

Plán Testů

I. Úvod

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. Celkový pohled na proces řízení kvality je popsán v dokumentu "Plán zajištění jakosti".

2. Vymezení rozsahu

Testy zde popsané se omezují na zajištění kvality a odpovídající funkčnosti vyvíjeného kódu. Nezabýváme se problematikou analýzy a návrhu, která je řešena jinými postupy (viz "Plán zajištění jakosti")

II. Plán testů

1. Testovaný software

Testovány budou jednotlivé složky Metamodelovacího nástroje, tedy funkčnost generátoru xml notace, rozhraní mezi MMN a Base, rozhraní mezi MMN a Notacním pluginem, pomocných knihoven, hlavního jádra programu, která bude zajišťovat ukládání 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. Testovací strategie

a. *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.

b. *Integrační testování*

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

c. *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.

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.

4. Uchovávání výsledku

Výsledky jednotlivých testů budou udržovány na Trac SVN v Other/Test_Results.

Nedílnou součástí dokumentace testování bude "Záznam o testech", ke kterému bude po každém provedeném testu připojen krátký záznam obsahující datum, charakteristiku daného testu a stručný přehled získaných výsledků. Tento dokument bude používán pro přehled o chronologické posloupnosti provádění testu.

5. 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.

Výsledkem testu bezpečnosti bude zpráva o bezpečnostních slabinách systému, která se stane podkladem pro finální opravy řízení přístupů k metodám a proměnným.

Výsledkem zátěžových testů bude opět informace o tom, zda se systém chová korektně i při nadměrné zátěži.

6. Testovací nástroje a prostředí

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

7. Harmonogram testů

Harmonogram testů je k dispozici jako samostatný dokument pod názvem "Harmonogram řízení jakosti".

III. **Testovací procedury**

1. *Testování jednotek*

a. *Generátoru XML notace*

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

b. *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.

c. *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.

2. *Integrační testování*

Díky návrhu vývoje jednotlivých částí aplikace, jak je uveden výše a v "Harmonogramu řízení jakosti", 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. *Vyšší testy*

a. *Testy zotavení*

MMN umožňuje meta-modelovat, takže při náhlém výpadku proudu nehrozí ztracení cenných dat.

b. *Bezpečnostní testy*

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

c. *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).

d. *Alfa/beta testy*

Při alfa testingu budou pozváni vybraní zástupci jednotlivých rolí a ve spolupráci s členy vývojového týmu 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. Připraveny budou pokyny, jak hlásit objevené chyby a závady, a k dispozici budou formuláře (případně elektronické) pro popisy chyb.