

# Qualidade de Software: Perspectiva sobre Testes

Renata Germano Bianchi, M.Sc  
Contribuições de Elis Hernandes, Ph.D



# Sobre mim

- CTFL-AT (Certified Tester Foundation Level – Agile Tester), 2018
  - CTFL (Certified Tester Foundation Level), 2018
  - Ms.C. Ciência da Computação/Interação Humano-Computador, UFSCar, 2016
  - Análise e Desenvolvimento de Sistemas, IFSP, 2013
- 
- Head de QA & Consultora de Qualidade e Processo de Desenvolvimento de Software, MonitoraTec, 2020 - atual
  - Líder de QA, MonitoraTec – 2017 - 2020
  - Analista de Teste, MonitoraTec, 2016 – 2017
  - Trainee em Teste, MonitoraTec, 2013 – 2014
- 
- Apaixonada por mentorar a qualidade do processo e do produto, questionar, analisar e entender como o software pode ser melhor. Não me limito ao software, também dou suporte para o desenvolvimento dos times. Busco a melhoria contínua, pessoal e profissional, diariamente refletindo sobre minhas experiências.



# MOVENDO-SE GLOBALMENTE

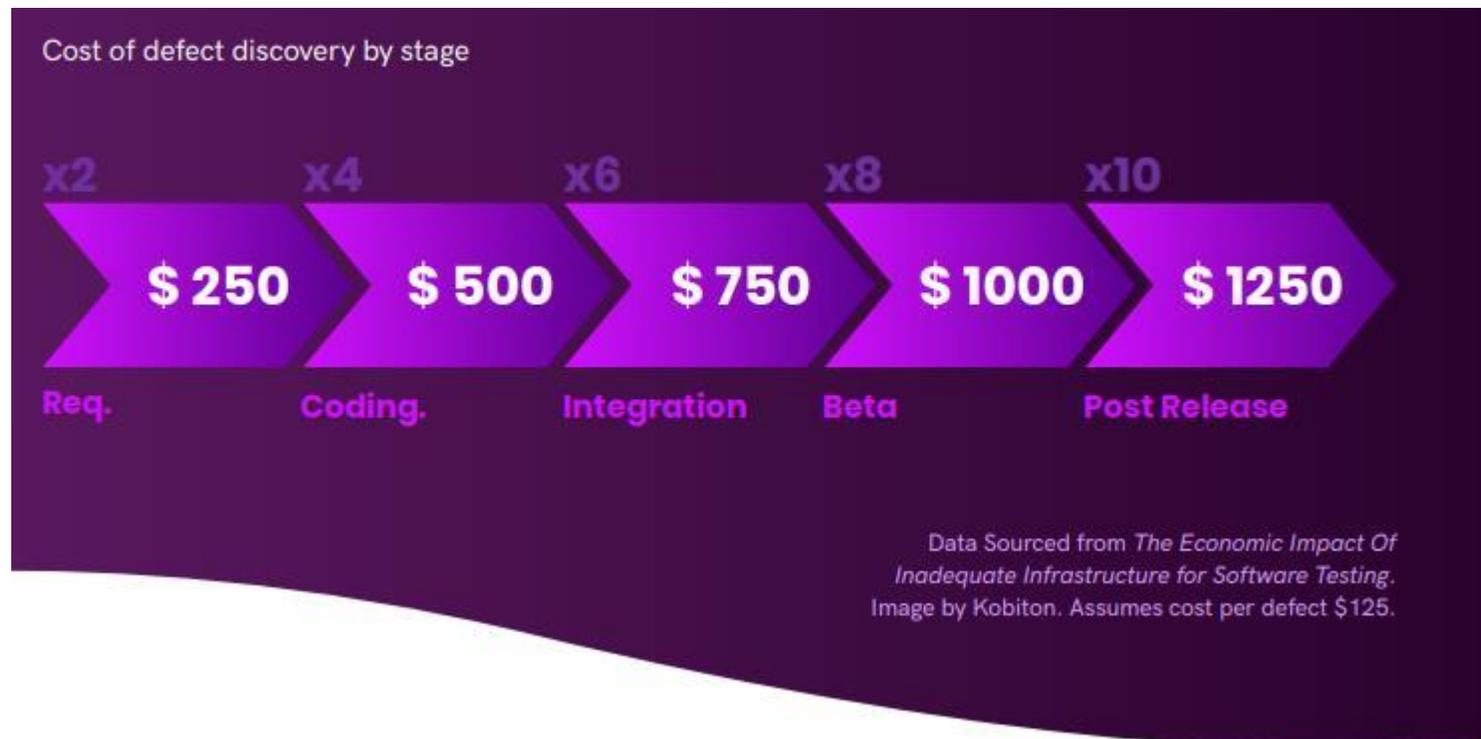
- Nossa missão, apoiar nossos clientes, nacionais e internacionais na jornada da **Transformação Digital**, trabalhando de forma ágil e orientada a **Resultados e Métricas**. Preocupação com ações ESG.
- Apoiamos a **Evolução Digital** de nossos clientes, desde processos de *Customer Experience*, passando pelo **Desenvolvimento das Soluções** com alocação de **squads ágeis**, suporte e sustentação.
- Com um *footprint global*, temos a capacidade de atuar em grandes projetos envolvendo **SalesForce, SAP, DevSecOps, Digital transformation, Data Analytics, Automation RPA+ML+AI**.



# Qualidade de Software

---

# Custo de um bug



# Qualidade de Software

- Você conhece um software de qualidade?
- No que você se baseou para julgar a qualidade do software?
- O que é qualidade de software? Como atingi-la?
- Muito mais do que aquilo que vemos e lemos!
- É uma cultura, um desejo de fazer e ser melhor
  - Requisito, Código, Teste etc



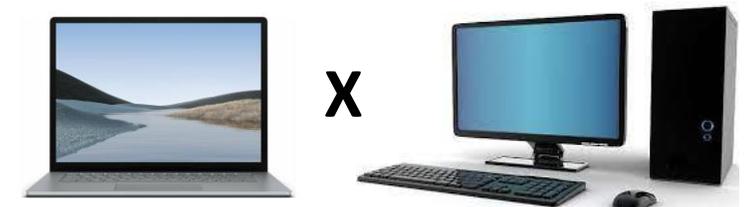
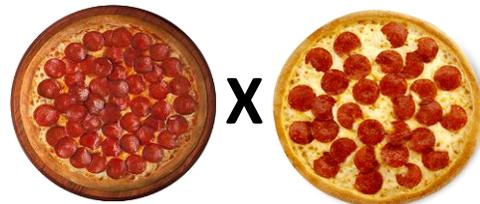
# Qualidade de Software

- Conformidade com requisitos funcionais e não-funcionais, padrões de desenvolvimento documentados, e características implícitas esperadas de todo software profissionalmente desenvolvido: corretismo, confiabilidade e testabilidade. (Pressman, 2011)



# Qualidade de Software

- Públicos diferentes possuem necessidades e desejos diferentes
- Mas nem tudo é subjetivo...
  - Existem normas que definem características e subcaracterísticas relacionadas à qualidade de software (ISO/IEC 25010)



# Qualidade de Software

- Algumas atividades que trazem maior qualidade:
  - Definição e seguimento de um Processo
  - Requisitos claros e bem-definidos
  - Boas práticas de programação
    - Estratégia de branch para desenvolvimento do software
    - Padronização dos commits
    - Teste do funcionamento do próprio código
    - Análise de impacto, performance, escalabilidade, segurança
  - Documentações – requisitos, arquitetura, infra, testes, reuniões
  - Revisão dos artefatos (requisito, código, teste)
  - Ambientes adequados (DEV, QA, PROD)
  - Comunicação
  - Testes

# Teste de Software

---

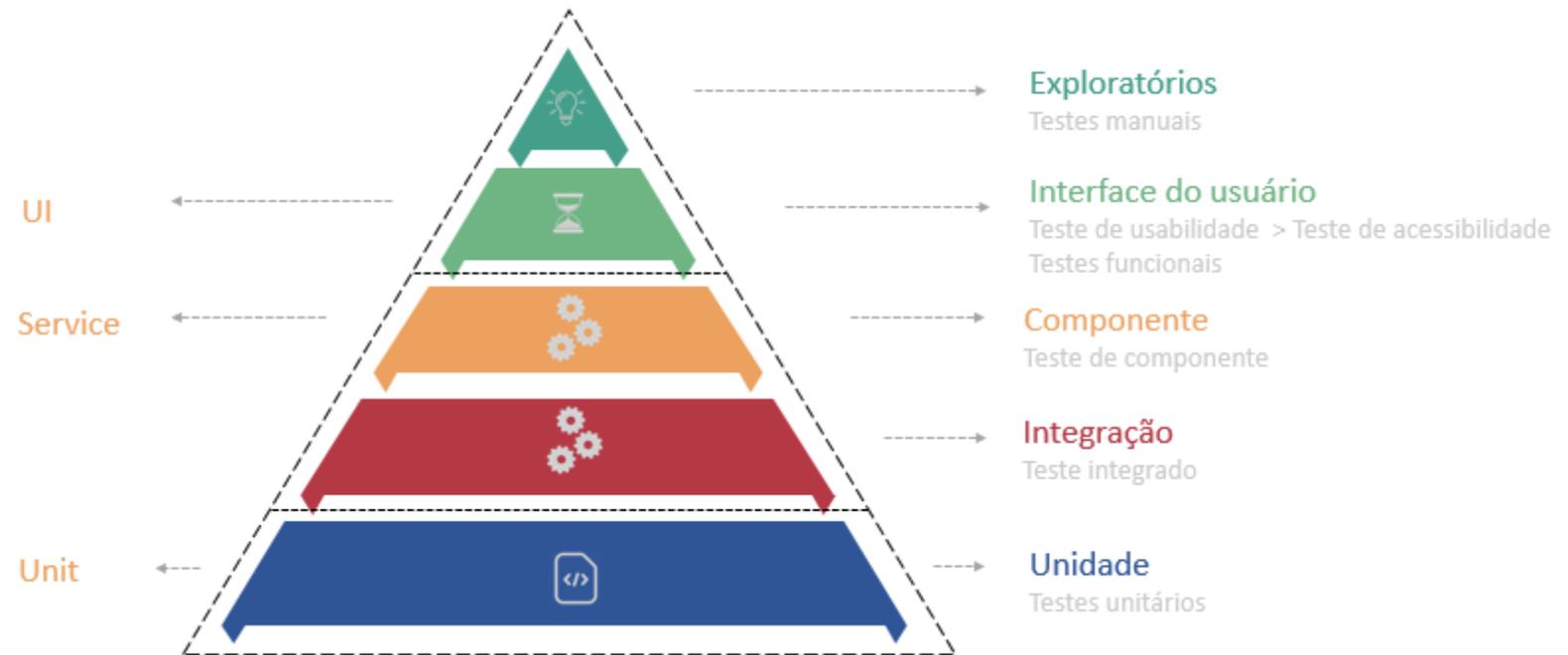


# Desafios

- Não há tempo para o teste exaustivo
- Muitas combinações de entrada para serem exercitadas
- Dificuldade em determinar os resultados esperados para cada caso de teste
- Requisitos do software inexistentes ou que mudam rapidamente
- Não há tempo suficiente para o teste
- Desconhecimento da área de teste ou falta de preocupação com qualidade

# Pirâmide de Teste

- Demonstração dos tipos de testes, níveis, velocidade de implementação e complexidade dos testes realizados



# Teste

- Como não conseguimos (e não existe tempo suficiente para) testar tudo de forma exaustiva, temos:
  - Técnicas de teste
  - Critérios de teste
  - Análise de risco
  - Automação de testes

Coming  
Soon...



# Atividades do Testador (QA)

- Suporte para seguimento do processo
- Levantar riscos e antecipar problemas
- Análise crítica dos requisitos – compreensível e testável por todos (não-técnico)
  - "I'll let you into a secret: if you are struggling to understand, you won't be the only one. But, if no one asks or questions, everyone continues to think they are the only one who doesn't understand, then the misunderstandings are never cleared up." Dorothy Graham
- Planejamento dos testes e estimativa de esforço
  - Testes funcionais e não-funcionais
- Escrita e execução de Casos de Teste
  - Uso de técnicas e critérios de teste
- Relatório dos testes, com levantamento de bugs e melhorias
- Testes de front, back, API, BD, integração, mensageria
- Testes de aceitação, regressão, sanidade, entre outros
- Automação de testes

# Dicas

- Entenda o requisito
- Identifique dúvidas e as esclareça
- Se coloque no lugar do usuário
- Identifique comportamentos rotineiros
- Modularize (por funcionalidade, módulo, persona etc)
- Identifique o caminho feliz e os caminhos alternativos
- Estude e se mantenha atualizado

# Teste de Software Ferramentas & Artefatos

---



# Ferramentas

## Documentação & Gerenciamento de Testes



## APIs testadas



## Clouds & Serviços



## Automação de Testes



## CI/CD



## Testes de Performance



## Qualidade de Código



## Versionamento



## Banco de Dados



## Monitoramento



# Caso de Teste

- Principal artefato de teste
- Definem as entradas a serem informadas pelo testador (manualmente ou com apoio ferramental) e os resultados esperados a partir desta ação
- Descreve uma condição particular a ser testada e é composto por valores de entrada, restrições para a sua execução e um resultado ou comportamento esperado

# Caso de Teste - Exemplo

- Composto por: pré-condição; passo; entrada e possíveis dados; saída esperada
- Na execução temos também: resultado obtido e evidências

A	B	C	D	E	F	G
Id	Passo	Valor	Ação	Resultado Esperado	Resultados Falhos	Evidências
1	Entrar com a palavra vazia no campo de Busca	""	Clicar em Buscar	Exibir mensagem: "Necessário entrar com a Palavra Chave"	Não exibiu o a mensagem, mas exibiu um erro desconhecido	Ver arquivo "CasoTeste 1.JPG"
2	Entrar com número de caracteres acima do permitido (maior que 20)	aaaaaaaaaaaaaaaaaaaaa	Clicar em Buscar	Exibir mensagem: "O número de caracteres ultrapassou o limite máximo de 20"		
3	Entrar com uma palavra <b>válida</b> no campo Busca, juntamente com uma data válida	célula	Clicar em Buscar	Listar até 2 vídeos com o título, Autor, Veículo, Categoria,	Listou mais de 2 vídeos, não encontrou o resultado de busca e exibiu um erro desconhecido	Ver arquivo "CasoTeste 3.JPG"
4	Entrar com uma palavra <b>não</b> válida no Campo Busca, juntamente com uma data <b>válida</b>	carro	Clicar em Buscar	Exibir mensagem: "Busca não encontrada"		
5	Entrar com uma palavra <b>não</b> válida no Campo Busca, juntamente com uma data <b>não</b> válida	carro e 01/24/2012	Clicar em Buscar	Exibir mensagem: "Busca não encontrada"		
6	Entrar com a data inicial igual a data final	01/02/2012 e 01/02/2012	Clicar em Buscar	Listar até 2 vídeos com o título, Autor, Veículo, Categoria,		
7	Entrar com a data Inicial maior que a data final	01/02/2012 e 01/02/2011	Clicar em Buscar	Apresentar Erro: "A data inicial deve ser menor que a data final"	Apresentou erro indefinido	Ver arquivo "CasoTeste 7.JPG"

# Ciclo de Teste

- Um Ciclo é um conjunto de Casos de Teste
  - Lembrando que cada Caso de Teste possui seus próprios passos

Ciclo 1
CT1 - Cadastro
CT2 - Login e logout
CT3 - Compra com produtos variados
CT4 - Busca de produtos

# Ciclo de Teste

- As divisões mais comuns de Ciclos de Teste são:
  - Sanidade - alto nível
  - Teste de nova funcionalidade e de bug (aceitação)
  - Regressão
  - Fumaça - mais curto que o Sanidade, cenários rápidos em áreas principais
- Temos também Testes Exploratórios

# Ciclo de Teste

- Nova funcionalidade: Cesta de produtos

Sanity
CT1 - Cadastro
CT2 - Login e logout
CT3 - Compra com produtos variados
CT4 - Busca de produtos

Teste de nova funcionalidade
CT5 - Adicionar um ou mais produtos
CT6 - Remover um ou mais produtos
CT7 - Valor unitário e total da cesta visível e correspondente
CT8 - Verificar tempo limite de validade da cesta

Teste de regressão
CT9 - Cadastro e login inválidos
CT10 - Cadastro e login válidos
CT11 - Compra sem produtos
CT12 - Pagamento com cartão de crédito
CT13 - Pagamento com boleto bancário
CT14 - Busca por categoria
etc

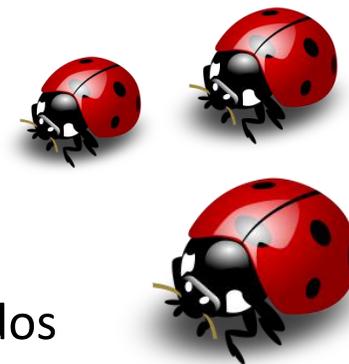
# Retomando... Cultura de Qualidade

---



# Gostei, quero ser QA

- Características comumente solicitadas:
  - Habilidade de comunicação clara e objetiva (fala-escrita);
  - Raciocínio lógico e boa interpretação de texto;
  - Conhecimentos sobre Engenharia de Software (modelos de processo, gerência de configuração, qualidade de software, UML e etc.);
  - Conhecimentos sobre métodos e técnicas de Teste;
  - Conhecimentos sobre SOs, SGBDs, redes e gestão de versões;
  - Habilidade de definir, coletar, analisar, gerir e reportar métricas de teste;
  - Automação – não é bala de prata, mas você tem que saber;
  - Testes de desempenho e requisitos não funcionais não podem ser esquecidos
  - CTFL (*Certified Tester Foundation Level*) desejável.



# Independente do seu papel



- Enquanto PO/ líder / PM ...
  - Defina o “conceito de pronto”;
  - Especifique os requisitos de maneira testável;
  - Garanta que o processo de desenvolvimento de código seja paralelo ao processo de teste
  - Provisione esforço de teste, de homologação
  - Garanta um ambiente viável para testes – usuários, instâncias, ferramentas...
- Enquanto desenvolvedor / cientista de dados, criador de código...
  - Atente-se ao requisito;
  - Produza o menor número de defeitos possíveis;
  - Entenda a interferência do seu código no “todo”;
  - Comente seu código / reporte o que foi feito com clareza;
  - Qualidade de código é fundamental – há ferramentas para te ajudar
  - Teste unitário é seu amigo e os QAs também
    - Todos querem o sucesso do software, da empresa, do cliente, e próprio!

# Independente do seu papel



- Enquanto DevSecOps / SRE / Operações
  - Esteja alinhado com os testes que podem ser feitos, dê ideias e cenários
  - Garanta documentação da infra – isso impacta nos testes e no todo
  - Acompanhe testes de desempenho com monitoramento
  - Pipelines de CI/CD fazem parte do SDLC – esteja alinhado e proponha melhorias
  - Teste a infra (infra as a code)
- Enquanto UI/UX
  - O time de desenvolvimento é seu amigo – estejam alinhados, analisem a viabilidade técnica, aceite sugestões
  - Dispositivos, versões diferentes, padrões... Alinhamento do que é viável para o projeto é fundamental para manter as expectativas



**monitora**  
a *marlabs* company



[www.monitoretec.com.br](http://www.monitoretec.com.br)