

SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE
FAKULTA INFORMATIKY A INFORMAČNÝCH TECHNOLOGIÍ

Tvorba testov s využitím L^AT_EXu

Ponuka tímu č. 11

Predmet: Tvorba softvérového/informačného systému v tíme

Vypracovali:

Bc. Rastislav Barlík (odbor informačné systémy)

Bc. Štefan Dlugolinský (odbor informačné systémy)

Bc. Michal Kvetan (odbor softvérové inžinierstvo)

Bc. Stanislava Leitmanová (odbor softvérové inžinierstvo)

Bc. Milan Šillík (odbor softvérové inžinierstvo)

Kontakt: jedenast@gmail.com

Dátum: 30. 9. 2007

1 Znenie zadania

Súčasťou takmer každého univerzitného predmetu sú testy ako jeden zo spôsobov overenia vedomosti študentov. Tieto testy často obsahujú otázky s ponúknutými odpoveďami, na ktoré sa odpovedá výberom. Sú dva typy otázok s ponúknutými odpoveďami: otázky s jednou správnou odpoveďou a otázky s viacerými správnymi odpoveďami. Bodovanie závisí od typu otázky a od existencie trestných bodov za zlú odpoveď.

Odhľadnúc od zložitosti tvorby samotných otázok, náročnou úlohou pri tvorbe testov s ponúknutými odpoveďami, ktorá by sa dala automatizovať, je preusporadúvanie otázok a odpovedí na ne za účelom obmedzenia možností odpisovania pri testovaní väčšieho počtu študentov súčasne. Pri preusporadúvaní otázok a odpovedí na ne vznikajú rôzne kľúče pre hodnotenie.

Ak prepokladáme existenciu bázy otázok, testy by bolo možné zostavovať automaticky z náhodne vybraných otázok. Otázky pritom majú rozdielnu hodnotu a patria do rôznych okruhov, a väčšinou je potrebné, aby v teste boli zastúpené rôzne okruhy v stanovenom pomere (vzhľadom na hodnotu otázok).

Testy s ponúknutými odpoveďami je najlepšie realizovať na počítačoch. Niekedy to však nie je možné (napr. pre veľký počet študentov), takže zostáva potreba vytlačenia otázok. Za týmto účelom je vhodné použiť \LaTeX . \LaTeX umožňuje sadzbu dokumentov, ktoré vyzerajú profesionálne a zároveň v mnohom poskytuje flexibilitu programovacieho jazyka potrebnú pre zostavenie testu.

Analyzujte problematiku tvorby testov, ktoré obsahujú otázky s ponúknutými odpoveďami. Navrhните systém na tvorbu takýchto testov, ktorý pomôže pri načrtnutých problémoch. Umožnite generovanie testu v \LaTeX u. Navrhnutý systém implementujte.

2 Úvod

Predkladaný dokument obsahuje ponuku tímu č.11 pre tému s názvom „Tvorba testov s využitím L^AT_EXu“ a je súčasťou dokumentácie tímového projektu v predmete *Tvorba softvérového systému v tíme*. Cieľom dokumentu je poskytnúť potrebné informácie o členoch tímu, ich odborných skúsenostiach súvisiacich s danou problematikou, dôvodoch prečo máme záujem o vybranú tému. Súčasťou dokumentu je taktiež stručný súhrn poznatkov týkajúcich sa danej témy a základný návrh vylepšení, prípadne nových funkcionalít, ktoré by sme chceli do systému implementovať.

V kapitole 3 predstavujeme náš tím, naše skúsenosti a zručnosti, ktoré využijeme pri implementácii systému, či naše záujmy týkajúce sa oblasti informačných technológií vo všeobecnosti. V kapitole 4 sú zhrnuté dôvody, ktoré nás viedli k výberu tejto témy a pre ktoré sa nám táto téma zdá byť zaujímavá. Kapitola 5 obsahuje informácie o vlastnostiach systému. Navrhujeme v nej spôsoby vylepšenia už existujúcich funkcionalít v systéme a taktiež predkladáme návrh nových funkcionalít, ktoré by sme chceli do systému implementovať. V prílohe prikladáme zoznam tém podľa našich preferencií a rozvrh jednotlivých členov tímu spolu s preferovaným časom vyhradeným pre stretnutia tímu.

3 Predstavenie tímu

Bc. Rastislav Barlík

Absolvoval bakalárske štúdium na Fakulte informatiky a informačných technológií STU v Bratislave, zameranie Informatika. V súčasnosti študuje inžinierske štúdium na rovnakej fakulte, odbor informačné systémy.

Má skúsenosti s programovaním v jazykoch C, C++. Počas štúdia na Fakulte informatiky a informačných technológií sa oboznámil s technológiou Java, ktorú použil aj pri implementácii bakalárskej práce. Počas štúdia nadobudol skúsenosti s objektovo-orientovaným prístupom a tiež získal praktické skúsenosti s databázami MySQL a PostgreSQL.

Popri štúdiu pracuje ako programátor vo vývojárskej firme, kde si osvojil prácu v tíme. Zaujíma sa aj o tvorbu hier a umelú inteligenciu.

Bc. Štefan Dlugolinský

Absolvoval bakalárske štúdium na Fakulte informatiky a informačných technológií STU v Bratislave v odbore Informatika. V súčasnosti je študentom inžinierskeho štúdia v odbore Informačné systémy na rovnakej fakulte.

Má skúsenosti s programovacími jazykmi C a C++. Počas štúdia spolupracoval so softvérovou spoločnosťou Quality Unit, kde nadobudol skúsenosti s vývojom internetových biznis aplikácií a technológiami PHP, MySQL, ako aj riešením prepojenia týchto aplikácií s rôznymi internet bankingovými systémami a internetovými obchodmi. Ďalej tu získal aj bohaté skúsenosti v komunikácii so zahraničnými klientmi a s prácou v tíme. Medzi jeho znalosti možno zaradiť technológie SQL, OpenGL, HTML, CSS, JavaScript, UML a základy assembleru.

Zaujíma sa o bezpečnosť internetových aplikácií, jazyk Java, počítačovú grafiku, televíznu techniku a o grafický dizajn.

Bc. Michal Kvetan

Absolvoval bakalárske štúdium na Fakulte informatiky a informačných technológií STU v Bratislave, odbor Informatika. V súčasnosti je študentom inžinierskeho štúdia na Fakulte informatiky a informačných technológií v Bratislave v študijnom odbore Softvérové inžinier-

stvo. Má skúsenosti s programovacími jazykmi C, C++, Java. Skúsenosti s programovacím jazykom Java využil aj pri implementácii svojej bakalárskej práce. Má základné znalosti aj v oblasti HTML, Assembler, OpenGL. Medzi jeho záujmy patrí práca s grafikou.

Bc. Stanislava Leitmanová

Absolvovala bakalárske štúdium na Fakulte informatiky a informačných technológií v odbore Informatika. V súčasnosti študuje inžinierske štúdium v odbore Softvérové inžinierstvo.

Popri štúdiu pracuje v softvérovej firme, kde sa venuje vývoju a údržbe kompilátorov. V práci získala praktické skúsenosti s programovaním v jazyku C++ a prácou v tíme. Ovláda základy práce s databázami a modelovaním v UML. Medzi jej záujmy patrí oblasť umelej inteligencie, čomu nasvedčuje aj výber predmetov v inžinierskom štúdiu.

Bc. Milan Šillík

Absolvoval bakalárske štúdium na Fakulte informatiky a informačných technológií STU v Bratislave zameranie Informatika. V súčasnosti študuje inžinierske štúdium na rovnakej fakulte, odbor Softvérové inžinierstvo.

Má skúsenosti s programovacím jazykom C, avšak posledné 3 roky preferuje programovací jazyk Java. Počas štúdia nadobudol skúsenosti s týmto programovacím jazykom vo vývojovom prostredí Eclipse, ktoré následne využil aj ako programátor v softvérovej spoločnosti Softec, kde pracuje od r. 2006. Medzi jeho ďalšie znalosti patrí jazyk UML a základná znalosť SQL, HTML, latexu, assembleru a OpenGL.

Medzi jeho prednosti patrí schopnosť pracovať v tíme. Jeho hlavným záujmom je platforma Java. Medzi jeho ďalšie záujmy patrí hlavne práca s multimédiami.

4 Motivácia

Pri veľkom počte študentov vzniká pre učiteľa problém, ako má po prebratom učive vyskúšať všetkých svojich študentov. V tejto situácii je najrozumnejším spôsobom preverenia znalostí študentov test, ktorým sa vyskúšajú všetci študenti naraz. No aj tento spôsob prináša skúšajúcemu učiteľovi niekoľko menších problémov. Medzi tieto problémy patrí najmä zostavenie testu, oprava testu alebo dokonca vytvorenie viacerých verzií testu kvôli žiakom, ktorí medzi sebou opisujú. Čas, ktorý musí pedagóg venovať na riešenie týchto problémov, by mohol byť využitý oveľa racionálnejšie, keby pre neho existovala šikovná pomôcka na tvorbu a správu testov.

Náš tím sa rozhodol riešiť túto tému preto, lebo ako študenti máme k jej problematike veľmi blízko a veľmi dobre ju poznáme. Taktiež sme sa oboznámili s aktuálnou situáciou riešenia tejto témy a máme konkrétnu predstavu riešenia k spokojnosti používateľa. Ďalej máme skúsenosti s jazykom \LaTeX a tie chceme využiť tak, aby používateľ neznalý tohto jazyka nemal najmenší problém s používaním nástroja riešiaceho túto tému. V neposlednom rade je zloženie nášho tímu, kde zjednotenie rôznych poznatkov jeho členov a prienik ich spoločných záujmov určite prispeje k úspešnému riešeniu témy.

Naším cieľom je teda vytvoriť spoľahlivý nástroj, ktorý pre svoju funkčnosť a jednoduchosť používania bude vyhľadávanou pomôckou pedagógov, nielen na našej fakulte, ale aj na iných školách. Veríme, že sa to s našimi doterajšími vedomosťami a skúsenosťami podarí, že sa naučíme nové veci a zdokonalíme v tom, čo vieme.

5 Vlastnosti systému

Rozhodli sme sa, že systém budeme implementovať v programovacom jazyku Java. Dôvod rozhodnutia ovplyvnila hlavne väčšia praktická skúsenosť členov tímu s platformou Java oproti platforme .NET. V tejto kapitole opíšeme vlastnosti navrhovaného systému.

Systém bude mať väčšinu vlastností ako systém minuloročného tímu, po diskusii s jeho tvorcami však pribudne niekoľko vylepšení. V tejto kapitole popíšeme základné vlastnosti systému, ako aj nami navrhované vylepšenia oproti riešeniu minuloročnému tímu.

5.0.1 Databáza otázok a ich generovanie

Keďže systém bude využívať databázu otázok, práve otázky sú dáta, s ktorými bude systém najviac pracovať. Každá otázka má viacero vlastností ako je obtiažnosť, kategória, kľúčové slová, počet správnych odpovedí a podobne. Testy budú vytvárané náhodným generovaním vopred zadaného počtu otázok určitej obtiažnosti (tá bude vopred zadaná, napríklad intervalom obtiažnosti od-do), určitej témy a podľa zadaných kľúčových slov. Umožníme tak variabilnú tvorbu testov, pokiaľ možno čo najviac podľa predstáv používateľa.

Oproti minuloročnému riešeniu však budeme generovanie otázok optimalizovať. Momentálny stav je taký, že generovanie otázok nie je rovnomerné. Napríklad, ak používateľ zadá generovanie testu, ktorý má obsahovať ľahké, stredné a ťažké otázky – systém nezaručí koľko bude ktorých. Môže sa stať, že test bude obsahovať najviac ľahkých a ťažkých najmenej, prípadne opačne. Aby bolo zaručené rovnomernejšie rozdelenie obtiažnosti, a teda objektivnosť testov, implementujeme zlepšené generovanie otázok aby bol zachovaný pomer otázok.

Ďalej chceme vyriešiť problém, aby sa vo vygenerovanom teste dali eliminovať navzájom súvisiace otázky. To by sa dalo dosiahnuť kontrolou kľúčových slov alebo kontrolou kategórií porovnávaných otázok. Jednu otázku by bolo možné zaradiť do viacerých kategórií. Kategórie otázok by si pedagóg vytváral sám podľa vlastného uváženia.

5.0.2 Vkladanie otázok a jednoduchý editor

Ďalšie vylepšenie sa bude týkať vkladania otázok – v existujúcom systéme sa priamo vkladá latex-ovský kód, čo je pomerne nepraktické. Plánujeme vkladanie vylepšiť, aby sa otázky vkladali v jednoduchom editore otázok v plain textovej forme. Ten bude umožňovať niektoré

základné operácie ako tučné písmo, kurzíva a podobne. Systém by mal podporovať aj import už hotových otázok z externých latex súborov.

5.0.3 Generovanie kľúčov

Keďže test bude pravdepodobne obsahovať pomerne veľké množstvo otázok v ráde niekoľko desiatok, štandardné ručné opravovanie by bolo časovo náročné, nevraťiac o pomerne veľkej pravdepodobnosti vzniku chýb. Na vyriešenie tohto problému navrhujeme, aby sa na koniec každého testu pridala súhrnná tabuľka s možnosťami, do ktorej študent zapíše svoje odpovede napríklad krížikom.

Následne by sa ku každému testu vygenerovala totožná tabuľka, ale s už vyplnenými správnymi odpoveďami – tj. kľúč testu. Opravovanie testov sa tak podstatne zjednoduší – namiesto prehliadania testu otázku po otázke a porovnávaní odpovedí so správnymi sa opravovanie zúži na jednoduché porovnanie dvoch tabuliek – testu a kľúča.

5.0.4 Objektívnosť testov

Podobne ako systém predchádzajúceho tímu, aj náš systém bude obsahovať náhodné prehádzanie otázok v teste (odpadá tak možnosť tvorby rôznych skupín, keďže rôzne skupiny navyše nie sú dokonale objektívne).

Jednoduché a náhodné prehádzanie otázok v teste minimalizuje možnosť odpisovania a ušetrí nutnosť tvorby viacerých skupín (a tým pádom zaručí pokiaľ možno čo najväčšiu objektívnosť a v neposlednom rade podstatne predĺži životnosť otázok – ich vyčerpanie nastane neskôr, keďže vždy bude existovať len 1 skupina).

5.0.5 Multimediálne prvky v teste

Systém bude umožňovať vkladať do testov obrázky. Obrázky sa budú do systému vkladať pomocou dialógu, v ktorom si používateľ zvolí súbor kde sa obrázok nachádza a následne sa celý obrázok uloží do databázy. Náš systém bude umožňovať upravovať veľkosť obrázkov a tiež ich meniť ich umiestnenie.

6 Predpokladané zdroje a prostriedky

V tejto časti popíšeme prostriedky, ktoré použijeme ku tvorbe výsledného systému.

6.0.6 Databáza

Ako databázový systém sme zvolili MySQL. Ide o veľmi rýchly voľne šíriteľný relačný databázový systém založený na jazyku SQL, ktorý je navyše jednoduchý na inštaláciu, udržiavanie a každý z nás sa už s ním v praxi stretol.

6.0.7 Aplikácia

Výber prostriedkov sme od začiatku zúžili na dve možnosti a to .NET alebo Java. Vzhľadom na väčšie skúsenosti členov tímu s platformou Java sme sa rozhodli vytvoriť výslednú aplikáciu v prog. jazyku Java. To je aj jeden z dôvodov, prečo chceme vyvíjať aplikáciu od začiatku, a nie pokračovať v už existujúcom .NET riešení. Vývoj predpokladáme na Windows platforme, implementačné prostredie sa zvolí po dohode všetkých členov tímu, predpokladá sa jedno z dvoch najpoužívanejších – Eclipse alebo NetBeans.

6.0.8 Časový plán

Čas riešenia je určený samotným rozvrhom tímového projektu, tj. približne 10 mesiacov. V dole priloženej tabuľke je podrobnejšie znázornený.

Týždeň	Činnosť
1.-3.	Vytvorenie tímu, výber témy, pridelenie témy
3.-7.	Pridelenie úloh členom tímu, analýza témy, špecifikácia požiadaviek, hrubý návrh riešenia, odovzdanie a konzultácia
7.-9.	Prípadná úprava hrubého návrhu, a následný podrobný návrh systému
9.-12.	Dokončenie dokumentácie, implementácia a odovzdanie funkčného prototypu

Príloha 1

Usporiadanie vybraných tém podľa priority členov tímu:

1. Tvorba testov s využitím \LaTeX u (TESTY)
2. Robocup tretí rozmer (RoboCup 3D)
3. RoboCup nové stratégie (RoboCup S)
4. Simulácia a vizualizácia osvetlenia budov pre architektov so zameraním na samostatne stojace budovy v horách (BUDOVY)
5. Oznamovanie požiarov dobrovoľným hasičom (HASIČI)

Príloha 2

Rozvrh:

		7:00	8:00	9:00	10:00	11:00	12:00	13:00	14:00	15:00	16:00	17:00	18:00	19:00	20:00		
		7:50	8:50	9:50	10:50	11:50	12:50	13:50	14:00	15:50	16:50	17:50	18:50	19:50	20:50		
Pon	R.B.	PRACA						PDT			TIST		VIS				
	Š.D.							PDT			TIST		VIS				
	M.K.	APS									TSST		VSS				
	S.L.									TSST		VSS					
	M.Š.	APS			PRACA						TIST		VSS				
Uto	R.B.		STRETNUTIE č.1			ZS						BMIS		BMIS			
	Š.D.							KOD						BMIS		BMIS	
	M.K.							KOD		MSI						MSI	
	S.L.							ZS						MSI			
	M.Š.									KOD		MSI					
Str	R.B.	PRACA										STRETNUTIE č.2					
	Š.D.							PDT									
	M.K.																
	S.L.	NS															
	M.Š.	PRACA															
Štv	R.B.	KOD						VINF									
	Š.D.	KOD		VINF		VINF		VINF		AIS							
	M.K.	KOD		NP						ASP							
	S.L.																
	M.Š.	KOD		NP						ASP							
Pia	R.B.	PRACA										STRETNUTIE č.3					
	Š.D.																
	M.K.	NP		NP													
	S.L.																
	M.Š.	NP		PRACA													

Prednáška
Cvičenie
Zamestnanie
Stretnutie

Legenda:

PDT, TIST, VIS, TSST, VSS, APS, ZS, KOD, MSI, BMIS, NS, VINF, ASP - skratky predmetov

R.B., Š.D., M.K., S.L, M.Š - iniciály mien členov tímu

Stretnutie č.1, č.2, č.3 – preferované termíny stretnutí k projektu (podľa priority)