

Тестирование

как перестать факанить
и начать жить

Игорь Шеко
Lead Frontend developer
Voximplant



4 POINT PLAN:

1. Писать код
2. Релизить
3. Обновлять
4. \$\$\$

Игорь!
Не рендерится Support!

Игорь! Не рендерится Support!



Игорь! Не рендерится Support!

Мы не меняли Support.vue в ЭТОМ релизе!!!

Кто трогал Support?!

Сами нашли или Клиент репортнул?

Ну нет, он не должен был сломаться!

Давайте откат или Хотфикс!

Игорь! Не рендерится Support!

Мы не меняли Support.vue в ЭТОМ релизе!!!

Кто трогал Support?!

Сами нашли или Клиент репортнул?

Ну нет, он не должен был сломаться!

Давайте откат или Хотфикс!

Игорь! Не рендерится Support!

1 стадия – Отрицание

Мы не меняли Support.vue в ЭТОМ релизе!!!

Кто трогал Support?!

Сами нашли или Клиент репортнул?

Ну нет, он не должен был сломаться!

Давайте откат или Хотфикс!

Игорь! Не рендерится Support!

1 стадия – Отрицание

Мы не меняли Support.vue в ЭТОМ релизе!!!

2 стадия – Гнев

Кто трогал Support?!

Сами нашли или Клиент репортнул?

Ну нет, он не должен был сломаться!

Давайте откат или Хотфикс!

Игорь! Не рендерится Support!

1 стадия – Отрицание

Мы не меняли Support.vue в ЭТОМ релизе!!!

2 стадия – Гнев

Кто трогал Support?!

3 стадия – Торг

Сами нашли или Клиент репортнул?

Ну нет, он не должен был сломаться!

Давайте откат или Хотфикс!

Игорь! Не рендерится Support!

1 стадия – Отрицание

Мы не меняли Support.vue в ЭТОМ релизе!!!

2 стадия – Гнев

Кто трогал Support?!

3 стадия – Торг

Сами нашли или Клиент репортнул?

4 стадия – Депрессия

Ну нет, он не должен был сломаться!

Давайте откат или Хотфикс!

Игорь! Не рендерится Support!

1 стадия – Отрицание

Мы не меняли Support.vue в ЭТОМ релизе!!!

2 стадия – Гнев

Кто трогал Support?!

3 стадия – Торг

Сами нашли или Клиент репортнул?

4 стадия – Депрессия

Ну нет, он не должен был сломаться!

5 стадия – Принятие

Давайте откат или Хотфикс!

Похоже, наш новый джун..



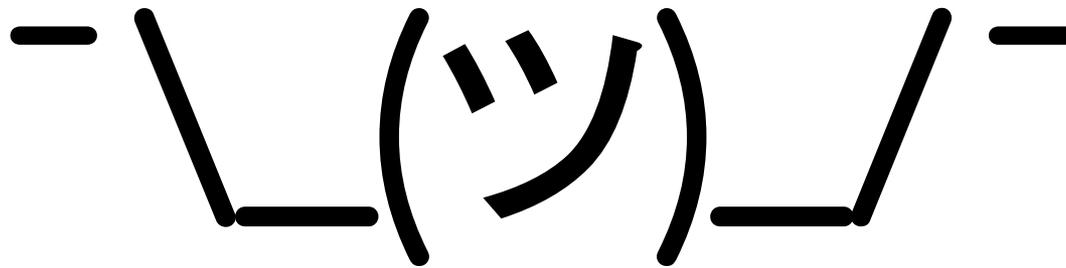
...решил оптимизировать
старый код...



...jQuery...не можно...



Человеческий фактор.



Что делать???

Я фронтенд!

1. Я пишу код
2. Мой код прекрасен
3. У меня все работает
4. Сроки всегда горят
5. Я тестирую фичи руками
6. Не надо чинить то что не сломалось
7. Сломается, починим
8. Это фронт, тут все визуально
9. Бизнес за это не платит
10. Не умею (лапки)

Исторически есть 2 пути

1 - QA

2 -

Исторически есть 2 пути

1 - QA

2 - Репрессии

Я
QA
не
любил



Всем спасибо!

Всем спасибо!
Вступление
закончилось!

Когда тестировать
не надо?

1
Ваша команда
божественна!

2 Проект только
начался

PARTY HARD



Типы авто-тестов

Unit тесты

End-to-end тесты

Мутационные тесты

Unit тесты

- Тестируем каждую функцию отдельно

Unit тесты

- Тестируем каждую функцию отдельно

```
function foo(a,b){  
    return a>b;  
}
```

Unit тесты

- Тестируем каждую функцию отдельно

Ожидания (Expectations)

- $a > b$ return true
- $a = b$ return false
- $a < b$ return false

Unit тесты

- Тестируем каждую функцию отдельно

Ожидания (Expectations)

- `a > b` return true
- `a = b` return false
- `a < b` return false
- `typeof a === 'string' && typeof b === 'object'` return ???
- `a & b` undefined return ???

Unit тесты

- Тестируем каждую функцию отдельно
- Помогает осознать и уточнить требования

Unit места

VOX-2177

Реализовать RSS

[Edit](#) [Comment](#) [Assign](#) [Close Issue](#) [Reopen Issue](#) [Admin](#) ▼

Type: + New Feature Status: RESOLVED [\(View workflow\)](#)
Priority: ^ Major Resolution: Fixed
Component/s: [REDACTED]
Labels: [REDACTED]
Sprint:

Assignee: [REDACTED]
[Assign to me](#)

Reporter:  Igor Sheko

Votes: 0

Watchers: 2 [Stop watching this issue](#)

Created: [REDACTED]

Updated: [REDACTED]

Resolved: [REDACTED]

Agile

[View on Board](#)

Stride

[Create room](#)

Description

Для блога нужно реализовать RSS для каждой категории. Можно для всего блога, если необходимо. Это нужно для отображения новостей и важных уведомлений в панели управления Voximplant

Ожидается:

- RSS доступен для каждой категории отдельно
 - RSS доступен для также категории, не отображаемой на сайте
 - RSS запись содержит дату публикации
 - RSS запись содержит заголовок публикации
 - RSS запись содержит ссылку на публикацию
 - RSS запись содержит заголовок публикации
 - RSS запись содержит название категории публикации
- Опционально:
- RSS доступен для всего блога целиком
 - Общий RSS отдает опубликованные записи из открытых категорий блога

InVision

 [Attach prototype](#)

Add a prototype's **share URL** to communicate your vision. [Learn more](#) • [Create a new prototype](#)

Attachments

...

Unit тесты

- Тестируем каждую функцию отдельно
- Помогает осознать и уточнить требования
- Тесты «документируют» код

Unit тесты

- Тестируем каждую функцию отдельно
- Помогает осознать и уточнить требования
- Тесты «документируют» код
- Код должен быть написан «тестируемо»

Unit тесты

- Тестируем каждую функцию отдельно
- Помогает осознать и уточнить требования
- Тесты «документируют» код
- Код должен быть написан «тестируемо»

```
function levenshtein_distance(a,b){  
    // ... около 70 строк  
}
```

Unit тесты

- Тестируем каждую функцию отдельно
- Помогает осознать и уточнить требования
- Тесты «документируют» код
- Код должен быть написан «тестируемо»
- Иногда приводят к TDD

Unit места для...



Unit места для...



- Jest
 - Mocha + Chai
 - Karma
- + jsdom

Unit места для...



Vue Test Utils

Руководство
по тестированию Vue

<https://vue-test-utils.vuejs.org/>

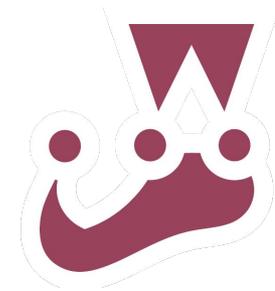
Unit места для меня



+



+



Unit тесты

```
const runSafeTest = ({testMsg, apiRequest, checkFn}: TestConfig)
=> {
  test(`${testMsg} or throw 'Network error'`, async () => {
    try {
      const data = await apiRequest;
      checkFn(data);
    } catch (e) {
      await expect(apiRequest).rejects.toThrow(e);
    }
  });
};
```



Unit тесты

```
const runSafeTest = ({testMsg, apiRequest, checkFn}: TestConfig)
=> {
  test(`${testMsg} or throw 'Network error'`, async () => {
    try {
      const data = await apiRequest;
      checkFn(data);
    } catch (e) {
      await expect(apiRequest).rejects.toThrow(e);
    }
  });
};
```



Unit тесты

```
const addApplicationResponseTest: TestConfig = {  
  testMsg: 'new Api.Client.requestWithAuth() with valid auth data  
and params should return AddApplicationResponse',  
  apiRequest:  
validApi.requestWithAuth(Api.Endpoint.AddApplication,  
addAppParams),  
  checkFn: (data: Api.Endpoint) => {  
    for (let field in Responses.AddApplication) {  
      expect(data).toHaveProperty(field);  
    }  
  }  
};  
  
runSafeTest(addApplicationResponseTest);
```



Снэпшоты компонентов

Снапшоты компонентов

```
it('renders correctly', () => {  
  const tree = renderer  
    .create(<Link page="http://www.example.com">Example</Link>)  
    .toJSON();  
  expect(tree).toMatchSnapshot();  
});
```



Снапшоты компонентов

```
it('renders correctly', () => {  
  const tree = renderer  
    .create(<Link page="http://www.example.com">Example</Link>)  
    .toJSON();  
  expect(tree).toMatchSnapshot();  
});
```



Снапшоты компонентов

```
exports[`renders correctly`] = `  
  <a  
    className="normal"  
    href="http://www.example.com"  
    onMouseEnter={ [Function] }  
    onMouseLeave={ [Function] }  
  >  
    Example  
  </a>  
`;  
;
```

<https://jestjs.io/docs/ru/snapshot-testing>



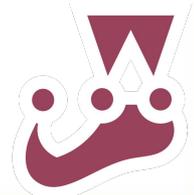
Снапшоты компонентов проблемы

- Изменился css - снапшот этого не покажет



Снапшоты компонентов проблемы

- Изменился css - снапшот этого не покажет
- Ложные срабатывания



Снапшоты компонентов проблемы

- Изменился css - снапшот этого не покажет
- Ложные срабатывания
 - Изменился текст, изменился и снапшот

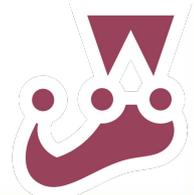


Снапшоты компонентов проблемы

- Изменился css - снапшот этого не покажет
- Ложные срабатывания
 - Изменился текст, изменился и снапшот
 - Вложенные компоненты

PageMenu

```
<CoolMenu>  
  <MenuItem>  
</CoolMenu>
```



End-to-end места

- работают с готовым приложением

End-to-end тесты

- работают с готовым приложением
- запускаются в браузере или чем-то похожем

End-to-end тесты

- работают с готовым приложением
- запускаются в браузере или чем-то похожем
- если слышите Selenium, то это точно они

End-to-end тесты

- работают с готовым приложением
- запускаются в браузере или чем-то похожем
- если слышите Selenium, то это точно они
- для тестирования не нужно знать особенности архитектуры приложения

What You See Is What You Test

End-to-end для Vue.js



Nightwatch

Ролук



End-to-end для Vue.js



Nightwatch

```
nightwatch -- bash -- 92x20
[~/workspace/nightwatch] on master* $ ./bin/nightwatch -t examples/tests/googleDemoTest.js
Starting selenium server... started - PID: 25060

[Google Demo Test] Test Suite
=====

Running: Demo test Google
✓ Element <body> was visible after 224 milliseconds.
✓ Testing if the page title equals "Google".
✓ Testing if element <input[type=text]> is visible.
✓ Element <button[name=btnG]> was visible after 332 milliseconds.
✓ Testing if element <ol#rso li:first-child> contains text: "Rembrandt - Wikipedia".

OK. 5 assertions passed. (7.729s)

[~/workspace/nightwatch] on master* $ █
```

Простота

client

```
// Run for 1 minute
.pause(60*sec)

// Click the share button
.click('.VideoToolbar-item--screenshot-test-screenshot-button')
.pause(300)

.rtcEvent('Screen Share ' + agentSession, 'global')
.rtcScreenshot('screen share')
// Open a new tab with a YouTube video and switch to that tab
.execute("window.open('" + videoURL + "', '_blank')")
.pause(2*sec)
.windowHandles(function (result) {
    var newWindow;
    newWindow = result.value[2];
    this.switchWindow(newWindow);
})
// Run for another minute
.pause(30*sec)
.rtcScreenshot('watching')
.pause(30*sec)
```



Прочтота

client

```
// Run for 1 minute
.pause(60*sec)

// Click the share button
.click('.VideoToolbar-item--screenshot-test-screenshot-button')
.pause(300)

.rtcEvent('Screen Share ' + agentSession, 'global')
.rtcScreenshot('screen share')
// Open a new tab with a YouTube video and switch to that tab
.execute("window.open('" + videoURL + "', '_blank')")
.pause(2*sec)
.windowHandles(function (result) {
    var newWindow;
    newWindow = result.value[2];
    this.switchWindow(newWindow);
})
// Run for another minute
.pause(30*sec)
.rtcScreenshot('watching')
.pause(30*sec)
```



Промона

client

```
// Run for 1 minute
.pause(60*sec)

// Click the share button
.click('.VideoToolbar-item--screenshot-jstest-screenshot-button')
.pause(300)

.rtcEvent('Screen Share ' + agentSession, 'global')
.rtcScreenshot('screen share')
// Open a new tab with a YouTube video and switch to that tab
.execute("window.open('" + videoURL + "', '_blank')")
.pause(2*sec)
.windowHandles(function (result) {
    var newWindow;
    newWindow = result.value[2];
    this.switchWindow(newWindow);
})
// Run for another minute
.pause(30*sec)
.rtcScreenshot('watching')
.pause(30*sec)
```



Варианты раннеров

- Chrome driver

<http://chromedriver.chromium.org/>

Варианты раннеров

- Chrome driver
- Selenium WebDriver



Варианты раннеров

- Chrome driver
- Selenium WebDriver
- Сервисы



Плюсы и минусы

- Можно протестировать все браузеры
- Нельзя ускориться

Selenium as a Service



Мутационные тесты

JS !==



Мутационные тесты



Мутационные тесты

```
function isUserOldEnough(user) {  
  return user.age >= 18;  
}
```



Мутационные тесты

```
function isUserOldEnough(user) {  
    return user.age >= 18;  
}  
function isUserOldEnough(user) {  
    return user.age < 18;  
}  
function isUserOldEnough(user) {  
    return false;  
}  
function isUserOldEnough(user) {  
    return true;  
}
```



Мутационные тесты

Mutant killed: /yourPath/yourFile.js: line 10:27

Mutator: BinaryOperator

```
-         return user.age >= 18;  
+         return user.age > 18;
```

Mutant survived: /yourPath/yourFile.js: line 10:27

Mutator: RemoveConditionals

```
-         return user.age >= 18;  
+         return true;
```



test coverage

100% кода покрыто
А все еще что-то
ломается



False test coverage

Unit тесты

```
const runSafeTest = ({testMsg, apiRequest, checkFn}: TestConfig)
=> {
  test(`${testMsg} or throw 'Network error'`, async () => {
    try {
      const data = await apiRequest;
      checkFn(data);
    } catch (e) {
      await expect(apiRequest).rejects.toThrow(e);
    }
  });
};
```

Вопросы?

Игорь Шeko
Lead Frontend developer
Voximplant