

Ponuka tímu č. 10

Tímový projekt

Predstavenie členov tímu

Bc. Jozef Krajčovič

Absolvent odboru Informatika na FPV UCM v Trnave. Vypracoval bakalársku prácu na tému „Návrh lekárskeho informačného systému ambulancie“. Má skúsenosti s vývojom webových ako aj desktopových aplikácií. Používa väčšinu technológií a nástrojov z dielne Microsoft. Zaujíma sa o tvorbu a vývoj používateľských rozhraní ako aj manažment. Ovláda technológií: PHP, JavaScript, C#, Visual Basic, C/C++, Java, Mysql, MSSQL a Oracle, .Net Framework (WPF, XAML, WCF), WindowsPowerShell.

Bc. Adrián Feješ

Je absolventom študijného odboru Informatika na FIIT STU. Vo svojej bakalárskej práci sa venoval procesu refaktorizácie zdrojových kódov a jej nástrojovej podpore. Výsledkom práce bol nástroj vo forme Eclipse plug-inu, podporujúci rozpoznávanie a označovanie antivzorov v kóde. Má skúsenosti s vývojom aplikácií hlavne v programovacom jazyku Java. Svoje vedomosti ďalej rozvíja aj v praxi, kde pracuje ako Java programátor a zaoberá sa vývojom podnikových aplikácií. Ovláda technológií: C/C++, Java SE/EE, XML, XMLSchema, XPath, SQL, JavaScript.

Bc. Lukáš Turský

Vyštudoval obor Informatika na FIIT STU. Počas štúdia sa zamerával najmä na vývoj aplikácií pre platformu Java SE a FX. V rámci bakalárskej práce analyzoval využitie Modelom riadenej architektúry pri tvorbe softvéru, pričom výstupom bolo úplné namodelovanie web aplikácie a jej následne implementovanie pre platformu Java EE (využitie Spring, Struts, Hibernate). Popri štúdiu získal skúsenosti v oblasti analýzy rizík a administrácie bezpečnosti bankových aplikácií. V dohľadnej dobe by sa chcel ďalej zamerať na vývoj webových aplikácií a rozšíriť znalosť databáz v rámci predmetu Pokročilé Databázové technológií.

Bc. Luboš Staraček

Absolvent študijného odboru Informatika na STU FIIT v Bratislave, vypracoval bakalársku prácu na tému „Štúdia realizácie zmien aspektovo-orientovaným spôsobom na úrovni modelu“. Za najpodstatnejšie získané zručnosti považuje osvojenie si objektovo a aspektovo orientovaného vývoja softvéru, metódy paralelného programovania a princípy umelej inteligencie. V rámci mimoškolskej činnosti vytvoril funkčnú web aplikáciu v jazyku JavaFX.

Bc. Maroš Jendrej

Absolvent študijného odboru Informatika na STU FIIT v Bratislave, vypracoval bakalársku prácu na tému „Manažovanie dokumentov“. Má skúsenosti s vývojom desktopových aplikácií pre platformu JAVA SE. Počas bakalárskeho štúdia si osvojil základy programovania v rôznych programovacích jazykoch a tiež získal znalosti o tvorbe softvérových systémoch. Po ukončení bakalárskeho štúdia sa zamestnal na pozícii QA/Tester v spoločnosti zaoberajúcej sa vývojom počítačových hier. V dohľadnej dobe by sa chcel hlbšie oboznámiť s počítačovou

grafikou a dizajnom používateľského rozhrania. Ovláda technológie: HTML/XHTML, XML, JAVA SE, C/C++, Assembler, UML, CUDA, MPI, OMP.

Bc. Marek Brath

Absolvent študijného odboru Informatika na FPV UCM v Trnave, vypracoval bakalársku prácu na tému „Programovanie v Jave“. Používa hlavne prostredie Eclipse na vytváranie desktopových aplikácií. Ovláda technológie: Java, C++, C#, PHP, HTML, XHTML, CSS, PHP, SQL.

1 Znalosti a zručnosti študentov (Znalosti)

1.1 Motivácia

V dnešnom svete plnom informácií je nájdenie a zostavenie tímu ľudí, obzvlášť takých ktorí sa takmer nepoznajú, často veľmi ťažko riešiteľný problém. Vidíme to aj teraz na nás, študentoch, že problémom je nedostatok a roztrúsenosť informácií o našich kolegoch. Veríme tomu, že my samí si možno časom začneme hovoriť, že zadelenie v rámci daného tímu nie je najideálnejšie.

Práve preto nás nadchla myšlienka vytvorenia centrálnej databázy schopností a znalostí jednotlivých študentov, ktorej plné využitie by mohlo siahať aj ďaleko do komerčnej sféry. Ved' pokiaľ by bol takýto systém dostatočne používateľsky prívetivý a interaktívny, mohol by uľahčiť prácu nielen učiteľom, ale určite aj neskôr študentom napr. pri hľadaní zamestnania.

Veľkú výhodu vidíme najmä v tom, že sami by sme boli motivovaný zlepšovať sa a týmto spôsobom ovplyvňovať svoje ohodnotenie v rámci systému.

Na druhej strane nielen pre profesorov ale aj vedúcich prác je to spôsob ako efektívne zostaviť tím podľa jeho preferencií a teda si môže rozhodnúť aké kvality by mal takýto tím, prípadne aj jednotlivec spĺňať. Taktiež je to informačný spôsob ako efektívne využiť potenciál každého jednotlivca v rámci vytváraného tímu a touto cestou aj zvýšenie miery na jeho budúci úspech.

Rozhodne by sme chceli stáť u zrodu takéhoto systému, lebo veríme tomu, že na to máme ako tím všetky predpoklady a bolo by veľmi zaujímavé pokiaľ by sa takýto systém podarilo reálne nasadiť do prevádzky.

1.2 Koncept riešenia

Vzhľadom na to, že väčšina nášho tímu má väčšie či menšie skúsenosti s konceptom a využívaním Java EE technológií, tak by sme chceli práve túto platformu využiť pre vytvorenie požadovanej client-server webovej aplikácie, ku ktorej budú môcť používatelia voľne pristupovať.

V rámci riešenia vidíme viacero hľadísk na ktoré bude potrebné sa zamerať. V rámci prezentačnej vrstvy je to nutnosť využiť interaktívne zaujímavý framework, ktorý by sa použil pre vytvorenie používateľského prostredia, a ktorý bude na použitie dostatočne prívetivý. Dobré vieme, že je to jediná časť s ktorou pracuje používateľ priamo a mnohokrát rozhoduje o zániku či úspechu systému. V tomto smere ešte nemáme jasno, o aký framework by šlo a teda by bolo nutné spraviť krátku analýzu.

Pre jednoduchšiu orientáciu, by mohlo vyhľadávanie a najmä pridávanie znalostí mať v hlavnej časti u každého študenta len nejaké zhrnutie jeho znalostí – správne zvolené väčšie celky, ktoré by zoskupovali podobné znalosti, napr. aká je miera technických znalostí, spoločenských schopností, využívanie určitých typov nástrojov, takisto by bolo zaujímavé mať aj indikátor ako sa študentovi darí v škole.

Keďže bolo spomenuté, že na takomto projekte sa už v minulosti pracovalo, tak by sme chceli využiť niektoré časti tohto riešenia, ktoré už sú dostatočne vyriešené a sústrediť sa na podstatnejšie veci, ktoré sú spomenuté nižšie.

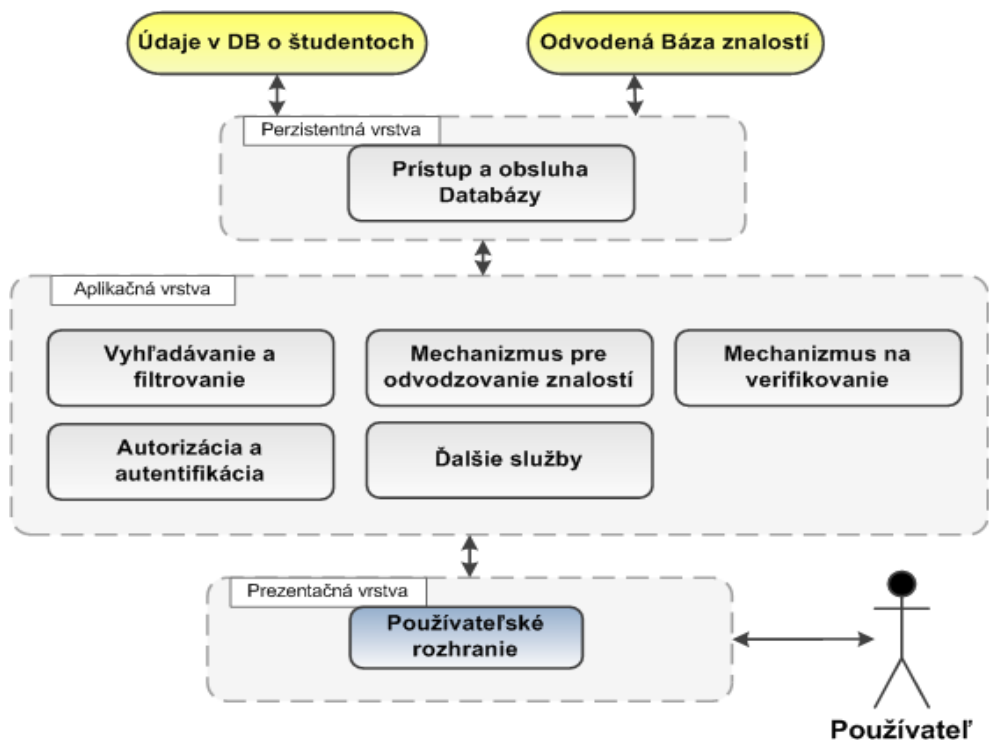
Z pohľadu aplikačnej logiky chceme venovať úsilie vytvoreniu mechanizmu, ktorý by vedel na základe daných schopností používateľa ďalej odvodiť bázu znalostí, ktorá by určite zlepšila šance pri filtrovaní a výbere. Reprezentácia znalostí by mala spĺňať požiadavky ako je rýchle vyhľadávanie a porovnávanie údajov, a preto spôsob reprezentácie musí byť jednoduchý a jednotný pre rôzne typy poznatkov a zručností.

Chceli by sme implementovať schopnosť automatizovane zadávať jednotlivé znalosti študentov, napr. ich hromadné pridávanie. V tomto smere by sa pre ich získavanie vo veľkej miere dalo využiť aj bodové hodnotenie v AIS pri jednotlivých zadaniach v rámci predmetov.

Vzhľadom na to, že pôjde o systém do ktorého bude mať prístup viacero skupín ľudí, navrhujeme vytvorenie viacerých úrovní prístupov (rolí) a k nim priradiť možné akcie, prípadne časti systému na ktoré by im tieto role dávali prístup. Teda logicky podľa toho do akej kategórie daný používateľ patrí, také akcie budú môcť vykonávať v systéme. Tu by bolo vhodné implementovať jednoduché pridávanie právomocí v rámci jednotlivých rolí, napr. odklikávanie akcií, alebo výber z listu.

Plánujeme implementovať spôsob overovania a kontroly študentmi zadávaných schopností, tak čo sa týka určitej miery verifikovateľnosti zadávaných schopností študentov, jeden spôsob vidíme v možnosti nechať zaslať požiadavku oprávnenej osobe na overenie.

Rozhodnutie koho potrebujeme zohnať a aké by mali byť požiadavky na študenta/tým by mali byť ponechané čisto na zadávateľa.



2 Digitálne divadlo (Divadlo)

2.1 Motivácia

Ovládanie softvéru pomocou ľudských pohybov a gest, bez nutnosti použitia klávesnice alebo myši, je samo o sebe veľmi zaujímavá a aktuálna téma. Obzvlášť, keď je tento projekt zameraný na tvorbu umeleckého diela, kde je zároveň výsledok tejto tvorby premietaný na plátno v reálnom čase. Myslíme si, že práca na takomto projekte bude pre nás zaujímavá, bude nás baviť, a tiež, v neposlednom rade, získame množstvo užitočných skúseností v zaujímavej oblasti IT.

Najmä kvôli týmto dôvodom náš tím zaujala táto téma, a chceli by sme sa podrobnejšie oboznámiť s možnosťami, ako využiť senzor Kinect na rozpoznávanie obrazovej informácie, pohybov a gest. Pokúsili by sme sa o vytvorenie originálneho riešenia, v ktorom by umelec pred plátnom pomocou svojho vlastného tela vytváral obraz. Ponúkli by sme mu na tvorbu diela nástroje, ktoré sú bežne v štandardných programoch na PC (Skicár, Adobe Photoshop, MyPaint...). Išlo by o vnorenie umelca do počítačovej reality, kde by aj bez tableta, či myšky mohol maľovať obraz.

2.2 Koncept riešenia

Standalone aplikácia pre platformu Windows XP, Vista a 7. Využívali by sme existujúce knižnice na detekciu pohybov a gest človeka. V rámci prípravy na vytvorenie tejto ponuky sme tiež vykonali analýzu niekoľkých video prezentácií umiestnených na portáli youtube, napríklad o používaní senzora Kinect na ovládanie konzoly X-Box 360 a podobne, čo nám môže poskytnúť veľa inšpirácie pri navrhovaní riešenia pre potreby tohto projektu.

Naše navrhované riešenie by mohlo byť akýmsi wrapperom na ľudské telo, pomocou ktorého sa bude vytvárať obraz, v prípade požiadavky na stereo projekciu sme pripravení pokúsiť sa o vytvorenie výstupného obrazu v troch dimenziách. Využili by sme pri tom rozpoznávanie hĺbky obrazu, ktorá nám je týmto senzorom ponúknutá. Prípadne, ak senzor Kinect umožňuje aj rozoznávanie hlasu, mohlo by stať za zváženie umožniť aj ovládanie kombináciou ľudských gest a hlasu. Tu by ale bolo dôležité zabezpečiť, aby bolo možné nastaviť ovládanie hlasom tak, že by príkazy hlasom mohla dávať iba oprávnená osoba, a nie ktokoľvek. Inak by mohli vznikať komplikácie, kde by počas používania tohto softvéru napríklad na prezentáciu mohol do tejto prezentácie vstupovať ktokoľvek z publika, čo je nežiaduce.

Jednou z alternatív pre overenia riešenia by mohlo ísť o vytvorenie prívetivého ovládania pre existujúcu open source aplikáciu MyPaint, slúžiacu na tvorbu obrázkov. Jej výhodou je jednoduché a minimalistické používateľské rozhranie, neobmedzený canvas bez nutnosti zmeny jeho rozmerov a schopnosť využívania grafického tabletu. Rovnako ako pri kreslení keď využívame grafický tablet by sme mohli využiť aj senzor Kinect, ktorý by za pomoci hĺbky obrazu dokázal určiť kedy umelec natáhuje ruku a teda snaží sa v obraze kresliť. Intenzitu kreslenia by sme určovali ako hlboko umelec ponorí svoju ruku do obrazu, je to

podobne ako sa na grafickom pere určuje stupeň prítlaku. Rozlišovali by sme tiež pravú a ľavú ruku, jedna by bola ako štetec a za pomoci druhej ruky by umelec vytváral gestá takto by prepínal medzi typmi štetcov, nastavoval farbu alebo inými funkciami. Trup umelca bude dynamickým stredom a maximálne natiahnutá ruka dopredu bude zaznamenané ako maximálna intenzita prítlaku štetca, nemôže sa tu stať niečo také, že štetec bude reagovať neprimerane.

Taktiež by mohlo byť zaujímavé implementovať ovládanie gestami do softvéru na tvorbu, respektíve spúšťanie prezentácií. Napríklad do open source programu OpenLP, ktorý okrem spúšťania prezentácií umožňuje aj prehrávanie videí, vytváranie a zobrazovanie galérií obrázkov a ďalšie. <http://openlp.org/en/features>.

3 Textový editor obohatený o grafické prvky (TextEdit)

3.1 Motivácia

Tato téma nás predovšetkým zaujala svojou myšlienkou vytvoriť akýsi multiplatformový grafický editor, ktorý využije grafické prvky na zvýraznenie štruktúr textu pomocou grafických blokov a tým podporili myšlienku „literate programming, pričom v súčasnosti veľa podobných riešení dosiaľ neexistuje a taktiež fakt, že práca na editore je z veľkej časti hotová a teda by sme sa mohli plne zamerať na vytvorenie akýchsi features nad touto aplikáciou a premyslieť jej čo možno najmasívnejšie zavedenie do praxe.

Ďalšou motiváciu pre nás je, že si pri tomto projekte môžeme rozšíriť svoje znalosti a zručnosti o nové technológie a postupy v danej doméne, ktorá je pre nás zaujímavá. Uvedomujeme si, že s danou doménou nemáme veľa praktických skúseností čo sa môže zdať ako nevýhoda, ale opak je však pravdou a o to viac to bude pre nás väčšia výzva, aby sme dobudovali ešte kvalitnejší produkt, ktorý by sa presadil aj v komerčnej sfére.

3.2 Koncept riešenia

Cieľom tohto projektu bude pokračovať vo vývoji existujúceho multiplatformového editora (TrollEdit), ktorý bol vytvorený predchádzajúcim tímom „UFOPAK“. Naším zameraním pre editor bude rozšírenie stávajúcej funkcionality pre reálne nasadenie editora do praxe. Najväčšiu zmenou bude vylepšenie používateľského rozhrania, ktoré v súčasnom editore je tak ako u podobných editor čo sa týka dizajnu mierne nezaujímavé t.j. zatiaľ klasický dizajn „ala notepad“.

Pri implementácii budeme predovšetkým vychádzať z už použitých technológií ako knižnica Qt, skriptovací jazyk Lua a podobne, plus niektoré nami zvolené technológie, ktoré sa rozhodneme použiť po podrobnej analýze súčasného editora.

Čo sa týka rozšírenia funkcionality plánujeme implementovať tieto vylepšenia:

- Možnosti „undo“/ „redo“.
- Detekcia pachov kódu.
- Možnosť rozšírených nastavení priamo v editore
- Určitý druh fulltextového vyhľadávania s prípadnou optimalizáciou na najčastejšie vyhľadávané výrazy.
- Možnosť exportovania súboru do iných formátov (XML, WORD)
- Schopnosť detegovať určité ukazovatele v zdrojovom kóde ako index udržateľnosti, cyklomatická zložitosť, alebo hodnoty „fan in“ a „fan out“, ktoré by boli zobrazené v určitej tabuľke.

Taktiež plánujeme čo najvhodnejšie použiť známe návrhové vzory, aby sme zabezpečili vysokú modularitu systému a tým umožnili neskoršie pridávanie a modifikovanie funkcionality.

Ohľadom spomínaného dizajnu používateľského rozhrania plánujeme vďaka podpore Qt modulu pre vývojové prostredie Visual Studio použiť najmodernejšie technológie ako WPF (Windows presentation foundations), XML.

Tieto technológie nám umožnia navrhnuť si dizajn podľa vlastnej fantázie bez zdĺhavého programovania pri ktorom by sme museli použiť rôzne grafické knižnice čo v tomto prípade odpadá. Plánujeme návrh dizajnu používateľského rozhrania v štýle „Office“ t.j. použiť dobre známí „*Ribbon*“, ktorý je stále častejšie používaní v desktopových aplikáciách.

Veríme, že nami navrhnuté riešenie vo finálnej verzii bude kvalitný produkt, ktorý nájde uplatnenie v praxi.

Príloha A – Zoradenie všetkých tém podľa priority

Priorita	Názov témy	Číslo témy
1.	Znalosti a zručnosti študentov (Znalosti)	13
2.	Digitálne divadlo (Divadlo)	3
3.	Textový editor obohatený o grafické prvky (TextEdit)	11
4.	Štatistický preklad voľného textu (Preklad)	9
5.	Inteligentná hra pre mobilné zariadenia (MobHra)	8
6.	Rozvrhový systém novej FIIT (Rozvrhy)	12
7.	Plagiáty na webe (Plagiáty)	4
8.	Simulácia davu (Dav)	15
9.	Personalizované odporúčanie (Odporúčanie)	5
10.	Osobný manažment fyzickej aktivity pomocou mobilných zariadení (Aktivita)	2
11.	Editovanie viacrozmerného grafu prepojenia informácií v dokumentoch (Dokumenty)	16a
12.	Virtuálna FIIT (VirtFIIT)	14
13.	RoboCup - tretí rozmer (RoboCup)	7
14.	Webový editor pre TeX (WebEdit)	10
15.	Tvorba "ľahko" sémantického obsahu pre adaptívny webový (výučbový) portál (ALEF)	6
16.	Imagine Cup 2012: Game Design (ICup2012) - pridelená	1
17.	3D UML (3D UML)	16b

Prednáška

Cvičenie

Vyhovující čas

Pokročilé databázové technológie - blokovo