

How to Test the Test Automation Framework?

Anton Angelov

Co-founder, Chief Development
AUTOMATE THE PLANET

Site: <http://bellatrix.solutions>

Blog: <http://automatetheplanet.com>

@angelovstanton

aangelov@bellatrix.solutions



Agenda

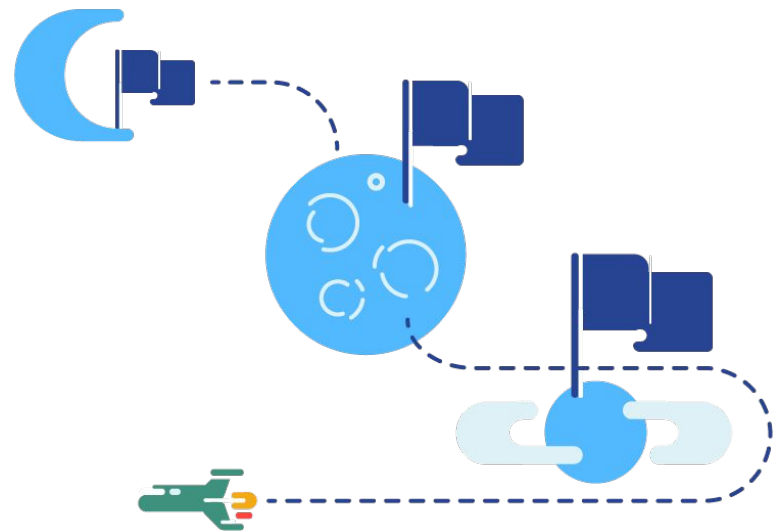
Definitions

Test Infrastructure

Branching Strategy

Release Automation

Types of Tests, Problems,
Solutions



Software Library

A software library generally consists of pre-written code, classes, procedures, scripts, configuration data and more. All of the available functions within a software library can just be called/used within the program body without defining them explicitly.

```
var currentTime = DateTime.Now.ToUniversalTime();  
var timeTomorrow = DateTime.Now.AddDays(1);
```

Software Framework

A software framework is an abstraction in which common code providing generic functionality can be selectively overridden or specialized by user code providing specific functionality.

- Inversion of control
- Default behaviour
- Extensibility
- Non-modifiable

Test Automation Framework

Various guidelines, coding standards, concepts, processes, practices, project hierarchies, modularity, etc. to support automated testing.

The Bellatrix Story

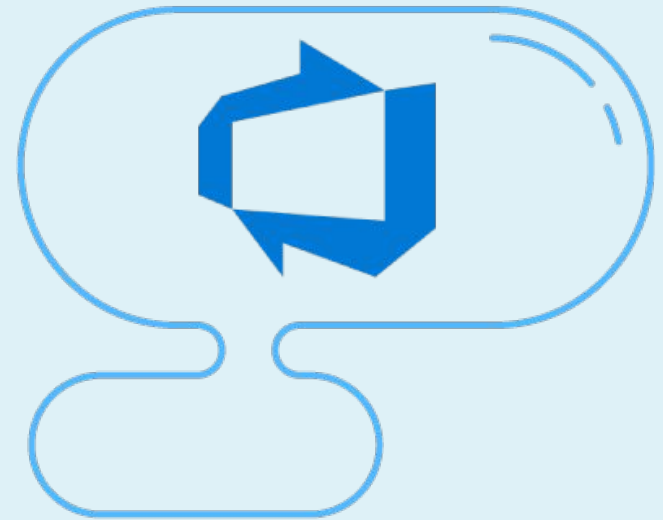
Bellatrix is a cross-platform, an easily customizable and extendable .NET test automation framework that increases tests' reliability.



Test Infrastructure



Jenkins






Test Infrastructure



Test Infrastructure

Run summary **Test results** Filter


 Create bug
  Update analysis

Outcome	Test Case Title	Priority	Duration	Owner	Configuration	Machine name	Error message
Failed	CaptureTrafficTests		0:00:07.983		None	DXU684	Test method Bellatrix.Web.GettingStarted.CaptureHttpTrafficTests.Captur...
Passed	PromotionsPageOpened_When_PromotionsButtonClicked		0:00:04.784		None	DXU684	
Passed	AddNewCookie		0:00:04.050		None	DXU684	
Passed	ControlBrowser		0:00:07.634		None	DXU684	
Passed	BlogPageOpened_When_PromotionsButtonClicked		0:00:03.470		None	DXU684	
Failed	PurchaseRocketWithGloballyOverridenMethods		0:00:37.090		None	DXU684	Test method Bellatrix.Web.GettingStarted.CommonServicesHooksTests.P...
Failed	PurchaseRocketWithGloballyOverridenMethods		0:00:26.563		None	DXU684	Test method Bellatrix.Web.GettingStarted.OverrideGloballyElementActio...
Passed	GetAllCookies		0:00:03.894		None	DXU684	
Failed	PurchaseRocketWithGloballyOverridenMethods		0:00:27.340		None	DXU684	Test method Bellatrix.Web.GettingStarted.OverrideLocallyElementAction...
Passed	PromotionsPageOpened_When_PromotionsButtonClicked		0:00:03.920		None	DXU684	
Failed	RedirectRequestsTest		0:00:01.573		None	DXU684	Test method Bellatrix.Web.GettingStarted.CaptureHttpTrafficTests.Redire...
Passed	OpenSalesPage_When_LocatedSaleButtonInsideProductImage		0:00:06.114		None	DXU684	
Passed	SortDropDownsAboveOfProtonRocketAnchor_GreaterThan40px		0:00:00.433		None	DXU684	
Passed	AssertCartPageFields		0:00:36.763		None	DXU684	
Passed	BlogPageOpened_When_PromotionsButtonClicked		0:00:00.593		None	DXU684	
Passed	BlogPageOpened_When_PromotionsButtonClicked		0:00:04.250		None	DXU684	
Failed	PurchaseRocketWithoutPageObjects		0:00:26.600		None	DXU684	Test method Bellatrix.Web.GettingStarted.PageObjectsTests.PurchaseRoc...
Passed	GetTabTitle		0:00:03.413		None	DXU684	
Passed	PromotionsPageOpened_When_PromotionsButtonClicked		0:00:01.107		None	DXU684	
Passed	BlogPageOpened_When_PromotionsButtonClicked		0:00:01.417		None	DXU684	
Passed	SortDropDownsNearTopOfProtonRocketAnchor_GreaterThan40px		0:00:00.430		None	DXU684	
Passed	SortDropDownsAboveOfProtonRocketAnchor_41px		0:00:00.440		None	DXU684	
Passed	PromotionsPageOpened_When_PromotionsButtonClicked		0:00:07.314	Anton	None	DXU684	
Passed	PromotionsPageOpened_When_PromotionsButtonClicked		0:00:04.866		None	DXU684	
Passed	HappyBirthdayCouponDisplayed_When_ClickOnCouponButton		0:00:04.870		None	DXU684	
Passed	GetSpecificCookie		0:00:04.546		None	DXU684	
Passed	AssertEnsureCartPageFields		0:00:37.433		None	DXU684	

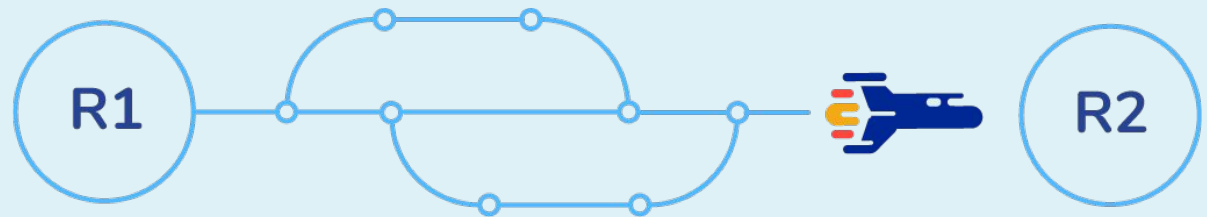
Test Infrastructure



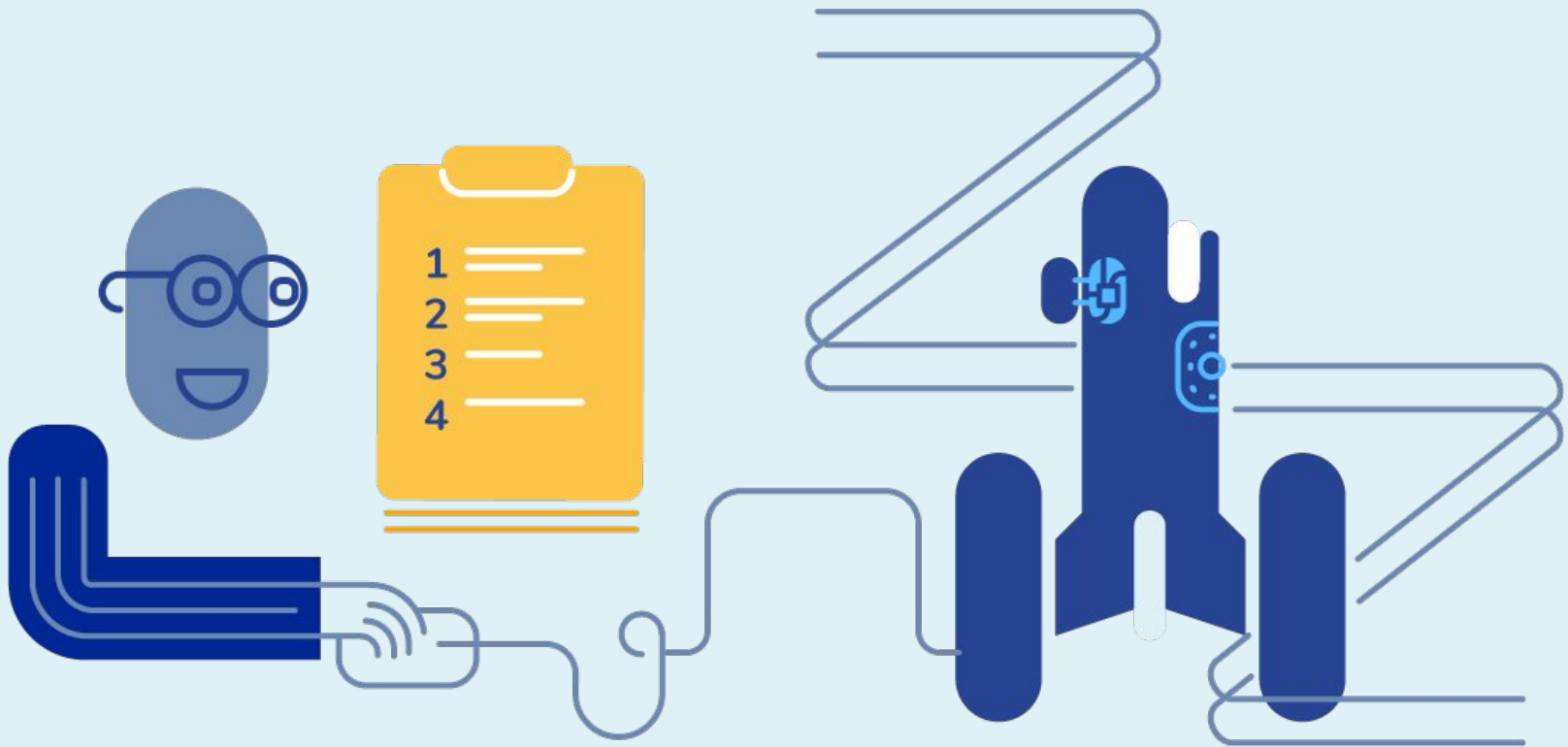
Mac OS X

Branching Strategy

- RELEASE
- LOAD_TESTING_RELEASE_5
- MAINTENANCE
- ENTERPRISE_EDITION
- INFRASTRUCTURE_SHARE



Release Automation



Release Automation

Bellatrix Release Manager 0.1

NuGet Templates Version: 1.4.3.0 [Update All]

Should Publish

Clear Logs Copy Logs Clear NuGet Folder

NuGet Cache User VS 2017 Local NuGet Cache

Build All Projects [Run]

Publish BELLATRIX Tools [Run]

Generate NuSpec NuGets [Run]

Copy NuGets Local [Run]

Publish NuGets [Run]

Publish Test NuGets [Run]

Publish Project Templates [Run]

Publish .NETCORE Templates [Run]

Build All Meissa Projects [Run]

Publish Meissa Installers [Run]

Generate Meissa NuGets [Run]

```
a package should not have a prerelease dependency. Either modify the version spec of dependency "Microsoft.Extensions.Logging [3.0.0-preview.19074.2, )" or update the version field in the nuspec. [D:\SourceCode\BELLATRIX\Framework\Bellatrix.Logging\Bellatrix.Logging.csproj]

C:\Program Files\dotnet\sdk\2.2.104\Sdks\NuGet.Build.Tasks.Pack\build\NuGet.Build.Tasks.Pack.targets(202,5): warning NU5125: The 'licenseUri' element will be deprecated. Consider using the 'license' element instead. [D:\SourceCode\BELLATRIX\Framework\Bellatrix.Logging\Bellatrix.Logging.csproj]

Babel Obfuscator version 9.1.0.0

Copyright (C) 2010-2019 babelfor.NET. All rights reserved.

Processing Bellatrix.TestExecutionExtensions.Common.dll...

Analyzing...

Target framework: .NETStandard,Version=v2.0

Running Agent...

Control flow obfuscation...

Obfuscating names...

Encrypting strings...

Writing D:\SourceCode\BELLATRIX\Framework\Bellatrix.TestExecutionExtensions.Common\bin\ObfuscatedNuGets\netstandard2.0\Bellatrix.TestExecutionExtensions.Common.dll
```

Types of Tests, Problems, Solutions



Functional Testing

```
[TestMethod]
[TestCategory(Categories.Chrome)]
public void DateSet_When_UseSetDateMethodWithDayBiggerThan9_Chrome()
{
    var dateElement = App.ElementCreateService.CreateById<Date>("myDate");

    dateElement.SetDate(2017, 11, 15);

    Assert.AreEqual("2017-11-15", dateElement.GetDate());
}
```

Functional Testing

[TestMethod]

```
public void DateSetThrowsArgumentException_When_Month0_Edge()
```

```
{
```

```
    var dateElement = App.ElementCreateService.CreateById<Date>("myDate");
```

```
    Assert.ThrowsException<ArgumentException>(() => dateElement.SetDate(2017, 0, 1));
```

```
}
```

Functional Testing

```
[TestMethod]
public void SettingDateCalled_BeforeActuallySetDate()
{
    Date.SettingDate += AssertValueAttributeEmpty;

    var dateElement = App.ElementCreateService.CreateById<Date>("myDate");

    dateElement.SetDate(2017, 7, 6);

    Assert.AreEqual("2017-07-06", dateElement.GetDate());

    Date.SettingDate -= AssertValueAttributeEmpty;

    void AssertValueAttributeEmpty(object sender, ElementActionEventArgs args)
    {
        Assert.AreEqual(string.Empty, args.Element.WrappedElement.GetAttribute("value"));
    }
}
```

Functional Testing

```
[TestMethod]
public void LocallyOverriddenActionCalled_When_OverrideGetDateGloballyNotNull()
{
    Date.OverrideGetDateGlobally = (u) => "2017-06-01";
    var dateElement = App.ElementCreateService.CreateById<Date>("myDate");

    Date.OverrideGetDateLocally = (u) => "2017-07-01";

    Assert.AreEqual("2017-07-01", dateElement.GetDate());

    Date.OverrideGetDateGlobally = null;
}
```

Functional Testing

```
private Mock<IBellaLogger> _mockedLogger;

public override void TestInit()
{
    _mockedLogger = new Mock<IBellaLogger>();
    App.RegisterInstance(_mockedLogger.Object);
}

[TestMethod]
public void CreatesBDDLog_When_ClickButton()
{
    var button = App.ElementCreateService.CreateByIdContaining<Button>("button");

    button.Click();

    _mockedLogger.Verify(x => x.LogInformation(
It.Is<string>(t => t.Contains("Click control (ID = button)"))), Times.Once());
}
```

Functional Testing

```
private Mock<IBellaLogger> _mockedLogger;
```

```
public override void TestInit()
```

```
{  
    _mockedLogger = new Mock<IBellaLogger>();  
    App.RegisterInstance(_mockedLogger.Object);  
}
```

```
[TestMethod]
```

```
public void CreatesBDDLog_When_ClickButton()
```

```
{  
    var button = App.ElementCreateService.CreateByIdContaining<Button>("button");  
  
    button.Click();  
  
    _mockedLogger.Verify(x => x.LogInformation(It.Is<string>(t => t.Contains("Click control (ID =  
button)"))), Times.Once());  
}
```

Learning Tests

```
[TestMethod]
public void OrientationTest()
{
    IRotatable rotatable = ((IRotatable)_driver);
    rotatable.Orientation = ScreenOrientation.Portrait;
    Assert.AreEqual(ScreenOrientation.Portrait, rotatable.Orientation);
}
```

```
[TestMethod]
public void LockTest()
{
    _driver.Lock();
    Assert.AreEqual(true, _driver.IsLocked());
    _driver.Unlock();
    Assert.AreEqual(false, _driver.IsLocked());
}
```

Features Not to Automate

```
[VideoRecording(VideoRecordingMode.OnlyFail)]  
[Android(Constants.AndroidNativeAppPath,  
  Constants.AndroidDefaultAndroidVersion,  
  Constants.AndroidDefaultDeviceName,  
  Constants.AndroidNativeAppAppExamplePackage,  
  ".view.Controls1",  
  AppBehavior.ReuseIfStarted)]  
public class VideoRecordingTests : AndroidTest
```

Features Not to Automate

The screenshot shows a Windows-style dialog box titled "Bellatrix Test Automation Framework Setup". The main heading is "Custom Setup" with a sub-instruction: "Select the way you want features to be installed." Below this, a tree view lists features with expandable icons. The "Bellatrix Web" feature is selected and highlighted in blue. To its right, a text box provides details: "This feature requires 0KB on your hard drive. It has 2 of 2 subfeatures selected. The subfeatures require 0KB on your hard drive." At the bottom right of the dialog is a "Browse..." button. The footer contains five buttons: "Reset", "Disk Usage", "Back", "Next" (which is highlighted with a blue border), and "Cancel".

Bellatrix Test Automation Framework Setup

Custom Setup

Select the way you want features to be installed.

Click the icons in the tree below to change the way features will be installed.

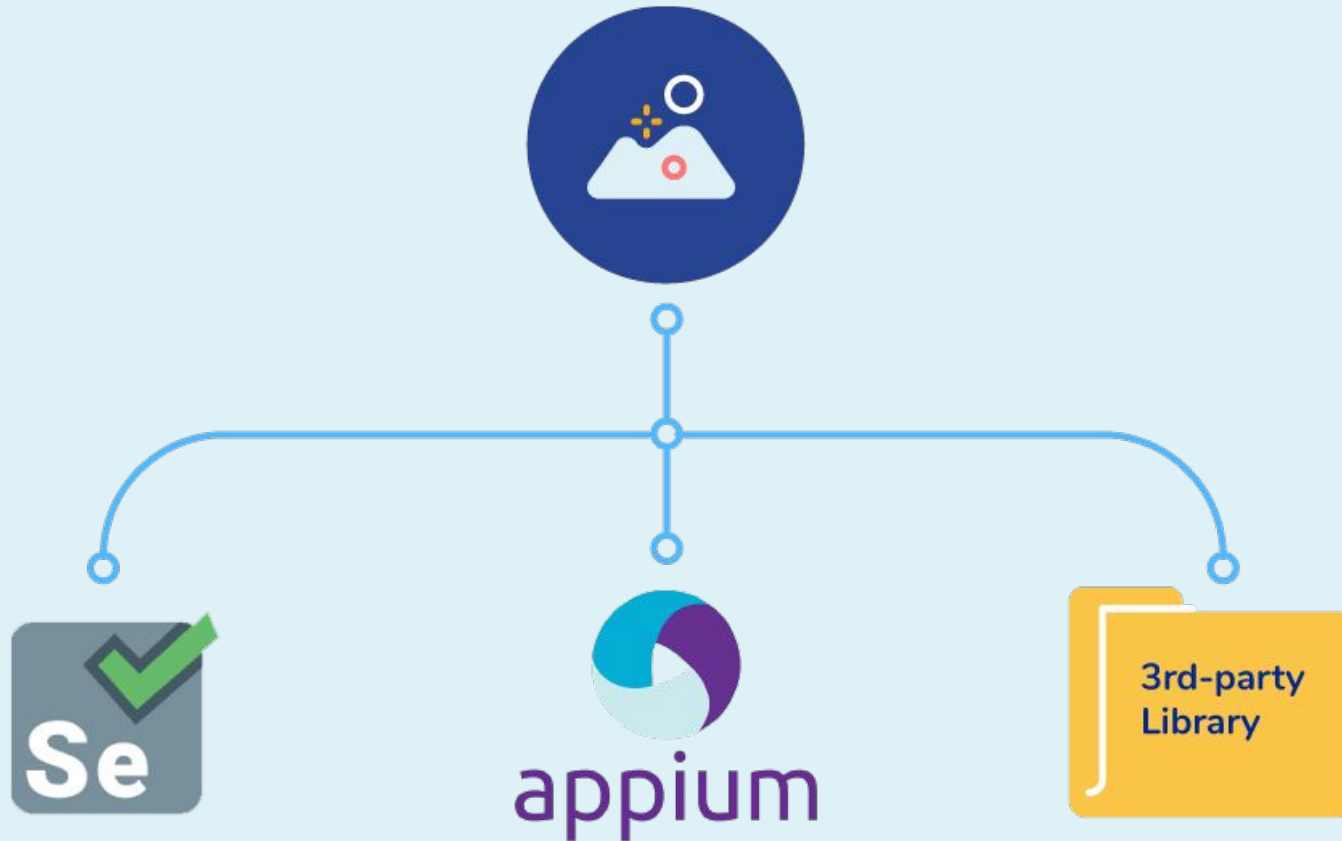
- Bellatrix Web
- Bellatrix Desktop
- Bellatrix Android
 - Item Templates
 - Snippets
- Bellatrix iOS
 - Item Templates
 - Snippets

This feature requires 0KB on your hard drive. It has 2 of 2 subfeatures selected. The subfeatures require 0KB on your hard drive.

Browse...

Reset Disk Usage Back **Next** Cancel

Upgradability Testing



Features Not to Automate

```
public interface ITimeouts
{
    TimeSpan ImplicitWait { get; set; }

    [Obsolete("This method will be removed in a future version. Please set the ImplicitWait property instead.")]
    ITimeouts ImplicitlyWait(TimeSpan timeToWait);
}
```

Installability Testing

```
dotnet new -i Bellatrix.Web.GettingStarted
```

```
dotnet new Bellatrix.Web.GettingStarted
```

Installability Testing

▼ Templates

- 👑 BELLATRIX Android MSTest Template Windows TEST
- 👑 BELLATRIX Android NUnit Template Windows TEST
- 👑 BELLATRIX API MSTest Template Windows TEST
- 👑 BELLATRIX API NUnit Template Windows TEST
- 👑 BELLATRIX Desktop MSTest Template Windows TEST
- 👑 BELLATRIX Desktop NUnit Template Windows TEST
- 👑 BELLATRIX iOS MSTest Template Windows TEST
- 👑 BELLATRIX iOS NUnit Template Windows TEST
- 👑 BELLATRIX Web MSTest Template Windows TEST
- 👑 BELLATRIX Web NUnit Template Windows TEST

Portability Testing

“The ease with which the software product can be transferred from one hardware or software environment to another”

Portability Testing

```
[TestClass]
[Android(Constants.AndroidNativeAppPath,
  Constants.AndroidDefaultAndroidVersion,
  Constants.AndroidGalaxyDeviceName,
  Constants.AndroidNativeAppAppExamplePackage,
  ".view.Controls1",
  AppBehavior.ReuseIfStarted)]
public class ButtonControlTests : AndroidTest
```

```
[TestClass]
[Android(Constants.AndroidNativeAppPath,
  Constants.AndroidDefaultAndroidVersion,
  Constants.AndroidNokiaDeviceName,
  Constants.AndroidNativeAppAppExamplePackage,
  ".view.Controls1",
  AppBehavior.ReuseIfStarted)]
public class ButtonControlTests : AndroidTest
```

Interoperability Testing

“The degree to which two or more components or systems can exchange information and use the information that has been exchanged.”

Interoperability Testing

New Project

Recent

Installed

Visual C#

- Get Started
- Windows Universal
- Windows Desktop
- Web
 - .NET Core
 - .NET Standard
 - Android
 - Apple TV
 - Apple Watch
 - Cloud
 - Cross-Platform
 - Extensibility
 - iOS Extensions
 - iPhone & iPad
 - Other
 - Test
 - WCF

Not finding what you are looking for?
[Open Visual Studio Installer](#)

Sort by: Default

Project Name	Type
Azure WebJob (.NET Framework)	Visual C#
Azure Cloud Service	Visual C#
Bellatrix.API.GettingStarted	Visual C#
Bellatrix.API.MSTest.Tests	Visual C#
Bellatrix.API.NUnit.Tests	Visual C#
Bellatrix.Desktop.GettingStarted	Visual C#
Bellatrix.Desktop.MSTest.Tests	Visual C#
Bellatrix.Desktop.NUnit.Tests	Visual C#
Bellatrix.Web.GettingStarted	Visual C#
Bellatrix.Web.MSTest.Tests	Visual C#
Bellatrix.Web.NUnit.Tests	Visual C#

Search (Ctrl+E)

Type: Visual C#

Examples how to do web testing with Bellatrix

Name: Bellatrix.Web.GettingStarted

Location: C:\Users\AutomateThePlanet\source\repos

Solution: Create new solution

Solution name: Bellatrix.Web.GettingStarted

Create directory for solution

Create new Git repository

OK Cancel

Performance Testing

```
[TestClass]
[IOS(Constants.IOSNativeAppPath,
    Constants.IOSDefaultVersion,
    Constants.IOSDefaultDeviceName,
    AppBehavior.RestartEveryTime)]
public class ButtonControlTests : IOSTest
{
    [TestMethod]
    [Timeout(30000)]
    public void ZeroReturnForButtonText_When_CallClickMethod()
    {
        var button = App.ElementCreateService.CreateByName<Button>("ComputeSumButton");

        button.Click();

        var answerLabel = App.ElementCreateService.CreateByName<Label>("Answer");

        answerLabel.EnsureTextIs("0");
    }
}
```

Security Testing

The screenshot displays the Visual Studio IDE with a C# project named 'Bellatrix.Desktop' (NETCore 2.1) on Any CPU. The left sidebar shows the 'References' and 'Solution Explorer' for the project. The main editor shows the 'Element' class with the following code:

```
400:     Action<T>, T, T> action2 = globalAction;
409:     Action<T>, T, T> action3 = defaultAction;
410:     if ((object)action1 == (object)null)
411:     {
412:         bool flag = (object)action2 != (object)null;
413:         action = (!flag ? new Action<TSecond, TThird>((T argument0, T argument1) => action3(_ue000, argument0, argument1)) : new Action<TSecond, TThird>((T argument2, T argument3) => action2(_ue000, argument2, argument3)));
414:     }
415:     else
416:     {
417:         action = (T argument4, T argument5) => action1(_ue000, argument4, argument5);
418:     }
419:     return action;
420: }
421:
422: protected Func<TReturn> InitializeAction<TElement, TReturn>(TElement element, Func<TElement, TReturn> globalAction, Func<TElement, TReturn> localAction, Func<TElement, TReturn> defaultAction)
423: where TElement : Element
424: {
425:     Func<TReturn> func;
426:     Func<T, T> func1 = localAction;
427:     T _ue000 = element;
428:     Func<T, T> func2 = globalAction;
429:     Func<T, T> func3 = defaultAction;
430:     if ((object)func1 == (object)null)
431:     {
432:         bool flag = (object)func2 != (object)null;
433:         func = (!flag ? new Func<TReturn>(() => func3(_ue000)) : new Func<TReturn>(() => func2(_ue000)));
434:     }
435:     else
436:     {
437:         func = () => func1(_ue000);
438:     }
439:     return func;
440: }
441:
442: protected Action<TSecond> InitializeAction<TElement, TSecond>(TElement element, Action<TElement, TSecond> globalAction, Action<TElement, TSecond> localAction, Action<TElement, TSecond> defaultAction)
443: where TElement : Element
444: {
445:     Action<TSecond> action;
446:     Action<T, T> action1 = localAction;
447:     T _ue000 = element;
448:     Action<T, T> action2 = globalAction;
449:     Action<T, T> action3 = defaultAction;
450:     if ((object)action1 == (object)null)
451:     {
452:         bool flag = (object)action2 != (object)null;
453:         action = (!flag ? new Action<TSecond>((T argument0) => action3(_ue000, argument0)) : new Action<TSecond>((T argument1) => action2(_ue000, argument1)));
454:     }
455:     else
456:     {
457:         action = (T argument2) => action1(_ue000, argument2);
458:     }
459:     return action;
460: }
461:
462: protected Action<TSecond, TThird, TFourth> InitializeAction<TElement, TSecond, TThird, TFourth>(TElement element, Action<TElement, TSecond, TThird, TFourth> globalAction, Action<TElement, TSecond, TThird, TFourth> localAction, Action<TElement, TSecond, TThird, TFourth> defaultAction)
463: where TElement : Element
464: {
465:     Action<TSecond, TThird, TFourth> action;
466:     Action<T, T, T> action1 = localAction;
467:     T _ue000 = element;
468:     Action<T, T, T> action2 = globalAction;
469:     Action<T, T, T> action3 = defaultAction;
470:     if ((object)action1 == (object)null)
471:     {
472:         bool flag = (object)action2 != (object)null;
473:         action = (!flag ? new Action<TSecond, TThird, TFourth>((T argument0, T argument1, T argument2) => action3(_ue000, argument0, argument1, argument2)) : new Action<TSecond, TThird, TFourth>((T argument0, T argument1, T argument2) => action2(_ue000, argument0, argument1, argument2)));
474:     }
475:     else
476:     {
477:         action = (T argument3, T argument4, T argument5) => action1(_ue000, argument3, argument4, argument5);
478:     }
479:     return action;
480: }
```

The bottom of the IDE shows the 'Assembly Name' as 'Bellatrix.Desktop, Version=1.3.0.0', 'Namespace' as 'Bellatrix.Desktop', and 'Type Name' as 'Element'.

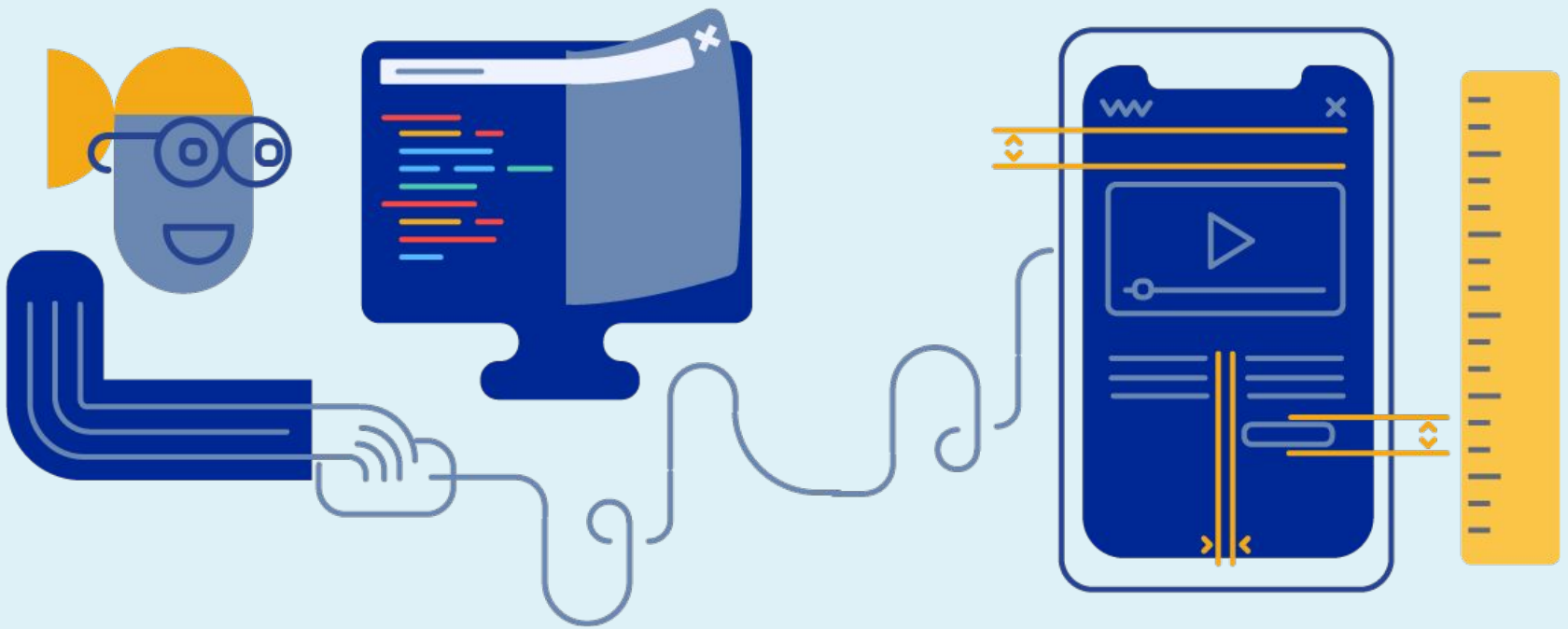
Usability, Understandability, Learnability Testing



Suitability Testing

“The capability of the software product to provide an appropriate set of functions for specified tasks and user objectives.”

Syntax Testing



Syntax Testing

The image shows a Visual Studio IDE with a C# code file named `BellatrixBrowserBehaviourTests.cs` open. The code is a test class that inherits from `WebTest` and contains two test methods. The first method, `PromotionsPageOpened_When_PromotionsButtonClicked()`, is marked with `[TestMethod]` and uses `App.NavigationService.Navigate` to go to a specific URL. The second method, `BlogPageOpened_When_PromotionsButtonClicked()`, is also marked with `[TestMethod]` and uses `App.NavigationService.Navigate` to go to a different URL. The code is annotated with comments explaining the behavior and the use of the `Browser` attribute.

```
22 // If you place attribute over the class all tests inherit the behaviour. It is possible to place it
23 [Browser(BrowserType.Firefox, BrowserBehavior.ReuseIfStarted)]
24
25 // 2.2. All web Bellatrix test classes should inherit from the WebTest base class. This way you can
26 public class BellatrixBrowserBehaviourTests : WebTest
27 {
28     // 2.3. All MSTest tests should be marked with the TestMethod attribute.
29     [TestMethod]
30     public void PromotionsPageOpened_When_PromotionsButtonClicked()
31     {
32         // There is more about the App class in the next sections. However, it is the primary point
33         // It comes from the WebTest class as a property. Here we use the Bellatrix navigation service
34         App.NavigationService.Navigate("http://demos.bellatrix.solutions/");
35
36         // Use the element creation service to create an instance of the anchor. There are much more
37         var promotionsLink = App.ElementCreateService.CreateByLinkText<Anchor>("Promotions");
38
39         promotionsLink.Click();
40     }
41
42     [TestMethod]
43
44     // 2.4. As mentioned above you can override the browser behaviour for a particular test. The glo
45     // Only for this particular test, Bellatrix opens Chrome and restarts it only on fail.
46     [Browser(BrowserType.Chrome, BrowserBehavior.RestartOnFail)]
47     public void BlogPageOpened_When_PromotionsButtonClicked()
48     {
49         App.NavigationService.Navigate("http://demos.bellatrix.solutions/");
```

The Solution Explorer on the right shows the project structure. The current project is `Bellatrix.Web.GettingStarted`, and the file `BellatrixBrowserBehaviourTests.cs` is selected under the `02. Control Browser` folder.

Syntax Testing

Type your title

For each test technology, we created a similar starter kit for Bellatrix. Each of them explains the features of the framework with detailed real-world examples. Moreover, `contains` detailed comments for each part. Something that makes it ideal for self-learning is that after each chapter it offers exercises that people can do themselves.

The screenshot displays the Grammarly web application interface. The browser address bar shows the URL `https://app.grammarly.com/ddocs/478116958`. The main content area features a section titled "ALL ALERTS" with a single alert card. The alert card has a green header with the text "it contains" and a body of text that reads: "This sentence appears to be missing a pronoun. Consider adding the pronoun." To the right of the alert card are icons for deleting, flagging, and a menu. On the right side of the interface, there is a sidebar with the following elements: "HIDE ASSISTANT" with a green 'G' icon, "PERFORMANCE" with a score of 97, "SET GOALS" with a target icon, "ALL ALERTS" with a red notification badge showing '1', and "GRAMMAR" with a notification badge showing '1'.

Static Analysis

The screenshot displays the Visual Studio IDE with a C# code file named `SampleRulesCode.cs` open. The code contains several variables, including `secondPlanetSize` on line 147, which is highlighted with a red box. Below the code editor, the Error List window is visible, showing a table of errors. The first error, CS0219, is highlighted with a red box and corresponds to the `secondPlanetSize` variable. The error message is "The variable 'secondPlanetSize' is assigned but its value is never used".

```
131     this.FuelType = fuelType;
132
133     // 13. Style var Elsewhere
134
135     // csharp_style_var_elsewhere = true:error
136     int moonSize = CalculateMoonSize();
137
138     // csharp_style_var_elsewhere = false:error
139     var secondMoonSize = CalculateMoonSize();
140
141     // 14. Style var for Built in Types
142
143     // csharp_style_var_for_built_in_types = true:error
144     var planetSize = 42;
145
146     // csharp_style_var_for_built_in_types = false:error
147     int secondPlanetSize = 42;
148
149     // 15. Style var When Type Is Apparent
150
151     // csharp_style_var_when_type_is_apparent = true:error
152     var robots = new List<int>();
153
154     // csharp_style_var_when_type_is_apparent = false:error
155     //List<int> robots = new List<int>();
156
157     // 17. Style Coalesce Expression
158
159     // dotnet_style_coalesce_expression = true:error
160     int? sunSize = null;
161     var blackHole = sunSize ?? 10000;
162
163     // dotnet_style_coalesce_expression = false:error
164     var secondBlackHole = sunSize != null ? sunSize : 10000;
165
```

Code	Description	Project	File	Line	Suppression St...
CS0219	The variable 'secondPlanetSize' is assigned but its value is never used	CodingStandardsEditorConfig	SampleRulesCode.cs	147	Active
CS0219	The variable 'moonSize' is assigned but its value is never used	CodingStandardsEditorConfig	SampleRulesCode.cs	136	Active
CS0219	The variable 'secondMoonSize' is assigned but its value is never used	CodingStandardsEditorConfig	SampleRulesCode.cs	139	Active
CS0219	The variable 'planetSize' is assigned but its value is never used	CodingStandardsEditorConfig	SampleRulesCode.cs	144	Active
CS0219	The variable 'robots' is assigned but its value is never used	CodingStandardsEditorConfig	SampleRulesCode.cs	152	Active
CS0219	The variable 'sunSize' is assigned but its value is never used	CodingStandardsEditorConfig	SampleRulesCode.cs	160	Active
CS0219	The variable 'blackHole' is assigned but its value is never used	CodingStandardsEditorConfig	SampleRulesCode.cs	161	Active
CS0219	The variable 'secondBlackHole' is assigned but its value is never used	CodingStandardsEditorConfig	SampleRulesCode.cs	164	Active

Summary

- Definitions
- Test Infrastructure
- Branching Strategy
- Release Automation
- Types of Tests, Problems, Solutions

Resources

- <http://docs.bellatrix.solutions>
- <https://www.automatetheplanet.com/>
- MSDN

Questions

