

Manažment testovania

Metodika pre testovanie implementácie v jazyku PHP pomocou nástroja Selenium IDE a PHPUnit

Téma číslo 2

Bc. Marek Sobôtka

| | |
|------------------------|--|
| Študijný odbor: | Softvérové inžinierstvo |
| Ročník: | 1., 1. semester štúdia |
| Predmet: | Manažment projektov softvérových a informačných systémov |
| Akademický rok: | 2010/2011, zimný semester |

1 Úvod

Účelom tejto metodiky je definovať celkový postup testovania produktu na vyššej úrovni. Metodika zahŕňa aj postup ako vytvárať jednotkové testy (unit tests) v nástroji Selenium IDE a ako ich spúšťať v nástroji PHPUnit a ako ich následne spracovať (od kapitoly 5). Metodika je určená najmä pre malé a stredne veľké tímy, ktoré vyvíjajú aplikácie v jazyku PHP.

2 Pojmy a skratky

- **Selenium IDE** – integrované vývojové prostredie pre vytváranie testovacích skriptov, poskytované ako rozšírenie do webového prehliadača Firefox
- **Selenium RC (Remote Control)** – testovací nástroj, ktorý umožňuje spúšťať aplikačné testy napísané v podporovanom jazyku voči stránkam dostupným cez HTTP protokol
- **PEAR** – rámec na distribúciu systémových a znovu použiteľných PHP komponentov
- **PHPUnit** – rámec pre písanie jednotkových testov, poskytujúci možnosť ich aj spúšťať a analyzovať získané výsledky
- **XML (eXtensible Markup Language)** – značkovací jazyk na ukladanie a výmenu štruktúrovaných dát
- **XPath (XML Path Language)** – jazyk na adresovanie jednotlivých častí XML dokumentu
- **XPather** – rozšírenie do webového prehliadača Firefox, ktoré poskytuje funkcionality na generovanie XPath-ov potrebných do testovacích skriptov

3 Roly a zodpovednosti

| Rola | Zodpovednosť v rámci manažmentu testovania |
|----------------------------|--|
| Softvérový návrhár | <ul style="list-style-type: none"> ➤ vytvorenie akceptačných testov pre navrhovanú funkcionality |
| Vývojár | <ul style="list-style-type: none"> ➤ programovanie samotnej funkcionality systému |
| Tvorca jednotkových testov | <ul style="list-style-type: none"> ➤ otestovanie funkcionality voči akceptačným testom ➤ tvorba jednotkových testov ➤ hlásenie nezhôd naprogramovanej funkcionality s príslušným akceptačným testom |
| Tester | <ul style="list-style-type: none"> ➤ integrácia jednotkových testov do skriptu na dávkové spúšťanie testov ➤ zbehnutie jednotkových testov ➤ zdokumentovanie výsledkov |
| Manažér kvality | <ul style="list-style-type: none"> ➤ analýza výsledkov testov |

| | |
|-------------------------|---|
| | ➤ hlásenie nezhôd naprogramovanej funkcionality s požiadavkami |
| Systemový administrátor | ➤ inštalácia a konfigurácia vyžadovaných nástrojov pre testovanie |

Jedna osoba môže zastávať aj viacero rolí, ale platí obmedzenie, že tvorca jednotkových testov, ktorý vykonáva aj akceptačné testy nesmie byť tá istá osoba ako vývojár príslušnej testovanej funkcionality.

4 Opis definovaných procesov

4.1 Proces inštalácie a konfigurácie nástrojov pre testovanie

Proces inštalácie a konfigurácie nástrojov pre testovanie zabezpečuje inštaláciu a správnu konfiguráciu nevyhnutných nástrojov. Tento proces prebieha iba raz pred prvým vykonaním procesu prípravy jednotkových testov. Proces prebieha v nasledujúcich krokoch:

| # | Krok | Príslušná kapitola |
|----|--|--------------------|
| 1. | Inštalácia a konfigurácia nevyhnutných nástrojov | 5.1 |

4.2 Proces prípravy jednotkových testov

Proces prípravy jednotkových testov zabezpečuje vytvorenie jednotkových testov, ktoré otestujú implementovanú funkcionality. Tento proces môže prebiehať až po vykonaní *procesu vytvorenia akceptačných testov* (Proces vytvorenia akceptačných testov je opísaný v *Metodike pre tvorbu softvérového návrhu*) a vykonaní *procesu implementácie požadovanej funkcionality* (Proces implementácie požadovanej funkcionality je opísaný v *Metodike pre vývoj a implementáciu novej požiadavky*). Proces prebieha po skončení implementácie každej dôležitej časti systému. Jednotkové testy sa vytvárajú v zásade až po implementácii funkcionality z dôvodu využitia možnosti vyvírať testy v nástroji Selenium IDE pomocou odklikania jednotlivých krokov testu. Proces prebieha v nasledujúcich krokoch:

| # | Krok | Príslušná kapitola |
|----|---|--------------------|
| 1. | Otestovanie implementovanej funkcionality voči akceptačným testom | 6.1 |
| 2. | Vytvorenie jednotkových testov na otestovanie funkcionality stanovenej v akceptačných testoch | 6.2 |
| 3. | Integrácia jednotkových testov do skriptu na dávkové spúšťanie testov | 6.3 |

V prípade, že v prvom kroku sa odhalia nedostatky, *proces prípravy jednotkových testov* nepokračuje druhým krokom, ale zahajuje sa *proces hlásenia chyby* (Proces hlásenia chyby je opísaný v *Metodike pre manažment chýb*).

4.3 Proces priebehu, zdokumentovania a zhodnotenia testov

Proces priebehu, zdokumentovania a zhodnotenia testov zabezpečuje vyhodnotenie, či všetky požadované funkcie systému sú v súlade s požiadavkami. Tento proces prebieha po dokončení *procesu prípravy jednotkových testov* (Proces prípravy jednotkových testov je opísaný v *Metodike pre testovanie implementácie v jazyku PHP pomocou nástroja Selenium IDE a PHPUnit*). Proces prebieha v nasledujúcich krokoch:

| # | Krok | Príslušná kapitola |
|----|--|--------------------|
| 1. | Zbehnutie skriptu na dávkové spúšťanie testov | 7.1 |
| 2. | Zdokumentovanie výsledkov priebehu testov | 7.2 |
| 3. | Analýza a zhodnotenie zdokumentovaných výsledkov | 7.3 |

V prípade kladného výsledku s nezistenými chybami sa dokumentácia výsledku priebehu testov priloží k celkovej dokumentácii systému. V prípade negatívneho výsledku sa zahajuje *proces hlásenia chyby* (Proces hlásenia chyby je opísaný v *Metodike pre manažment chýb*).

5 Opis krokov procesu inštalácie a konfigurácie nástrojov pre testovanie

5.1 Inštalácia a konfigurácia nevyhnutných nástrojov

Vstup: Požiadavka na inštaláciu a konfiguráciu nástrojov pre testovanie

Výstup: Nainštalované a správne nakonfigurované nástroje potrebné pre testovanie

Zodpovedný: Systémový administrátor

Zoznam nevyhnutných programov a nástrojov pre vytváranie a spúšťanie jednotkových testov:

| Názov programu/nástroja | Krátky opis programu/nástroja | Odkaz na stiahnutie |
|--------------------------|---|---|
| Zend Server CE | webový aplikačný server pre PHP aplikácie | http://www.zend.com/en/products/server-ce/downloads |
| Mozilla Firefox | webový prehliadač | http://www.mozilla.com/en-US/ |
| Selenium IDE | rozšírenie do Firefox-u na vytváranie jednotkových testov | http://seleniumhq.org/download/ |
| XPather | rozšírenie do Firefox-u na generovanie XPath-ov | https://addons.mozilla.org/en-US/firefox/addon/1192/ |
| Java Runtime Environment | rámec pre spúšťanie Java programov | http://www.oracle.com/technetwork/java/javase/downloads/index.html |
| Selenium RC | testovací nástroj na spúšťanie aplikačných testov | http://seleniumhq.org/download/ |
| PEAR | rámec na distribúciu PHP komponentov | súčasť Zend Server-u CE |
| PHPUnit | rámec pre písanie a spúšťanie jednotkových testov | stiahnuteľné cez PEAR |

Postup inštalácie a konfigurácie:

1. Stiahnuť a nainštalovať Zend Server CE
2. Stiahnuť a nainštalovať Mozilla Firefox
3. Stiahnuť a nainštalovať rozšírenie Selenium IDE do Mozilla Firefox-u
4. Stiahnuť a nainštalovať rozšírenie XPath do Mozilla Firefox-u (toto rozšírenie nie je nevyhnutné, ale urýchli prácu pri procese prípravy jednotkových testov)
5. Stiahnuť a nainštalovať Java Runtime Environment
6. Stiahnuť a nainštalovať Selenium RC

- a. samotná inštalácia prebehne rozbalením archívu do ľubovoľného priečinka, napríklad `C:\Program Files\Selenium`
7. Nainštalovať PEAR, ktorý je súčasťou Zend Server-u CE
 - a. pri štandardnej inštalácii Zend Server-u CE sa inštalačný súbor nachádza v priečinku `C:\Program Files\Zend\ZendServer\bin`
 - b. dvojklikom na `go-pear.bat` spustiť inštaláciu
 - c. v prípade problému s podpisom zmeniť obsah `go-pear.bat` tretieho riadku na `%PHP_BIN% -d output_buffering=0 -dphar.require_hash=0 PEAR\go-pear.phar` a inštaláciu zopakovať
 - d. v prípade, že je PEAR správne nainštalovaný, príkaz `pear -V` vráti aktuálnu verziu
8. Aktualizovať PEAR cez konzolu príkazom `pear upgrade pear`
9. Nainštalovať PHPUnit pomocou PEAR-u
 - a. zadať do konzoly príkaz `pear channel-discover pear.phpunit.de`
 - b. zadať do konzoly príkaz `pear channel-discover pear.symfony-project.com`
 - c. zadať do konzoly príkaz `pear channel-discover components.ez.no`
 - d. zadať do konzoly príkaz `pear install phpunit/PHPUnit`

6 Opis krokov procesu prípravy jednotkových testov

6.1 Otestovanie implementovanej funkcionality voči akceptačným testom

Vstup: Implementovaná funkcionality systému a príslušné akceptačné testy

Výstup: V závislosti či je implementovaná funkcionality v súlade s akceptačnými testami buď požiadavka na vytvorenie jednotkových testov alebo správa o chybe

Zodpovedný: Tvorca jednotkových testov

Postup otestovania:

1. Preštudovať akceptačný test
2. Overiť správnosť implementácie voči akceptačnému testu
3. Zapísať výsledky akceptačného testu do dokumentu
4. V prípade, že dokument obsahuje ďalší akceptačný test, pokračovať krokom č. 1, inak krokom č. 5
5. V prípade, že implementácia je v súlade so všetkými akceptačnými testami, pokračovať vytvorením jednotkových testov (Kapitola 6.2), inak odoslať správu o chybe (*Metodike pre manažment chýb* - Proces hlásenia chyby)

6.2 Vytvorenie jednotkových testov na otestovanie funkcionality stanovenej v akceptačných testoch

Vstup: Požiadavka na vytvorenie jednotkových testov

Výstup: Implementované jednotkové testy

Zodpovedný: Tvorca jednotkových testov

Postup vytvorenia jednotkového testu:

1. Spustiť Mozilla Firefox
2. Spustiť webovú stránku, ktorá sa ide testovať
3. Spustiť nahrávanie testovacieho scenára pre vytvorenie jednotkového testu v Selenium IDE
 - a. Spustiť Selenium IDE v Menu Firefox → Tools → Selenium IDE
 - b. Stlačiť červené tlačidlo pre spustenie nahrávania
4. Nahrať jednotkový test odklikaním postupnosti testovacích krokov na zobrazenej stránke
5. Vyexportovať jednotkový test v Menu Selenium IDE → File → Export Test Case As... → PHP Selenium RC
6. Premenovať triedu a súbor podľa vzoru
 - a. názov triedy je v tvare TC_<číslo úlohy na tri miesta>_<číslo akceptačného testu na dve miesta>_Test
 - b. názov súboru je zhodný s názvom triedy, iba sa pridá prípona .php
 - c. je nevyhnutné dodržiavať túto konvenciu, pretože PHPUnit pri dávkovom testovaní testuje iba súbory, ktorých názov nezačína na číslo a končí práve na Test
7. Upraviť vyexportovaný jednotkový test
 - a. Doplniť kód podľa potreby

Príklad exportovaného a upraveného jednotkového testu v súbore TC_001_01_Test.php:

```
<?php
```

```
require_once 'PHPUnit/Extensions/SeleniumTestCase.php';
```

```
class TC_001_01_Test extends PHPUnit_Extensions_SeleniumTestCase
{
    protected function setUp()
    {
        $this->setBrowser("*firefox C:\Program Files\Mozilla
Firefox\Firefox.exe");
        $this->setBrowserUrl("http://delicious.com/");
    }
}
```

```

}

public function testMyTestCase()
{
    $this->open("/");
    $this->type("homepage-searchinput", "selenium");
    $this->click("homepage-searchsubmit");
    $this->waitForPageToLoad("30000");
    $count = $this-
>getText("xpath=/html/body[@id='index']/div[@id='doc3']/div[@id=
'bd']/div[@id='yui-main']/div[@id='content']/h3/div/em");
    $count = str_replace(",", "", $count);
    $this->assertGreaterThan(1000, $count);
}
}
?>

```

6.3 Integrácia jednotkových testov do skriptu na dávkové spúšťanie testov

Vstup: *Nové jednotkové testy*

Výstup: *Zaradenie nových testov k starým testom*

Zodpovedný: *Tester*

Postup integrácie nových testov:

1. Skopírovať nové súbory do priečinka, kde sa nachádzajú staré testovacie súbory
 - a. v prípade, že žiadne staré testovacie súbory ešte nie sú vytvorené, určíme priečinok, kde budeme testovacie súbory ukladať, napríklad C:\Users\XYZ\Documents\UnitTesting\Tests

7 Opis krokov procesu priebehu, zdokumentovania a zhodnotenia testov

7.1 Zbehnutie skriptu na dávkové spúšťanie testov

Vstup: *Priečinok s jednotkovými testami a stránka s implementovanou funkcionalitou, ktorú testujeme*

Výstup: *Výpis o zbehnutí testov, prípadne zoznam chýb*

Zodpovedný: *Tester*

Aby mohol byť skript na dávkové spustenie testov zbehnutý, je nutné ho najprv vytvoriť. V skutočnosti je potrebné mať dva súbory. Prvý z nich spúšťa Selenium RC, druhý vykonáva spúšťanie jednotlivých jednotkových testov a zapisuje výsledok.

Obsah dávkového súboru na spustenie Selenium RC:

```
cd "C:\Program Files\Selenium\selenium-server-1.0.3"  
java -jar selenium-server.jar
```

V prvom riadku sa nachádza cesta k nainštalovanému Selenium RC, ktorá sa môže pre každý počítač líšiť.

Obsah dávkového súboru na vykonanie jednotkových testov:

```
phpunit "C:\Users\XYZ\Documents\UnitTesting\Tests" >  
"C:\Users\XYZ\Documents\UnitTesting\out.txt"
```

Prvá cesta smeruje na priečinok s jednotkovými testami, druhá cesta smeruje na súbor, kde sa má uložiť výpis.

Postup zbehnutia skriptu:

1. Spustiť skript na spustenie Selenium RC
2. Spustiť skript na vykonanie jednotkových testov

7.2 Zdokumentovanie výsledkov priebehu testov

Vstup: Výpis o zbehnutí testov

Výstup: Dokumentácia o stave výsledkov testovania

Zodpovedný: Tester

Príklad výpisu o zbehnutí testov:

```
PHPUnit 3.5.3 by Sebastian Bergmann.
```

```
FF
```

```
Time: 01:04, Memory: 3.50Mb
```

```
There were 2 failures:
```

```
1) TC_001_01_Test::testMyTestCase
```

```
Current URL:
```

```
http://www.delicious.com/search?p=selenium&chk=&context=main%7C&  
fr=del_icio_us&lc=
```

```
Failed asserting that <string:9191> is greater than  
<integer:20000>.
```

```
C:\Users\XYZ\Documents\UnitTesting\Tests\TC_001_01_Test.php:22
```

```
2) TC_001_02_Test::testMyTestCase
```

Current URL:

`http://www.delicious.com/search?p=selenium&chk=&context=main%7C&fr=del_icio_us&lc=`

Failed asserting that <string:9191> is greater than <integer:10000>.

`C:\Users\XYZ\Documents\UnitTesting\Tests\TC_001_02_Test.php:21`

FAILURES!

Tests: 2, Assertions: 3, Failures: 2.

Najpodstatnejšie sú posledné dva riadky, ktoré prezrádzajú, či boli testy úspešné, alebo nie (ako v tomto prípade).

Postup zdokumentovania výsledkov:

1. Vytvoriť dokumentáciu o stave výsledkov testovania (podrobnosti ako tvoriť dokumentáciu v *Metodike pre manažment technickej dokumentácie* - Proces tvorby dokumentácie o stave výsledkov testovania)
2. Odoslať vytvorenú dokumentáciu manažérovi kvality

7.3 Analýza a zhodnotenie zdokumentovaných výsledkov

Vstup: Dokumentácia o stave výsledkov testovania

Výstup: V prípade, že sa testy boli neúspešné, požiadavka na opravenie chýb

Zodpovedný: Manažér kvality

Postup zhodnotenia výsledkov:

1. Zanalyzovať stav výsledkov testovania
2. V prípade, že testy boli neúspešné, odoslať správu o chybe (*Metodike pre manažment chýb* - Proces hlásenia chyby), inak priložiť dokumentáciu k celkovej dokumentácii systému