

Tímový projekt
Rozvrhový systém FIIT
Dokumentácia k riadeniu

Autori: Bc. Radoslav Zachar, Bc. Viliam Kubis, Bc. Jakub Calík,
Bc. Pavol Škvarenina, Bc. Marián Hlavenka, Bc. Matúš Hitka,
Bc. Ondrej Danada

Tím: tím č. 6

Kontakt: tim6@danada.sk

Študijný odbor: Softvérové inžinierstvo, Informačné systémy

Ročník: 1. Ing.

Akademický rok: 2011/2012

Vedúci práce: Ing. Miroslav Galbavý

História vývoja dokumentu

Dátum	Verzia	Opis	Autor
7.11.2011	1.0	Vytvorenie dokumentu	Jakub Calík
8.11.2011	1.1	Motivácia, koncepcia riešenia, zadanie	Jakub Calík, Radoslav Zachar
9.11.2011	1.2	Vloženie motivácie na tému do príloh	Jakub Calík, Radoslav Zachar
13.11.2011	1.3	Spísané podporné prostriedky	Radoslav Zachar, Jakub Calík
15.11.2011	1.4	Vloženie zápisníc	Jakub Calík
16.11.2011	1.5	Vloženie metodiky	Radoslav Zachar
23.11.2011	1.6	Vloženie druhej metodiky medzi prílohy	Jakub Calík
28.11.2011	1.7	Vloženie preberacieho protokolu medzi prílohy	Radoslav Zachar
29.11.2011	1.8	Pridaný dlhodobý plán projektu	Jakub Calík
5.12.2011	1.9	Upravený dlhodobý plán	Radoslav Zachar, Jakub Calík
11.12.2011	1.10	Pridané plány jednotlivých iterácií	Všetci
11.12.2011	1.11	Úprava celého dokumentu, oprava diakritiky a interpuncie	Radoslav Zachar
12.12.2011	1.12	Vložené a oštylované aplikácie poznatkov	Radoslav Zachar

Obsah

1 ÚVOD	1-3
1.1 ÚČEL DOKUMENTU.....	1-3
1.2 VÝVOJOVÝ TÍM.....	1-3
2 MOTIVÁCIA	2-5
3 KONCEPCIA RIEŠENIA	3-6
4 ZADANIE	4-7
5 ROLE A ZODPOVEDNOSTI ČLENOV TÍMU.....	5-8
5.1 POZÍCIE ČLENOV TÍMU	5-8
5.2 ZODPOVEDNOSTI JEDNOTLIVÝCH POZÍCIÍ	5-8
6 PLÁN PROJEKTU	6-10
6.1 DLHODOBÝ PLÁN	6-10
6.2 ITERÁCIA I1	6-12
6.3 ITERÁCIA I2	6-13
6.4 ITERÁCIA I3	6-14
6.5 NÁVRHY NA VYLEPŠENIE SYSTÉMU	6-15
7 PODPORNÉ PROSTRIEDKY	7-16
7.1 REDMINE	7-16
7.2 SVN	7-16
7.3 MAILING-LIST	7-17
8 APLIKÁCIA POZNATKOV	8-18
8.1 MANAŽMENT RIZÍK	8-18
8.2 MANAŽMENT ROZVRHU A PLÁNOVANIA	ERROR! BOOKMARK NOT DEFINED.
8.3 MANAŽMENT KVALITY	8-20
8.4 MANAŽMENT TVORBY DOKUMENTÁCIE	8-21
8.5 MANAŽMENT PODPORY VÝVOJA	8-22
8.6 MANAŽMENT KOMUNIKÁCIE	8-23
8.7 MANAŽMENT MONITOROVANIA PROJEKTU	8-24
9 PRÍLOHA A (ODOVZDANÁ PONUKA)	9-27
10 PRÍLOHA B (PREBERACÍ PROTOKOL).....	10-28
11 PRÍLOHA C (METODIKY).....	11-29
12 PRÍLOHA D (ZÁPISNICE ZO STRETNUTÍ).....	12-30

Zoznam obrázkov, tabuliek, diagramov a grafov

Table 1 Dlhodobý plán	6-11
Table 2 Plán prvej iterácie	6-12
Table 3 Plán druhej iterácie	6-13
Table 4 Plán tretej iterácie	6-14
Figure 1 Všeobecný iteračný model vývoja	6-10
Figure 2 Redmine nástroj.....	7-16
Figure 3 SVN Schéma	7-16
Figure 4 Mailing-list komunikácia	7-17

1 Úvod

1.1 Účel dokumentu

Predkladaný dokument obsahuje dokumentáciu k riadeniu tímového projektu k rozvrhového systému vytváraného v akademickom roku 2011/2012. Tvorba systému pozostáva z pokračovania minuloročného systému. Tento dokument predstavuje oficiálnu dokumentáciu vytváranú na predmete Tímový projekt na Fakulte informatiky a informačných technológií Slovenskej technickej univerzity v Bratislave.

1.2 Vývojový tím

Viliam Kubis, skúsený webový vývojár so šiestimi rokmi praxe pri vývoji dynamických webových stránok, ktorý sa aktívne venuje programovaniu pre mobilné platformy. V súčasnosti programuje aplikácie pre mobilný operačný systém Android. Vyniká schopnosťou učiť sa nové veci a objavovať nepoznané, čo je iste výhodné nielen v obyčajnom živote, ale aj vo svete programovania. Ovláda jazyky a technológie PHP, MySQL, XHTML, XHTML, XML, CSS (1-3), JavaScript (+jQuery) na profesionálnej úrovni, medzi jeho obľúbené jazyky taktiež patrí štandardné C, v ktorom má mnoho skúseností. Jeho koníčkom je počítačová bezpečnosť a efektívnosť programov a skriptov, ktoré vytvára, alebo databáz, ktoré navrhuje. Má tiež skúsenosti s jazykmi Java a Lua. Taktiež ho zaujíma nízkoúrovňové poznanie počítača (assembler, hardware), rôzne komunikačné protokoly a informatika ako celok. Rád presne vie, ako každá vec funguje, aby si bol istý jej spoľahlivosťou a efektívnosťou.

Radoslav Zachar je cieľavedomý, zodpovedný a komunikatívny mladý programátor. Zaujíma sa o moderné informačne technológie a informatiku ako celok. Ovláda programovacie jazyky C, C++, C#, Java, PHP, HTML, CSS, XML, Delphi, databázy MySQL a PostgreSQL. Orientuje sa v operačných systémoch OS WIN a OS LINUX. Jeho programovacie znalosti sú najväčšie z oblasti vytvárania multiplatformových webových aplikácií v jazyku PHP prepojeným s CSS a akoukoľvek databázou. Vyniká schopnosťami v oblasti programovania v jazyku C++ alebo C#. Bez problémov programuje aj v prostredí .NET. Pri svojej práci sa snaží najmä o jednoduché a užívateľovi prívetivé grafické používateľské prostredie.

Jakub Calík, všestranne zameraný programátor, ktorý preferuje hlavne objektovo orientovaný prístup k programovaniu. Ovláda viaceré programovacie jazyky ako napr. C, C++, C#, Java, HTML. Taktiež sa orientuje v jednoduchých databázových systémoch a databázach MySQL a SQLite. Aktívne sa venuje vývoju aplikácií pre mobilné operačné systémy, hlavne iOS a Android. Prínosom sú aj základné poznatky z oblasti počítačovej grafiky a návrhu používateľských rozhraní.

Marián Hlavenka, zodpovedný a ambiciózný programátor, ktorého obľúbeným jazykom je C#. Preferuje objektovo orientovaný prístup k programovaniu a jeho skúsenosti zahŕňajú prácu s Pascalom, Assemblermi, HTML, Javou, C, C++ a Macromedia Flash Playerom. Má základy pri používaní operačných systémov na báze

Unix, avšak jeho primárnym operačným systémom je Microsoft Windows, v ktorom je veľmi zbehlý.

Ondrej Danada, ovláda technológie a jazyky HTML, CSS, SQL, XML, JavaScript, AJAX, PHP, Java, UML, C/C++, Visual Basic, ASP, databázové systémy MsSQL a MySQL spolu so softvérovými produktami ako Visual Studio, Eclipse, Tortoise SVN a IBM RSA. Medzi jeho praktické skúsenosti patrí napríklad vývoj webovej JAVA aplikácie – šach na webe, vývoj, údržba, nasadenie a prevádzka informačného systému malej firmy a integrácia s Active Directory, CMS Joomla! a phpBB.

Matúš Hitka, programátor, ktorý si počas práce na rôznych školských zadaniach osvojil základné programovacie techniky jazykov C, C#, Java, HTML, XML, SQL, Pascal. Prednosť však dáva objektovo orientovanému programovaniu. Taktiež zvláda prácu v jednoduchých databázových systémoch. Je mladý, sebavedomý a veľmi prispôsobivý, schopný rýchlo sa naučiť nové veci.

Pavol Škvarenina je softvérový vývojár so štvorročnou praxou vo vývoji webových informačných systémov. Vo firme, v ktorej pracuje, nesie plnú zodpovednosť za vývoj určitých častí IS a taktiež, v prípade potreby, pomáha pri nasadzovaní IS u zákazníka. Do bázy jeho znalostí patria najmä technológie a prostredia .NET, ASP.NET, Silverlight, HTML, JavaScript, Oracle, MS SQL.

2 Motivácia

Hlavná motivácia k danej téme spočíva v tom, že ide o vývoj, respektíve doladovanie systému, ktorý bude reálne nasadený v praxi. Táto práca by mala uľahčiť tvorbu rozvrhov ako študentom, tak aj pedagógom, teda výsledný produkt úsilia nášho tímu by mal prispieť k celkovému blahu všetkých ľudí na fakulte.

Samotný rozvrhový systém je napísaný v jazyku PHP a jedná sa o webovú aplikáciu, s čím má veľa ľudí v našom tíme enormné skúsenosti. Aplikácia založená na databáze PostgreSQL tvorí značne bezpečný a výkonný systém, ktorý je pre túto úlohu ako stvorený. Je pre nás veľkou výzvou pracovať na projekte, ktorý neskončí len niekde v archíve, keďže cieľom je systém nasadiť a uviesť do prevádzky čo najskôr. Teší nás, že sa bude nový systém používať pre potreby celej novej FIIT a sme si istí, že uľahčí prácu nie len mnohým profesorom, ale v konečnom dôsledku aj nám, študentom. Jeden člen nášho tímu pracoval na problematike tvorby rozvrhov v rámci predmetu PSI v bakalárskom štúdiu. Nakoľko bola jeho práca veľmi dobre ohodnotená, očakávame, že jeho skúsenosti dopomôžu k vyššej úrovni vypracovávaného projektu. Veríme, že každý študent či pedagóg by bol vďačný za spustenie takéhoto systému a že každému z nich by zlepšil a spríjemnil život na fakulte.

Za dôležité považujeme snahu podporiť proces tvorby rozvrhov na fakulte lepším nástrojom na tvorbu rozvrhov s využitím podporného systému na zber požiadaviek k samotnej tvorbe týchto rozvrhov. Vypracovaním tohto projektu sa každému z nás určite rozšíria naše odborné znalosti a poznatky pri tvorbe moderných dynamických webových systémov na tvorbu rozvrhov a takisto aj zber informácií.

3 Konceptia riešenia

Aktuálne rozpracovaný rozvrhový systém novej FIIT minuloročným tímom je v štádiu, v ktorom je už mnoho častí systému navrhnutých a napísaných, avšak taktiež je ešte treba mnoho súčastí vytvoriť / upraviť. Keďže je celý systém napísaný v skriptovacom jazyku PHP, pričom konkrétne používa PHP framework Yii a databázu PostgreSQL, bude pre náš tím veľmi ľahké na projekt nadviazať a vo vývoji pokračovať.

Našou konkrétnou víziou je dorobenie plne automatizovaného systému na tvorbu rozvrhov, pričom by sme sa chceli zamerať na nasledovné vlastnosti a funkcie:

- automatická tvorba rozvrhov na základe požiadaviek
- burza cvičení (študenti si budú môcť vymieňať cvičenia, kus sa kus alebo doplatkom "bodov")
- prepojenie systému s akademickým informačným systémom (AIS) - vygenerované rozvrhy sa do AIS automaticky vložia
- možnosť vytvoriť si záujmové skupiny pri vsádzaní si na rozvrhové akcie - študent si založí skupinu, v ktorej so svojimi kamarátmi spoja svoje kredity a budú si môcť vsádzať na rozvrhové akcie spoločne
- podpora riešenia kolízií v rozvrhu

Náš tím má bohaté skúsenosti s bezpečnosťou web stránok, pred nasadením systému do praxe bude systém dôkladne bezpečnostne otestovaný proti prípadným bezpečnostným chybám, ktoré budú následne opravené.

Je vízia, že sa systém dostane do ostrej prevádzky už tento školský rok a že bude úspešne vytvárať rozvrhy už pre budúce ročníky študentov.

4 Zadanie

Aj po nasadení systému AIS, zostáva stále otvorená problematika prípravy a vytvárania semestrálnych rozvrhov a následne aj skúškových rozvrhov.

V súčasnosti je rozpracovaný špecializovaný systém pre tvorbu rozvrhov (momentálne viazaný na prostredia PHP a PostgreSQL), ktorý už teraz umožňuje evidenciu údajov potrebných pre vytvorenie rozvrhov, poloautomatické (núdzovo aj "ručné") zapísanie/importovanie/exportovanie rozvrhu oprávneným používateľom, formuláciu požiadaviek na rozvrhy od pedagógov a čiastočne aj študentov, ako aj ich zverejňovanie pre rôzne komunity používateľov (so základnou podporou riešenia "konfliktných" požiadaviek a notifikácie pre všetky dotknuté strany), publikovanie rozvrhu pre ostatných používateľov cez internet.

Hlavné úlohy pre tento rok budú dopracovať doteraz vytváraný systém aj s prípadnými úpravami pre podmienky novej budovy (podľa vhodnosti a dostupných kapacít učebni a laboratórií) a s ohľadom na externe zabezpečené prednášky a cvičenia (personálne a priestorovo / najmä väzby s FEI), podsystémy prípravy podkladov, zberu požiadaviek (garantov, prednášajúcich, cvičiacich, študentov), vytvárania a publikovania rozvrhov a nasadiť vytvorený systém do skúšobnej prevádzky a použiť ho pri vytvorení rozvrhu FIIT pre akad. rok 2012/2013.

5 Role a zodpovednosti členov tímu

5.1 Pozície členov tímu

Jednotlivé zodpovednosti a pozície podľa špecifických rolí boli rozdelené medzi členov tímu. Vykonávané činnosti na projekte však nemusia spadať len do týchto kategórií, jedná sa o oblasti z hľadiska manažmentu, ktorým sa členovia tímu venujú primárne.

1. Bc. Viliam Kubis (IS) – Manažment rizík
2. Bc. Radoslav Zachar (SI) – Manažment tvorby dokumentácie
3. Bc. Jakub Calík (SI) – Manažment kvality
4. Bc. Marián Hlavenka (IS) – Manažment komunikácie
5. Bc. Matúš Hitka (IS) – Monitorovanie projektu
6. Bc. Pavol Škvarenina (IS) – Manažment rozvrhu a plánovania
7. Bc. Ondrej Danada (SI) – Manažment podpory vývoja

5.2 Zodpovednosti jednotlivých pozícií

Manažér rizík

- Zodpovedá za predvídanie a včasnú identifikáciu rizík v projekte. Podľa povahy rizika navrhuje, vykonáva a kontroluje vykonávanie rôznych okamžitých ako aj dlhodobých korekčných akcií na zmiernenie alebo úplnú elimináciu všetkých identifikovaných rizík.

Manažér rozvrhu a plánovania

- Zodpovedný za vytvorenie plánu plnenia úloh pre tím a jeho jednotlivých členov a taktiež zadávanie vytvorených úloh do podporného nástroja Redmine. Takisto zodpovedá za vyhodnocovanie plánov a dohľad nad dodržiavaním termínov. V prípade potreby je zodpovedný za modifikáciu plánu.

Manažér kvality

- Zodpovedný za testovanie prototype aj výsledného produktu. Taktiež dohliada na nápravu nedostatkov zistených ri testovaní. Jeho úlohou je dohliadať na to, aby boli dodržiavané dohodnuté pracovné postupy, a aby bol prototype aj výsledný product odovzdaný v požadovanej kvalite.

Manažér dokumentácie

- Zodpovedný za vytvorenie a prípravu šablón, potrebných na písanie dokumentácie, ako aj preberacích protokolov. Jeho ďalšou úlohou je kompletizovať všetky dokumenty vznikajúce v priebehu projektu. Je zodpovedný za výslednú dokumentáciu po formálnej aj obsahovej stránke.

- Zodpovedný za sprístupnenie aktuálnej verzie dokumentácie a pravidelné dopĺňanie zápisníc zo stretnutí na webovej stránke

Manažér podpory vývoja

- Zodpovedá za výber vhodných podporných nástrojov a ich správu. Do toho spadá administrácia používateľských účtov a riešenie konfliktov v SVN dátovom úložisku. Ďalej zodpovedá za fungovanie servera a tiež funkčnosť podporného nástroja Redmine.

Manažér komunikácie

- Zodpovedný za efektívnu komunikáciu v rámci tímu. V prípade vyskytnutia problém je jeho úlohou riešiť požiadavky tímu so zodpovednými osobami. Ďalšou úlohou je poskytnúť potrebné informácie jednotlivým členom tímu. V rámci riadenia projektu podáva hlásenia, prípadne ďalšie nevyhnutné formuláre a dokumenty o priebehu projektu príslušnej osobe.

Manažér monitorovania projektu

- Zodpovedný za doľad nad vývojom projektu. Kontroluje, či sa dodržiavajú dohodnuté pracovné postupy. Je úlohou je vykonávať prehliadky zdrojového kódu.

6 Plán projektu

Plán projektu sa v prvom rade odvíja od chýb nájdených v minuloročnom projekte.

System vyvíjame podľa všeobecného iteračného modelu vývoja. Dlhodobý plán je rozdelený do niekoľkých etáp – iterácií.

V rámci predmetu Tímový projekt I sú naplánované 3 iterácie, pričom každá iterácia obsahuje vopred stanovené úlohy, ktoré sú rozdelené medzi jednotlivých členov tímu.



Figure 1 Všeobecný iteračný model vývoja

6.1 Dlhodobý plán

Dlhodobý plán pozostáva z úloh, z ktorých niektoré súvisia s nájdenými chybami minuloročného systému. Viaceré úlohy sú časovo náročné a možno ich rozdeliť na podúlohy.

Celkový plán a postup jeho vykonávania je zaznamenávaný v softvérovom prostriedku pre manažment úloh – Redmine.

Hlavné úlohy:

- Inštalácia najnovšej verzie Yii a Gii a integrácia so starým systémom
- Pregenerovanie všetkých modulov a controllerov do novej verzii Yii
- Spojazdnenie plnohodnotného loginu cez LDAP
- Vytvorenie „poker hry“ na výber cvičení predmetov a vytvorenie rozvrhu pre študenta

Vývoj nového rozvrhového systému FIIT v rámci Tímového projektu je riadený iteratívnym vývojom. Pri vývoji tohto systému sme sa rozhodli používať metodiku k interatívnemu a inkrementálnemu vývoju softvérového systému.

Tímový projekt je rozdelený medzi dva predmety, Tímový projekt I a Tímový projekt II, aj náš dlhodobý plán je rozdelený na dve etapy: zimnú a letnú etapu. V rámci

každej etapy sú naplánované 3 iterácie (I1, I2 a I3), pričom každá z nich má zvolený interval 4 týždne.

Dlhodobý plán s jednotlivými iteráciami, cieľmi iterácie sa nachádza v nižšie uvedenej tabuľke.

Table 1 Dlhodobý plán

Zimný semester		
Iterácia	Trvanie	Ciel iterácie a hlavné body iterácie
Iterácia I1	5.10 – 2.11	<p>Príprava technických a softvérových prostriedkov</p> <ul style="list-style-type: none"> • Konfigurácia prideleného virtuálneho stroja • Inštalácia zvoleného systému na manažment úloh v tíme • Inštalácia prebraného rozvrhového systému od tímu z predchádzajúceho roku • Zriadenie webového sídla projektu • Zoznámenie sa prebraným systémom (základy aplikačnej logiky, technológie atď)
Iterácia I2	2.11 – 30.11	<p>Zanalyzovanie prebraného systému a vytvorenie dokumentácie k projektu a k riadeniu</p> <ul style="list-style-type: none"> • Analyzovanie prebraného rozvrhového systému • Odhaľovanie chýb prebraného systému • Možnosti vylepšenia prebraného systému • Štúdium aplikačnej logiky prebraného systému a technológií pre pokračovanie vo vývoji (programovacie jazyky, framework atď.) • Vytvorenie dokumentácie k projektu a riadeniu
Iterácia I3	30.11 – 21.12	<p>Implementácia zmien v systéme a odovzdanie upraveného systému spolu s prezentáciou a dokumentáciou</p> <ul style="list-style-type: none"> • Implementácia vybraných vylepšení systému • Opravenie odhalených kritických chýb • Odovzdanie upraveného systému spolu s dokumentáciou • Prezentácia rozšírenia systému

Letný semester

Iterácia	Trvanie	Ciel iterácie a hlavné body iterácie
Iterácia I4	13.2 – 12.3	Implementácia vybraných častí systému <ul style="list-style-type: none"> • Zhodnotenie výsledkov zimného semestra • Implementácia vybraných vylepšení systému • Opravenie odhalených kritických chýb
Iterácia I5	13.3 -9.4	Implementácia, dolad'ovanie, testovanie systému, odovzdanie systému spolu s dokumentáciou <ul style="list-style-type: none"> • Dokončenie implementácie vybraných vylepšení • Oprava chýb a dolad'ovanie nedostatkov • Testovanie systému • Odovzdanie systému spolu s dokumentáciou
Iterácia I6	9.4 – 14.5	Testovacia prevádzka a dokončenie dokumentácie <ul style="list-style-type: none"> • Testovacia prevádzka • Externé testovanie • Údržba odhalených nedostatkov • Dokončenie dokumentácie • Odovzdanie celého produktu spolu s dokumentáciou produktu a riadenia

6.2 Iterácia I1

Prvá iterácia bola zameraná na inštaláciu a integráciu minuloročného systému a potrebných podporných nástrojov pre vývoj.

Podporné nástroje pre vývoj boli zvolené napr.: Redmine, SVN, NetBeans a operačný systém Fedora. Iterácia zahŕňala aj import a úpravu starej databázy pre naše potreby, ako aj prispôbenie celého systému, čo zahŕňalo aj zmenu prístupových práv k jednotlivým tabuľkám databázy.

V prvej iterácii sa tím zamerával aj na vytvorenie webového sídla. Webová prezentácia tímu obsahuje novinky z aktuálneho diania v tíme, informácie o členoch tímu a mailový kontakt na celý tím, dlhodobý plán tímu v rámci tímového projektu, všetky dokumentácie k projektu spolu s preberacím protokolom a aj všetky zápisnice ku každému oficiálnemu stretnutiu tímu.

Table 2 Plán prvej iterácie

ITERÁCIA I1 (5.10 2011 – 2.11 2011)				
ID	ÚLOHA		DOKONČIŤ	ZODPOVEDNÝ
1.1	PREŠTUDOVANIE DOKUMENTÁCIE OD PRECHÁDZAJÚCEHO TÍMU		2.11	J. CALÍK
1.2	SPRÍSTUPNENIE A KONFIGURÁCIA PRIDELENÉHO VIRTUÁLNEHO STROJA		12.10	V. KUBIS

1.3	INŠTALÁCIA A NASTAVENIE WEB SERVERA	12.10	V. KUBIS
1.4	VYTVORENIE ZÁKLADNEJ HTML PREZENTÁCIE TÍMU	12.10	J. CALÍK
2.1	NAINŠTALOVAŤ SYSTÉM REDMINE	19.10	V. KUBIS
2.2	NAKONFIGUROVAŤ SYSTÉM REDMINE (SYSTÉM, PROJEKT, PRÁVA, ÚLOHY)	19.10	P. ŠKVARENINA
2.3	PRÍPRAVA ÚLOH V SYSTÉME REDMINE	19.10	P. ŠKVARENINA
3.1	ZVEREJNIŤ ZÁPISNICE ZO STRETNUTIA NA WEBE	26.10	R. ZACHAR
3.2	ŠTUDOVAŤ FRAMEWORK Yii V ROZSAHU 16 HODÍN	2.11	J. CALÍK
3.3	INŠTALÁCIA PLUG-INU BACKLOGS PRE SYSTÉM REDMINE	26.10	V. KUBIS
4.1	NAKOPÍROVANIE CELÉHO SYSTÉMU OD MINULOROČNÉHO TÍMU	2.11	V. KUBIS
4.2	NAPLNENIE DATABÁZ POTREBNÝMI DÁTAMI	2.11	V. KUBIS

V stĺpci *ID* sa nachádza identifikátor úlohy, pričom prvé číslo udáva týždeň v iterácii (prvý až štvrtý) od ktorého sa má začať pracovať na úlohe, a druhé poradie úlohy v danom týždni. V stĺpci *Dokončiť* sa nachádza dátum, do ktorého je potrebné danú úlohu dokončiť. Stĺpec *Zodpovedný* obsahuje meno člena tímu zodpovedného za danú úlohu, pričom keďže na viacerých úlohách sa podieľajú viacerí členovia tímu, je následne za úlohu zodpovedný kontrolór kvality Bc. Jakub Calík.

6.3 Iterácia I2

Druhá iterácia bola zameraná na analýzu minuloročného systému a štúdium jeho architektúry a navrhutej logiky.

V rámci tejto iterácie sme celý systém preniesli do novej verzie Yii frameworku, a taktiež sme sfunkčnili Gii grafický generátor modulov a controllerov v tomto frameworku. Pochopenie a štúdium frameworku Yii a doinštalovanie grafického generátora modulov Gii a PHPUnit pre tvorbu unit testov jednotlivých modulov systému.

Počas tejto iterácie sa neustále testoval minuloročný systém a hľadali jeho chyby, čo značne prispelo k lepšiemu pochopeniu celého systému. Taktiež bola spísaná analýza celého systému a dokumentácia k riadeniu a prezentácia použitého podporného systému na manažment úloh v tíme.

Zhodnotenie výsledkov predchádzajúcej iterácie:

Prvá iterácia prebehla podľa navrhnutého plánu. Nenastali žiadne problémy, potrebné technické a softvérové prostriedky boli nainštalované a nakonfigurované tak, aby sa tím mohol začať zoznamovať s rozvrhovým systémom z minulého roka a mohol pokračovať vo vývoji rozvrhového systému. Tím si tiež čiastočne naštudoval framework Yii na ktorom je založený prebraný rozvrhový systém. Pri plánovaní prvej iterácie I1 sa pozabudlo na inštaláciu a nakonfigurovanie systému na správu zdrojového kódu Subversion (SVN), čo bude napravené hneď prvý týždeň nasledujúcej iterácie.

Table 3 Plán druhej iterácie

ITERÁCIA I2 (2.11 2011 – 30.11 2011)			
ID	ÚLOHA	DOKONČIŤ	ZODPOVEDNÝ

5.1	NAUČIŤ SA POUŽÍVAŤ REDMINE	9.11	J. CALÍK
5.2	SPOJAZDNIŤ SVN	9.11	V. KUBIS
5.3	ANALÝZA MINULOROČNÉHO SYSTÉMU	2.12	J. CALÍK
5.4	FRAMEWORK YII V RÁMCI MINULOROČNÉHO SYSTÉMU	2.12	J. CALÍK
6.1	MOŽNOSTI GRAFOV V SYSTÉME REDMINE	16.11	P. ŠKVARENINA
6.2	DOKONČIŤ ANALÝZU A NÁVRH RIEŠENIA	16.11	R. ZACHAR
7.1	PRIEBEŽNÉ SPISOVANIE ZISTENÝCH CHÝB V MINULOROČNOM SYSTÉME	30.11	J. CALÍK
7.2	ROZCHODENIE VYTVÁRANIA ROZVRHOVÝCH AKCIÍ	23.11	V. KUBIS
7.3	SKRIPTU PRE AUTOMATICKÝ CHECKOUT NAJNOVŠIEHO KÓDU Z SVN DO WEBROOTU	23.11	V. KUBIS
8.1	INŠTALÁCIA NAJNOVŠIEHO YII FRAMEWOKU	30.11	J. CALÍK
8.2	PREGENEROVANIE MODULOV A VYTVORENIE CONTROLLEROV PRE NOVÚ VERZIU YII FRAMEWORK	30.11	J. CALÍK
8.3	SFUNKČNENIE LDAP PRIHLÁSENIA DO SYSTÉMU	30.11	J. CALÍK

6.4 Iterácia I3

Tretia iterácia zahŕňala prezentáciu dosiahnutých výsledkov v rámci predmetu Tímový projekt I. spolu s dokončením dokumentácie k riadeniu a dlhodobého plánu za celý prvý semester trvania tohto predmetu.

V tejto iterácii boli postupne opravené niektoré odhalené chyby a finálne zmenený framework, na ktorom beží systém. Framework je momentálne v najvyššej dostupnej verzii a všetky jeho moduly sú plne funkčné a pregenerované do tejto verzie. Taktiež je funkčné prihlásenie pomocou LDAP, kvôli prepojeniu so školským systémom AIS.

Zhodnotenie výsledkov predchádzajúcej iterácie:

Prvá polovica Iterácie I2 prebehla podľa zvoleného plánu. Počas prvého týždňa boli zistené nedostatky nainštalovanej verzie frameworku Yii, ktoré vyústili k zjemneniu pôvodného plánu, pričom bola naplánované inštalácia novšej verzie frameworku Yii a oprava chýb, ktoré boli dovtedy identifikované. Úlohy 8.2 a 8.3 sa nepodarilo dokončiť úplne a prechádzajú tak do nasledujúcej Iterácie I3.

Table 4 Plán tretej iterácie

ITERÁCIA I3 (30.11 2011 – 21.12 2011)			
ID	ÚLOHA	DOKONČIŤ	ZODPOVEDNÝ
8.2	PREGENEROVANIE MODULOV A VYTVORENIE CONTROLLEROV PRE NOVÚ VERZIU YII FRAMEWORK	13.12	J. CALÍK
8.3	SFUNKČNENIE LDAP PRIHLÁSENIA DO SYSTÉMU	13.12	J. CALÍK
9.1	VYTVORENIE PREZENTÁCIE PRE ODOVZDANIE PROTOTYPU ZA PRVÝ SEMESTER	13.12	J. CALÍK
10.1	DOKONČENIE DOKUMENTÁCIE K RIADENIU	13.12	J. CALÍK
10.2	DOKONČENIE A KONSOLIDÁCIA PROTOTYPU PRED ODOVZDANÍM	13.12	J. CALÍK
10.3	DOHODNUTIE DÁTUMU KONANIA PREZENTÁCIE S DRUHÝM TÍMOM	14.12	M. HLAVENKA

11.1	ODOVZDANIE PROTOTYPU A DOKUMENTÁCIE ZA PRVÝ SEMSTER	14.12	J. CALÍK
11.2	PREZENTÁCIA PROTOTYPU	12-16.12	J. CALÍK

6.5 Návrhy na vylepšenie systému

System pre tvorbu rozvrhov, by bolo vhodné vylepšiť o zlepšený algoritmus pridelovania termínov predmetov, aby sa zminimalizoval počet „medzier“ medzi jednotlivými predmetmi študentov. Toto vylepšenie má však menšiu prioritu, keďže prvoradá je sfunkčnenie samotného systému pridelovania cvičení podľa stávk študentov.

V ďalšej verzii systému by sme navrhovali vytvorenie miesta pre komunikáciu študentov (fórum, chat...), ktoré by umožňovalo študentom diskusiu o termínoch jednotlivých cvičení, prípadne obchod s termínmi cvičení.

7 Podporné prostriedky

7.1 Redmine

Pre správu úloh a plánovanie jednotlivých iterácií projektu používame podporný nástroj Redmine.

Tento systém umožňuje nielen plánovanie, ale aj pridelenie jednotlivých úloh členom, zaznamenávanie postupu práce na úlohách a aj ich ukončenie. Umožňuje jednoducho generovať výstupy práce do niekoľkých typov grafov, čím sprehľadňuje a uľahčuje dohľad nad projektom.

Do systému sú vkladané jednotlivé tasky a supertasky, pričom percentá ukončenia hlavných úloh sa automaticky prerátavajú zo všetkých podúloh a ich postupným ukončovaním.

Naša vlastná inštalácia Redmine systému je dostupná na webovej adrese: <http://vm07.ucebne.fkit.stuba.sk:443/>.



Figure 2 Redmine nástroj

7.2 SVN

Na správu verzií nášho systému používame SVN. Subversion (SVN) je systém na správu verzií. V súčasnosti je jeden z najpoužívanejších podporných programov vytvorených pre tento účel.

Subversion funguje na báze klient - server. Serverová časť je súborový server - repository, ktorý si pamätá všetky zmeny súborov a štruktúry priečinkov.

Každý z členov tímu má checkoutnutú vlastnú lokálnu verziu, ktorú upravuje. Stabilná verzia je uložená na serveri.

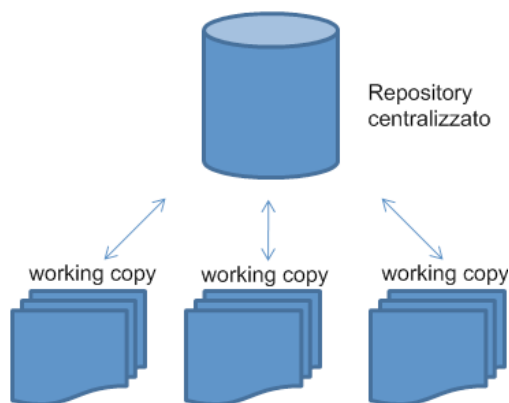


Figure 3 SVN Schéma

7.3 Mailing-list

Na komunikáciu v tíme sme vytvorili mailing-list, aby mal každý člen tímu prehľad, čo sa aktuálne robí na projekte.

Vytvorený tímový mail je: tim6@danada.sk.



Figure 4 Mailing-list komunikácia

8 Aplikácia poznatkov

8.1 Manažment rizík

O našom tímovom projekte - Rozvrhári 2.0

Manažment rizík v tímovom projekte tvorí podobne ako pri iných softvérových projektoch veľmi dôležitú súčasť životného cyklu softvérového projektu, a preto je v rámci nášho tímového projektu na jeho vykonávanie a zabezpečovanie kladený veľký dôraz.

Keďže zme zdrojový kód nášho tímového projektu preberali od minuloročného tímu s cieľom pokračovať vo vývoji, ovplivňovali náš projekt mnohé iné riziká ako v prípade projektu vyvíjaného od začiatku, ktoré boli spojené s údržbou zdrojového kódu, jeho porozumením, modifikáciami a zmenami požadovanej funkcionality.

Preto sa už na začiatku tímového projektu zvolila vhodná metóda manažmentu rizík, ktorá v našom prípade predstavuje neustály manažment rizík (CRM - continuous risk management). Táto metóda je veľmi vhodná pre malé, často sa stretávajúce tímy pracujúce u jedného zamestnávateľa v jednej budove (v škole), nakoľko v takýchto tímoch je dokonale podporovaný prirodzený princíp otvorenej komunikácie, čo je základom pre úspech neustáleho manažmentu rizík. Riziká v našom tíme teda môžu byť (aj boli) komunikované neustále, minimálne na každom tímovom stretnutí alebo v prípade potreby keďkoľvek pomocou e-mailovej komunikácie.

Mnohé riziká v našom tímovom projekte boli úspešne eliminované, niektoré sa podarilo zmierniť, iné aspoň trochu obmedziť.

Manažment rizík - ako sme ho prevádzkovali

Na začiatku projektu bol vyplnený dotazník, ktorý odhalil niektoré pôvodné prvotné riziká a vďaka technickým otázkam sa pomohol viacerým členom tímu dôkladnejšie zorientovať v problematike. Počas projektu bolo nájdených viacero rizík, ktoré popisujeme v nasledujúcich kapitolách.

Prvotná analýza rizík a ich zvládnutie

Prvotná analýza rizík (dotazník a rozhovor) odhalil prvý vážny problém - veľa členov nášho tímu neovládalo skriptovací jazyk PHP, iní, ktorí ovládali jazyk PHP, neovládali framework Yii, v ktorom je projekt napísaný. Dá sa povedať, že nikto v tíme nemal s Yii skúsenosti, niektorí o ňom už počuli, iní ho videli v akcii ale to je asi tak všetko.

Taktiež hrozilo pokročilé riziko nedostatku času, keďže sme si väčšina dali o jeden predmet navyše (aby sme potom nemuseli mať toľko práce v treťom semestri spolu s diplomovou prácou), čiže sme boli viacej vyťažení ako obvykle. Aj so štandardným rozvrhom však škola dáva dosť zabráť, čiže toto riziko sme vnímali ako veľké a podstatné - snažili sme sa ho zmierniť včasným (veľmi včasným) vypracovávaním úloh, ktoré začalo ihneď po prvom tímovom stretnutí.

Členovia sa začali učiť PHP, a ja som pracoval spolu s inými na rozbehaní web servera pre naše tímové potreby - aby sa projekt mohol čo najskôr začať.

Ďalšie riziká, s ktorými sme sa zapodievali, bolo riziko odchodu členov, avšak toto sme nepokladali, ani doteraz nepokladáme za reálne, pretože všetci chceme školu dokončiť a dúfame, že nikto z našich členov nebude mať dlhotrvajúce zdravotné problémy ktoré by ohrozili jeho schopnosť spolupráce v tíme.

Riziko odchodu členov (a teda riziko nedostatku personálu) by sme nemohli aj tak zmieniť, pretože nemáme možnosť do projektu najat' nových ľudí (keďže je to v rozpore s pravidlami predmetu tímový projekt).

Ako poslednú okamžitú korekčnú akciu sme naplánovali pravidelné stretnutia s minuloročným tímom, s ktorým sme na každom stretnutí preberali nejasnosti a videné budúce ťažkosti alebo problémy v projekte, ktoré boli vždy kladne vyriešené.

Riziká objavené počas vývoja projektu

Počas vývoja projektu sme odhalili viaceré nové riziká, ktoré boli spôsobené viacerými faktormi - medzi jedno z najhlavnejších patrilo (aj patrí) riziko nezvládnutia technológie, pretože nám programovanie s pomocou Yii frameworku príliš nejde, skôr naopak, aj keď sa Yii pravidelne učíme. Faktom však zostáva že tak obrovským frameworkom sa v tak krátkom časovom období nedá naučiť pracovať na profesionálnej, ani efektívnej úrovni.

Navyše náš projekt ovplyvnili rôzne chyby a nedostatky samotného frameworku Yii, ktoré sme nedokázali nijak obísť, čo vyústilo až do kompletného upgrade frameworku Yii na najnovšiu verziu (ktorá však nie je spätne plne kompatibilná) a prepisú stávajúceho kódu do novej verzie Yii.

Ťažkosťami s vývojom pomocou Yii frameworku sa ďalej prehĺbuje riziko nedodania projektu na čas, avšak nie je tak výrazné ako v čase, keď ešte 80% tímu neovládalo ani jazyk PHP.

Pri vývoji webových stránok však znalosť PHP a Yii nestačí, a aj keď nám Yii framework ponúka naozaj veľké a pestré množstvo "widgetov" alebo rôznych iných vstavaných funkcií, ktoré pracujú priamo s HTML alebo JavaScript výstupom pre používateľa, v mnohých prípadoch je dodatočná znalosť HTML, JavaScript a CSS3 potrebná - napríklad pri navrhovaní vlastných obrazoviek, ako bude hra o cvičenia.

Takéto obrazovky budú mať na starosti členovia tímu, ktorí majú dokonalé skúsenosti s tvorbou webových stránok, pretože celý frontend bude potrebné napísať od začiatku - Yii nám v tomto príliš nepomôže (čo ale samozrejme ani neočakávame).

Aktuálny stav projektu

Aktuálne sa projekt nachádza v štádiu, kedy sa nám podarilo úspešne rozbehnúť novú verziu platformy Yii a zároveň spojzdníť staré zdrojové súbory, ktoré boli nekompatibilné s najnovším frameworkom Yii. Projekt je teda pripravený na ďalší vývoj - ktorý bude spočívať v kompletnom doladení celého používateľského rozhrania a implementácii viacerých nových obrazoviek a funkcionalít - vrátane stávk na cvičenia, hry o cvičenia, skupinového stávkovania a podobne.

Z pohľadu rizík stále pretrvávajú dve hlavné riziká - riziko nezvládnutia technológie a riziko nedostatku času. Pri prepisovaní kódu na novšiu verziu sa objavilo aj riziko neporozumenia minuloročného kódu, avšak bolo rýchlo eliminované jedným opatrením - kód, ktorý nerozumieme, zmažeme a vygenerujeme, modifikujeme na novo. Keďže sa jednalo najmä o kusy vygenerovaného kódu Yii frameworkom, ktorý bol upravený pre potreby aplikácie, tento štýl kódu sa s novým Yii frameworkom zmenil. Preto musel byť väčšinou kompletne prepísaný, teda nebolo časovo efektívne ani dôležité pokúsiť sa starému kódu plne porozumieť, stačilo odpozorovať správanie sa danej časti aplikácie a operáciu implementovať znovu, s použitím najnovšieho frameworku Yii.

Keďže je už veľká časť projektu plne implementovaná, iné operácie za nás plne kryje framework Yii, práce nezostáva až tak veľa, aby sa to za ďalší semester nedalo zvládnuť. Riziko nedostatku času stále hrozí, avšak je omnoho menšie ako v minulosti (na začiatku projektu).

8.2 Manažment kvality

Výsledný systém tímového projektu má slúžiť ako nástroj na vytváranie rozvrhov pre pedagógov, ako aj pre študentov. Základnou požiadavkou pre tento systém preto je, aby systém spĺňal najvyššie kritériá kvality z hľadiska funkčnosti a bezpečnosti. Aby sme tieto požiadavky zabezpečili, museli sme ešte pred samotným vývojom prijať viaceré rozhodnutia a schváliť adekvátne pracovné postupy.

Pracovné postupy (metodiky):

- Manažment iterácií (*Pavol Škvarenina*)
- Tvorba dokumentácie a proces dokumentácie zdrojových kódov (*Radoslav Zachar*)
- Manažment testovania a tvorba unit testov (*Jakub Calík*)

Všetky vykonávané úlohy by sa takto mali riadiť metodikou príslušnej kategórie úlohy, pod ktorú jednotlivé úlohy spadajú. Toto bola však naša interná dohoda, a vzhľadom k tomu, že väčšina úloh bola vypracovávaná idividuálne z domu, nie je preto možné skontrolovať, či boli 100% dodržiavané pracovné postupy pri všetkých úlohách. Napriek tomu všetky úlohy musia prejsť duplicitnou kontrolou splnenia, zo strany manažéra kvality, ktorý má za úlohu dodatočne overiť nielen či je daná úloha skutočne splnená, ale aj či je splnená s dostatočnou kvalitou (nevykazuje žiadne ďalšie bugy, prípadne neželané a neočakávané chovanie sa systému). Do tejto kontroly, a súčasne aj pod povinnosti manažéra kvality, spadá aj kontrola používania a plnenia úloh v podpornom nástroji Redmine. V Redmine sú všetky úlohy hierarchicky usporiadané tak, že každá jednotlivá úloha má priradenú nadúlohu, a za splnenie tejto nadúlohy zodpovedá samotný manažér kvality.

Ďalšou internou tímovou dohodou bolo, že jednotlivé upravované moduly systému a ich funkcionality budú testované pomocou unit testov. Pre tento účel sa bude používať nástroj PHP Unit, ktorý je možné priamo zintegrovat' do Yii frameworku.

Ako už bolo spomenuté v časti „Podporné nástroje“, na správu verzií systému využíva náš tím SVN server. Za neustálu dostupnosť najnovšej funkčnej verzie taktiež zodpovedá manažér kvality.

V prípade, že manažér kvality pri kontrole odhalí akékoľvek funkčné alebo kvalitatívne nedostatky v systéme, musí znovuotvoriť úlohu v redmine s príslušným komentárom s odôvodnením tejto akcie a následne informovať osobu, ktorá úlohu vykonávala. Príslušná osoba je potom zodpovedná za nápravu chyby alebo nedostatkov do dohodnutého termínu splnenia príslušnej úlohy.

8.3 Manažment tvorby dokumentácie

Manažér tvorby dokumentácie v tíme je zodpovedný za:

- vytvorenie a priebežné doplňovanie dokumentácie projektu
- prípravu šablón pre zápisnice zo stretnutí a preberacie protokoly
- udržiavanie obsahu prezentačnej web stránky.
- zverejňovanie publikovateľných verzií dokumentácie k riadeniu na webe
- vytvorenie prezentácie projektu s výsledkami práce

Medzi jeho povinnosti nevyhnutne patrí aj zbieranie dokumentov písaných ostatnými členmi tímu a ich kompletizácia do jedného čitateľného dokumentu.

Tvorba dokumentácie

Tvorba dokumentácie je dôležitý proces pri vytvaraní softvérového produktu.

Pri tvorbe dokumentácie sa tím dohodol, že kostra dokumentu bude zložená z:

- titulného listu obsahujúceho názov projektu a zoznam všetkých členov tímu
- historie zmien dokumentu v tabuľke s dátumami zmien
- aktuálny obsah aj s prílohami
- zoznam obrázkov, tabuliek, diagramov a grafov použitých v dokumente
- kapitolu Úvod
- motiváciu spísanú celým tímom spoločne, ktorá slúžila na získanie tohto projektu
- koncepciu riešenia podľa tímu a tímových schopností
- presného textu získaného zadania
- príloh

Zápisnice zo stretnutí

Oficiálne tímové stretnutia sa konali každý týždeň vždy v stredu. Obsah jednotlivých stretnutí je zaznamenaný v zápisnici zo stretnutia.

Zápisnica má minimálne jednu A4 a obsahuje zoznam prítomných členov, tému a obsah stretnutia, opis stavu úloh, ktoré bolo potrebné vyriešiť a aj zoznam úloh, ktoré sme sa dohodli splniť do ďalšieho stretnutia.

Zápisnice boli písané podľa šablóny, ktorú sme si ako tím spoločne dohodli. Všetky zápisnice sú priložené v prílohe a tak isto aj zverejnené na webovej stránke tímu. Každá zápisnica má jedného autora a jedného kontrolóra, ktorý je zodpovedný za pravdivosť a korektnosť obsahu.

Po zapísaní zápisnice a jej skontrolovania, manažér tvorby dokumentácie zverejnil túto zápisnicu na internete, väčšinou do dvoch pracovných dní od konania stretnutia.

Preberacie protokoly

Preberacie protokoly boli tiež vytvorené podľa šablóny. Na obsah tejto šablóny boli kladené nasledovné požiadavky:

- musí obsahovať názov nášho tímu
- meno preberacej osoby
- mailový kontakt na celý tím
- predmet prebratia
- dátum a miesto prebratia
- podpisy oboch zúčastnených strán

Protokoly sú samozrejme v prílohách a aj na webovej stránke.

Web stránka

Webová stránka slúži ako prezentácia nášho tímu a našej práce. Za dizajn a aj obsah je zodpovedný manažér dokumentácie. Dizajn sme si navrhli spoločne ako tím.

Stránka obsahuje niekoľko sekcií:

- Novinky
- O nás
- Použité technológie
- Zápisnice
- Dlhodobý plán
- Dokumentáciu

Okrem týchto sekcií stránka obsahuje aj úvod, kde máme spoločnú tímovú fotografiu.

Záložka Dokumentácia obsahuje tímovú dokumentáciu, dokumentáciu k riadeniu a aj preberacie protokoly. Záložka dlhodobý plan obsahuje len náš spoločný dlhodobý plán, ktorý je ešte podrobnejšie opísaný v dokumentácii. V záložke Zápisnice sú zverejnené postupne všetky zápisnice v poradí, v akom sa konali stretnutia. V záložke použité technológie je kompletný zoznam technológií, ktoré ako tím využívame od technológii nevyhnutných pre vytváranie system až po technológie, ktoré sú potrebné pre server alebo webové sídlo. Predposledná záložka obsahuje informácie o každom členovi a jeho fotografiu. A nakoniec medzi novinkami nájdete novinky, ktoré sa udiali v tíme s výnimkou toho, že sme sa ako tím dohodli, že zverejňovanie zápisníc do noviniek nebudeme pridávať.

Stránka je pravidelne aktualizovaná minimálne raz týždenne uploadovaním zápisníc a aktuálnej dokumentácie.

8.4 Manažment podpory vývoja

Redmine

Pre rozdelenie úloh, reportovanie bugov a ich priradovanie jednotlivým členom tímu využívame podporný nástroj Redmine, ktorý je umiestnený na našom tímovom serveri. Redmine spravuje a prácu v ňom rozdeľuje manažér plánovania.

SVN

Verzie zdrojových kódov si spravujeme v nástroji SVN, ktorý beží na báze klient-server. Klientov používajú jednotliví členovia tímu rôznych v závislosti od osobných preferencií. Pre všetkých členov tímu však platí, že každý commit musí byť stručne, jasne a výstižne pomenovaný.

Mailing list

Náš tímový mail je *tim6@danada.sk*, mailová schránka je zriadená na serveri manažéra podpory vývoja, od kiaľ sú maily automaticky preposielané na adresy jednotlivých členov tímu. Mailová skužba, ktorá beží na serveri, je Kerio Mail Server.

Textový editor

Pre udržanie konzistencie štýlov a vzhľadu dokumentácie používame iba jeden textový editor, a to OpenOffice. Z tohto dôvodu všetci členovia tímu dokument ani needitujú, ale odosielajú svoje príspevky manažérovi dokumentácie, ktorý ich do dokumentácie zapracúva.

CASE nástroj

Na kreslenie diagramov používame Enterprise Architect a na kreslenie GUI generátor, ktorý poskytuje framework YII.

Vývojové prostredie

Zatiaľ nemáme dohodnuté žiadne jednotné vývojové prostredie. Každý člen tímu píše zdrojový kód vo vlastnom textovom editore a ladí podľa chybových správ z internetového prehliadača.

8.5 Manažment komunikácie

Manažér komunikácie ma na starosti zabezpečenie prenosu informácií v rámci tímu, ale aj dodania potrebných správ a dokumentov kompetentným osobám. Taktiež rieši získavanie potrebných informácií pre potreby členov tímu.

Komunikácia v tíme a s vedúcim projektu Ing. Galbavým

Ako hlavnú formu komunikácie sme zvolili osobné stretnutia, ktoré sa pravidelne konajú každý týždeň v stredu od 11:00 do 13:30. V rámci osobných stretnutí sme mali možnosť rýchlej výmeny informácií a názorov, ako aj možnosť spoločného rozhodnutia a riešenia nájdených problémov projektu. Taktiež sme diskutovali smer, ktorým projekt budeme viesť, medzi sebou v tíme a zároveň aj s vedúcim projektu pánom Ing. Galbavým.

Druhým využívaným komunikačným kanálom je e-mail. Prostredníctvom e-mailu sme sa informovali a upozorňovali na dodržanie potrebných termínov alebo sme zistovali a preposielali potrebné informácie a dokumenty pre schopnosť dôkladného a správneho vykonávania priradenej úlohy.

Zvolili a používali sme telefónne hovory a telefónnu konferenciu cez Skype ako naše alternatívne komunikačné kanály, v prípade potreby riešenia obširných problémov, ktoré nemohli počkať do ďalšieho osobného stretnutia.

Komunikácia s minuloročným tímom

Prvé stretnutie nášho tímu s minuloročným tímom, na ktorom sme prezali ich systém bolo zorganizované prostredníctvom vedúceho projektu pána Ing. Galbavého. Stretnutie sa konalo počas druhého osobného stretnutia 12.10.2011. Okrem zverenia nimi vyvinutého systému nám dali aj rady ako postupovať a ako sa stavať k riešenie tohoto projektu a zverili nám aj ich skúsenosti s používanými technológiami v rámci práce na projekte, ako aj s podpornými nástrojmi na organizáciu práce. Posledným bodom nášho prvého stretnutia s minuloročným tímom bola výmena kontaktných informácií vo forme e-mailov a telefónnych čísiel.

Jednoduché otázky sa riešili priamo cez e-mail, ale pri tých zložitejších, ako vysvetľovanie funkcionality niektorej časti systému, bolo potreba dohodnúť ďalšie stretnutie. Aby sme zbytočne časovo nezaťažovali všetkých členov minuloročného tímu, bolo organizovanie ďalších stretnutí riešené priamo cez ich vedúceho tímu, ktorý nám povedal kto mal na zodpovednosti danú časť systému, a následne nám aj pomohol skontaktovať danú osobu a aj dohodnúť potrebné osobné stretnutie.

Odobzdanie potrebných dokumentov ku kontrolným bodom

Samotná tvorba dokumentov prebiehala individuálne, ale bolo treba aj tieto príspevky každého člena tímu spojiť dokopy tak, aby aj dávali nejaký zmysel. Určili sme si na to miesto pri našich osobných stretnutiach, ktoré sme mali naplánované na každú stredu. Nápady na možnú úpravu dokumentu boli zdieľané na stretnutiach ale aj cez e-mail a telefónne hovory. Po úspešnom dokončení dokumentu, alebo iba jeho časti, bolo zo strany manažéra komunikácie požadované jej odovzdanie kompetentnej osobe. Súčasťou odovzdávky je preberací protokol, na ktorý sa manažér komunikácie a preberajúca osoba podpisujú.

Teambuilding

Okrem naplánovaných osobných stretnutí sa uskutočňovali a aj ďalej budú uskutočňovať neformálne stretnutia, na ktorých sme riešili problémy, ktoré niekedy aj súviseli s projektom, alebo sme sa iba bližšie spoznávali. Takéto neformálne stretnutia boli návštevy školskej jedálne alebo pizzérie na obed.

8.6 Manažment monitorovania projektu

Počas realizácie projektu sa monitorujú jeho 2 hlavné časti:

- stav projektu
- zdrojové kódy

Monitorovanie stavu projektu

Na monitorovanie stavu projektu sa používa nástroj na podporu manažmentu vývoja RedMine. Monitoruje sa priebeh plnenia jednotlivých úloh. Na začiatku projektu sa vytvorí dlhodobý plán pre jednotlivé iterácie. Pred začiatkom každej iterácie sa podrobnejšie naplánuje jej priebeh.

Pre každú úlohu je odhadnutý čas jej vykonávania. Riešiteľ každej úlohy má v systéme RedMine priebežne aktualizovať jej stav, t.j. koľko času venoval jej riešeniu a na koľko percent je úloha hotová. Podľa toho je možné kontrolovať priebeh plnenia jednotlivých úloh i celých iterácií. V prípade, že aktuálny stav zaostáva za očakávaným stavom, je možné prijať potrebné opatrenia, ako napr. preplánovanie úloh či presunutie zdrojov na plnenie daných úloh tak, aby v konečnom dôsledku došlo k dodržaniu pôvodného plánu.

Písanie zdrojových kódov

Táto kapitola opisuje štábnu kultúru tímu pri písaní zdrojových kódov. V distribuovanom projekte je nutné zachovanie konzistencie kvôli lepšej čitateľnosti a organizácii kódu. Dodržiavaním týchto pravidiel sa zamedzí vzniku konfliktov a nedorozumení, ktoré by mohli vyplynúť z rôznych štýlov programovania.

Významný vplyv na samotnú štábnu kultúru má použitie frameworku Yii, ktorý automaticky generuje množstvo kódu, ale aj spôsob názvoslovia, ktoré používal predchádzajúci tím *Schedule of Pain*.

Pravidelne sa vykonávajú prehliadky zdrojových kódov. Pri zistení porušenia niektorého z nasledovných pravidiel je potrebné čím skôr opraviť konkrétnu časť kódu.

Základné pravidlá

Názvoslovie premenných, tried, metód a iných kľúčových slov je v **angličtine**, avšak komentáre sú písané v slovenčine. Texty v komentároch sa píšu bez diakritiky kvôli lepšej prehľadnosti, ale napr. i pre lepšie vyhľadávanie v textoch.

Premenné

- Názvy premenných sú písané malými písmenami.
- Pri viacslovných názvoch premenných sa používa *lowerCamelCase* spôsob pomenovania, nepoužívajú sa podtrhovníky.
- Názvy premenných treba voliť tak, aby z nich bolo možné určiť ich zmysel.
- Globálne premenné začínajú malým písmenom `g`: `global gLog;`

Konštanty

- Všetky konštanty sú písané veľkými písmenami.

Entity

- Pre názvy entít sa používa spôsob *CamelCase* pomenovania.
- Názvy tried, metód a funkcií začínajú veľkým písmenom: `class UsersController`
- Atribúty tried začínajú malým písmenom *m* kvôli odstráneniu možného konfliktu s názvami metód:

```
class NameOneTwo
{
    function VarAbc() {};
    function ErrorNumber() {};
    var $mVarAbc;
    var $mErrorNumber;
    var $mrName;
}
```

- Názvy premenných, ktoré sú referenciami, začínajú s písmenom *r*. Názvy funkcií a metód vracajúcich referenciu tiež začínajú s *r*:

```
class Test
{
    var $mrStatus;
    function DoSomething(&$rStatus) {};
    function &rStatus() {};
}
```

Osadenie textu

Každý vnorený blok textu musí byť osadený tabulátorom o jednu pozíciu vpravo:

```
if (condition1)
{
    if (condition2)
    {
        someCommand;
    }
}
```

Komentáre

Používa sa formát komentárov kompatibilný so systémom *phpDocumentor*.
Pravidlá pre komentovanie jednotlivých elementov:

- Súborny - uviesť stručne obsah súboru (implementované triedy –jedna trieda na jeden súbor).

```
/**
@Nazov          [nazov suboru]
-----
@Popis          [popis suboru, na co sluzi a co sa v nom
nachadza]
@Kategoria     [typ suboru ]
@Autor         [meno autora suboru]
@Verzia
*/
```

- Triedy – uviesť, na aký účel je trieda určená.

```
/**
@Nazov          [nazov triedy]
-----
@Popis
                [popis triedy, na co sluzi]
@Autor         [meno autora]
*/
```

- Premenné a atribúty - význam premenných a atribútov a ich očakávané typy
- Metódy - uviesť význam vstupných parametrov, návratových hodnôt a stručne opísať účel konkrétnej metódy

```
/**
@Popis          [popis funkcie]
@param1        [popis vstupneho parametru]
@param2        [popis vstupneho parametru]
                .
                .
                .
@return         [popis navratovej hodnoty metody]
*/
```

9 Príloha A (Odovzdaná ponuka)

10 Príloha B (Preberací protokol)

11 Príloha C (Metodiky)

12 Príloha D (Zápisnice zo stretnutí)