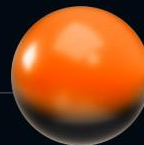


Конвертируем Selenium тесты в Playwright



Александр
Шарков

Монитор электрик



aleks.sharkov@gmail.com

HEISENBUG



План

Почему мы решили перейти на PW

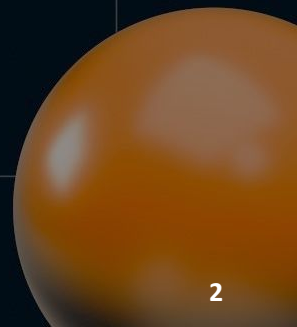
Отказ от WebDriver

Паттерн "Proxy"

Замена API

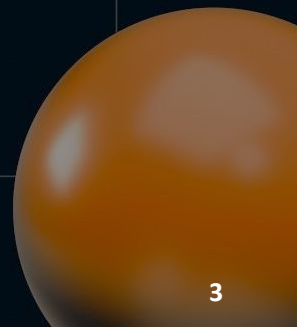
Исправление ошибок

Результаты



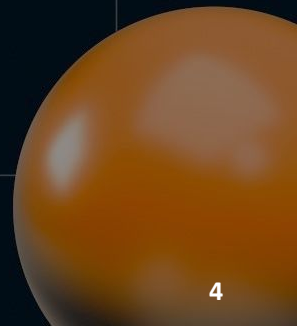
ЭКСПОЗИЦИЯ

- Долго сидели на Selenium
- Молодежь просит Playwright
- Цель: стабильность и скорость



ЭКСПОЗИЦИЯ

- Долго сидели на Selenium
- Молодежь просит Playwright
- Цель: стабильность и скорость



Playwright: Web testing without drama



Андрей Лушников

<https://www.youtube.com/watch?v=ttODF00XWis>

Почему мы решили перейти на PW

<https://playwright.dev>

QA



Selenium



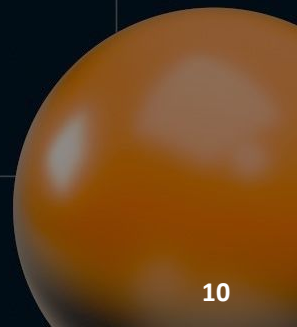
Playwright

Playwright

Selenium

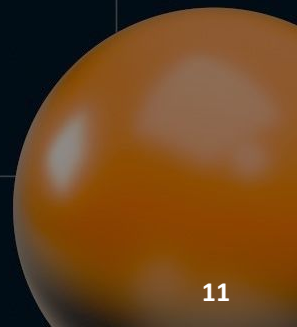


Критерии приемки РВ



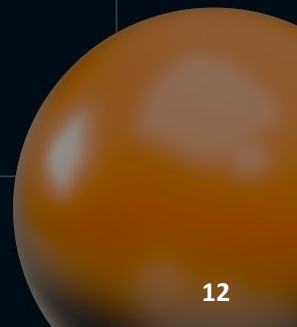
Критерии приемки РW

- Минимум времени на миграцию



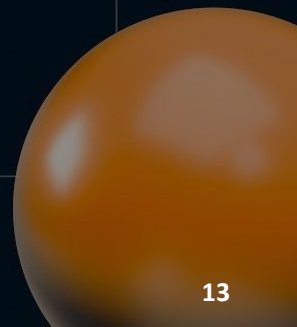
Критерии приемки РВ

- Минимум времени на миграцию
- Стабильные тесты



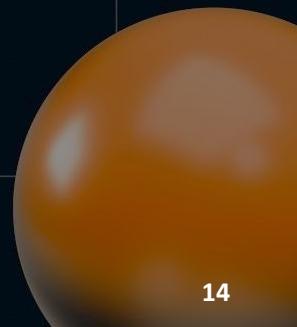
Критерии приемки PW

- Минимум времени на миграцию
- Стабильные тесты
- Тесты быстрее на 30%



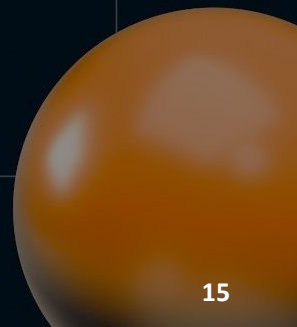
Критерии приемки PW

- Минимум времени на миграцию
- Стабильные тесты
- Тесты быстрее на 30%
- Автономность браузера



Критерии приемки PW

- Минимум времени на миграцию
- Стабильные тесты
- Тесты быстрее на 30%
- Автономность браузера
- Многопоточность



Минимум времени на миграцию



Браузер

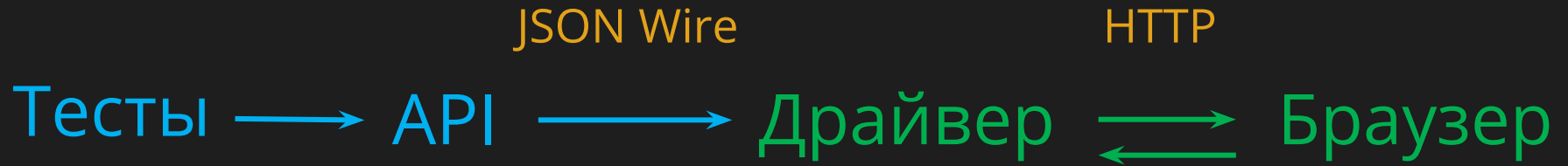


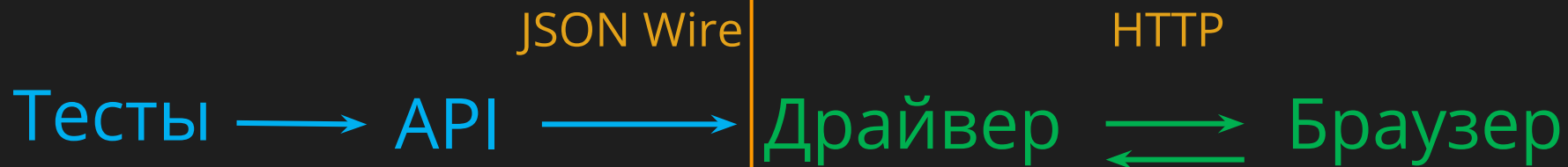
Драйвер $\xrightleftharpoons{\text{HTTP}}$ Браузер





Схема работы тестов на Selenium







Proxy pattern



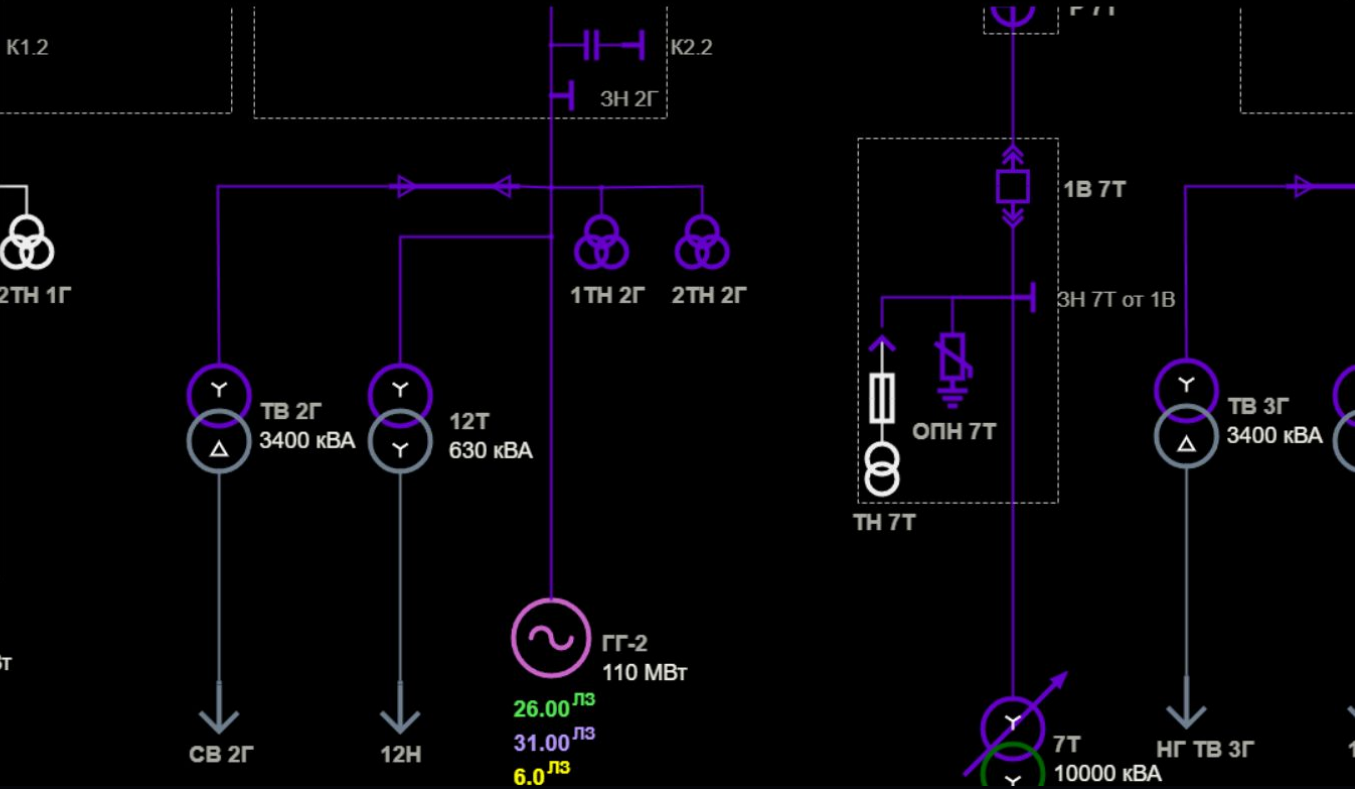
Выбор времени архива

<
АПРЕЛЬ 2024 Г.
>

ПН	ВТ	СР	ЧТ	ПТ	СБ	ВС
25	26	27	28	29	30	31
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	1	2	3	4	5

15 : 22 : 49

Сейчас
Применить
Отменить



Какой здесь драйвер ?

```
Driver.Navigate().GoToUrl("url");
```

```
Driver.FindElement(By.XPath("//app-calendar//dx-number-box"))  
    .SendKeys("15");
```

```
Driver.FindElement(By.XPath("//input[@value='Применить']"))  
    .Click();
```

Playwright server :)

<https://playwright.dev/>

Selenium to PW proxy

```
public interface OpenQA.Selenium.IWebDriver
{
    public string Url { get; set; }
```

Selenium to PW proxy

```
public class WebDriverProxy : OpenQA.Selenium.IWebDriver
{
    public Microsoft.Playwright.IPage Page;

    public string Url
    {
        get => Page.Url;
        set => Page.GotoAsync(value).Wait();
    }
}
```

Selenium to PW proxy

```
public class WebElementProxy : OpenQA.Selenium.IWebElement
{
    public Microsoft.Playwright.ILocator Locator;

    public void Click() => Locator.ClickAsync().Wait();
}
```

Selenium to PW proxy

```
OpenQA.Selenium.IWebDriver Driver;
```

```
public async Task TestInit()  
{  
    var Context = await NewContextAsync();  
    var Page = await NewPageAsync(Context);  
    Driver = new WebDriverProxy(Page);  
}
```

Базовый класс для всех команд

```
[TestClass]
public class AppTests : WebTestClass
{
    [TestMethod]
    public void OpenApp()
    {
        Driver.Navigate().GoToUrl("url");
    }
}
```

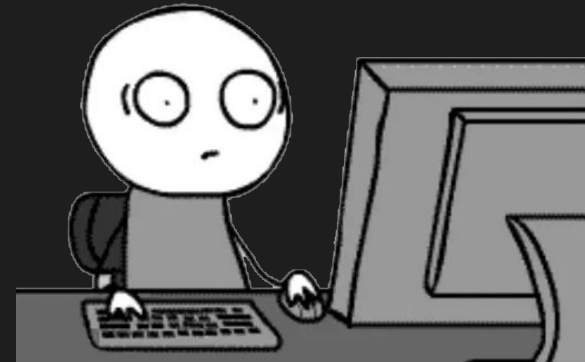


И такое бывает...

```
public class WebElementProxy : OpenQA.Selenium.IWebElement
{
    public void Click()
    {
```

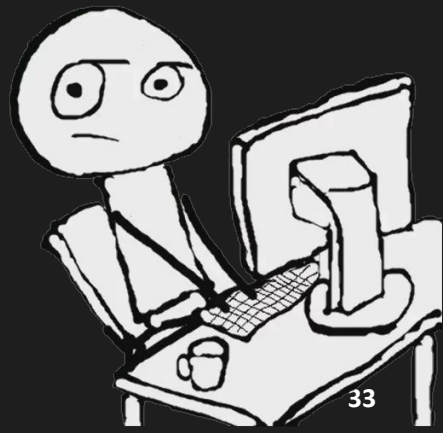
CENSORED

```
Element.Click();
```



Не надо так...

```
public class WebElementProxy : OpenQA.Selenium.IWebElement
{
    public void Click()
    {
        Thread.Sleep(1000);
        Element.Click();
    }
}
```

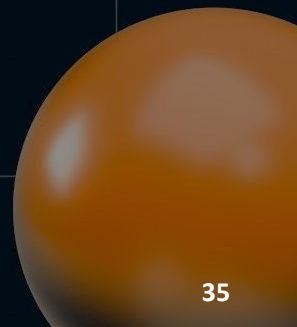


Критерии приемки PW

- Минимум времени на миграцию
- Стабильные тесты
- Тесты быстрее на 30%*
- Автономность браузера
- Многопоточность

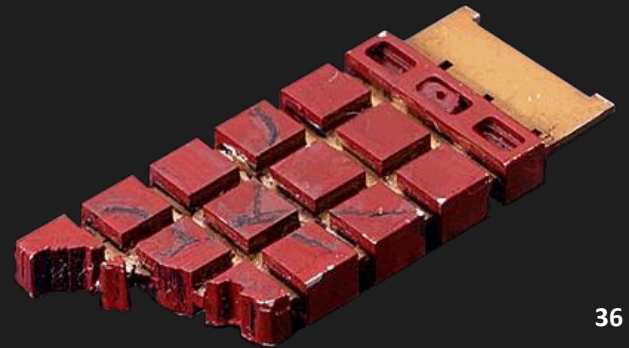
<https://github.com/britka/playwrightium>

Первые проблемы

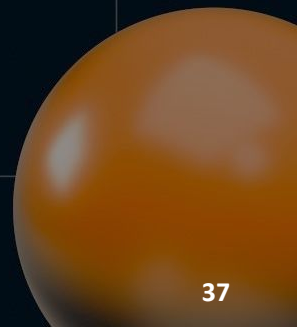


Не смогли поставить последнюю версию

- Конфликт с другими пакетами
- Критическая информационная инфраструктура (КИИ)



strict mode :(



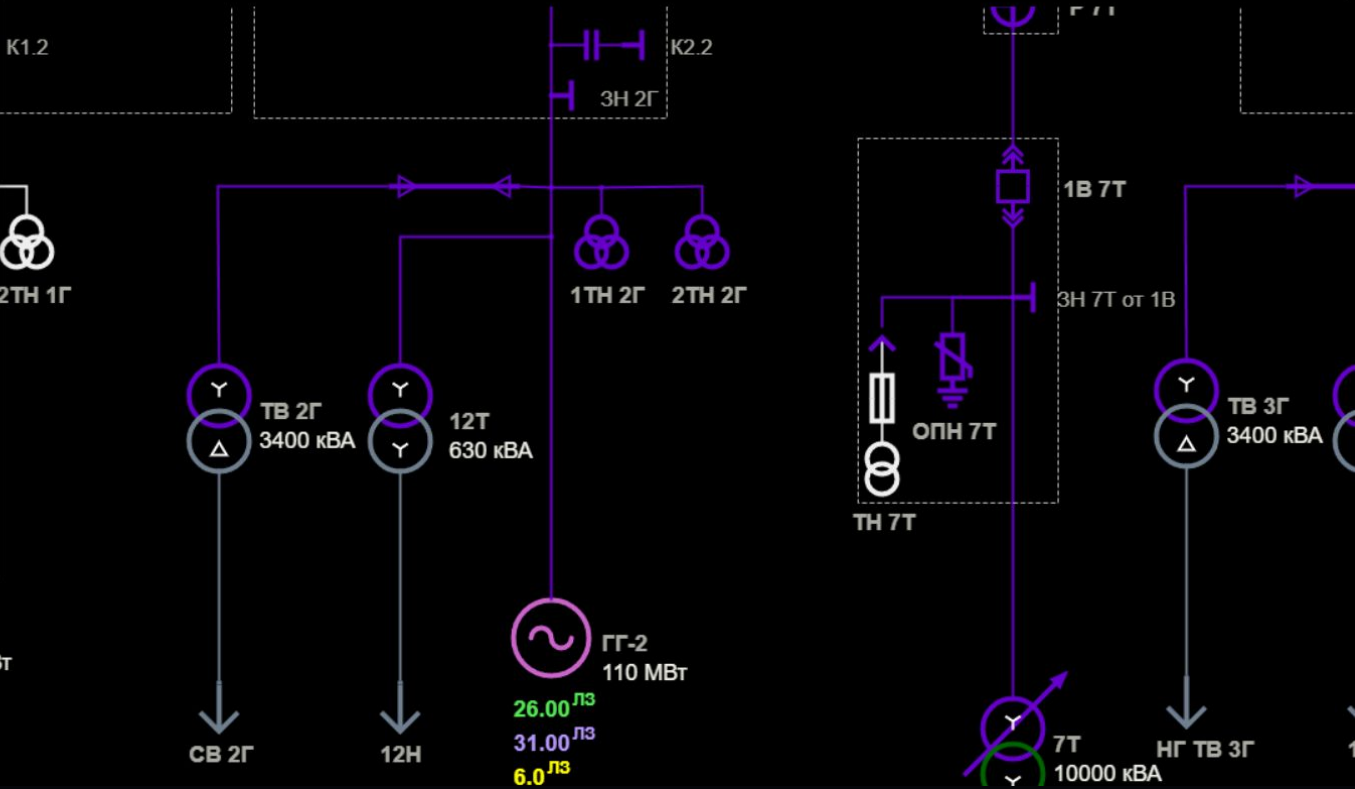
Выбор времени архива

АПРЕЛЬ 2024 Г.

ПН	ВТ	СР	ЧТ	ПТ	СБ	ВС
25	26	27	28	29	30	31
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	1	2	3	4	5

15 : 22 : 49

Сейчас Применить Отменить



```
Driver.FindElement(By.XPath(".//app-calendar//dx-number-box"))  
    .Click();
```

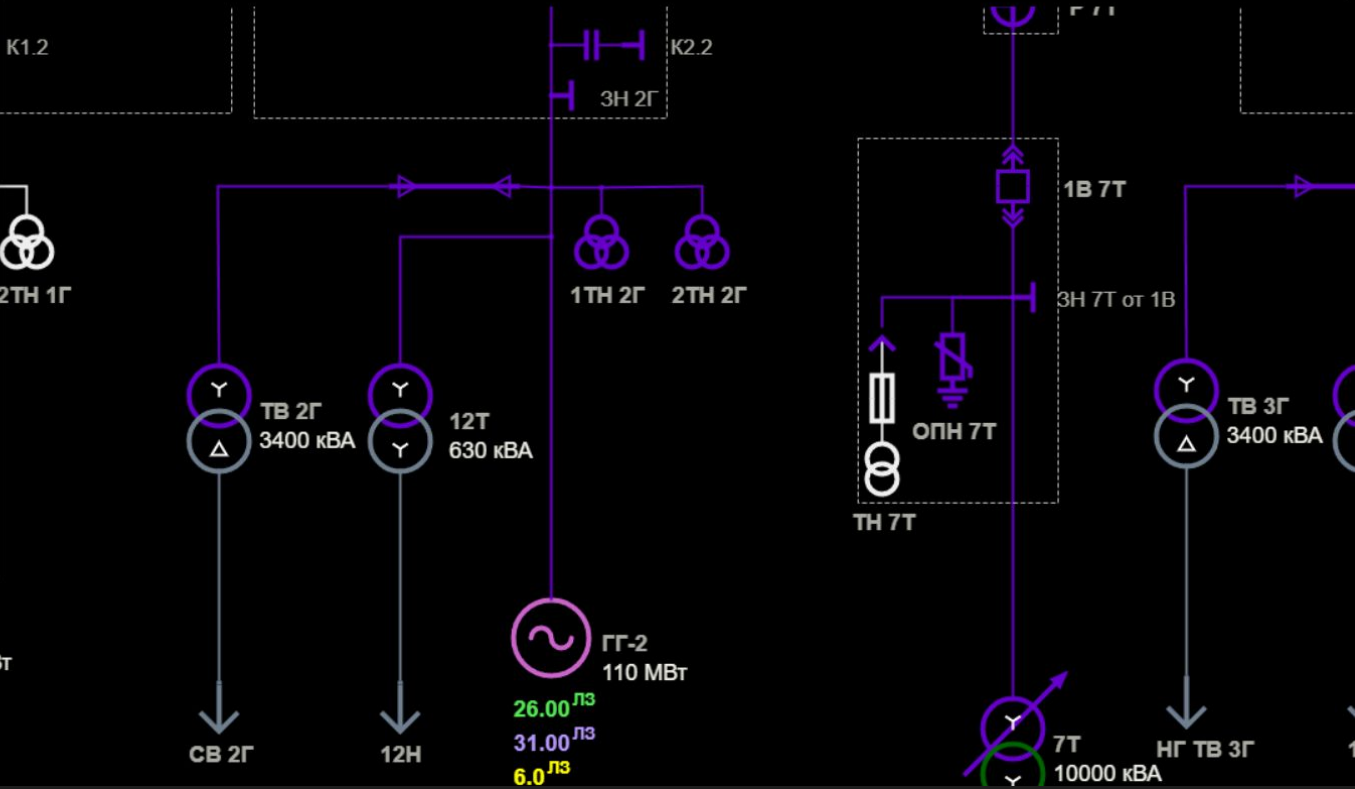


Выбор времени архива X

< АПРЕЛЬ 2024 Г. >

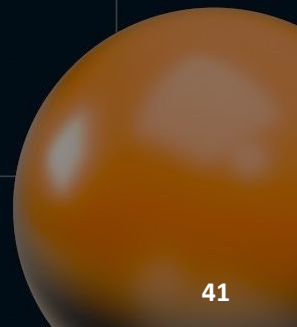
ПН	ВТ	СР	ЧТ	ПТ	СБ	ВС
25	26	27	28	29	30	31
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	1	2	3	4	5

15 : 22 : 49



Критерии приемки PW

- Минимум времени на миграцию
- Стабильные тесты
- Тесты быстрее на 30%*
- Автономность браузера
- Многопоточность



Было

Debug → Открытый браузер

Local → Запуск установленного браузера

Remote → Selenium

.runsettings

```
<TestRunParameters>
```

```
<Parameter name="browser_source" value="local" />
```

```
<Parameter name="browser_name" value="chrome" />
```

```
<Parameter name="browser_version" value="115.0" />
```

```
<Parameter name="browser_repository" value="https://host:port/" />
```

```
<Parameter name="headless" value="true" />
```

```
</TestRunParameters>
```

Chrome for testing

WebServer

- | - Chrome

- | - 113.0

- | - win64

- | - chrome-win64.zip

- | - chromedriver-win64.zip



<https://googlechromelabs.github.io/chrome-for-testing/>

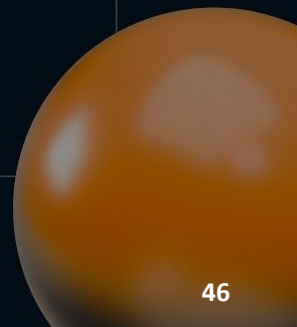
Стало

Debug → Открытый браузер

Local → Установка и запуск браузера

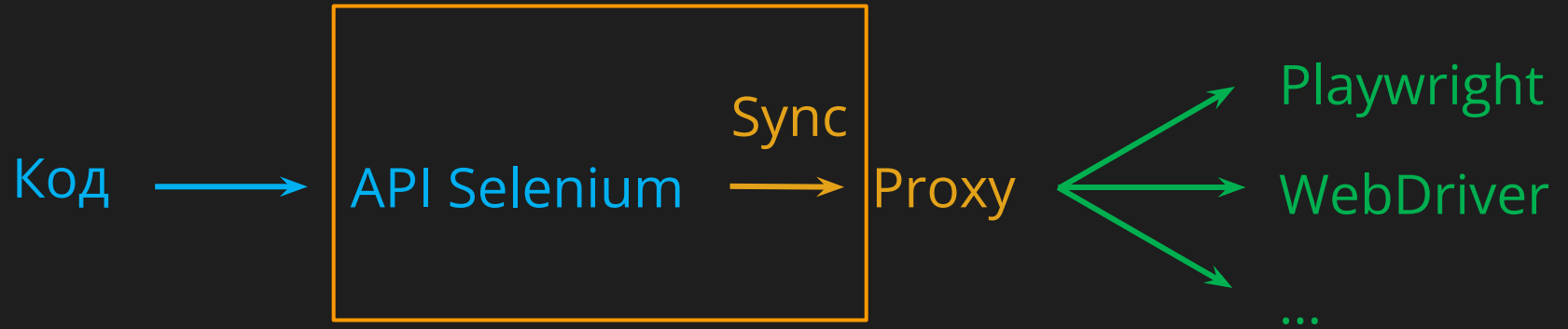
Критерии приемки PW

- Минимум времени на миграцию
- Стабильные тесты
- Тесты быстрее на 30%
- Автономность браузера
- Многопоточность

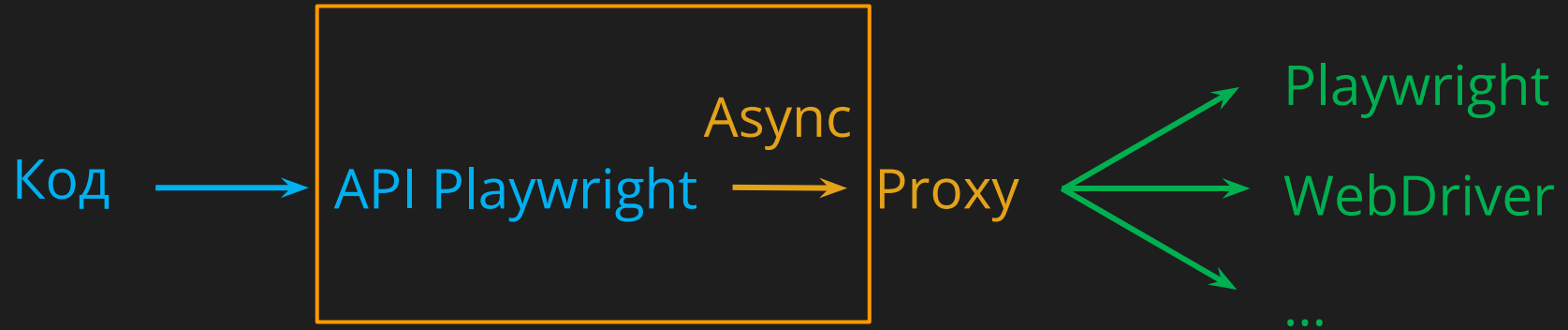




Архитектурный тупик



Архитектурный тупик



Почему мы решили перейти на PW

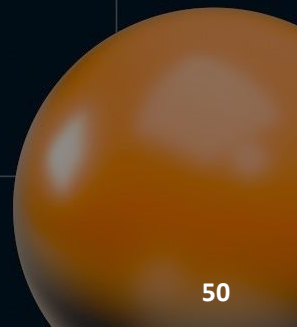
Отказ от WebDriver

Паттерн "Proxy"

Замена API

Исправление ошибок

Результаты



Тест Selenium

```
Driver.Navigate().GoToUrl("url");
```

```
Driver.FindElement(By.XPath("./app-calendar//dx-number-box"))  
    .SendKeys("15");
```

```
Driver.FindElement(By.XPath("./input[@value='Применить']"))  
    .Click();
```

Тест PW очень похож

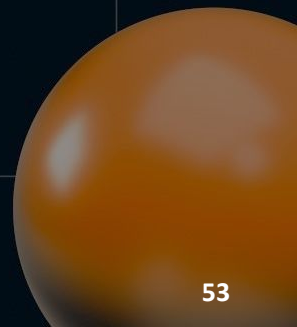
```
await Page.GotoAsync("url");
```

```
await Page.Locator("xpath=//app-calendar//dx-number-box")  
    .FillAsync("15");
```

```
await Page.Locator("xpath=//input[@value='Применить']")  
    .ClickAsync();
```

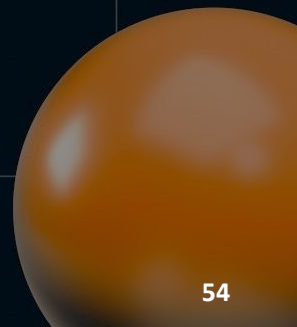
Замена API

- найти инструмент



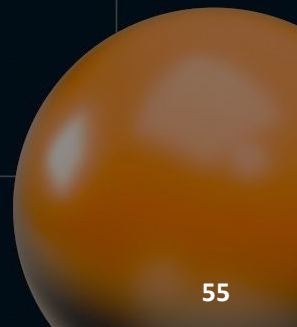
Замена API

- найти инструмент
- учесть асинхрон



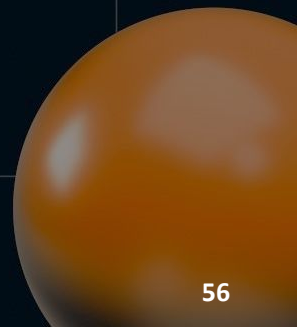
Замена API

- найти инструмент
- учесть асинхрон
- заменить код



Замена API

- найти инструмент
- учесть асинхрон
- заменить код
- исправить ошибки компиляции



Инструменты замены API

- ChatGPT



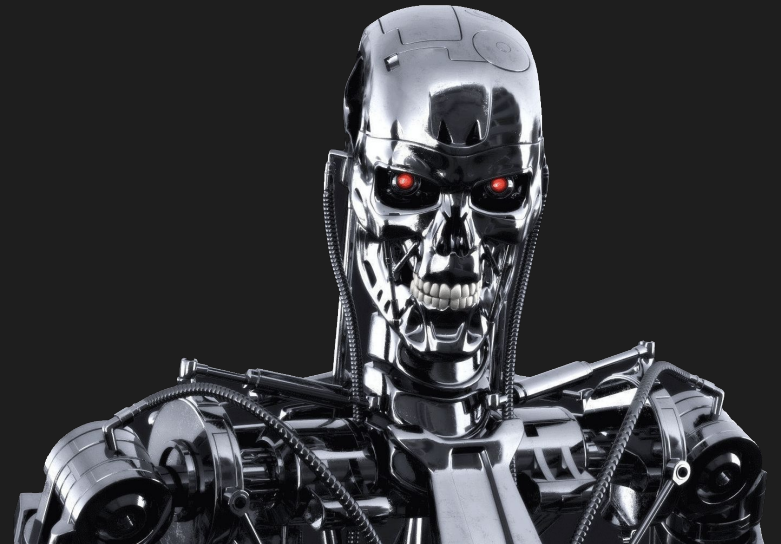
Инструменты замены API

- ChatGPT
- Mistral



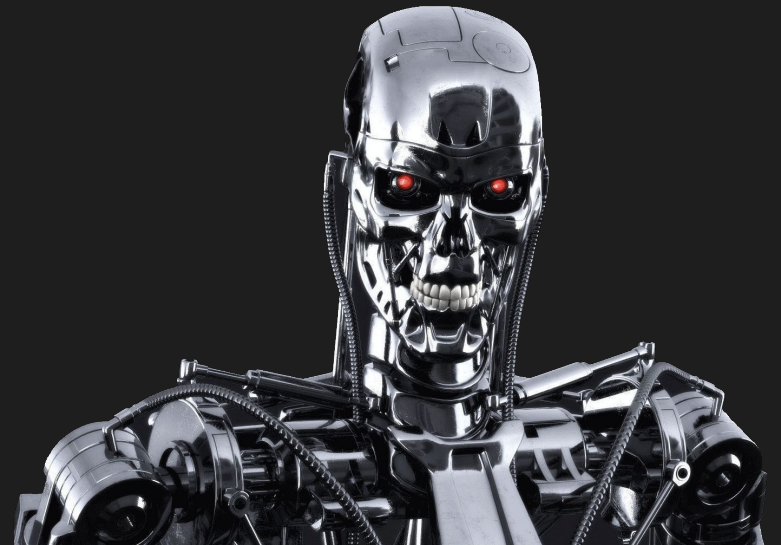
Инструменты замены API

- ChatGPT
- Mistral
- Roslyn API



Инструменты замены API

- ChatGPT
- Mistral
- Roslyn API
- Регулярки



Замена API



Структура проекта

AppTests.csproj

- | - Sources

- | - Pages

 - | - MainPage.cs

 - | - MainPageExt.cs

- | - Tests.cs

Структура проекта

AppTests.csproj

- | - Sources

- | - Pages

 - | - MainPage.cs

 - | - MainPageExt.cs

- | - Tests.cs

Структура проекта

AppTests.csproj

- | - Sources

- | - Pages

- | - MainPage.cs

- | - MainPageExt.cs

- | - Tests.cs

Структура проекта

AppTests.csproj

- | - Sources

- | - Pages

 - | - MainPage.cs

 - | - MainPageExt.cs

 - | - Tests.cs

Замена API

- найти инструмент
- учесть асинхрон
- заменить код
- исправить ошибки компиляции



Синхронность

```
public void Apply()  
{  
    Driver.Locator(" .//input[@value=' Применить ' ]")  
        .Click();  
}
```

Асинхронность

```
public void Apply()  
{  
    await Driver.Locator("xpath=//input[@value='Применить']")  
        .ClickAsync();  
}
```

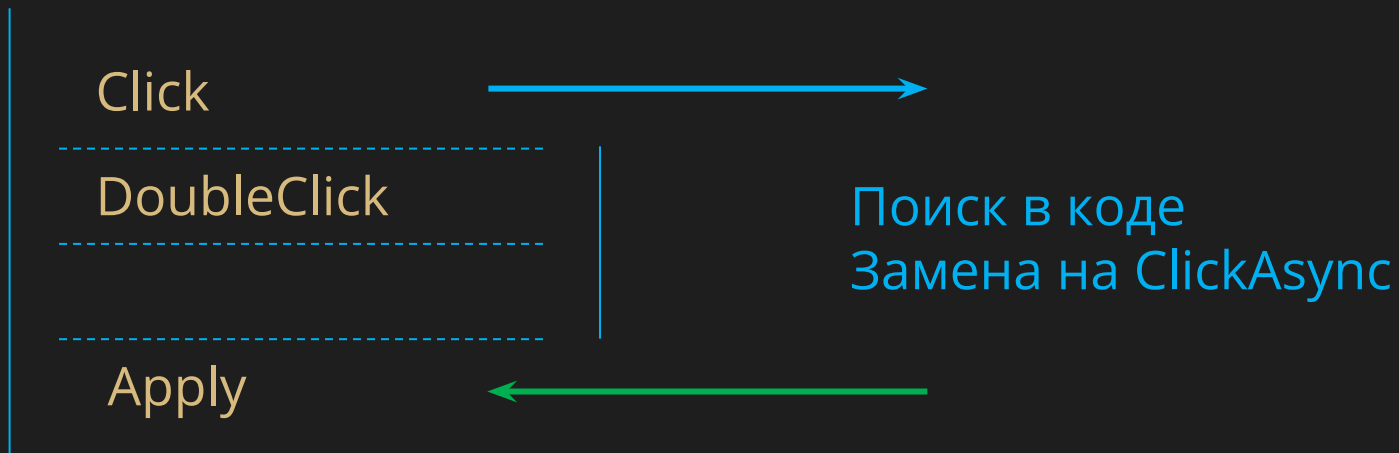
```
public async Task ApplyAsync()  
{  
    await Driver.Locator("xpath=//input[@value='Применить']")  
        .ClickAsync();  
}
```

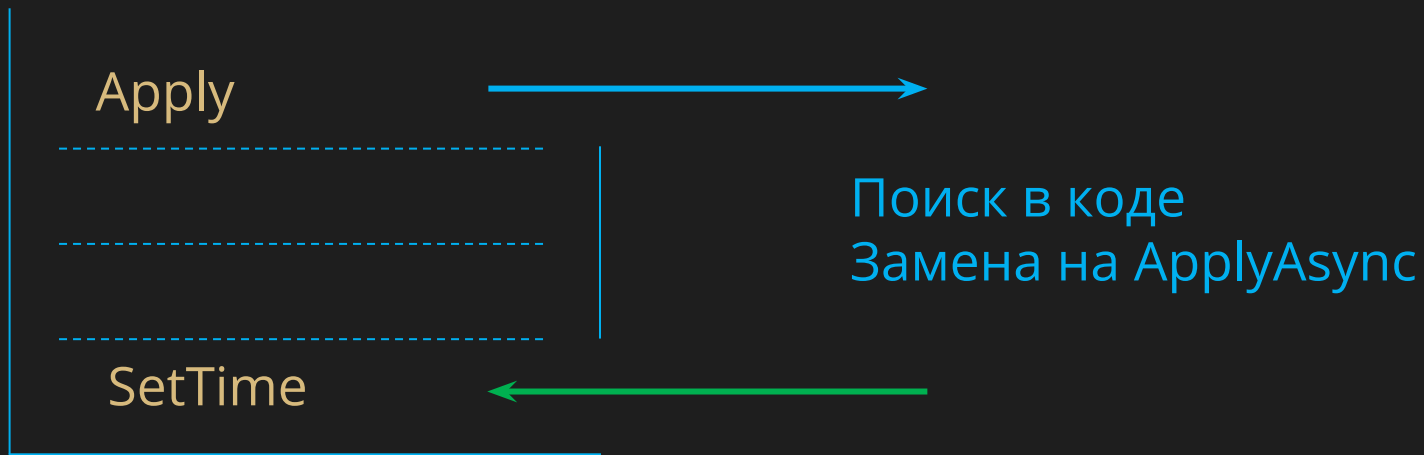
```
[TestMethod]
public async Task TestMethodAsync()
{
    await SetTimeAsync();
}
```

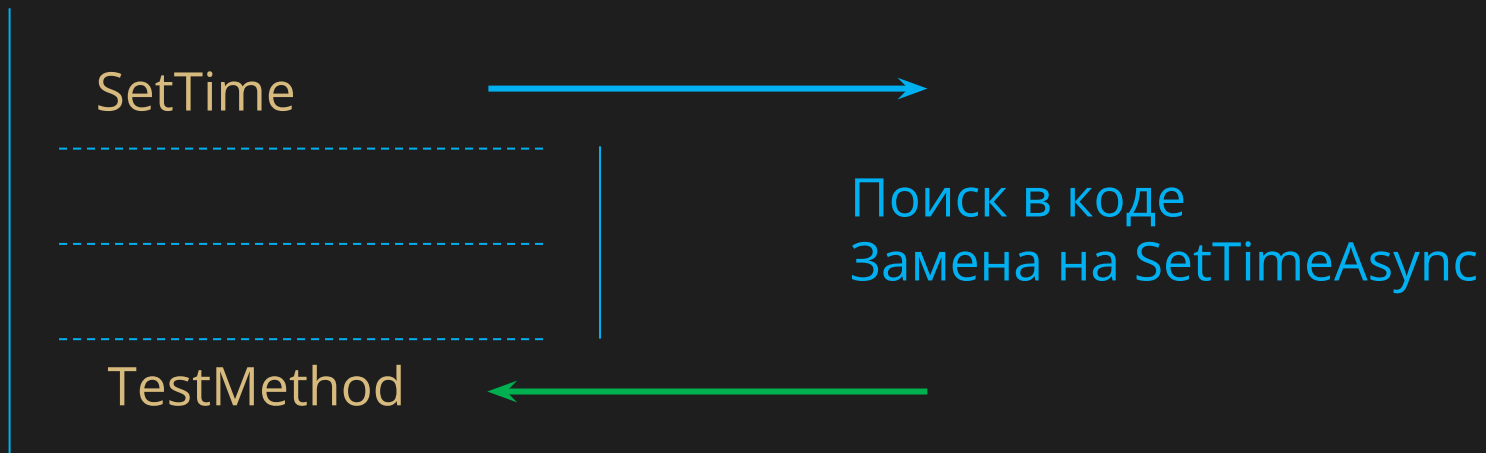
```
public async Task SetTimeAsync() {
    await Page.ApplyAsync();
}
```

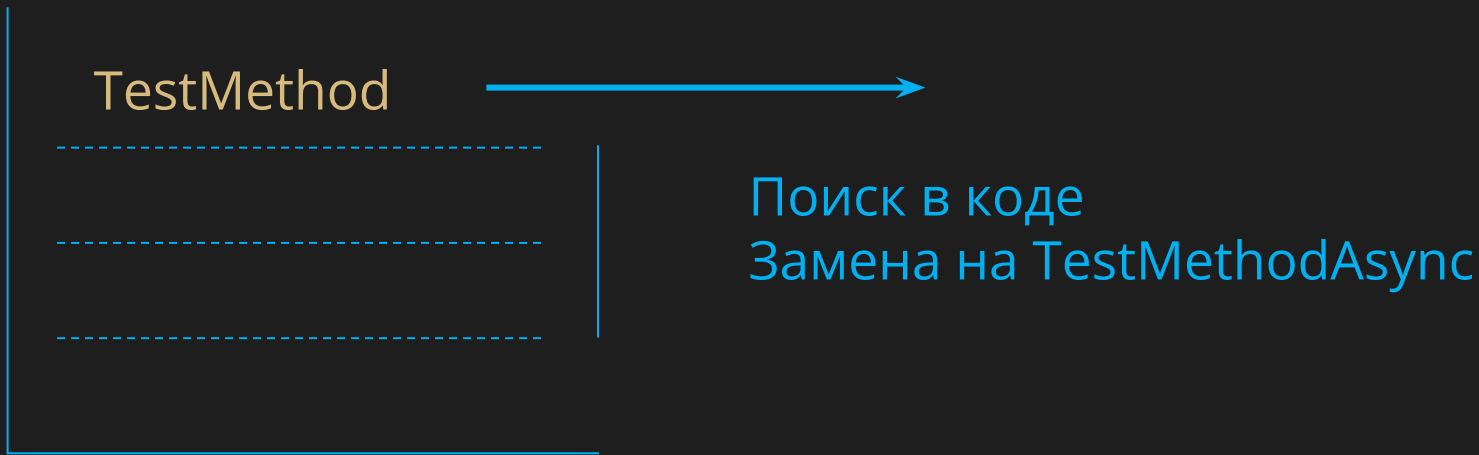
```
[TestMethod]
public async Task TestMethodAsync()
{
    await SetTimeAsync();
}
```

```
public async Task SetTimeAsync() {
    await Page.ApplyAsync();
}
```









Пример регулярки

```
(?<spaces>[\s\/]+)(?<Level>[\w]+)(?<Type>\s+[\S]+)?\s(?<Result>[\S]+)\s+(?<Method>[\w\d]+)\((?<Params>.*)\)
```

```
public void SetTime(string value)
```

Spaces:

Level : public

Result: void

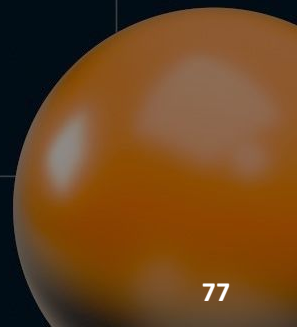
Method: SetTime

Params: string value

<https://regex101.com/>

Замена API

- найти инструмент
- учесть асинхрон
- заменить код
- исправить ошибки компиляции



Меняем код

using OpenQA.Selenium



using Microsoft.Playwright

`.Navigate().GoToUrl`



`.GotoAsync`

`.Navigate().Refresh()`



`.ReloadAsync()`


```
.FindElement(By.XPath("value"))
```



```
.Locator("xpath=value")
```

```
.FindElements(By.XPath("value"))
```



```
.Locator("xpath=value").AllAsync()
```

`.Click()`



`.ClickAsync()`

`.DoubleClick()`



`.Db1ClickAsync()`

`.Clear()`



`.ClearAsync()`

`.SendKeys()`



`.FillAsync()`

.Text



.TextContextAsync()

`.Displayed()`



`.IsVisibleAsync()`

`.Check()`



`.CheckAsync()`

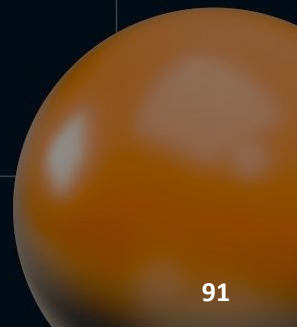
`.Uncheck()`



`.UncheckAsync()`

Замена API

- найти инструмент
- учесть асинхрон
- заменить код
- исправить ошибки компиляции

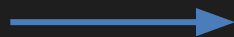




Авторизация

Команда платформы

```
class Application
```



Команда продукта

```
new Application().Login()
```

selenium.qatools.dll

playwright.qatools.dll



Fluent :(

```
[TestMethod]
```

```
public void TestMethod()
```

```
{
```

```
    Driver.Get<AppPage>()
```

```
        .SetTime("10:46:00")
```

```
        .Apply()
```

```
}
```

Fluent :(

```
public class AppPageExt
```

```
{
```

```
    public AppPage SetTime(this AppPage page, string text)
```

Fluent :)

```
public class AppPageExt
{
    public async Task<AppPage> SetTimeAsync(this Task<AppPage> page, string text)
```


Fluent :)

```
public class AppPageExt
```

```
{
```

```
    public async Task<AppPage> SetTimeAsync(this Task<AppPage> page, string text)  
        => SetTimeAsync(await page, text)
```

```
public async Task<AppPage> SetTimeAsync(this AppPage page, text)
```

Out, Ref :(

```
public class AppPageExt
{
    public Task<AppPage> GetTime(this AppPage, out string result)
    {
    }
}
```

Out, Ref :)

```
public class AppPageExt
{
    public Task<string> GetTime(this AppPage)
    {
    }
}
```

FileInput :(

.SendKeys(



.FillAsync(

.SetInputFilesAsync(

FileInput :)

.SendKeys(DeployPath(



.FillAsync(

.SetInputFilesAsync(DeployPath(

Скачивание файлов

```
.DownloadByLink(a.GetProperty("href"))
```



```
waitForDownloadTask = page.WaitForDownloadAsync();  
...  
var download = await waitForDownloadTask;  
await download.SaveAsAsync(download.SuggestedFilename);
```

Assert

```
.Driver.WaitTitle(
```



```
Driver.Expect().ToHaveTitleAsync(
```

Assert

`Locator.WaitText(`



`Locator.Expect().ToHaveTextAsync(`

Assert

Locator.WaitAttribute(



Locator.Expect().ToHaveAttributeAsync(

Assert

`FindElements().WaitCount(`



`Locator().Expect().ToHaveCountAsync(`

Как происходит правки ошибок

1 000 ошибок

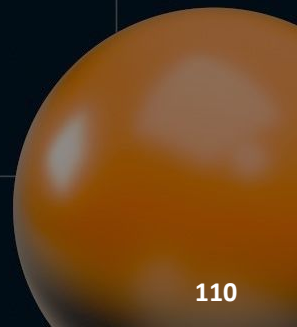


Итого

2 000 ошибок :)

Замена API

- найти инструмент
- учесть асинхрон
- заменить код
- исправить ошибки компиляции



Почему мы решили перейти на PW

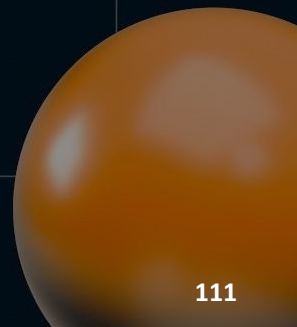
Отказ от WebDriver

Паттерн "Proxy"

Замена API

Исправление ошибок

Результаты



Как изменился тест для команды

```
[TestMethod]
public void TestMethod()
{
    Driver.Get<AppPage>()
        .SetTime("10:46:00")
        .Search()
}
```


Как изменился тест для команды

```
[TestMethod]
public async Task TestMethodAsync()
{
    await Driver.Get<AppPage>()
        .SetTimeAsync("10:46:00")
        .SearchAsync()
}
```

Что оставили

- Proxy
- Расширения для PageObject

`Driver.Get<Page>()`

`Element.CastTo<Type>()`

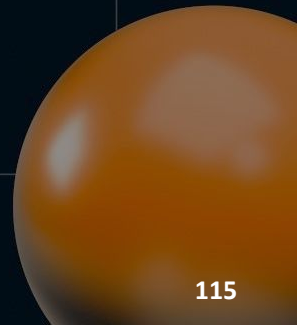
Затраты

2 дня - анализ РW

1 день - проху

3 дня - регулярки

2 дня - исправление ошибок

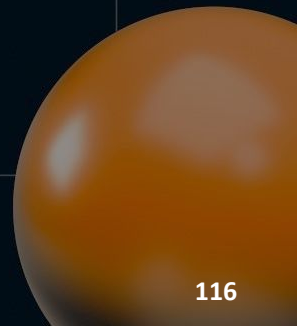


Профит

Перевели 1000 тестов, 2000 локаторов

Тесты стали быстрее на 50%*

Нет архитектурных проблем



Selenium 4



Выводы

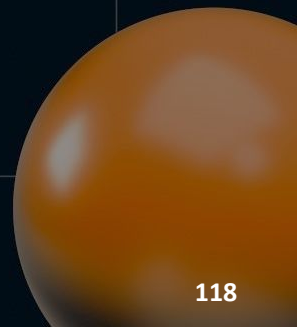
Миграция на PW не сложная

Проще всего написать регулярки

95% кода переносится автоматически

Времени на миграцию уходит немного

Будут ошибки, они решаемы



Вопросы?



Александр
Шарков

Монитор электрик



aleks.sharkov@gmail.com

HEISENBUG


МОНИТОР
ЭЛЕКТРИК

ССЫЛКИ

Playwright

<https://playwright.dev/>

Chrome for testing

<https://googlechromelabs.github.io/chrome-for-testing/>

Отладка регулярок

<https://regex101.com/>

WebDriver BiDi

<https://www.youtube.com/watch?v=6oXic6dcn9w>

PlayWrightium (Java Proxy)

<https://github.com/britka/playwrightium>

