

Testing using FitNesse and JUnit

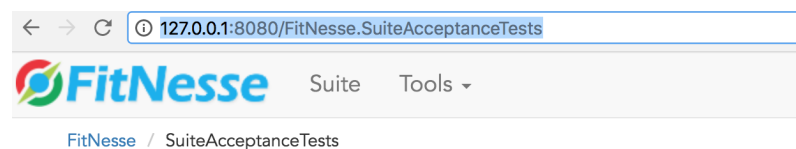
Overview

In this tutorial, we will use [FitNesse](#) as the master of information for creating and editing acceptance tests.

We will use FitNesse JUnit output capabilities in order to get visibility of the results of the automated tests managed by FitNesse.

Description

Whenever you install FitNesse, you have access to several examples of acceptance tests.



FitNesse acceptance tests suites

>SuiteWidgetTests	Specify the markup language of wiki pages.
>SuiteWikiPageResponderTests	Specify general behavior of all wiki pages.
>SuiteEditResponderTests	Specify the behavior of edit requests.
>SuiteResponderTests	Specify the behavior of the responders.
>SuiteWikiImportTests	Specify the behavior of the Wiki Import feature
>SuiteAuthenticationTests	Specify the authentication design
>SuiteFixtureTests	Specify the behavior of various fixtures
>SuiteSymbolicLinkTests	Demonstrate usage and behavior of symbolic links
>SuiteFitDecoratorTests	Demonstrate usage and behavior of Fit Decorators
>SuiteSlimTests	Slim Specific Tests
>SuiteTestHistory	Test History tests
>SuiteVersionsControllerTests	Versions controller tests
>SuiteTestSystems	Test interaction with custom test systems

If you take out the relative path of where you have your tests specified, you can use it in order to run those tests in the command line.

```
java -jar fitnesse-standalone.jar -p 8080 -c "FitNesse.SuiteAcceptanceTests?suite&format=junit" > tmp.xml
```

Unfortunately, FitNesse does not generate a clean, ready-to-use JUnit XML file; thus, you'll need to "clean" it.

```
fgrep -A 1000000 -i "<?xml" tmp.xml > results.xml
```

After successfully running the Test Case and generating the JUnit XML report (e.g., [results.xml](#)), it can be imported to Xray (by using either the REST API or the **Import Execution Results** action within the Test Execution).

Tests

Overall Execution Status

301 PASSED1 FAILED

CREATE TEST

+ Add

TOTAL TESTS: 302

10

Columns

Key	Summary
<input type="checkbox"/> CALC-57	FitNesse.SuiteAcceptanceTests.SuiteAuthenticationTests.AlwaysSecureOperation
<input type="checkbox"/> CALC-14	FitNesse.SuiteAcceptanceTests.SuiteAuthenticationTests.TestReadChildOfSecurePage
<input type="checkbox"/> CALC-25	FitNesse.SuiteAcceptanceTests.SuiteEditResponderTests.TestEditNewPage
<input type="checkbox"/> CALC-45	FitNesse.SuiteAcceptanceTests.SuiteEditResponderTests.TestUserNameSaved
<input type="checkbox"/> CALC-34	FitNesse.SuiteAcceptanceTests.SuiteResponderTests.AddChildResponderSuite.AddNormalWithNameOfSuite
<input type="checkbox"/> CALC-8	FitNesse.SuiteAcceptanceTests.SuiteResponderTests.AddChildResponderSuite.CantAddChildWithNoNameTe
<input type="checkbox"/> CALC-43	FitNesse.SuiteAcceptanceTests.SuiteResponderTests.SuiteMoveResponder.TestMovePageThreeLevelsDown
<input type="checkbox"/> CALC-28	FitNesse.SuiteAcceptanceTests.SuiteResponderTests.SuiteMoveResponder.TestMovePageWithSubPages
<input type="checkbox"/> CALC-256	FitNesse.SuiteAcceptanceTests.SuiteResponderTests.SuitePropertiesResponder.TestPropertyCheckboxesAre
<input type="checkbox"/> CALC-98	FitNesse.SuiteAcceptanceTests.SuiteResponderTests.SuiteRenameResponder.TestRenameMovesChildPages

Prev

12345...31Next

JUnit's Test Case is mapped to a Generic Test in Jira, and the **Generic Test Definition** field contains the name of the class, and the method name that implements the Test Case.

The Execution Details of the Generic Test contains information about the Test Suite, which in this case corresponds to the Test Case class.

Test Details

Test Type:

Generic

Definition:

FitNesse.SuiteAcceptanceTests.FitNesse.SuiteAcceptanceTests.SuiteResponderTests.AddChildResponderSuite.AddNormalWithNameOfSuiteTest

Results

Context	Error Message	Duration	Status
TestSuite FitNesse.SuiteAcceptanceTests	-	21 millisec	PASSED

References

- <http://docs.fitnesse.org/FrontPage>