



MASARYKOVA UNIVERZITA
FAKULTA INFORMATIKY

Posudek oponenta bakalářské práce

Autor práce: Tomáš Kácel

Název práce: Inteligentní vyhledávač hodnocení knih

Jméno oponenta: Zuzana Nevěřilová

Bakalářská práce Tomáše Kácela se zabývá databázemi knih a jejich subjektivních hodnocení s cílem implementovat inteligentní hledání v hodnoceních. Práce má dvě části: implementační a písemnou.

Na přiloženém DVD se kromě souboru s heslem nacházejí dva soubory s archivy. `Zdrojové kódy.rar` obsahuje zdrojové kódy vyhledávače, `PyMySQL3-0.4.tar.gz` obsahuje knihovnu třetí strany. Projekt má také své webové stránky na http://nlp.fi.muni.cz/projekty/hodnoceni_knih/, odkazy ze stránky projektu jsou většinou nefunkční.

Kód jednotlivých skriptů není dokumentovaný, ale je dobře čitelný. Považovala bych za rozumné, kdyby adresy URL databází knih i knihoven byly definované na začátku skriptu jako konstanty, program by byl lépe udržitelný. Rovněž je potřeba, aby program před vydáním prošel jazykovou korekturou. Chyby jako „vypíše výsledky ze serveru uživatele“ (knihovna.cgi) nebo „dle nejbližší schody“ (vyhledane.cgi) jsou autorovi pro ostudu. Textové popisy v kódu jsou česky (což není úplně obvyklé) i anglicky (opět s mnoha chybami), proč ne jednotně?

Vyhledávač hodnocení jsem vyzkoušela na stránkách projektu. Stránky, které program vrací, jsou výpisem odpovědí různých serverů na tentýž dotaz, inteligentním vyhledáváním bych výsledek nenazvala. Naštěstí je možné výsledky hledání seřadit podle hodnocení, takže se uživatel nakonec k „inteligentnímu“ hledání dobere. Výsledky hledání jsou poměrně nepřehledné, myslím, že aby byly přehlednější, bylo by potřeba zapracovat sloučení jednotlivých polí, která mají stejný význam, z různých serverů do jednoho pole. Zatímco v zadání je kladen důraz na „inteligentní hledání hodnocení“, v implementaci se s hodnocením nijak promyšleně nepracuje, je to zkrátka jen jedno z polí pro seřazení výsledků. Program tedy nedokáže „inteligentně“ odpovědět na požadavek „najdi mi dobrou knížku o rybách“.

Aplikace je webová, tj. implementována pomocí skriptů v Pythonu, Common Gateway Interface a relační databáze MySQL. Výsledky hledání jsou webové stránky, které jsou validní HTML 5.

Písemnou část práce tvoří text o šesti kapitolách a 45 stranách (bez příloh).

První kapitola je úvodní, kde autor vysvětluje cíle práce.

Druhá kapitola popisuje weby, které obsahují hodnocení knih. Autor vybral ty nejpožívanější v ČR. Sekce 2.1 Analýza cbdb.cz obsahuje tvrzení, že „vyhledávání se nachází v pravém horním rohu“, které je naprosto irelevantní pro obvyklý způsob zjišťování obsahu

databáze (požadavek HTTP). Naopak neobsahuje všechny informace důležité pro obvyklý způsob dolování dat, např. jména jednotlivých polí, text obsahuje jen obsah jednotlivých polí, přitom autor musel s jednotlivými poli pracovat. Podobně i ostatní dvě „analýzy“.

Kapitola 3 se věnuje popisu knihovních katalogů Aleph a Carmen.

V kap. 4 autor uvádí, že zařadit knihu do určité kategorie či žánru je nemožné a dokazuje svoje tvrzení na knize K. Čapka Krakatit, která je v každé knižní databázi zařazena jinak. Za rozumnější řešení než prostě tvrdit, že zařazení není možné, považují zařazení do kategorií implementovat formou příznaků (tags).

Autor se v sekci 4.2 krátce věnoval aplikacím v oblasti vyhledávání hodnocení knih. Proč nepadla zmínka o Goodreads? Tento server považují za velmi známý, obsáhlý a inspirovatelný zdroj hodnocení knih.

Část 4.4 obsáhle pojednává o návrhu grafického uživatelského rozhraní (GUI), ve skutečnosti je implementováno jen to nejnútnejší. V části 4.4 autor popisuje scénáře tj. případy užití. Podle mého názoru je ve scénářích kladen důraz na data o knihách, ale ne na to, co bylo v zadání tj. hodnocení. Nejvíce se zadání blíží třetí scénář.

Pátá kapitola se zabývá implementací systému. Překvapující je, že zatímco v návrhu autor popisuje zvýrazněný nadpis a jemnou vodorovnou čáru pro oddělení nadpisu a dalšího obsahu (vizuální prvky), v páté části přiznává, že součástí implementace nejsou kaskádové styly (CSS). Myslím, že použití CSS případně Javascriptu by přehlednost (a tudíž použitelnost) aplikace velmi zvýšilo. V 5.3 je popis skriptů a tříd, v další sekci (také 5.3) je popis vytvoření databáze v MySQL. V sekci 5.4 autor popisuje, jak by bylo možné do práce přidat další databázi hodnocení knih, v 5.5 popisuje průchody reálným systémem podle scénářů popsanych v 4.4.

Šestá kapitola je závěrečná.

V textu je zpočátku poměrně málo pravopisných chyb, což hodnotím kladně. Později však počet chyb na stránku narůstá – občas se vyskytují nadbytečné čárky ve větách (např. 1. věta na str. 18), překlapy („vyhledávání výsledů“ na str. 20), chybějící čárky ve větách (např. „Jediný atribut, který musí zadavatel zadat povinně je jméno knihy a jejího autora.“ na str. 22), syntaktické chyby („tyto data“ na str. 21, „identifikovat uživatele“ na str. 23) a dokonce i hrubky („neznámí autor“ na str. 18, „data o autorovy“ na str. 31, „standartní knihovna“ na str. 35).

Práce obsahuje mnoho stylistických neobratností, které snižují její srozumitelnost a které je možné tréninkem vymýtit: „dle“ místo „podle“, příliš mnoho ukazovacích zájmen „Z těchto důvodů nebyly brány tyto zdroje jako relevantní.“ (str. 9), „více jak“ místo „více než“, používání slangových termínů (např. frameworky, tagy na str. 30).

Seznam literatury obsahuje 31 položek. Toto úctyhodné číslo je však třeba snížit o položky jako „Vyhledání výrazu krakatit“. Skutečně relevantních položek literatury, se kterou student pracoval, jsem napočítala méně než polovinu. Řazení bibliografických záznamů je naprosto nejasné. V zápisech jsou vážné chyby, např. autory [22] nejsou Slaimon Lewis a John Rielman, ale Clayton Lewis a John Rieman.

Jak plyne z výše uvedeného, práce splňuje požadavky na rozsah i obsah, ale s vážnými výhradami. Implementace je také funkční, ne však skutečně použitelná.

Práci doporučuji k obhajobě a navrhuji známku E.

4. června 2012
Zuzana Nevěřilová