

---

# WebAii™ Automation Framework

Wojciech Pająk  
Konferencja TESTWAREZ 2008  
16-17 października 2008

[testerzy.pl](http://testerzy.pl)

# Agenda

1. WebAii - wprowadzenie
2. Metodyka automatyzacji
3. Budowa biblioteki
4. Integracja
5. Podsumowanie

## Co to jest, do czego służy

- Darmowa biblioteka .NET, .NET framework 3.5  
Autor: Art Of Test, Inc.
- Automatyzacja testów funkcjonalnych aplikacji internetowych
- Można zastosować do testów wydajnościowych (performance)
- Zawiera przykłady w C# i VB .NET

## Technologie internetowe

Biblioteka dedykowana do testów aplikacji Web 2.0

- Rich Internet Applications (RIA)
- AJAX
- ASP .NET
- Obsługa ramek
- Obsługa okien dialogowych i pop-upów
- Pełen dostęp do struktury DOM

## Przeglądarki internetowe

- Wspólne API dla Internet Explorer 6 i 7, Firefox 2 i 3
- Konieczna odpowiednia konfiguracja przeglądarek

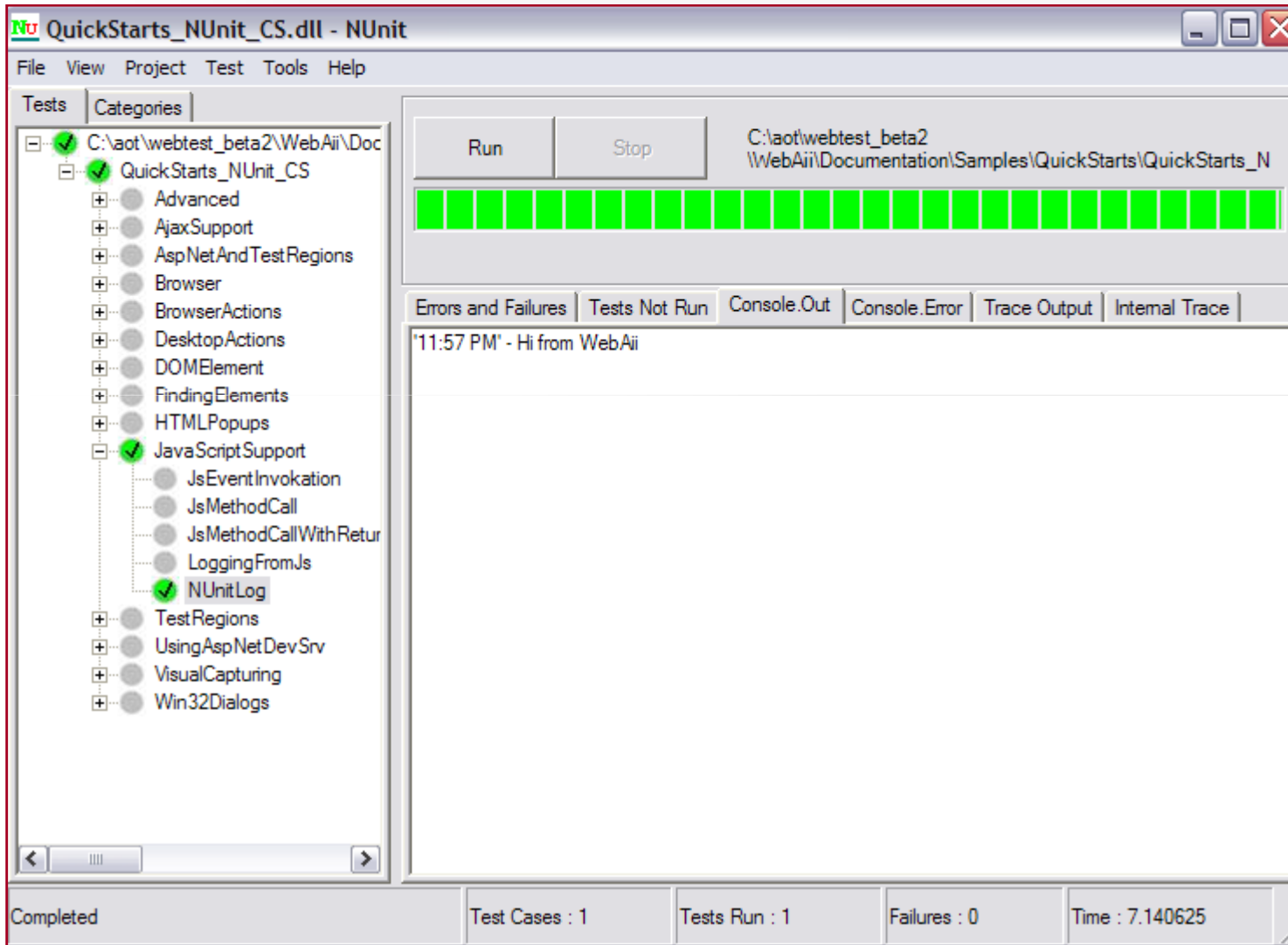


## Wymagania systemowe, instalacja, narzędzia

- NUnit lub Visual Studio Team System
- .NET framework 3.5, Visual Studio C# Express 2008
- Primary Interop Assemblies (PIA)
- Firebug, Internet Explore Developer Toolbar



# Uruchamianie testów w NUnit



## Metodyka automatyzacji

- Kontrola przeglądarki
- Automatyzacja DOM
- Automatyzacja UI  
Założenia:  
Element zawsze musi być widoczny

Desktop -> Mouse, KeyBoard, Metoda GetRectangle() dla  
każdego elementu DOM

Podejście to można także stosować do okienek Win32

Sprawdza się doskonale przy „drag & drop” - trudne do  
wykonania w DOM



## Akcje

- Klikanie elementów
- Akcje myszy na elementach
- Zaznaczanie/odznaczanie check boxów i radio buttonów
- Wywoływanie zdarzeń, np. OnClick, OnFocus
- Wykonywanie zrzutów ekranowych
- Wybieranie z elementów typu „drop down”

## Budowa biblioteki

19 namespace'ow, m.in.

- `ArtOfTest.WebAii.TestTemplates`
- `ArtOfTest.WebAii.BrowserSpecialized.Firefox`
- `ArtOfTest.WebAii.BrowserSpecialized.InternetExplorer`
- `ArtOfTest.WebAii.Core`
- `ArtOfTest.WebAii.Controls`
- `ArtOfTest.WebAii.Controls.HtmlControls`
- `ArtOfTest.WebAii.Controls.HtmlControls.HtmlAsserts`
- `ArtOfTest.WebAii.Synchronization`

## Obiekt Manager

- `Manager.ActiveBrowser.Actions`
- `Manager.ActiveBrowser.Find`
- `Manager.ActiveBrowser.DomTree`
- `Manager.ActiveBrowser.Window`
- `Manager.Desktop`
- `Manager.Log`

## Identyfikacja elementów

- Notacja skrótowa
- Wyszukiwanie po zawartości (Content, np. kiedy nie ma innych atrybutów, lub ich wartości są dynamiczne)
- Wyszukiwanie zawężone (np. po elementach określonego typu: tabela, div)
- Wyszukiwanie przy użyciu Language Integrated Query (LINQ)
- Test Fixtures dla często występujących elementów
- Serializacja/Deserializacja (identyfikacja elementów w plikach XML)

## Identyfikacja elementów, metody

- `FindById()`
- `FindByName()`
- `FindByTagIndex()`
- `FindByAttributes()`, `Find.AllByAttributes()`
- `FindByContent`, `Find.AllByContent()`
- `FindByXPath`, `Find.AllByXPath()`
- `Find.AllByTagName()`
- `FindByNodeIndexPath()`
- `FindByParam()`, `Find.AllByParam()`

## HTML Controls Suite

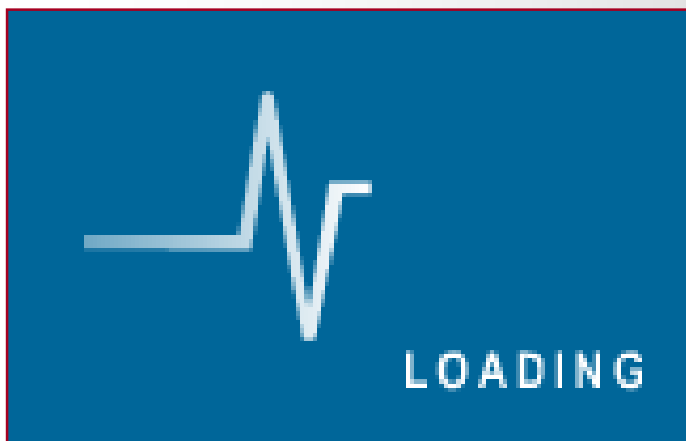
- Get/Set - właściwości DOM
- Filtrowanie elementów
- Specjalizowane Asserty (np. `Exists` - nie ma konieczności podawania `Expected/Actual`):

```
AssertAttribute()  
AssertStyle()  
AssertContent()  
AssertSelect()  
AssertTable()
```

## Klasa HtmlWait

Oczekiwanie na:

- stan kontrolki HTML (visible/not visible)
- zmianę stylu
- spełnienie lub nie dowolnego warunku



# Test Regions

- Logiczna segmentacja
- Niższe koszty utrzymania testów
- Wyizolowanie testów

```
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <title>Basic TestRegions Usage</title>
</head>
<body>
  <table><tr><td>MyCompany, Inc. HomePage.</td></tr>
</table>
  <div>
    <!--testregion id="userinput"-->
      <input type="text" id="Text1" />
      <input type="text" id="Text2" />
    <!--/testregion-->
  </div>
</body>
</html>
```

```
TestRegion userInput =
Manager.ActiveBrowser.Regions["userinput"];
TestRegion restaurantData =
Manager.ActiveBrowser.Regions["restaurantdata"];
```



## ASP .NET

- In-Proc ASP .NET
- ASP .NET Local Server
- Container Search (Asp.Net calendar, grid lub treeview – wyszukiwanie childów w rozbudowanych elementach)
- Wyszukiwanie po częściowym ID (dla automatycznie generowanych ID)

## Testowanie AJAX/JavaScript

- Synchronizacja
- Wywoływanie JavaScript
- Obsługa „drag & drop”
- Prawdziwa interakcja z interfejsem graficznym aplikacji (mysz, klawiatura)

# Integracja

- Konfiguracja .NET
- Logowanie
- VSTS unit testing framework, gotowe template'y
- NUnit, template w VS
- Biblioteka utworzona w .NET - możliwa integracja z dowolną aplikacją .NET

Dziękuję za uwagę

Więcej informacji na:

<http://www.artoftest.com/webaiifxproduct.aspx>

Pytania?

Chętnie odpowiem na Państwa zapytania drogą elektroniczną:  
[wojciech.pajak@testerzy.pl](mailto:wojciech.pajak@testerzy.pl)