

Slovenská technická univerzita
FAKULTA INFORMATIKY A INFORMAČNÝCH TECHNOLOGIÍ

Metodika manažmentu testovania

Patrik Polakovič

November 2011

Obsah

1. Úvod	3
2. Súvisiace metodiky a manuály	3
3. Manažment testovania	3
3.1. Použité pojmy	4
3.2. Roly zahrnuté do procesov testovania.....	4
3.3. Procesy definované v oblasti manažmentu testovania	4
3.3.1. Proces zahájenia testovania.....	5
3.3.2. Proces vytvorenia stratégie testovania.....	6
3.3.3. Proces vytvorenia testovacích scenárov	7
3.3.4. Proces vykonania testovacích scenárov.....	8
3.3.5. Proces reportovania chýb	9
4. Vytvorenie jednotkového testu v prostredí PHPUnit	10
4.1. Určenie rozsahu testu	10
4.1.1. Scenáre testovania.....	10
4.2. Vytvorenie testu.....	11
4.2.1. Zdanlivé objekty	11
4.2.2. Testovanie rozhraní.....	11
4.2.3. Generovanie testu.....	11
4.3. Spustenie testu	11
4.3.1. Konfigurácia PHPUnit	11
4.3.2. Paralelizácia testu	12
4.3.3. Automatizácia testu	12
4.4. Analýza nepokrytých častí kódu.....	12
5. Záver.....	13

1. Úvod

Účelom tejto metodiky je určenie procesov prebiehajúcich počas celej fázy testovania. K procesom definuje zodpovedné osoby ako aj výstupy ktoré musia byť výsledkom daného procesu. Táto metodika je určená pre sekciu **Testovania a Kvality**.

Predmetom metodiky nižšej úrovne je spôsob vytvorenia a vyhodnotenia jednotkového testu v prostredí PHPUnit.

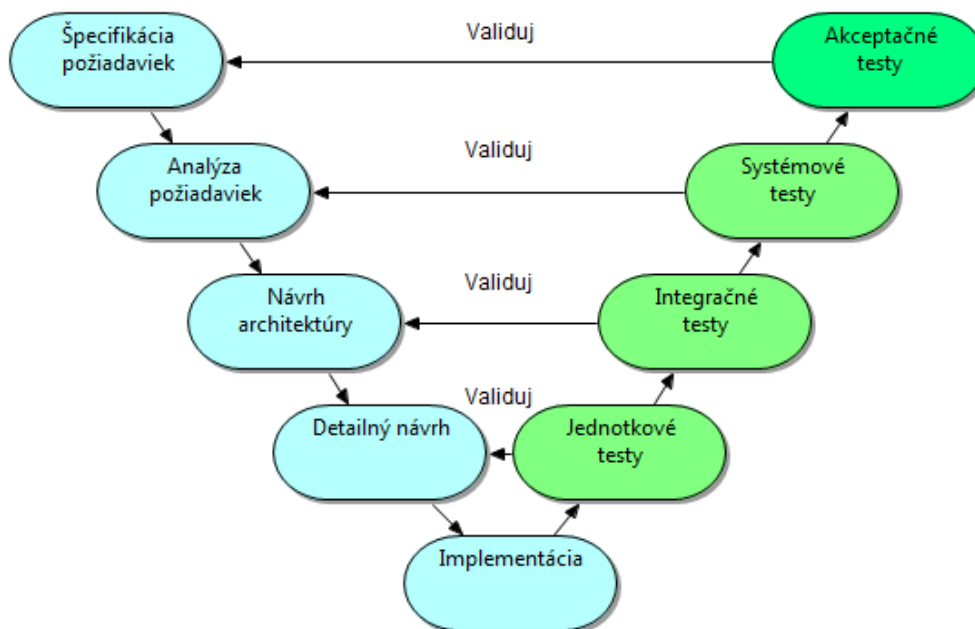
2. Súvisiace metodiky a manuály

Táto metodika sa opiera a odkazuje na nasledovné metodiky a manuály:

- metodika pre reportovanie chýb
- manuál k nástroju PHPUnit
- manuál k nástroju Apache Ant

3. Manažment testovania

Proces testovania je dôležitý v každej fáze života softvérového projektu:



Obr. 1.: V-model životného cyklu softvéru.

3.1. Použité pojmy

- *jednotkový test* – nízkoúrovňový test zameraný na testovanie funkčných jednotiek (zväčša tried) zdrojového kódu
- *PHPUnit* – nástroj pre testovanie zdrojových kódov písaných v jazyku PHP
- *framework* – programový balík nástrojov pre riešenie komplexných úloh
- *testovací scenár* – postupnosť akcií zväčša zameraných na testovanie správania

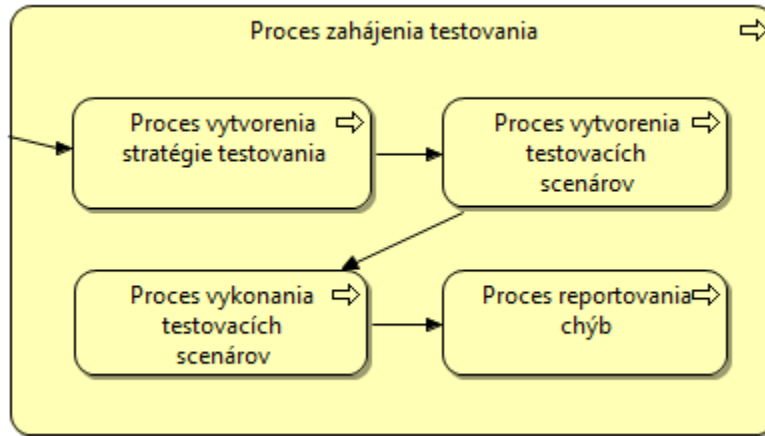
3.2. Roly zahrnuté do procesov testovania

Rola	Zodpovednosti
Projektový manažér	<ul style="list-style-type: none">• inicializácia procesu testovania• koordinácia manažéra testovania
Manažér testovania	<ul style="list-style-type: none">• vedenie a koordinácia vedúcich osôb v testovacích tímoch• vytýčenie cieľov testovania• určenie zodpovedných osôb, ak je to potrebné
Vedúci testovacieho tímu	<ul style="list-style-type: none">• koordinácia testovacieho tímu• výber testovacieho nástroja• určenie efektívneho procesu testovania• vytvorenie plánu testovania
Tester	<ul style="list-style-type: none">• automatické/manuálne vytvorenie testovacích scenárov• spúšťanie samotných testov• identifikácia vzniknutých problémov• vytvorenie postupnosti krokov pri reprodukovateľných chybách• reportovanie odhalených chýb

3.3. Procesy definované v oblasti manažmentu testovania

Proces	Kedy	Kapitola
Proces zahájenia testovania	začatie testovacej fázy	3.3.1
Proces vytvorenia stratégie testovania	vytvorenie cieľov testovania, zahájenie vytvorenia stratégie	3.3.2
Proces vytvorenia testovacích scenárov	ukončenie tvorby stratégie, zahájenie procesu tvorby scenárov	3.3.3
Proces vykonania testovacích scenárov	ukončenie tvorby scenárov	3.3.4
Proces reportovania chýb	získanie a spracovanie testovacích reportov	3.3.5

Nasledovný obrázok znázorňuje postupnosť hore uvedených procesov:



Obr. 1.: Nadväznosť procesov v manažmente testovania

3.3.1. Proces zahájenia testovania

Popis:	analýza vstupnej dokumentácie, stanovenie cieľov testovania na základe charakteru projektu a analýzy vstupného dokumentu	
Iniciátor:	projektový manažér	
Vstup:	projektová dokumentácia	
Výstup:	stanovené ciele testu, otvorená dokumentácia testovania	
Koniec:	vykonanie všetkých scenárov a reportovanie odhalených nedostatkov	
Krok	Popis	Kapitola
1.	Otvorenie a analýza projektovej dokumentácie.	3.3.1.1
2.	Stanovenie cieľov testovania.	3.3.1.2
3.	Otvorenie dokumentácie testovania.	3.3.1.3
4.	Zahájenie procesu výberu stratégie testovania.	3.3.1.4
5.	Uzatvorenie dokumentácie testovania.	3.3.1.5

3.3.1.1. Otvorenie a analýza projektovej dokumentácie

Popis:	analýza charakteru dokumentácie pre potreby vytvorenia cieľov a určenia typu testov
Vstup:	projektová dokumentácia
Výstup:	spracovaná analýza projektovej dokumentácie
Zodpovedná osoba:	projektový manažér, manažér testovania

3.3.1.2. Určenie cieľov testovania

Popis:	stanovenie cieľov, ktoré má dané testovanie splniť
Vstup:	analýza projektovej dokumentácie
Výstup:	ciele testovania

Zodpovedná osoba: manažér testovania

3.3.1.3. Otvorenie dokumentácie testovania

Popis: vytvára sa dokumentácia, v ktorej bude zaznamenaný priebeh testovania

Vstup: ciele testovania

Výstup: otvorená dokumentácia testovania

Zodpovedná osoba: manažér testovania

3.3.1.4. Zahájenie procesu výberu stratégie testovania

Popis: určenie zodpovedných osôb za výber testovacej stratégie, v prípade potreby oboznámenie s cieľmi testovania

Vstup: otvorená dokumentácia testovania

Výstup: proces výberu stratégie testovania

Zodpovedná osoba: manažér testovania, vedúci testovacieho tímu

3.3.1.5. Uzatvorenie dokumentácie testovania

Popis: kompletizácie dokumentácie, zhodnotenie výsledkov, odovzdanie testovacej dokumentácie

Vstup: otvorená dokumentácia testovania

Výstup: dokumentácia testovania

Zodpovedná osoba: projektový manažér, manažér testovania, vedúci testovacieho tímu

3.3.2. Proces vytvorenia stratégie testovania

Popis:	výber správnej stratégie pre testovanie, pri výbere sa sledujú najmä ciele, ktoré má testovanie naplniť a charakter projektu	
Iniciátor:	manažér testovania	
Vstup:	otvorená dokumentácia testovania	
Výstup:	stratégia testovania, nástroj testovania, mílniky testovania, aktualizovaná dokumentácia testovania	
Koniec:	vytvorenie a schválenie stratégie testovania	
Krok	Popis	Kapitola
1.	Výber stratégie testovania.	3.3.2.1
2.	Výber testovacieho nástroja.	3.3.2.2
3.	Vytýčenie mílnikov testovania.	3.3.2.3
4.	Zahájenie procesu tvorby scenárov testovania.	3.3.2.4

3.3.2.1. Výber stratégie testovania

Popis: výber správnej stratégie testovania, výber sa opiera o stanovené ciele testovania

Vstup: otvorená dokumentácia testovania
Výstup: aktualizovaná dokumentácia testovania
Zodpovedná osoba: vedúci tímu, manažér testovania

3.3.2.2. Výber testovacieho nástroja

Popis: výber testovacieho nástroja ovplyvnení stratégiou testovania
Vstup: otvorená dokumentácia testovania
Výstup: určený nástroj testovania, aktualizovaná dokumentácia testovania
Zodpovedná osoba: vedúci tímu, tester

3.3.2.3. Vytýčenie míľnikov testovania

Popis: na základe určenej stratégie sa určia míľniky pre kontrolu testovacieho tímu
Vstup: otvorená dokumentácia testovania
Výstup: aktualizovaná dokumentácia testovania
Zodpovedná osoba: vedúci tímu, manažér testovania

3.3.2.4. Zahájenie procesu tvorby scenárov testovania

Popis: oboznámenie testovacieho tímu so stratégiou testovania
Vstup: otvorená dokumentácia testovania
Výstup: proces tvorby scenárov testovania
Zodpovedná osoba: vedúci tímu, tester

3.3.3. Proces vytvorenia testovacích scenárov

Popis:	vytvorenie testovacích scenárov, určenie rozsahu pokrytia testovaného diela, oboznámenie testerov s cieľmi testovania, pridelenie úloh	
Iniciátor:	vedúci tímu	
Vstup:	otvorená dokumentácia testovania	
Výstup:	scenár testovania, aktualizovaná dokumentácia testovania	
Koniec:	implementované a schválené scenáre testovania	
Krok	Popis	Kapitola
1.	Analýza pokrytia diela testom.	3.3.3.1
2.	Implementácia testovacieho scenára.	3.3.3.2
3.	Zahájenie procesu vykonávania scenárov testov.	3.3.3.3

3.3.3.1. Analýza pokrytia diela testom

Popis: oboznámenie sa so stratégiou testovania a stanovenie rozsahu scenárov pre potreby pokrytia
Vstup: otvorená dokumentácia testovania
Výstup: aktualizovaná dokumentácia testovania
Zodpovedná osoba: vedúci tímu, tester

3.3.3.2. Implementácia testovacieho scenára

Popis:	naprogramovanie pridelenej časti testovacieho scenára
Vstup:	otvorená dokumentácia testovania
Výstup:	aktualizovaná dokumentácia testovania, implementované scenáre
Zodpovedná osoba:	tester

3.3.3.3. Zahájenie procesu vykonávania scenárov testov

Popis:	skompletizovanie a kontrola testovacích scenárov
Vstup:	otvorená dokumentácia testovania, implementované scenáre
Výstup:	proces testovania, aktualizovaná dokumentácia testovania
Zodpovedná osoba:	vedúci tímu, tester

3.3.4. Proces vykonania testovacích scenárov

Popis:	samotné vykonanie testu, konfigurácia testovacieho prostredia pre potreby testu	
Iniciátor:	vedúci tímu	
Vstup:	otvorená dokumentácia testovania, implementovaný testovací scenár	
Výstup:	vytvorený/vygenerovaný report, aktualizovaná dokumentácia testovania	
Koniec:	zálohovanie vygenerovaných reportov	
Krok	Popis	Kapitola
1.	Konfigurácia testovacieho nástroja.	3.3.4.1
2.	Spustenie vykonania testovacích scenárov.	3.3.4.2
3.	Zber vygenerovaných reportov.	3.3.4.3
4.	Zahájenie procesu reportovania chýb.	3.3.4.4

3.3.4.1. Konfigurácia testovacieho nástroja

Popis:	nastavenie nástroja pre automatizáciu testov pre požiadavky testu
Vstup:	otvorená dokumentácia testovania, manuál k nástroju testovania
Výstup:	nakonfigurovaný nástroj
Zodpovedná osoba:	tester

3.3.4.2. Spustenie vykonania testovacích scenárov

Popis:	spustenie samotného nástroja, ktorý vykoná test
Vstup:	implementovaný testovací scenár
Výstup:	vygenerovaný report
Zodpovedná osoba:	tester

3.3.4.3. Zber vygenerovaných reportov

Popis:	uloženie vygenerovaných reportov na určené úložisko
---------------	---

Vstup: vygenerovaný report
Výstup: uložený report
Zodpovedná osoba: tester

3.3.4.4. Zahájenie procesu reportovania chýb

Popis: začatie procesu reportovania chýb podľa metodiky reportovania chýb
Vstup: vygenerovaný report
Výstup: proces reportovania
Zodpovedná osoba: tester

3.3.5. Proces reportovania chýb

Popis:	reportovanie odhalených chýb do nástroja na to určeného podľa metodiky pre report chýb	
Iniciátor:	tester	
Vstup:	otvorená dokumentácia testovania, vygenerovaný report	
Výstup:	popis chýb v nástroji, aktualizovaná dokumentácia testovania	
Koniec:	reportované všetky objavené chyby	
Krok	Popis	Kapitola
1.	Analýza vygenerovaného reportu.	3.3.5.1
2.	Zadávanie chýb do nástroja pre reportovanie.	3.3.5.2
3.	Uzavretie procesu testovania.	3.3.5.3

3.3.5.1. Analýza vygenerovaného reportu

Popis: výber a kategorizácia nájdených chýb
Vstup: vygenerovaný report
Výstup: analýza reportu
Zodpovedná osoba: tester

3.3.5.2. Zadávanie chýb do nástroja pre reportovanie

Popis: vytvorenie novej chyby v nástroji a vyplnenie jej popisu
Vstup: analýza reportu, manuál reportovania chýb
Výstup: zadané chyby v nástroji
Zodpovedná osoba: tester

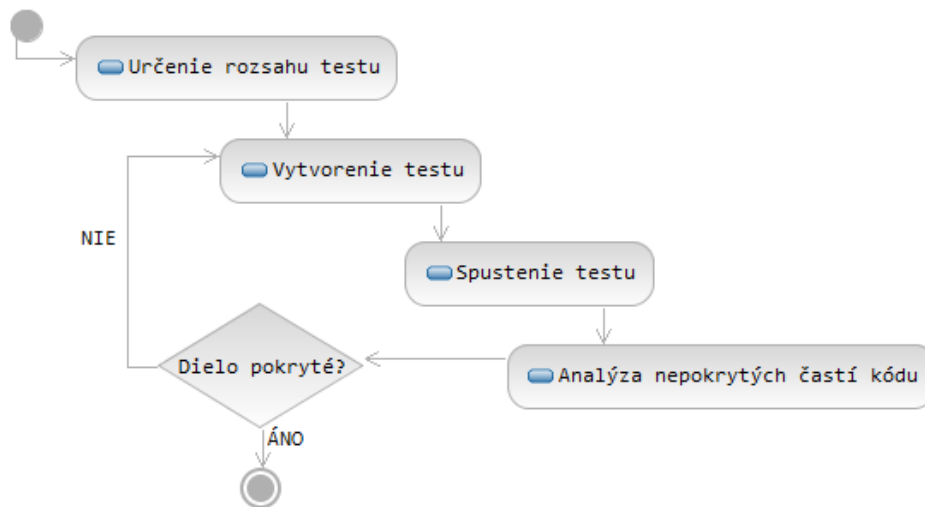
3.3.5.3. Uzavretie procesu testovania

Popis: dokončenie prislúchajúcej časti dokumentácie, oboznámenie s výsledkami nadriadeného
Vstup: otvorená dokumentácia testovania
Výstup: aktualizovaná dokumentácie testovania
Zodpovedná osoba: tester, vedúci testovacieho tímu

4. Vytvorenie jednotkového testu v prostredí PHPUnit

Táto časť metodiky pokrýva proces **vykonania testovacích scenárov** a čiastočne proces **vytvorenia testovacích scenárov**.

Vytvorenie jednotkového testu podľa postupnosti krokov opísaných nižšie možno vyjadriť nasledovným diagramom:



Obr. 2.: Postupnosť krokov vytvorenia jednotkového testu

4.1. Určenie rozsahu testu

Samotný rozsah je definovaný súčtom pokrytia **testovacích scenárov** a testovania **technických detailov**.

To čo je potrebné otestovať použitím jednotkového testu určí v prvotnej fáze prípravy vývojár. Využitím direktív (začínajúcich „@“) okomentuje zdrojový kód podľa manuálu k PHPUnit. Povinnosťou vývojára je označiť direktívami všetky potrebné miesta kódu pre testovanie **technických detailov** testovaného diela.

Ak analýza rozsahu testu pokrýva časti kódu pre ktoré nie sú určené direktívy, tak tester je povinný chýbajúce direktívy doplniť.

4.1.1. Scenáre testovania

Pre testovanie správania sa kódu využívame scenáre testovania. Samotný scenár sa opäť vytvorí pomocou direktív @scenario vložených do zdrojového kódu. Samotný scenár, teda návaznosť akcií musí byť výstupom **analýzy pokrytia diela testom** (viď kapitolu 3.3.3.1). Úlohou testera je túto analýzu pretaviť do implementácie.

4.2. Vytvorenie testu

Kapitola sa zaoberá generovaním testov z pripravených zdrojových kódov a rieši generovanie testu pre správne testovanie zložitých objektov a rozhraní.

4.2.1. Zdanlivé objekty

Pre kontrolu správnosti volania objektov sa využívajú tzv. *mock* objekty. Rozhodnutie použitia *mock* objektov je ponechané na testera. Odporúča sa využiť tieto objekty pri testovaní tried, ktoré sú náročné z hľadiska konfigurácie alebo pri simulácií tried (napr. databázové spojenie). Každé použitie *mock* objektu musí byť zdokumentované v dokumentácii testovania s uvedeným dôvodom použitia.

4.2.2. Testovanie rozhraní

Test rozhrania prebieha pomocou nástroja *Selenium*. Všetky testy rozhrania musia byť napísané a pripravené na použitie práve pre tento nástroj. Všetky testy rozhraní prebiehajú využitím prehliadača *Mozilla Firefox 7*. Prednastavený prehliadač je možné zmeniť s uvedením dôvodu v dokumentácii v konfiguračnom `phpunit.xml` súbore modifikovaním obsahu tagu `<browser>` podľa manuálu k PHPUnit.

4.2.3. Generovanie testu

Vygenerovať samotné jednotkové triedy je možné až po revízií kroku vytvorenia testu. Generovanie sa vykonáva pomocou príkazu:

```
phpunit --skeleton-class %nazov triedy%
```

4.3. Spustenie testu

Kapitola sa zaoberá konfigurovaním testovacieho nástroja, paralelizáciou testov a ich samotným spustením.

4.3.1. Konfigurácia PHPUnit

Pred spustením nástroja PHPUnit treba nakonfigurovať globálne premenné a cesty k iným interným/externým modulom (ak je to potrebné). Nastavenia sa vykonávajú pomocou `bootstrap` prepínača. Pre vzorové príklady nastavení viď PHPUnit manuál.

4.3.2. Paralelizácia testu

Pre urýchlenie testu sa využívame *grid*. Grid sa vytvára použitím direktívy `@group`. Nástroj *Apache Ant* pre automatizáciu testovania vytvára paralelné úlohy, v ktorých sa testujú rôzne takéto grupy. Je úlohou testera aby identifikoval triedy na sebe závislé a tieto zaradil do rovnakej grupy aby zabezpečil integritu vykonávaného testu! Pri testovaní scenárov sa každý scenár zaraďuje do samostatnej grupy.

4.3.3. Automatizácia testu

Pre automatizáciu testu sa používa nástroj *Apache Ant*. Pred spustením samotného vykonania vygenerovaných tried je nutné pridať úlohy do konfiguračného súboru `build.xml` tohto nástroja. Spustenie príkazu `phpunit` s parametrami sa zapisuje podľa nasledovného vzoru:

```
<project name="%nazov%" default="build">

  <target name="phpunit">
    <exec dir="{basedir}" executable="phpunit" failonerror="true">
      <arg line="--log-xml %log% %trieda%" />
    </exec>
  </target>

  <target name="build" depends="clean,prepare,phpunit"/>
</project>
```

Konfiguračný súbor samozrejme dovoľuje nastavenie ciest k logom, spúšťanie gridu ako aj akcie potrebné vykonať pred a po teste. Pre kompletný zoznam možností ponúkaných nástrojmi viď dokumentáciu k *Apache Ant*.

4.4. Analýza nepokrytých častí kódu

Po vygenerovaní testovacieho reportu je tester povinný vykonať analýzu reportu pre odhalenie nepokrytých častí kódu. Analýza slúži pre meranie kompletnosti testu a vykonáva sa použitím *Xdebug* rozšírenia použitím príkazu (formát výstupu je ponechaný na testera):

```
phpunit --report %cesta k reportu% %trieda%
```

Ak sa z vygenerovaného reportu odhalia testom nepokryté časti, tester je povinný rozšíriť rozsah testu o nepokryté časti a vykonať krok testovania odznova.

5. Záver

Metodika vyššej úrovne preberala procesy a osoby zainteresované v týchto procesoch pre naše vnútorné potreby. Takisto metodika nižšej úrovne preberala vytváranie jednotkových testov pre potreby našich procesov. Pre bližšie informácie týkajúcich sa oboch úrovní je potrebné preštudovať príručky a manuály na ktoré sa táto metodika odkazuje.