Posudek oponenta bakalářské práce

Student: Janeček Vít
Téma: Exportování obsahu webového provozu do MAFF (id 18770)
Oponent: Pluskal Jan, Ing., UIFS FIT VUT

1. Náročnost zadání **obtížnější zadání**

Obtížnost zadání není definována zadáním jako takovým, ale hloubkou jeho vypracování. Programová část je zpracována velmi detailně s výrazným přínosem studenta, proto zadání hodnotím jako obtížné.

2. Splnění požadavků zadání **zadání splněno s podstatným rozšířením**

Student vypracoval heuristiku, která zajistí zobrazení aproximací (snapshotů) webových stránek měnících se v čase. Jedná se o zcela nový přístup ke zpracování, který nemá obdoby v již existujících nástrojích.

3. Rozsah technické zprávy **je v obvyklém rozmezí**

Práce je psána v husté LaTeXové šablonce v obvyklém rozsahu.

4. Prezentační úroveň předložené práce **65 b. (D)**

Členění technické zprávy do kapitol a sekcí je logické a uspořádané. Nekonzistenci pouze vidím v obsahu jednotlivých sekcí vzhledem k jejich nadpisu. Student nedodržuje jednotnou hloubku abstrakce a střídá vágní popis problému s velmi detailním. Nejhůře čitelnou je kapitola 3 - "Návrh a implementace ...", ze které je vidět, že student ví, co chce říct, ale nenachází patřičný aparát k vyjádření.

5. Formální úprava technické zprávy **75 b. (C)**

Díky LaTeXové šablonce je práce typograficky korektní až na drobnosti jako je malé písmeno v nadpise 3.2.5. Výtku mám k nekonzistentní velikosti základních textů v obrázcích vůči hlavnímu textu dokumentu. Dále pak v obr. 3.6 jsou typograficke chyby v popiscich uzlu, a obr. 3.8 v notaci diagramu.

6. Práce s literaturou **80 b. (B)**

Vzhledem k charakteru práce je pochopitelné, že většina z použitých zdrojů bude online. Některé zdroje by bylo vhodnější převést do poznámek pod čarou, např. 1 a 3.

7. Realizační výstup **92 b. (A)**

Samotný kód vykazuje prvky OOP návrhu, ale nejsou dodrženy veškeré konvence jazyka C#, které by zpřehlednily čitelnost kódu. Bohužel, Unit testy úplně chybí. Popis implementace v práci a implementace jako taková se mírně liší, např. ObjectParser není možné, aby byl vytvořen "jádrem", protože se jedná o statickou třídu, a jiné. Pro ujasnění pojmů a zvýšení kvality kódu doporučuji studium knížek C# in Nushell a Clean Code.

8. Využitelnost výsledků

Výsledek práce bude dále využit v nástroji Netfox Detective a samotný modul taktéž vykázán v projektu IGA.

9. Otázky k obhajobě

1. Jakým způsobem probíhá zpracování dynamických webů? Simulujete nějaké akce pomocí analýzy AJAX komunikace?
2. V práci popisujete sadu testů. Proč tyto testy a jejich automatické zpracování není součástí odevzdaného řešení?

10. Souhrnné hodnocení **85 b. velmi dobře (B)**

Studentovi se podařilo vypracovat funkční programové řešení, které výrazně zvyšuje použitelnost síťového forenzního nástroje Netfox Detective. I přes výše zmíněné typografické chyby či nižší čitelnost textu práce navrhuji hodnocení 85B. Student svým vlastním přínosem prokázal pochopení problematiky a implementované řešení rozšířil nad rámec zadání.

Prohlášení: Uděluji VUT v Brně souhlas ke zveřejnění tohoto posudku v listinné i elektronické formě.

V Brně dne: 30. května 2016

....
podpis