
Test Plan

for

Metamodel tool

Version 2.0 approved

sirljan

MMN group

6.12.2010

Table of Contents

Table of Contents	- 2 -
Revision History.....	- 2 -
1. Úvod.....	- 3 -
1.1. Účel.....	- 3 -
2. Plán testů	- 3 -
2.1. Testovaný software	- 3 -
2.2. Testovací strategie	- 3 -
2.2.1. Testování jednotek	- 3 -
2.2.2. Integrační testování	- 3 -
2.2.3. Vyšší testy	- 3 -
2.3. Výsledky testů	- 3 -
2.4. Metriky testu	- 4 -
2.5. Testovací nástroje a prostředí	- 4 -
3. Testovací procedury.....	- 4 -
3.1. Testování jednotek.....	- 4 -
3.1.1. Generátoru XML notace	- 4 -
3.1.2. Rozhraní MMN	- 4 -
3.1.3. Jádro programu	- 4 -
3.2. Integrační testování.....	- 4 -
3.3. Vyšší testy	- 5 -
3.3.1. Bezpečnostní testy	- 5 -
3.3.2. Zátěžové a výkonové testy	- 5 -
3.3.3. Alfa/beta testy.....	- 5 -

Revision History

Name	Date	Reason For Changes	Version
sirljan	5.11.2010		1.0
sabolmi1	6.12.2010	changed tasks for tests	2.0

1. Úvod

1.1. Účel

Účelem testů popsaných v tomto dokumentu je zajištění a udržení požadované kvality v průběhu vývoje Metamodelovacího nástroje (MMN). Vývoj testů bude probíhat současně s vývojem aplikace a umožňovat tak odhalování chyb už v průběhu implementace, čímž sníží náklady na pozdější opravy. Výsledkem každé provedené testování bude zpráva zaznamenávající odhalené chyby, které je nutné opravit.

2. Plán testů

2.1. Testovaný software

Testovány budou jednotlivé složky Metamodelovacího nástroje, tedy funkčnost generátoru xml notace, rozhraní MMN, rozhraní mezi MMN a Notáčním pluginem, hlavního jádra programu, která bude zajišťovat vkládání objektů a konexí, a funkčnost jednotlivých tlačítek a nabídek a jejich zpracování.

V dalších etapách bude prověřena bezchybná integrace těchto částí a její vlastností: reakce na chyby, zátěž a použitelnost v konfrontaci s uživatelem.

2.2. Testovací strategie

2.2.1. Testování jednotek

První částí testování, která položí základy pro vývoj funkčního a kvalitního celku bude testování jednotek. Testován bude generátor xml, metody třídy Shape, Work Space (ukládání notace, načítání notace), Metamodel, Connection, funkčnost tlačítek GUI.

2.2.2. Integrační testování

Integrace bude probíhat převážně shora dolů a do kostry budou postupně zapojovány jednotlivé metody.

2.2.3. Vyšší testy

Součástí vyšších testů budou testy zotavení z poruch, testy bezpečnosti, testy zátěžové a výkonové. Celé testování pak uzavře série alfa a beta testů.

2.3. Výsledky testů

Výsledkem testů budou zprávy o provedených testech s výsledky. Uvedeno vždy bude, zda kvalita odpovídá požadavkům, a pokud ne, jak dalece se jim vzdaluje. Budou zaznamenány činnosti vedoucí k nápravě takto zjištěných rozporů a po jejich provedení budou testy zopakovány. V případě, že budou nalezeny chyby závažnějšího charakteru, bude připravena nová série testů, aby se zamezilo "cíleným" opravám chyb pro konkrétní případy.

2.4. Metriky testu

Absolutní většina testů je rozhodovacích (testovaná jednotka má/nemá požadovanou kvalitu). Jedná se o všechny testy jednotkové a integrační a dále o testy zotavení z chyb.

2.5. Testovací nástroje a prostředí

Pro ladění programu a zátěžové testy bude využíván Netbeans Profiler a Netbeans Code Coverage Plugin.

3. Testovací procedury

3.1. Testování jednotek

3.1.1. Generátoru XML notace

Umožňuje vytvářet XML popisující pravidla na základě vymodelovaného vzoru. Testování těchto úseků kódu bude provedeno metodou White i Black Box.

3.1.2. Rozhraní MMN

MMN má rozhraní s Base, který spravuje přístup všech modulů do souborového systému a paměti. Dále spravuje projektovou strukturu (tzn., které modely jsou v daném projektu). V případě, že si uživatel vybere model, tak Base zjistí, o jakou notaci modelu jde a předá data z paměti o tomto modelu příslušnému pluginu, který má notaci na starosti.

Base je modul kolegy mimo vývojovou skupinu MMN, musí se tedy pečlivě otestovat, jestli interface splňuje potřebné požadavky. MMN má rozhraní s Notáčním pluginem (NP). NP umožňuje tvorbu modelů v konkrétní notaci. Využívá MMN při vykreslování objektů. Využívá XML z MMN, aby si ověřil, co jsou validní konstrukce v modelu. Pro ukládání svých dat používá base. Testovat se bude správné předání XML notace NP.

3.1.3. Jádro programu

Hlavní kostra programu bude zajišťovat vkládání tvarů, spojení mezi nimi a změny tributů tvarů. Testovány budou postupně všechny uložené procedury a bude sledováno, zda jejich volání přináší očekávané změny v modelu. Poprvé se zde bude testovat také GUI.

3.2. Integrační testování

Díky návrhu vývoje jednotlivých částí aplikace, by měla integrace proběhnout poměrně bezbolestně a měla by se sestávat z postupné implementace jednotlivých metod. Po každé náhradě se provede jednoduchá kontrola, zda se změny skutečně promítly do aplikace a zda se objevil nový tvar nebo spojení, či atribut. Teprve

po úplné integraci se provede intenzivní Black Box testování a testování GUI.

3.3. Vyšší testy

3.3.1. Bezpečnostní testy

Úkolem těchto testů bude ověřit, že Notáční pluginy mají přístup skutečně jen k těm metodám, na které mají oprávnění.

3.3.2. Zátěžové a výkonové testy

Zátěžové testy budou mít za úkol ověřit, že v naší aplikaci nedojde při vysoké zátěži k chybám, které se jinak nevyskytují. Zátěžový test se bude provádět vytvořením rozlehlého modelu. Při změnách v něm se bude měřit doba provedení úkonu(ů).

3.3.3. Alfa/beta testy

Při alfa testingu budou vybrání potencionální uživatelé, kteří budou používat Metamodelář k operacím, které běžně provádějí. Pozorně sledováno bude zejména uživatelské rozhraní, jeho účelnost a snadnost jeho používání. Zároveň je to příležitost odhalit další chyby, které přežily intenzivní testování v předchozích etapách a nadále se skrývají v produktu.

Beta testing bude již probíhat u klienta.