

Aplikácia pre vyhodnocovanie a korekciu výsledkov analýzy textových recenzií

Bakalárska práca

Matúš Ďurík

Vedúci práce:
Ing. Jan Přichystal, Ph.D.

Brno 2020

1 Úvod a cieľ práce

1.1 Úvod

MTA (Multicriteal Text Analysis) sa zaoberá spracovaním recenzií. Recenzie sa spracovávajú a ukladajú do relačnej databázy. Nanešťastie algoritmy, ktorými sa toto spracovanie vykonáva nie sú vždy 100 percentné, a preto dáta uložené v relačnej databáze nemusia byť vždy korektné. Pre zvýšenie korekcie dát je potrebné tieto dáta skontrolovať, a v prípade nesprávnosti opraviť. Preto bude vytvorená webová aplikácia, ktorá tieto dáta bude svojim užívateľom zobrazovať, a tí budú mať možnosť potvrdiť správnosť dát alebo navrhnúť zmeny. Dané zmeny budú musieť byť odsúhlasené kontom administrátora. Po schválení návrhu sa následne tieto navrhované zmeny prevedú do relačnej databázy.

1.2 Cieľ práce

Cieľom práce je vytvorenie aplikácie, vďaka ktorej budú mať užívatelia prístup k dátam, budú môcť potvrdzovať správnosť dát, a v prípade nesprávnosti navrhnúť zmenu – tým sa zvýši presnosť. Aplikácia bude pre bežného užívateľa obsahovať výpis priradených recenzií na spracovanie a evidenciu skontrolovaných recenzií. Administrátorské konto bude navyše obsahovať zložku s návrhmi od užívateľov a správu kont.

2 Rešerš

Aplikácia ktorá by spĺňala požiadavky sa momentálne na trhu nenachádza. Je to spôsobené tým, že požiadavky sú veľmi špecifické – aplikáciu je potrebné vyrobiť na mieru.

2.1 Súčasné webové aplikácie

Ako bolo už vyššie spomenuté, webová aplikácia, ktorá by vyhovovala stanoveným podmienkam neexistuje, takže je potrebné zanalyzovať webové aplikácie všeobecne. Budú skúmané riešenia zo záverečných prác, ktoré pojednávali o tejto problematike.

Ako prvá bude preskúmaná bakalárska práca od HOJNÉHO (2015), ktorá sa zaoberala vytvorením webovej aplikácie pre správu skúšobnej agendy. Práca bola tvorená ako náhrada za už existujúci systém. Medzi podmienky, ktoré boli kladené, patrila jednoduchosť, spoľahlivosť, príjemný vzhľad, či rezponzívny dizajn.

Na začiatku práce sú porovnávané programovacie jazyky vhodné pre vývoj webových aplikácií. Voľba padla na Python, a to z dôvodu jeho vysokej prehľadnosti, stručnosti a jednoduchosti voči iným jazykom. Taktiež je možná aj viacnásobná dedičnosť objektov. Pre jednoduchší vývoj aplikácií je veľmi vhodné použiť nejaký framework. Dajú sa rozdeliť na veľké, mohutné frameworky, ktoré obsahujú mnoho knižníc a pripojených modulov, napríklad Django, TurboGears a mnoho iných. Na druhej strane existujú aj mikroframeworky, do ktorých je potrebné doinštalovať moduly poprípade knižnice, ktoré sú pre danú prácu nevyhnutné. Medzi najznámejšie patria Bottle, CherryPy, Flask a ďalšie (PYTHON, 2020).

V rámci tejto práce bude využitý mikroframework Flask, ktorého hlavné výhody sú jednoduchosť, vynikajúce spracovanie, možnosť pridávania len potrebných knižníc a v neposlednom rade prehľadná dokumentácia. Pre zobrazovanie stránok využíva Flask šablónovací systém Jinja2. Pre prácu s dátami bol vybraný databázový systém SQLite, ktorý ako uvádza KUMST (2016), nefunguje na princípe klient-server ako MySQL a PostgreSQL, ale dáta získava z uloženého súboru. Nanešťastie Flask neobsahuje nástroj pre prácu s databá-

zou, a preto bolo v práci použité rozšírenie SQLAlchemy. Ako je uvedené na portáli ITNETWORK (2018), toto rozšírenie dovoľuje pracovať s tabuľkami ako s kolekciami objektov.

Pri testovaní aplikácie bola využitá technológia Python Selenium. V aplikácii je potrebné renderovanie PDF dokumentu z práve otvorenej stránky. Táto možnosť bude mať využitie napríklad pri exporte zoznamu študentov prihlásených na daný termín skúšky. Pre tento účel bolo potrebné pripojenie rozšírenia s názvom Flask-WeasyPrint.

Aplikácia bude obsahovať rôzne typy užívateľských kont – študent a učiteľ. Študent si po úspešnej autentifikácii dokáže prehliadať svoje zapísané predmety a taktiež zobrazí termíny skúšok, na ktoré sa môže prihlásiť. Učiteľ bude mať možnosť prehliadať predmety, ktoré učí a taktiež mu bude umožnené pridať nový termín skúšky s určitými parametrami, ako napríklad kde sa skúška nachádza, kedy sa koná a podobne. Ďalej tu bude možnosť spravovania výsledkov študentov z vybraného predmetu.

Pre nadesignovanie stránky bol použitý CSS framework s názvom Twitter-Bootstrap. HOJNÝ (2015) v závere práce uvádza, že aplikácia uspela v testovaní – boli vykonávané záťažové testy. V budúcnosti by bolo možné uvažovať o renderovaní aj iných formátov. Taktiež spomína aj možnosť využitia technológie AJAX, vďaka ktorej by bola odľahčená práca servera od znovu načítania.

HORVÁTHOVÝM (2017) cieľom bolo vytvorenie odlišnej webovej aplikácie, ktorá slúžila pre zber a vydávanie článkov. Systém je primárne určený pre študentov Katedry počítačov a informatiky Technickej univerzity v Košiciach. Aplikácia by mala poskytovať možnosť editácie už existujúceho článku. Tieto články môžu byť ďalej pridávané do rôznych čísiel magazínu a následne publikované. Nasledujúcou požiadavkou bola možnosť komentovať vybraný článok. Aplikácia bude taktiež obsahovať rôzne role užívateľov, pričom každá rola bude mať rôzne práva. Príkladom môže byť Autor, ktorý bude môcť článok vytvoriť, upraviť a komentovať, pričom Redaktor ho bude môcť navyše označiť ako dobrý, poprípade ho zamietnuť. Dôvodom tvorby tejto aplikácie bolo vytvorenie prehľadnejšieho rozhrania pre užívateľa. Predchádzajúce riešenie bolo vytvorené pomocou publikačného systému Wordpress.

HORVÁTH (2017) sa rozhodoval medzi využitím PHP frameworku Laravel, Ruby on Rails alebo Django. Aplikácia bude napísaná v jazyku Python vo verzii 3. Uvádza, že výber na Django padol z dôvodu implementácie ORM, REST API a LDAP. HORVÁTH (2017) zdôvodňuje svoj výber poskytnutím administratívneho rozhrania pre správcu aplikácie, taktiež obsahom ORM modulu a dobrou implementáciou oproti iným frameworkom. Django využíva MVT (model-view-template) architektúru pričom oproti MVC (model-view-controller) sa líši len použitou terminológiou. Dáta, ktoré budú posielané zo serveru na stranu klienta budú vo formáte JSON.

Návrh stránky bol vytvorený v rastrovom editore Photoshop (od firmy Adobe). Na strane klienta bude obsah stránky zobrazovaný pomocou HTML, pričom o vzhľad sa budú starať kaskádové štýly (CSS). Pre zjednodušenie práce sa HORVÁTH (2017) rozhodol využiť CSS framework Foundation. Foundation ale na rozdiel od frameworku Bootstrap neobsahuje dostatočne veľký počet ikon, a preto bol HORVÁTH (2017) nútený použiť aj ikony z Font Awesome. Vďaka využitiu Foundation mu bola uľahčená tvorba responzívnej stránky. Taktiež bola použitá aj knižnica jQuery, ktorá stránku „rozhybe“.

HORVÁTH (2017) sa rozhodol použiť aj frontendový framework a to konkrétne AngularJS. Tento framework zvolil aj napriek tomu, že v čase vývoja aplikácie bola dostupná už aj novšia verzia – Angular2. Ako dôvod svojho výberu uvádza obavy z chybovosti tohto novo vzniknutého frameworku. AngularJS prevažne slúži na vývoj SPA, vďaka využitiu technológie AJAX. Angular využíva tvorenie stránok pomocou komponent.

MOUČKOVÁ (2018) bakalárska práca bola zameraná na vytvorenie webovej aplikácie pre vzdialenú správu systému Fitcrack. Toto spravovanie zahŕňa možnosť pridávania nových úloh, monitorovanie prebiehajúcich úloh, analyzovanie a iné. Predchádzajúce riešenie používalo na strane servera programovací jazyk PHP a na strane klienta trojicu HTML, CSS a JS. Toto riešenie bolo taktiež nerozšíriteľné a veľmi ťažko testovateľné, a preto MOUČKA (2018) začal s vývojom novej webovej aplikácie. Ako uvádza FAKULTA INFORMAČNÍCH TECHNOLOGIÍ VUT (2018), Fitcrack je systém slúžiaci pre lámanie kryptografických heslov. Hlavným cieľom MOUČKU (2018) bolo zabezpečenie aplikácie, ktoré bolo v predchádzajúcej verzii nedostatočné. Taktiež medzi ciele patrilo zníženie záťaže, ktorá prebieha na strane servera.

MOUČKOVÝM (2018) plánom bolo využitie architektúry REST. Aplikácia na strane klienta mala byť zobrazovaná vo forme SPA, ktorá so stranou servera komunikuje pomocou technológie AJAX s využitím frameworku Axios. Architektúra REST bola vybraná z dôvodu možnosti zabezpečenia práce klienta s databázou – čítanie, editácia a mazanie.

Zvoleným programovacím jazykom bol Python vo verzii 3. Pre vývoj aplikácie bol zvolený mikroframework Flask. Nakoľko framework Flask neobsahuje v základe databázovú vrstvu bolo pridané rozšírenie SQLAlchemy, ktoré mu túto funkcionality dodalo. Taktiež bolo potrebné pridať rozšírenie Flask-RESTPlus. Ďalej bolo pridané aj rozšírenie Flask-Login, ktoré pomáha s prihlasovaním a odhlasovaním užívateľov. Ako frontendový framework zvolil MOUČKA (2018) Vue kvôli jeho jednoduchosti. Vo Vue je nutné vytváranie komponent, ktoré je možné opakovane využívať. Údaje zo strany servera budú posielané na stranu klienta vo formáte JSON. MOUČKA (2018) využil abstraktnú vrstvu pre lepšiu ochranu databázy pred potenciálnymi útokmi. Voľba databázového systému padla na MySQL z toho dôvodu, že bol používaný aj v predchádzajúcej verzii aplikácie Fitcrack.

Návrh stránky MOUČKA (2018) designoval tak, aby bol pútavý a štruktúrovaný, pričom neodmysliteľnou súčasťou bola responzivnosť. Vo webovej aplikácii sú vykresľované z dát grafy, ktoré sú vďaka asynchrónnej komunikácii menené okamžite. Pre vykresľovanie týchto grafov je využívaná javascriptová knižnica `Morris.js`.

Po úspešnom ukončení vývoja aplikácie nasledovalo testovanie funkcionality systému. Pri tomto testovaní, boli objavené drobné chyby s oprávneniami užívateľov, ktoré boli okamžite odstránené. Ďalej bolo vykonané aj užívateľské testovanie, ktorého ktorého sa zúčastnilo niekoľko osôb. Tie skúmali mieru prehľadnosti a intuitívneho využitia prostredia.

Literatúra

- FAKULTA INFORMAČNÍCH TECHNOLOGIÍ VUT. Detail produktu Fitcrack 2.0 [on-line]. In *Detail Produktu*. 2018 [cit. 2020-01-25]. Dostupné na: <https://www.fit.vut.cz/research/product/583/.cs>.
- HOJNÝ T. *Systém (webová aplikácia) pre spravovanie skúšobnej agendy*. (Bakalárska práca.) Žilina : Žilinská univerzita v Žiline, 2015. 42s.
- HORVÁTH V. *Webová aplikácia pre zber a vydávanie článkov od študentov*. (Bakalárska práca.) Košice : Technická univerzita v Košiciach, 2017. 39s.
- ITNETWORK. SQLAlchemy a registrace uživatelů ve Flask frameworku [on-line] In *Článok o SQLAlchemy*. 2018 [cit. 2020-01-25]. Dostupné na: <https://www.itnetwork.cz/python/flask/sqlalchemy-a-registrace-uzivatelu-ve-flask-frameworku>.
- KUMST M. Seznámení s SQLite [on-line]. In *Článok o SQLite*. 2016 [cit. 2020-01-25] Dostupné na: <https://blog.root.cz/maertienuv-obcasny-blog/seznameni-s-sqlite/>.
- MÚČKA M. *Webová aplikácia na vzdialenú správu systému Fitcrack*. (Bakalárska práca.) Brno : Vysoké učení technické v Brně, 2018. 56s.
- PYTHON. Web Frameworks for Python [on-line]. In *Wiki Python*. 2020 [cit. 2020-01-25]. Dostupné na: <https://wiki.python.org/moin/WebFrameworks>.