



Ssauce
LABS

Saucery For SauceLabs

Making Automated Testing Simple

Saucery operates within a context

- Jenkins or Bamboo
- Sauce OnDemand Plugin
 - Saucery starts at plugin-set environment variable
 - Extending Saucery to support other CI ecosystems like CircleCI would be simple.

Different Implementations

.NET

- Saucery2 for NUnit 2.6.4 (released on [NuGet](#))
- Saucery3 for NUnit 3.2 (released on [NuGet](#))

Java (Maven 3)

- Saucery for JUnit 4.12 (releasing soon)



Major Benefits

- Desired Capabilities configured transparently
 - Virtually transparent to developer written tests
- Flow control capability baked in
 - Respect for parallel limits ensures tests are submitted smoothly
- Videos only recorded for failed tests



Rather than...

```
namespace SauceLabs.SeleniumExamples {  
    [TestFixture]  
    //Without Saucery, you need to write these magic strings at compile time  
    [Header("browser", "version", "platform")]  
    [Row("internet explorer", "11", "Windows 7")]  
    [Row("chrome", "35", "linux")]  
    [Row("safari", "6", "OS X 10.8")]  
    public class GuineaPigTests {  
        [Setup]  
        private IWebDriver _Setup(string browser, string version, string platform) {  
            //Without Saucery you have to create DesiredCapabilities and initialise the driver  
        }  
  
        [Teardown]  
        public void CleanUp(IWebDriver _Driver) {  
            //Without Saucery you have to tell SauceLabs the result of the test  
        }  
  
        [Test]  
        [Parallelizable] // Without Saucery you have specify this!  
        public void PageTitle(string browser, string version, string platform) {  
            //Write your selenium test  
        }  
    }  
}
```



do...

```
namespace MyTester.Tests {  
    public class MyTests : SauceryBase {  
        public MyTests(SaucePlatform platform) : base(platform) {  
        }  
  
        [Test]  
        [TestCase(5)]  
        [TestCase(4)]  
        public void MyDataDrivenTest(int data) {  
            // Write my test here  
        }  
  
        [Test]  
        public void MySecondTest() {  
            // Write my test here  
        }  
    }  
}
```

Console Benefits

—Human-readable output for each test

- “Setting up <TestName>_<OS>_<Browser>_<BrowserVersion> for Desktop on WebDriver”
- “Setting up <TestName>_<OS>_<Browser>_<BrowserVersion>_<Orientation> for IOS on Appium”
- “Setting up <TestName>_<OS>_<Browser>_<BrowserVersion>_<Orientation> for IOS on WebDriver”
- “Setting up <TestName>_<OS>_<Browser>_<BrowserVersion>_<LongName>_<Orientation> for Android on WebDriver”
- “Setting up <TestName>_<OS>_<Browser>_<BrowserVersion>_<LongName>_<Orientation> for Android on Appium”

—Human-readable output once for entire suite

- “Testing on <n> platforms”



Other Benefits

- Status notification baked in
 - Tests show "Pass" or "Fail" rather than "Finished"
- SauceLabs SessionID captured
 - Supporting Jenkins Sauce Labs Test Publisher
- Test names for SauceLabs website
 - Human-readable names (same scheme as written to console)

Utility classes provided

- WebControl (Java and .NET)
 - Greatly streamlines WebElement interaction
- PageObjectBase (Java and .NET)
 - Common methods and graceful handling of Apple device page load issue
- SeleniumExtensions (.NET RemoteWebDriver)
 - IWebElement and page load methods for jQuery

Other Relevant Links

- <http://fullcirclesolutions.com.au/benefits.html>
- <http://fullcirclesolutions.com.au/SauceryNUnit/home.html>
- <https://github.com/Full360Solutions/Saucery>
- <http://fullcirclesolutions.com.au/howto.html>
- <http://jenkinsheaven.blogspot.com.au>



Contact Us

- Web: <http://fullcirclesolutions.com.au>
- Email: contact@fullcirclesolutions.com.au
- LinkedIn: <https://au.linkedin.com/in/andrewpaulgray>
- Skype: [apgray3](#)