

Integration Testing

Integration testing involves the testing of larger segments of your Tapestry module or web application, typically including the user interface.

The Tapestry Test Utilities is a small library to make it easier to build integration test suites around [Selenium](#) version 2.14.0.

Related Articles

- [Unit testing pages or components](#)
- [Integration Testing](#)

The strategy is to start, in-process, a Selenium Server (which, in turn, starts and manages a web browser), a Jetty instance (for the web browser to talk to), and a Selenium client (which talks to the server).

The client is able to request URLs, fill in form data, click links, and make assertions about output and behavior.

Usage

The core part of this library is a base class for you to extend your tests classes : [SeleniumTestCase](#).

This class is responsible for starting an instance of Jetty to server your web application, as well as a copy of Selenium Server. It also implements the [Selenium](#) interface.



Before Tapestry 5.2, your class should extend AbstractIntegrationTestSuite

Here's an example from one of the Tapestry modules:

Your Integration Test Class : SinglePersistenceUnitIntegrationTest.java

```
package org.apache.tapestry5.jp.integration.app2;

import org.apache.tapestry5.test.SeleniumTestCase;
import org.testng.annotations.Test;

public class SinglePersistenceUnitIntegrationTest extends SeleniumTestCase
{
    @Test
    public void persist_entities()
    {
        open("/persistitem");
        assertEquals(getText("//span[@id='name']").length(), 0);

        clickAndWait("link=create item");
        assertText("//span[@id='name']", "name");
    }
}
```

With the SeleniumTestCase class, you can use basic Selenium methods (such as open() and type()) and methods added by the SeleniumTestCase base class (clickAndWait() and assertFieldValue()).

In addition, the SeleniumTestCase base class extends the normal exception reporting provided by Selenium; when a failure occurs inside Selenium server, a more detailed message, including the current page's HTML source, is reported to System.err.

Configuration

All the configuration of your Integration Tests should be in your testng.xml file. Tapestry provides some parameters, in order to have the right environment for your tests.

Parameter	Default Value	Value
tapestry.web-app-folder	src/main/webapp	The path to a web app
tapestry.servlet-container	jetty7	the server container to use for the integration tests (jetty7 or tomcat6)
tapestry.context-path		The context path
tapestry.port	9090	The web server port
tapestry.ssl-port	8443	The web server ssl port

tapestry.browser-start-command	*firefox	The browser command to pass to Selenium
--------------------------------	----------	---

Here's an example :

testng.xml

```
<suite name="Selenium Tests Suite" annotations="1.5">
  <test name="Integration Tests" enabled="true">
    <parameter name="tapestry.browser-start-command" value="*googlechrome" />
    <parameter name="tapestry.port" value="9091" />

    <classes>
      <class name="com.example.newapp.SeleniumTest"></class>
    </classes>
  </test>
</suite>
```

Some Interesting Tools

Here are some interesting plugins you can use to write your integration tests.

- Google Chrome : [xpathOnClick](#)
- FireFox : [Selenium IDE](#)