

Automação de testes com Selenium WebDriver Introdução

Antônio Moraes Trindade

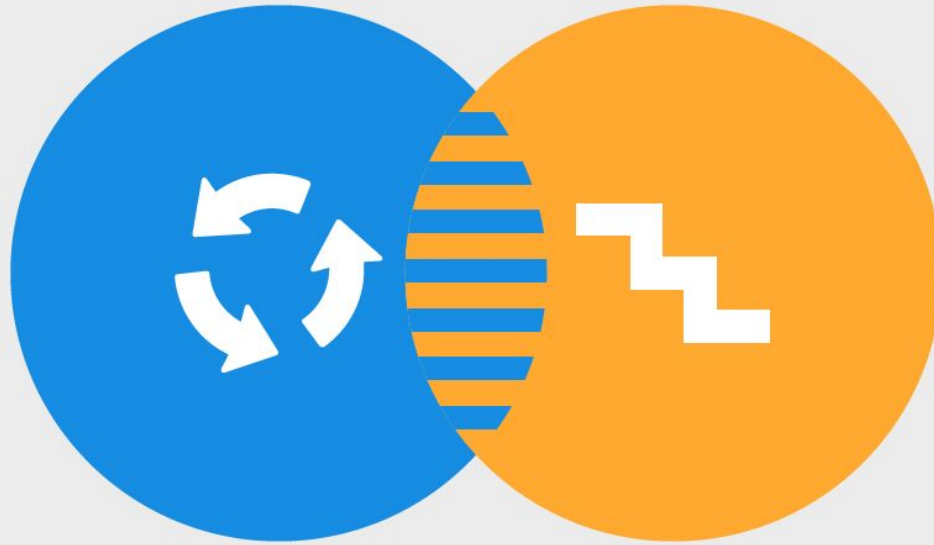
<https://about.me/amtrindade>

O que é automação de testes?

Automação de teste é o uso de software para controlar a execução do teste de software, a comparação dos resultados esperados com os resultados reais, a configuração das pré-condições de teste e outras funções de controle e relatório de teste. **(Wikipedia)**

AGILE

WATERFALL



Metodologias ágeis e o conceito de iterações e entregas frequentes

- Satisfazer o cliente através de entregas rápidas e contínuas de software em produção
- Efetuar entrega de software funcional com frequência e com intervalos mais curtos

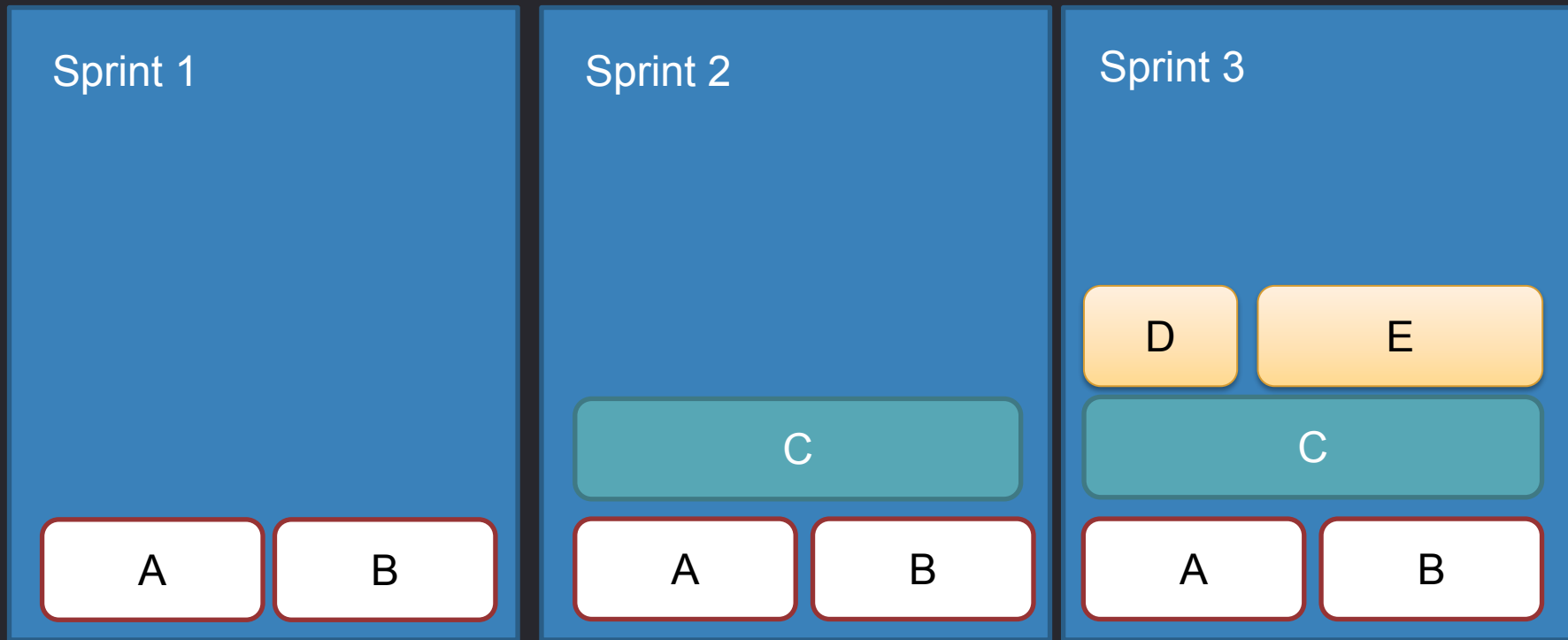


**E a automação, funciona com o modelo
Waterfall???**

E quanto aos testes realizados por sprint?



Sprints com as funcionalidades a serem entregues



Como a gerência da empresa imagina que os testes são feitos

Sprint 1

TESTES

Sprint 2

TESTES

TESTES

Sprint 3

TESTES

TESTES

TESTES

Como o time de desenvolvimento imagina que testa

Sprint 1

TESTES

Sprint 2

TESTES

A

B

Sprint 3

TESTES

C

A

B

Como o teste realmente é realizado

Sprint 1



Sprint 2



Sprint 3



Como realmente deveria ser

Sprint 1

TESTES

Sprint 2

TESTES

TESTES

Sprint 3

TESTES

TESTES

TESTES

Equipe de testes não pode se tornar o gargalo do processo

- Mesmo com uma equipe madura e com conhecimento do negócio, o número de funcionalidades e casos de testes podem chegar a um número tão grande que inviabiliza o software de ser testado manualmente a cada entrega.

E aí entra a automação de testes!



- Gerar feedback rápido para o time, com alta velocidade de execução, viabilizando a correção de bugs em funcionalidades impactadas de forma mais efetiva.
- Nenhuma interação humana, garantindo resultados consistentes.

E aí entra a automação de testes!



- Garantir que os testes de regressão continuem passando a cada nova funcionalidade
- Validar que o que funcionava na entrega anterior, continua funcionando para a próxima entrega

Mercado a procura por profissionais de automação de testes

- Busca contínua por qualidade aumentou consideravelmente o número de profissionais de QA
- Velocidade e entregas frequentes aquecem o mercado de automação de testes



Mitos que envolvem a automação de testes

Mitos da automação de testes!!!

- Uma ferramenta de automação de testes é a chave para todos os nossos problemas
- Testes automatizados vão reduzir a quantidade de defeitos
- Devo apenas melhorar a eficiência e eficácia da equipe de testes para reduzir o número de defeitos

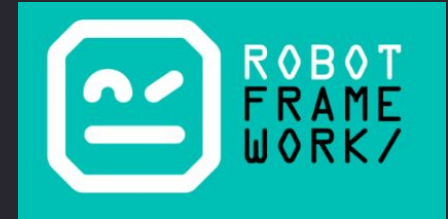
ROI em automação de testes

Expectativa de retorno no investimento em automação de testes seja no curto prazo

Complexidade Requisito	Teste manual		Teste automatizado	
	Criação	Execução	Criação	Execução
Baixa	30 (min)	2 (h)	9 (h)	09 (min)
Média	2 (h)	4 (h)	18 (h)	12 (min)
Alta	4 (h)	6 (h)	26 (h)	20 (min)

Adaptado de DONEGAN, Paula, et al. Automação de Testes Sistêmicos Funcionais. Instituto Atlântico.

E quais ferramentas estão disponíveis para automação de testes funcionais?



Diferenças entre Testes Unitários e Testes Funcionais

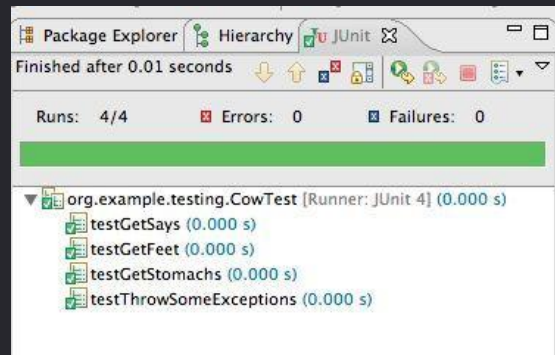
Testes unitários automatizados

Testa as classes e métodos da aplicação, validando entradas e saídas comparando resultados esperados.

Exemplos: JUnit, DUnit, NUnit, PHPUnit, etc...

JUnit:

- Framework open source criado para execução de testes unitários.
- Permite criar classes de com métodos de testes com asserções;
- Test Runner que executa testes e apresenta o resultado da execução.
- Integrado a IDE de desenvolvimento Java Eclipse.



Testes funcionais automatizados

Workflow of Selenium Webdriver



© <http://www.helloselenium.com>

Testa funcionalidades da aplicação, semelhante as ações que um usuário faria se estivesse interagindo com a aplicação



Mantra dos testes funcionais automatizados

- Identifica um elemento
- Realiza alguma interação com este elemento
- Valida um resultado esperado



7 boas práticas utilizadas na automação de testes para garantir qualidade e manutenibilidade



1. Cada ação deve ter no mínimo uma validação

- Assert (Valor esperado, Valor atual)
- O JUnit tem diversos comandos de Assert para serem utilizados, é função do automatizador saber qual o melhor momento e o que validar com cada uma deles.
- O assert é quem nos indica se o teste passou ou não.

2. Cada Test Case ser responsável por um único teste dentro de uma suíte

Os scripts elaborados (ou test cases) devem ser menores possíveis e com a nomenclatura o mais clara possível. Isto se deve para facilitar a identificação dos test cases que falham dentro da suíte e sua manutenção.



3. Test Cases devem ser independentes quanto a execução

Independência quanto a ordem e ao momento de execução, garantindo as suas próprias pré-condições para execução



4. Test Cases devem ser repetíveis

Testes automatizados podem ser executados quantas vezes forem necessários sem a intervenção humana.



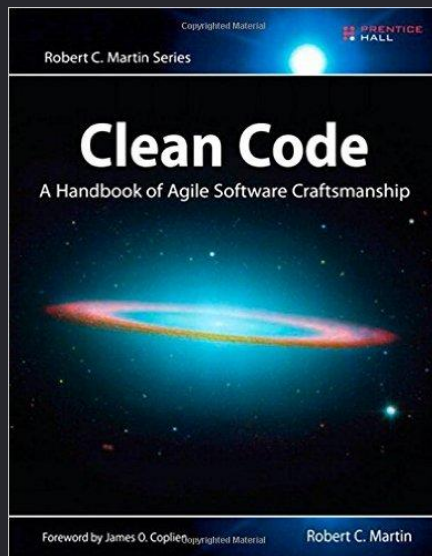
5. Suíte de automação deve ser robusta

Deve produzir os mesmos resultados a cada execução e não podem ser afetada por modificações externas ou no ambiente



6. Suíte de automação deve ser clara

Código deve ser limpo e de fácil entendimento e manutenção, seguindo orientações da linguagem de programação utilizada e princípios de Clean Code.



7. Suíte de automação deve ser eficiente

O desempenho da suíte de automação deve ser satisfatório e sem intervenção humana quanto a execução.

Os relatórios gerados devem ser claros, permitindo ao automatizador identificar facilmente o ponto onde o problema foi identificado providenciando o ajuste.

