



Testing of Web Applications

Patrice Neff <patrice@local.ch>

t.camp 2006 at the lovely Kartause Ittingen

Agenda

■ Theory

- + Unit Testing
- + Integration testing
- + Test-Driven Development

■ xUnit Tools

- + JUnit
- + PHPUnit, SimpleTest
- + JSUnit

■ Selenium

- + Demo

■ Best Practices

■ Challenges

Theory

(not too boring I hope)

Unit Testing

- Tests modules (usually classes)
- Verifies the public interface (API)
- A test should test just one module
- Great for refactoring - you are reassured that your code still works
- Can test for the module requirements

Integration Tests

- Tests interaction of modules (“the big picture”)
- Also for interaction with other systems (mails for example)

Test-Driven Development

■ Dogmatic:

- + Tests are written before the code
- + Only code is written for which tests already exist

■ Pragmatisch:

- + Tests werden vor und/oder mit dem Code geschrieben
- + Wichtigste Fälle werden getestet
- + Für jeden Bugfix wird ein Test geschrieben

xUnit Tools

xUnit

- x is for J, PHP, N,
- Principles are always the same:
 - + Test classes
 - + Test methods with assertions
 - + setUp / tearDown methods for fixtures
 - + Framework gives detailed information about failed tests (stack trace and comparison of expected/received value)
- List of tools: <http://www.xprogramming.com/software.htm>

JUnit

- Made unit testing popular
- Based on ideas for SUnit (for Smalltalk)
- URL: <http://www.junit.org/>

PHPUnit, SimpleTest

- PHPUnit2

- + Good documentation

- SimpleTest:

- + Can also be used for functional testing (can do HTTP requests to the application)

- URLs:

- + <http://pear.php.net/package/PHPUnit2>
- + <http://simpletest.sourceforge.net/>

JSUnit

- Unit Testing for JavaScript
- Good when there are JavaScript objects with logic
- Runs in the browser
- URL: <http://www.jsunit.net/>

Selenium

Selenium

- Tests run in the browser
- Can execute actions such as clicks, typing text, etc.
- Tests are written in Selenese (HTML table with commands)
- Extensions in JavaScript
- URLs:
 - + <http://www.openqa.org/selenium/>
 - + <http://www.openqa.org/selenium-core/documentation.html>

Selenium Versions

- Three different versions (that matter)
- Core: JavaScript library
- IDE: Firefox extension
- Remote Control: Proxy

	Core	IDE	Remote Control
Browsers	All	Firefox	Many
Installation on server	Yes	No	No
HTTPS	Yes	Yes	No
Multidomain	Nein	Yes	No
Java required	Nein	No	Yes

Selenese

- Easy to write test syntax
- Example (Initialization of our tests):

Make sure we're logged in		
open	/static/php/resetsession.php	
open	/login/?logout	
open	/de/	
verifyPath	/de/login/?path=http*	
type	username	selenium
type	password	WiccovLarc
clickAndWait	//input[@value='Einloggen']	
verifyPath	/de/	

Make sure we're logged in		
open	/static/php/resetsession.php	
open	/login/?logout	
open	/de/	
verifyPath	/de/login/?path=http*	
type	username	selenium
type	password	WiccovLarc
clickAndWait	//input[@value='Einloggen']	
verifyPath	/de/	

Step by Step

1. Copy Selenium Core to the server
2. Create test directory
3. TestSuite.html - Table with links to all tests
4. Write test - 3-column table filled with Selenese
5. Open test runner in browser (core/TestRunner.html)
6. Select test suite
7. Execute tests

Automation

- Tests result can be posted to a page:
`core/TestRunner.html?test=..%2Ftests%2FTestSuite.html&resultsUrl=%2Fstatic%2Fphp%2Ftestpost.php&auto=true`
- See what values are posted:

```
foreach ($_POST as $key => $value) {  
    echo "$key = $value\n";  
}
```
- At local.ch the results are written to a database

Automation: Values to read

- result ('passed' when successful)
- totalTime
- numTestPasses
- numTestFailures
- numCommandPasses
- numCommandFailures
- numCommandErrors
- numTestTotal
- selenium_version and selenium_revision

Date	2006-08-30 04:42
Type	selenium
Status	OK
Time running	107 seconds
Tests passed	8
Tests failed	0
Commands passed	137
Commands failed	0
Commands with errors	0
Notes	SLD: tel Result: passed Failed Tests: Selenium Version: 0.7.1 Selenium Revision: 1253M Browser: Mozilla/5.0 (Windows; U; Windows NT 5.2; en-US; rv:1.8.0.6) Gecko/20060728 Firefox/1.5.0.6 Host: 10.181.32.13

Continuous Testing

- You need: Computer with UI
- Schedule a browser to open `core/TestRunner.html?.....&auto=true` frequently

Caveats

- Domain handling
- XPath not consistent over browsers
- Umlaut in URLs not consistent over browsers
- Cookies
 - + Need to reset at beginning of tests
 - + .local.ch cookies are also valid on .trunk.local.ch (can be removed though)
- HTTP authentication

Domain Handling

- Selenium Core can only test on one domain
- Can be circumvented:
 - + testpost script redirects to next domain after tests are posted to the database (JavaScript redirect)

Example this Week

- `http://www.local.ch/de/nonexistingpage` should return 404
- Is tested in Selenium:
- I saw red:

open	/de/nonexistingpage
verifyPath	/de/nonexistingpage
verifyTextPresent	Es tut uns leid, die von Ihnen gesuchte Seite existiert nicht.
verifyTitle	local.ch - Telefonbuch und Karte der Schweiz

2006-08-30 14:44	selenium	OK	Detail To Jira
2006-08-30 14:43	selenium	ERROR	Detail To Jira
2006-08-30 14:43	selenium	ERROR	Detail To Jira
2006-08-30 14:41	selenium	ERROR	Detail To Jira
2006-08-30 12:43	selenium	OK	Detail To Jira

- Subversion showed the cause and 15 minutes later the bug was fixed

Best Practices

General

- Write a test when you fix bugs: I link to the bug in question

LCL-1619 : Test that latin1 encoded URLs are handled	
open	/de/r%FCti
verifyPath	/de/Frauenfeld
open	/de/Z%FCrich
verifyPath	/de/Z%C3%BCrich

Selenium

- Use Selenium IDE to quickly record tests
- Initialize environment at the beginning
- Use `<th colspan="3">` for documentation
- `user-extensions.js` for your own tests
- Regexp testing when more than one condition is possible

<code>verifyText</code>	<code>//div[@id='topad']/p/a</code>	<code>regexp:Traianou Christian Cantelmi Gianni Malak Oliver</code>
-------------------------	-------------------------------------	---

Challenges

Social

- It's NOT a waste of time

- + It's not much extra work when you do it right away
- + Manual testing is reduced to a minimum
- + You know a bug really is fixed (and will remain fixed)
- + But: requires some initial effort

- Get the team on board

- + For the local.ch frontend a bug is only fixed when a test exists
- + Show the value (“I found this bug because of Selenium”)

Questions or Remarks?

Hasta Luego

Für Fragen und Feedback

<patrice@local.ch>

oder Skype: patriceneff

oder Face2Face

