




Реализация паттернов PageElement и PageObject в тестировании Web клиента DirectumRX

Как это сделано сейчас и как это могло бы быть



Кому подойдёт описываемый подход

- У вас приложение с интерфейсом, похожим на десктопный.
 - Есть части интерфейса, которые присутствуют на экране всегда.
 - Есть части интерфейса, которые часто повторяются на разных страницах.
 - Фронтенд вашего приложения написан на современном фреймворке.
- 



Пример интерфейса

Aypa + Create Search for documents, assignments, others Advanced Search ? Profile

Inbox 56 New In Process 30 Days Filter ...

Sent 458 ^

Favorites 458 ^

TestFolderUDm3GXJB4Y8Yfg==

TestFolderZcVs1RzQDofGyA==

TestFolderbUOLg7wM+kypZA...

TestFolderxSAOGmXSifXA2Q==

Recent Documents

Recent Assignments

Public Folders

AutoTestModule

Inbox Items: 60 Refresh Folder Print Trash Card Unread/Read New Subtask Send ...

Subject

- 🕒 LSeFq7xDUAqdUg==
- 🕒 szGg2WVoVtKZMLoetrx9
- 🕒 qKEqd+Ne+kenlfrXeOMQ
- 🕒 XjDuvqqhS53ubpXIIHbR
- 🕒 3IJxi4U7tvAjY1jITWa5
- 🕒 9s6LsgxGhdq8TZRkLLid
- 🕒 TNNmUMiHjZluaRZ8Vwij
- 🕒 xzDc/L0fvYI9By9m8vtl
- 🕒 KrKTDCWOZacQQa70lpb3
- 🕒 C/NTzyor/KCuvUzbsRcs
- 🕒 >>RTMm4I9unLoTQZ+B2z0N
- 🕒 qljbm64TBDNRNsjVmdp6

Как это сделано сейчас



Класс страницы:

```
public abstract class PageObject
```

```
{
```

```
    protected RemoteWebDriver driver;
```



Card



Unread/Read



New Subtask



Send



```
public virtual Ribbon MainRibbon {
```

```
    get {
```

```
        var r = this.driver.S(By.Id("main-ribbon"),
```

```
        TimeSpan.FromSeconds(10));
```

```
        return new Ribbon(this.driver, r);
```

```
    }
```

```
}
```

```
public virtual Ribbon PopupRibbon {
```

```
    get {
```

```
        var r = this.driver.S(By.ClassName("popup"),
```

```
        TimeSpan.FromSeconds(10));
```












```
        return new Ribbon(this.driver, r);
```

```
    }
```

```
}
```

```
}
```



-  Редактировать
-  Читать
-  Карточка 
-  Настроить доступ
-  Создать копию
-  Создать версию
-  Экспортировать
-  Отправить >
-  Сменить тип
-  Копировать ссылку



Класс ленты:

```
public class Ribbon : RemoteWebElement
{
    public Ribbon(IWebDriver driver, IWebElement webElement) : base(driver,
webElement.GetSecretElementID()) { }

    protected By ButtonLocator => By.CssSelector(".button.toggle-button");

    public virtual IEnumerable<IWebElement> Buttons => this.SS(this.ButtonLocator);

    public void ClickButton(string name)
    {
        var action = this.Button.First(a => a.Text.IsStringEqual(name));
        action.ClickElement();
    }
}
```



Недостатки:

- Поиск элементов выполняется медленно, потому что:
 - Поиск в драйвере быстрее
 - Приходится искать сначала элемент ленты, потом элемент кнопки
- Возможно `StaleElementReferenceException` сразу в нескольких местах



Альтернатива

Альтернатива, сынок – это утки!






Класс страницы:

```
public class PageObjectNew
{
    protected readonly RemoteWebDriver driver;

    protected virtual string MainRibbonSelector = "#context-ribbon";

    public virtual RibbonNew MainRibbon => new RibbonNew (this.driver,
this.MainRibbonSelector);
}
```



Класс ленты:

```
public class RibbonNew
{
    private RemoteWebDriver driver;
    private readonly string baseSelector;

    public RibbonNew(RemoteWebDriver driver, string locator) {
        this.driver = driver;
        this.baseSelector = locator;
    }

    private string ButtonSelector => baseSelector + " button.button_flat");

    public void ClickButton(string name) {
        var thisButtonSelector = this.ButtonSelector + $"[textContent='{name}']";
        this.ClickElement(3, thisButtonSelector);
    }
}
```

Класс ленты, продолжение:

```
private void ClickElement(int _maxAttempts, string selector)
{
    var count = 0;
    var maxAttempts = _maxAttempts;

    while(true) {
        try {
            var element = driver.S(By.CssSelector(selector));
            element.Click();
            break;
        }

        catch (StaleElementReferenceException) {
            if(++count == maxAttempts) throw;
        }
    }
}
```

Преимущества:

- Элементы ищутся быстро, т.к. напрямую в браузере и только один раз.
- Исключается `StaleElementReferenceException`.
- Код тестов становится проще



Вопросы?



Dmitry Abashin

Automated QA engineer

 DIRECTUM development office

Ufa city

Abashin_DV@directum.ru

