

Slovenská technická univerzita

Fakulta informatiky a informačných technológií

Ilkovičova 3, 842 16 Bratislava 4

Milan Frelm, David Chalupa, Marek Mego,
Peter Mindek, Michal Noskovič a Matej Sabo

Podpora kontroly plagiarizmu

Tímový projekt I. – riadenie

Študijné programy: Informačné systémy, Softvérové inžinierstvo

Vedúca projektu: Mgr. Daniela Chudá, PhD.

Ak. rok: 2009/10

Obsah

1	ÚVOD	1-2
2	PLÁNY	2-1
2.1	Zimný semester	2-1
2.2	Letný semester	2-3
2.3	Ganttov diagram	2-7
3	ÚLOHY ČLENOV TÍMU	3-1
3.1	Roly a zodpovednosti v tíme	3-1
3.2	Rozdelenie zodpovedností za technickú dokumentáciu	3-1
3.3	Rozdelenie zodpovedností za dokumentáciu k riadeniu	3-3
3.4	Rozdelenie zodpovedností za implementáciu	3-3
3.5	Rozdelenie zodpovedností za dokumentáciu v letnom semestri	3-4
3.6	Rozdelenie zodpovedností za implementáciu v letnom semestri	3-6
4	ZÁPISY ZO STRETNUTÍ	4-1
4.1	Zápis zo stretnutia 1	4-1
4.2	Zápis zo stretnutia 2	4-3
4.3	Zápis zo stretnutia 3	4-4
4.4	Zápis zo stretnutia 4	4-6
4.5	Zápis zo stretnutia 5	4-8
4.6	Zápis zo stretnutia 6	4-10
4.7	Zápis zo stretnutia 7	4-12
4.8	Zápis zo stretnutia 8	4-15
4.9	Zápis zo stretnutia 9	4-18
4.10	Zápis zo stretnutia 10	4-20
4.11	Zápis zo stretnutia 11	4-23
4.12	Zápis zo stretnutia 12	4-26
5	PODPORNÉ PROSTRIEDKY	5-1
5.1	Využívanie služby Google groups	5-1
5.2	Manažment úloh v nástroji Redmine	5-1
5.3	Manažment verzií v nástroji Git	5-3
5.4	Linkovanie knižnice GSL v IDE QtCreator	5-5
5.5	Použitie knižnice QTestLib	5-5
6	ŠTÁBNA KULTÚRA	6-1
6.1	Konvencie písania dokumentácie k inžinierskemu dielu	6-1
6.2	Konvencie písania zdrojových kódov	6-5
6.3	Konvencie komentovania zdrojových kódov	6-8
7	DOKUMENTY K TESTOVANIU	7-1
7.1	Návrh testov	7-1
7.2	Popis testovacích prípadov a procedúr	7-1
7.3	Protokoly o vykonaní testov	7-5
	PRÍLOHA A – PONUKA	A-1
	PRÍLOHA B – PREBERACIE PROTOKOLY	B-1

1 Úvod

Tento dokument obsahuje informácie a materiály tímu Copypaste (tím č. 9), ktoré sa týkajú riadenia a ktoré vznikli počas riešenia projektu.

Ponuku tímu sme pôvodne mali v kapitole č. 2, ale z dôvodu prehľadnosti sme ju presunuli do prílohy. Touto ponukou sme sa uchádzali o pridelenie témy Textový editor obohatený o grafické prvky. Po pridelení témy Podpora kontroly plagiarizmu sme ju analyzovali a navrhli predbežný časový plán.

Časový plán sme priebežne aktualizovali a jeho aktuálna podoba sa nachádza v kapitole č. 2. Obsahuje ako rámcový, tak aj podrobný plán.

V tretej kapitole sú zhrnuté hlavné úlohy a zodpovednosti každého člena tímu. Taktiež obsahuje zodpovednosti jednotlivých členov tímu pri tvorbe dokumentácií a pri implementácii. Tieto úlohy a zodpovednosti ale nie sú konečné a môžu sa meniť v závislosti od rôznych priorit aktuálnych úloh projektu. Na každom stretnutí vždy jeden z nás plnil úlohu zapisovateľa a vedúceho stretnutia. Jednotlivé zápisnice, tvoriace pomocný výstupný dokument každého stretnutia, sú v kapitole č. 4. Sú zoradené od prvej zápisnice po najnovšiu.

Piata kapitola obsahuje informácie o použitých podporných prostriedkoch. Zo začiatku sme používali Google Groups, neskôr sme prešli na nástroj Redmine. Ako nástroj na manažment verzií sme použili Git. Taktiež obsahuje informácie o knižnici QTestLib, ktorá je súčasťou knižníc QT a ktorú sme používali pri testovaní aplikácie.

Šiesta kapitola obsahuje metodiky a konvencie pre písanie zdrojového kódu, jeho komentárov a projektových dokumentácií. Viaceré tieto konvencie sme si vytvorili z dôvodov jednotnosti písania a väčšej prehľadnosti.

V siedmej kapitole sú návrhy na vykonanie testov a protokoly z jednotlivých testovaní.

2 Plány

2.1 Zimný semester

2.1.1 Rámcový plán

Týždeň	Dátum	Popis činnosti	Stav
1.	21.9. - 27.9.	Vytvorenie tímu, výber témy	☀
2.	28.9. - 4.10.	Vytvorenie a odovzdanie ponuky	☀
3.	5.10. - 11.10.	Návrh webstránky	☀
4.	12.10. - 18.10.	Analýza existujúcich riešení, špecifikácia požiadaviek, návrh riešenia	☀
5.	19.10. - 25.10.	Analýza existujúcich riešení, nasadenie podporného prostriedku, špecifikácia požiadaviek	☀
6.	26.10. - 1.11.	Návrh riešenia, hotová web prezentácia, tvorba dokumentu	☀
7.	2.11. - 8.11.	Tvorba dokumentu, odovzdanie analýzy, špecifikácie požiadaviek a návrhu riešenia	☀
8.	9.11. - 15.11.	Implementácia prototypu	☀
9.	16.11. - 22.11.	Implementácia prototypu	☀
10.	23.11. - 29.11.	Implementácia prototypu	☀
11.	30.11. - 6.12.	Implementácia prototypu, spracovanie dosiahnutých výsledkov	☀
12.	7.12. - 13.12.	Odovzdanie prototypu, dokumentácie a používateľská prezentácia prototypu	☀

Legenda:

- ☀ činnosť ukončená
- ☁ činnosť prebieha
- ☁ činnosť ešte nezačala

2.1.2 Podrobný plán














Týždeň	Dátum	Popis činnosti	Zodpovednosť	Priorita	Stav
1.	21.9. - 27.9.	Vytvorenie ponuky	Všetci	Vysoká	☀
2.	28.9. - 4.10.	Odovzdanie ponuky	Všetci	Vysoká	☀
3.	5.10. - 11.10.	Vytvorenie a rozpracovanie plánu na zimný semester	Michal Noskovič Peter Mindek	Vysoká	☀
3.	5.10. - 11.10.	Rozdelenie úloh v tíme	David Chalupa	Stredná	☀
3.	5.10. - 11.10.	Výroba a tlač tímového plagátu	Peter Mindek	Stredná	☀
3.	5.10. - 11.10.	Naštudovať problematiku plagiátorstva	Marek Mego	Vysoká	☀
4.	12.10. - 18.10.	Pozrieť prácu Bianky Kováčovej a Pavla Humaya	Všetci	Vysoká	☀
4.	12.10. - 18.10.	Analyzovať existujúce prostriedky na detekciu plagiarizmu	Všetci	Vysoká	☀
5.	19.10. - 25.10.	Vytvorenie webovej prezentácie projektu	Michal Noskovič	Normálna	☀
5.	19.10. - 25.10.	Diskusia o implementačnom	Peter Mindek	Stredná	☀

		prostredí			
5.	19.10. - 25.10.	Analyzovať požiadavky na slovenčinu	David Chalupa Michal Noskovič Milan Freml	Stredná	☀
5.	19.10. - 25.10.	Analyzovať požiadavky na zdrojové kódy	Peter Mindek Marek Mego Matej Sabo	Stredná	☀
6.	26.10. - 1.11.	Diskusia o podpornom prostriedku	Všetci	Vysoká	☀
6.	26.10. - 1.11.	Analyzovať program Moss	Peter Mindek Milan Freml	Stredná	☀
6.	26.10. - 1.11.	Analyzovať program Sim	Marek Mego	Stredná	☀
6.	26.10. - 1.11.	Analyzovať program Sherlock	Michal Noskovič	Stredná	☀
6.	26.10. - 1.11.	Analyzovať program Yap	David Chalupa	Stredná	☀
6.	26.10. - 1.11.	Vytvoriť metodiku na tvorbu testovacích dát	Matej Sabo	Vysoká	☀
7.	2.11. - 8.11.	Rozpracovať architektúru na nižšiu úroveň	Peter Mindek Matej Sabo	Vysoká	☀
7.	2.11. - 8.11.	Začiatok implementácie	Peter Mindek	Vysoká	☀
7.	2.11. - 8.11.	Implementovať jadro	Peter Mindek	Urgentná	☀
7.	2.11. - 8.11.	Implementovať loader	Milan Freml Marek Mego Peter Mindek	Vysoká	☀
7.	2.11. - 8.11.	Vytvorenie dát na testovanie (texty a kódy)	Matej Sabo Marek Mego	Stredná	☀
8.	9.11. - 15.11.	Analyzovať možnosti vizualizácie	Michal Noskovič Peter Mindek	Stredná	☀
8.	9.11. - 15.11.	Porovnať nástroje na slovenské texty	Marek Mego Michal Noskovič	Stredná	☀
8.	9.11. - 15.11.	Implementovať parser manažér	Michal Noskovič Matej Sabo	Vysoká	☀
8.	9.11. - 15.11.	Napísať prihlášku TP Cup	David Chalupa	Urgentná	☀
9.	16.11. - 22.11.	Implementovať rozhranie pre príkazový riadok	David Chalupa	Stredná	☀
9.	16.11. - 22.11.	Nainštalovať git	Milan Freml	Stredná	☀
9.	16.11. - 22.11.	Odovzdať prihlášku na TP Cup	David Chalupa	Urgentná	☀
9.	16.11. - 22.11.	Rozšírenie Parser Managera	Michal Noskovič	Urgentná	☀
9.	16.11. - 22.11.	Pridanie ďalšej porovnávacej metódy (gst)	David Chalupa	Stredná	☀
9.	16.11. - 22.11.	Okomentovanie zdrojových kódov pomocou DoxyGenu	Milan Freml	Normálna	☀
9.	16.11. - 22.11.	Predspracovanie slovenských textov - odstránenie stop slov	Michal Noskovič	Stredná	☀
9.	16.11. - 22.11.	Predspracovanie slovenských textov - lematizácia	Marek Mego	Stredná	☀
9.	16.11. - 22.11.	Compare Manager - porovnávanie každý s každým obojstranne	Matej Sabo	Stredná	☀
9.	16.11. - 22.11.	Compare Manager - porovnávanie jedného súboru s ostatnými	Matej Sabo	Stredná	☀
10.	23.11. - 29.11.	Rozšíriť parser manager o string bluring	Peter Mindek	Stredná	☀
10.	23.11. - 29.11.	Porovnať náš prototyp s inými	Marek Mego	Stredná	☀

		programami			
10.	23.11. - 29.11.	Otestovať korektnosť jednotlivých porovnávacích metód	Matej Sabo	Stredná	
10.	23.11. - 29.11.	Dopracovanie dokumentácie	David Chalupa Milan Freml	Vysoká	
10.	23.11. - 29.11.	Predspracovanie zdrojových kódov, tokenizácia	Milan Freml Peter Mindek	Vysoká	
10.	23.11. - 29.11.	Analýza prístupu na Moodle	Marek Mego Milan Freml	Stredná	
11.	30.11. - 6.12.	Dokončenie prototypu	Peter Mindek	Vysoká	
11.	30.11. - 6.12.	Dopracovanie nedostatkov	David Chalupa Milan Freml	Vysoká	
11.	30.11. - 6.12.	Doplnenie dokumentácie k prototypu	Milan Freml Matej Sabo	Vysoká	
11.	30.11. - 6.12.	Automatické vygenerovanie dokumentácie	Milan Freml	Vysoká	
12.	7.12. - 13.12.	Odovzdanie prototypu vybraných častí systému spolu s dokumentáciou a používateľskou prezentáciou prototypu	Všetci	Vysoká	
12.	7.12. - 13.12.	Pokračovanie v implementácii	Všetci	Vysoká	
12.	7.12. - 13.12.	Používateľská prezentácia prototypu	David Chalupa	Vysoká	




2.2 Letný semester

2.2.1 Rámcový plán








Týždeň	Popis činnosti	Zodpovednosť	Priorita	Stav
1.	Dopracovanie nedostatkov na dokumentácii	David Chalupa	Vysoká	
1.	Nainštalovať Moodle	Marek Mego Milan Freml	Urgentná	
1.	Analýza architektúry Moodle	Marek Mego Milan Freml	Urgentná	
1.	Integrácia lematizácie	Peter Mindek	Vysoká	
1.	Vytvorenie podrobnejšieho plánu	Michal Noskovič Peter Mindek	Normálna	
2.	Funkčný skript na prístup na Moodle	Marek Mego Milan Freml	Urgentná	
2.	Import *.doc súborov	Michal Noskovič	Vysoká	
3.	Prepojenie aplikácie s Moodleom	Peter Mindek Milan Freml	Urgentná	
4.	Začiatok prác na GUI aplikácie	Peter Mindek	Vysoká	
4.	Analýza vizualizačného modulu	Matej Sabo	Vysoká	
5.	Integrácia konzolovej aplikácie s GUI	Peter Mindek David Chalupa	Vysoká	
5.	Implementovať skriptovateľný parser manager	Michal Noskovič	Vysoká	
5.	Side-by-side vizualizácia	Marek Mego	Stredná	

6.	Grafová vyzualizácia	Peter Mindek	Stredná	
6.	Rozšírenie GUI aplikácie	David Chalupa Matej Sabo	Vysoká	
7.	Dokončenie implementácie GUI	David Chalupa Matej Sabo	Vysoká	
7.	Použiť lua skripty na tokenizáciu programových kódov	Michal Noskovič	Vysoká	
8.	Dokončiť implementáciu parser manažera	Michal Noskovič	Vysoká	
8.	Revízia kódu	Peter Mindek	Vysoká	
9.	Kompletizácia dokumentácie	David Chalupa	Vysoká	
9.	Dokončiť implementáciu	Peter Mindek Milan Freml Michal Noskovič	Urgentná	
9.	Revízia dokumentácie	David Chalupa Marek Mego Matej Sabo	Vysoká	
9.	Odovzdanie produktu	Všetci	Vysoká	
9.	Odovzdanie dokumentáciu k produktu	Všetci	Vysoká	
10.	Komplexné testovanie	Peter Mindek Milan Freml Michal Noskovič	Vysoká	
11.	Testovanie na viacerých platformách	Marek Mego Milan Freml	Vysoká	
11.	Dopracovanie zistených nedostatkov	Michal Noskovič David Chalupa	Urgentná	
12.	Odovzdanie celkového výsledku projektu (produkt so zmenami v rámci údržby, dokumentácia)	Všetci	Vysoká	

Legenda:

-  činnosť ukončená
-  činnosť prebieha
-  činnosť ešte nezačala

2.2.2 Podrobný plán

Týždeň	Dátum	Popis činnosti	Zodpovednosť	Priorita	Stav
1.	15.2. - 21.2.	Dopracovanie nedostatkov na dokumentácii	Všetci	Vysoká	
2.	22.2. - 28.2.	Prípraviť podklady pre side-by-side vizualizáciu	Michal Noskovič	Vysoká	
2.	22.2. - 28.2.	Pozrieť, ako riešia side-by-side vizualizáciu iné riešenia	Matej Sabo	Vysoká	
2.	22.2. - 28.2.	Návrh riešenia integrácie s LMS Moodle	Marek Mego	Vysoká	
2.	22.2. - 28.2.	Napísať priebežnú správu na TP Cup	David Chalupa	Stredná	
2.	22.2. - 28.2.	Návrh riešenia prihlasovania do LMS cez HTTPS	Peter Mindek	Vysoká	
3.	1.3. - 7.3.	Unit testy	Matej Sabo	Vysoká	
3.	1.3. - 7.3.	Návrh riešenia integrácie s LMS Moodle	Marek Mego	Vysoká	
3.	1.3. - 7.3.	Side-by-side vizualizácia	Michal Noskovič	Vysoká	

3.	1.3. - 7.3.	Implementovať tokenizáciu zdrojových kódov	Milan Freml Peter Mindek	Vysoká	☀
3.	1.3. - 7.3.	Optimalizovať implementáciu metódy LCS	Matej Sabo	Stredná	☀
3.	1.3. - 7.3.	Metóda LSA	David Chalupa	Vysoká	☀
4.	8.3. - 14.3.	Dokončiť tokenizáciu zdrojových kódov	Milan Freml Peter Mindek	Vysoká	☀
4.	8.3. - 14.3.	Optimalizovať vizualizácie	Peter Mindek	Stredná	☀
4.	8.3. - 14.3.	Integrácia s moodlom	Marek Mego	Vysoká	☀
4.	8.3. - 14.3.	Automatické testovanie	Matej Sabo	Stredná	☀
4.	8.3. - 14.3.	Nájsť prípadné chyby v implementácii LSA	David chalupa	Vysoká	☀
4.	8.3. - 14.3.	Optimalizovať metódu LCS	Matej Sabo	Vysoká	☀
4.	8.3. - 14.3.	Priebežné vypisovanie výsledkov porovnávania, na základe signálov	Milan Freml Peter Mindek	Stredná	☀
4.	8.3. - 14.3.	Prerobenie algoritmu na side-by-side vizualizáciu	Michal Noskovič	Stredná	☀
4.	8.3. - 14.3.	Spájanie zdrojových súborov, v prípade situácie 1 adresár = 1 program	Michal Noskovič	Nízka	☀
4.	8.3. - 14.3.	Dokončiť konvertor z doc do txt, kontrolovať spustené procesy	Milan Freml	Stredná	☀
4.	8.3. - 14.3.	Chybová konzola do GUI, výpis varovaní a chýb na chybový výstup	Marek Mego	Stredná	☀
4.	8.3. - 14.3.	Formalizácia procesov	David Chalupa	Stredná	☀
4.	8.3. - 14.3.	Integrácia lematizácie	Milan Freml	Vysoká	☀
5.	15.3. - 21.3.	Modifikáciu klientskej časti pre integráciu s Moodle	Marek Mego Peter Mindek	Vysoká	☀
5.	15.3. - 21.3.	Vylepšenie integrácie s Moodle o identifikáciu autorov súborov	Marek Mego	Vysoká	☀
5.	15.3. - 21.3.	Unit test trigramov	Matej Sabo	Vysoká	☀
5.	15.3. - 21.3.	Revízia abstraktu	David Chalupa	Stredná	☀
5.	15.3. - 21.3.	Príprava podkladov pre testovacie dáta pre zdrojové kódy	David Chalupa	Vysoká	☀
5.	15.3. - 21.3.	TP Cup dotazník a fotka	David Chalupa	Vysoká	☀
5.	15.3. - 21.3.	Integrácia GST	Michal Noskovič	Vysoká	☀
5.	15.3. - 21.3.	Integrácia tokenizácie	Peter Mindek	Vysoká	☀
5.	15.3. - 21.3.	Vytvorenie framework-u pre side-by-side vizualizáciu	Peter Mindek	Vysoká	☀
5.	15.3. - 21.3.	Rozpakovávanie zip súborov	Milan Freml	Vysoká	☀
5.	15.3. - 21.3.	Riešenie kolízií mien súborov	Milan Freml	Vysoká	☀
5.	15.3. - 21.3.	Posielanie dát na vizualizovanie na štandardný výstup	Peter Mindek	Vysoká	☀
5.	15.3. - 21.3.	Aplikácia ľahkých zmien kódu na testovacie zdrojové kódy	David Chalupa, Matej Sabo	Vysoká	☀
5.	15.3. - 21.3.	Doplnenie názvov súborov do kindershifter	Peter Mindek	Vysoká	☀
5.	15.3. - 21.3.	Commit-nuť php skript do gitu	Marek Mego	Vysoká	☀
6.	22.3. - 28.3.	Prerobiť tabuľku do samostatného vlákna	Peter Mindek	Vysoká	☀
6.	22.3. - 28.3.	Grafový vizualizátor má texty	Peter Mindek	Vysoká	☀

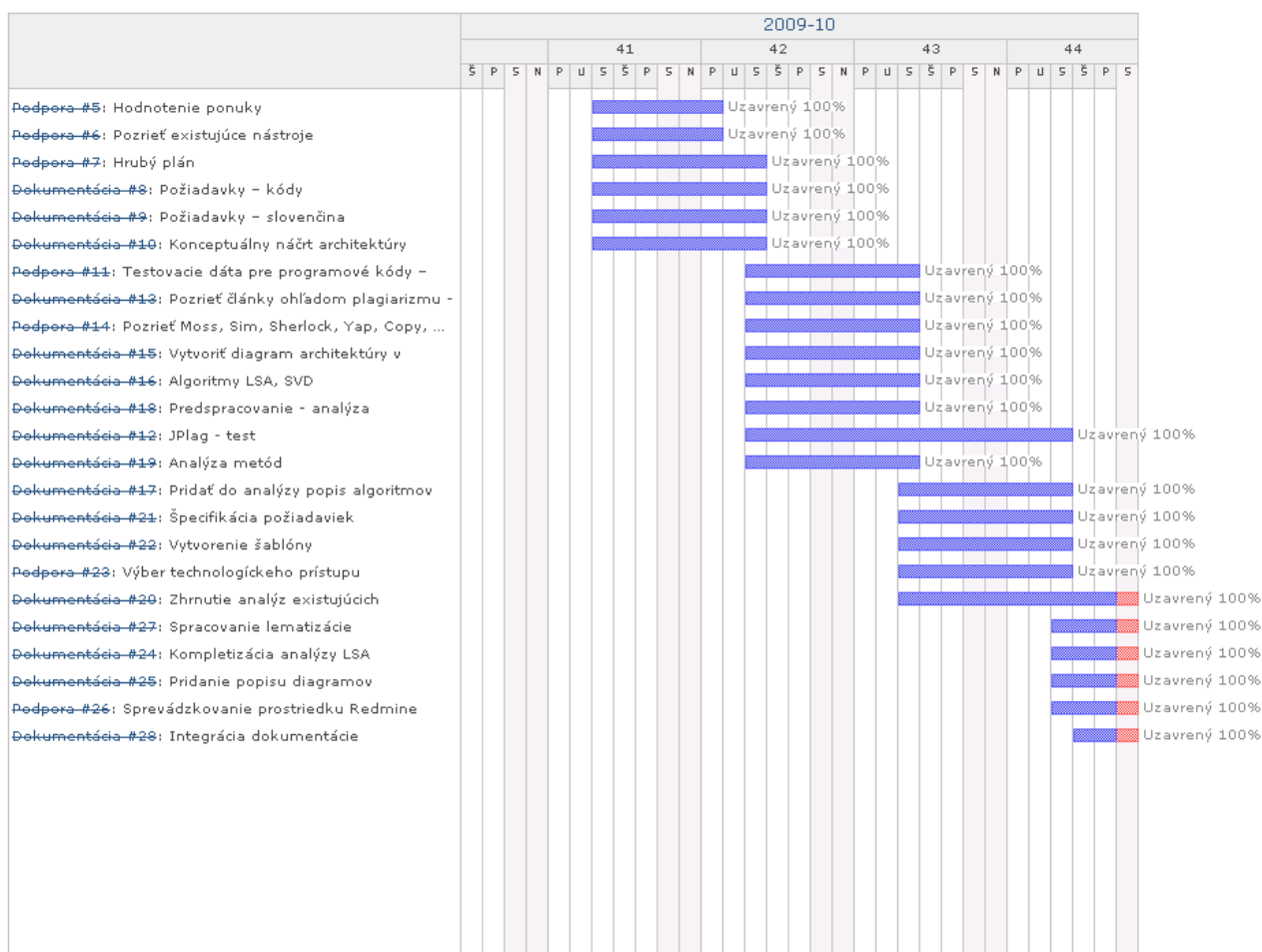
		mimo určenej plochy			
6.	22.3. - 28.3.	Unit test predspracovania textu	Michal Noskovič	Vysoká	☀️
6.	22.3. - 28.3.	Implementácia tokenizéra pre jazyk Java	Matej Sabo	Vysoká	☀️
6.	22.3. - 28.3.	Vizualizovať priebeh tokenizácie	Peter Mindek	Vysoká	☀️
6.	22.3. - 28.3.	Unit test komparátora N gramov	Marek Mego	Vysoká	☀️
7.	29.3. - 4.4.	Testovanie exportu výsledkov GST	David Chalupa	Vysoká	☀️
7.	29.3. - 4.4.	Napísanie helpu do Gui	Peter Mindek	Vysoká	☀️
8.	5.4. - 11.4.	Upraviť tokenizér	Peter Mindek, Milan Freml	Normálna	☀️
8.	5.4. - 11.4.	Vytvorenie logu pre export výsledkov	Peter Mindek	Vysoká	☀️
8.	5.4. - 11.4.	Vytvoriť predbežný návrh postera	Peter Mindek	Vysoká	☀️
8.	5.4. - 11.4.	Implementácia plagiárizátora	Milan Freml	Stredná	☀️
8.	5.4. - 11.4.	Prerobiť prepínač v CLI	Milan Freml	Vysoká	☀️
8.	5.4. - 11.4.	Vytvoriť mapovanie pre vizualizáciu výsledkov N-gramov	Michal Noskovič Matej Sabo	Vysoká	☀️
8.	5.4. - 11.4.	Napísať manuál k aplikácii	Matej Sabo	Vysoká	☀️
8.	5.4. - 11.4.	Napísať pokyny pre dotvorenie dokumentácie	David Chalupa	Urgentná	☀️
8.	5.4. - 11.4.	Integrovať existujúce súčasti do dokumentácie k riadeniu	David Chalupa	Vysoká	☀️
8.	5.4. - 11.4.	Porovnať naše výsledky s inými aplikáciami	David Chalupa Marek Mego	Vysoká	☀️
9.	12.4. - 18.4.	Odovzdanie produktu	Všetci	Vysoká	☀️
9.	12.4. - 18.4.	Odovzdanie dokumentácie k produktu	Všetci	Vysoká	☀️
9.	12.4. - 18.4.	Prevádzka, externé testovanie, údržba, kompletizácia dokumentácie	Všetci	Urgentná	☁️
10.	19.4. - 25.4.	Prevádzka, externé testovanie, údržba, kompletizácia dokumentácie	Všetci	Urgentná	☁️
11.	26.4. - 2.5.	Prevádzka, externé testovanie, údržba, kompletizácia dokumentácie	Všetci	Urgentná	☁️
12.	3.5. - 9.5.	Odovzdanie celkového výsledku projektu	Všetci	Urgentná	☁️

2.3 Ganttov diagram

Táto kapitola obsahuje Ganttove diagramy podľa vytvorených úloh počas semestra, ktoré sme taktiež vkladali do prostriedku Redmine. Redmine ponúka výbornú podporu pre manažment úloh projektu a tvorbu Ganttových diagramov, preto sú priamo exportované z tohto prostriedka. Diagramy pre jednotlivé mesiace sú uvedené v nasledujúcich podkapitolách.

2.3.1 Október 2009

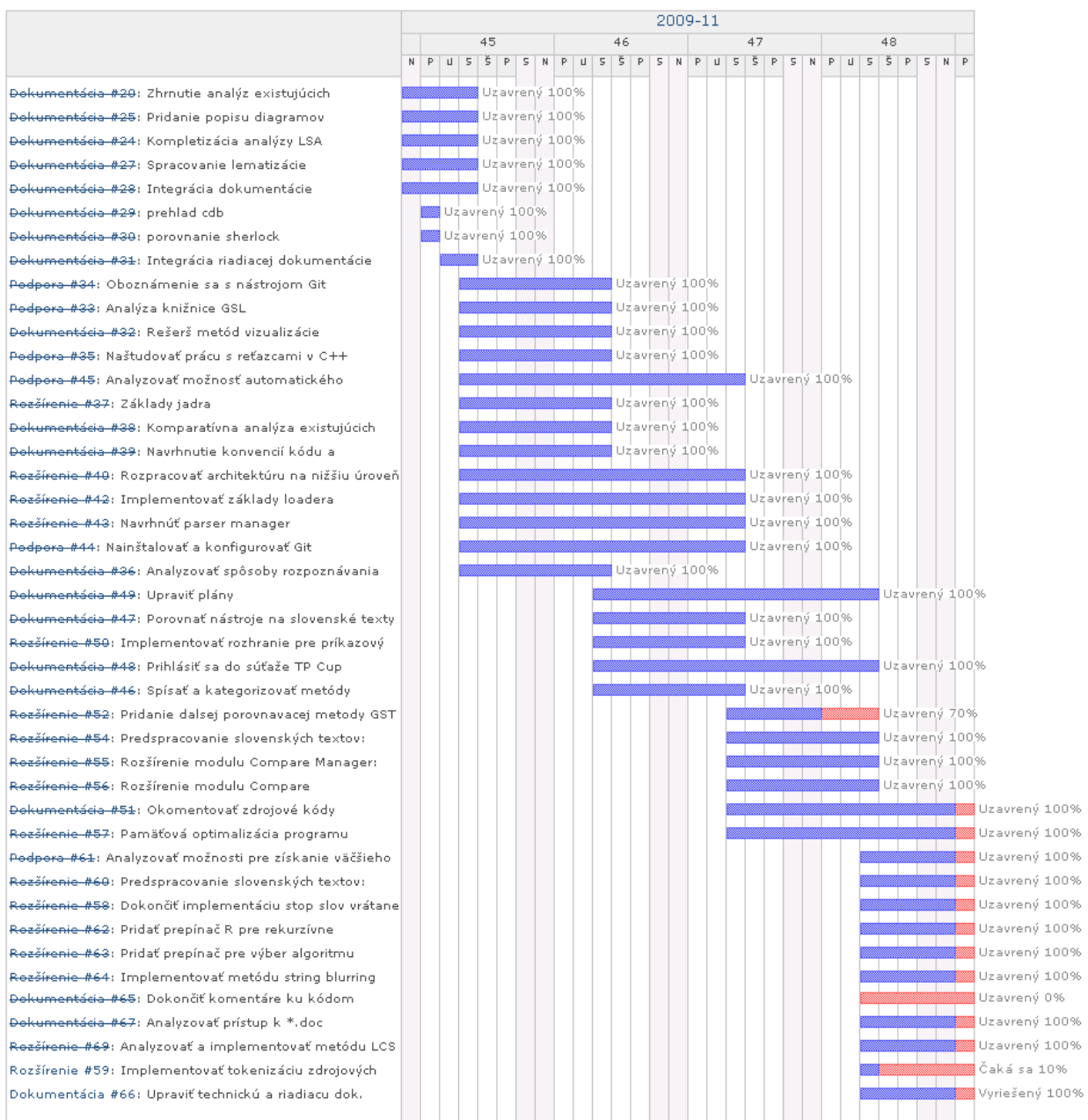
V októbri sme sa zaoberali hlavne analýzou metód, algoritmov a existujúcich riešení. Na začiatku sme vytvorili hrubý plán, ktorý sme neskôr upravili hlavne z dôvodov jeho prílišnej všeobecnosti. Potom sme špecifikovali požiadavky na našu budúcu aplikáciu a vytvorili náčrt architektúry. Následne sme sa oboznamovali s riešeniami iných programov a analýzou implementácie algoritmov. Keďže na analýze pracoval celý tím, bolo nutné jej výsledky integrovať a upraviť do výslednej podoby, čo sme vykonávali na konci mesiaca a v novembri. Na konci mesiaca sme taktiež sprevádzkovali prostriedok Redmine.



2.3.2 November 2009

V prvej polovici mesiaca sme pokračovali v analýze a integrovaní jej výsledkov do dokumentácie. Neskôr sme sa zaoberali štúdiom knižníc C++ a QT a metodikou pre jednotné písane zdrojových kódov. Potom sme pokračovali samotnou implementáciou, počas ktorej sme si taktiež inštalovali prostriedok pre manažment verzií - Git. Implementovali sme jadro aplikácie a pokračovali implementáciou ďalších modulov podľa architektúry, navrhutej v predošlom mesiaci. Túto architektúru sme ešte čiastočne pozmenili po odhalení niektorých nedostatkov. Rozhodli sme sa, že prototyp aplikácie bude fungovať iba ako konzolová aplikácia, pretože sme sa zamerali hlavne na funkcionálnu. Grafické užívateľské rozhranie plánujeme implementovať v ďalšom semestri.

V druhej polovici mesiaca sme sa zaoberali hlavne implementáciou. Implementovali sme triedy pre načítavanie, porovnávanie a kombinovanie súborov a zobrazovanie výsledkov. Program ponúka taktiež kontrolu súborov vo formáte pdf, ktoré automaticky konvertuje na textové súbory (txt). Počas implementácie sme taktiež pracovali na technickej aj riadiacej dokumentácii.



2.3.3 December 2009

Na začiatku mesiaca sme pokračovali v implementovaní a dokončovaní prototypu. Rozšírili sme ho o ďalšie možnosti porovnávania, práce so súbormi a implementovali predspracovanie textov, na ktorom budeme ešte pracovať. Taktiež sme aktualizovali konzolové rozhranie, čo nadviazovalo na implementáciu funkcionality. Pred koncom semestra sme sa zaoberali dopísaním a integrovaním dokumentácií.



2.3.4 Február 2010

Vo februári sme spočiatku hlavne opravovali vzniknuté chyby na prototypu. Ďalej sme pokračovali implementáciou a rozšírením aplikácie. Medzi hlavné procesy implementácie sme zaradili implementáciu tokenizácie zdrojového kódu, vizualizácie výsledkov a prepojenie s prostriedkom Moodle. Vo všetkých prípadoch sa jednalo o dlhšie procesy, ktoré trvali viacero týždňov a zahrňovali viaceré úlohy. Okrem toho sme pracovali na implementácii novej metódy pre porovnávanie – latentno sémantickej analýze.

2.3.5 Marec 2010

V marci sme sa začali zaoberať testovaním implementácie a optimalizáciou niektorých funkcií a algoritmov. Pokračovali sme v implementácii vizualizácie a prepojenia s Moodle. Tokenizér sme postupne rozširovali o podporu zdrojových kódov v jazyku Java. Okrem toho sme pracovali aj na lepšom spracovaní vstupných súborov implementovaním možnosti automatického rozpakovania archívov vo formáte zip a podporou súborov vo formáte doc. Veľkú prioritu sme dávali aj integrácii rôznych súčastí a modulov, ktoré implementovali viacerí členovia tímu a neboli súčasťou aplikácie. Na konci mesiaca sme vykonali viaceré unit testy. Tie odhalili niektoré chyby, ktoré sme následne opravili.

2.3.6 Apríl 2010

V apríli sme vykonávali ďalšie testovania aplikácie a pracovali sme hlavne na dokončení implementácie a oprave chýb. V polke mesiaca sme sa zamerali aj na úpravu dokumentácie, aby zodpovedala aktuálnemu stavu aplikácie. Jednotlivé zápisy zo stretnutí, protokoly testov a ďalšie súčasti sme zahrnuli do riadiacej dokumentácie.

3 Úlohy členov tímu

3.1 Roly a zodpovednosti v tíme

V nasledujúcom zozname sa nachádzajú roly jednotlivých členov tímu v našom projekte. Každý člen má základnú rolu v projekte, špecializáciu a zodpovednosti za určité rámcové aspekty projektu. Implementáciu explicitne nespomíname, nakoľko sa na nej podieľajú všetci členovia tímu.

David Chalupa

- *manažér projektu*
- špecialista na problematiku algoritmov
- zodpovedný za koordináciu projektu ako celku, pridelovanie úloh a koncepcie projektu

Michal Noskovič

- *zástupca manažéra projektu a manažér rizík*
- špecialista na odhady, riziká v projekte a na webové technológie
- zodpovedný za odhady v projekte, správu webovej prezentácie; spolupracuje na plánovaní

Peter Mindek

- *manažér vývoja, plánovač*
- špecialista na architektúry a technologické prístupy
- zodpovedný za dodržiavanie konvencií písania zdrojových kódov a revízie zdrojového kódu

Milan Freml

- *manažér podporných procesov*
- špecialista na Unix-like systémy, multiplatformovosť a otvorené softvérové riešenia
- zodpovedný za softvérovú podporu riadenia a dokumentovania (Redmine, Git, Doxygen)

Matej Sabo

- *softvérový architekt*
- špecialista na modelovanie softvéru
- zodpovedný za dodržiavanie navrhnutých modelov; spolupracuje na testovaní

Marek Mego

- *tester*
- špecialista na testovanie a integráciu systému
- zodpovedný za metodológie prípravy testovacích dát a ich aplikáciu

3.2 Rozdelenie zodpovedností za technickú dokumentáciu

Táto časť opisuje zodpovednosti členov tímu za jednotlivé časti dokumentácie. Vyššia úroveň zodpovedností predstavuje kapitoly dokumentácie, nižšia úroveň menšie časti v rámci kapitol.

Zodpovednosti pre 1. míľnik projektu (odovzdanie v 7. týždni)

Vyššia úroveň zodpovedností

Tab. 3.1. Zodpovednosti za technickú dokumentáciu (vyššia úroveň, 1. odovzdanie)

ID	Názov časti dokumentácie	Zodpovedné osoby
1.	Úvod do problematiky	Michal Noskovič
2.	Metódy a prístupy k detekcii plagiarizmu	Marek Mego
3.	Analýza existujúcich riešení	Milan Freml

4.	Požiadavky na systém	Matej Sabo
5.	Architektonický návrh riešenia	Peter Mindek
-	Typografické úpravy, integrácia dokumentácie	Milan Freml David Chalupa

Nižšia úroveň zodpovedností

Tab. 3.2. Zodpovednosti za technickú dokumentáciu (nižšia úroveň, 1. odovzdanie)

ID	Názov časti dokumentácie	Zodpovedné osoby
1.1.	Čo je plagiarizmus?	Michal Noskovič
1.2.	Formy plagiarizmu a ich špecifiká ¹	Michal Noskovič
1.3.1.	Prehľad metód spracovania zdrojových kódov	Milan Freml
1.3.2.	Prehľad metód spracovania slovenských textov	David Chalupa
2.1.	Predspracovanie zdrojových kódov a dokumentácie ¹	Marek Mego
2.2.	Porovnávanie zdrojových kódov ¹	David Chalupa
2.3.	Porovnávanie slovenských textov ¹	David Chalupa
2.4.	Alternatívne prístupy k detekcii plagiarizmu	Peter Mindek
3.1.1.	SIDPlag	Marek Mego
3.1.2.	JPlag	Matej Sabo
3.1.3.	SIM	David Chalupa
3.1.4.	MOSS	Milan Freml
3.1.5.	YAP	Peter Mindek
3.2.1.	PlaDeS	Michal Noskovič
3.2.2.	Sherlock	Michal Noskovič
4.1.	Ciele projektu	Matej Sabo
4.2.	Analýza požiadaviek	Peter Mindek
4.3.	Model prípadov použitia	Matej Sabo
4.4.	Procesný model	Matej Sabo

Zodpovednosti pre 2. míľnik projektu (odovzdanie v 12. týždni)

Vyššia úroveň zodpovedností

Tab. 3.3. Zodpovednosti za technickú dokumentáciu (vyššia úroveň, 2. odovzdanie)

ID	Názov časti dokumentácie	Zodpovedné osoby
1.	Úvod do problematiky	Michal Noskovič
2.	Metódy a prístupy k detekcii plagiarizmu	Marek Mego
3.	Analýza existujúcich riešení	Milan Freml
4.	Metódy vizualizácie výsledkov	Michal Noskovič
5.	Požiadavky na systém	Matej Sabo
6.	Návrh riešenia ²	Peter Mindek
7.	Implementácia	David Chalupa
8.	Testovanie	Matej Sabo Marek Mego
9.	Zhodnotenie	David Chalupa
A.	Referečná dokumentácia z nástroja Doxygen	Milan Freml
-	Konverzia dokumentácie z DOC do TeX	Milan Freml
-	Typografické úpravy, integrácia dokumentácie	Milan Freml David Chalupa

¹ Platí pre všetky „podkapitoly“ danej kapitoly.

² Názov kapitoly sa počas projektu zmenil z „Architektonický návrh riešenia“ na „Návrh riešenia“.

Nižšia úroveň zodpovedností

Tab. 3.4. Zodpovednosti za technickú dokumentáciu (nižšia úroveň, 2. odovzdanie)

ID	Názov časti dokumentácie	Zodpovedné osoby
7.1.	Jadro aplikácie	Peter Mindek
7.2.	Modul rozhrania pre príkazový riadok	David Chalupa
7.3.	Modul manažmentu porovnávania	Matej Sabo Peter Mindek
7.4.	Modul nahrávania	Milan Freml Michal Noskovič
7.5.1.	Jadro modulu manažmentu analýzy	Michal Noskovič
7.5.2.	Preparácia textu	Michal Noskovič Marek Mego
7.5.3.	Tokenizácia zdrojového kódu	Milan Freml Peter Mindek
7.5.4.	Komparátor pre metódu N-gramy	Marek Mego
7.5.5.	Komparátor pre metódu Greedy String Tiling	David Chalupa
7.5.6.	Komparátor pre algoritmus String-blurring	Peter Mindek

3.3 Rozdelenie zodpovedností za dokumentáciu k riadeniu

V nasledujúcej tabuľke sa nachádza rozdelenie zodpovedností za jednotlivé časti, resp. aspekty dokumentácie k riadeniu nášho projektu.

Tab. 3.5. Zodpovednosti za dokumentáciu k riadeniu projektu

ID	Názov časti dokumentácie	Zodpovedné osoby
1.	Úvod	Matej Sabo
2.	Ponuka	-
3.	Plány	Peter Mindek Michal Noskovič
4.	Úlohy členov tímu	David Chalupa
5.	Zápisy zo stretnutí	všetci
6.	Podporné prostriedky	Milan Freml David Chalupa
7.	Štábná kultúra	Milan Freml Peter Mindek
A.	Ponuka	-
B.	Preberacie protokoly	David Chalupa
-	Typografické úpravy, integrácia dokumentácie	Marek Mego David Chalupa

3.4 Rozdelenie zodpovedností za implementáciu

V nasledujúcich tabuľkách sa nachádza rozdelenie úloh pre konkrétne časti implementovaného riešenia. Vyššia úroveň predstavuje moduly systému, nižšia úroveň menšie súčiastky jednotlivých modulov.

Zodpovednosti pre 1. a 2. míľnik projektu (odovzdanie v 12. týždni)

Vyššia úroveň zodpovedností

Tab. 3.6 Zodpovednosti za implementáciu (úroveň modulov systému)

ID	Názov modulu aplikácie	Zodpovedné osoby
1.	Jadro aplikácie (CORE)	Peter Mindek
2.	Parser manager (PM)	Peter Mindek
3.	Rozhranie pre príkazový riadok (CLI)	David Chalupa
4.	Loader (LOAD)	Milan Freml
5.	Compare manager (CM)	Peter Mindek

Nižšia úroveň zodpovedností

Tab. 3.7 Zodpovednosti za implementáciu (úroveň súčastok modulov)

ID	Názov softvérovej súčastky	Zodpovedné osoby
1.1.	CORE: vstupný bod aplikácie	Peter Mindek
1.2.	CORE: nástroje jadra aplikácie	Peter Mindek
2.1.	PM: jadro modulu Parser Manager (PM CORE)	Michal Noskovič
2.2.	PM: všeobecný komparátor	Peter Mindek
2.3.	PM: komparátor pre metódu N-gramy	Marek Mego
2.4.	PM: komparátor pre metódu Greedy String Tiling (GST)	David Chalupa
2.5.	PM: komparátor pre algoritmus String-blurring	Peter Mindek
3.1.	CLI: súčastka pre spracovanie argumentov	David Chalupa
4.1.	LOAD: základný konvertor (TXT - TXT)	Peter Mindek
4.2.	LOAD: PDF konvertor (PDF - TXT)	Milan Freml
4.3.	LOAD: DOC konvertor (DOC - TXT)	Michal Noskovič
5.1.	CM: kombinátor pre všetky dvojice súborov	Matej Sabo
5.2.	CM: kombinátor pre jeden súbor s ostatnými	Matej Sabo

3.5 Rozdelenie zodpovedností za dokumentáciu v letnom semestri

Zodpovednosti pre technickú dokumentáciu

Vyššia úroveň zodpovedností

Tab. 3.8. Zodpovednosti za technickú dokumentáciu (vyššia úroveň, letný semester)

ID	Názov časti dokumentácie	Zodpovedné osoby
7.	Implementácia	Peter Mindek
8.	Testovanie	Marek Mego
9.	Experimentálne výsledky	David Chalupa
9.	Zhodnotenie	David Chalupa
A.	Používateľská príručka	Matej Sabo
B.	Referečná dokumentácia z nástroja Doxygen	Milan Freml
-	Typografické úpravy, integrácia dokumentácie	Milan Freml David Chalupa

Nižšia úroveň zodpovedností

Tab. 3.9. Zodpovednosti za technickú dokumentáciu (nižšia úroveň, letný semester)

ID	Názov časti dokumentácie	Zodpovedné osoby
7.1.	Jadro aplikácie	Peter Mindek

7.2.	Modul rozhrania pre príkazový riadok	Milan Freml
7.3.	Modul manažmentu porovnávania	Matej Sabo
7.4.	Modul nahrávania	Peter Mindek Marek Mego Milan Freml Michal Noskovič
7.5.1.	Jadro modulu manažmentu analýzy	Peter Mindek Michal Noskovič
7.5.2.	Predspracovanie textu	Michal Noskovič Marek Mego
7.5.3.	Tokenizér zdrojového kódu	Milan Freml Peter Mindek
7.5.4.	Komparátor pre metódu N-gramy	Marek Mego
7.5.5.	Komparátor pre metódu Greedy String Tiling	David Chalupa
7.5.6.	Komparátor pre algoritmus String-blurring	Peter Mindek
7.6.	Komparátor pre metódu LSA	David Chalupa
7.7.	Modul grafického používateľského rozhrania	Peter Mindek

Zodpovednosti pre dokumentáciu k riadeniu

Vyššia úroveň zodpovedností

Tab. 3.10. Zodpovednosti za dokumentáciu k riadeniu projektu v letnom semestri (vyššia úroveň)

ID	Názov časti dokumentácie	Zodpovedné osoby
1.	Úvod	Marek Mego
2.	Plány	Michal Noskovič Peter Mindek Matej Sabo
3.	Úlohy členov tímu	David Chalupa
4.	Zápisy zo stretnutí	všetci
5.	Podporné prostriedky	David Chalupa Milan Freml
6.	Štábná kultúra	Milan Freml Peter Mindek
7.	Testovacia dokumentácia	David Chalupa
A.	Ponuka	-
B.	Preberacie protokoly	David Chalupa
-	Typografické úpravy, integrácia dokumentácie	Marek Mego David Chalupa

Nižšia úroveň zodpovedností

Tab. 3.11. Zodpovednosti za dokumentáciu k riadeniu projektu v letnom semestri (nižšia úroveň)

ID	Názov časti dokumentácie	Zodpovedné osoby
7.1.	Návrh testov	David Chalupa
7.2.	Popis testovacích prípadov a procedúr	David Chalupa
7.3.	Protokoly o vykonaní testov	David Chalupa Marek Mego Matej Sabo Michal Noskovič

3.6 Rozdelenie zodpovedností za implementáciu v letnom semestri

Tab. 3.12. Zodpovednosti za implementáciu (úpravy a rozšírenia)

ID	Názov softvérovej súčiastky	Zodpovedné osoby
1.1.	CORE: vstupný bod aplikácie	Peter Mindek
1.2.	CORE: nástroje jadra aplikácie	Peter Mindek
2.1.	PM: jadro modulu Parser Manager (PM CORE)	Peter Mindek
2.2.	PM: všeobecný komparátor	Peter Mindek
2.3.	PM: komparátor pre metódu N-gramy	Marek Mego
2.4.	PM: komparátor pre metódu Greedy String Tiling (GST)	David Chalupa
2.5.	PM: komparátor pre algoritmus String-blurring	Peter Mindek
3.1.	CLI: súčiastka pre spracovanie argumentov	Milan Freml
4.1.	LOAD: základný konvertor (TXT - TXT)	Peter Mindek
4.2.	LOAD: PDF konvertor (PDF - TXT)	Milan Freml
4.3.	LOAD: DOC konvertor (DOC - TXT)	Milan Freml
5.1.	CM: kombinátor pre všetky dvojice súborov	Matej Sabo
5.2.	CM: kombinátor pre jeden súbor s ostatnými	Matej Sabo
6.	LSA: komparátor pre metódu LSA	David Chalupa
7.1.	MOOD: integrátor s LMS Moodle	Peter Mindek
7.2.	MOOD: PHP skript pre download dát z LMS Moodle	Marek Mego
8.1.	TOK: tokenizér pre C	Peter Mindek Milan Freml
8.2.	TOK: tokenizér pre C - blocknode	Peter Mindek Milan Freml
8.3.	TOK: tokenizér pre jazyk Java	Matej Sabo
9.1.	GUI: hlavné okno	Peter Mindek
9.2.	GUI: vizualizátory	Peter Mindek
9.3.	GUI: integrátor s konzolovou aplikáciou	Peter Mindek

4 Zápisy zo stretnutí

TVORBA INFORMAČNÉHO SYSTÉMU V TÍME I.
TVORBA SOFTVÉROVÉHO SYSTÉMU V TÍME I.

Tím IX
Cypaste

4.1 Zápis zo stretnutia 1

<i>Téma stretnutia</i>	Úvodné stretnutie			
<i>Dátum stretnutia</i>	7.10.2009			
<i>Čas stretnutia</i>	12:00			
<i>Miesto stretnutia</i>	softvérové štúdio			
<i>Účastníci</i>	Milan Freml ✓	David Chalupa ✓	Peter Mindek ✓	
	Marek Mego ✓	Michal Noskovič ✓	Matej Sabo ✓	
<i>Zapisovateľ</i>	David Chalupa			
<i>Vedúca stretnutia</i>	Mgr. Daniela Chudá, PhD.			

Priebeh stretnutia

- ✓ zoznámenie sa s témou projektu
- ✓ rozdelenie konkrétnych úloh na ďalší týždeň
- ✓ dohoda ohľadne spracovávania formálnych záležitostí
- ✓ konzultácia ohľadne „miesta, kde začať“
- ✓ dohoda, že začneme s analýzou existujúcich produktov a metód
- ✓ spomenutie problému podporného prostriedku pre riadenie
- ✓ ďalšie stretnutie - dohodnuté na pondelok

Zhodnotenie doterajšieho pôsobenia

ID	Úloha	Začiatok	Koniec	Stav	Priorita	Zodpovedné osoby
0.1.	Zoznámenie členov tímu	24.9.2009	29.9.2009	✓	Vysoká	Všetci
0.2.	Vypracovanie ponuky	24.9.2009	29.9.2009	✓	Vysoká	Všetci
0.3.	Návrhy na rozdelenie úloh	24.9.2009	7.10.2009	✓	Vysoká	Všetci

Rozdelenie úloh pre nasledujúce obdobie

ID	Úloha	Začiatok	Koniec	Stav	Priorita	Zodpovedné osoby
1.1.	Hodnotenie ponuky	7.10.2009	12.10.2009	Aktívna	Vysoká	Všetci
1.2.	Pozrieť existujúce nástroje	7.10.2009	12.10.2009	Aktívna	Veľmi vysoká	Všetci
1.3.	Hrubý plán	7.10.2009	14.10.2009	Aktívna	Vysoká	Michal Noskovič
1.4.	Požiadavky - kódy	7.10.2009	14.10.2009	Aktívna	Vysoká	Peter Mindek, Marek Mego, Matej Sabo
1.5.	Požiadavky slovenčina	7.10.2009	14.10.2009	Aktívna	Vysoká	David Chalupa, Michal Noskovič, Milan Freml

1.6.	Konceptuálny náčrt architektúry	7.10.2009	14.10.2009	Aktívna	Vysoká	Všetci
------	---------------------------------	-----------	------------	---------	--------	--------

Zhodnotenie stretnutia

Úvodné stretnutie splnilo svoj účel v rozsahu, ktorý bol prijateľný pre členov tímu, ako aj pre vedúcu. Prezreli sme materiály, s ktorými nás oboznámila vedúca a vykonali sme „brainstorming“ niekoľkých zaujímavých inovatívnych myšlienok. Tešíme sa na ďalšie stretnutie, na ktorom sa zoznámime s projektmi, ktoré v danej výskumnej téme na fakulte existujú a ľuďmi, ktorí na nich pracovali, resp. pracujú.

4.2 Zápis zo stretnutia 2

<i>Téma stretnutia</i>	Úvodné stretnutie		
<i>Dátum stretnutia</i>	12.10.2009		
<i>Čas stretnutia</i>	11:30		
<i>Miesto stretnutia</i>	D124		
<i>Účastníci</i>	Milan Freml ✓	David Chalupa ✓	Peter Mindek ✓
	Marek Mego ✓	Michal Noskovič ✓	Matej Sabo ✓
<i>Zapisovateľ</i>	Michal Noskovič		
<i>Vedúca stretnutia</i>	Mgr. Daniela Chudá, PhD.		

Priebeh stretnutia

1. krátke zhrnutie témy projektu
2. zoznámenie sa s riešiteľmi minuloročných tímových projektov
3. krátka diskusia o predspracovávaní slovenského textu
4. ukážka programu PlaDeS
5. návrh využitia slovníka slovenského jazyka
6. ďalšie stretnutie - dohodnuté na stred
7. iniciatíva na výmenu mailov na komunikáciu v budúcnosti

Zhodnotenie doterajšieho pôsobenia

ID	Úloha	Začiatok	Koniec	Stav	Priorita	Zodpovedné osoby
1.1.	Hodnotenie ponuky	7.10.2009	12.10.2009	✓	Vysoká	Všetci
1.2.	Pozrieť existujúce nástroje	7.10.2009	12.10.2009	✓	Veľmi vysoká	Všetci
1.3.	Hrubý plán	7.10.2009	14.10.2009	✓	Vysoká	Michal Noskovič
1.4.	Požiadavky - kódy	7.10.2009	14.10.2009	Aktívna	Vysoká	Peter Mindek, Marek Mego, Matej Sabo
1.5.	Požiadavky - slovenčina	7.10.2009	14.10.2009	Aktívna	Vysoká	David Chalupa, Michal Noskovič, Milan Freml
1.6.	Konceptuálny náčrt architektúry	7.10.2009	14.10.2009	Aktívna	Vysoká	Všetci

Zhodnotenie stretnutia

Toto stretnutie malo za úlohu predstaviť existujúce riešenia vypracované na FIIT STUBA. Ich autori zhrnuli postupy ktoré použili pri ich návrhu a implementácii, pričom načrtli aj ich vlastnosti. Výsledok tohto projektu bol aj odskúšaný na niekoľkých vstupoch. Návrh na spoločnú komunikáciu pri riešení projektu hodnotíme veľmi kladne.

4.3 Zápis zo stretnutia 3

<i>Téma stretnutia</i>	Úvodné stretnutie		
<i>Dátum stretnutia</i>	14.10.2009		
<i>Čas stretnutia</i>	12:00		
<i>Miesto stretnutia</i>	softvérové štúdio		
<i>Účastníci</i>	Milan Freml ✓	David Chalupa ✓	Peter Mindek ✓
	Marek Mego ✓	Michal Noskovič ✓	Matej Sabo ✓
<i>Zapisovateľ</i>	Milan Freml		
<i>Vedúca stretnutia</i>	Mgr. Daniela Chudá, PhD.		

Priebeh stretnutia

1. Zhodnotenie existujúcich nástrojov
2. Diskusia o metodológii vytvárania testovacích dát pre programové kódy
3. Diskusia o metódach plagiarizovania
4. Diskusia k architektúre riešenia
5. Rozdelenie úloh do ďalšieho obdobia
6. Výber podporného prostriedku pre projekt

Zhodnotenie doterajšieho pôsobenia

ID	Úloha	Začiatok	Koniec	Stav	Priorita	Zodpovedné osoby
0.1.	Zoznámenie členov tímu	24.9.2009	29.9.2009	✓	Vysoká	Všetci
0.2.	Vypracovanie ponuky	24.9.2009	29.9.2009	✓	Vysoká	Všetci
0.3.	Návrhy na rozdelenie úloh	24.9.2009	7.10.2009	✓	Vysoká	Všetci
1.1.	Hodnotenie ponuky	7.10.2009	12.10.2009	✓	Vysoká	Všetci
1.2.	Pozrieť existujúce nástroje	7.10.2009	12.10.2009	✓	Veľmi vysoká	Všetci
1.3.	Hrubý plán	7.10.2009	14.10.2009	✓	Vysoká	Michal Noskovič
1.4.	Požiadavky - kódy	7.10.2009	14.10.2009	✓	Vysoká	Peter Mindek, Marek Mego, Matej Sabo
1.5.	Požiadavky - slovenčina	7.10.2009	14.10.2009	✓	Vysoká	David Chalupa, Michal Noskovič, Milan Freml
1.6.	Konceptuálny náčrt architektúry	7.10.2009	14.10.2009	✓	Vysoká	Všetci

Rozdelenie úloh pre nasledujúce obdobie

ID	Úloha	Začiatok	Koniec	Stav	Priorita	Zodpovedné osoby
3.1	Testovacie dáta	14.10.2009	21.10.2009	Aktívna	Vysoká	D. Chudá

	pre slovenské texty					
3.2	Testovacie dáta pre programové kódy - metodológia	14.10.2009	21.10.2009	Aktívna	Vysoká	Marek Mego
3.3	Jplag - test	14.10.2009	28.10.2009	Aktívna	Vysoká	Matej Sabo
3.4	Pozrieť články ohľadom plagiazizmu, Paul Clough, Češka	14.10.2009	21.10.2009	Aktívna	Urgentná	všetci
3.5	Pozrieť Moss, Sim, Sherlock, Yap, Copy, ...	14.10.2009	21.10.2009	Aktívna	Urgentná	všetci
3.7	Vytvoriť diagram architektúry v nejakom nástroji (nie na papieri)	14.10.2009	21.10.2009	Aktívna	Vysoká	všetci
3.8.	Algoritmy LSA, SVD	14.10.2009	21.10.2009	Aktívna	Vysoká	David Chalupa

Zhodnotenie stretnutia

Tretie stretnutie rozhodlo o podpornom prostriedku, rozdelilo ďalšie úlohy a naplnilo nás elánom do ďalšej práce.

4.4 Zápis zo stretnutia 4

<i>Téma stretnutia</i>	Zhrnutie výsledkov analýzy		
<i>Dátum stretnutia</i>	21.10. 2009		
<i>Čas stretnutia</i>	12:00		
<i>Miesto stretnutia</i>	softvérové štúdio		
<i>Účastníci</i>	Milan Freml ✓	David Chalupa ✓	Peter Mindek ✓
	Marek Mego ✓	Michal Noskovič ✓	Matej Sabo ✓
<i>Zapisovateľ</i>	Matej Sabo		
<i>Vedúca stretnutia</i>	Mgr. Daniela Chudá, PhD.		

Priebeh stretnutia

1. *Diskusia o spôsobe implementácie a metodike*
2. *Diskusia o stránke a podpornom prostriedku*
3. *Zhodnotenie existujúcich nástrojov, ich výsledkov n vzorke*
4. *Diskusia k algoritmom Robin-Karp, SVD, LSA, GST*
5. *Rozdelenie úloh do ďalšieho stretnutia a pre tvorbu dokumentu na odovzdanie*

Zhodnotenie doterajšieho pôsobenia

ID	Úloha	Začiatok	Koniec	Stav	Priorita	Zodpovedné osoby
3.1	Testovacie dáta pre slovenské texty	14.10.2009	15.10.2009	✓	Vysoká	D. Chudá
3.2	Testovacie dáta pre programové kódy - metodológia	14.10.2009	20.10.2009	✓	Vysoká	Marek Mego
3.3	Jplag - test	14.10.2009	28.10.2009	Aktívna	Vysoká	Matej Sabo
3.4	Pozrieť články ohľadom plagiarizmu, Paul Clough, Češka	14.10.2009	21.10.2009	✓	Vysoká	všetci
3.5	Pozrieť Moss, Sim, Sherlock, Yap, Copy, ...	14.10.2009	21.10.2009	✓	Vysoká	všetci
3.7	Vytvoriť diagram architektúry v nejakom nástroji (nie na papieri)	14.10.2009		✓	Vysoká	všetci
3.8.	Algoritmy LSA, SVD	14.10.2009	21.10.2009	✓	Vysoká	David Chalupa

Rozdelenie úloh pre nasledujúce obdobie

4.1	Pridať do analýzy popis	21.10.2009	4.11.2009	Aktívna	Stredná	všetci
-----	-------------------------	------------	-----------	---------	---------	--------

	algoritmov					
3.3	Jplag - test	14.10.2009	28.10.2009	Aktívna	Vysoká	Matej Sabo
4.2	Predspracovanie - analýza	21.10.2009	28.10.2009	Aktívna	Vysoká	Michal Noskovič, Marek Mego
4.3	Analýza metód	21.10.2009	28.10.2009	Aktívna	Vysoká	Michal Noskovič, Dávid Chalupa
4.4	Zhrnutie analýz existujúcich produktov	21.10.2009	4.11.2009	Aktívna	Vysoká	Milan Freml
4.5	Špecifikácia požiadaviek	21.10.2009	28.10.2009	Aktívna	Vysoká	Peter Mindek, Matej Sabo
4.6	Vytvorenie šablóny	21.10.2009	28.10.2009	Aktívna	Kritická	Dávid Chalupa
4.7	Výber technologického prístupu	21.10.2009	28.10.2009	Aktívna	Vysoká	všetci

Zhodnotenie stretnutia

Na tomto stretnutí sme zhodnotili existujúce riešenia podľa analýzy, rozdelili sme si ďalšie úlohy a debatovali o našom budúcom magnificentnom nástroji.

4.5 Zápis zo stretnutia 5

<i>Téma stretnutia</i>	Zhrnutie výsledkov analýzy		
<i>Dátum stretnutia</i>	28.10. 2009		
<i>Čas stretnutia</i>	12:00		
<i>Miesto stretnutia</i>	softvérové štúdio		
<i>Účastníci</i>	Milan Freml ✓	David Chalupa ✓	Peter Mindek ✓
	Marek Mego ✓	Michal Noskovič ✓	Matej Sabo ✓
<i>Zapisovateľ</i>	Marek Mego		
<i>Vedúca stretnutia</i>	Mgr. Daniela Chudá, PhD.		

Priebeh stretnutia

1. *Diskusia o experimentálnom prístupe metódou string-blurring (Peter Mindek)*
2. *Diskusia o algoritmoch LSA, SVD, n-gramy*
3. *Prehliadka vytvoreného dokumentu*
4. *Diskusia k použitiu technológií v implementácii*
5. *Rozdelenie úloh do ďalšieho stretnutia a pre tvorbu dokumentu na odovzdanie*

Zhodnotenie doterajšieho pôsobenia

ID	Úloha	Začiatok	Koniec	Stav	Priorita	Zodpovedné osoby
4.1	Pridať do analýzy popis algoritmov	21.10.2009	4.11.2009	✓	Stredná	všetci
3.3	Jplag - test	14.10.2009	28.10.2009	✓	Vysoká	Matej Sabo
4.2	Predspracovanie - analýza	21.10.2009		✓	Vysoká	Michal Noskovič, Marek Mego
4.3	Analýza metód	21.10.2009	28.10.2009	✓	Vysoká	Michal Noskovič, Dávid Chalupa
4.5	Špecifikácia požiadaviek	21.10.2009	28.10.2009	✓	Vysoká	Peter Mindek, Matej Sabo
4.6	Vytvorenie šablóny	21.10.2009	28.10.2009	✓	Kritická	Dávid Chalupa
4.7	Výber technologického prístupu	21.10.2009	28.10.2009	✓	Vysoká	všetci

Rozdelenie úloh pre nasledujúce obdobie

4.4	Zhrnutie analýz existujúcich produktov	21.10.2009	28.10.2009	Aktívna	Vysoká	Milan Freml
5.1	Kompletizácia analýzy LSA	28.10.2009	4.11.2009	Aktívna	Stredná	David Chalupa
5.2	Pridanie popisu diagramov	28.10.2009	4.11.2009	Aktívna	Vysoká	Matej Sabo

5.3	Sprevádzkovanie prostriedku RedMine	28.10.2009	31.10.2009	Aktívna	Kritická	Milan Freml
5.4	Spracovanie lematizácie	28.10.2009	4.11.2009	Aktívna	Vysoká	Marek Mego
5.5	Integrácia dokumentácie	28.10.2009	4.11.2009	Aktívna	Vysoká	Všetci

Zhodnotenie stretnutia

Na piatom stretnutí smerujeme k ukončujeme časť analýzy, venujeme sa možným technologickým prístupom a pomaly sa blížime k implementácii.

4.6 Zápis zo stretnutia 6

<i>Téma stretnutia</i>	Zhrnutie výsledkov analýzy		
<i>Dátum stretnutia</i>	4.11. 2009		
<i>Čas stretnutia</i>	12:00		
<i>Miesto stretnutia</i>	softvérové štúdio		
<i>Účastníci</i>	Milan Freml ✓	David Chalupa ✓	Peter Mindek ✓
	Marek Mego ✓	Michal Noskovič ✓	Matej Sabo ✓
<i>Zapisovateľ</i>	Peter Mindek		
<i>Vedúca stretnutia</i>	Mgr. Daniela Chudá, PhD.		

Priebeh stretnutia

1. *Diskusia k metódam vizualizácie*
2. *Prehliadka odovzdanej dokumentácie*
3. *Diskusia o knižnici GSL*
4. *Diskusia o diskusii*
5. *Diskusia o Architektúre*

Zhodnotenie doterajšieho pôsobenia

ID	Úloha	Začiatok	Koniec	Stav	Priorita	Zodpovedné osoby
4.4	Zhrnutie analýz existujúcich produktov	21.10.2009	28.10.2009	✓	Vysoká	Milan Freml
5.1	Kompletizácia analýzy LSA	28.10.2009	4.11.2009	✓	Stredná	David Chalupa
5.2	Pridanie popisu diagramov	28.10.2009	4.11.2009	✓	Vysoká	Matej Sabo
5.3	Sprevádzkovanie prostriedku RedMine	28.10.2009	31.10.2009	✓	Kritická	Milan Freml
5.4	Spracovanie lematizácie	28.10.2009	4.11.2009	✓	Vysoká	Marek Mego
5.5	Integrácia dokumentácie	28.10.2009	4.11.2009	✓	Vysoká	Všetci

Rozdelenie úloh pre nasledujúce obdobie

ID	Úloha	Začiatok	Koniec	Stav	Priorita	Zodpovedné osoby
6.1	Rešerš metód vizualizácie	4.11.2009	11.11.2009	Aktívna	Vysoká	Všetci
6.2	Analýza knižnice GSL	4.11.2009	11.11.2009	Aktívna	Vysoká	David Chalupa
6.3	Oboznámenie sa s nástrojom Git	4.11.2009	11.11.2009	Aktívna	Vysoká	Všetci
6.4	Rozpracovanie architektúry na	4.11.2009	18.11.2009	Aktívna	Vysoká	Všetci

	nižšiu úroveň					
6.5	Naštudovať prácu s reťazcami v C++	4.11.2009	11.11.2009	Aktívna	Stredná	Všetci
6.6	Analyzovať prístup k Moodle	4.11.2009		Aktívna	Stredná	
6.7	Analyzovať spôsoby rozpoznávania programovacích jazykov	4.11.2009	11.11.2009	Aktívna	Stredná	Peter Mindek
6.8	Základy jadra	4.11.2009	11.11.2009	Aktívna	Normálna	Peter Mindek
6.9	Základy loadera	4.11.2009	18.11.2009	Aktívna	Normálna	Marek Mego Milan Freml
6.10	Navrhnuť parser manager	4.11.2009	18.11.2009	Aktívna	Normálna	Peter Mindek Matej Sabo Michal Noskovič
6.11	Rozbehať GIT	4.11.2009	18.11.2009	Aktívna	Normálna	Milan Freml
6.12	Komparatívna analýza existujúcich produktov	4.11.2009	11.11.2009	Aktívna	Vysoká	David Chalupa Milan Freml
6.13	Navrhnutie konvencií kódu a komentárov	4.11.2009	11.11.2009	Aktívna	Normálna	Peter Mindek Milan Freml

Zhodnotenie stretnutia

Na tomto stretnutí sme zhodnotili existujúce riešenia podľa analýzy, rozdelili sme si ďalšie úlohy a debatovali o našom budúcom magnificentnom nástroji.

4.7 Zápis zo stretnutia 7

<i>Téma stretnutia</i>	Možnosti implementácie		
<i>Dátum stretnutia</i>	11.11.2009		
<i>Čas stretnutia</i>	12:00 - 14:30		
<i>Miesto stretnutia</i>	softvérové štúdio		
<i>Účastníci</i>	Milan Freml ✓	David Chalupa ✓	Peter Mindek ✓
	Marek Mego ✓	Michal Noskovič ✓	Matej Sabo ✓
<i>Zapisovateľ</i>	David Chalupa		
<i>Vedúci stretnutia</i>	Peter Mindek		
<i>Vedúca projektu</i>	Mgr. Daniela Chudá, PhD.		

Priebeh stretnutia

- ✓ P. Mindek prezentoval výsledky analýzy automatického rozpoznávania programovacích jazykov pomocou frekvencií znakov. Výsledky sú celkom zaujímavé, najväčším problémom sa zdá, že je jazyk C. Ako heuristika ale navrhovaná metóda pôsobí zaujímavo.
- ✓ P. Mindek predstavil takisto konvencie písania kódov, na ktorých sme sa dohodli na základe demokratického hlasovania a P. Mindek s M. Fremlom ich spoločne skompletizovali.
- ✓ M. Noskovič prezentoval výsledky svojho bádania v oblasti vizualizácie podobnosti programových kódov. Získali sme informácie o dvoch metódach vizualizácie - grafmi v 2D alebo 3D priestore, resp. pomocou histogramov. Spomenuli sa aj pokročilejšie metódy - arc diagramy a patterngramy.
- ✓ D. Chalupa začal s krátkou ukážkou práce s Git-om a predstavil myšlienku distribuovanej správy verzií.
- ✓ Spomenuli sme knižnicu GNU GSL a diskutovali sme o jej využití na rôznych platformách.
- ✓ D. Chalupa prezentoval výsledky komparatívnej analýzy nástrojov na porovnávanie programových kódov na vytvorených grafoch. Dospeli sme k záveru, že JPlag je najúspešnejší nástroj a že dĺžka zdrojového kódu hrá veľmi dôležitú úlohu pri rozpoznávaní podobnosti.
- ✓ Určili sme, že začneme pracovať na moduloch parser manager-u, jadra, loadera a rozhrania pre príkazový riadok.

Zhodnotenie doterajšieho pôsobenia

ID	Úloha	Začiatok	Koniec	Stav	Priorita	Zodpovedné osoby
6.1	Rešerš metód vizualizácie	4.11.2009	11.11.2009	Splnená	Vysoká	Michal Noskovič
6.2	Analýza knižnice GSL	4.11.2009	11.11.2009	Splnená	Vysoká	David Chalupa
6.3	Oboznámenie sa s nástrojom Git	4.11.2009	11.11.2009	Splnená	Vysoká	Všetci
6.5	Naštudovať prácu s reťazcami v C++	4.11.2009	11.11.2009	Splnená	Stredná	Všetci
6.7	Analýza spôsobov rozpoznávania programovacích jazykov	4.11.2009	11.11.2009	Splnená	Stredná	Peter Mindek
6.8	Základy jadra	4.11.2009	11.11.2009	Splnená	Stredná	Peter Mindek
6.12	Komparatívna analýza existujúcich produktov	4.11.2009	11.11.2009	Splnená	Vysoká	David Chalupa Milan Freml
6.13	Navrhnutie konvencií kódu a komentárov	4.11.2009	11.11.2009	Splnená	Stredná	Peter Mindek Milan Freml

Rozdelenie úloh pre nasledujúce obdobie

ID	Úloha	Začiatok	Koniec	Stav	Priorita	Zodpovedné osoby
6.4	Rozpracovať architektúru na nižšiu úroveň	4.11.2009	18.11.2009	Aktívna	Vysoká	Peter Mindek Matej Sabo
6.6	Analýza prístupu k Moodlu	4.11.2009	Otvorený	Aktívna	Stredná	Marek Mego Milan Freml
6.9	Implementovať základy loadera	4.11.2009	18.11.2009	Aktívna	Stredná	Peter Mindek Milan Freml Marek Mego
6.10	Navrhnuť parser manager	4.11.2009	18.11.2009	Aktívna	Stredná	Peter Mindek Matej Sabo Michal Noskovič
6.11	Nainštalovať a konfigurovať Git	4.11.2009	18.11.2009	Aktívna	Stredná	Milan Freml
7.1.	Analýza možnosti automatického dokumentovania	11.11.2009	18.11.2009	Aktívna	Stredná	David Chalupa
7.2.	Spísať a kategorizovať metódy vizualizácie	11.11.2009	18.11.2009	Aktívna	Urgentná	Michal Noskovič Peter Mindek
7.3.	Porovnať nástroje na slovenské texty	11.11.2009	18.11.2009	Aktívna	Urgentná	Marek Mego Michal Noskovič

7.4.	Prihlásiť sa do súťaže TP Cup	11.11.2009	25.11.2009	Aktívna	Stredná	David Chalupa
7.5.	Upraviť plány	11.11.2009	25.11.2009	Aktívna	Urgentná	Michal Noskovič Peter Mindek
7.6.	Implementovať rozhranie pre príkazový riadok	11.11.2009	18.11.2009	Aktívna	Vysoká	David Chalupa

Zhodnotenie stretnutia

Na stretnutí sme dospeli k rozhodnutiam ohľadne implementácie - začíname s implementáciou loadera, parser managera a jadra systému. Rovnako začneme vytvárať rozhranie aplikácie pomocou príkazového riadka. Chceme sa ďalej zaoberať aj metódami vizualizácie výsledkov. Do budúceho stretnutia chceme začať využívať systém Git. Od tohto zápisu nahradzujeme kvôli problémom s textovým editorom značku ✓ za označenie „splnená“.

4.8 Zápis zo stretnutia 8

<i>Téma stretnutia</i>	Zhodnotenie priebehu implementácie		
<i>Dátum stretnutia</i>	18.11.2009		
<i>Čas stretnutia</i>	12:00 - 14:30		
<i>Miesto stretnutia</i>	softvérové štúdio		
<i>Účastníci</i>	Milan Freml ✓	David Chalupa ✓	Peter Mindek ✓
	Marek Mego ✓	Michal Noskovič ✓	Matej Sabo ✓
<i>Zapisovateľ</i>	Michal Noskovič		
<i>Vedúci stretnutia</i>	David Chalupa		
<i>Vedúca projektu</i>	Mgr. Daniela Chudá, PhD.		

Priebeh stretnutia

- ✓ Mgr. Daniela Chudá navrhla stretnutie s tvorcami programu Plades, ktoré by malo otvoriť otázku vizualizácie.
- ✓ P. Mindek predviedol modul Loader a vysvetlil jeho funkcionality.
- ✓ M. Noskovič objasnil fungovanie modulu Parser Manager, ktorý vytvára vlákna pre porovnanie súborov.
- ✓ Bola otvorená diskusia ohľadom vizualizácie n-gramov. Výsledkom bol návrh ukladať podobné oblasti do poľa, ktoré bude obsahovať jej začiatok a koniec. Diskusia bude pokračovať na stretnutí, ktoré navrhla Mgr. Daniela Chudá.
- ✓ P. Mindek prezentoval svoj návrh na vizualizáciu výsledkov porovnania prostredníctvom fyzikálneho grafu.
- ✓ D. Chalupa oboznámil celý tím s programom doxygen, ktorý budeme používať pri dokumentovaní.
- ✓ M. Noskovič zhrnul možnosti vizualizácie a zhodli sme sa na použití grafu v našom programe.
- ✓ M. Noskovič a P. Mindek prezentovali podrobný plán na najbližšie týždne.
- ✓ M. Mego prezentoval výsledky komparatívnej analýzy nástrojov na porovnanie textov na vytvorených grafoch. Dospeli sme k záveru, že niektoré metódy nedávajú očakávané výsledky.
- ✓ Prebehla diskusia o zmenšení pamäťových nárokov aplikácie. Padol návrh textové súbory nahrávať do pamäte, čím sa ušetrí pamäť.
- ✓ D. Chalupa prezentoval rozhranie pre príkazový riadok a navrhol uprednostniť ovládanie aplikácie prostredníctvom príkazového riadku a vyrobiť grafické rozhranie, ktoré bude jeho rozšírením.

Zhodnotenie doterajšieho pôsobenia

ID	Úloha	Začiatok	Koniec	Stav	Priorita	Zodpovedné osoby
6.4	Rozpracovať architektúru na nižšiu úroveň	4.11.2009	18.11.2009	Splnená	Vysoká	Peter Mindek Matej Sabo

6.6	Analyzovať prístup k Moodlu	4.11.2009	Otvorený	Odložená	Stredná	Marek Mego Milan Freml
6.9	Implementovať základy loadera	4.11.2009	18.11.2009	Splnená	Stredná	Peter Mindek Milan Freml Marek Mego
6.10	Navrhnuť parser manager	4.11.2009	18.11.2009	Splnená	Stredná	Peter Mindek Matej Sabo Michal Noskovič
6.11	Nainštalovať a konfigurovať Git	4.11.2009	18.11.2009	Splnená	Stredná	Milan Freml
7.1.	Analyzovať možnosť automatického dokumentovania	11.11.2009	18.11.2009	Splnená	Stredná	David Chalupa
7.2.	Spísať a kategorizovať metódy vizualizácie	11.11.2009	18.11.2009	Splnená	Urgentná	Michal Noskovič Peter Mindek
7.3.	Porovnať nástroje na slovenské texty	11.11.2009	18.11.2009	Splnená	Urgentná	Marek Mego Michal Noskovič
7.4.	Prihlásiť sa do súťaže TP Cup	11.11.2009	25.11.2009	Odložená	Stredná	David Chalupa
7.5.	Upraviť plány	11.11.2009	25.11.2009	Splnená	Urgentná	Michal Noskovič Peter Mindek
7.6.	Implementovať rozhranie pre príkazový riadok	11.11.2009	18.11.2009	Splnená	Vysoká	David Chalupa

Rozdelenie úloh pre nasledujúce obdobie

ID	Úloha	Začiatok	Koniec	Stav	Priorita	Zodpovedné osoby
7.4	Prihlásiť sa do súťaže TP Cup	18.11.2009	25.11.2009	Aktívna	Urgentná	David Chalupa
8.1	Okomentovať zdrojové kódy	18.11.2009	25.11.2009	Aktívna	Vysoká	Milan Freml
8.2	Pridanie ďalšej porovnavacej metódy GST do Parser Managera	18.11.2009	25.11.2009	Aktívna	Urgentná	David Chalupa
8.3	Okomentovanie zdrojových kódov JavaDocom	18.11.2009	Otvorený	Aktívna	Normálna	Milan Freml
8.4	Predspracovanie slovenských textov: STOP slová + lematizácia	18.11.2009	25.11.2009	Aktívna	Vysoká	Michal Noskovič Milan Freml Marek Mego
8.5	Rozšírenie modulu Compare Manager: porovnávanie každý s každým obojstranne	18.11.2009	25.11.2009	Aktívna	Vysoká	Matej Sabo
8.6	Rozšírenie	18.11.2009	25.11.2009	Aktívna	Vysoká	Matej Sabo

	modulu Compare Manager: porovnávanie jedného súboru s ostatnými					
--	--	--	--	--	--	--

Zhodnotenie stretnutia

Na tomto stretnutí sme zhodnotili doterajší priebeh implementácie. Bol predvedený jej doterajší výsledok, jednoduchý prototyp, ktorý už dokázal porovnávať súbory. Podrobnejšie sme sa zaoberali otázkou vizualizácie a touto témou sa budeme zaoberať aj na ďalších stretnutiach. Našou snahou na nastávajúci týždeň bude pridanie novej porovnávacej metódy a spracovanie slovenského textu.

4.9 Zápis zo stretnutia 9

<i>Téma stretnutia</i>	Informatívne stretnutie o projektoch zaoberajúcich sa plagiarizmom na FIIT		
<i>Dátum stretnutia</i>	23.11.2009		
<i>Čas stretnutia</i>	10:00 - 12:00		
<i>Miesto stretnutia</i>	D124		
<i>Účastníci</i>	Milan Freml ✓	David Chalupa ✓	Peter Mindek ✓
	Marek Mego ✓	Michal Noskovič ✓	Matej Sabo ✓
<i>Ďalší účastníci</i>	Tvorcovia programu Plades, študenti tvoriaci bakalársku a diplomovú prácu u D. Chudej, zaoberajúci sa plagiarizmom, Prof. Ing. Pavol Návrat, PhD.		
<i>Zapisovateľ</i>	Milan Freml		
<i>Vedúci stretnutia</i>	Mgr. Daniela Chudá, PhD.		
<i>Vedúca projektu</i>	Mgr. Daniela Chudá, PhD.		

Priebeh stretnutia

- ✓ Daniela Chudá otvorila stretnutie prezentáciou o plagiarizme všeobecne a uviedla všetkých prítomných vrátane hostí do problematiky, ktorou sa momentálne na FIIT zaoberá okrem nášho tímu aj niekoľko diplomových a bakalárskych prác. Výsledok: opätovné stretnutie prítomných a predstavenie témy prof. Pavlovi Návratovi
- ✓ Jeden zo študentov bakalárskeho štúdia prezentoval výsledky ohľadom výskumu typov plagiátov (viac menej z oblasti tvorby výskumných prác a textov). Výsledok: P. Návrat položil podnetné otázky o možnostiach plagiarizmu textov a poukázal na to, že v každej oblasti sú malé odlišnosti v odkazovaní sa na literatúru a trochu iné štandardy tvorby prác.
- ✓ Milan Freml predniesol metodiku tvorby testovacích dát pre plagiarizované zdrojové kódy v našom tímovom projekte. Stručne zhrnul motiváciu, aké typy plagiátov rozoznávame a aké výsledky sme pomocou nich dosahovali. Výsledok: P. Návratom priniesol nové nápady o možnostiach detekcie pohoršovania zdrojových kódov, keďže to je jedna z metód „tvorby plagiátov“
- ✓ David Chalupa prezentoval čo sme doposiaľ dosiahli v tímovom projekte, stručne zhrnul jednotlivé metódy, ktoré sa používajú na detekciu plagiarizmu a ktoré plánujeme využiť, zobrazil a okomentoval grafy ktoré vzišli z testovania existujúcich programov na detekciu plagiarizmu v zdrojových kódoch. Výsledok: Ukazuje sa, že pohoršovanie kódov je prakticky nedetekovateľnou metódou plagiarizmu, ako aj to, že JPlag a MOSS sú najefektívnejšie nástroje pre detekciu plagiátov.
- ✓ Peter Mindek predstavil svoju metódu na detekciu podobnosti, nazvanú String Blurring. Jedná sa o metaforu známeho rozmazávania z obrázkov, avšak algoritmus rozmazáva text. Ďalej prezentoval aj jednu z možností ako vizualizovať podobnosť dokumentov, metaforu fyzikálnych pružín, kde sila pružiny znamená vyššiu podobnosť. Výsledok:
- ✓ Marek Mego predstavil grafy porovnávajúce existujúce programy pre detekciu plagiarizmu v textoch. Výsledok: Zhodnotenie programov, poukázanie na zlé implementácie niektorých algoritmov v Pladese.
- ✓ Michal Noskovič prezentoval možnosti vizualizácie plagiarizmu. Rozdelil metódy na 2 typy, také, ktoré zobrazujú podobnosť jednotlivých súborov a také, ktoré zobrazujú

priamo podobný text v týchto súboroch. Výsledok: P. Návrat pripomienkoval, že je vhodné zobrazovať aj, na koľko sa text podobá. Avšak protiargumentom je, že väčšina metód rieši skôr zhodu ako podobnosť, prípadne zhodu tokenov. Preto je vhodné vždy používať dobrú terminológiu.

- ✓ Tvorcovia Pladesu priniesli prezentáciu o problémoch, ktoré majú s vizualizáciou. Výsledok: vizualizáciou je vhodné sa zaoberať už pri implementácii metód na detekciu plagiarizmu. Taktiež je nutné používať vhodné štrukturálne prvky GUI, pretože vzniká problém efektívnosti algoritmov.
- ✓ Pavol Humaj na konci stretnutia ešte zhrnul jeho problémy, zaoberá sa najefektívnejším pokrytím reťazcov pri disjunktné množine nájdených zhodných úsekov. Výsledok: návrh aby skúsil použiť zametací algoritmus.

Zhodnotenie doterajšieho pôsobenia

ID	Úloha	Začiatok	Koniec	Stav	Priorita	Zodpovedné osoby
7.4	Prihlásiť sa do súťaže TP Cup	18.11.2009	25.11.2009	Aktívna	Urgentná	David Chalupa
8.1	Okomentovať zdrojové kódy	18.11.2009	2.12.2009	Aktívna	Vysoká	Milan Freml
8.2	Pridanie ďalšej porovnávacíj metódy GST do Parser Managera	18.11.2009	25.11.2009	Aktívna	Urgentná	David Chalupa
8.3	Okomentovanie zdrojových kódov JavaDocom	18.11.2009	25.11.2009	Aktívna	Normálna	Milan Freml
8.4	Predspracovanie slovenských textov: STOP slová + lematizácia	18.11.2009	25.11.2009	Aktívna	Vysoká	Michal Noskovič Milan Freml Marek Mego
8.5	Rozšírenie modulu Compare Manager: porovnávanie každý s každým obojstranne	18.11.2009	25.11.2009	Aktívna	Vysoká	Matej Sabo
8.6	Rozšírenie modulu Compare Manager: porovnávanie jedného súboru s ostatnými	18.11.2009	25.11.2009	Aktívna	Vysoká	Matej Sabo

Zhodnotenie stretnutia

Na tomto viac menej informatívnom stretnutí sa stretli všetci, čo majú niečo spoločné s projektami zaoberajúcimi sa plagiarizmom na FIIT. Boli uvedené typy plagiátov, tvorba testovacích dát v našom tímovom projekte, metódy na detekciu plagiarizmu ako aj metódy na jej vizualizáciu. P. Návrat niekoľkokrát odštartoval podnetnú diskusiu, z ktorej vzišli podnetné nápady. V neposlednom rade mohol tím vidieť, s akými problémami sa stretávajú naši kolegovia pri vizualizácii plagiarizmu ako aj implementácii niektorých nápadov. Našou snahou bude využiť poznatky, ktoré sme vďaka stretnutiu získali na tvorbu nášho nástroja.

4.10 Zápis zo stretnutia 10

<i>Téma stretnutia</i>	Pokračovanie v implementácii		
<i>Dátum stretnutia</i>	25.11.2009		
<i>Čas stretnutia</i>	12:00 - 14:30		
<i>Miesto stretnutia</i>	softvérové štúdio		
<i>Účastníci</i>	Milan Freml ✓	David Chalupa ✓	Peter Mindek ✓
	Marek Mego ✓	Michal Noskovič ✓	Matej Sabo
<i>Zapisovateľ</i>	Marek Mego		
<i>Vedúci stretnutia</i>	Michal Noskovič		
<i>Vedúca projektu</i>	Mgr. Daniela Chudá, PhD.		

Priebeh stretnutia

- ✓ M. Noskovič prezentoval odstraňovanie stop slov programom, ktorý vytvoril. Zistili sme nedostatky z ktorých plynie úloha 9.1.
- ✓ Diskutujeme o možnostiach zachovať naviazanosť upraveného dokumentu na pôvodný dokument po aplikovaní lematizácie, či odobraní stop slov. Je vkladať do dokumentu značky aby naviazanosť bola možná.
- ✓ Diskutujeme o možnostiach tokenizácie zdrojového kódu. Je potrebné tokenizovať aspoň základné inštrukcie spomenuté v dokumentácii. (úloha 9.2)
- ✓ M. Mego prezentoval problémy ktoré nastali pri načítavaní CDB databázy. Je potrebné zistiť či je chyba v API alebo v databáze. (úloha 9.3)
- ✓ D. Chalupa prezentoval výsledky metódy GST ktorú implementoval. Metóda funguje, je potrebné doriešiť spôsob počítania výsledku podobnosti.
- ✓ P. Mindek prezentuje optimalizovanú pamäťovú náročnosť programu, kde vyriešil korektné rušenie vlákien
- ✓ M. Freml prezentuje výsledky komentovania zdrojových kódov. Zdrojové kódy je potrebné komentovať v anglickom jazyku a treba dodržiavať metodiku komentovania navrhnutú P. Mindekom.

Zhodnotenie doterajšieho pôsobenia

ID	Úloha	Začiatok	Koniec	Stav	Priorita	Zodpovedné osoby
7.4	Prihlásiť sa do súťaže TP Cup	18.11.2009	25.11.2009	Splnená	Urgentná	David Chalupa
8.1	Okomentovať zdrojové kódy	18.11.2009	25.11.2009	Čiastoč ne splnená	Vysoká	Milan Freml
8.2	Pridanie dalsej porovnavacej metódy GST do Parser Managera	18.11.2009	25.11.2009	Splnená	Urgentná	David Chalupa
8.3	Okomentovanie zdrojovych kodov JavaDocom	18.11.2009	25.11.2009	Splnená	Normálna	Milan Freml
8.4	Predspracovanie	18.11.2009	25.11.2009	Splnená	Vysoká	Michal Noskovič

	slovenských textov: STOP slová + lematizácia					
8.5	Predspracovanie slovenských textov: lematizácia	18.11.2009	25.11.2009	pokračuje	Vysoká	Marek Mego
8.6	Rozšírenie modulu Compare Manager: porovnávanie každý s každým obojstranne	18.11.2009	25.11.2009	Splnená	Vysoká	Matej Sabo
8.7	Rozšírenie modulu Compare Manager: porovnávanie jedného súboru s ostatnými	18.11.2009	25.11.2009	Splnená	Vysoká	Matej Sabo

Rozdelenie úloh pre nasledujúce obdobie

ID	Úloha	Začiatok	Koniec	Stav	Priorita	Zodpovedné osoby
9.1	Dokončiť implementáciu stop slov vrátane znakov #10 a #13	25.11.2009	2.12.2009	Aktívna	Vysoká	Michal Noskovič
9.2	Implementovať tokenizáciu zdrojových kódov	25.11.2009	2.12.2009	Aktívna	Stredná	Milan Freml
9.3	Predspracovanie slovenských textov: lematizácia	25.11.2009	2.12.2009	Aktívna	Vysoká	Marek Mego
9.4	Analyzovať možnosti pre získanie väčšieho množstva testovacích dát	25.11.2009	2.12.2009	Aktívna	Stredná	David Chalupa
9.5	Pridať prepínač R pre rekurzívne prehľadávanie adresárov	25.11.2009	2.12.2009	Aktívna	Stredná	Peter Mindek
9.6	Pridať prepínač pre výber algoritmu	25.11.2009	2.12.2009	Aktívna	Stredná	Peter Mindek
9.7	Implementovať metódu string blurring	25.11.2009	2.12.2009	Aktívna	Stredná	Peter Mindek
9.8	Dokončiť komentáre ku kódom	25.11.2009	2.12.2009	Aktívna	Stredná	Milan Freml

9.9	Upraviť technickú a riadiacu dok.	25.11.2009	2.12.2009	Aktívna	Stredná	David Chalupa
9.10	Analyzovať prístup k *.doc dokumentom	25.11.2009	2.12.2009	Aktívna	Vysoká	Michal Noskovič
9.11	Analyzovať prístup k Moodle	25.11.2009	2.12.2009	Aktívna	Vysoká	Marek Mego
9.12	Analyzovať a implementovať metódu LCS	25.11.2009	2.12.2009	Aktívna	Stredná	Matej Sabo

Zhodnotenie stretnutia

Na stretnutí sme zhodnotili pokrok v implementácii. Je vytvorený dostatočne robustný základ pre pridávanie ďalších modulov, ktoré rozšíria funkčnosť systému. Na stretnutí sme zadefinovali moduly potrebné pre ďalší vývoj a rozdelili sme si ich implementáciu.

4.11 Zápis zo stretnutia 11

<i>Téma stretnutia</i>	Pokračovanie v implementácii		
<i>Dátum stretnutia</i>	2.12.2009		
<i>Čas stretnutia</i>	12:00 - 14:30		
<i>Miesto stretnutia</i>	softvérové štúdio		
<i>Účastníci</i>	Milan Freml ✓	David Chalupa ✓	Peter Mindek ✓
	Marek Mego ✓	Michal Noskovič ✓	Matej Sabo ✓
<i>Zapisovateľ</i>	Matej Sabo		
<i>Vedúci stretnutia</i>	Marek Mego		
<i>Vedúca projektu</i>	Mgr. Daniela Chudá, PhD.		

Priebeh stretnutia

- ✓ Upravili sme nastavenie skupinového mailu, pretože nám neprechádzali maily od užívateľov, ktorí neboli členmi skupiny.
- ✓ Diskutovali sme k spôsobu tokenizácie pri kontrole plagiarizmu programov
- ✓ Diskutácia k podpornému prostriedku a potrebe doplnenia chýbajúcich údajov.
- ✓ P. Mindek prezentuje nedostatky, ktoré sa vyskytli v syntaxe zdrojového kódu. Potreba opätovného naštudovania metodiky písania programov.
- ✓ Diskutácia ku kompilovaniu kontrolovaných programov a výskyt problémov pri kontrole chybných programov.
- ✓ D. Chalupa prezentuje vzniknuté problémy s dokumentáciami a prechádzaním na Latex. Potreba aktualizácie dokumentácií, s čím súviseli ďalšie úlohy, ktoré sme vytvorili a priradili.
- ✓ Diskutovanie prístupu k systému Moodle s Ing. Steinmullerom.
- ✓ M. Mego prezentoval dosiahnuté výsledky s lematizáciou slovenských textov.
- ✓ Diskutovanie k výsledkom implementácie LCS metódy.
- ✓ M. Noskovič prezentuje výsledky práce na prístupe k *.doc súborom na platformách Windows a Linux.

Zhodnotenie doterajšieho pôsobenia

ID	Úloha	Začiatok	Koniec	Stav	Priorita	Zodpovedné osoby
9.1	Dokončiť implementáciu stop slov vrátane znakov #10 a #13	25.11.2009	2.12.2009	Splnená	Vysoká	Michal Noskovič
9.2	Implementovať tokenizáciu zdrojových kódov	25.11.2009	2.12.2009	Aktívna	Stredná	Milan Freml
9.3	Predspracovanie slovenských textov: lematizácia	25.11.2009	2.12.2009	Splnená	Vysoká	Marek Mego
9.4	Analyzovať možnosti pre	25.11.2009	2.12.2009	Splnená	Stredná	David Chalupa

	získanie väčšieho množstva testovacích dát					
9.5	Pridať prepínač R pre rekurzívne prehľadávanie adresárov	25.11.2009	2.12.2009	Splnená	Stredná	Peter Mindek
9.6	Pridať prepínač pre výber algoritmu	25.11.2009	2.12.2009	Splnená	Stredná	Peter Mindek
9.7	Implementovať metódu string blurring	25.11.2009	2.12.2009	Splnená	Stredná	Peter Mindek
9.8	Dokončiť komentáre ku kódom	25.11.2009	2.12.2009	Splnená	Stredná	Milan Freml
9.9	Upraviť technickú a riadiacu dok.	25.11.2009	2.12.2009	Splnená	Stredná	David Chalupa
9.10	Analyzovať prístup k *.doc dokumentom	25.11.2009	2.12.2009	Splnená	Vysoká	Michal Noskovič
9.11	Analyzovať prístup k Moodle	25.11.2009	2.12.2009	Splnená	Vysoká	Marek Mego
9.12	Analyzovať a implementovať metódu LCS	25.11.2009	2.12.2009	Splnená	Stredná	Matej Sabo

Rozdelenie úloh pre nasledujúce obdobie

ID	Úloha	Začiatok	Koniec	Stav	Priorita	Zodpovedné osoby
9.1	Optimalizovať implementáciu stop slov	2.12.2009	2.2.2010	Aktívna	Nízka	Michal Noskovič
9.2	Implementovať tokenizáciu zdrojových kódov	25.11.2009	9.12.2009	Aktívna	Stredná	Milan Freml
9.12	Optimalizovať implementáciu metódy LCS	2.12.2009	16.12.2009	Aktívna	Stredná	Matej Sabo
10.1	Doplniť chýbajúce údaje do Redminu	2.12.2009	9.12.2009	Aktívna	Vysoká	Milan Freml
10.2	Integrovať všetky obrázky z dokumentácie	2.12.2009	5.12.2009	Aktívna	Urgentná	David Chalupa
10.3	Rozvrhnúť plán na letný semester	2.12.2009	9.12.2009	Aktívna	Stredná	Michal Noskovič, Peter Mindek

10.4	Pridať aktuálny Ganttov diagram a dopísať kapitolu	2.12.2009	9.12.2009	Aktívna	Vysoká	Matej Sabo, Peter Mindek
10.5	Napísať kapitolu o podporných prostriedkoch do dokumentácie	2.12.2009	9.12.2009	Aktívna	Vysoká	Milan Freml
10.6	Napísať konvencie písania dokumentov a komentárov programov do dokumentácie	2.12.2009	9.12.2009	Aktívna	Vysoká	Milan Freml, Peter Mindek
10.7	Napísať konvencie tvorby plagiatov programov do dokumentácie	2.12.2009	9.12.2009	Aktívna	Vysoká	Marek Mego
10.8	Inštalovať systém Moodle, odskúšať si mechaniky a konzultovať s Ing. Steinmullerom	2.12.2009	23.12.2009	Aktívna	Nízka	Marek Mego, Milan Freml
10.9	Spracovať komparatívnu analýzu a dopísať do dokumentácie	2.12.2009	9.12.2009	Aktívna		David Chalupa, Marek Mego
10.10	Napísať kapitolu k vizualizácii	2.12.2009	9.12.2009	Aktívna		Michal Noskovič
10.11	Napísať kapitoly k implementácii, testovaniu a zhodnoteniu do dokumentácie	2.12.2009	9.12.2009	Aktívna		David Chalupa

Zhodnotenie stretnutia

Na stretnutí sme prebrali dosiahnuté výsledky pri tvorbe dokumentácie a implementácie. Prototyp máme takmer dokončený a po doimplementovaní niektorých nedostatkov bude pripravený na odovzdanie. Rozdelili sme si úlohy na ďalší týždeň, ktoré poväčšine zahrňovali písanie a kompletizáciu dokumentácie.

4.12 Zápis zo stretnutia 12

<i>Téma stretnutia</i>	Finalizácia prototypu a dokumentácie		
<i>Dátum stretnutia</i>	9.12.2009		
<i>Čas stretnutia</i>	12:00 - 14:30		
<i>Miesto stretnutia</i>	softvérové štúdio		
<i>Účastníci</i>	Milan Freml ✓	David Chalupa ✓	Peter Mindek ✓
	Marek Mego ✓	Michal Noskovič ✓	Matej Sabo ✓
<i>Zapisovateľ</i>	Peter Mindek		
<i>Vedúci stretnutia</i>	Matej Sabo		
<i>Vedúca projektu</i>	Mgr. Daniela Chudá, PhD.		

Priebeh stretnutia

- ✓ Kontrola projektovej dokumentácie. Tím zistil niekoľko nedostatkov, ktoré boli operatívne riešené.
- ✓ Testovanie prototypu na rôznych počítačoch.
- ✓ Prezentácia prototypu
- ✓ Diskusia o prezentácii
- ✓ Diskusia o neistote ohľadom vlákien a prisľúbenie Petra Mindeka, že to vybaví.

Zhodnotenie doterajšieho pôsobenia

ID	Úloha	Začiatok	Koniec	Stav	Priorita	Zodpovedné osoby
9.1	Optimalizovať implementáciu stop slov	2.12.2009	2.2.2010	Aktívna	Nízka	Michal Noskovič
9.2	Implementovať tokenizáciu zdrojových kódov	25.11.2009	9.12.2009	Aktívna	Stredná	Milan Freml
9.12	Optimalizovať implementáciu metódy LCS	2.12.2009	16.12.2009	Aktívna	Stredná	Matej Sabo
10.1	Doplniť chýbajúce údaje do Redminu	2.12.2009	9.12.2009	Aktívna	Vysoká	Milan Freml
10.2	Integrovať všetky obrázky z dokumentácie	2.12.2009	5.12.2009	Splnená	Urgentná	David Chalupa
10.3	Rozvrhnúť plán na letný semester	2.12.2009	9.12.2009	Splnená	Stredná	Michal Noskovič, Peter Mindek

10.4	Pridať aktuálny Ganttov diagram a dopísať kapitolu	2.12.2009	9.12.2009	Splnená	Vysoká	Matej Sabo, Peter Mindek
10.5	Napísať kapitolu o podporných prostriedkoch do dokumentácie	2.12.2009	9.12.2009	Splnená	Vysoká	Milan Freml
10.6	Napísať konvencie písania dokumentov a komentárov programov do dokumentácie	2.2.2009	9.12.2009	Splnená	Vysoká	Milan Freml, Peter Mindek
10.7	Napísať konvencie tvorby plagiátov programov do dokumentácie	2.2.2009	9.12.2009	Splnená	Vysoká	Marek Mego
10.8	Inštalovať systém Moodle, odskúšať si mechaniky a konzultovať s Ing. Steinmullerom	2.2.2009	23.12.2009	Odložená	Nízka	Marek Mego, Milan Freml
10.9	Spracovať komparatívnu analýzu a dopísať do dokumentácie	2.12.2009	9.12.2009	Splnená		David Chalupa, Marek Mego
10.10	Napísať kapitolu k vizualizácia	2.12.2009	9.12.2009	Splnená		Michal Noskovič
10.11	Napísať kapitoly k implementácii, testovaniu a zhodnoteniu do dokumentácie	2.12.2009	9.12.2009	Splnená		David Chalupa

*Rozdelenie úloh pre nasledujúce obdobie
Zhodnotenie doterajšieho pôsobenia*

ID	Úloha	Začiatok	Koniec	Stav	Priorita	Zodpovedné osoby
9.1	Optimalizovať implementáciu stop slov	2.12.2009	2.2.2010	Aktívna	Nízka	Michal Noskovič
9.2	Implementovať tokenizáciu zdrojových kódov	25.11.2009	2.2.2010	Aktívna	Stredná	Milan Freml
9.12	Optimalizovať implementáciu metódy LCS	2.12.2009	2.2.2010	Aktívna	Stredná	Matej Sabo

10.1	Doplniť chýbajúce údaje do Redminu	2.12.2009	16.2.2010	Aktívna	Vysoká	Milan Freml
11.1	Pripraviť prezentáciu prototypu	9.12.2009	16.12.2009	Aktívna	Vysoká	Matej Sabo, Dávid chalupa

Zhodnotenie stretnutia

Na stretnutí sme zhodnotili vyrobenú dokumentáciu, opravili zopár drobných chýb. Prezentovali sme prototyp a dohodli sa na prezentácii prototypu.

Zápis zo stretnutia 13

<i>Téma stretnutia</i>	Plánovanie priebehu letného semestra		
<i>Dátum stretnutia</i>	23.2.2010		
<i>Čas stretnutia</i>	14:00 - 17:00		
<i>Miesto stretnutia</i>	softvérové štúdio		
<i>Účastníci</i>	Milan Fremi ✓	David Chalupa ✓	Peter Mindek ✓
	Marek Mego ✓	Michal Noskovič ✓	Matej Sabo ✓
<i>Zapisovateľ</i>	David Chalupa		
<i>Vedúci stretnutia</i>	Peter Mindek		
<i>Vedúca projektu</i>	Mgr. Daniela Chudá, PhD.		

Priebeh stretnutia

- ✓ Bola uzavretá dohoda o náhradnom stretnutí namiesto stretnutia, ktoré budúci týždeň nebude. Stretnutie bude v piatok.
- ✓ Milan začal rozprávať o svojich výsledkoch bádania v oblasti tokenizácie. Máme zatiaľ spravený veľmi zjednodušený model tokenizéra. Hovoril tiež o úskaliach, ktorými sú časová náročnosť implementácie a výpočtová efektívnosť.
- ✓ Peter predviedol výsledky svojej práce s GUI. Implementácia prebehla úspešne, rozhranie je použiteľné a veľmi prívetivé. Volanie rozhrania pre príkazový riadok využíva pre potreby GUI nový prepínač.
- ✓ Marek prezentoval výsledky zoznamovania sa so štruktúrou systému Moodle. Máme dve možnosti - buď stiahneme súbory priamo zo súborového systému, alebo vytvoríme modul do systému Moodle.
- ✓ Michal nás oboznámil s optimalizáciou odstraňovania stop-slov. Rozprávali sme sa o možnosti využitia hashovacej tabuľky. Rozhodli sme, že ju využijeme.
- ✓ David prezentoval progres vo svojej práci na metóde LSA. Máme koncepciu implementácie, je možné začať implementovať metódu.
- ✓ Rozprávali sme sa o testovaní. Vyzdvihli sme najmä potrebu zisku ďalších relevantných testovacích dát pre zdrojové kódy.

Zhodnotenie doterajšieho pôsobenia

ID	Úloha	Začiatok	Koniec	Stav	Priorita	Zodpovedné osoby
9.1	Optimalizovať implementáciu stop slov	2.12.2009	2.2.2010	Splnená	Nízka	Michal Noskovič
10.1	Doplniť chýbajúce údaje do Redminu	2.12.2009	16.2.2010	Splnená	Vysoká	Milan Fremi
11.1	Pripraviť prezentáciu prototypu	9.12.2009	16.12.2009	Splnená	Vysoká	Matej Sabo, Dávid chalupa

Rozdelenie úloh pre nasledujúce obdobie

ID	Úloha	Začiatok	Koniec	Stav	Priorita	Zodpovedné osoby
----	-------	----------	--------	------	----------	------------------

9.2	Implementovať tokenizáciu zdrojových kódov	25.11.2009	9.3.2010	Aktívna	Urgentná	Milan Freml, Peter Mindek
9.12	Optimalizovať implementáciu metódy LCS	2.12.2009	9.3.2010	Aktívna	Stredná	Matej Sabo
13.1	Pripraviť podklady pre side-by-side vizualizáciu	23.2.2010	25.2.2010	Aktívna	Vysoká	Michal Noskovič
13.2	Pozrieť, ako riešia side-by-side vizualizáciu iné riešenia	23.2.2010	25.2.2010	Aktívna	Vysoká	Matej Sabo
13.3	Návrh riešenia integrácie s LMS Moodle	23.2.2010	9.3.2010	Aktívna	Vysoká	Marek Mego
13.4	Napísať priebežnú správu na TP Cup	23.2.2010	24.2.2010	Aktívna	Vysoká	David Chalupa
13.5	Návrh riešenia prihlasovania do LMS cez HTTPS	23.2.2010	25.2.2010	Aktívna	Vysoká	Peter Mindek

Zhodnotenie stretnutia

Stretli sme sa, zhodnotili sme progres z obdobia medzi semestrami a vyjadrili sme entuziazmus pre prácu v letnom semestri. Konceptiu implementácie máme hotovú, predbežný plán rovnako. Postupovať budeme najmä podľa urgency jednotlivých záležitostí. Otvorili sme aj otázky konkrétnych možností integrácie s LMS, definitívne rozhodnutie ešte padne. Ďalšie stretnutie bude tento týždeň v piatok.

Zápis zo stretnutia 14

<i>Téma stretnutia</i>	Finalizácia prototypu a dokumentácie		
<i>Dátum stretnutia</i>	26.2.2010		
<i>Čas stretnutia</i>	9:00 - 10:30		
<i>Miesto stretnutia</i>	softvérové štúdio		
<i>Účastníci</i>	Milan Freml ✓	David Chalupa ✓	Peter Mindek ✓
	Marek Mego ✓	Michal Noskovič ✓	Matej Sabo ✓
<i>Zapisovateľ</i>	Michal Noskovič		
<i>Vedúci stretnutia</i>	David Chalupa		
<i>Vedúca projektu</i>	Mgr. Daniela Chudá, PhD.		

Priebeh stretnutia

- ✓ Milan Freml a Peter Mindek prezentovali pokrok v tokenizácii zdrojových kódov. Prebehla diskusia, aké tokeny a koľko používať. Bola predvedená aj názorná ukážka na jednoduchom programe.
- ✓ Matej Sabo prezentoval, akým spôsobom riešia side-by-side vizualizáciu iné programy, pričom navrhol aj možné riešenie, o ktorom sme diskutovali.
- ✓ Marek Mego podrobnejšie popísal fungovanie systému Moodle a spôsob získavania súborov.
- ✓ Michal Noskovič navrhol metódu, pomocou ktorej sa budú dať spätne vizualizovať spracované texty.

Zhodnotenie doterajšieho pôsobenia

ID	Úloha	Začiatok	Koniec	Stav	Priorita	Zodpovedné osoby
9.2	Implementovať tokenizáciu zdrojových kódov	25.11.2009	9.3.2010	Aktívna	Urgentná	Milan Freml Peter Mindek
9.12	Optimalizovať implementáciu metódy LCS	2.12.2009	9.3.2010	Aktívna	Stredná	Matej Sabo
13.10	Pripraviť podklady pre side-by-side vizualizáciu	23.2.2010	25.2.2010	Splnená	Vysoká	Michal Noskovič
13.2	Pozrieť, ako riešia side-by-side vizualizáciu iné riešenia	23.2.2010	25.2.2010	Splnená	Vysoká	Matej Sabo
13.3	Návrh riešenia integrácie s LMS Moodle	23.2.2010	9.3.2010	Aktívna	Vysoká	Marek Mego

13.4	Napísať priebežnú správu na TP Cup	23.2.2010	24.2.2010	Splnená	Vysoká	David Chalupa
13.5	Návrh riešenia prihlasovania do LMS cez HTTPS	23.2.2010	25.2.2010	Aktívna	Vysoká	Peter Mindek

Rozdelenie úloh pre nasledujúce obdobie

ID	Úloha	Začiatok	Koniec	Stav	Priorita	Zodpovedné osoby
14.1	Unit testy	23.2.2010	9.3.2010	Aktívna	Vysoká	Matej Sabo
13.3	Návrh riešenia integrácie s LMS Moodle	23.2.2010	9.3.2010	Aktívna	Vysoká	Marek Mego
14.2	Side-by-side vizualizácia	23.2.2010	9.3.2010	Aktívna	Vysoká	Michal Noskovič
9.2	Implementovať tokenizáciu zdrojových kódov	25.11.2009	9.3.2010	Aktívna	Vysoká	Milan Freml Peter Mindek
9.12	Optimalizovať implementáciu metódy LCS	2.12.2009	9.3.2010	Aktívna	Stredná	Matej Sabo
14.2	Metóda LSA	23.2.2010	9.3.2010	Aktívna	Vysoká	David Chalupa

Zhodnotenie stretnutia

Na stretnutí sme sa venovali najmä problematike tokenizácie a vizualizácie zdrojových kódov. Padlo viacero návrh akým spôsobom spätne zrekonštruovať a vizualizovať upravený text a zdrojové kódy. Na stretnutí sme sa taktiež zaoberali problémom, ktorý vzniká pri spustení nášho programu na operačnom systéme Linux. Príčinu sa budeme snažiť čo najskôr odhaliť a odstrániť.

Zápis zo stretnutia 15

<i>Téma stretnutia</i>	Tokenizácia, vizualizácia, LSA, integrácia, testovanie		
<i>Dátum stretnutia</i>	9.3.2010		
<i>Čas stretnutia</i>	14:00 - 16:30		
<i>Miesto stretnutia</i>	softvérové štúdio		
<i>Účastníci</i>	Milan Freml ✓	David Chalupa ✓	Peter Mindek ✓
	Marek Mego ✓	Michal Noskovič ✓	Matej Sabo ✓
<i>Zapisovateľ</i>	Milan Freml		
<i>Vedúci stretnutia</i>	Michal Noskovič		
<i>Vedúca projektu</i>	Mgr. Daniela Chudá, PhD.		

Priebeh stretnutia

- ✓ Marek Mego zhrnul stav prepojenia na moodle. Výsledok: treba už vyriešiť len problém vylistovania adresára a spakovania súborov.
- ✓ Matej Sabo zhrnul možnosti pre automatické testovanie
- ✓ Milan Freml a Peter Mindek zhrnuli stav tokenizácie. Tokenizátor vie rozoznávať bloky kódu, ostáva už len spraviť regulárne výrazy na matchovanie jednotlivých tokenov. Celkovo sa javí, že tokenizátor na C je na 75% hovový. Už treba dorobiť iba jednotlivé regulárne výrazy na vytvorenie tokenov.
- ✓ Milan Freml informoval o stave konverzie z doc do txt, konverzia na linuxe funguje, problémom ostáva otvorený proces open office, ktorý ostáva otvorený, keďže je spustaný na pozadí. Treba ešte dorobiť to isté pre Windows OS.
- ✓ Peter Mindek a Milan Freml zhodnotili vyriešenie chyby v kóde, ktorý spôsoboval zastavenie vykonávania programu. Bolo to spôsobené zlým čítaním z buffera, ktoré bolo blokujúce a vždy čakalo istý počet riadkov textu, ktoré však v bufferi nemuseli byť. Opravené neblokujúcim načítaním všetkého a následným parsovaním.
- ✓ Michal Noskovič podal správu o vizualizácii a mechanizmu na offset slova v porovnávanom slove. To je potrebné pre spätnú vizualizáciu, kde sa slovo nachádzalo v pôvodnom súbore.
- ✓ David Chalupa zhrnul pokroky v LSA algoritme, ktorý úspešne naimplementoval.

Zhodnotenie doterajšieho pôsobenia

ID	Úloha	Začiatok	Koniec	Stav	Priorita	Zodpovedné osoby
14.1	Unit testy	23.2.2010	9.3.2010	Splnená	Vysoká	Matej Sabo
13.3	Návrh riešenia integrácie s LMS Moodle	23.2.2010	9.3.2010	Splnená	Vysoká	Marek Mego
14.2	Side-by-side vizualizácia	23.2.2010	9.3.2010	Splnená	Vysoká	Michal Noskovič

9.2	Implementovať tokenizáciu zdrojových kódov	25.11.2009	9.3.2010	Splnená	Vysoká	Milan Freml Peter Mindek
9.12	Optimalizovať implementáciu metódy LCS	2.12.2009	9.3.2010	Aktívna	Stredná	Matej Sabo
14.3	Metóda LSA	23.2.2010	9.3.2010	Splnená	Vysoká	David Chalupa

Rozdelenie úloh pre nasledujúce obdobie

ID	Úloha	Začiatok	Koniec	Stav	Priorita	Zodpovedné osoby
15.1	Dokončiť tokenizáciu zdrojových kódov	9.3.2010	14.3.2010	Aktívna	Vysoká	Milan Freml, Peter Mindek
15.2	Optimalizovať vizualizácie	9.3.2010	23.03.10	Aktívna	Stredná	Peter Mindek
15.3	Integrácia s moodlom	2.12.2009	16.3.2010	Aktívna	Vysoká	Marek Mego
15.4	Automatické testovanie	9.3.2010	16.3.2010	Aktívna	Stredná	Matej Sabo
15.5	Nájsť prípadné chyby v implementácii LSA	9.3.2010	16.3.2010	Aktívna	Vysoká	David chalupa
9.12	Optimalizovať metódu LCS	9.3.2010	16.3.2010	Aktívna	Vysoká	Matej Sabo
15.7	Priebežné vypisovanie výsledkov porovnávania, na základe signálov	9.3.2010	23.03.10	Aktívna	Stredná	Milan Freml, Peter Mindek
15.8	Prerobenie algoritmu na side-by-side vizualizáciu	9.3.2010	16.3.2010	Aktívna	Stredná	Michal Noskovič
15.9	Spájanie zdrojových súborov, v prípade situácie 1 adresár = 1 program	9.3.2010	30.03.10	Aktívna	Nízka	Michal Noskovič

15.10	Dokončiť konvertor z doc do txt, kontrolovať spustené procesy	9.3.2010	23.03.10	Aktívna	Stredná	Milan Freml
15.11	Chybová konzola do GUI, výpis varovaní a chýb na chybový výstup	9.3.2010	23.03.10	Aktívna	Stredná	Marek Mego
15.12	Formalizácia procesov	9.3.2010	16.03.10	Aktívna	Stredná	David Chalupa
15.13	Integrácia lematizácie	9.3.2010	16.3.2010	Aktívna	Vysoká	???

Zhodnotenie stretnutia

Na stretnutí sme zhodnotili stav projektu a načerpali nové sily do ďalšej práce, keďže záver sa blíži a s jeho blízkosťou narastá počet úkonov, ktoré je nutné vykonať.

Zápis zo stretnutia 16

<i>Téma stretnutia</i>	Implementácia, integrácia, diskusia ďalšieho postupu		
<i>Dátum stretnutia</i>	16.3.2010		
<i>Čas stretnutia</i>	14:00 - 16:30		
<i>Miesto stretnutia</i>	softvérové štúdio		
<i>Účastníci</i>	Milan Freml ✓	David Chalupa ✓	Peter Mindek ✓
	Marek Mego ✓	Michal Noskovič ✓	Matej Sabo ✓
<i>Zapisovateľ</i>	Matej Sabo		
<i>Vedúci stretnutia</i>	Milan Freml		
<i>Vedúca projektu</i>	Mgr. Daniela Chudá, PhD.		

Priebeh stretnutia

- ✓ Milan Freml a Peter Mindek prezentujú výsledky práce na tokenizácii zdrojového kódu pre jazyk C. Prezentujú akú používali množinu tokenov a postupnosť vytvorenia jednotlivých tokenov zo zdrojového kódu. Toto riešenie problému bolo vyhodnotené a uzavreté.
- ✓ Michal Noskovič programoval nástroj pre vizualizovanie podobného textu, ktorý zakomponujeme v našej aplikácii. Ukázal nám ako sa text vizualizuje a aké technológie použil pri implementácii.
- ✓ Zamysleli sme sa nad myšlienkou vytvoriť taktiež webovú aplikáciu, pre ktorú by sme museli vytvoriť webové užívateľské rozhranie.
- ✓ Riešime postupy ďalšej práce, ako napríklad potrebu integrácie hotových modulov a nástrojov, ktoré ešte nie sú zakomponované v aplikácii.
- ✓ David Chalupa prezentuje výsledky implementácie GST a LSA, ktoré taktiež treba integrovať do aplikácie.
- ✓ Vyriešili sme problém s distribúciou verzií pomocou nástroja pre podporu verziovania GIT, kedy niektoré zdrojové kódy nebolo vidno. Problém bol v tom, že neboli odoslané aj súbory projektu.
- ✓ Marek Mego hovorí o výsledku práce na skripte do systému Moodle, vďaka ktorému budeme môcť sťahovať zadania z tohto systému pre kontrolu plagiarizmu. Vznikol problém pri identifikácii týchto súborov ich autorom, ktorú vyriešime tiež skriptom.
- ✓ Milan Freml vyriešil načítavanie súborov vo formáte .doc, ktorý diskutujeme.
- ✓ Ďalej diskutujeme a prezeráme užívateľské rozhranie aplikácie. Program už vykresluje stavový riadok procesu testovania. Taktiež riešime spôsob komunikácie medzi GUI a jadra aplikácie s algoritmami testovania.
- ✓ Matej Sabo hovorí o výsledkoch testovania. Rozmýšľame čo budeme ďalej testovať a či formou unit testing-u, service testing-u a podobne.
- ✓ Rozdeľujeme úlohy na ďalší týždeň. Úlohy identifikujeme a pridujeme tak, aby sme stihli dokončiť implementáciu v stanovenom termíne. Najdôležitejšie je dokončiť rozrobenú prácu a integrovať aplikáciu.
- ✓ Vyplňujeme dotazník do súťaže TPCup o prínose práce, projektu, informáciách o našom tíme atď.

Zhodnotenie doterajšieho pôsobenia

ID	Úloha	Začiatok	Koniec	Stav	Priorita	Zodpovedné osoby
15.1	Dokončiť tokenizáciu	9.3.2010	14.3.2010	Splnená	Vysoká	Milan Freml, Peter Mindek
15.2	Optimalizovať vizualizácie	9.3.2010	23.3.2010	Aktívna	Stredná	Peter Mindek
15.3	Integrácia s moodlom	2.12.2009	16.3.2010	Splnená	Vysoká	Marek Mego
15.4	Automatické testovanie	9.3.2010	16.3.2010	Aktívna	Stredná	Matej Sabo
15.5	Nájsť prípadné chyby v implementácii LSA	9.3.2010	16.3.2010	Splnená	Vysoká	David chalupa
9.12	Optimalizovať metódu LCS	9.3.2010	16.3.2010	Splnená	Vysoká	Matej Sabo
15.7	Priebežné vypisovanie výsledkov porovnávania, na základe signálov	9.3.2010	23.3.2010	Splnená	Stredná	Milan Freml, Peter Mindek
15.8	Prerobenie algoritmu na side-by-side vizualizáciu	9.3.2010	16.3.2010	Splnená	Stredná	Michal Noskovič
15.9	Spájanie zdrojových súborov, v prípade situácie 1 adresár = 1 program	9.3.2010	30.3.2010	Aktívna	Nízka	Michal Noskovič
15.10	Dokončiť konvertor z doc do txt, kontrolovať spustené procesy	9.3.2010	23.3.2010	Splnená	Stredná	Milan Freml
15.11	Chybová konzola do GUI, výpis varovaní a chýb na chybový výstup	9.3.2010	23.3.2010	Aktívna	Stredná	Marek Mego
15.12	Formalizácia procesov	9.3.2010	16.3.2010	Splnená	Stredná	David Chalupa

15.13	Integrácia lematizácie	9.3.2010	23.3.2010	Aktívna	Vysoká	Milan Freml
-------	------------------------	----------	-----------	---------	--------	-------------

Rozdelenie úloh pre nasledujúce obdobie

ID	Úloha	Začiatok	Koniec	Stav	Priorita	Zodpovedné osoby
16.1	Optimalizovať vizualizácie	9.3.2010	23.3.2010	Aktívna	Stredná	Peter Mindek
16.2	Modifikáciu klientskej časti pre integráciu s Moodle	16.3.2010	30.3.2010	Aktívna	Vysoká	Marek Mego, Peter Mindek
16.3	Vylepšenie integrácie s Moodle o identifikáciu autorov súborov	16.3.2010	30.3.2010	Aktívna	Vysoká	Marek Mego
16.4	Unit test trigramov	16.3.2010	23.3.2010	Aktívna	Vysoká	Matej Sabo
16.5	Spájanie zdrojových súborov, v prípade situácie 1 adresár = 1 program	9.3.2010	30.3.2010	Aktívna	Nízka	Michal Noskovič
16.6	Chybová konzola do GUI, výpis varovaní a chýb na chybový výstup	9.3.2010	23.3.2010	Aktívna	Stredná	Marek Mego
16.7	Integrácia lematizácie	9.3.2010	23.3.2010	Aktívna	Vysoká	Milan Freml
16.8	Revízia abstraktu	16.3.2010	23.3.2010	Aktívna	Stredná	David Chalupa
16.9	Príprava podkladov pre testovacie dáta pre zdrojové kódy	16.3.2010	23.3.2010	Aktívna	Vysoká	David Chalupa
16.10	TP Cup dotazník a fotka	16.3.2010	22.3.2010	Aktívna	Vysoká	David Chalupa

16.11	Integrácia GST	16.3.2010	23.3.2010	Aktívna	Vysoká	Michal Noskovič
16.12	Integrácia tokenizácie	16.3.2010	23.3.2010	Aktívna	Vysoká	Peter Mindek
16.13	Vytvorenie framework-u pre side-by-side vizualizáciu	16.3.2010	23.3.2010	Aktívna	Vysoká	Peter Mindek
16.14	Rozpakovávanie zip súborov	16.3.2010	23.3.2010	Aktívna	Vysoká	Milan Freml
16.15	Riešenie kolízií mien súborov	16.3.2010	29.3.2010	Aktívna	Vysoká	Milan Freml
16.16	Posielanie dát na vizualizovanie na štandardný výstup	16.3.2010	23.3.2010	Aktívna	Vysoká	Peter Mindek
16.17	Aplikácia ľahkých zmien kódu na testovacie zdrojové kódy	16.3.2010	23.3.2010	Aktívna	Vysoká	David Chalupa, Matej Sabo
16.18	Doplnenie názvov súborov do kindershifter	16.3.2010	23.3.2010	Aktívna	Vysoká	Peter Mindek
16.19	Commit-nuť php skript do gitu	16.3.2010	23.3.2010	Aktívna	Vysoká	Marek Mego

Zhodnotenie stretnutia

Na stretnutí sme hlavne prezerali a diskutovali výsledky práce na jednotlivých moduloch, skriptoch, algoritmoch a technológiách. Dosiahli sme výrazný pokrok v implementácií, a preto sme uznali za urgentné tieto moduly postupne integrovať do aplikácie. Dohodli sme sa na ďalšom postupe práce, aby sme stihli implementáciu v stanovenom termíne. Vytvorili sme podľa toho úlohy a priradili dôležitosť. V tejto fáze implementácie sa väčšina práce už musí ale vykonávať prakticky okamžite, aby nedošlo k spomaleniu práce celého tímu .

Zápis zo stretnutia 17

<i>Téma stretnutia</i>	Implementácia, integrácia, diskusia ďalšieho postupu		
<i>Dátum stretnutia</i>	23.3.2010		
<i>Čas stretnutia</i>	14:00 - 16:30		
<i>Miesto stretnutia</i>	softvérové štúdio		
<i>Účastníci</i>	Milan Freml ✓	David Chalupa ✓	Peter Mindek ✓
	Marek Mego ✓	Michal Noskovič ✓	Matej Sabo ✓
<i>Zapisovateľ</i>	Marek Mego		
<i>Vedúci stretnutia</i>	Matej Sabo		
<i>Vedúca projektu</i>	Mgr. Daniela Chudá, PhD. ✓		

Priebeh stretnutia

- ✓ Zamýšľame sa nad efektívnosťou kódu LCS metódy, ktorá sa zdá byť pomalá
- ✓ Marek Mego prezentuje pokrok vo vývoji skriptu v systéme moodle.

- ✓ Matej Sabo hovorí o unit testingu. Treba pokračovať v úsilí a sfunkčniť ohodnotenie výrazov pri testovaní

- ✓ Michal Noskovič vysvetľuje problém pri spájaní zdrojových súborov do jedného. Je potrebné zistiť ako java usporadúva package štruktúru

- ✓ Peter Mindek a Milan Freml diskutujú o riešení chýb vo vizualizácii

- ✓ Peter Mindek optimalizoval vizualizáciu súborov tabuľky

- ✓ Marek Mego a Milan Freml riešia použitie implementácie lematizácie

- ✓ Všetci riešime zhodnosť mien súborov, zbytočné pridávanie txt koncovky a rozípanie súborov a následnú unikátnosť mien súborov

- ✓ Zhodnocujeme pridelené úlohy a ich plnenie

Zhodnotenie doterajšieho pôsobenia

ID	Úloha	Začiatok	Koniec	Stav	Priorita	Zodpovedné osoby
16.1	Optimalizovať vizualizácie	9.3.2010	23.3.2010	Splnená	Stredná	Peter Mindek
16.2	Modifikáciu klientskej časti pre integráciu s Moodle	16.3.2010	30.3.2010	Aktívna	Vysoká	Marek Mego, Peter Mindek
16.3	Vylepšenie integrácie s Moodle o identifikáciu autorov súborov	16.3.2010	30.3.2010	Splnená	Vysoká	Marek Mego
16.4	Unit test trigramov	16.3.2010	23.3.2010	Splnená	Vysoká	Matej Sabo
16.5	Spájanie zdrojových súborov, v prípade situácie 1 adresár = 1 program	9.3.2010	30.3.2010	Splnená	Nízka	Michal Noskovič
16.6	Chybová konzola do GUI, výpis varovaní a chýb na chybový výstup	9.3.2010	23.3.2010	Aktívna	Stredná	Marek Mego
16.7	Integrácia lematizácie	9.3.2010	23.3.2010	Aktívna	Vysoká	Milan Freml
16.8	Revízia abstraktu	16.3.2010	23.3.2010	Splnená	Stredná	David Chalupa
16.9	Príprava podkladov pre testovacie dáta pre zdrojové kódy	16.3.2010	23.3.2010	Splnená	Vysoká	David Chalupa
16.10	TP Cup dotazník a fotka	16.3.2010	22.3.2010	Splnená	Vysoká	David Chalupa
16.11	Integrácia GST	16.3.2010	23.3.2010	Splnená	Vysoká	Michal Noskovič
16.12	Integrácia tokenizácie	16.3.2010	23.3.2010	Splnená	Vysoká	Peter Mindek

16.13	Vytvorenie framework-u pre side-by-side vizualizáciu	16.3.2010	23.3.2010	Splnená	Vysoká	Peter Mindek
16.14	Rozpakovávanie zip súborov	16.3.2010	23.3.2010	Splnená	Vysoká	Milan Freml
16.15	Riešenie kolízií mien súborov	16.3.2010	29.3.2010	Splnená	Vysoká	Milan Freml
16.16	Posielanie dát na vizualizovanie na štandardný výstup	16.3.2010	23.3.2010	Splnená	Vysoká	Peter Mindek
16.17	Aplikácia ľahkých zmien kódu na testovacie zdrojové kódy	16.3.2010	23.3.2010	Splnená	Vysoká	David Chalupa, Matej Sabo
16.18	Doplnenie názvov súborov do kindershifter	16.3.2010	23.3.2010	Splnená	Vysoká	Peter Mindek
16.19	Commit-nuť php skript do gitu	16.3.2010	23.3.2010	Splnená	Vysoká	Marek Mego

Rozdelenie úloh pre nasledujúce obdobie

ID	Úloha	Začiatok	Koniec	Stav	Priorita	Zodpovedné osoby
17.1	G-3-0002 Kindershifter nie je vhodný názov	23.3.2010	30.3.2010	Aktívna	Stredná	David Chalupa
17.2	Modifikáciu klientskej časti pre integráciu s Moodle	16.3.2010	30.3.2010	Aktívna	Vysoká	Marek Mego, Peter Mindek
17.3	D-4-003 GST nefunguje nad reťazcom	23.3.2010	30.3.2010	Aktívna	Vysoká	David Chalupa
17.4	A-1001 Nápoved' v CLI nezodpovedá špecifikácii	23.3.2010	30.3.2010	Aktívna	Vysoká	David Chalupa
17.5	D-4004 FST delí nulou pri porovnávaní prázdnych súborov	23.3.2010	30.3.2010	Aktívna	Stredná	David Chalupa

17.6	Chybová konzola do GUI, výpis varovaní a chýb na chybový výstup	9.3.2010	30.3.2010	Aktívna	Stredná	Marek Mego
17.7	Integrácia lematizácie	9.3.2010	30.3.2010	Aktívna	Vysoká	Milan Freml , Marek Mego
17.8	Implementácia plagiárizatora	23.3.2010	30.3.2010	Aktívna	Stredná	Milan Freml
17.9	Posielanie dát vizualizácie na štandardný výstup	23.3.2010	30.3.2010	Aktívna	Vysoká	Peter Mindek
17.10	Unit test komparátora N gramov	23.3.2010	30.3.2010	Aktívna	Vysoká	Marek Mego
17.11	G 3001 Tabuľka nie je v samostatnom vlákne	23.3.2010	30.3.2010	Aktívna	Vysoká	Peter Mindek
17.12	G 3-003 Grafový vizualizátor má texty mimo určenej plochy	23.3.2010	30.3.2010	Aktívna	Vysoká	Peter Mindek
17.13	D 2-001 Priebeh tokenizácie nie je vizualizovaný	23.3.2010	30.3.2010	Aktívna	Vysoká	Peter Mindek
17.14	Napísanie helpu do Gui	23.3.2010	30.3.2010	Aktívna	Vysoká	Peter Mindek
17.15	Spájanie zdrojových súborov do 1	23.3.2010	30.3.2010	Aktívna	Vysoká	Michal Noskovič
17.16	D-4002 Pri porovnávaní textu s voľbou GST aplikácia padá	23.3.2010	30.3.2010	Aktívna	Vysoká	Michal Noskovič
17.17	Unit test predspracovania textu	23.3.2010	30.3.2010	Aktívna	Vysoká	Michal Noskovič
17.18	Implementácia tokenizéra pre jazyk Java	23.3.2010	30.3.2010	Aktívna	Vysoká	Matej Sabo
17.19	Commit-nuť php skript do gitu	23.3.2010	30.3.2010	Aktívna	Vysoká	Marek Mego
17.20	Text procesu B	23.3.2010	30.3.2010	Aktívna	Normálna	Matej Sabo

Zhodnotenie doterajšieho pôsobenia

Zhodnotenie stretnutia

Na stretnutí sme hlavne prezerali a diskutovali výsledky práce na jednotlivých moduloch, skriptoch, algoritmoch a technológiách. Dosiahli sme výrazný pokrok v implementácií, a preto sme uznali za urgentné tieto moduly postupne integrovať do aplikácie. Dohodli sme sa na ďalšom postupe práce, aby sme stihli implementáciu v stanovenom termíne. Vytvorili sme podľa toho úlohy a priradili dôležitosť. V tejto fáze implementácie sa väčšina práce už musí ale vykonávať prakticky okamžite, aby nedošlo k spomaleniu práce celého tímu .

Zápis zo stretnutia 18

<i>Téma stretnutia</i>	Integrácia, testovanie, diskusia ďalšieho postupu		
<i>Dátum stretnutia</i>	30.3.2010		
<i>Čas stretnutia</i>	14:00 - 16:30		
<i>Miesto stretnutia</i>	softvérové štúdio		
<i>Účastníci</i>	Milan Freml ✓	David Chalupa ✓	Peter Mindek ✓
	Marek Mego ✓	Michal Noskovič ✓	Matej Sabo ✓
<i>Zapisovateľ</i>	Peter Mindek		
<i>Vedúci stretnutia</i>	Marek Mego		
<i>Vedúca projektu</i>	Mgr. Daniela Chudá, PhD. ✓		

Priebeh stretnutia

- ✓ Marek hovorí o postupe s php skriptom na sťahovanie súborov z Moodle
- ✓ Milan hovorí o bottlenecku tokenizéra - diskusia
- ✓ Pokračovanie diskusie o tokenizéry
- ✓ Diskusia o porovnávaní archívu voči ostatným súborom
- ✓ Bug vo vizualizácii side by side
- ✓ Maťo hovorí o tokenizácii Javy

Rozdelenie doterajších úloh

ID	Úloha	Začiatok	Koniec	Stav	Priorita	Zodpovedné osoby
17.1	G-3-0002 Kindershifter nie je vhodný názov	23.3.2010	30.3.2010	Aktívna	Stredná	David Chalupa
17.2	Modifikáciu klientskej časti pre integráciu s Moodle	16.3.2010	30.3.2010	Aktívna	Vysoká	Marek Mego, Peter Mindek
17.3	D-4-003 GST nefunguje nad reťazcom	23.3.2010	30.3.2010	Aktívna	Vysoká	David Chalupa
17.4	A-1001 Nápoveda v CLI nezodpovedá špecifikácii	23.3.2010	30.3.2010	Splnená	Vysoká	David Chalupa
17.5	D-4004 GST delí nulou pri porovnávaní prázdnych súborov	23.3.2010	30.3.2010	Aktívna	Stredná	David Chalupa
17.6	Chybová konzola do GUI, výpis varovaní a chýb na chybový výstup	9.3.2010	30.3.2010	Aktívna	Stredná	Marek Mego

17.7	Integrácia lematizácie	9.3.2010	30.3.2010	Splnená	Vysoká	Milan Freml , Marek Mego
17.8	Implementácia plagiárizatora	23.3.2010	30.3.2010	Aktívna	Stredná	Milan Freml
17.9	Posielanie dát vizualizácie na štandardný výstup	23.3.2010	30.3.2010	Splnená	Vysoká	Peter Mindek
17.10	Unit test komparátora N gramov	23.3.2010	30.3.2010	Aktívna	Vysoká	Marek Mego
17.11	G 3001 Tabuľka nie je v samostatnom vlákne	23.3.2010	30.3.2010	Aktívna	Vysoká	Peter Mindek
17.12	G 3-003 Grafový vizualizátor má texty mimo určenej plochy	23.3.2010	30.3.2010	Splnená	Vysoká	Peter Mindek
17.13	D 2-001 Priebeh tokenizácie nie je vizualizovaný	23.3.2010	30.3.2010	Splnená	Vysoká	Peter Mindek
17.14	Napísanie helpu do Gui	23.3.2010	30.3.2010	Aktívna	Vysoká	Peter Mindek
17.15	Spájanie zdrojových súborov do 1	23.3.2010	30.3.2010	Splnená	Vysoká	Michal Noskovič
17.16	D-4002 Pri porovnávaní textu s voľbou GST aplikácia padá	23.3.2010	30.3.2010	Aktívna	Vysoká	Michal Noskovič
17.17	Unit test predspracovania textu	23.3.2010	30.3.2010	Splnená	Vysoká	Michal Noskovič
17.18	Implementácia tokenizéra pre jazyk Java	23.3.2010	30.3.2010	Aktívna	Vysoká	Matej Sabo
17.19	Commit-nuť php skript do gitu	23.3.2010	30.3.2010	Splnená	Vysoká	Marek Mego
17.20	Test procesu B	23.3.2010	30.3.2010	Aktívna	Normálna	Matej Sabo

Rozdelenie úloh pre nasledujúce obdobie

17.1	G-3-0002 Kindershifter nie je vhodný názov	23.3.2010	30.3.2010	Aktívna	Stredná	David Chalupa
17.2	Modifikáciu klientskej časti pre integráciu s Moodle	16.3.2010	30.3.2010	Aktívna	Vysoká	Marek Mego, Peter Mindek
17.3	D-4-003 GST nefunguje nad reťazcom	23.3.2010	30.3.2010	Aktívna	Vysoká	David Chalupa
17.5	D-4004 GST delí nulou pri porovnávaní prázdných súborov	23.3.2010	30.3.2010	Aktívna	Stredná	David Chalupa
17.6	Chybová konzola do GUI, výpis varovaní a chýb na chybový výstup	9.3.2010	30.3.2010	Aktívna	Stredná	Marek Mego
17.8	Implementácia plagiárizatora	23.3.2010	30.3.2010	Aktívna	Stredná	Milan Fremi
17.10	Unit test komparátora N gramov	23.3.2010	30.3.2010	Aktívna	Vysoká	Marek Mego
17.11	G 3001 Tabuľka nie je v samostatnom vlákne	23.3.2010	30.3.2010	Aktívna	Vysoká	Peter Mindek
17.14	Napísanie helpu do Gui	23.3.2010	30.3.2010	Aktívna	Vysoká	Peter Mindek
17.16	D-4002 Pri porovnávaní textu s voľbou GST aplikácia padá	23.3.2010	30.3.2010	Aktívna	Vysoká	Michal Noskovič
17.18	Implementácia tokenizéra pre jazyk Java	23.3.2010	30.3.2010	Aktívna	Vysoká	Matej Sabo
17.20	Test procesu B	23.3.2010	30.3.2010	Aktívna	Normálna	Matej Sabo
18.1	Testovanie exportu výsledkov GST	30.3.2010	6.4.2010	Aktívna	Vysoká	David Chalupa

Zhodnotenie stretnutia

Prerozdělili sme si úlohy a pokračujeme v dokončovaní aplikácie a v jej testovaní.

Zápis zo stretnutia 19

<i>Téma stretnutia</i>	Rozhodnutie o prioritách pre odovzdanie projektu		
<i>Dátum stretnutia</i>	6.4.2010		
<i>Čas stretnutia</i>	14:00 - 16:30		
<i>Miesto stretnutia</i>	softvérové štúdio		
<i>Účastníci</i>	Milan Fremi ✓	David Chalupa ✓	Peter Mindek ✓
	Marek Mego ✓	Michal Noskovič ✓	Matej Sabo ✓
<i>Zapisovateľ</i>	David Chalupa		
<i>Vedúci stretnutia</i>	Peter Mindek		
<i>Vedúca projektu</i>	Mgr. Daniela Chudá, PhD. ✓		

Priebeh stretnutia

- ✓ Diskutovali sme o nedostatkoch tokenizéra a jeho padaní nad nesprávne vnorenými blokmi kódu. Dohodli sme sa, že sa bude jednať o prioritnú úlohu.
- ✓ Diskutovali sme o kódovaní CDB databázy a problémoch s lematizáciou. Rozhodli sme o tom, že upravíme stratégiu prístupu k dátam na stratégiu bez špeciálneho riešenia kolízií pri prístupe k CDB.
- ✓ Máme problém s vkladáním súboru do ZIP pri sťahovaní dát z LMS Moodle. Riešenie bolo nájdené i prekonzultované.
- ✓ Tabuľka nebude v samostatnom vlákne, úlohu sme zrušili. Aplikácia kvôli tomu veľmi nepredvídateľne reaguje.
- ✓ Rozprávali sme o porovnávaní voči archívu. Treba dorobiť prepínač, ktorý rozhodne, či sa obsah adresára bude spájať do jedného súboru alebo nie.
- ✓ Rozhodli sme, že volanie PHP skriptu budeme vykonávať pomocou prepínača, raz sa vrátia dáta, raz sa vrátia metadáta.

Zhodnotenie doterajšieho pôsobenia

17.2	Modifikáciu klientskej časti pre integráciu s Moodle	16.3.2010	13.4.2010	Aktívna	Vysoká	Marek Mego, Peter Mindek
17.3	D-4-003 GST nefunguje nad reťazcom	23.3.2010	6.4.2010	Splnená	Vysoká	David Chalupa
17.5	D-4004 GST delí nulou pri porovnávaní prázdnych súborov	23.3.2010	6.4.2010	Splnená	Stredná	David Chalupa
17.6	Chybová konzola do GUI, výpis varovaní a chýb na chybový výstup	9.3.2010	13.4.2010	Aktívna	Stredná	Marek Mego

17.8	Implementácia plagiárizatora	23.3.2010	13.4.2010	Aktívna	Stredná	Milan Freml
17.10	Unit test komparátora N gramov	23.3.2010	6.4.2010	Splnená	Vysoká	Marek Mego
17.11	G 3001 Tabuľka nie je v samostatnom vlákne	23.3.2010	6.4.2010	Zrušená	Vysoká	Peter Mindek
17.14	Napísanie helpu do Gui	23.3.2010	6.4.2010	Splnená	Vysoká	Matej Sabo
17.16	D-4002 Pri porovnávaní textu s voľbou GST aplikácia padá	23.3.2010	6.4.2010	Splnená	Vysoká	Michal Noskovič
17.18	Implementácia tokenizéra pre jazyk Java	23.3.2010	13.4.2010	Aktívna	Vysoká	Matej Sabo
18.1	Testovanie exportu výsledkov GST	30.3.2010	6.4.2010	Splnená	Vysoká	David Chalupa

Rozdelenie úloh pre ďalšie obdobie

17.2	Modifikácia klientskej časti pre integráciu s Moodle	16.3.2010	13.4.2010	Aktívna	Vysoká	Peter Mindek, Marek Mego
19.2	D-3-001 Tokenizér padá na zle uzavretých blokoch	6.4.2010	13.4.2010	Aktívna	Normálna	Peter Mindek, Milan Freml
19.1	Vytvorenie logu pre export výsledkov	6.4.2010	13.4.2010	Aktívna	Vysoká	Peter Mindek
19.9	Vytvoriť predbežný návrh postera	6.4.2010	13.4.2010	Aktívna	Vysoká	Peter Mindek
17.8	Implementácia plagiárizatora	23.3.2010	13.4.2010	Aktívna	Stredná	Milan Freml
19.6	Prerobiť prepínač v CLI	6.4.2010	13.4.2010	Aktívna	Vysoká	Milan Freml

19.3	Vytvoriť mapovanie pre vizualizáciu výsledkov N-gramov	6.4.2010	13.4.2010	Aktívna	Vysoká	Michal Noskovič, Matej Sabo
17.6	Chybová konzola do GUI, výpis varovaní a chýb na chybový výstup	9.3.2010	13.4.2010	Aktívna	Stredná	Marek Mego
17.18	Implementácia tokenizéra pre jazyk Java	23.3.2010	13.4.2010	Aktívna	Vysoká	Matej Sabo
19.5	Napísať manuál k aplikácii	6.4.2010	13.4.2010	Aktívna	Vysoká	Matej Sabo
19.4	Napísať pokyny pre dotvorenie dokumentácie	6.4.2010	9.4.2010	Aktívna	Urgentná	David Chalupa
19.8	Integrovať existujúce súčasti do dokumentácie k riadeniu	6.4.2010	13.4.2010	Aktívna	Vysoká	David Chalupa
19.7	Porovnať naše výsledky s inými aplikáciami	6.4.2010	13.4.2010	Aktívna	Vysoká	David Chalupa, Marek Mego

Zhodnotenie stretnutia

Na stretnutí sme zhodnotili naše doterajšie pôsobenie a spomedzi ďalších požadovaných úloh sme vybrali tie, ktoré sme schopní reálne dokončiť do stanoveného termínu (integrácia s LMS Moodle, úprava tokenizéra a dotvorenie dokumentácie do výslednej podoby). Priradili sme im najvyššiu prioritu, ostatné sme posunuli do úzadia.

Zápis zo stretnutia 20

<i>Téma stretnutia</i>	Rozhodnutie o prioritách pre odovzdanie projektu		
<i>Dátum stretnutia</i>	18.4.2010		
<i>Čas stretnutia</i>	14:00 - 16:30		
<i>Miesto stretnutia</i>	softvérové štúdio		
<i>Účastníci</i>	Milan Freml ✓	David Chalupa ✓	Peter Mindek ✓
	Marek Mego ✓	Michal Noskovič ✓	Matej Sabo ✓
<i>Zapisovateľ</i>	Michal Noskovič		
<i>Vedúci stretnutia</i>	David Chalupa		
<i>Vedúca projektu</i>	Mgr. Daniela Chudá, PhD. ✓		

Priebeh stretnutia

- ✓ Na stretnutí sme odovzdali dokumentáciu spolu s programom.
- ✓ Diskutovali sme o úlohách, ktoré treba pred finálnym odovzdaním bezpodmienečne dokončiť.

Zhodnotenie doterajšieho pôsobenia

17.2	Modifikácia klientskej časti pre integráciu s Moodle	16.3.2010	13.4.2010	Aktívna	Vysoká	Peter Mindek, Marek Mego
19.2	D-3-001 Tokenizér padá na zle uzavretých blokoch	6.4.2010	13.4.2010	Aktívna	Normálna	Peter Mindek, Milan Freml
19.1	Vytvorenie logu pre export výsledkov	6.4.2010	13.4.2010	Splnená	Vysoká	Peter Mindek
19.9	Vytvoriť predbežný návrh postera	6.4.2010	13.4.2010	Splnená	Vysoká	Peter Mindek
17.8	Implementácia plagiárizatora	23.3.2010	13.4.2010	Splnená	Stredná	Milan Freml
19.6	Prerobiť prepínač v CLI	6.4.2010	13.4.2010	Splnená	Vysoká	Milan Freml
19.3	Vytvoriť mapovanie pre vizualizáciu výsledkov N-gramov	6.4.2010	13.4.2010	Splnená	Vysoká	Michal Noskovič, Matej Sabo

17.6	Chybová konzola do GUI, výpis varovaní a chýb na chybový výstup	9.3.2010	13.4.2010	Aktívna	Stredná	Marek Mego
17.18	Implementácia tokenizéra pre jazyk Java	23.3.2010	13.4.2010	Splnená	Vysoká	Matej Sabo
19.5	Napísať manuál k aplikácii	6.4.2010	13.4.2010	Splnená	Vysoká	Matej Sabo
19.4	Napísať pokyny pre dotvorenie dokumentácie	6.4.2010	9.4.2010	Splnená	Urgentná	David Chalupa
19.8	Integrovať existujúce súčasti do dokumentácie k riadeniu	6.4.2010	13.4.2010	Splnená	Vysoká	David Chalupa
19.7	Porovnať naše výsledky s inými aplikáciami	6.4.2010	13.4.2010	Splnená	Vysoká	David Chalupa, Marek Mego

Rozdelenie úloh pre ďalšie obdobie

17.6	Chybová konzola do GUI, výpis varovaní a chýb na chybový výstup	9.3.2010	13.4.2010	Aktívna	Stredná	Marek Mego
17.2	Modifikácia klientskej časti pre integráciu s Moodle	16.3.2010	13.4.2010	Aktívna	Vysoká	Peter Mindek, Marek Mego
20,1	Kódovanie diakritiky v aplikácii	18.4.2010	25.4.2010	Aktívna	Vysoká	Milan Freml
20,2	Pridať scrollbar do histogramu	18.4.2010	25.4.2010	Aktívna	Vysoká	Peter Mindek
20,3	Upraviť vizualizáciu s thresholdom a farby a pri side-by-side	18.4.2010	25.4.2010	Aktívna	Vysoká	Michal Noskovič

Zhodnotenie stretnutia

Na stretnutí sme odovzdali dokumentáciu a zhotovený program. Určili sme úlohy, ktoré musia byť do finálneho odovzdania splnené.

5 Podporné prostriedky

5.1 Využívanie služby Google groups

Na manažment v tíme sme predovšetkým spočiatku používali službu Google Groups. Založili sme si skupinu pod názvom `copypaste_2009`, ktorú sme využívali najmä na zdieľanie potrebných materiálov, článkov, textových súborov a ďalších dokumentov. Využili sme aj tímové fórum, ktoré nám táto súžba poskytovala, diskutovali sme v ňom o možnej účasti tímu v súťaži TP Cup. Skupina nám slúžila hlavne vo fáze analýzy, neskôr ju nahradil komplexnejší nástroj Redmine.

5.2 Manažment úloh v nástroji Redmine

V tejto časti sa nachádza návod na pripojenie sa k nástroju Redmine, keďže server, na ktorom beží nemá verejnú IP adresu a je nutné sa pre prístup k systému tunelovať. Ďalej sa tu nachádzajú stručné informácie o používaní samotného systému.

5.2.1 SSH tunel na redmine stroj

Server pre redmine zatiaľ nemá verejnú IP, takže sú 2 možnosti, ako sa dostať k nášmu skvelému podpornému prostriedku:

- použiť ssh tunel a pripojiť sa na redmine cez putty/ssh a vlastný browser
- ak máte konto na WINTERM, použiť RDP login na winterm a pripojiť sa na redmine cez prehliadač na winterme

Obe možnosti sú použiteľné, ale mne je príjemnejší ssh tunel, už len z toho dôvodu, že na *linuxe/unixe* ho vytvoríme príkazom:

```
ssh -L3000:10.62.5.203:3000 freml@labss2.fiit.stuba.sk
```

SSH tunel:

1. Pre *linux/unix* – použite vyššie spomenutý ssh príkaz s vlastným loginom na labss2 a pokračujte bodom 6.
Pre Windows - otvorte/stiahnite putty
2. Pozrite si tento navod: <<http://oldsite.precedence.co.uk/nc/putty.html>>
3. Source port nastavte na 3000, Destination port na 10.62.5.203:3000
4. **DOLEŽITÉ:** kliknite na Add
5. Normálne sa prihláste na labss2
6. Otvorte prhliadač na vašom počítači a namiesto adresy napíšte:
`localhost:3000`
7. Prihláste sa, všetko malými
`login: vaše_priezvisko`
`heslo: vaše_meno`
8. A heslo si pre istotu zmeňte, aby ho nemohol zneužiť niekto iný ako administrátor = Milan Freml, administrátor môže prípadne zabudnuté heslo zmeniť

Poznámka1: ak to nejde, niekde ste spravili chybu...

Poznámka2: počas práce s redmine nevypínajte ssh, lebo stratíte ssh tunel

WINTERM login

1. Prihláste sa cez RDP na winterm.fiit.stuba.sk
2. Otvorte prehliadač na winterme a namiesto adresy napíšte:
`10.62.5.203:3000`

3. Pokračujte bodom 7. z predchádzajúceho návodu, poznámky si nemusíte všímať, toto by malo byť naozaj chybám odolné.

5.2.2 Použitie Redmine

Po úspešnom vytvorení SSH tunelu a prihlásení sa na redmine stojíme pred otázkou, ako ho používať a načo nám je takýto systém dobrý. V tejto časti si teda vysvetlíme, aké časti zo systému redmine sa v našom projekte reálne využívajú.

Moja stránka

Po prihlásení môže každý člen kliknúť v menu na Moja stránka. Na tejto stránke vidí jemu pridelené úlohy, za ktoré je zodpovedný, ich stav. Túto stránku je navyše možné prispôsobiť podľa požiadaviek každého používateľa.

Prehľad

Na tejto karte sa zobrazujú údaje o projekte, členovia, ich úlohy v projekte redmine. Taktiež sa tu nachádza stručná štatistika o úlohách, celkovom čase strávenom nad projektom, je možné zobrazit' gantov diagram, kalendár.

Aktivita

Na tejto karte sa zobrazuje aktivita za posledné dni. V základnom zobrazení sa zobrazuje aktivita v úlohách, repozitári a v súboroch. Je možné zobrazit' aj aktivitu vo wiki a správach.

Úlohy

Asi najzaujímavejšia karta z pohľadu manažmentu úloh v projekte. Tu sa zobrazujú jednotlivé úlohy v tabuľke, ich stav, fronta (frontou rozumieme typ úlohy), priorita, zodpovedný člen tímu, ako aj časové údaje (dokedy má byť splnená, kedy bola naposledy aktualizovaná). Úlohy je možné filtrovať na základe niekoľkých pravidiel, najzaujímavejším je asi podľa fronty a stavu v akom sa úloha nachádza. Po kliknutí na úlohu je možné ju editovať. Ak je úloha ukončená, nastaví sa jej stav na *vyriešený*. Stav *uzavretý* priraduje len tímový vedúci = David Chalupa, prípadne člen, ktorého David špeciálne určí.

Okrem zobrazenia úloh v tabuľke sa na tejto karte dajú zobrazit' aj gantov graf, kalendár a štatistické údaje o náraste počtu úloh za jednotlivé týždne projektu.

Rozoznávame tieto typy úloh:

- *Chyba* - chyba v kóde, prípadne dokumente, v žargóne informatikov sa často používa slovo *bug*.
- *Rozšírenie* - rozšírenie kódu, pridanie nového modulu, implementácia niektorej úlohy
- *Podpora* – všetky úlohy súvisiace s manažmentom projektu, ako aj systémové záležitosti (tvorba web stránky, tvorba metodiky, zabezpečenie servera pre...)
- *Dokumentácia* – všetky úlohy súvisiace s písaním dokumentácie, ak sa nejedná o chybu v dokumentácii

Rozoznávame tieto stavy:

- *Nový* – úloha, ktorá je vytvorená, avšak ešte nemá priradenú zodpovednú osobu
- *Čaká sa* – úloha, ktorá má priradenú zodpovednú osobu a aktuálne sa na nej pracuje, prípadne sa na nej ešte nezačalo pracovať. Tento stav majú všetky úlohy pred vyriešením.
- *Vyriešený* – ak zodpovedná osoba považuje úlohu za vyriešenú, označí stav úlohy za vyriešený.
- *Uzavretý* – David Chalupa, prípadne ním poverená osoba skontroluje vyriešenú úlohu a v prípade, že spĺňa vopred stanovené podmienky, označí ju za uzavretú.
- *Odmietnutý* – ak sa zistí, že úloha je redundantná, zbytočná, prípadne sa projekt pustiť iným smerom, môže sa označiť jej stav za odmietnutý.

Rozoznávame tieto typy priorít:

- *Nízka* – prideliť sa úlohe, ktorá nemusí byť nutne splnená do určeného termínu prípadne úlohe, ktorú považujeme za akýsi bonus navyše, ak sa splní, super, ak nie, v podstate sa nič závažné nestane.

- *Normálna* – základný stav úlohy, pre všetky úlohy, na ktorých nie je projekt extra závislý a je na ich splnenie dostatočný čas.
- *Vysoká* – pre všetky úlohy, ktoré sú nutné k ďalšiemu pokračovaniu v projekte, dátum ich splnenia je záväzný
- *Urgentná* – pre všetky úlohy, ktoré sú nutné k ďalšiemu pokračovaniu v projekte, dátum ich splnenia je záväzný a navyše vzniknú veľké problémy z jeho nedodržania, prípadne je na jej splnenie nižší časový úsek ako obvykle.
- *Okamžitá* – pre úlohy, ktoré sa musia splniť v deň zadania a platí pre ne to čo pre urgentné
- *Včera bolo neskoro* – pre úlohy, ktoré sme objavili, že by bolo vhodné splniť do časového okamihu pred ich zadaním, prípadne úlohy, ktoré boli urgentné/okamžité a ich splnenie sa oneskorilo.

Nová úloha

Umožňuje veľmi jednoduché zadanie úlohy na základe vyplnenia formulára. Nutne využívajte stanovené pravidlá pre priority, stavy a typy úloh. V prípade, že úloha má viac zodpovedných osôb, vyplňte toto pole prázdny políčkou (na to bolo definované).

Wiki

Na wiki sa nachádzajú všetky metodiky a návody nutné k tvorbe projektu. Tieto sú obsiahnuté v kapitole 6 a 7 tejto dokumentácie.

Repozitár

Je veľmi dôležitou súčasťou projektu čo sa implementačnej časti týka. Repozitár na redmine sleduje stav projektového repozitára na gitorious.org (viac v nasledujúcej časti *Manažment verzií v nástroji Git*). Pre sledovanie stavu slúži automatický skript na serveri, kde beží redmine, tento sa spúšťa každú hodinu, preto aj zobrazený repozitár na redmine je aktualizovaný každú hodinu. Keďže štandardne dokáže zobrazovať len jednu branch a rozširovacie moduly pre zobrazenie viacerých branch sa nepodarilo úspešne do projektu začleniť, v súčasnosti je nastavený aby zobrazoval *testing* branch.

Okrem bežného sledovania commitov, rozdielov v súboroch a pod. umožňuje redmine akýkoľvek súbor priamo stiahnuť z webového rozhrania a taktiež umožňuje pridať code review. Code review je súčasťou, ktorá predpokladá veľké využitie v ďalšej fáze projektu, pravidlá pre code review doplníme keď to bude čas vyžadovať.

5.3 Manažment verzií v nástroji Git

V tejto metodike je návod na rozbehnutie Git, ako aj stručné rady ako ho používať. Git je distribuovaný nástroj na manažment verzií súborov/zdrojového kódu. Zo zásady všetky zmeny najprv dávame do testing branch, v prípade ak sa na tom tím dohodne, sa merge testing s master branch. Ale pravidlom bude práca vždy iba s testing branch, ak sa nepovie inak.

Ak používate windows git klienta, všetky príkazy vkladajte cez Shift+Insert, vyhnite sa tak chybám pri vkladaní, orezaní znakov a rozdielom vo windows a unix kódovaní znaku pre koniec riadku. Hlavný výsledok toho, že použijete Shift+Insert bude to, že vám to naozaj pôjde a nebudete to musieť opakovať 3x ako Peťo Mindek.

5.3.1 Návod na rozbehnutie

Tu sa nachádza návod ako spojiť git na systéme windows (niektoré časti platia aj pre unix/linux, svetoznámí ľudia tam však poznajú RTM (Read The Manual)). Stručný návod teda spočíva v týchto krokoch:

1. Stiahnite a nainštalujte git z <http://code.google.com/p/msysgit/downloads/list>
2. Otvorte Git bash Start->All programs->Git->Git bash
3. Do príkazového riadka, ktorý sa otvorí napíšte

```
git config --global user.name "Vase Meno"
git config --global user.email "vas@email.email"
```
4. Vygenerujte si RSA public key:

```
ssh-keygen
```

A potom už len stláčajte enter. Ak chcete, môžete si zadať aj heslo pre šifrovanie, ale potom ho od vás git bude pýtať pri každom push do repozitára.

5. Otvorte súbor `.ssh/id_rsa.pub`

```
cat .ssh/id_rsa.pub
```

6. Prihláste sa na gitorious.org a kliknite na svoj profil (to s tildou), potom na *Manage SSH keys*

7. Skopírujte obsah súboru `.ssh/id_rsa.pub` do formulára na gitorious.org

8. Chodte do adresára kde chcete mať projekt

9. Napíšte:

```
git clone git@gitorious.org:copy-paste/copy-paste.git
```

10. Teraz máte repozitár v adresári `copy-paste`

11. Chodte do `copy-paste` (`cd copy-paste`)

12. Spustite:

```
git checkout -b master origin/master
```

```
git checkout -b release-win origin/release-win
```

```
git checkout -b release-unix origin/release-unix
```

```
git checkout -b testing origin/testing
```

Hlášky o tom, že branch už existuje si nevšímajte

13. Nakoniec pre istotu vykonajte:

```
git checkout testing
```

aby ste pracovali s testing branch, pretože s ňou pracovať máte.

14. Teraz máte repozitár pripravený na prácu

5.3.2 Používané príkazy

Nasledujú príkazy, ktoré budete denne potrebovať na získanie súborov z repozitára, commit a ich poslanie späť na repozitár. Keď teda chcete poslať súbory na gitorious.org, treba vykonať `add->commit->push`, jednotlivé príkazy sú vystetlené nižšie.

Add

Add pridá novo vytvorené súbory do indexu vášho lokálneho repozitára. Vždy keď vytvoríte nové súbory, je vhodné ich okamžite pridať príkazom:

```
git add -A
```

V prípade, že ste *nevytvorili* nové súbory, *add nemá zmysel*, nie je teda potrebné, nutné, dôležité ho vtedy vykonávať. Ak ho však napriek tomu vykonáte, nič sa nestane.

Commit

Commit pridá všetky zmeny v súboroch do vášho lokálneho repozitára, spolu s vašim zmysluplným komentárom. Tento je povinný, pri prázdnom komentári sa zmeny neprejaví a commit nenastane. Preto tam vždy píšete **ZROZUMITELNÉ** hodnotenia. Ak ste naprogramovali nejakú úlohu, uveďte jej názov na začiatku commitu. Commit sa vykoná nasledovne:

```
git commit -a
```

Push

Aby sme zmeny mohli vidieť aj my ostatní a redmine, je potreba spraviť ešte push. Push spôsobí, že súbory sa objavia na gitorious.org. Následne sa dostanú aj na redmine, automatický skript tam každú hodinu kontroluje obsah repozitára na gitorious.org. Je veľmi jednoduché spraviť push nad jednou branch:

```
git push origin testing
```

V prípade, že ste robili zmeny vo viacerých branch naraz, mal by fungovať aj nižšie napísaný príkaz, avšak ten nebol testovaný, keďže sa zatiaľ neukázala jeho potrebnosť... V prípade potreby sa ozvite Milanovi Fremlovi.

```
git push origin
```

Pull

Keď už máte repozitár u seba, je ešte treba ho vždy synchronizovať s gitorious.org pred začiatkom práce. Nieкто totiž mohol zmeniť súbory, ktorých sa práve chcete chytiť a keby ste si nestiahli novú verziu, vznikol by problém. Aby sme sa problémom vyhli a mali u seba vždy najnovšiu verziu nám slúži pull:

```
git pull origin testing
```

Dávajte si však pozor, aby ste boli na správnej branch!!!

5.3.3 Každodenný cyklus

1. Každý deň pred začatím práce je nutné vykonať pull.
2. Ďalej je potrebné čo najčastejšie vykonávať commit (každých zopár hodín = každé 2-3 hodiny, prípadne 4-5 podľa náročnosti úlohy, ktorú vykonávate. Zároveň by mal programátor commitnúť skompilovateľný a funkčný kód. Výnimkou z tohto pravidla je neočakávaný výpadok, prípadne nutnosť spánku:)
3. Aspoň dvakrát za deň vykonať push do gitorious.org v prípade, že ste pracovali... To znamená vykonať push okolo obeda a večer (každý podľa svojho denného režimu).
4. Ak skončíte s prácou, rozhodne treba vykonať push do gitorious.org
5. Rovnako ako dvakrát za deň vykonávať push, vykonať aj pull. Prípadné konflikty budeme zatiaľ riešiť individuálne.

5.4 Linkovanie knižnice GSL v IDE QtCreator

Na implementáciu metódy LSA je v riešení využitá knižnica GNU GSL, ktorú je potrebné do projektu prilinkovať nasledujúcim spôsobom.

1. Otvoriť QtCreator.
2. Otvoriť projekt Ccopypaste.
3. Projects, čo je naľavo -> treba nájsť Build Steps -> QMake -> Additional arguments:
tam dopísať
 - a. na Linuxe LIBS+="-lgsl (pokiaľ nefunguje, skúsiť LIBS+="-lgslcblas LIBS+="-lgsl)
 - b. na Windowse INCLUDEPATH+="../include" LIBS+="../lib/libgsl.a ../lib/libgslcblas.a"

5.5 Použitie knižnice QTestLib

QTestLib je knižnica, ktorá slúži na vykonávanie automatizovaných testov v prostredí Qt. Testovaciu triedu je možné vytvoriť nasledovne. Vykonáme zahrnutie knižnice k súboru pomocou #include:

```
#include <QtTest/QtTest>
```

Definujeme testovaciu triedu, ktorá dedí od triedy QObject. Príslušná metóda, ktorú budeme v rámci testu volať, musí byť definovaná ako private slot napr. nasledovne:

```
class TestQString: public QObject
{
    Q_OBJECT
private slots:
    void toUpper();
};
```

Testovacia metóda potom vykoná samotný automatizovaný test. Korektnosť výstupu testovanej súčiastky môžeme overiť pomocou makra QCOMPARE:

```
void TestQString::toUpper()
{
    QString str = "Hello";
    QCOMPARE(str.toUpper(), QString("HELLO"));
}
```

Telo programu nemusíme v tomto prípade explicitne písať. Vykoná to za nás makro QTEST_MAIN. Na záver je potrebné vykonať zahrnutie meta-object-code súboru, ktorý je generovaný:

```
QTEST_MAIN(TestQString)
#include "testqstring.moc"
```

Takto pripravenú triedu môžeme potom skompilovať. Príslušný *.pro súbor pritom musí obsahovať iba nasledujúce riadky:

```
SOURCES = testqstring.cpp
CONFIG += qtestlib
```

Samozrejme, v prípade použitia viacerých zdrojových súborov je možné *.pro súbor rozšíriť, iné záležitosti by však obsahovať nemal (môže to spôsobiť problémy pri kompilácii). Po uskutočnení týchto krokov je možné vykonať build projektu a následné spustenie testovacieho prípadu. Príslušný protokol s výsledkami možno sledovať dole v konzole QtCreator v časti Application Output.

6 Štábna kultúra

6.1 Konvencie písania dokumentácie k inžinierskemu dielu

V tejto časti sa nachádza metodika pre písanie dokumentácie k inžinierskemu dielu. Keďže sa tým rozhodol prejsť z wordu na typografický systém latex, nachádza sa v nej stručný návod, ako písať zdrojový kód v latexu a ako tvoriť niektoré základné konštrukcie. Samotný text metodiky je nižšie:

Keďže sme sa rozhodli pre latex ako nástroj na tvorbu našej dokumentácie, tu sú stručné návody, ako dokumentáciu písať. Sú rozdelené do niekoľkých sekcií, prosím pozorne prečítať každú.

V prípade výskytu chýb kontaktujte Milana Fremla.

6.1.1 Nadpisy

Latex rozoznáva niekoľko druhov nadpisov, podobne ako MS Word. Pre jednoduchosť budeme uvažovať, že poznáte štýl dokumentu z PSI a úrovne číslovania (hlavne tie úrovne číslovania).

Popis príkazov k jednotlivým úrovňam od najvyššej po najnižšiu. Nad príkazom je vždy náhľad, ako asi bude vyzerat':

Kapitola 1 Názov kapitoly

```
\chapter{Názov kapitoly}
```

1.1 Názov sekcie

```
\section{Názov sekcie}
```

1.1.1 Názov subsekcie

```
\subsection{Názov subsekcie}
```

Nasledovné štýly nadpisov už nie sú číslované, nedostanú sa ani do obsahu:

Názov nadpisu

```
\subsubsection{Názov nadpisu}
```

Názov odstavca text odstavca

```
\paragraph{Názov odstavca}
```

Názov subodstavca text subodstavca

```
\subparagraph{Názov subodstavca}
```

6.1.2 Štruktúra textu

Text je vhodné štrukturovať do odstavcov, ak má text viac ako 8 riadkov bez odstavca, je potrebné ho niekde

začať. Inak sa text zle číta, nemá štruktúru.

Pre vytvorenie *odstavcov* stačí vynechať riadok v texte. Ak ho nevynecháte, odstavec nevytvoríte. Ak len odenterujete, latex vás odignoruje (viď nižšie tvorba riadku).

Lorem ipsum dolor sit amet consectetur est Ut quis dignissim urna. Netus faucibus et sit Nulla habitasse ac nibh sem risus sed. Molestie urna Quisque et lacus Cras sem ac dolor sed Morbi. Suspendisse porttitor sociis pretium est mauris lacinia euismod augue Vestibulum ornare. Vel id elit risus ut Aenean urna in Curabitur tortor convallis. Porta Nam enim convallis Phasellus urna diam massa lacus.

Leo nec ante consectetur aliquam In ut libero pretium magna sollicitudin. Praesent eros tristique mattis parturient libero neque Quisque euismod ante lacus. In convallis volutpat pretium Vivamus sagittis dui leo ac mauris libero. Tellus adipiscing laoreet quis faucibus vitae suscipit Quisque vitae fames porttitor. Curabitur metus pellentesque fermentum orci dolor Nunc Vivamus tellus pretium wisi. Porta at sit et a.

```
Lorem ipsum dolor sit amet consectetur est Ut quis dignissim urna.
Netus faucibus et sit Nulla habitasse
ac nibh sem risus sed. Molestie urna Quisque et lacus Cras sem ac
dolor sed Morbi. Suspendisse porttitor
sociis pretium est mauris lacinia euismod augue Vestibulum ornare. Vel
id elit risus ut Aenean urna in
Curabitur tortor convallis. Porta Nam enim convallis Phasellus urna
diam massa lacus.
```

```
Leo nec ante consectetur aliquam In ut libero pretium magna
sollicitudin. Praesent eros tristique mattis
parturient libero neque Quisque euismod ante lacus. In convallis
volutpat pretium Vivamus sagittis dui leo
ac mauris libero. Tellus adipiscing laoreet quis faucibus vitae
suscipit Quisque vitae fames porttitor.
Curabitur metus pellentesque fermentum orci dolor Nunc Vivamus tellus
pretium wisi. Porta at sit et a.
```

Ak chceme z nejakého dôvodu vložiť nový riadok, urobíme to nasledovne:

Tu je koniec riadku

A tu je ďalší riadok

```
Tu je koniec riadku\\
A tu je ďalší riadok
```

Pokiaľ tam nedáte tie spätné lomítka, latex váš nový riadok odignoruje.

Ďalším vhodným doplnkom je *číslovanie*:

1. Prvá položka
2. Druhá položka
3. Tretia položka

```
\begin{enumerate}
  \item Prvá položka
  \item Druhá položka
  \item Tretia položka
\end{enumerate}
```

Pre *odrážky* je nutné zadať nasledovné:

- Prvá položka

- Druhá položka
- Tretia položka

```
\begin{itemize}
  \item Prvá položka
  \item Druhá položka
  \item Tretia položka
\end{itemize}
```

Posledným typom vymenovávanania je *popis*:

Prvá položka nejaký popis položky
Druhá položka nejaký popis položky
Tretia položka nejaký popis položky

```
\begin{description}
  \item[Prvá položka] nejaký popis položky
  \item[Druhá položka] nejaký popis položky
  \item[Tretia položka] nejaký popis položky
\end{description}
```

6.1.3 Zvýrazňovanie textu

Pre zvýrazňovanie textu sa štandardne používa *kurzíva* alebo jej obdoba. Je možné použiť aj **tučné** písmo, ale nedoporučujem. V neposlednom rade sa používa podčiarknuté písmo a vlastný typ zvýraznenia pre inline kód.

V latexu dosiahneme kurzívu napríklad takto:

Zvýraznený text, normálny text

```
\emph{Zvýraznený text}, normálny text
```

Tučné písmo:

Zvýraznený text, normálny text

```
\textbf{Zvýraznený text}, normálny text
```

Podčiarknuté písmo:

Zvýraznený text, normálny text

```
\underline{Zvýraznený text}, normálny text
```

Inline zdrojový kód:

Inline zdrojový kód, normálny text

```
\texttt{Inline zdrojový kód}, normálny text
```

6.1.4 Obrázky a zdrojový kód

Ak chceme vkladať obrázky, alebo dlhší zdrojový kód, použijeme špeciálne príkazy latexu. Obrázky sa

vkladajú vycentrované, s popisom, možnosťou referencie na obrázok a v prípade prevzatia aj s poznámkou pod čiarou, v ktorej sa nachádza odkaz na zdroj. Scale udáva zväčšenie/zmenšenie obrázku tak, aby sa zmestil na stranu, toto budeme riešiť s Davidom samostatne, ale môžete defaultne dávať 0.7:



Obr. 4: Klasická žiarovka, žrút energie³.

```
\begin{figure} [htp]
\begin{center}
\includegraphics[scale=0.7]{ziarovka.jpg}
\caption[Klasická žiarovka]{Klasická žiarovka, žrút energie}
\footnotemark\label{obrZiarovka}
\end{center}
\end{figure}
\footnotetext{Zdroj:
\url{http://richardwiseman.files.wordpress.com/2009/08/light-bulb-
glowing-filament-light-blue-uncropped-lores-3-ahd.jpg}}
```

Pre vloženie zdrojového kódu sa používa prostredie verbatim, použitie je veľmi jednoduché nasledovne (až na to, že latex nám nezvýrazní syntax použitého jazyka, v prípade potreby takejto vlastnosti ma kontaktujte Milana Fremla, možno niečo vymyslíme):

```
float myRound(float x)
{
    if (x > ROUND1) return (1);
    else if (x < ROUND0) return (0);
    else return (x);
}

\begin{verbatim}
float myRound(float x)
{
    if (x > ROUND1) return (1);
    else if (x < ROUND0) return (0);
    else return (x);
}
\end{verbatim}
```

6.1.5 Referencie, nálepky a bibliografia

Vo veľkých dokumentoch je zvykom odkazovať sa na jednotlivé jeho časti. Pre odkazovanie sa na časti dokumentu v latexu je potrebné poznať 2 príkazy, *label* a *ref*. Použitie vysvetlené na príklade:

³ Zdroj: <<http://richardwiseman.files.wordpress.com/2009/08/light-bulb-glowing-filament-light-blue-uncropped-lores-3-ahd.jpg>>

Dajme tomu, že máme sekciu 1.2 Zdrojové kódy, na ktorú sa chceme odkázať v sekcii 1.3 Slovenské texty. Preto, aby sme sa mohli odkázať na sekciu 1.2, musíme použiť pri deklarácii sekcie 1.2 label a v mieste odkazovania ref. Nutno dodať, že label musí byť pre celý dokument jednoznačný identifikátor, používajte teda rozumné označenia, ktoré sa pravdepodobne nebudú opakovať. (tento príkaz ešte zrejme upresníme:) Nasledovne:

1.2 Zdrojové kódy

1.3 Slovenské texty

V tejto kapitole budeme pokračovať vo vysvetľovaní problému, načrtnutému v kapitole 1.2 - *Zdrojové kódy*.

```
\section{Zdrojové kódy \label{zdrojoveKody}}

\section{1.3 Slovenské texty}
V tejto kapitole budeme pokračovať vo vysvetľovaní problému,
načrtnutému v kapitole \emph{\ref{zdrojoveKody} - Zdrojové kódy}.
```

Rovnakým spôsobom sa odkazujeme aj na literatúru:

Viac je možné zistiť na [2].

```
Viac je možné zistiť na \ref{MOSS}
```

6.1.6 Záver

Táto metodika by vám mala poskytnúť základy pre tvorbu dokumentácie v latexu. V prípade potreby doplním/zmením informácie tak, aby vyhovovali nášmu projektu.

6.2 Konvencie písania zdrojových kódov

V tejto časti sa nachádza metodika pre písanie zdrojových kódov. Sú v nej stručne zhrnuté všetky príkazy ohľadom písania zátvoriek, názvov premenných, indentovania. Zdrojový kód ako aj referenčná dokumentácia bude v anglickom jazyku.

6.2.1 Názvy

Názvy všetkých funkcií, metód, tried a premenných by mali byť v angličtine. Nemali by sa v nich používať skratky, okrem prípadov, keď je význam skratky zrejmý (max, min a pod.). Názvy funkcií a metód by mali byť slovesá a začínať malým písmenom. Ak sú viacslovné, ďalšie slová by mali začínať veľkým písmenom:

```
count();
parseString();
getText();
```

Názvy premenných by mali byť opisné a čo najkratšie. Nemali by sa v nich používať podtržníky:

```
int width;
int totalLength;
```

Výnimkou je premenná `_`, ktorá sa môže používať napríklad ako počítadlo, prípadne ako iná špeciálna alebo pomocná premenná:

```
register int _;
```

Riadiace premenné cyklov, prípadne iné počítadlá, by sa mali označovať písmenami i, j, k, l, m, n, ak sa jedná o celé čísla. Reálne čísla by mali byť označené ako f, g, h, znaky ako c, d, e.

Každá premenná by mala byť deklarovaná na samostatnom riadku, aby k nej bolo možné pridať komentár.

```
int i; //number of cars
int j; //number of people
```

Názvy konštánt by mali byť veľkými písmenami, s prípadným oddelením slov podtržníkmi:

```
#define MAX_WIDTH 3
```

6.2.2 Indentovanie

Jedna úroveň = jeden tabulátor. Nepoužívajte medzery. Vnútro každého bloku kódu je odsadené o jednu úroveň. Blokom sa myslí aj logický blok, teda nie len bloky oddelené zloženými zátvorkami:

```
if (a == 1)
{
    printf("a");
}
```

Riadiace štruktúry by vždy mali obsahovať kód uzavretý do zložených zátvoriek, aj keď to syntax jazyka nevyžaduje:

```
if (a == 1)
    printf("a");    <- nesprávne

if (a == 1) printf("a");    <- nesprávne

if (a == 1)
{
    printf("a");    <- správne
}
```

Switch sa indentuje nasledovne:

```
switch (a)
{
case 0:
    printf("0");
    break;

case 1:
    printf("1");
    break;

default:
    printf("a");
    break;
}
```

Jednotlivé návesia začínajú na rovnakej úrovni, ako switch. Podobne postupujeme pri deklarácii triedy s modifikátormi prístupu:

```

class a
{
public:
    string s;
    int i;
private:
    int p;
};

```

6.2.3 Štruktúra kódu

Každý riadok obsahuje najviac jednu samostatnú operáciu.

```

a++;
a--; //správne

a++; a--; //nesprávne

```

Zložené zátvorky sú vždy na samostatnom riadku:

```

class a
{
    int b();
}

int a::b()
{
    ...
}

int main()
{
    ...
}

if (a < b)
{
    ...
}

if (a < b)
{
    ...
}
else if (a < c)
{
    ...
}
else
{
    ...
}

for (;;)
{

```

```

    ...
}

do
{
    ...
}
while (a == b);

```

Obe zátvorky musia byť na rovnakej indentovacej úrovni, rovnako ako if a príslušné else.

6.2.4 Medzery a prázdne riadky

Prázdne riadky by sa mali používať na oddelenie logických častí kódu. Mali by sa používať za deklaráciami lokálnych premenných, medzi funkciami, triedami a metódami.

Zátvorky by mali nasledovať bezprostredne za menom funkcie:

```

funkcia(); //správne
funkcia (); //nesprávne

```

Medzera by sa vždy mala vkladať medzi kľúčové slovo a zátvorky:

```

while (...);
if (...);
for (...);

```

Pri pretypovávaní by sa medzera mala používať nasledovne:

```

f = (float) i;

```

Medzery by sa mali taktiež vkladať medzi binárne operátory a ich operandy, medzi a výrazy vo for cykloch a medzi parametre funkcií a metód. Nemali by sa naopak nikdy vkladať medzi unárny operátor a jeho operand.

```

a = b + c;
a++;
printf("%d\n", a);
for (i = 0; i < 10; i++);

```

6.3 Konvencie komentovania zdrojových kódov

Komentáre sa píše vždy v anglickom jazyku, keďže celá referenčná dokumentácia je v angličtine a aj celý kód je v angličtine. *Doxygen* komentáre píšeme vždy so začiatočným veľkým písmenom, po každom tagu doxygenu takisto má nasledovať veľké písmeno (až na pomenovania metód, tried, premenných...).

Nad deklaráciou triedy, je *nutné* vložiť komentár v tvare *Doxygenu*, tento sa vkladá do .h súborov. Obzvlášť treba dodržať konvenciu pre indentovanie a písanie tých hviezdíčiek, vzdialenosť textu od hviezdíčiek. Pri triede je nutné vždy uviesť autora kódu (zodopovedný za kvalitu). Ak triedu tvorí viac ľudí naraz, uvedie sa autor pri každej metóde. Triedu teda okomentujeme takto:

```

/** @author Jozko Mrkvicka
 *
 * Management of compare process
 */

```

```
class CompareManager
{
}
```

Nad deklaráciu metódy ktorá je zložitejšia ako `getPremenna` alebo `setPremenna` je *nutné* vložiť komentár v tvare Doxygenu, toto sa vkladá do `.cpp` súborov. V prípade, že má parametre, je nutné ich stručne charakterizovať. Ak metóda vracia hodnotu, je nutné ju tiež stručne charakterizovať.

```
/** @author Jozko Mrkvicka
 *
 * Calculates pi value in precision of scale
 *
 * @param scale Number which defines precision in floating point
 calculation
 *
 * @return Returns calculated double pi
 */
double calculatePi(int scale)
{
}
```

Okomentované by mali byť deklarácie premenných, predovšetkým tie bez opisných názvov. Komentáre by mali vysvetľovať, ČO kód robí, nie AKO to robí:

```
a++; //increment      <- NESPRÁVNE
a++; //pass to next element    <- SPRÁVNE
```

Komentár by sa mal nachádzať na samostatnom riadku pred blokom kódu:

```
//match management
if (a == b)
{
    ...
}

//pass to next element
a++;
```

V prípade jednoriadkových výrazov je možné umiestniť komentár do toho istého riadku:

```
a++; //pass to next element
```

V prípade jednoriadkových doxygen komentárov je možné napísať celý komentár do jedného riadku:

```
/** Default constructor */
CompareManager::CompareManager()
{
```

Doxygen pozná aj html tagy, pomocou nich je teda možné a aj želané dosiahnuť iné riadkovanie výslednej dokumentácie, odrážky, číslovanie...

```
/** Something useful here.....<br>
 * And on another line this
 *
 * Compare methods (as of Nov 21st 2009):
 * <ol>
```

```
* <li>3G (3-grams)</li>  
* <li>GST (Greedy String Tiling)</li>  
* </ol>  
*/
```

V prípade že komentujete súbor, ktorý nemá triedu, aj tento treba okomentovať, slúži na to tag *@file*:

```
/** @file  
* @author David Chalupa  
*  
* Command line interface
```

7 Dokumenty k testovaniu

7.1 Návrh testov

Testy budú prebiehať v dvoch základných fázach.

1. fáza testovania komponentov (unit testy) - Bude pokrývať testy použitých softvérových súčiastok na úrovni tried a bude vykonávaná samotnými autormi príslušných tried.
2. fáza integračného a akceptačného testovania – Integračné a akceptačné testovanie bude prebiehať v spoločnej fáze.

Unit testy budú prebiehať simultánne s vývojom a vstupno-výstupnú špecifikáciu stanovia priamo autori príslušných modulov. Väčšina sa bude vykonávať manuálne, s výnimkou nasledujúcich testov.

1. komparátor N-gramov (comparatorngram.cpp)
2. komparátor GST (comparatorgst.cpp)
3. odstraňovanie stop-slov (textpreparation.cpp)

V rámci integračného a akceptačného testovania sa bude vykonávať testovanie spolupráce jednotlivých súčiastok, korektnosť a prezentácia výsledkov. Tie integračné testy, ktoré sa viažu priamo na zadanie projektu, budú zároveň akceptačnými testami.

- A. načítanie dát
- B. kontrola slovenských textov metódou 3-gramy
- C. kontrola slovenských textov algoritmom String-blurring
- D. kontrola zdrojových kódov metódou Greedy String Tiling
- E. kontrola slovenských textov metódou LSA
- F. GUI

Nájdene nezrovnalosti budú zaznačené do protokolu o vykonaní testu a o prijatí / neprijatí nezrovnalosti ako defektu sa rozhoduje na stretnutí, resp. v urgentných prípadoch o tom rozhoduje operatívne manažér testovania. Reportovanie chýb bude prebiehať pomocou nástroja Redmine, pričom identifikátor chyby je *prípado-podproces-id chyby* a nasleduje stručný popis chyby, napr:

A-4-001 PDF súbor je skonvertovaný do TXT s porušením kódovania

7.2 Popis testovacích prípadov a procedúr

7.2.1 Unit testy

Unit testy, t.j. testy komponentov budú vykonávať priamo vývojári a to zvoleným prístupom (manuálne alebo automaticky). Pre automatické testovanie je predpísané použitie knižnice QTestLib. Reportovanie defektov budú vykonávať v tomto prípade takisto vývojári.

Nasledujúce unit testy sa budú vykonávať automatizovane prostredníctvom QTestLib a kvôli ich dôležitosti sa zapíšu aj do separátnych protokolov o testoch.

Unit test	Komparátor N-gramov (comparatorngram.cpp)
Fáza	unit test
U-1	vstup: dva tokenizované vektory vo formáte QString

	akcia:	vykonanie porovnania pomocou metódy GST, ako je špecifikovaná v analýze
	výstup:	podobnosť v intervale <0,1>

Unit test		Komparátor GST (comparatorgst.cpp)
Fáza		unit test
U-2	vstup:	dva tokenizované vektory vo formáte QString
	akcia:	vykonanie porovnania pomocou metódy GST, ako je špecifikovaná v analýze
	výstup:	podobnosť v intervale <0,1>

Unit test		Odstraňovanie stop-slov (textpreparation.cpp)
Fáza		unit test
U-3	vstup:	súbor vo formáte QString
	akcia:	vykonanie odstránenia stop-slov podľa súboru stop.txt
	výstup:	súbor vo formáte QString, s odstránenými stop-slovami

7.2.2 Prípady A - Načítanie dát

Vstup a výstup procesu

Vstup: požiadavka na načítanie dát

Výstup: množina dát v hlavnej pamäti, všetky súbory sú reťazce

Priebeh procesu

Prípady A		Načítanie dát
Fáza		integračný test, akceptačný test
A-1	vstup:	používateľom vybraný zdroj dát
	akcia:	ak je zdrojom dát Moodle, vykoná sa A-2 ak je zdrojom dát zoznam lokálnych adresárov, vykoná sa A-3 ak je zdrojom dát zoznam lokálnych súborov, vykoná sa A-4
	výstup:	identifikovaný tok dát v procese načítania
A-2	vstup:	požiadavka na kurz, z ktorého chceme dáta
	akcia:	spustí sa download PHP skript
	výstup:	všetko v požadovanom kurze, uložené na disk v jednom adresári, pokračuje sa bodom A-3
A-3	vstup:	požadovaný adresár / požadované adresáre, spôsob prehľadania
	akcia:	prehľadá sa adresár – priamo alebo rekurzívne (podľa požiadavky používateľa)
	výstup:	zoznam požadovaných súborov, pokračuje sa bodom A-4
A-4	vstup:	zoznam požadovaných súborov
	akcia:	determinovanie typu súboru, volanie správneho konvertora
	výstup:	zoznam konvertovaných súborov, pokračuje sa bodom A-5
A-5	vstup:	zoznam konvertovaných súborov
	akcia:	nahratie súborov do pamäte
	výstup:	načítané dáta, informácia o úspešnom zbehnutí procesu

7.2.3 Prípady B – Kontrola slovenských textov metódou 3-gramy

Vstup a výstup procesu

Vstup: požiadavka na dáta na testovanie, mód párovania súborov, prah podobnosti

Výstup: percentuálne ohodnotenia podobnosti, podozrivé dvojice, vizualizované podozrenia

Priebeh procesu

Prípado B		Kontrola slovenských textov metódou 3-gramy
Fáza		integračný test, akceptačný test
B-1	vstup:	používateľ vyberie metódu 3-gramy, mód párovania, prah podobnosti a identifikuje zdroj dát
	akcia:	GUI odošle požiadavku na konzolovú aplikáciu, vykoná sa proces A
	výstup:	načítané dáta v hlavnej pamäti, pokračuje sa bodom B-2
B-2	vstup:	načítané dáta v hlavnej pamäti
	akcia:	vyhádžanie bielych znakov a stop-slov
	výstup:	dáta bez stop-slov v hlavnej pamäti, pokračuje sa bodom B-3
B-3	vstup:	dáta bez stop-slov v hlavnej pamäti
	akcia:	lematizácia
	výstup:	predspracované dáta hlavnej pamäti, pokračuje sa bodom B-4
B-4	vstup:	predspracované dáta hlavnej pamäti
	akcia:	spárovanie súborov podľa požadovaného módu
	výstup:	dvojice súborov, pokračuje sa bodom B-5
B-5	vstup:	dvojice súborov
	akcia:	volanie komparátora pre 3-gramy nad definovanými dvojicami
	výstup:	ohodnotenie podobnosti, pokračuje sa bodom B-6
B-6	vstup:	ohodnotenie podobnosti
	akcia:	odoslanie údajov do GUI
	výstup:	údaj v GUI, pokračuje sa bodom B-7
B-7	vstup:	údaj v GUI
	akcia:	vizualizácia údajov v GUI pomocou jednej z metód
	výstup:	údaje zobrazené v GUI

7.2.4 Prípado C – Kontrola slovenských textov metódou String-blurring

Vstup a výstup procesu

Vstup: požiadavka na dáta na testovanie, prah podobnosti

Výstup: percentuálne ohodnotenia podobnosti, podozrivé dvojice, vizualizované podozrenia

Priebeh procesu

Prípado C		Kontrola slovenských textov metódou String-blurring
Fáza		integračný test
C-1	vstup:	používateľ vyberie metódu String-blurring, mód párovania, prah podobnosti a identifikuje zdroj dát
	akcia:	GUI odošle požiadavku na konzolovú aplikáciu, vykoná sa proces A
	výstup:	načítané dáta v hlavnej pamäti, pokračuje sa bodom C-2
C-2	vstup:	načítané dáta v hlavnej pamäti
	akcia:	vyhádžanie stop-slov
	výstup:	dáta bez stop-slov v hlavnej pamäti, pokračuje sa bodom C-3
C-3	vstup:	dáta bez stop-slov v hlavnej pamäti
	akcia:	lematizácia
	výstup:	predspracované dáta hlavnej pamäti, pokračuje sa bodom C-4
C-4	vstup:	predspracované dáta hlavnej pamäti
	akcia:	spárovanie súborov podľa požadovaného módu
	výstup:	dvojice súborov, ich očakávané podobnosti a podobnosti ostatných súborov, pokračuje sa bodom C-5
C-5	vstup:	dvojice súborov
	akcia:	volanie komparátora pre String-blurring nad definovanými dvojicami

	výstup:	ohodnotenie podobnosti, pokračuje sa bodom C-6
C-6	vstup:	podobnosť z SB z bodu C-5
	akcia:	odoslanie údajov do GUI
	výstup:	údaj v GUI, pokračuje sa bodom B-7
C-7	vstup:	údaj v GUI
	akcia:	vizualizácia údajov v GUI pomocou jednej z metód
	výstup:	údaje zobrazené v GUI

7.2.5 Prípád D – Kontrola zdrojových kódov metódou Greedy String Tiling

Vstup a výstup procesu

Vstup: požiadavka na dáta na testovanie, mód párovania, prah podobnosti

Výstup: percentuálne ohodnotenia podobnosti, podozrivé dvojice, vizualizované podozrenia

Priebeh procesu

Prípád D		Kontrola zdrojových kódov metódou Greedy String Tiling
Fáza		integračný test, akceptačný test
D-1	vstup:	používateľ vyberie metódu GST, mód párovania, prah podobnosti a identifikuje zdroj dát
	akcia:	GUI odošle požiadavku na konzolovú aplikáciu, vykoná sa proces A
	výstup:	načítané dáta v hlavnej pamäti, pokračuje sa bodom D-2
D-2	vstup:	načítané dáta v hlavnej pamäti
	akcia:	tokenizácia
	výstup:	tokenizované vektory, pokračuje sa bodom D-3
D-3	vstup:	tokenizované vektory v hlavnej pamäti
	akcia:	spárovanie súborov podľa požadovaného módu
	výstup:	dvojice súborov, pokračuje sa bodom D-4
D-4	vstup:	dvojice súborov
	akcia:	volanie komparátora pre GST nad definovanými dvojicami
	výstup:	ohodnotenie podobnosti, pokračuje sa bodom D-5
D-5	vstup:	ohodnotenie podobnosti
	akcia:	odoslanie údajov do GUI
	výstup:	údaj v GUI, pokračuje sa bodom D-6
D-6	vstup:	údaj v GUI
	akcia:	vizualizácia údajov v GUI pomocou jednej z metód
	výstup:	údaje zobrazené v GUI

7.2.6 Prípád E – Kontrola slovenských textov metódou LSA

Vstup a výstup procesu

Vstup: požiadavka na dáta na testovanie, prah podobnosti

Výstup: percentuálne ohodnotenia podobnosti, podozrivé dvojice, vizualizované podozrenia

Priebeh procesu

Prípád E		Kontrola slovenských textov metódou LSA
Fáza		integračný test

E-1	vstup:	používateľ vyberie metódu LSA, prah podobnosti a identifikuje zdroj dát, mód porovnávania sa ignoruje
	akcia:	GUI odošle požiadavku na konzolovú aplikáciu, vykoná sa proces A
	výstup:	načítané dáta v hlavnej pamäti, pokračuje sa bodom E-2
E-2	vstup:	načítané dáta v hlavnej pamäti
	akcia:	vyhádzanie stop-slov
	výstup:	dáta bez stop-slov v hlavnej pamäti, pokračuje sa bodom E-3
E-3	vstup:	dáta bez stop-slov v hlavnej pamäti
	akcia:	lematizácia
	výstup:	predspracované dáta hlavnej pamäti, pokračuje sa bodom E-4
E-4	vstup:	predspracované dáta hlavnej pamäti
	akcia:	extrakcia fráz zo súborov
	výstup:	všetky frázy korektne uložené v tabuľke fráz aj s príslušnými metadátami, pokračuje sa bodom E-5
E-5	vstup:	všetky frázy korektne uložené v tabuľke fráz aj s príslušnými metadátami
	akcia:	konštrukcia matice dokumentov a jej spracovanie metódou LSA
	výstup:	matica podobnosti, pokračuje sa bodom E-6
E-6	vstup:	matica podobnosti
	akcia:	odoslanie údajov z matice podobnosti do GUI
	výstup:	údaje v GUI, pokračuje sa bodom E-7
E-7	vstup:	údaje v GUI
	akcia:	vizualizácia údajov v GUI pomocou jednej z metód
	výstup:	údaje zobrazené v GUI

7.2.7 Prípád G – GUI

Prípád G		Test grafického používateľského rozhrania
Fáza		test GUI
G-1	vstup:	otvorené GUI - menu
	akcia:	klikne sa na každú položku v menu
	výstup:	aplikácia vykoná akciu uvedenú v menu a nespadne
G-2	vstup:	panel možností
	akcia:	klikne sa na každú možnosť, zvolia sa všetky prípustné kombinácie
	výstup:	aplikácia nespadne
G-3	vstup:	vizualizátory
	akcia:	vykoná sa detekcia a klikne sa na každú kartu vizualizácií
	výstup:	aplikácia zobrazí vizualizáciu podľa špecifikácie a nespadne

7.3 Protokoly o vykonaní testov

Na nasledujúcich stranách sa nachádzajú jednotlivé protokoly o testoch vo forme, v akej ich vypracovali tester. Protokoly boli písané podľa navrhutej šablóny.

PROTOKOL O VYKONANÍ TESTU

Meno testera: Mego Marek

Názov testovacieho prípadu: Komparátor pre metódu N gram-ov

Fáza testov⁴: unit test

Forma testu⁵: automatizovaný

Popis testovacieho prípadu:

- Tester použije testovaciu triedu pre komparátor metódy N gram-ov.
- Vstup: séria dvoch podobných alebo rovnakých textov.
- Výstup: percentuálny podiel podobnosti textov

Priebeh testu:

Testovací prípad bol spustený a automaticky sa vykonal. Očakávané podiely boli dosiahnuté vo všetkých prípadoch, okrem prázdnych a krátkych reťazcov.

Nájdene nezrovnalosti:

<i>Opis nezrovnalosti</i>	<i>Dôvod nezrovnalosti</i>	<i>Prijaté ako chyba</i>
Pri porovnaní dvoch prázdnych reťazcov metóda vráti výsledok 0 %.	implementácia	áno
Pri porovnaní dvoch rovnakých jednoslovných reťazcov metóda vráti výsledok 0 %.	implementácia	áno

Záver:

Bola nájdená chyba v implementácii. Po opravení chyby programátorom je potrebné vykonať retest.

4 Fáza testov - unit test, integračný test alebo akceptačný test

5 Forma testu - automatizovaný alebo manuálny

PROTOKOL O VYKONANÍ TESTU

Meno testera: David Chalupa

Názov testovacieho prípadu: Komparátor pre metódu Greedy String Tiling

Fáza testov⁶: unit test

Forma testu⁷: automatizovaný

Popis testovacieho prípadu:

- Tester použije testovaciu triedu pre komparátor metódy Greedy String Tiling.
- Vstup:
 - úplne rovnaké súbory (príliš malé - menej ako 8 tokenov)
 - úplne rovnaké súbory (dosť veľké - aspoň 8 tokenov)
 - rôzne súbory, ale jeden príliš malý
 - klasický plagiát s minimálnou detegovateľnou sériou rovnakých tokenov
 - klasický plagiát so zmenou uprostred skopírovanej série
 - prázdne súbory
 - jeden zo súborov prázdny
 - séria rovnakých tokenov odlišnej veľkosti.
- Výstup (podiel podozrivých tokenov a celkového počtu tokenov):
 - 0
 - 1
 - 0
 - 8/11
 - 0
 - 0
 - 0
 - 0
 - 8/9
- Porovnáva sa očakávaný podiel podozrivých tokenov s reálnym.

Priebeh testu:

Testovací prípad bol spustený a automaticky sa vykonal. Očakávané podiely boli dosiahnuté v 6 prípadoch z 8. Prípady 6 a 8 detegovali nasledujúce odlišnosti.

Prípad	Očakávaná hodnota	Reálna hodnota
6	0	INF
8	8/9	1

Nájdene nezrovnalosti:

Opis nezrovnalosti	Dôvod nezrovnalosti	Prijaté ako chyba
V prípade 6 metóda vrátila veľmi vysokú hodnotu, nastalo delenie nulou.	implementácia	áno
V prípade 8 metóda nevrátila 8/9, ale hodnotu 1.	analýza	nie

Záver:

Bola nájdená chyba v implementácii, ktorá spôsobuje vrátenie nekorektnej hodnoty a nezrovnalosť spôsobená nekorektným odhadom správania algoritmu. Druhá nezrovnalosť bola prijatá ako vlastnosť, nie chyba.

6 Fáza testov - unit test, integračný test alebo akceptačný test

7 Forma testu - automatizovaný alebo manuálny

PROTOKOL O VYKONANÍ TESTU

Meno testera: Michal Noskovič

Názov testovacieho prípadu: Predspracovanie textu

Fáza testov⁸: unit test

Forma testu⁹: automatizovaný

Popis testovacieho prípadu pre metódu mazania bielych znakov:

- Tester použije metódu na mazanie bielych znakov zo vstupného reťazca.
- Vstup: Reťazec premenlivej dĺžky, ktorého jednotlivé slová sú oddelené rôznym počtom bielych znakov. Medzi biele znaky patrí medzera, znak nového riadku a tabulátor.
- Výstup: Pôvodný reťazec, v ktorom sú nahradené všetky viacnásobné za sebou idúce výskyty bielych znakov za jednu medzeru. Upravená veta neobsahuje ani jeden tabulátor a znak nového riadku.
- Porovnáva sa upravený reťazec s predpokladaným výsledkom.

Priebeh testu:

Testovací prípad bol spustený a automaticky sa vykonal. Test odhalil chybu v implementácii.

Nájdené nezrovnalosti:

<i>Opis nezrovnalosti</i>	<i>Dôvod nezrovnalosti</i>	<i>Prijaté ako chyba</i>
Pri spracovávaní reťazca, ktorý obsahoval sekvenciu medzera, nový riadok, medzera a nový riadok bol vrátený reťazec, ktorý obsahoval 2 znaky nového riadku. Vrátený reťazec nemal obsahovať ani jeden znak nového riadku.	implementácia	nie

Záver:

Bola nájdená chyba, ktorá spôsobovala, že vrátený reťazec obsahuje znaky nového riadku. Chyba bola okamžite odstránená využitím knižničnej funkcie *simplified()*.

⁸ Fáza testov - unit test, integračný test alebo akceptačný test

⁹ Forma testu - automatizovaný alebo manuálny

PROTOKOL O VYKONANÍ TESTU

Meno testera: Mego Marek

Názov testovacieho prípadu: Kontrola zdrojových kódov metódou Greedy String Tiling

Fáza testov¹⁰: integračný test

Forma testu¹¹: manuálny

Popis testovacieho prípadu:

- Tester vyberie požadovaný test podobnosti, spôsob zobrazovania a sleduje či priebeh Testu vrátane procesu testovania prebieha podľa očakávaní
- Vstup: požiadavka na dáta na testovanie, mód párovania, prah podobnosti
- Výstup: percentuálne ohodnotenia podobnosti, podozrivé dvojice, vizualizované podozrenia

Priebeh testu:

V priebehu testu boli testu podrobené viaceré kombinácie zdrojových súboroch. Pri všetkých sa podarilo prejsť jednotlivými fázami. Načítanie súborov do pamäte, tokenizácia, porovnávanie, zobrazovanie výsledkov. Ďalej bolo kontrolované správne zobrazovanie výsledkov pri jednotlivých možnostiach zobrazovania.

Nájdené nezrovnalosti:

<i>Opis nezrovnalosti</i>	<i>Dôvod nezrovnalosti</i>	<i>Prijaté ako chyba</i>
V prípade neoznačenia nutnosti tokenizácie sú výsledky vždy 100%	implementácia	
Pri načítaní súborov do pamäte nie je zobrazovaný vývoj na vývojovom bare	implementácia	

Záver:

Boli nájdené chyby v implementácii. V prípade, že používateľ postupuje podľa používateľskej príručky nemajú tieto chyby žiadny vplyv na výsledné hodnoty detekcie plagiátorstva.

10 Fáza testov - unit test, integračný test alebo akceptačný test

11 Forma testu - automatizovaný alebo manuálny

PROTOKOL O VYKONANÍ TESTU

Meno testera: Matej Sabo

Názov testovacieho prípadu: Test integračného procesu B

Fáza testov¹²: integračný test

Forma testu¹³: manuálny

Popis testovacieho prípadu:

- Test ma za úlohu skontrolovať správnosť výmeny dát medzi jadrom aplikácie a používateľským rozhraním.
- Pri procese dochádza k viacerým výmenám údajov medzi modulmi aplikácie. Najprv používateľské rozhranie pošle parametre kontroly jadra aplikácie, ktoré načíta súbory a vykoná kontrolu. Následne pošle späť výsledky testovania používateľskému rozhraniu, ktoré ich vizualizuje rôznymi metódami vizualizácie.
- Tester spustí aplikáciu s používateľským rozhraním, vyberie metódu 3G, nastaví parametre kontroly, identifikuje zdroj dát a spustí kontrolu podobnosti. Výsledky kontroly porovná s výsledkami pri použití konzolového rozhrania aplikácie
- Vstup: cesty k súborom, alebo adresárom.
- Výstup: vizualizované dáta
- Porovnajú sa očakávané načítané súbory s reálnymi.

Priebeh testu:

Testovanie sa vykonalo ručne. Po nastavení parametrov kontroly plagiarizmu program automaticky spustil konzolovú aplikáciu pre kontrolu. Súbory boli načítané podľa očakávaní v prípade načítavania súborov aj adresárov a aj pri ich väčšom počte. Následne sa porovnali jednotlivé výsledky kontroly v prípade použitia používateľského rozhrania a bez neho. Výsledky kontroly boli zhodné vo všetkých prípadoch testovania (načítavania jednotlivých súborov, alebo celých adresárov a aj pri ich väčšom počte). Súčasťou testovania bolo aj testovanie vizualizácie, teda či rôzne druhy vizualizácie zobrazujú dáta správne. Vo všetkých prípadoch testovania zobrazovali vizualizačné mechanizmy rovnaké výsledky tej istej kontroly podľa ich charakteru a správne.

Nájdene nezrovnalosti:

<i>Opis nezrovnalosti</i>	<i>Dôvod nezrovnalosti</i>	<i>Prijaté ako chyba</i>
Žiadne		

Záver:

Test prebehol úspešne bez nájdenej nezrovnalosti.

12 Fáza testov - unit test, integračný test alebo akceptačný test

13 Forma testu - automatizovaný alebo manuálny

PROTOKOL O VYKONANÍ TESTU

Meno testera: Marek Mego

Názov testovacieho prípadu: Kontrola slovenských textov algoritmom String-blurring

Fáza testov¹⁴: integračný test

Forma testu¹⁵: manuálny

Popis testovacieho prípadu:

- Tester vyberie požadovaný test podobnosti, spôsob zobrazovania a sleduje či priebeh testu vrátane procesu testovania prebieha podľa očakávaní
- Vstup: požiadavka na dáta na testovanie, mód párovania, prah podobnosti
- Výstup: percentuálne ohodnotenia podobnosti, podozrivé dvojice, vizualizované podozrenia

Priebeh testu:

V priebehu testu boli testu podrobené viaceré kombinácie zdrojových súboroch. Pri všetkých sa podarilo prejsť jednotlivými fázami. Načítanie súborov do pamäte, tokenizácia, porovnávanie, zobrazovanie výsledkov. Ďalej bolo kontrolované správne zobrazovanie výsledkov pri jednotlivých možnostiach zobrazovania.

Nájdene nezrovnalosti:

<i>Opis nezrovnalosti</i>	<i>Dôvod nezrovnalosti</i>	<i>Prijaté ako chyba</i>
Pri načítaní súborov do pamäte nie je zobrazovaný vývoj na vývojovom bare	implementácia	

Záver:

Bola nájdená chyba v implementácii ktorá sa netýka priamo String-blurring metódy ale načítania súborov všeobecne. Viditeľná je predovšetkým pri načítaní veľkého množstva súborov.

14 Fáza testov - unit test, integračný test alebo akceptačný test

15 Forma testu - automatizovaný alebo manuálny

PROTOKOL O VYKONANÍ TESTU

Meno testera: Michal Noskovič

Názov testovacieho prípadu: Grafické rozhranie aplikácie

Fáza testov¹⁶: integračný

Forma testu¹⁷: automatizovaný

Popis testovacieho prípadu pre GUI:

1. Tester postupne používa ovládacie prvky ponúkané GUI.
2. Vstup: Textové reťazce, používateľom vybrané súbory a interakcia s programom.
3. Výstup: Spracované dáta zobrazené v aplikácii.

Priebeh testu:

Tester vykonal daný test, ktorý odhalil chybu v implementácii.

Nájdené nezrovnalosti:

<i>Opis nezrovnalosti</i>	<i>Dôvod nezrovnalosti</i>	<i>Prijaté ako chyba</i>
Side-by-side vizualizácia nie je asociatívna. Porovnanie súborov A-B nie je totožné ako porovnanie súborov B-A.	implementácia	áno
Po spustení testovania súborov je ikona spúšťajúca testy naďalej aktívna.	implementácia	áno
Program umožňuje exportovať výsledky, napriek tomu že nebol spustený test a výsledky neexistujú.	implementácia	áno
Animácia v pravej časti programu sa nezobrazuje na všetkých počítačoch.	implementácia	áno
Po porovnávaní metódou LSA, nastavením hranice podobnosti na hodnotu 20 a zobrazením záložky graf aplikácia spadne.	implementácia	áno
Porovnávanie textov metódov GST vracia na všetky porovnania hodnotu 100%.	implementácia	áno
Po nastavení hĺbky vnorenia pri porovnávaní adresára v ktorom sa nachádzajú len textové súbory aplikácia spadne.	implementácia	áno

Záver:

Pri testovaní boli odhalené chyby v grafickom rozhraní aplikácie, pričom niektoré z nájdených chýb spôsobujú pád spustenej aplikácie. Chyby boli ohlásené a pracuje sa na ich odstránení.

¹⁶ Fáza testov - unit test, integračný test alebo akceptačný test

¹⁷ Forma testu - automatizovaný alebo manuálny

PROTOKOL O VYKONANÍ TESTU

Meno testera: David Chalupa

Názov testovacieho prípadu: Kontrola slovenských textov metódou LSA

Fáza testov¹⁸: integračný test

Forma testu¹⁹: manuálny

Popis testovacieho prípadu:

- Tester použije GUI aplikácie, z ktorého podľa testovacieho prípadu spustí metódu LSA.
- Vstup:
 - žiadne súbory
 - množina 10 súborov
 - množina 88 súborov
 - množina 125 súborov
- Výstup:
 - chybové hlásenie
 - výsledná podobnosť, korektne vizualizovaná, export prebehol v poriadku
 - výsledná podobnosť, korektne vizualizovaná, export prebehol v poriadku
 - algoritmus pre metódu "LSA" nezbehol v prijateľnom čase, stochastické LSA výsledky dávajú v prijateľnom čase

Priebeh testu:

- spustená bola kontrola pre 4 rôzne charaktery vzoriek
- použilo sa aj rekurzívne prehliadanie
- kontrolovalo sa aj využitie stochastických LSA (5% a 20%)

Nájdene nezrovnalosti:

<i>Opis nezrovnalosti</i>	<i>Dôvod nezrovnalosti</i>	<i>Prijaté ako chyba</i>
Počas kontroly metódou LSA sa zobrazuje progress bar so stálou hodnotou 0%.	implementácia	nie
Pri kontrole veľkej vzorky (125 súborov) je LSA časovo veľmi náročná.	implementácia	nie

Záver:

Progress bar s hodnotou 0% nie je chybou, ale je spôsobený tým, že metóda vypočíta výsledky v jednom kroku, t.j. neposiela ich postupne. Implementácia je preto v súlade s návrhom. Časová náročnosť LSA nie je tiež chybou, nakoľko metóda je skutočne pri veľkých množstvách dát výpočtovo zložitá (práve pre tento dôvod máme aj stochastické verzie).

18 Fáza testov - unit test, integračný test alebo akceptačný test

19 Forma testu - automatizovaný alebo manuálny

Príloha A – Ponuka Tím č.9

Kvôli prehľadnosti dokumentu uvádzame ponuku, ktorou sme sa prezentovali na začiatku semestra, ako samostatnú prílohu tohto dokumentu.

Slovenská technická univerzita v Bratislave

FAKULTA INFORMATIKY A INFORMAČNÝCH TECHNOLOGIÍ

Milan Freml, Dávid Chalupa, Peter Mindek, Michal Noskovič, Matej Sabo, Marek Mego

Textový editor obohatený o grafické prvky

Tímový projekt – tím č. 9

Email: noskovicmichal@gmail.com
September, 2009

1 Tím

Milan Fremi



Úspešne absolvoval odbor Informatika na bakalárskom štúdiu FIIT STU, pokračuje v odbore IS. Je nadšencom Linuxu a všetkých open-source technológií, pri bakalárskej práci sa zaoberal oblasťou virtualizácie. Počas štúdia a popri menších pracovných projektoch získal skúsenosti s programovacími jazykmi C, Pascal, C++, Java, Python, ako aj s webovými technológiami (X)HTML, XML, XSL, CSS, PHP, MySQL, JavaScript, Flash.

Dávid Chalupa



Ukončil bakalárske štúdium v odbore Informatika na FIIT STU v Bratislave, v súčasnosti pokračuje v odbore Softvérové inžinierstvo. Počas svojho doterajšieho štúdia sa stretol s jazykmi C, Pascal, ako aj webovými technológiami (X)HTML, CSS, PHP, MySQL. Získal vedomosti z oblasti princípov softvérového inžinierstva, ale aj záujem o analýzu a návrh algoritmov.

Peter Mindek



Absolvoval bakalárske štúdium na FIIT STU v odbore Informatika, momentálne pokračuje v odbore Informačné systémy. Venuje sa predovšetkým témam súvisiacim s počítačovou grafikou a interakciou človeka s počítačom (podobne bola zameraná jeho bakalárska práca s názvom Interaktívne prehliadanie multimediálneho obsahu v 3D priestore, ako aj špeciálne zadanie na predmet Interakcia človeka s počítačom s názvom Room Information System). Má skúsenosti s programovacími jazykmi Delphi, C/C++, Java, PHP, Lua, Assembler.

Michal Noskovič



Absolvent FIIT STU v odbore Informatika. Momentálne pokračuje v štúdiu v odbore Informačné systémy. Bakalárske štúdium ukončil prácou zameranou na elektronickú podporu vzdelávania. Počas štúdia a popri tvorbe menších projektov získal skúsenosti s programovacími jazykmi Java, PHP, C/C++, Delphi.

Matej Sabo



Po troch rokoch štúdia informatiky na FIIT STU pokračuje inžinierskym štúdiom informačných systémov na rovnakej fakulte. Počas štúdia a práci na bakalárskej práci, ktorej témou bola Online jazyková hra, získal skúsenosti najmä s jazykmi a technológiami Java, C, SQL, Flex.

Marek Mego



Obhájil bakalársku prácu na tému Vzďialený dohľad nad pracovnou plochou na FIIT STU a stal sa jej úspešným absolventom. Momentálne pokračuje na rovnakej fakulte na inžinierskom štúdiu v odbore Informačné systémy. Počas štúdia získal skúsenosti s programovacími jazykmi Java, C, Delphi, PHP a databázovým systémom MySQL.

2 Motivácia

Tému Textový editor obohatený o grafické prvky sme si zvolili na základe tímového konsenzu. Domnievame sa, že vďaka tejto téme získame nové skúsenosti s tvorbou užitočných aplikácií, použiteľných každodenne, predovšetkým pri práci softvérového vývojára. Využitelnosť takéhoto textového editora vidíme aj pri našich ďalších projektoch.

Multiplatformovosť knižnice Qt môže priniesť vyššiu mieru nasadenia editora v reálnej prevádzke. Môže tým prispieť aj k úspechu aplikácie u rôznych skupín expertov v oblasti informačných technológií. V našom tíme spolupracujú používatelia rôznych operačných systémov a platforiem, menovite Windows a Linux.

Obohacovanie zdrojového kódu o grafické prvky má potenciál vniesť lepšiu vizualizáciu toku riadenia do inak šedivých vôd oceánu textových editorov. Táto predstava nás naplňuje radosťou a radi by sme priložili ruku k vytvoreniu takéhoto magnificentného nástroja.

Zaujímavou črtou spomínaného produktu je zjednotenie zdrojového kódu a príslušnej dokumentácie, čo môže výrazne podporiť komunikáciu v tíme. Dalším problémom, ktorý aplikácia môže zmierniť je dĺžka času stráveného nad tvorením dokumentácie existujúceho softvéru. Pri sklbení vlastností klasického textového editora s rich-text editorom je možné dosiahnuť vyššiu efektivitu práce, čo môže mať aj ekonomické výhody. Vkladaním unikátnych značiek je možné zrýchliť vyhľadávanie a orientáciu v dokumente, zatraktívniť a sprístupniť zdrojový kód aj laickému oku.

Viaceri členovia nášho tímu majú skúsenosti s GUI knižnicami ako napríklad VCL (Visual Component Library – štandardná knižnica Delphi), alebo Swing. Veľmi radi by sme sa však oboznámili aj s knižnicou Qt, ktorá má podľa nás veľký potenciál pri vytváraní užitočných a na pohľad príjemných multiplatformových aplikácií. Vytváranie multiplatformových aplikácií je taktiež oblasť, v ktorej by sme sa radi zdokonalili. Prejavujeme entuziazmus k novým myšlienkam, technológiám a princípom, ktoré nebudeme váhať využiť.

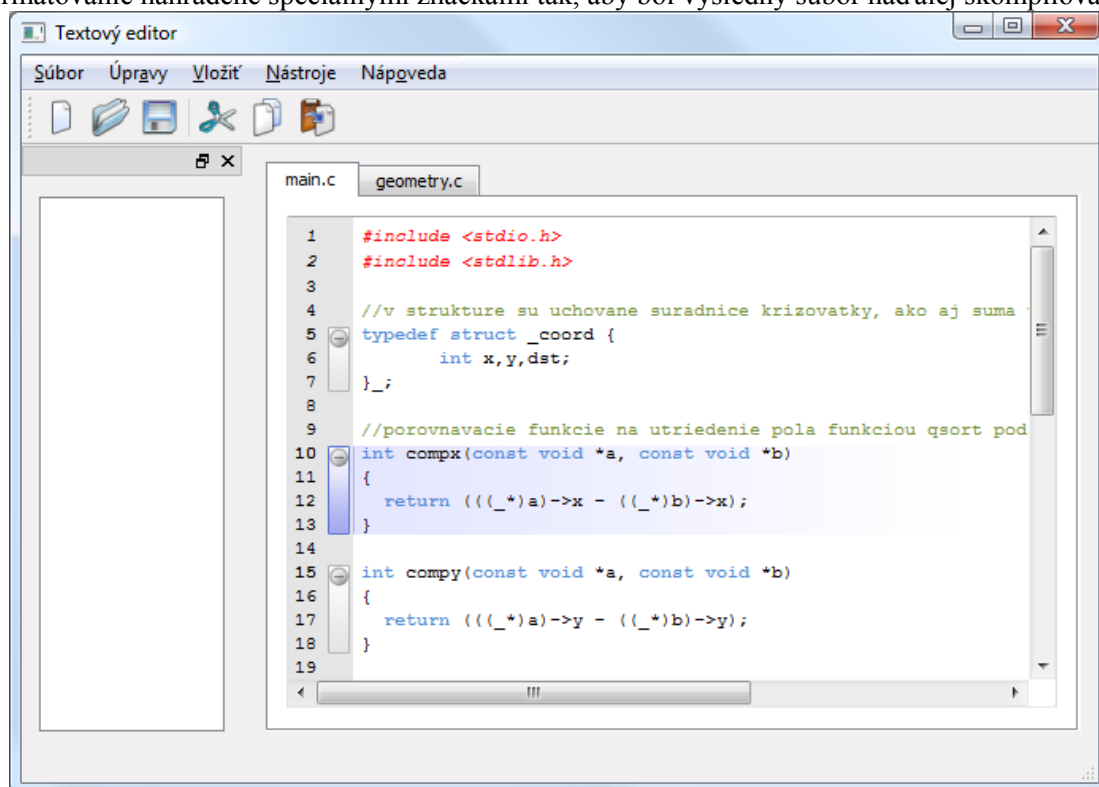
3 Konceptia riešenia

Výsledkom našej práce by mal byť pokročilý textový editor rozšírený o grafické prvky. Nerealizáciu tohto projektu bude použitý multiplatformový toolkit Qt a vývojové prostredie Qt Creator. Aplikácia bude implementovaná v jazyku C++.

Hlavným prínosom textového editora by bol parser, ktorý by rozdeľoval zdrojové kódy do blokov. Tieto bloky by sa dali jednoducho skrývať a opätovne zobrazovať pomocou tlačidla „-“ resp. „+“, ktoré by bolo zobrazené na bočnom paneli pri príslušnom bloku (obr. 1). Bloky by sa dali taktiež presúvať myšou (Drag & Drop). Bloky by boli chápané rovnako, ako v logike programovacieho jazyka (napr. pre jazyk C by bol blok ohraničený zloženými zátvorkami { a }). Z toho vyplýva, že bloky by mohli byť aj vnorené, pričom vizuálne by boli naznačené iba bloky najvyššej úrovne, ako aj blok, v ktorom sa momentálne nachádza textový kurzor (na obrázku je tento blok zvýraznený modrou farbou).

Editor by ponúkal možnosť zvýrazňovania syntaxe rôznych programovacích jazykov. Rozmýšľame o použití Qt Style Sheets, vďaka čomu by bolo možné jednoducho meniť nastavenia vzhľadu zvýrazňovania (typ a veľkosť písma, farbu, pozadie...) Táto funkcionality by mohla byť rozšíriteľná pomocou skriptovacieho jazyka Lua.

Ďalšou vlastnosťou editora by bola možnosť vytvárania formátovaných komentárov. Tieto by boli reprezentované tiež ako bloky, rovnocenné s blokmi zdrojového kódu. V rámci týchto blokov by používateľ mohol vkladať obrázky do textu, ľubovoľne meniť veľkosť, farbu a druh písma, a pod. Pri uložení súboru by bolo formátovanie nahradené špeciálnymi značkami tak, aby bol výsledný súbor naďalej skompilovateľný.



Obr. 1.: Návrh používateľského rozhrania

Pri implementácii editora by sme radi zväžili využitie komponentu QScintilla, ktorý poskytuje pokročilé funkcie pri úprave najmä zdrojových kódov. Ďalšou možnosťou by bolo použitie triedy QTextEdit. Reimplementáciou jej metódy paintEvent() by bolo možné vytvoriť grafické prvky textového editora.

Inšpiráciou by pre nás mohlo byť napríklad vývojové prostredie Eclipse, ktoré ponúka funkcionality vhodnú pre editáciu zdrojových kódov (skrytie/rozbalenie triedy alebo metódy pomocou grafických ikon pri príslušných riadkoch, a pod.), ako aj textové editory ako PSPad, Notepad++, alebo SciTE.

Zoradenie všetkých tém podľa priority

1. Textový editor obohatený o grafické prvky (Editor)
2. Portál pre časopis (Časopis)
3. Podpora kontroly plagiarizmu (Plagiarizmus)
4. Informačný systém stredných škôl (SS IS)
5. Mobilný cestovný poriadok pre iPhone (Mobilný Poriadok)
6. Virtuálna FIIT (VFIIT)
7. Evidencia publikačnej činnosti (EPCA) (EPCA)
8. Webový portál pre zdravotne postihnutých občanov (ZŤP Portál)
9. Vizualizácia softvérových artefaktov v 3D priestore (3DVizual)
10. Využitie sociálnych sietí pri vytváraní pracovných tímov - druhý pokus :) (Sociálne siete)
11. Knižnica (Knižnica)
12. Dizajn s použitím obohatenej reality (ARDizajn)
13. Elastické komunikačné centrum (EKCentrum)
14. Grafická podpora vyhľadávania znalostí v dokumentoch (Dokumenty)
15. Webové stránky pre cestovnú kanceláriu (Cestovka)
16. Digitálne mapy (Digmapy)
17. Web 2.0 v knižniciach alebo od OPACu k portálu (DLPortál)
18. Automatizovaná podpora predmetu z oblasti programovania (DSAPodpora)
19. Tvorba rozvrhov (Rozvrhy)
20. Hierarchická wiki s právami (Wiki)
21. RoboCup tretí rozmer (RoboCup 3D)

Rozvrh výučby tímu

	07:00 07:50	08:00 08:50	09:00 09:50	10:00 10:50	11:00 11:50	12:00 12:50	13:00 13:50	14:00 14:50	15:00 15:50	16:00 16:50	17:00 17:50	18:00 18:50	19:00 19:50	20:00 20:50	21:00 21:50	
Po						PDS (PM)		PDS (všetci okrem PM) ML2 (PM)		TP (všetci)		VSS(DCH) VIS(všetci okrem DCH)				
Ut	Kod (DCH)			NS (MF)					MSI (všetci)		MSI (všetci)					
St			NS (MF)					ML2 (PM)		DD (MN, MM)						
									DD (MN, MM)	Práca (MF)						
Št	Kod (DCH)			Squash (MN)				ASS (DCH)								
										AIS (všetci – DCH)						
Pi	BIS (MM)		PDT (všetci)								Práca (MF)					

Rozvrh hodín členov tímu (zelená = prednáška, červená = cvičenie, žltá = iné činnosti, modrá = preferovaný čas)

Príloha B – Preberacie protokoly

V tejto prílohe sa nachádzajú preberacie protokoly k našej dokumentácii v jednotlivých kontrolných bodoch. Tieto protokoly máme iba v tlačenej podobe (kvôli podpisom), v elektronickej podobe sa tu preto nenachádzajú.