

### **Nejvýznamnější klady:**

Autor prokázal základní znalost jazyka C# a testovacího nástroje Nunit.

### **Nejzávažnější nedostatky:**

#### ***Míra splnění zadání***

Předmětem práce bylo vytvořit testovací nástroj zpětně kompatibilní s Nunit 2.2, který bude navíc podporovat testování programů určených pro běh v paralelním prostředí, testování výkonnosti a detailnější analýzu chybových stavů modulů.

Podpora pro testování paralelních programů je nedostatečná, řeší se pouze do jisté míry paralelní spouštění testů, nikoliv ale testování paralelních programů – vůbec se nepodporuje synchronizace. Existující nástroje testování paralelních programů včetně synchronizace přitom řeší (např. pnunit). Navržené a implementované rozšíření, paralelní spouštění nezávislých testů ve více vláknech, je v prostředí .NET triviální a neřeší zadaný problém.

Podpora pro testování změn výkonnosti je nedostatečná – neřeší se opakování a statistické vyhodnocení testů. Navržené a implementované rozšíření, limit na čas běhu testu, je triviální a není vhodné pro automatickou detekci změn výkonnosti.

Detailnější analýza chybových stavů se omezila na podporu vícenásobného spuštění testu s různými parametry. Realizovaná implementace je triviální v prostředí .NET, užitečnost řešení je diskutabilní – parametry není možné generovat dynamicky. V práci není vysvětleno, v čem by detailnější analýza chybových stavů měla spočívat.

Funkcionalita z Nunit 2.2 je plně reimplementována, možnost rozšířit stávající open source implementaci (taktéž v jazyce C#) není vůbec diskutována a implementace nejsou nijak porovnány. Přitom Nunit je navržen tak, aby byl rozšiřitelný, a např. pnunit je rozšíření Nunitu, které řeší spouštění paralelních testů.

Metodika test-driven development je popsána nekriticky, jazykem popularizační literatury, který není vhodný pro akademickou práci. Typická reálná aplikace nástrojů typu Nunit – unit testing – není zmíněna.

Práce samotná nezahrnuje žádné vyhodnocení dosažených výsledků vzhledem k zadání.

Srovnání s konkurenčními nástroji nezahrnuje relevantní nástroje vzhledem k zadání (např. pnunit pro paralelní testování, rtjunit pro měření výkonnosti).

#### ***Struktura textu práce***

Cíle práce nejsou dostatečně popsány ani analyzovány – povrchně jsou zmíněny v úvodu, podrobnosti jsou ale patrné až z popisu implementace (např., že cílem testování výkonnosti je detekce změn výkonnosti během vývoje).

Práce neobsahuje vyhodnocení dosažených výsledků. Neobsahuje žádný příklad použití vytvořeného nástroje k otestování reálné aplikace.

Příliš velký prostor je věnován „tutorialu“ pro testování pomocí nástroje Nunit, jedná se o triviální problematiku, navíc lze snadno odkázat na existující dokumentaci.

## ***Analýza***

Analýza se nezabývá zadanými problémy v dostatečné šířce (testování paralelních programů, testování výkonnosti a detekce změn), omezuje se na nedůležitá rozhodnutí o tom, jaké parametry bude mít který atribut. Analýza nebere v úvahu existující řešení a publikované výsledky. Často jsou do analýzy zahrnuty i zjevně nerealistické varianty (např. parametrizace výkonnosti počítače – dokonce manuální - činitelem, kterým lze upravit výsledky z jiného počítače).

Analýza nedemonstruje pochopení základní problematiky testování výkonnosti a testování paralelních programů.

## ***Vývojová dokumentace***

Dokumentace zbytečně podrobně popisuje základní fungování nástroje, nezaměřuje se ale na konkrétní třídy a rozhraní. Nezmiňuje rozdíly oproti původní implementaci Nunit 2.2. Rozhraní nejsou dostatečně okomentována ani ve zdrojovém kódu.

## ***Uživatelská dokumentace***

Dokumentace se nezaměřuje na testování paralelních programů ani na testování výkonnosti, pouze na popis uživatelského rozhraní.

## ***Jazyková a typografická úroveň***

Práce obsahuje gramatické chyby (shoda přísudku s podmětem), hovorové obraty a větší množství překlepů.

	<b>výborně</b>	<b>velmi dobře</b>	<b>dobře</b>	<b>neprospěl/a</b>
<b>Návrh známky</b>				<b>x</b>

Datum: 13. června 2007

Podpis: