- Ciclo de vida de um defeito:



6. Gerenciamento de bugs

- Ferramentas de teste:

Automatizam atividades ligadas aos testes;

Organizam o processo de teste como um todo;

Ferramentas:

- Elaborar planos de testes: MS Excel, MS Project;
- Projetar testes: MS Excel;
- Executar testes: MS Excel;
- Avaliar testes: MS Excel;
- Implementação: JUnit (unidade), JTest (análise estática);
- Gerência de defeitos: Bugzilla, Mantis, Redmine, Github.

Referência



- Material do Professor Dr. Sandro Bezerra (ICEN, UFPA)
- Disponível em: <http://srbo.ufpa.br/> <último acesso: março/2020>



Teste de Software

Prof. Dr. Alan Souza

alan.souza@unama.br

2020

Exercícios



- 1) Baseado em um aplicativo de celular para pesquisa e compra de produtos, faça o que se pede:
 - a) Crie e descreva três requisitos funcionais com as seguintes informações: Requisito, Ator, Descrição, Pré-condição, Pós-condição, Fluxo principal, Validação;
 - b) A partir dos casos de uso, gere o plano de testes com seguintes itens: Objetivo, Requisitos a serem testados, Tipos de Teste, Abordagens de Teste, Critérios de parada/aceitação, Recursos Computacionais, Cronograma;
 - c) Com base no plano de testes, crie a planilha de teste.

Exercícios



- 2) Faça um relatório de como as seguintes ferramentas funcionam para facilitar o controle de bugs:
 - a) Mantis
 - b) Github
- Esse relatório deve conter as seguintes informações: nome da ferramenta; breve descrição (um parágrafo de cinco linhas no mínimo); gratuito? (sim, não, planos de pgto); explicar como os bugs podem ser organizados (mostrar através de print de telas).