

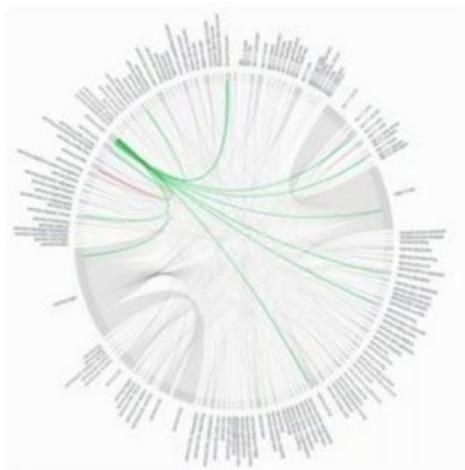
Testes em Java

Daniela Morais

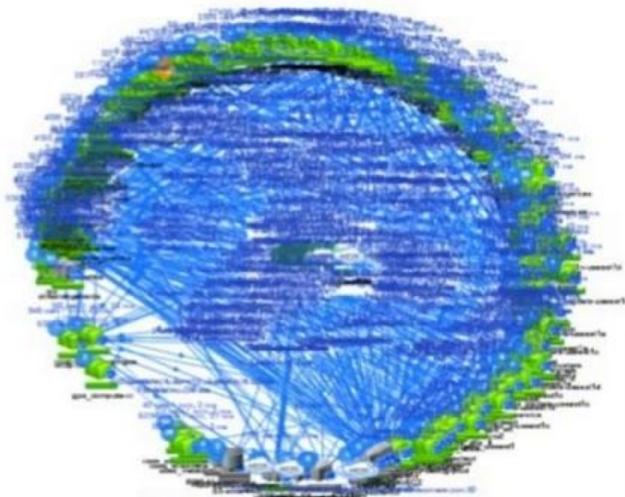
“é só uma correção pequena, não pode ser tão difícil”



450 microservices

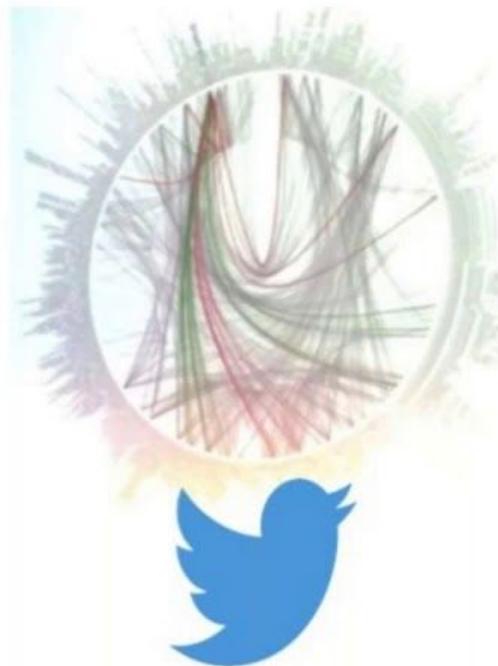


500+ microservices



NETFLIX

500+ microservices



Source:

Netflix: <http://www.slideshare.net/BruceWong3/the-case-for-chaos>

Twitter: <https://twitter.com/adrianco/status/441883572618948608>

Hail-o: <https://sudo.hailoapp.com/services/2015/03/09/journey-into-a-microservice-world-part-3/>

Lifecycle



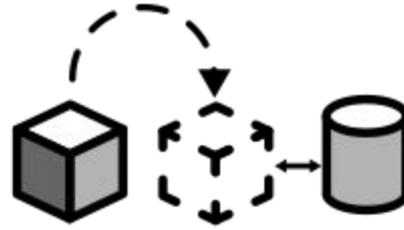
Build



Test



**Provide
Infrastructure**



Deploy



**Test
More**

carreira, ritos delimitados, grandes clientes e visibilidade por conta de grandes projetos. Não existe hierarquia, todos podem opinar sobre tudo.

Contras: Empresa extremamente exigente com qualidade de código, teste unitarios e funcionais, nenhum metodo é commitado sem seu devido teste de integração as vezes isso atrapalha a entrega e gera pressão e stress na equipe.

Conselhos para presidência: Analizar mais os gestores, muitos deles cobram muito além da realidade de queima da equipe.

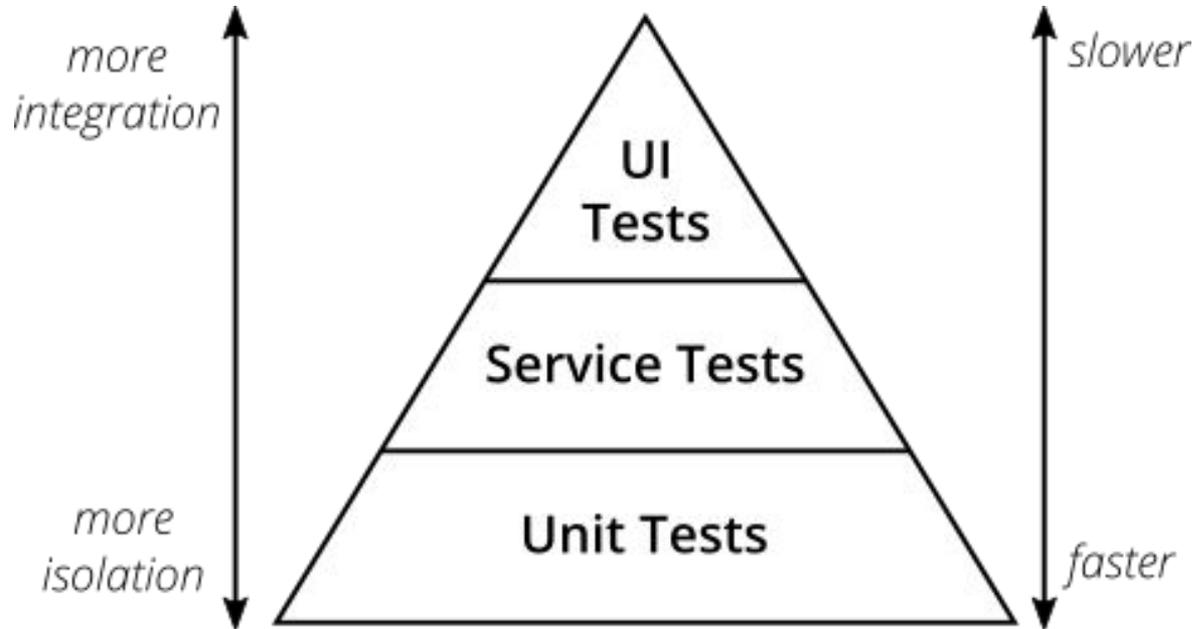
Recomenda a empresa:  Sim

ÚTIL?

You Can't Be Agile without Automated Unit Testing

Você é o responsável

Pirâmide de testes



UI Tests

End-to-end: Maior confiabilidade

Caro e frágil (mudanças de CSS)

Escolha de elementos/estilos de CSS

Ferramentas de front-end ou Selenium

Exemplo Selenium

Testes de integração

Comunicação entre as classes

Comportamento e conexão com serviços externos

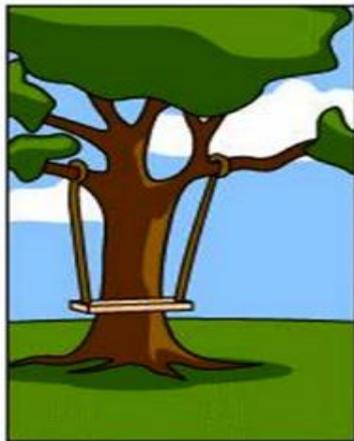
Testes de integração

Comunicação entre as classes

Comportamento e conexão com serviços externos



How the customer explained it



How the Project Leader understood it



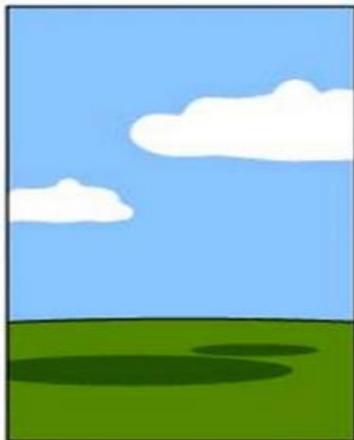
How the Analyst designed it



How the Programmer wrote it



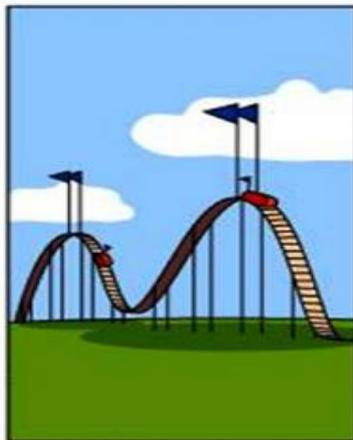
How the Business Consultant described it



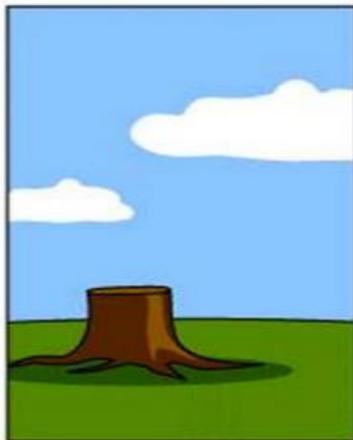
How the project was documented



What operations installed



How the customer was billed



How it was supported



What the customer really needed

BDD

Usado bastante com TDD

Foco nos requisitos de negócio

Garantia maior do entendimento da funcionalidade

BDD

Título claro que deixe explícito: quem deve executar a user story, o que deve fazer o por quê

O fornecedor ao obter o pedido deve ter como retorno todas as informações no JSON e JSON deve ter informações de endereço e de itens de pedido

BDD

Dado um produto custando £15, quando eu adicionar na cesta de compras, o preço total da cesta de compras deve ser £21

Given... when... then....

BDD

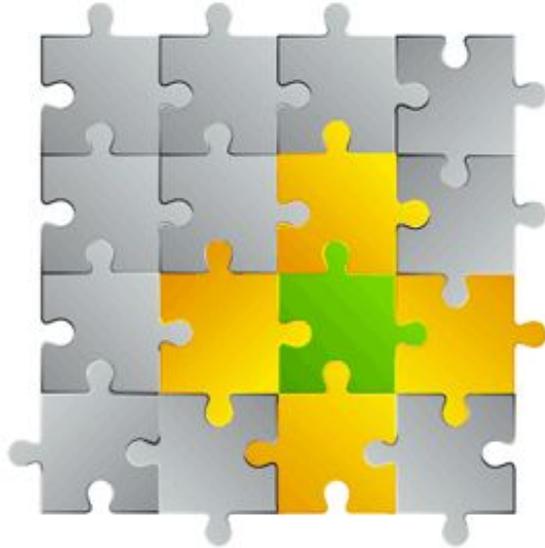
Dado um produto custando £15, quando eu adicionar na cesta de compras, o preço total da cesta de compras deve ser £21

Given... when... then....

Exemplo teste de integração do Postman com BDD

Unit test

REAL SYSTEM



Green = class in focus
Yellow = dependencies
Grey = other unrelated classes

CLASS IN UNIT TEST



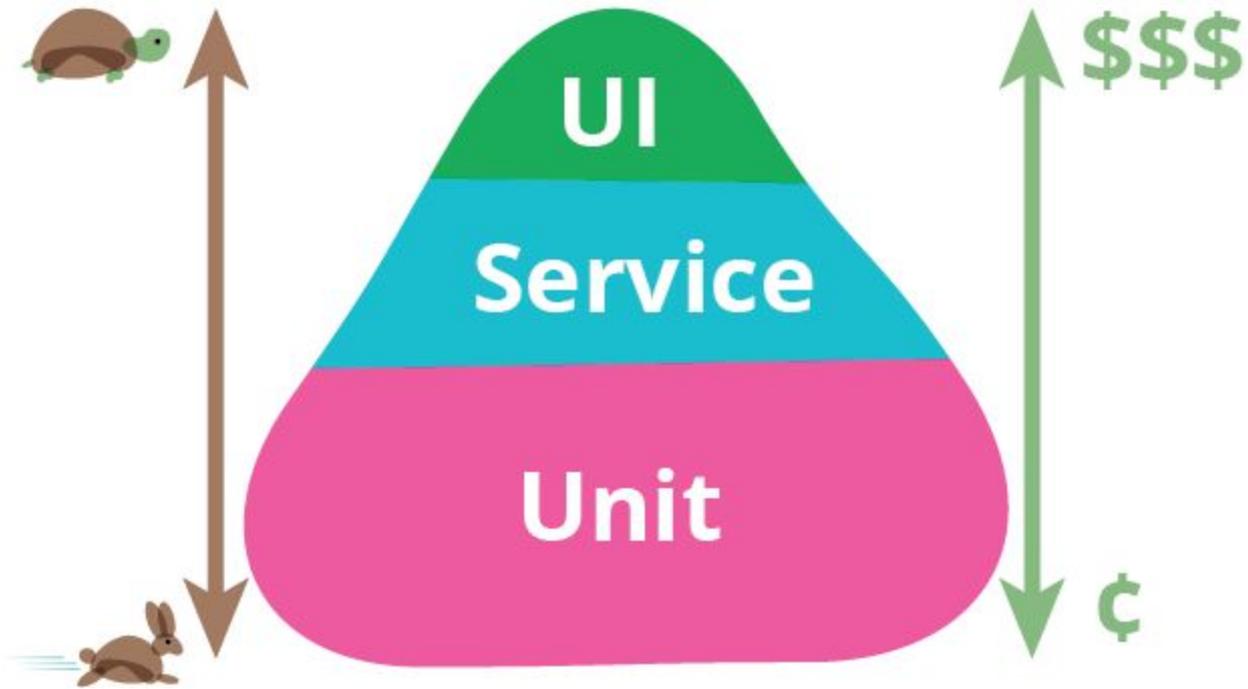
Green = class in focus
Yellow = mocks for the unit test

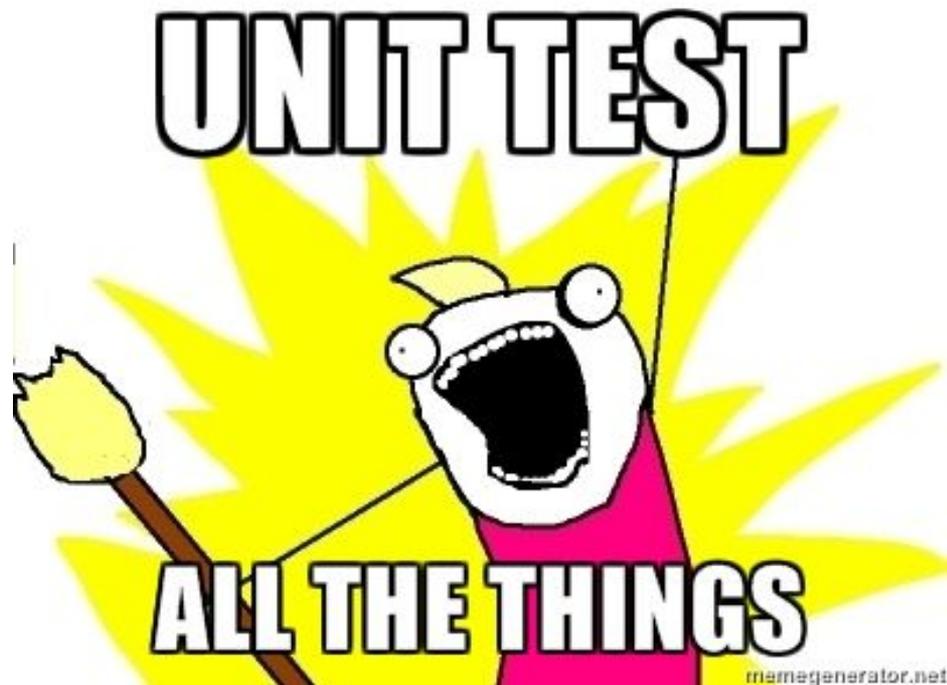
Unit test

Dado um produto custando £15, quando eu adicionar na cesta de compras, o preço total da cesta de compras deve ser £21

Given... when... then....

Exemplo teste unitários



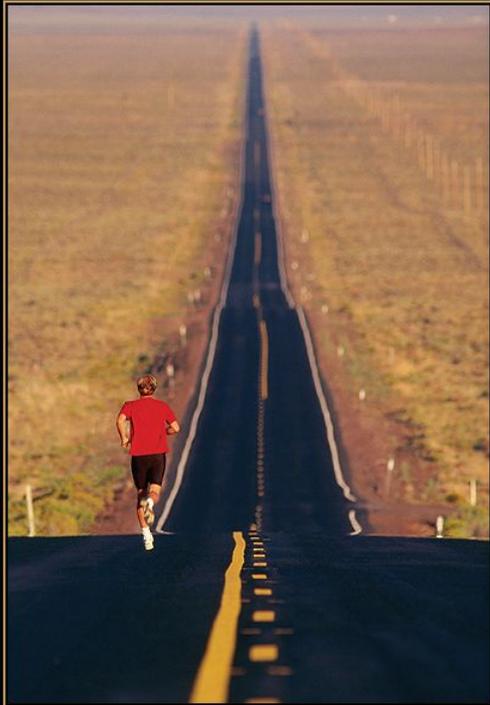


testes continuam sendo códigos

Documento para comprovar funcionalidade

Code smells

Dificuldade de leitura



QUALITY

THE RACE FOR QUALITY HAS NO FINISH LINE-
SO TECHNICALLY, IT'S MORE LIKE A DEATH MARCH.

Referências

Beating the Quality Death March with BDD

<http://jlhood.com/beating-the-quality-death-march-with-bdd/>

The Practical Test Pyramid

<https://martinfowler.com/articles/practical-test-pyramid.html#TheImportanceOftestAutomation>