

Coma seus vegetais

Eduardo Mendes



5 perguntas

Só pra gente começar a
interagir

**1. Quem desenvolve
software
profissionalmente?**

— — —

5 perguntas

Só pra gente começar a
interagir

**2. Quem trabalha com
metodologias ágeis?**

— — —

5 perguntas

Só pra gente começar a
interagir

**3. Quem trabalha com
testers na equipe?**

— — —

5 perguntas

Só pra gente começar a
interagir

**4. Quem entrega software
com qualidade?**

5 perguntas

Só pra gente começar a
interagir

**5. Quem desenvolve
software para usuários?**

— — —



Nome:

Eduardo Mendes

Instituição:

DieboldNixdorf

Email:

mendesxeduardo@gmail.com

GIT:

github.com/z4r4tu5tr4

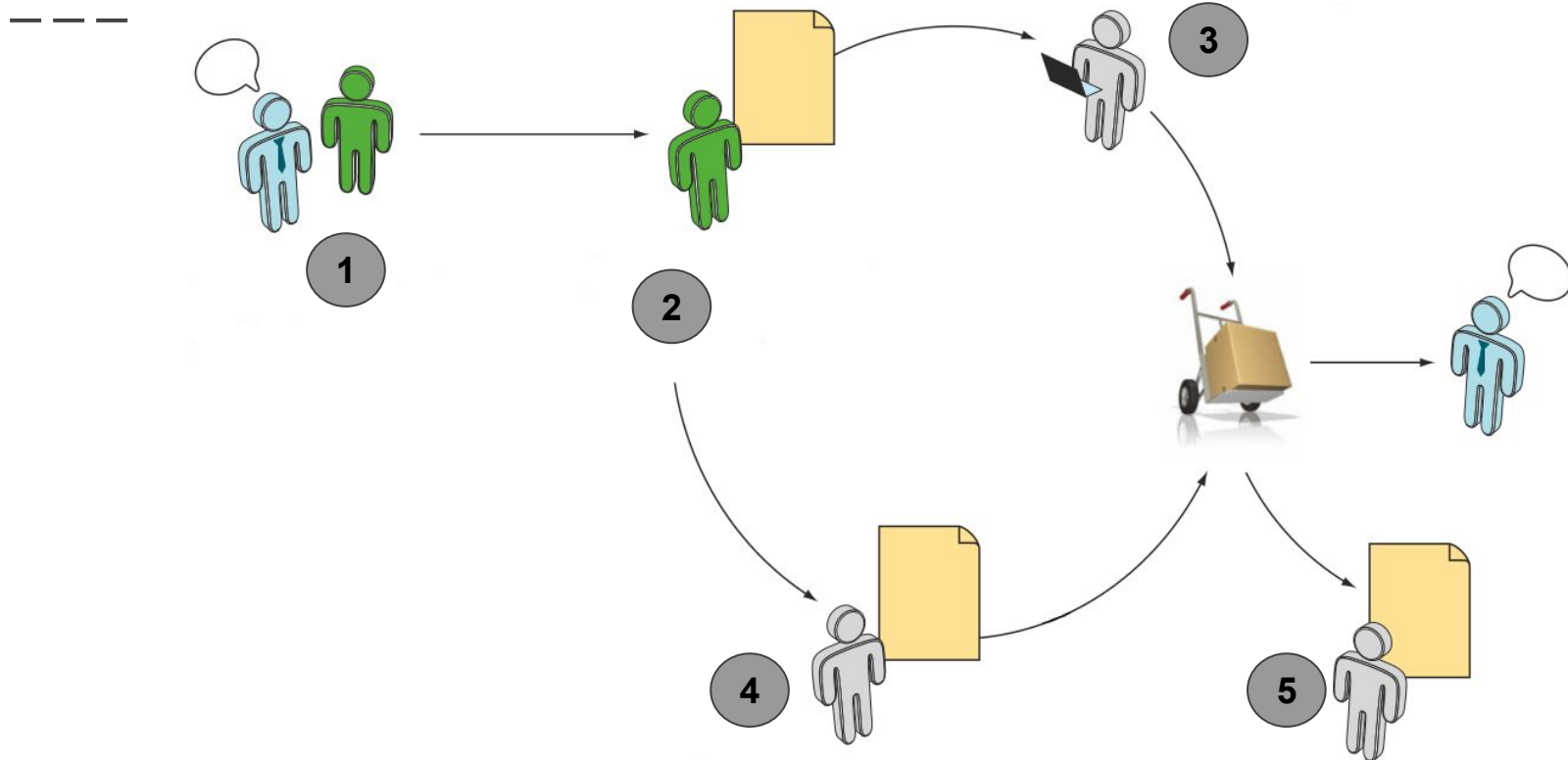
Sim, eu vim aqui falar de qualidade de software

Qualidade de software?

— — —

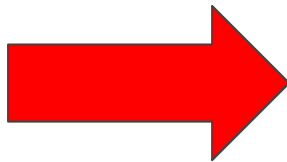


Processo de desenvolvimento (Tradicional)



Sim, eu não queria colocar esse slide

— — —



Testes automatizados / Integração contínua (P)

— — —

- Testes unitários
 - Testar todas as menores parcelas de código possíveis
- TDD
 - Escrever o teste para falhar
 - Fazer o teste passar
 - Refatorar o código
- CI
 - Garantir que toda modificação possível no código não faça nenhuma outra parte do mesmo falhar

Testes automatizados / Integração contínua (N)

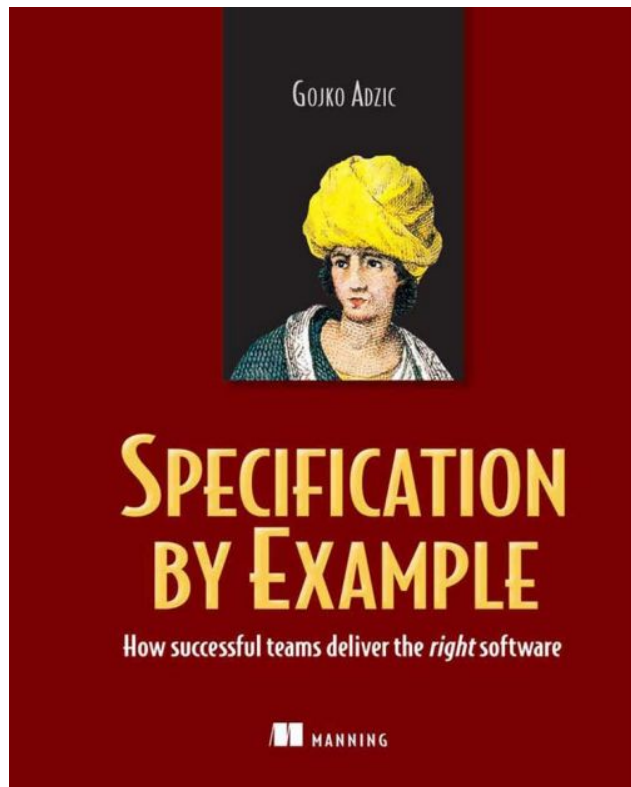
— — —

- Testes unitários
 - Testes unitários geralmente testam funções/métodos/classes
 - Quando você escreve os testes, eles testam as regras de negócio?
- TDD
 - Garante que o que foi feito realmente está funcionando, porém garante a integração de tudo que foi feito?
- CI
 - Quando você roda uma bateria completa de testes e garante que tudo está funcionando a cada commit você garante que a entrega tem mesmo de ser aquela?
 - A documentação é atualizada a cada commit?

Então, a idéia é tentar fazer testes como centro do desenvolvimento.

**Não testes como testes, mas como assertivas de
critério de aceite. Ou seja, documentação
executável**

Especificações por exemplo !



- Testes que agregam valor
- Documentação viva
- Noção de progresso
- Testes automatizados
- Funcionalidades funcionais

Mas como? (Testes unitários, como eu fazia)

```
class Testes_de_contas_bancarias(TestCase):  
    def teste_transferencia(self):  
        pass  
  
    def teste_deposito(self):  
        pass
```

Mas como? (Testes unitários, visando a regra de negócio) - ATDD

```
class Testes_de_transferencias_e_depositos(TestCase):  
    def teste_deve_transferir_fundos_para_contas_locais(self):  
        pass  
  
    def teste_deve_transferir_fundos_para_contas_de_outros_bancos(self):  
        pass  
  
    def teste_deve_depositar_fundos_em_contas_locais(self):  
        pass  
  
    def teste_deve_depositar_fundos_em_contas_de_outros_bancos(self):  
        pass
```

Mas como? (Requisitos no formato de estórias) - BDD

— — —

Cenário: Transferir dinheiro para uma conta no exterior

Dado que um cliente possui uma conta corrente

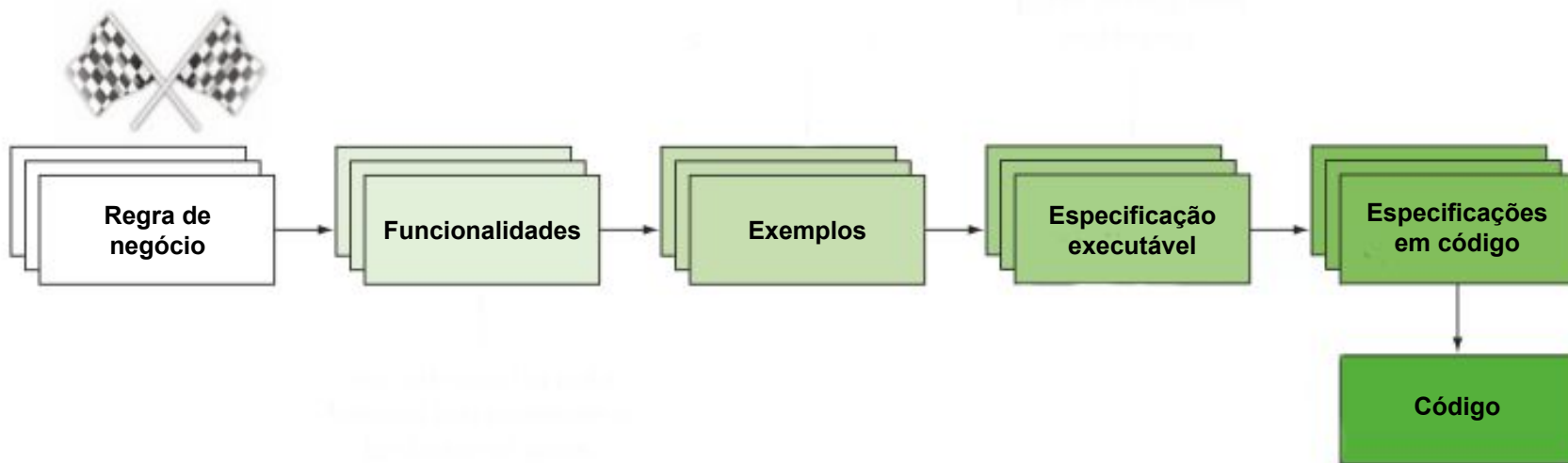
Quando o cliente transfere fundos desta conta para uma conta no exterior

Então os fundos devem ser depositados na conta no exterior

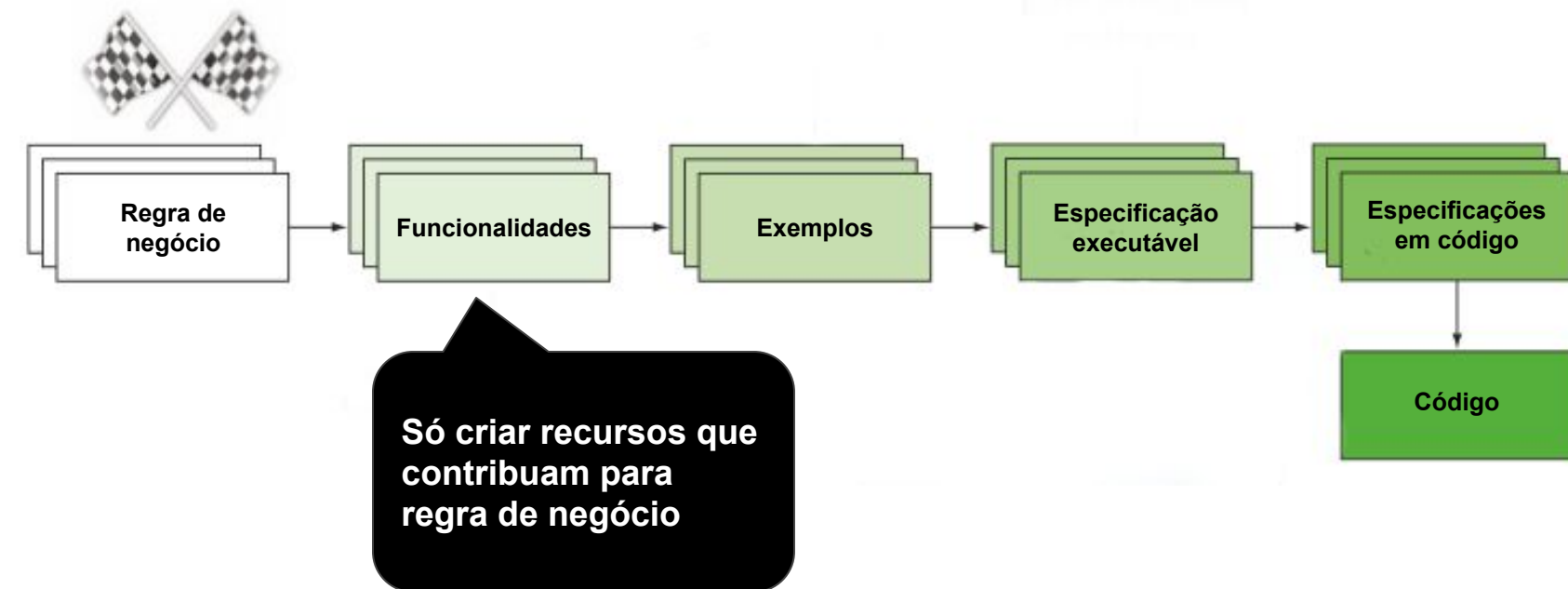
E a taxa de transação deve ser deduzida da conta corrente

Mas como? Uma visão geral do modelo

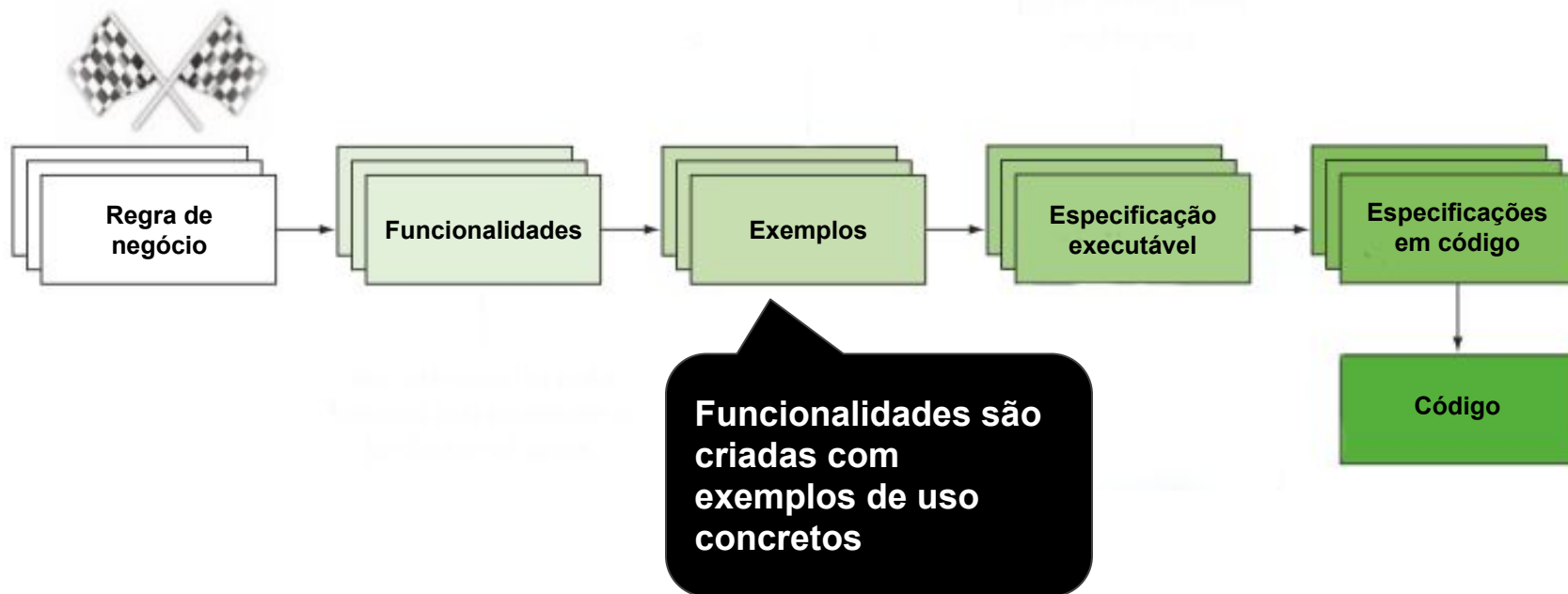
— — —



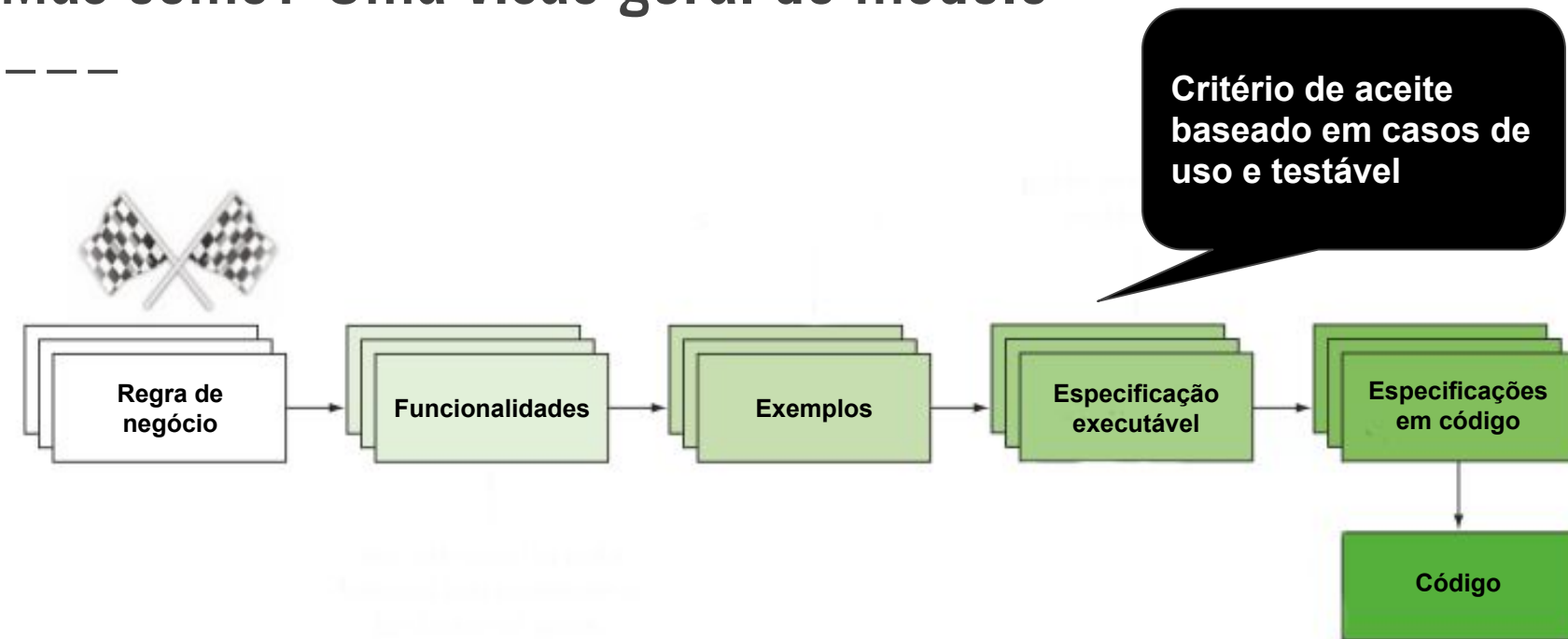
Mas como? Uma visão geral do modelo



Mas como? Uma visão geral do modelo

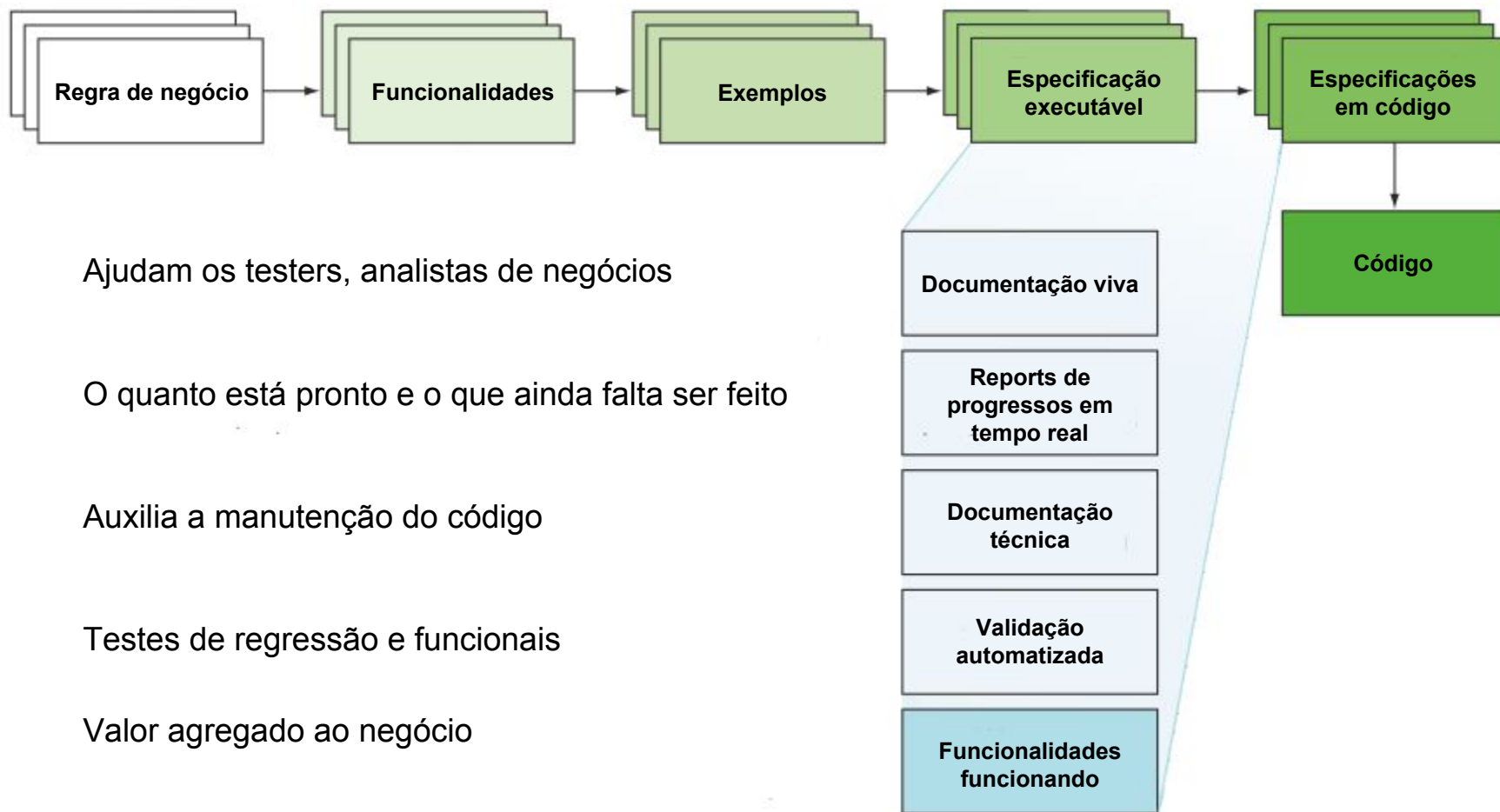


Mas como? Uma visão geral do modelo

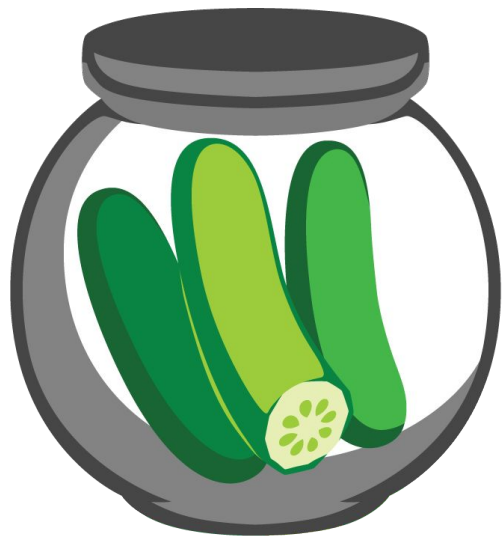


Mas como? (especificação em código)

```
---  
@given('que um cliente possui uma conta corrente')  
def check_if_user_has_account(context):  
    pass  
  
@then('os fundos devem ser depositados na conta no exterior')  
def transfer_money_to_other_country(context):  
    pass
```

Documentação viva (C#)



Pickles

Living Documentation

Noção de progresso (cucumber report - Java)

Project	Number	Date
Damian - project with Cucumber	10	11 gru 2016, 00:16

Branch	release/1.0
Browser	Firefox
Platform	Windows

Features Statistics

The following graphs show passing and failing statistics for features

Steps



Feature	Steps						Scenarios			Features	
	Passed	Failed	Skipped	Pending	Undefined	Total	Passed	Failed	Total	Duration	Status
1st feature	10	0	0	0	0	10	1	0	1	1m 39s 263ms	Passed
Second feature	5	1	2	1	2	11	1	1	2	092ms	Failed
2	15	1	2	1	2	21	2	1	3	1m 39s 355ms	
	71.43%	4.76%	9.52%	4.76%	9.52%		66.67%	33.33%			50.00%

Noção de progresso (cucumber report - Java)

Feature Report

	Steps						Scenarios			Features	
Feature	Passed	Failed	Skipped	Pending	Undefined	Total	Passed	Failed	Total	Duration	Status
Second feature	5	1	2	1	2	11	1	1	2	092ms	Failed

Feature Second feature

As an Account Holder I want to withdraw cash from an ATM,
so that I can get money when the bank is closed

@checkout

Scenario Outline Account may not have sufficient funds ▼

Account holder withdraws more cash

Hooks ▼

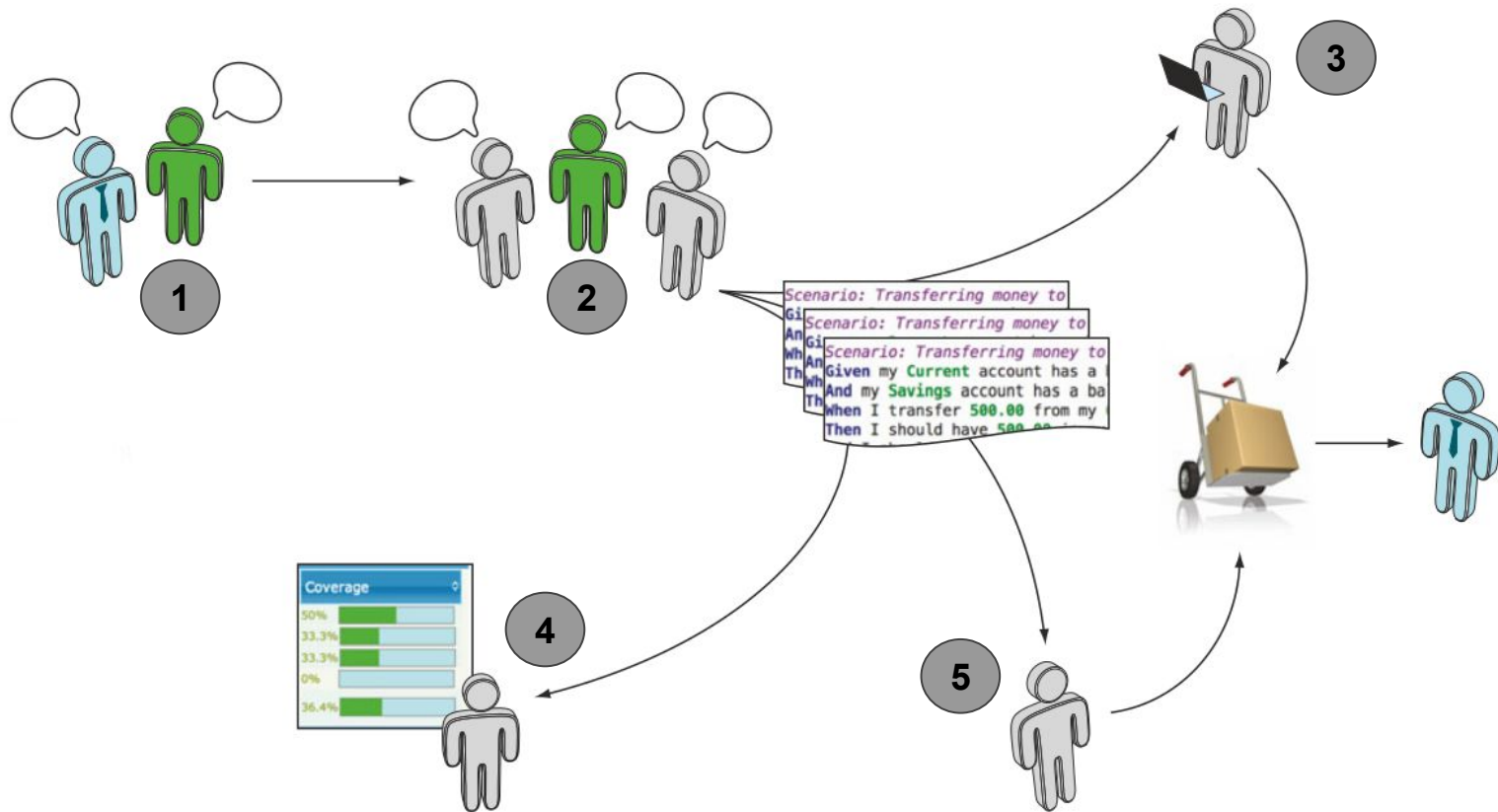
Before MachineFactory.findCachMachine()	010ms
Before MachineFactory.wait()	001ms

Steps ▼

Given the account balance is 100	000ms
And the card is valid	000ms
And the machine contains 100	000ms
When the Account Holder requests 20	000ms
Then the ATM should dispense 20	000ms
And the account balance should be 90	001ms

Error message

Processo de desenvolvimento (com BDD)



Nome da palestra

— — —

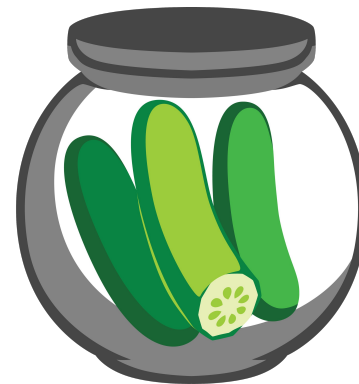


Cucumber

Lettuce



Radish



pickles

Gherkin

“Queremos construir softwares bons, mas também precisamos construir softwares que valham a pena construir”

**- John Ferguson
Smart**



XOXO

Dúvidas?

youtube.com/c/eduardomendes
mendesxeduardo@gmail.com