



# О себе

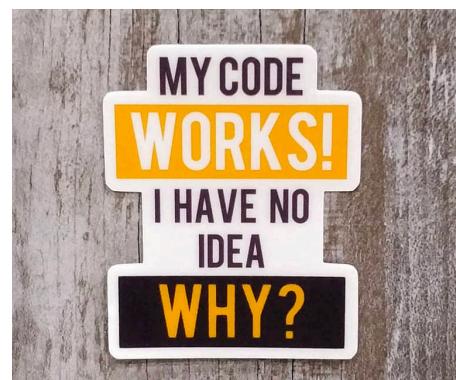


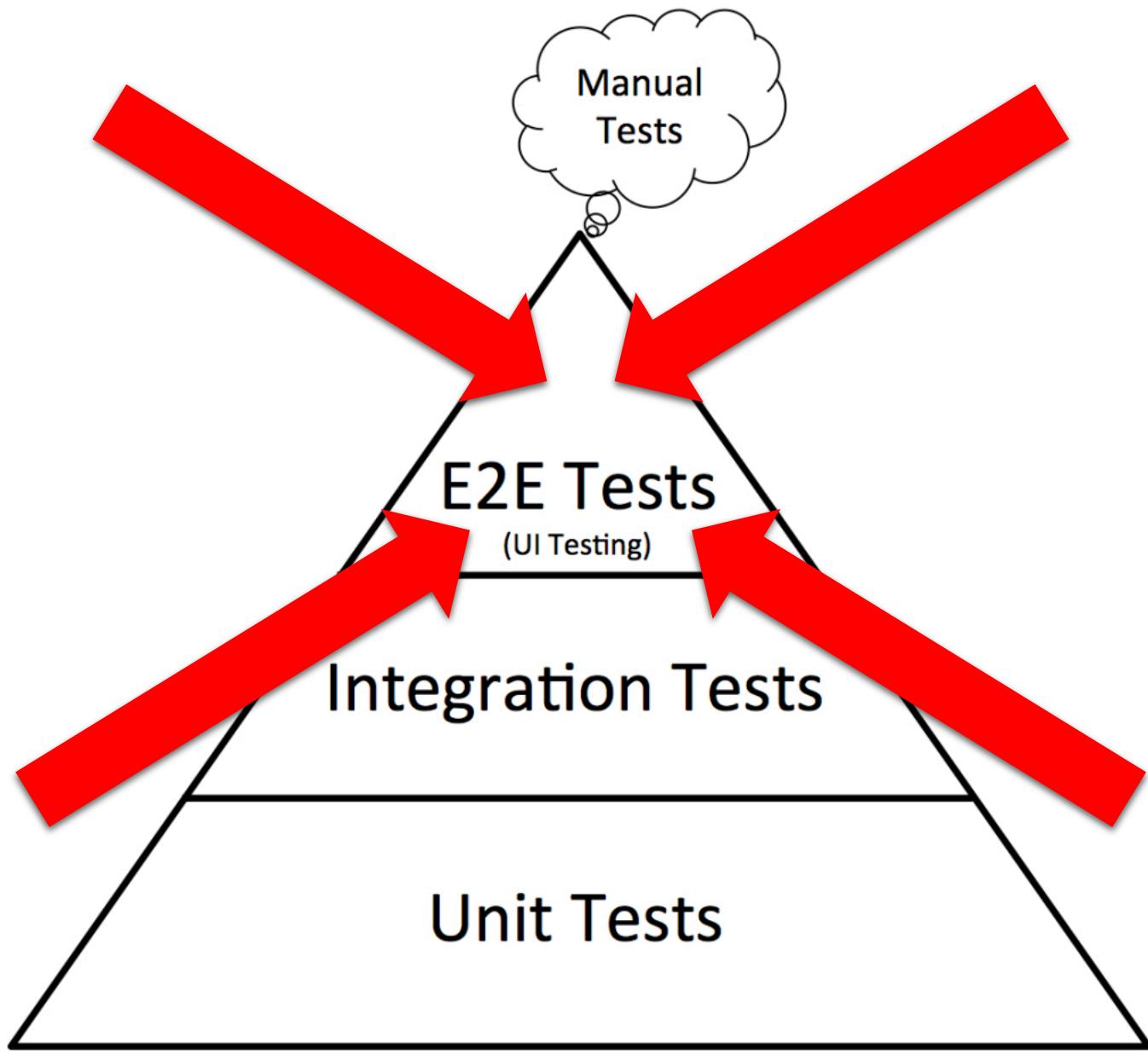
Александр Иосса

+7-999-155-06-87  
iossa92@gmail.com  
telegram: @Aiossa  
@software\_engineering\_practice

Руководитель команды Frontend  
разработки в Diginavis. Software  
Engineer

Из Питера. Сдауншифтил в Иннополис  
4 года назад



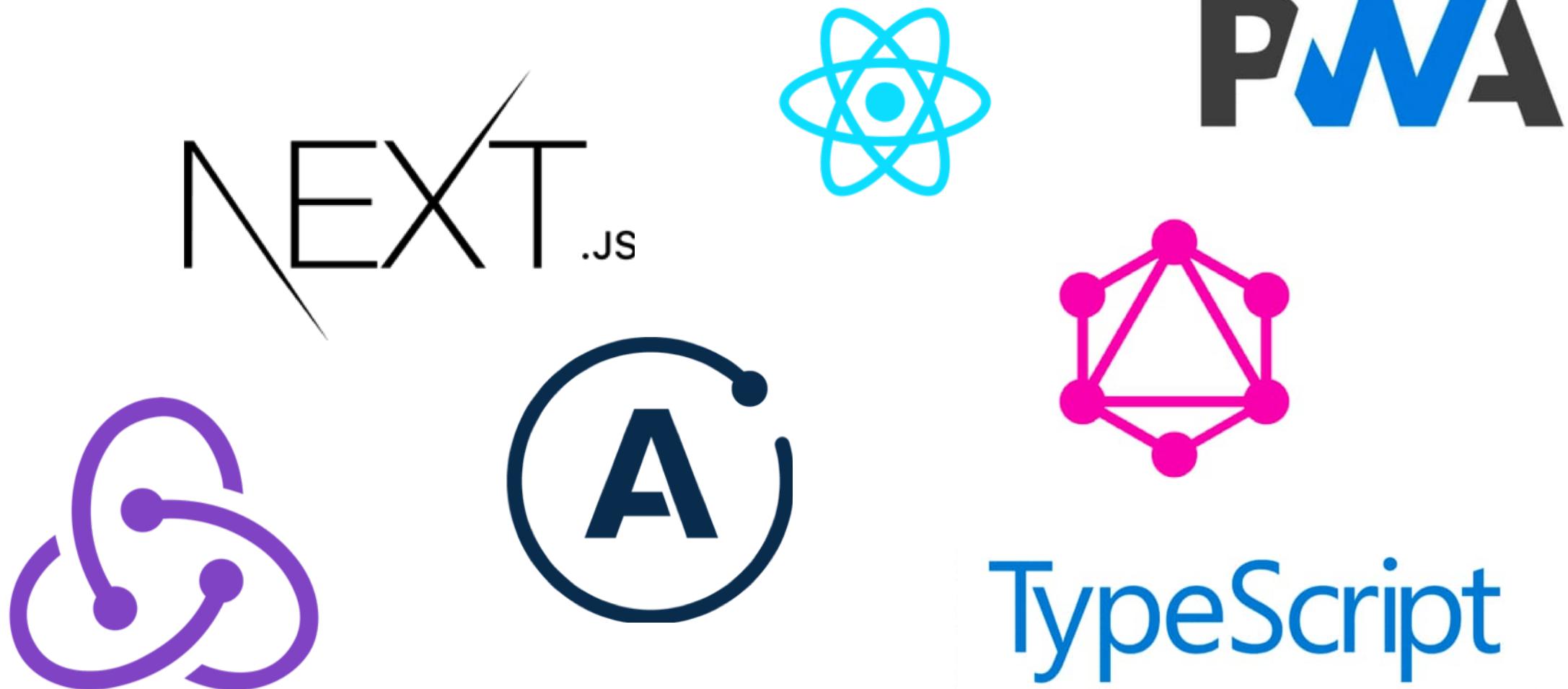


An aerial photograph of the Gazprom Arena stadium in St. Petersburg, Russia. The stadium features a large, white, dome-shaped roof supported by a network of cables and masts. It is situated on a peninsula, with a bridge visible in the background. The surrounding area includes roads, green spaces, and other urban infrastructure. The sky is filled with warm, orange and yellow hues of a setting sun.

ПИШЕМ SELENIUM ТЕСТЫ?

Я ЗНАЮ ПРОЕКТЫ ПОПРОЩЕ

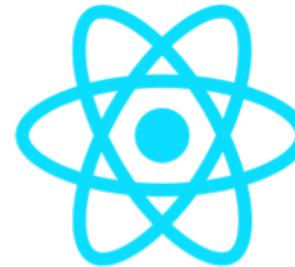
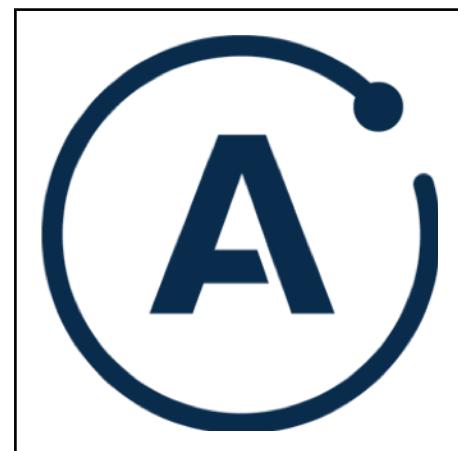
# Наш стэк



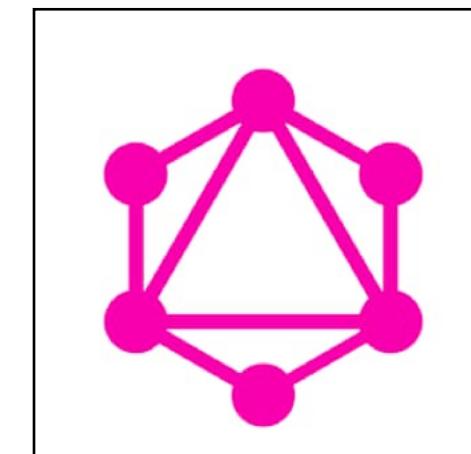
# Наш стэк



NEXT  
.JS



PWA

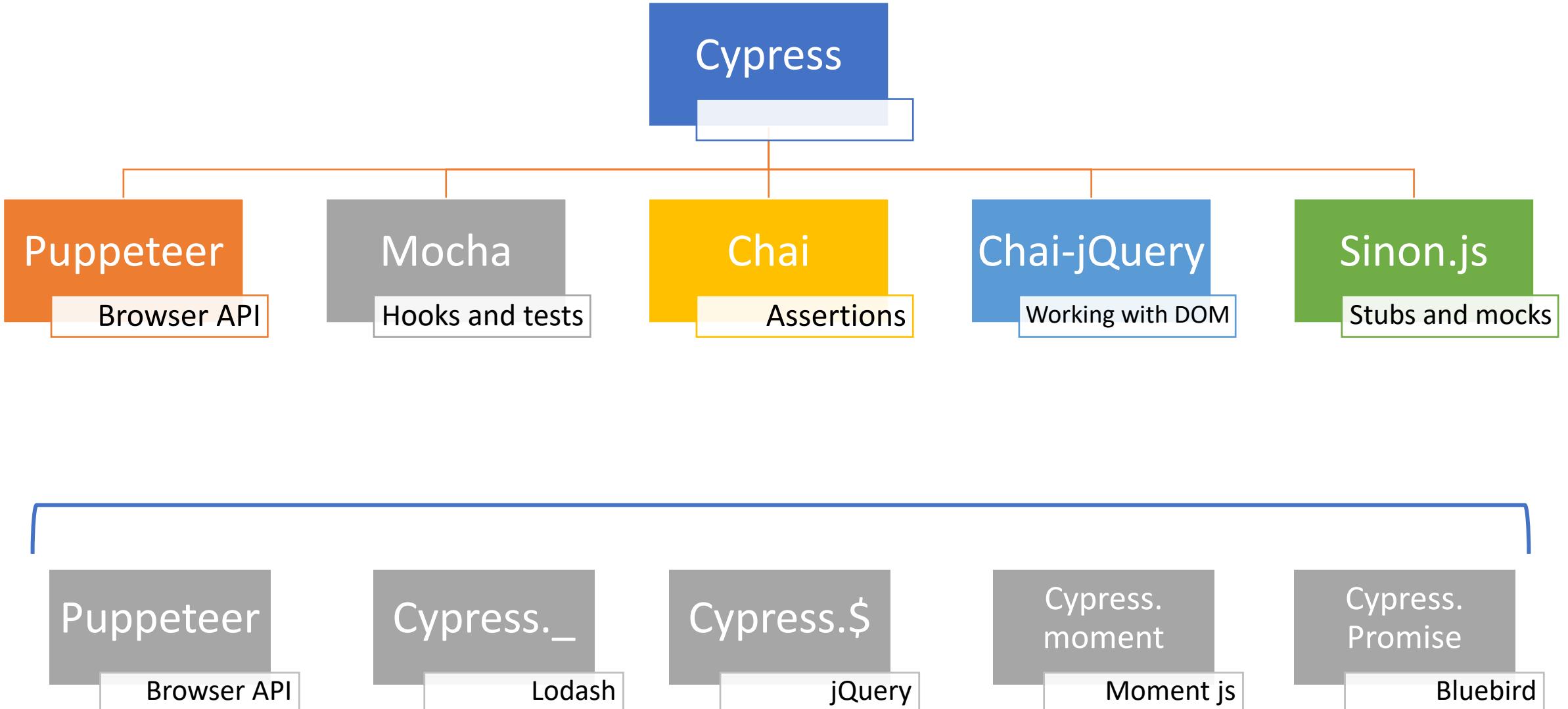


TypeScript

# Cypress

- Отличная обёртка на Puppeteer
- Headless режим (на базе Electron)
- Замечательная документация и поддержка
- Большое количество плагинов (повторы тестов, скриншот тестирование, загрузка файлов и т д)
- GUI интерфейс





# X-Path, CSS selectors



```
1 When('User clicks {string} button', (name: string) => {
2   cy.get(`[data-testid=${name}Button]`)
3     .click();
4});
```

# Chai assertions



```
1 Then('User sees {string} on the screen', (message: string) => {
2   cy.get('body').should('contain', message);
3});
```

# Cypress commands API

Навигация
Cookies && LocalStorage
Перехват http запросов

and	each	invoke	readFile	submit
as	end	its	reload	task
blur	eq	last	request	then
check	exec	location	root	tick
children	filter	log	route	title
clear	find	next	screenshot	trigger
clearCookie	first	nextAll	scrollIntoView	type
clearCookies	fixture	nextUntil	scrollTo	uncheck
clearLocalStorage	focus	not	select	url
click	focused	parent	server	viewport
clock	get	parents	setCookie	visit
closest	getCookie	parentsUntil	should	wait
contains	getCookies	pause	siblings	window
dblclick	go	prev	spread	within
debug	hash	prevAll	spy	wrap
document	hover	prevUntil	stub	writeFile

# Cypress retry-ability

Если assertion свалится, то cypress будет повторять get()  
до положительного исхода или до истечения таймаута

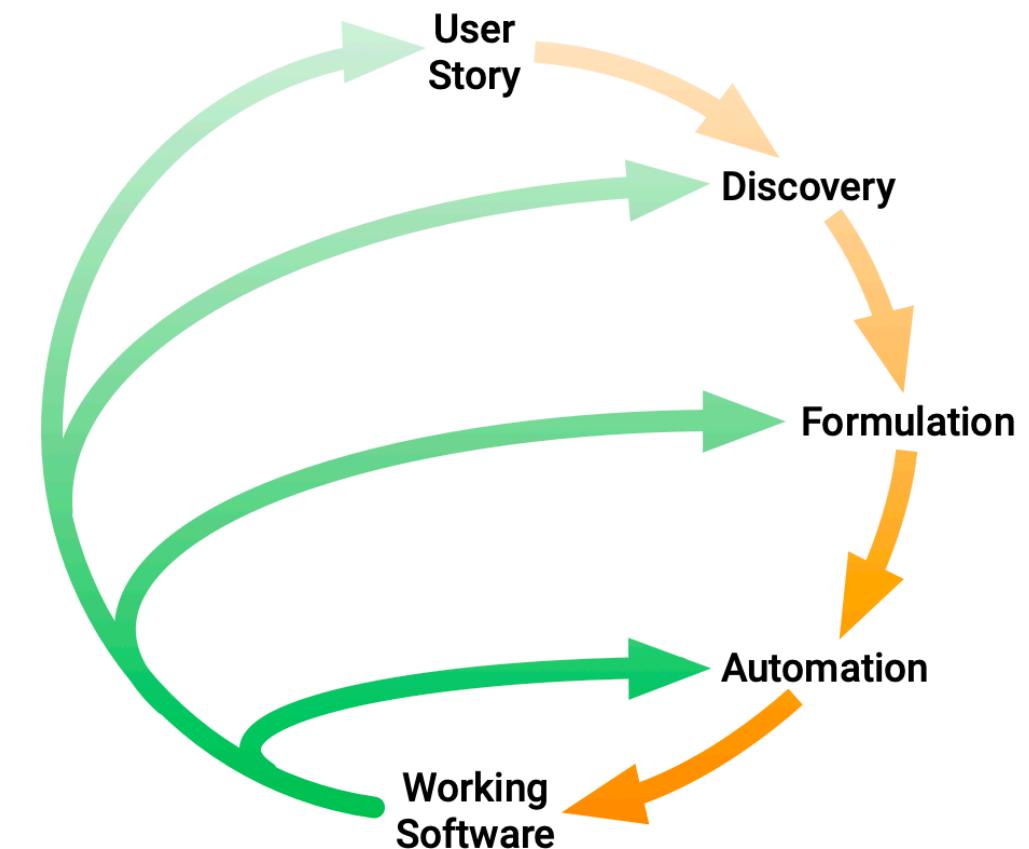


```
1 cy.get('.todo-list li') //command  
2   .should('have.length', 2); //assertion
```



# BDD development process

- Пишем User story/Job story, обговариваем конкретные примеры
- Документируем эти примеры так, чтобы они могли быть автоматизированы
- Разрабатываем. Сначала пишем тесты, затем пишем функционал



Вместе с этим товаром покупают



Телеприставка Apple TV 4K 64 ГБ

15 990 ₽

[Купить](#)



Кабель Apple USB-C для зарядки  
(1 м)

1 990 ₽

[Купить](#)



Телеприставка Apple TV 4K 32 ГБ

13 990 ₽

[Купить](#)



Кабель Apple USB-C для зарядки  
(2 м)

1 990 ₽

[Купить](#)



# Создание подборки товаров

Подборки

Русский

Сохранить ▾

Основная информация

Название  
Техника Apple

Товары в подборке

Apple AirPods (X) Apple iMac 27" (X) Apple MacBook Pro 13" (X)

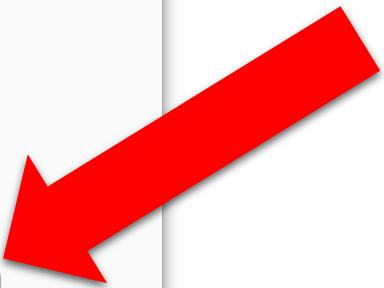
Apple Mac Pro (X) Apple AirPods 2 (X) Apple Watch Series 4 (X) ▾

Apple Mac mini (X)

# Сценарии

Операторы: Gherkin syntax (Domain specific language)

- Feature
- Scenario
- Given
- When
- Then
- And
- But
- Examples



```
1 Feature: Selections creation and modification
2
3 Background:
4   Given User is logged in
5   And User is on "/main/selections" screen
6
7 Scenario: Create selection
8   When User sees the "selections" table
9   And User clicks "create new selection" button
10  And User sees the "/main/selection" screen
11  And User inputs "New selection" in "name"
12  And User searches product "Морковь" and selects it
13  And User clicks submit button
14  Then User sees confirmation message
```

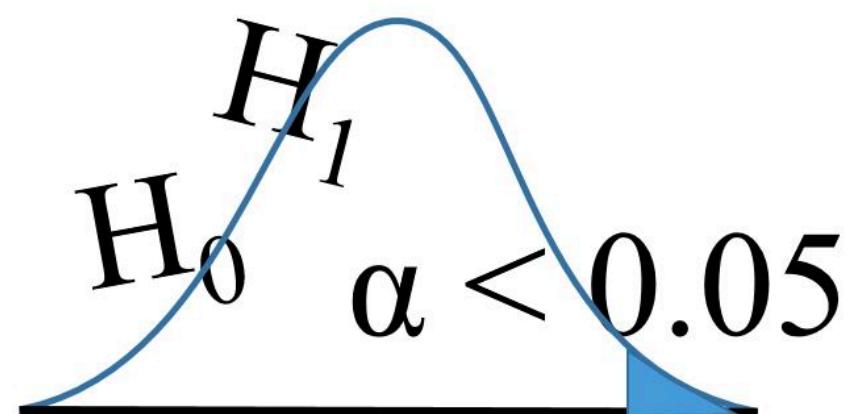
# Step definitions



```
1 When('User clicks {string} button', (name: string) => {
2   cy.get(`[data-testid=${name}Button]`)
3     .click();
4 });
```

# Гипотеза

- BDD тесты писать проще:
  - Кривая обучения ниже
  - Легче структурировать код
  - В дальнейшем достаточно легко подключить тестировщиков

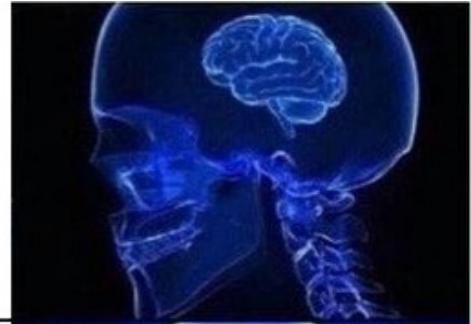


# Почему не работает

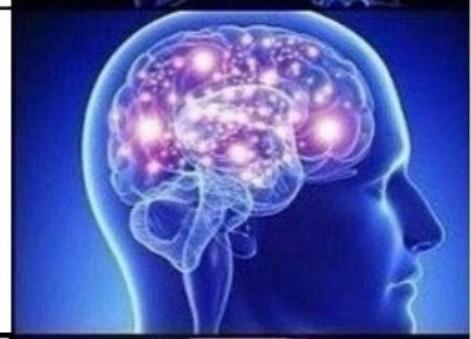
- Мы продуктовая команда, но:
  - Тогда эти сценарии надо обсуждать в первую очередь с дизайнером
  - Сценарии, утверждённые дизайнером не всегда котируются разработчиками)
  - Макеты – лучшее требование как для разработчиков, так и для Product owner'a. Тогда зачем эти сценарии?

Зачем  
придерживаться BDD,  
когда можно просто  
использовать DSL?

**MANUAL  
E2E TESTS**



**AUTOMATED  
E2E TESTS**



**BDD  
TESTING**



**USE DSL,  
NOT BDD**





# Создание и редактирование подборки товаров



```
1 Feature: Selections creation and modification
2
3 Scenario: Create selection
4 When User sees the "selections" table
5 And User clicks "create new selection" button
6 And User sees the "/main/selection" screen
7 And User inputs "New selection" in "name"
8 And User searches product "Морковь" and selects it
9 And User clicks submit button
10 Then User sees confirmation message
```



```
1 Feature: Selections creation and modification
2
3 Background:
4   Given User is logged in
5   And User is on "/main/selections" screen
6
7 Scenario: Create selection
8   When User sees the "selections" table
9   And User clicks "create new selection" button
10  And User sees the "/main/selection" screen
11  And User inputs "New selection" in "name"
12  And User searches product "Морковь" and selects it
13  And User clicks submit button
14  Then User sees confirmation message
```





```
1 Given('User is logged in', () => {
2   return cy.fixture('ls.json').then(async data => {
3     window.localStorage.setItem('apollo-cache-persist', JSON.stringify(data));
4   });
5 });
```



```
1 Feature: Selections creation and modification
2
3 Background:
4   Given User is logged in
5   And User is on "/main/selections" screen
6
7 Scenario: Create selection
8   When User sees the "selections" table
9   And User clicks "create new selection" button
10  And User sees the "/main/selection" screen
11  And User inputs "New selection" in "name"
12  And User searches product "Морковь" and selects it
13  And User clicks submit button
14  Then User sees confirmation message
15  And Remove selection
```



# Что у нас с переиспользованием?



```
1 Feature: Selections creation and modification
2
3   Background:
4     Given User is logged in
5     And User is on "/main/selections" screen
6
7   Scenario: Create selection
8     When User sees the "selections" table
9     And User clicks "create new selection" button
10    And User sees the "/main/selection" screen
11    And User inputs "New selection" in "name"
12    And User searches product "Морковь" and selects it
13    And User clicks submit button
14    Then User sees confirmation message
15    And Remove selection
```

# Что у нас с переиспользованием?



```
1 Feature: Selections creation and modification
2
3 Background:
✓ 4 Given User is logged in
✓ 5 And User is on "/main/selections" screen
6
7 Scenario: Create selection
✓ 8 When User sees the "selections" table
✓ 9 And User clicks "create new selection" button
✓ 10 And User sees the "/main/selection" screen
✓ 11 And User inputs "New selection" in "name"
✓ 12 And User searches product "Морковь" and selects it
✓ 13 And User clicks submit button
✓ 14 Then User sees confirmation message
15 And Remove selection
```

# Статистика

- Около 30 фич
- Фича содержит 2-3 сценария
- Сценарий ~ 7 шагов (мин 4, макс 16)
- 30 переиспользуемых шагов
- На 1 фичу 1 непереиспользуемый шаг





# Не надо так делать

```
1 Feature: Selections creation and modification
2
3 Scenario: Create selection
4   When User sees the "selections" table
5   And User clicks "create new selection" button
6   And User fills required fields
7   And User clicks submit button
8   Then User sees confirmation message
```



# Не надо так делать

```
1 Feature: Selections creation and modification
2
3 Scenario: Create selection
4   When User sees the "selections" table
5   And User clicks "create new selection" button
6   And User inputs "New selection" in "name"
7   And User sees "New selection" in "name"
8   And User searches product "Морковь" and selects it
9   And User clicks submit button
10  Then User sees confirmation message
```

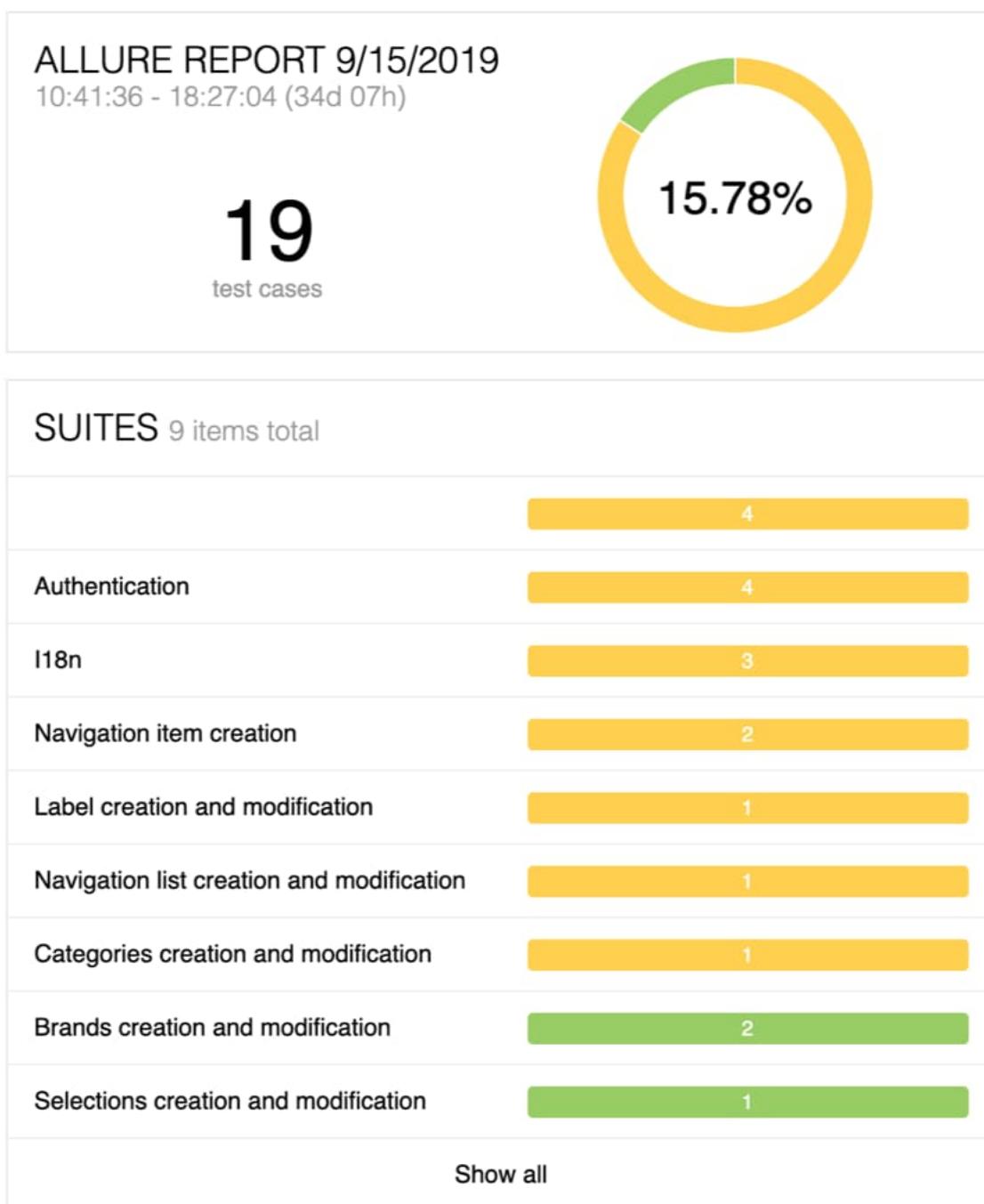


# Где место таким тестам в ci/cd процессе?



 **Allure**

-  Overview
-  Categories
-  Suites
-  Graphs
-  Timeline
-  Behaviors
-  Packages





## Suites

Status: 0  16  3  0  0[Overview](#)[Categories](#)[Suites](#)[Graphs](#)[Timeline](#)[Behaviors](#)[Packages](#)

En

&lt; Collapse

[order](#) [name](#) [duration](#) [status](#)

Marks:

> component.tree.unknown	4
↳ Authentication	4
! #4 Failure Log in	11s 847ms
! #2 Log out	11s 635ms
! #1 Refresh and stay logged in	12s 007ms
! #3 Success Log in	11s 403ms
> Brands creation and modification	2
↳ Categories creation and modification	1
! #1 Create category	12s 940ms
↳ I18n	3
! #2 Choosing language	6s 454ms
! #1 Default language by user location (Russia)	11s 611ms
! #3 Refresh page and keep the chosen language	6s 245ms
> Label creation and modification	1
> Navigation item creation	2
> Navigation list creation and modification	1
> Selections creation and modification	1

I18n.Choosing language

## Broken Choosing language

[Overview](#) [History](#) [Retries](#)

Timed out retrying: Expected to find element: '[data-test=chooseLanguageButton]', but never found it.

Categories: Test defects

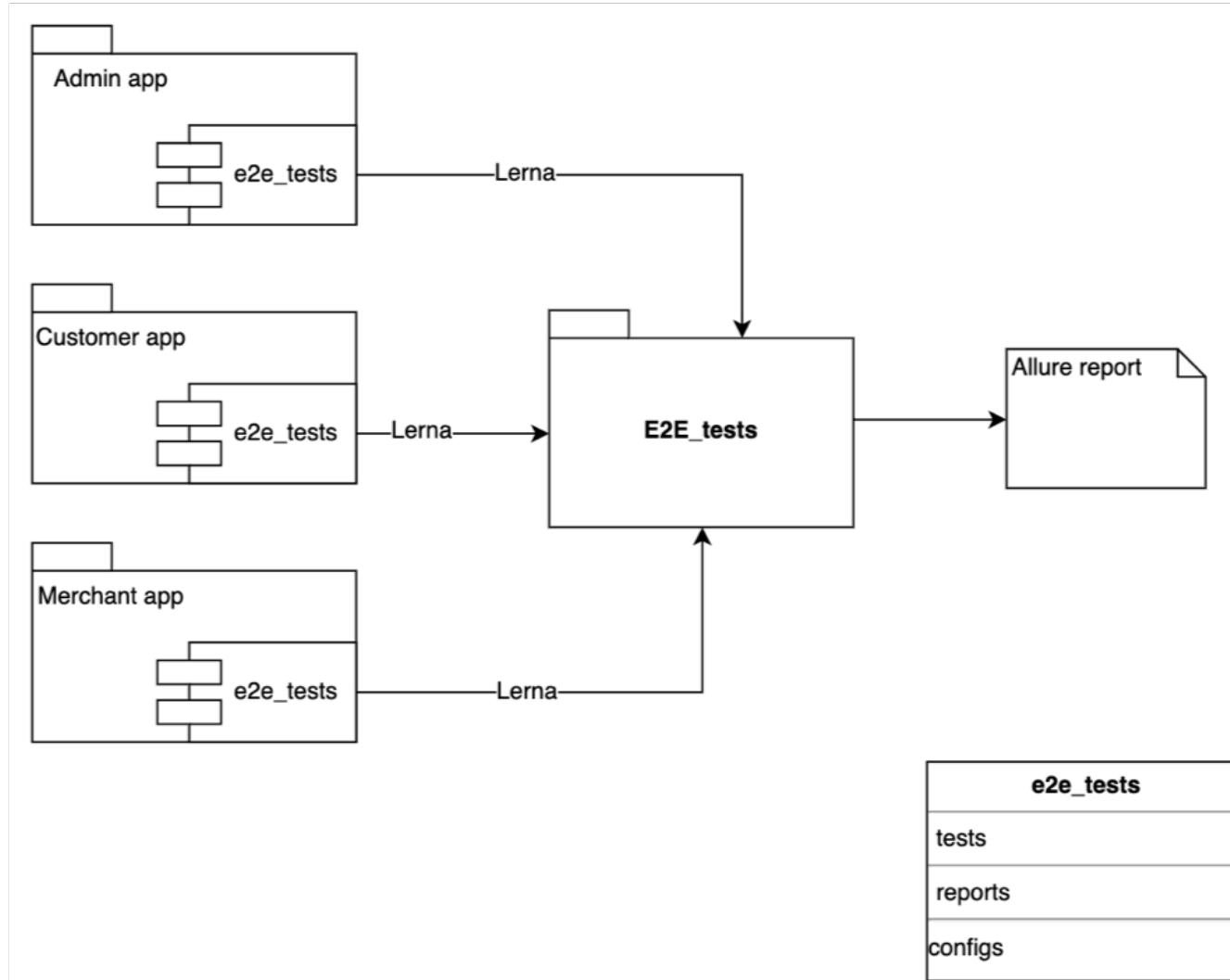
Severity: normal

Duration: 0 6s 454ms

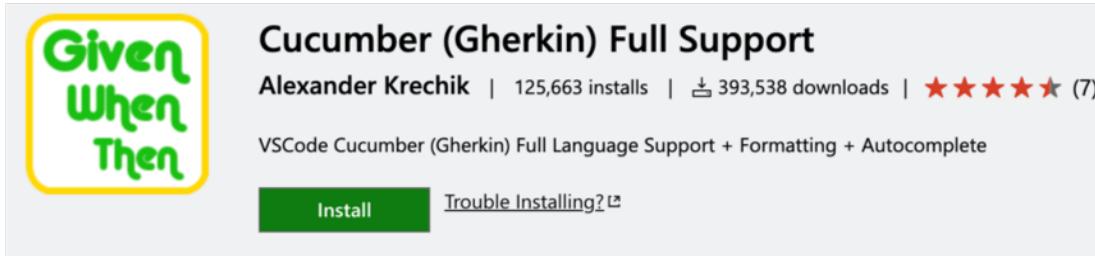
### Execution

No information about test execution is available.

# Тестирование релиза



# Dev tools & dependencies



A dark-themed terminal window is shown with a portion of a package.json file. The code lists various dependencies and devDependencies, including cypress, cucumber, and typescript.

```
1 "devDependencies": {  
2   "@cypress/webpack-preprocessor": "^4.1.0",  
3   "cucumber": "^5.1.0",  
4   "cypress": "^3.4.1",  
5   "cypress-cucumber-preprocessor": "^1.16.0",  
6   "ts-loader": "^6.0.4",  
7   "typescript": "^3.6.2",  
8   "webpack": "^4.39.3"  
9 },  
10 "dependencies": {  
11   "@types/node": "^12.7.2",  
12   "cosmiconfig": "^5.2.1",  
13   "cypress-file-upload": "^3.3.3",  
14   "cypress-image-snapshot": "^3.1.1",  
15   "cypress-plugin-retries": "^1.2.2",  
16   "mocha": "^6.2.0",  
17   "mocha-allure-reporter": "^1.4.0",  
18   "path-to-regexp": "^3.1.0"  
19 },  
20 }
```

# Выводы

- Дружим с TS
- Плагины VS Code помогают работать
- Отличный старт для небольших и средних команд
- Низкий порог входа за счёт:
  - Документации
  - Готовых бойлерплейтов
- Проблемы с параллельным запуском
- Отсутствует кросс-браузерность



# Cypress Dashboard Service



**Seed**

**Free**



**Sprout**

**\$99 / month**



**Tree**

**\$199 / month**



**Forest**

**\$399 / month**

Users **3**

Test recordings **500**

Support **Community**

Test parallelization **✓**

Run grouping **✓**

GitHub Integration **✓**

Users **5**

Test recordings **25000**

Support **Email**

Test parallelization **✓**

Run grouping **✓**

GitHub Integration **✓**

Users **10**

Test recordings **75000**

Support **Email**

Test parallelization **✓**

Run grouping **✓**

GitHub Integration **✓**

Users **20**

Test recordings **150000**

Support **Email**

Test parallelization **✓**

Run grouping **✓**

GitHub Integration **✓**

**Get Started**

**Get Started**

**Get Started**

**Get Started**

# Ссылки

[Cucumber](#)

[Cypress](#)

[Cypress + Cucumber + TS](#)

[Lerna](#)

# О себе



Александр Иосса

+7-999-155-06-87  
iossa92@gmail.com  
telegram: @Aiossa  
@software\_engineering\_practice

Руководитель команды Frontend  
разработки в Diginavis. Software  
Engineer

Из Питера. Сдауншифтил в Иннополис  
4 года назад

