JNario

Executable Specifications for Java

Sebastian Benz - E.S.R. Labs

Birgit Engelmann - BMW Car IT GmbH

```
public class CustomerLookupTest{

  @Test
  public void findsCustomerById() { ...
  }
  @Test
  public void failsForDuplicateCustomers() { ...
  }

   ...

}
```

CustomerLookup

findsCustomerById

failsForDuplicateCustomers

...

Customer Lookup

-finds Customer by Id

-fails for Duplicate Customers

-...

Executable Specification

```java
public class ResourceSetTest {

   @Test
   public void loadingResourcesByFileUri() {
      givenAnEcoreFile("Example.ecore");

      URI fileUri = URI.createFileURI("Example.ecore");
      Resource resource = new ResourceSetImpl().getResource(fileUri);

      assertThat(resource.getContents(), is(not(empty())));
   }

}
```

ResourceSet

Loading Resources by File Uri:

```
givenAnEcoreFile("Example.ecore");

URI fileUri = URI.createFileURI("Example.ecore");
Resource resource = new ResourceSetImpl().createResource(fileUri);

assertThat(resource.getContents(), is(not(empty())));
```

Executable Documentation

Tests can be:

- executable specifications
- executable documentation

...with JUnit?

Jnario

- A DSL for testing based on Xtend
- Readable acceptance specifications
- Succinct unit tests
- Executable documentation

Demo: Coffee Trackr

Cucumber-style Specs with a Twist

Calculator.feature

```
package calculator

Feature: Addition

In order to avoid silly mistakes
As a math idiot
I want to be told the sum of two numbers

Scenario: Add two numbers
    When I entered "50" and "70"
    Then the result should be "120"
```

Cucumber-style Specs with a Twist

Calculator.feature

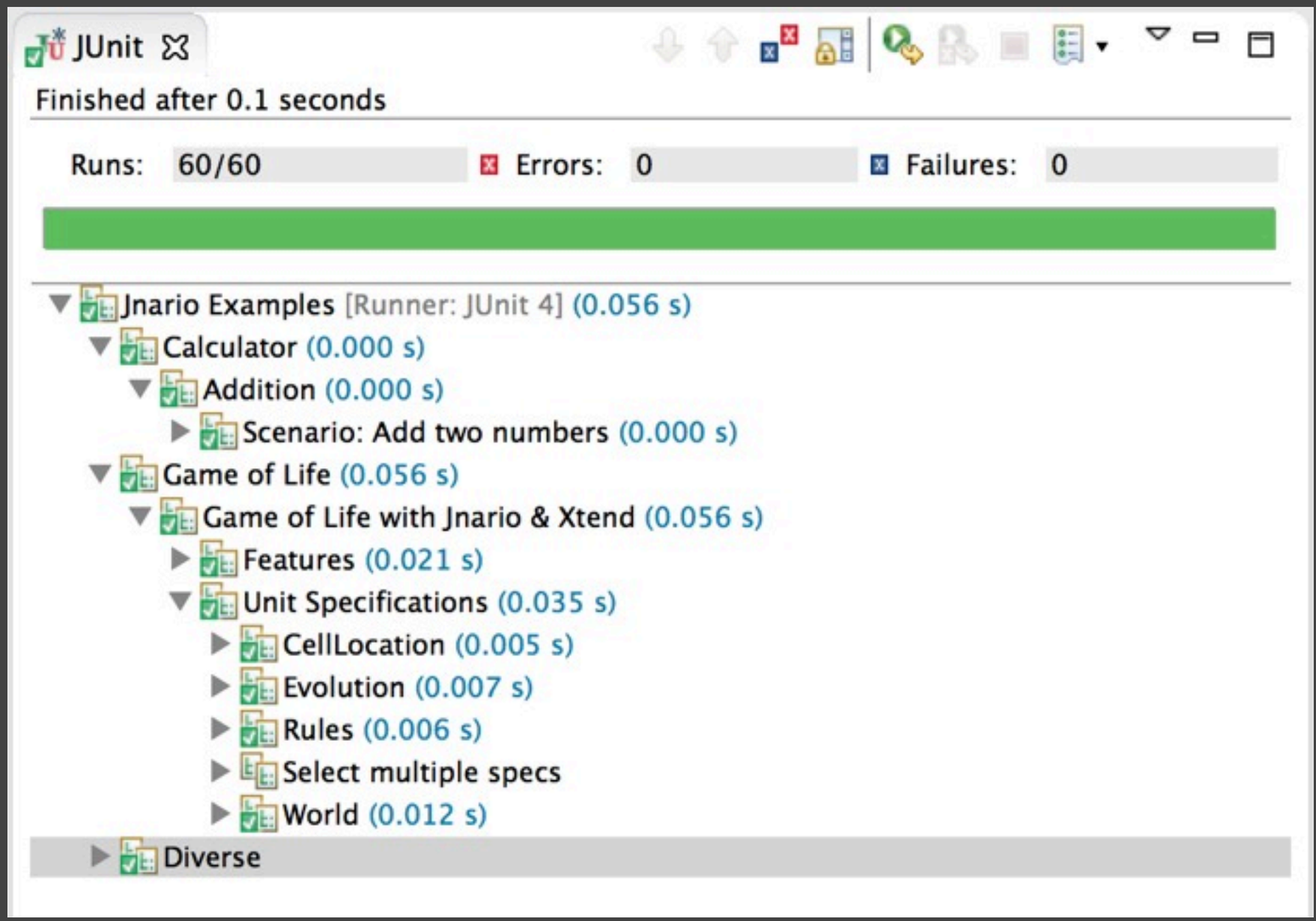
```
package calculator

Feature: Addition

  In order to avoid silly mistakes
  As a math idiot
  I want to be told the sum of two numbers

  Scenario: Add two numbers
    val calculator = new Calculator()
    int result
    When I entered "50" and "70"
      result = calculator.add(args.first, args.second)
    Then the result should be "120"
      result => args.first.toInt
```

Write succinct Unit Specifications

Stack.spec

```
describe Stack{
    context "empty"{
        fact subject.empty should be true
        fact subject.pop() throws EmptyStackException
    }
    context "not empty"{
        fact "increases size when pushing"{
            subject.push("something")
            subject.size => 1
        }
        fact "decreases size when popping"{
            subject.push("something")
            subject.pop()
            subject.size => 0
        }
    }
}
```

100% Java & JUnit compatible



Test Suites on Steroids

Example4.suite

#Test Suites on Steroids

Describe your test suite in plain **formatted** text.

##List existing Specs

Reference existing specs.

- "Stack": this is a reference to our Stack specification.

##Select multiple specs

Select multiple specs using regular expressions.

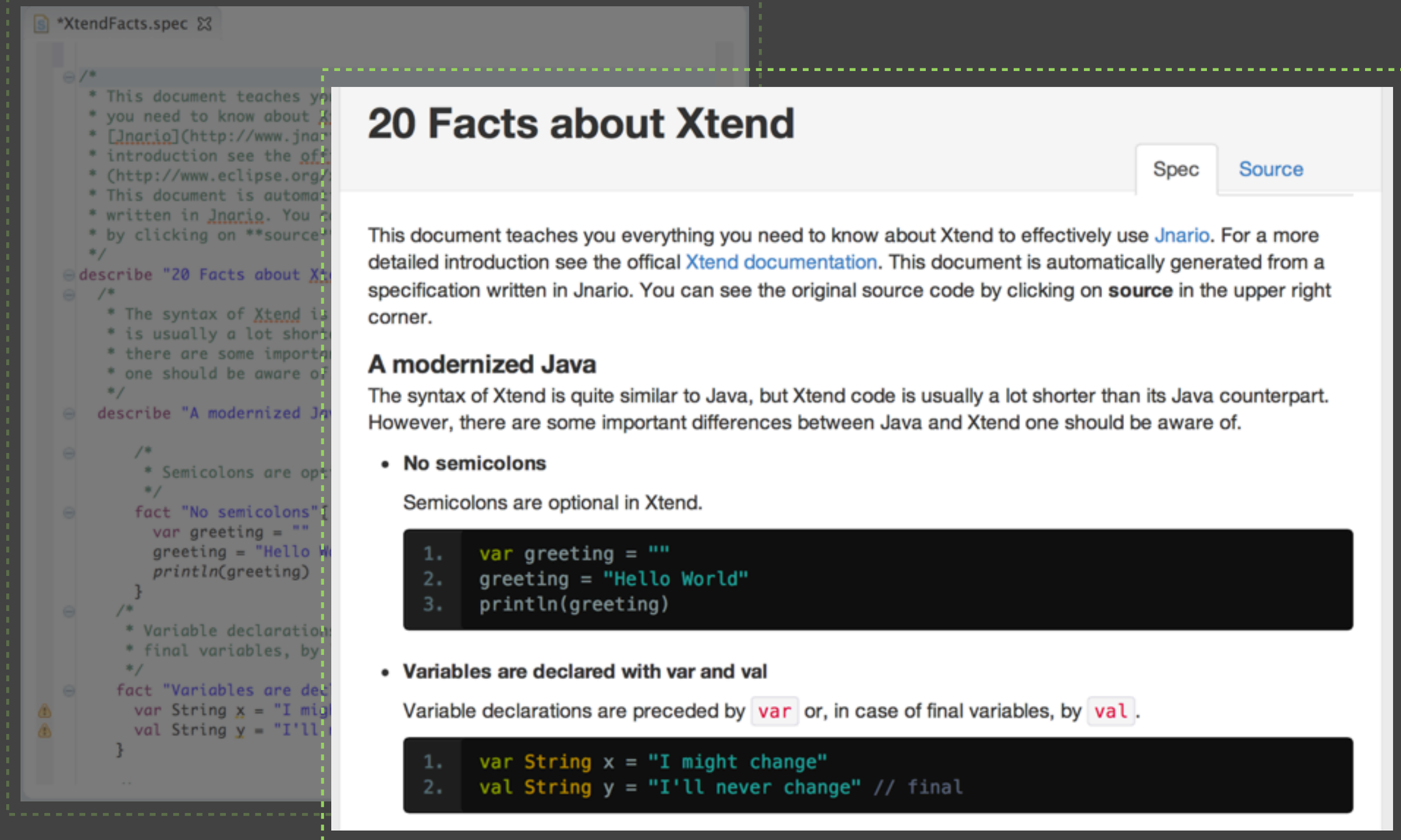
- \diverse.*\

Executable Documentation

*XtendFacts.spec

```
/*
 * This document teaches you everything
 * you need to know about Xtend to effectively use
 * [Jnario](http://www.jnario.org). For a more detailed
 * introduction see the offical [Xtend documentation]
 * (http://www.eclipse.org/xtend/documentation.html).
 * This document is automatically generated from a specification
 * written in Jnario. You can see the original source code
 * by clicking on **source** in the upper right corner.
 */
describe "20 Facts about Xtend"{
  /*
   * The syntax of Xtend is quite similar to Java, but Xtend code
   * is usually a lot shorter than its Java counterpart. However,
   * there are some important differences between Java and Xtend
   * one should be aware of.
   */
  describe "A modernized Java"{

      /*
       * Semicolons are optional in Xtend.
       */
      fact "No semicolons"{
        var greeting = ""
        greeting = "Hello World"
        println(greeting)
      }
    /*
     * Variable declarations are preceded by `var` or, in case of
     * final variables, by `val`.
     */
    fact "Variables are declared with var and val"{
      var String x = "I might change"
      val String y = "I'll never change" // final
    }
```

Executable Documentation



www.jnario.org

Sebastian Benz - @sebabenz

Brigit Engelmann - @engelmannb