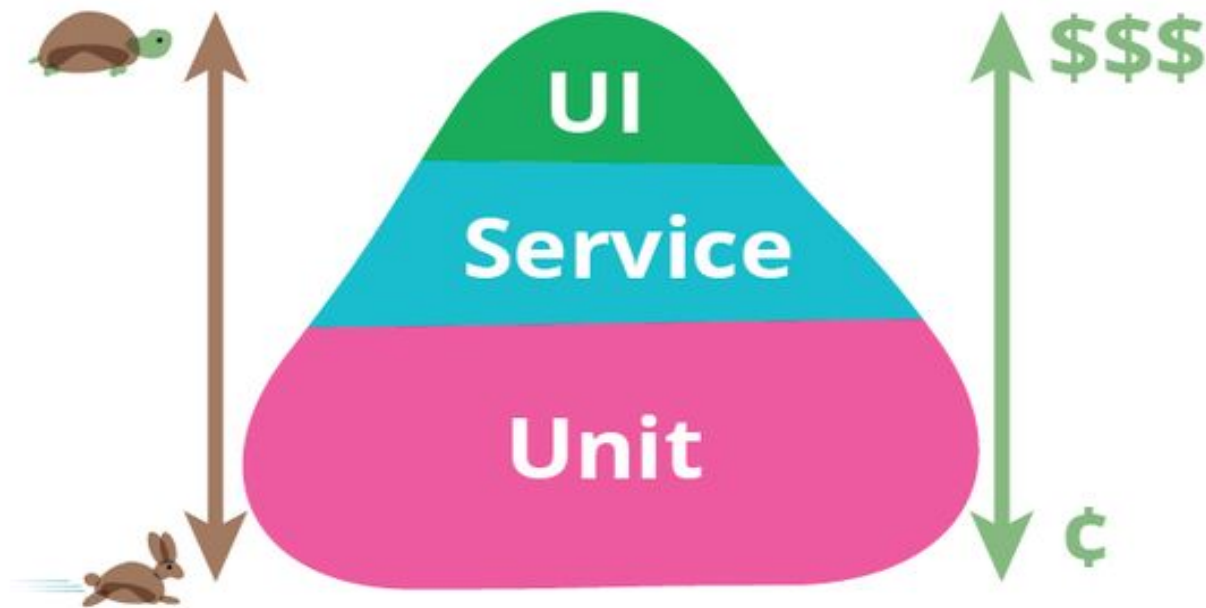


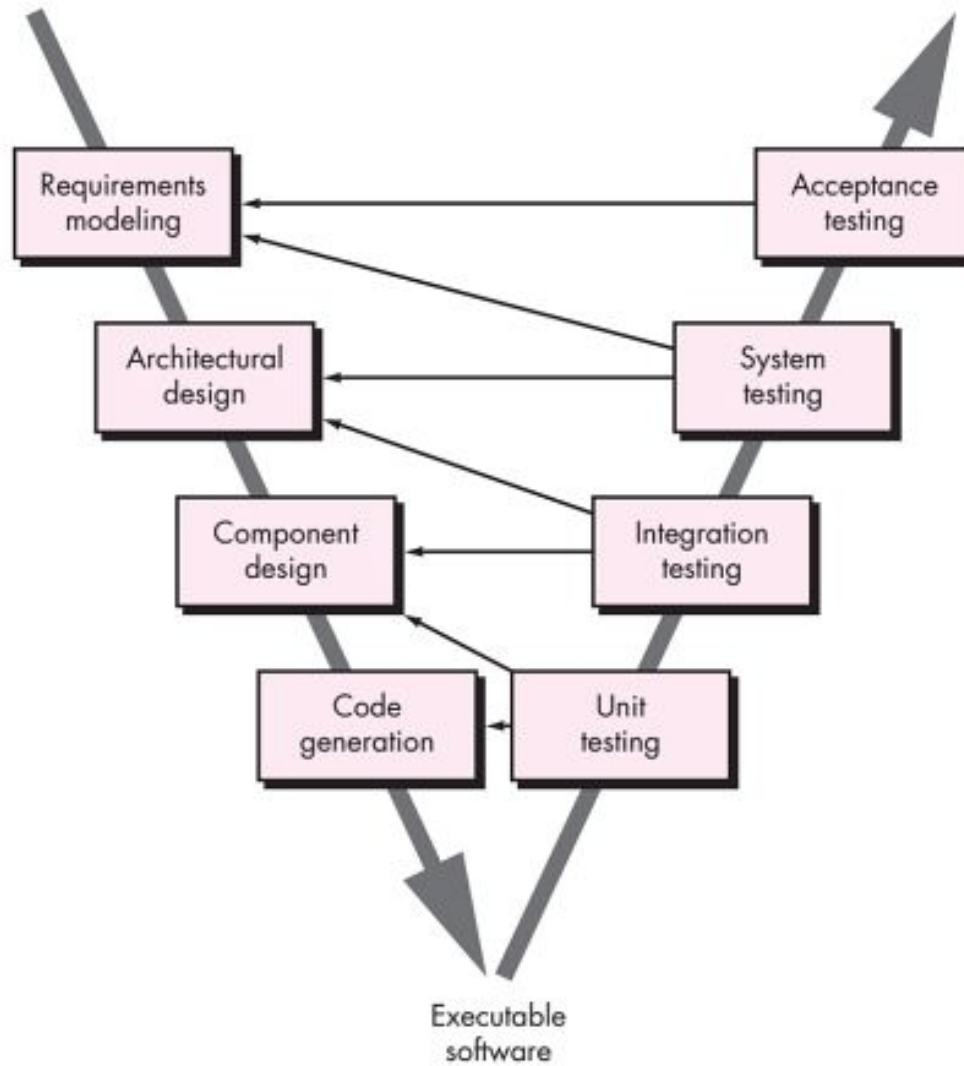
Automated Software Testing

João Henrique Victorino da Silva

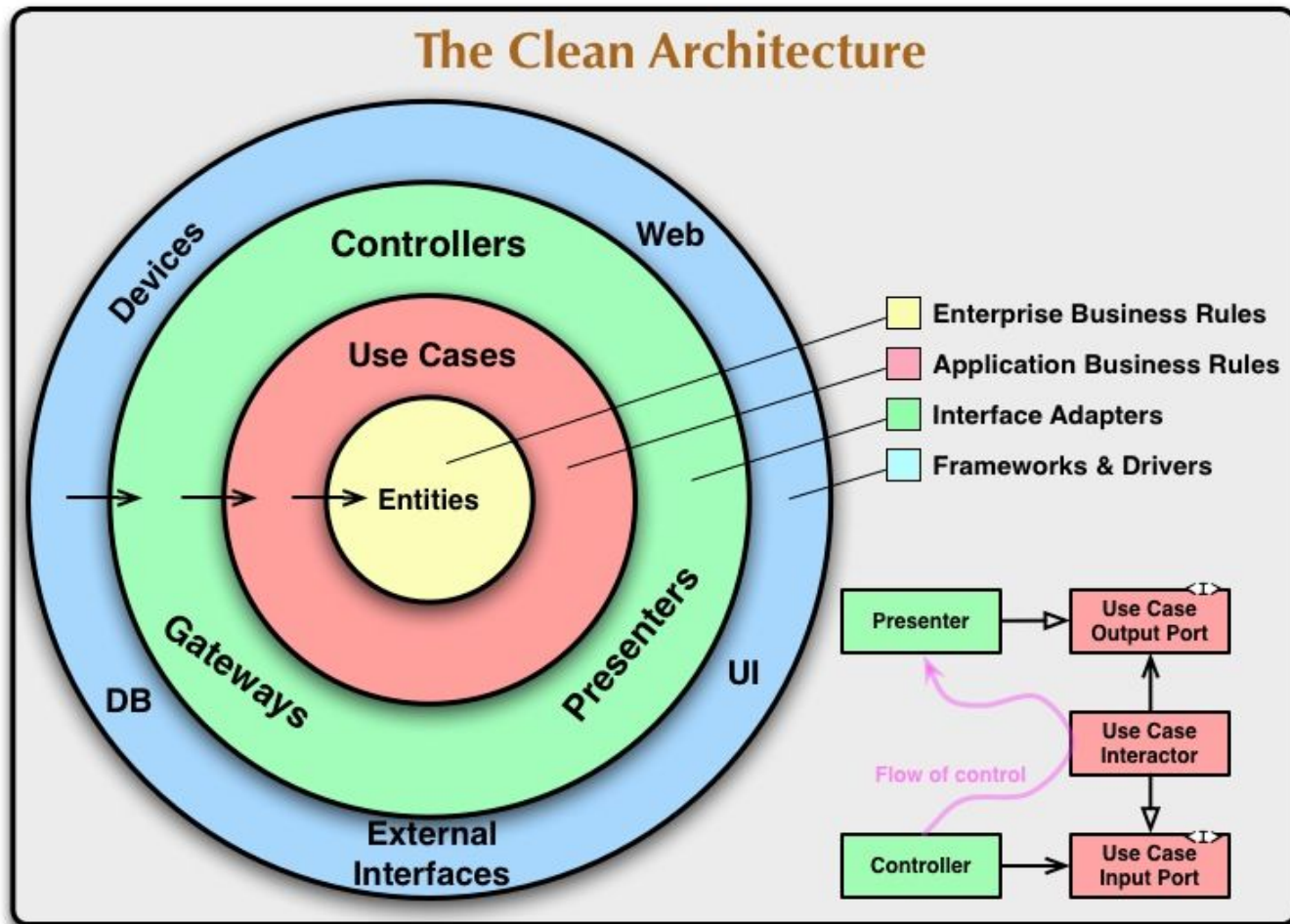
Pirâmide de testes



V-Model



Clean Architecture



Granularidade dos testes

- **Testes de unidade**, classes e métodos isolados
- **Testes de integração**, um componente ou mais
- **Teste de sistema**, todo o sistema, relação entre componentes e infraestrutura (RNF)
- **Teste de aceitação**, todo o sistema, relação entre componentes e infraestrutura (RF)



Framework

NEXT.js

<https://nextjs.org/>

Criando nosso projeto Next.js

<https://gist.github.com/joaovictorino/90114d1e857741c88d9b79e1bfad489e>

(20 minutos)



ORM



<https://www.prisma.io/>

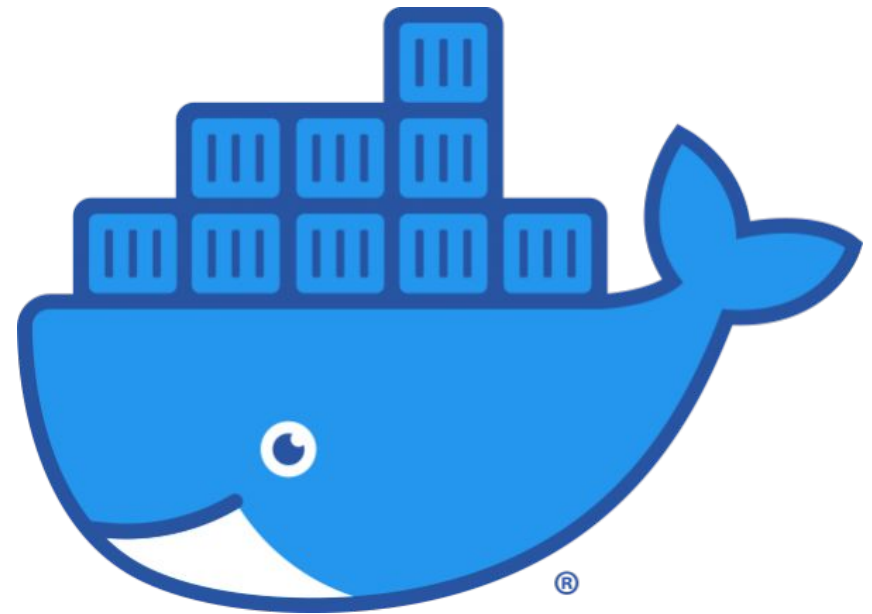
Banco de dados



<https://www.mysql.com/>

Docker

<https://docs.docker.com/>



Adicionando o Prisma

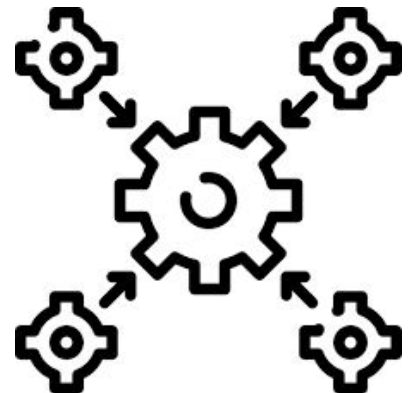
<https://gist.github.com/joaovictorino/33806a01fbd971ac90789b31daa7d693>

(20 minutos)

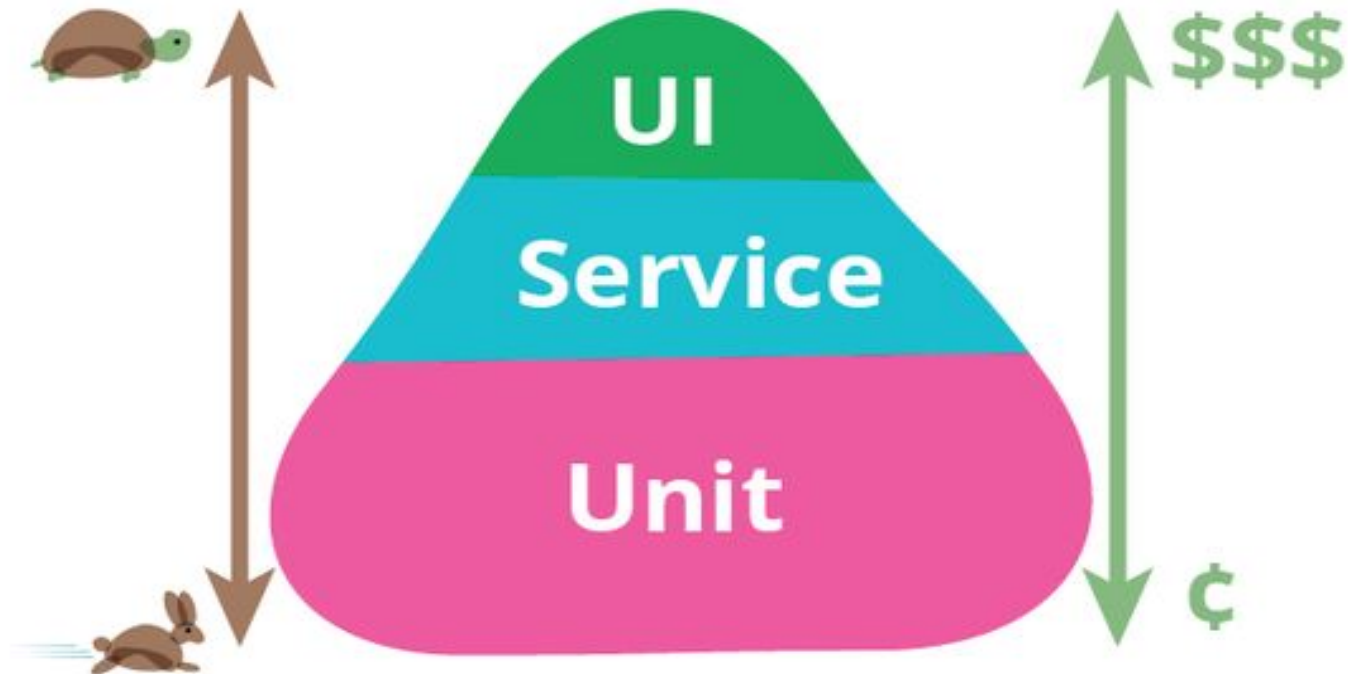


Testes de integração

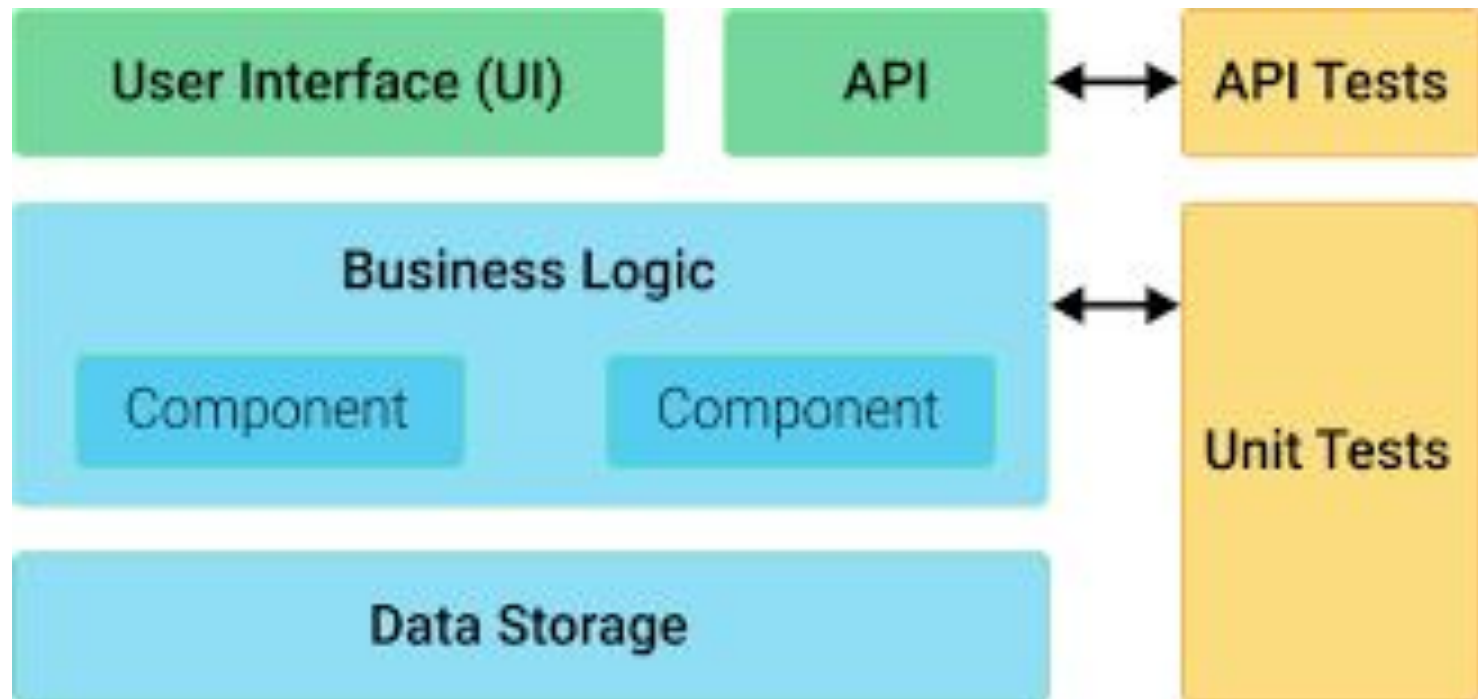
- Testar a relação/colaboração entre classes e componentes
- Se necessário ainda exclua a infraestrutura
 - Ainda será necessário usar mocks para outras classes ou componentes
- Pode testar recursos externos para garantir qualidade
- Explorar exceções que não puderam ser tratadas durante o teste de unidade



Pirâmide de testes



API Testing



Modelagem de API

<https://adventure-works.com/orders> //Correto

<https://adventure-works.com/create-order> //Evitar

POST <https://adventure-works.com/orders>

Content-Type: application/json

```
{"Name":"Gizmo","Category":"Widgets","Price":1.99}
```

201 Created

Content-Type: application/json

```
{"id":1}
```

<https://www.openapis.org/>

Modelagem de API

Recurso	POST	GET	PUT	DELETE
/clientes	Criar cliente	Listar clientes	Atualizar clientes em lote	Remover todos os cliente
/clientes/1	Erro	Detalhar cliente 1	Atualizar os dados do cliente 1	Remover o cliente 1
/clientes/1/pedidos	Criar pedidos no cliente 1	Listar pedidos do cliente 1	Atualizar pedidos do cliente em lote	Remover todos os pedidos do cliente 1
/clientes/1/aprovar *	Erro	Erro	Aprovar o cliente 1	Erro

<https://www.openapis.org/>

Modelagem de API

Retorno	POST	GET	PUT	DELETE
Sucesso	201 (created) 200 (ok) 204 (no content)	200 (ok)	201 (created) 200 (ok) 204 (no content)	202 (accepted) 200 (ok) 204 (no content)
Falha	400 (bad request)	404 (not found)	409 (conflict)	404 (not found) 409 (conflict) 403 (forbidden)

<https://www.openapis.org/>

Criando API de consulta de contas

<https://gist.github.com/joaovictorino/d030cfc1b768ca2008c4c79fae21a5ea>

(20 minutos)



Testes da API de consulta de contas

<https://gist.github.com/joaovictorino/f16a0f28168a328b35bc9c0894f61cca>

(10 minutos)



Mockando o prisma

<https://gist.github.com/joaovictorino/d4537368c8c4e54a26fad39efb0b6164>

(10 minutos)



Criando API de criação de contas

<https://gist.github.com/joaovictorino/c022d6ffca926eed9c98a00672390676>

(5 minutos)



Testes da API de criação de contas

<https://gist.github.com/joaovictorino/4ebc43d10640fdf8fadc48970e85742>

(5 minutos)



Criando e testando a API de transferência

<https://gist.github.com/joaovictorino/3d60abbdd114d8cf6627e0e851b74f25>

(10 minutos)



Cucumber



<https://cucumber.io/>

Incluindo BDD nos testes de API

<https://gist.github.com/joaovictorino/2a9f693fe15c52f47a7702d3dd56a850>

(10 minutos)

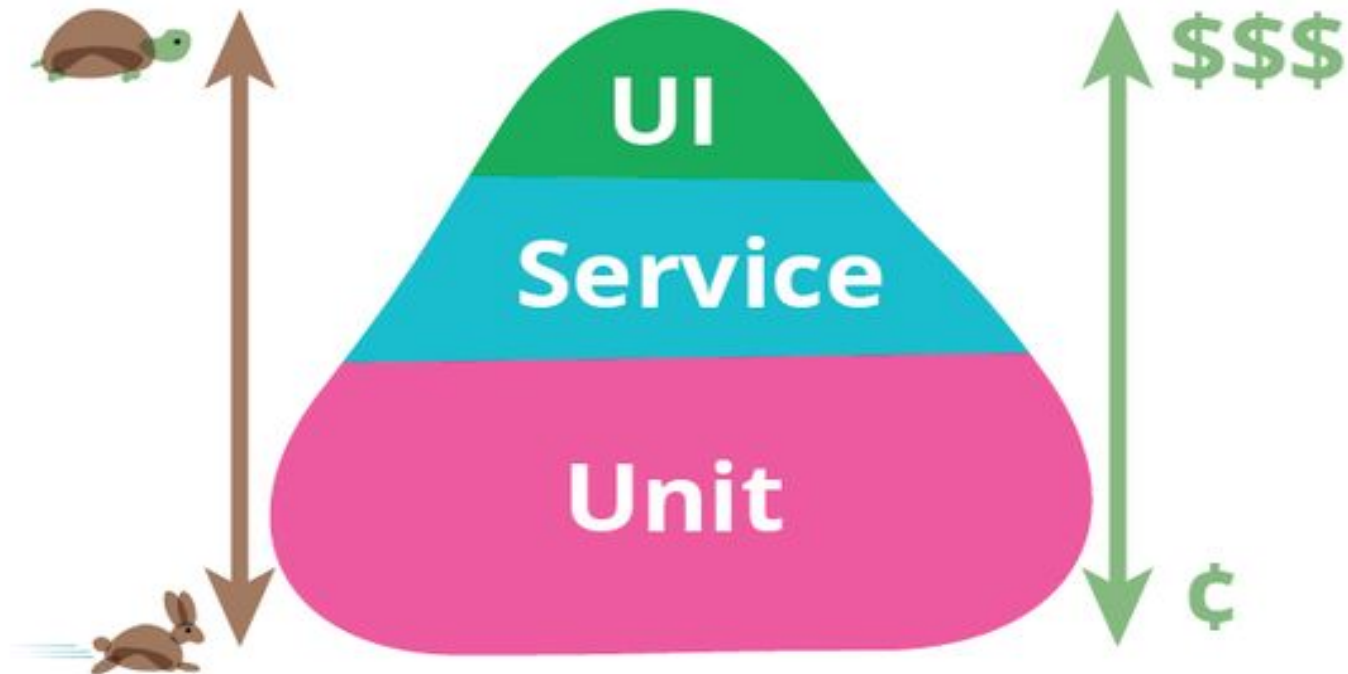


Testes de aceitação

- Testes dos requisitos funcionais do sistema, é o momento de incluir a infraestrutura e outros elementos que façam parte da solução.
- Este é um teste automatizado também, mas feito na visão do usuário, mostrando exatamente como o sistema será usado, também é conhecido como teste ponta a ponta.



Pirâmide de testes



Criando as telas iniciais

<https://gist.github.com/joaovictorino/f53f451e6d4a208a7a9790149791d759>

(20 minutos)



Criando as outras telas

<https://gist.github.com/joaovictorino/3556f45bfd21ddbef0dfe494620dc780>

(10 minutos)



Playwright



<https://playwright.dev/>

Instalando e configurando o Playwright

<https://gist.github.com/joaovictorino/0ade99dc775e9bf85997630ffc97de97>

(15 minutos)



Criando o teste do nosso projeto

<https://gist.github.com/joaovictorino/ecac716b679f999f2337b01ccb4bd58f>

(10 minutos)



Criando através da interface

<https://gist.github.com/joaovictorino/f356b173a965461b7b171c80a350003e>

(10 minutos)



Playwright e Cucumber



<https://cucumber.io/>



<https://playwright.dev/>

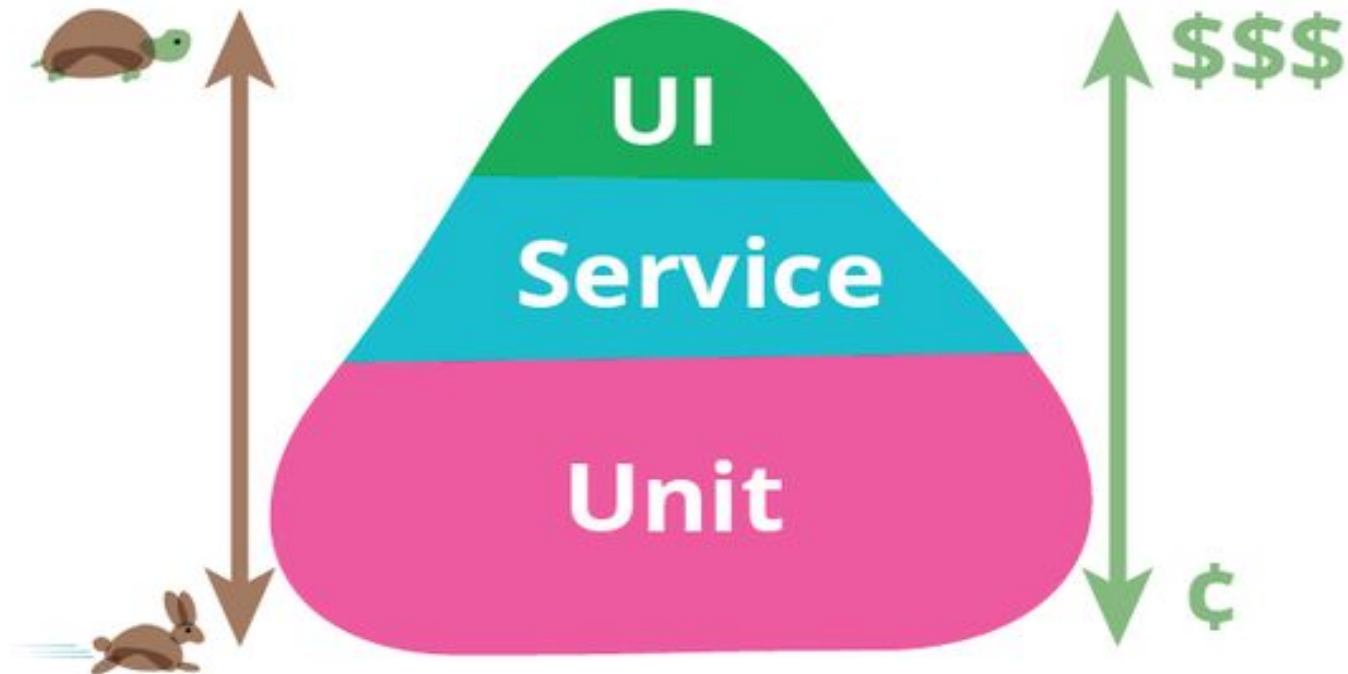
Unindo BDD e Playwright

<https://gist.github.com/joaovictorino/d7e5fb04e4b69a2df607ee43b2ed60e3>

(10 minutos)



Pirâmide de testes



BOA NOITE!

Referência no ensino de Tecnologia em **Graduação, Pós, EAD, MBA, cursos livres e treinamentos empresariais**. Lidera iniciativas em Inteligência Artificial, Ciência de Dados, Robótica, Engenharia da Computação, Big Data, UX e Transformação Digital. Criou a incubadora Impacta Open Startup, exclusiva para seus estudantes. É campeã na formação de times para Hackathons, eventos de inovação patrocinados por gigantes: NASA, IBM, Deloitte, Shell, Santander, Itaú, Globo e Fiesp. É uma das marcas mais admiradas pela Comunidade Tech da América Latina. Atua no mercado desde 1988 e formou mais de 1 milhão de pessoas e certificou mais de 25 mil empresas de diversas áreas da Economia. Ver mais em: impacta.edu.br e impacta.com.br

