

Bug Hunters

System pre podporu tvorby rozvrhov

(Tvorba softvérového systému v tíme)

Autori (tím 19): Vladimír Krajčovič, Dušan Palo, Peter Líška, Matej Krchniak,
Samuel Števaňák, Ondrej Buch, Peter Mikuška
Vedúci tímu: Ing. Miroslav Galbavý
Študijný odbor: Softvérové inžinierstvo, Informačné systémy
Ročník: 1. ing.
Akademický rok: 2009/2010
Kontakt: tp0910_tim19@googlegroups.com

Obsah

1	ZADANIE	1
2	ÚVOD	2
2.1	ÚČEL A ROZSAH DOKUMENTU	2
2.2	SKRATKY A POUŽITÁ NOTÁCIA	2
2.3	POUŽITÁ LITERATÚRA.....	3
3	ANALÝZA PROBLÉMU TVORBY ROZVRHOV	4
3.1	VSTUPNÉ ÚDAJE PRE PROCES TVORBY ROZVRHU	4
3.2	ZBER ÚDAJOV PRE SEMESTRÁLNY ROZVRH	5
3.3	TVORBA SEMESTRÁLNYCH ROZVRHOV	8
3.4	PREPOJENIE S AIS.....	8
3.5	HRAČI V SYSTÉME	9
4	ŠPECIFIKÁCIA POŽIADAVIEK	10
4.1	NEFUNKCIONÁLNE POŽIADAVKY	10
4.2	FUNKCIONÁLNE POŽIADAVKY	10
5	ANALÝZA EXISTUJÚCEHO RIEŠENIA	12
5.1	KONTROLA POŽIADAVIEK NA SYSTÉM	12
5.2	ARCHITEKTÚRA EXISTUJÚCEHO SYSTÉMU	13
5.3	ÚPRAVY FYZIKÉHO DÁTOVÉHO MODELU	14
5.4	EVIDENCIA A OPRAVA CHÝB EXISTUJÚCEHO SYSTÉMU.....	15
6	TESTOVACIA PREVÁDZKA PROTOTYPU ZS 09/10	19
6.1	ANALÝZA PRIPOMIENOK Z TESTOVACEJ PREVÁDZKY	20
7	IMPLEMENTÁCIA	21
7.1	ZAPRACOVANIE PRIPOMIENOK K SYSTÉMU ZO ZIMNÉHO SEMESTRA 09/10	21
7.2	ROZŠÍRENIE A ÚPRAVA FUNKCIONALITY SYSTÉMU	22
7.3	KONKRÉTNÉ VYPRACOVANÉ ÚLOHY	25
8	TESTOVACIA PREVÁDZKA PROTOTYPU LS 09/10	28
9	NÁVRH ROZŠÍRENÍ A ÚPRAV EXISTUJÚCEHO SYSTÉMU	30
9.1	NÁVRH ĎALŠEJ PRÁCE NA PROJEKTE.....	30
10	PRÍLOHY	34
PRÍLOHA A	DIAGRAM TRIED SYSTÉMU VYTVORENÉHO V ŠKOLSKOM ROKU 2008/2009	
PRÍLOHA B	MODEL ÚDAJOV SYSTÉMU ZO ŠKOLSKÉHO ROKU 2008/2009	
PRÍLOHA C	MODEL ÚDAJOV SYSTÉMU PO ÚPRAVÁCH V ZIMNOM SEMESTRI ŠKOL. ROKU 2009/2010	
PRÍLOHA D	PRIPOMIENKY K SYSTÉMU V ZIMNOM SEMESTRI ŠKOLSKÉHO ROKU 2009/2010	
PRÍLOHA E	NÁVRH NA ÚPRAVU SKUPÍN POUŽÍVATEĽOV A ICH PRIVILÉGIÍ	
PRÍLOHA F	AKCEPTAČNÉ TESTY LS 2009/2010	
PRÍLOHA G	POUŽÍVATEĽSKÁ PRÍRUČKA	
PRÍLOHA H	INŠTALAČNÁ PRÍRUČKA	
PRÍLOHA I	OBSAH ELEKTRONICKÉHO MÉDIA	

1 Zadanie

Napriek nasadeniu systému AIS, zostáva stále otvorená problematika vytvárania semestrálnych a skúškových rozvrhov. V súčasnosti je rozpracovaný špecializovaný systém pre tvorbu rozvrhov, ktorý už teraz umožňuje:

- evidenciu údajov potrebných pre vytvorenie rozvrhov,
- samotné ("ručné") vytvorenie/importovanie/exportovanie rozvrhu oprávneným používateľom,
- formuláciu požiadaviek na rozvrhy od pedagógov, ako aj ich zverejňovanie pre rôzne komunity používateľov (so základnou podporou riešenia "konfliktných" požiadaviek),
- sprístupnenie rozvrhu pre ostatných používateľov cez internet.

Vzhľadom na doterajší dlhodobější vývoj tohto systému a zmeny v prostredí fakulty, treba kriticky a tvorivo prehodnotiť všetky používateľské požiadavky, navrhnúť nutné zmeny systému a následne ich implementovať. Zmeny by mali odzrkadľovať najmä:

- širšie údajové prepojenie na akademický informačný systém FIIT a STU - AIS,
- komplexné bezpečnostné podmienky prevádzkovania celého systému na FIIT (aj za FEI - pre zdieľané prostriedky) a kompletizáciu príslušnej dokumentácie,
- zdokonalenie správy používateľov,
- experimentálna prevádzka pri príprave rozvrhu na LS 2009/2010,
- vyhodnotenie experimentálnej prevádzky,
- rozšírenie podsystému formulovania meniacich sa požiadaviek na rozvrhy od pedagógov a študentov, ako aj ich riadeného sprístupňovania pre rôzne komunity používateľov (detailný ako aj prehľadový formát),
- kvantitatívnu analýzu požiadaviek z hľadiska kapacitných možností FIIT/FEI,
- efektívnejšiu podporu riešenia konfliktov požiadaviek,
- analýzu momentálneho stavu rozvrhu s identifikovaním vznikajúcich "konfliktov" a vyvolanie "riadeného riešenia konfliktu".

V zimnom semestri bolo hlavnou úlohou analyzovať momentálny stav systému, pričom bolo potrebné sústrediť sa na funkčnosť, korekciu existujúcich chýb a doplnenie nevyhnutne potrebných požiadaviek na systém. Výsledkom práce v zimnom semestri bolo spustenie experimentálnej prevádzky systému. Úlohou na letný semester bolo predovšetkým vyhodnotenie a zapracovanie pripomienok od používateľov z experimentálnej prevádzky. Následne bolo potrebné zamyslieť sa nad ďalšími možnými rozšíreniami a modifikáciami systému, pričom identifikované úpravy bolo potrebné zoradiť podľa priority ako aj časovej náročnosti. Vzhľadom na časovú náročnosť boli vypracované úlohy s najvyššou prioritou a niektoré sú určené pre zapracovanie v ďalšom pokračovaní projektu.

2 Úvod

Plánovanie a organizovanie času je v dnešnom zrýchlenom svete veľmi dôležité. V niektorých prípadoch dokonca nevyhnutné. Takýmto prípadom je aj vyučovanie a študovanie v prostredí na vysokej škole. Väčšinou sú v reálnom prostredí kladené najrôznejšie požiadavky na rozvrhy a plány. Od priestorových obmedzení, časových obmedzení až po rôzne požiadavky na vybavenie a pomôcky potrebné pri vyučovaní. Takisto treba naplno využívať potenciál prístupných zdrojov (miestností, prístrojov...).

Tvorba rozvrhu je proces, pri ktorom treba zorganizovať čas pomerne veľkého množstva ľudí – študentov, pedagógov, cvičiacich... Zaužívané postupy papierovej evidencie požiadaviek a podobne sa stávajú málo flexibilné a málo efektívne, najmä pri tak veľkom množstve študentov. Je potrebné zefektívniť proces tvorby rozvrhov od zberu požiadaviek, cez samotnú tvorbu rozvrhu až po overenie rozvrhu. Pre tento účel bol navrhnutý a implementovaný elektronický systém na podporu tvorby rozvrhov, ktorý sa primárne zameriava na zber požiadaviek na rozvrh. Na tomto systéme pracovali predchádzajúce tímy a podarilo sa im tento projekt dostať do fázy, kedy môžeme hlavnú vývojovú fázu považovať za ukončenú. Hlavnou úlohou nášho tímu je preto tento systém otestovať a vykonať ešte potrebné zmeny na to, aby bol tento systém prevádzkyschopný.

2.1 Účel a rozsah dokumentu

Tento dokument obsahuje analýzu procesu tvorby rozvrhov na vysokej škole a prehľad základných požiadaviek na systém pre podporu tvorby rozvrhov. Ďalej sa v dokumente nachádza analýza prevzatého systému od minuloročného tímu. Do tejto analýzy spadá zhodnotenie stavu systému (známe a novo objavené chyby v systéme), prehľad dátového modelu a architektúry celého systému a model implementovaných tried.

Tiež sú spomenuté zmeny, ktoré sme vykonali v systéme od jeho prevzatia (úprava dátového modelu, oprava najpodstatnejších chýb a pod.). V poslednej časti sa zaoberáme návrhom procesov a činností súvisiacich s nasadením a udržiavaním systému v prevádzke.

2.2 Skratky a použitá notácia

V Tab. 1 je uvedený zoznam skratiek používaných v tomto dokumente a ich význam.

Tab. 1 Použité skratky a ich význam

Skratka	Význam
AIS	Akademický informačný systém
LDAP	Lightweight Directory Access Protocol
APE	Administrátor prípravnej etapy

Skratka	Význam
STU	Slovenská technická univerzita
FIIT	Fakulta informatiky a informačných technológií
FEI	Fakulta elektrotechniky a informatiky
HTTP	Hypertext Transfer Protocol
PHP	Hypertext Preprocessor
HTML	Hypertext markup language
SQL	Standard Query Language
MVC	Model View Controller
CSS	Cascading Style Sheets

2.2.1 Notácia v modeloch

V diagramoch tried je použitá notácia UML 2.0.

Tab. 2 Notácia použitá v diagramoch tried

Diagram	Význam
<pre> classDiagram class AutoLoadable class Model Model -- > AutoLoadable </pre>	<p>Dedenie – trieda Model je rozšírením triedy AutoLoadable</p>
<pre> classDiagram class RequestHandler class Session RequestHandler --> Session : - session </pre>	<p>Asociácia – trieda RequestHandler môže volať metódy triedy Session (Session alebo referencia na Session je súčasťou triedy RequestHandler)</p>

2.3 Použitá literatúra

- [1] Tím BackSpace, Systém pre podporu tvorby rozvrhov, 2008/2009
- [2] Tím Fénix, Systém pre podporu tvorby rozvrhov, 2007/2008

3 Analýza problému tvorby rozvrhov

Proces tvorby rozvrhov môžeme rozdeliť do niekoľkých krokov. V prvom kroku je potrebné zozbierať všetky potrebné údaje pre vytvorenie rozvrhu. Následne je potrebné analyzovať jednotlivé zozbierané údaje a postupne ich zapracovávať do rozvrhu. Po zapracovaní všetkých údajov do rozvrhu vzniká základný prototyp rozvrhu. Konečným krokom tvorby je modifikovanie zostaveného rozvrhu do finálnej verzie a následne zverejnenie vytvoreného rozvrhu. Tu je potrebné analyzovať možnosti zverejňovania rozvrhov. Po nasadení rozvrhu do praxe je potrebné riešiť korekcie v rozvrhu, ktoré často krát vznikajú počas semestra. Po zapracovaní korekcií do semestrálneho rozvrhu je potrebné novú verziu rozvrhu znovu zverejniť.

3.1 Vstupné údaje pre proces tvorby rozvrhu

Pred odštartovaním procesu tvorby rozvrhu je nutné poznať všetky údaje, ktoré sa pri tvorbe využívajú. Údaje môžeme rozdeliť do týchto množín: množina miestností a ich kapacitné ohraničenie, vybavenie učebnými pomôckami (výpočtová a komunikačná technika, videoprojektory a pomôcky v laboratóriách, ...), množina otváraných predmetov a ich charakteristiky (maximálny počet študentov, počet dostupných pedagógov spolu s ich časovým harmonogramom, potrebné vybavenie, rozsah prednášok a rozsah cvičení, ...), množina študentov zapísaných na jednotlivé predmety a časový harmonogram výučby (štátne sviatky, dekanské voľno, študentská vedecká konferencia,...).

3.1.1 Množina predmetov a ich charakteristiky

Prvú skupinu tvoria všetky otvárané predmety a ich charakteristiky. Tieto informácie o predmete poskytujú garanti jednotlivých predmetov alebo nimi poverené osoby. Základné charakteristiky predmetov sú meno predmetu a jeho skratka, nominálny ročník, odbor, kapacitné ohraničenie predmetu (maximálny počet študentov na predmete, minimálny počet študentov potrebných pre otvorenie predmetu ...), vybavenie učební, prípadne požiadavka na špecializované učebne (laboratóriá, učebne s výpočtovou technikou, telocvičňa ,...). Okrem základných charakteristík predmetu možno zadefinovať rôzne prípady špeciálnych požiadaviek, ako napríklad:

- V niektorých týždňoch sa budú konať spoločné cvičenia vo veľkých miestnostiach
- Prednášky budú rozdelené do dvoch dní (napríklad 2 + 2 namiesto 4 súvislých hodín prednášok)
- Prednášky budú odprednášané za menší počet týždňov – v týchto týždňoch sa bude prednášať viac hodín, ako by sa malo, aby zostal celkový počet hodín zachovaný

- Rozdelenie prednášky na jednotlivé týždne a pridelenie prednášajúcich na každý týždeň v prípade, že prednášať chodia napríklad aj externí prednášatelia.
- Cvičenia sa nemusia konať každý týždeň, ale môžu byť napríklad každý druhý týždeň s dvakrát väčším počtom hodín – celkový počet hodín cvičení za semester musí zostať zachovaný
- Písomka (priebežný test) - je potrebné rezervovať miestnosť a dozor.

3.1.2 Množina pedagógov

Túto skupinu tvoria všetci pedagógovia, ktorí sa podieľajú na vyučovaní daných predmetov (môže ísť o garanta, prednášajúceho, vedúceho cvičení alebo cvičiaceho pedagóga). Ich počet je známy po registrácii predmetov študentmi, kedy jednotliví garanti vyčlenia pedagógov na výučbu daného predmetu. K charakteristikám tejto množiny pedagógov patrí akceptovanie časového rozvrhu pedagóga (napríklad práca mimo školy) a tiež sledovanie vyťaženia (aby nemal veľký počet vyučovacích hodín do týždňa). Pri tejto skupine musíme brať do úvahy aj absencie jednotlivých pedagógov. Táto situácia sa nedá predvídať ale treba ju počas semestra korektne zapracovať do rozvrhu a tak ju vyriešiť.

3.1.3 Množina študentov

Po registrácii predmetov jednotlivými študentmi získame prehľad o počte študentov na daný predmet. Musíme tiež brať do úvahy aj časové hľadisko študentov (aby nemal vyučujúce hodiny v preťažujúcom objeme resp. neprimerane „rozsekané“ od rána do večera), ďalej tiež špeciálne požiadavky študentov, ktoré väčšinou zadávajú zástupcovia študentov na základe sťažností alebo pripomienok.

3.1.4 Časový plán výučby

Časový harmonogram štúdia zostavuje študijné oddelenie fakulty. Je potrebné aby sa do rozvrhu zapracovali aj rôzne udalosti, ktoré môžu vzniknúť počas semestra. Tieto udalosti vznikajú napríklad pri dekanskom voľne alebo študentskej vedeckej konferencii, kde je potrebná ich korekcia buď v procese tvorby rozvrhu alebo počas semestra.

3.2 Zber údajov pre semestrálny rozvrh

Pri začatí procesu tvorby rozvrhu je potrebné aby rozvrhár mal prístup ku všetkým množinám údajov, z ktorých vychádza pri zostavovaní rozvrhu. Naša fakulta používa AIS, ktorý okrem iného umožňuje aj predregistráciu predmetov jednotlivými študentmi, ktorí si chcú v nasledujúcom akademickom roku zapísať tieto predmety. Spolu s týmito elektronickými predregistráciami je potrebné k povinne voliteľným predmetom zadať aj ich alternatívy (napr. v prípade prekročenia počtu študentov na daný predmet). Na základe predregistrácie je možné zistiť približný záujem o jednotlivé predmety, čiže je možné určiť, ktoré predmety budú nasledujúci akademický rok otvorené. Predregistrácia

slúži iba informatívne, nakoľko nie každý študent, ktorý urobil predregistráciu môže pokračovať v štúdiu. Predregistrácia sa v danom systéme nerieši nakoľko je v určitom rozsahu možná v AISe.

Ďalším krokom po predregistrácii je zápis študentov do jednotlivých ročníkov. Až vtedy je možné určiť presné počty študentov na predmety, nakoľko je zápis záväzný.

Zber údajov nezávisí len od študentov ale do zberu údajov zasahujú aj ďalšie typy používateľov:

- garant
- prednášajúci
- vedúci cvičení
- cvičiaci
- pracovníci ŠO

Jeden zamestnanec môže patriť do rôznych skupín používateľov zároveň. Napríklad môže byť garantom predmetu, prednášajúcim a cvičiacim pedagógom v rámci toho istého predmetu. Môže vzniknúť situácia kde je na daný predmet prihlásený veľký počet študentov, kedy sa daný predmet spravidla vyučuje vo viacerých inštanciách, kde jednotlivé prednášky prednášajú iní ľudia.

Základné požiadavky na jednotlivé predmety zadávajú garanti alebo nimi poverené osoby (prednášajúci, cvičiaci,...). čiže ide o definovanie hlavných vlastností a parametrov predmetu ako sú:

- číslo a názov predmetu
- skratka predmetu
- semester, v ktorom sa predmet vyučuje (LS alebo ZS)
- študijný program
- spôsob ukončenia predmetu (zs, z, kz)
- počet hodín prednášok - ak má semester 12 týždňov a každý týždeň by mala byť trojhodinová prednáška, tak počet hodín prednášok je 36.
- počet hodín cvičení - rovnaký výpočet ako pri počte hodín prednášok.
- počet prednášok - v prípade ak chce garant viac prednášok ako je štandardných 12, potom môže napríklad 12 trojhodinových prednášok nahradiť 18-timi dvojhodinovými prednáškami.

- počet cvičení - ak by mala byť každý týždeň jedna hodina cvičenia, tak garant môže zadať 6 cvičení, ktoré sa môžu vyučovať napr. každý druhý týždeň alebo druhú polovicu semestra.
- počet študentov predmetu - najskôr dajú z predregistrácie, potom podľa zápisu počtu ľudí, ktorý si predmet zapísali.
- kapacita cvičenia - maximálny možný počet študentov, ktorí sa môžu zúčastniť jedného termínu cvičenia. V tomto prípade sa jedná hlavne o cvičenia, ktoré sa vyučujú v počítačových učebniach alebo laboratóriách, kde je obmedzený počet miest.

Po definovaní základných vlastností predmetu je možné zdefinovať jednotlivé požiadavky na:

- miestnosti konania prednášky

prednášajúci pedagóg zadá tento typ požiadavky kde určí počet študentov zapísaných na daný predmet, vyberie vhodnú miestnosť s dostatočnou kapacitou

- miestnosti konania cvičení

cvičiaci predmetu má možnosť zdefinovať počet študentov v jednej skupine a následne vybrať kapacitne vhodnú miestnosť a špecifikovať typ miestnosti (napr. softvérové štúdio, laboratórium, PC učebňa,...)

- vybavenie

prednášajúci pedagóg má možnosť si vybrať vybavenie na daný predmet (dataprojektor, notebook, stoličky navyše)

- spôsob výučby

prednášajúci a cvičiaci pedagógovia môžu definovať časový harmonogram výučby predmetu v danom semestri, to znamená, že predmet môže mať pravidelný aj nepravidelný harmonogram výučby. Pravidelný spočíva v tom, že prednášky a cvičenia sú v priebehu semestra v rovnakom čase a v rovnakej dĺžke. Pri nepravidelnom harmonograme ako napríklad keď potrebuje prednášajúci odprednášať učivo čím skôr, tak sa prednášky konajú na začiatku semestra intenzívnejšie ako ku koncu semestra (napr. pri externom prednášajúcom pedagógovi zo vzdialenejších miest). Inou možnosťou je konanie prednášok a cvičení v každom druhom prípadne treťom týždni semestra.

3.3 Tvorba semestrálnych rozvrhov

Po zozbieraní všetkých potrebných množín údajov sa môže pristúpiť k samotnému procesu tvorby rozvrhu.

3.3.1 Postup tvorby

Najprv sa do rozvrhu pridávajú prednášky, tak aby spĺňali požiadavky na daný predmet (kapacita, vybavenie miestností,...) a tiež aby neboli v kolízií s inými predmetmi ako napríklad v jednej miestnosti dve úplne rozdielne prednášky v ten istý čas. Následne sa stanoví rozdelenie študentov do krúžkov. V prípade prvákov sa študenti do krúžkov zaraďujú direktívne algoritmicke, najčastejšie abecedne v definovaných n-členných „krúžkoch“, v prípade vyšších ročníkov sa jednotliví študenti sami registrujú na dané termíny cvičení. Následné sa jednotlivým cvičeniam pridelia pedagógovia tak, aby sme čo najlepšie spĺňali požiadavky na ich časový plán. Jednotliví pedagógovia sú v zozname s pridelenou prioritou, ktorá nám určuje ktorého pedagóga musíme uprednostniť pri tvorbe rozvrhu ako napríklad externých študentov ich časový plán kedy môžu sa zúčastňovať na výučbe.

Tiež musíme riešiť situáciu keď môže nastať kolízia jednotlivých predmetov pre daného študenta alebo pedagóga, kedy nemôžu mať súčasne viac vyučovacích hodín. Výnimku tvoria predmety, ktoré nemajú spoločných študentov alebo pedagógov kedy nenastáva kolízia v rozvrhu. Tiež pre nižšie ročníky je dobre rozdeliť jednotlivé vyučovacie hodiny rovnomerne na celý týždeň. V prípade vyšších ročníkov je vhodné rozložiť ich do niekoľkých dní (dvoch až troch). Je vhodné zabezpečiť optimálnosť rozvrhu pre každého, čo nemožno vždy dosiahnuť, preto treba aspoň minimalizovať možné odchýlky od optimálneho rozvrhu (minimalizovať počet voľných hodín v rozvrhu).

Po ukončení iterácie tvorby nám vznikne prototyp rozvrhu. Tvorca rozvrhu musí neustále kontrolovať, či daná iterácia rozvrhu nie je v rozpore s požiadavkami. V prípade, že je v rozpore, musí sa rozhodnúť ako rozpor vyriešiť. Pri riešení takýchto situácií musí uprednostňovať požiadavky s vyššou prioritou (napr. neprítomnosť pedagóga). Ak sa nedá situácia vyriešiť, je potrebné zmeniť rozvrh alebo jeho časť.

Opakovaním procesu tvorby predchádzajúcej iterácie vzniká ucelenejší pohľad na rozvrh kde sa zapracúvajú všetky požiadavky a výsledný rozvrh sa viac a viac podobá na finálny časový plán výučby.

3.4 Prepojenie s AIS

Prepojením s AIS sa zaoberali už tri predošlé tímy. Všetky dospeli k záveru, že keďže sa jedná o komerčný produkt, nie je možný prístup k zdrojovému kódu a databáze. Preto nie je možné priamo prepojiť systémy navzájom. Autori kvôli bezpečnosti systému neumožňujú prístup priamo k databáze. AIS má možnosť exportu niektorých množín údajov z databáz ako napríklad množinu predmetov, pedagógov, študentov,... Tiež umožňuje import údajov pre zverejňovanie už vytvorených rozvrhov. Prepojenie AIS a systému je nepriamo zabezpečené cez vopred dohodnutý súborový formát. Kde minuloročné tímy narazili na nedostatočnú spoluprácu s autormi AIS z ich strany.

Momentálne sú k dispozícii určité exporty z databázy AIS vo forme excelovských súborov.

Ak to zhrnieme, tak niektoré časti AIS sa stále prispôbujú potrebám STU. Požiadaviek na tvorcov AIS je však veľa, a tak sa zatiaľ k zmenám v systéme na tvorbu rozvrhov nijak nepokročilo. Komunikácia s tvorcami AIS je veľmi zdĺhavá a komplikovaná, a preto sa aj jednoduché problémy riešia veľmi dlho. Preto je potrebné riešiť samostatný systém na tvorbu rozvrhov s vlastnou databázou zamestnancov, miestností, vybavenia...

3.5 Hráči v systéme

Do systému bude mať prístup niekoľko rôznych typov používateľov. Jednotliví používatelia budú mať sprístupnenú len určitú funkčnosť systému. Je to dôležité z pohľadu zadávania požiadaviek a tvorby rozvrhu.

Používatelia systému:

Rozvrhár – používateľ (používatelia) systému, ktorý sa priamo podieľa na vytváraní rozvrhu na základe zozbieraných údajov (zoznam predmetov, miestností, požiadaviek, ...). Tomuto typu používateľa systém sprístupňuje rôznu funkčnosť, ktorá umožňuje zisťovať kolízie. Ďalej systém umožňuje prehľadné zobrazenie všetkých druhov požiadaviek. Ďalšou funkčnosťou systému je vytváranie viacerých verzií rozvrhu.

Administrátor – spravuje systém, databázu a používateľov.

Administrátor prípravnej etapy – tento typ používateľov má za úlohu naplnenie a aktualizáciu systému dôležitými údajmi. Tieto údaje tvoria: množina miestností, predmetov, pedagógov, časový harmonogram. Medzi týchto používateľov sa začleňujú aj pracovníci ŠO, ktorí môžu naplňať daný systém údajmi.

Garant – používateľ zadáva do systému požiadavky na predmet (čas prednášok alebo cvičení, priestory a vybavenie). Garant si môže prezerať vytvorené rozvrhy.

Pedagóg – používateľ systému, môže vkladať do systému požiadavky na prednášku a na svoj osobný časový plán. Ďalej má možnosť prezerať si rozvrh.

Študent – každý študent si môže pozrieť vytvorený rozvrh. Študent nijako nezasahuje do tvorby rozvrhu, ani do systému nevkladá žiadne údaje.

Je potrebné aby používatelia systému, ktorí majú prístup ku funkčnosť systému na vkladanie a modifikovanie údajov, mali do systému prístup pomocou autentifikácie. V prípade študentov nie je potrebné prihlasovanie sa do systému. Títo budú mať len možnosť prezerať si vytvorené rozvrhy.

4 Špecifikácia požiadaviek

V tejto kapitole sa zaoberáme špecifikáciou požiadaviek na systém. Opíšeme špecifikáciou funkcionálnych a nefunkcionálnych požiadaviek.

4.1 Nefunkcionálne požiadavky

Systém bude stavaný na tieto základné nefunkcionálne požiadavky:

- Počet používateľov bude rádovo v desiatkach (počas vytvárania rozvrhu) až stovkách (prezeranie publikovaného rozvrhu).
- Použijeme aplikačný server Apache s nainštalovaným modulom PHP.
- Použijeme databázový server PostgreSQL.
- Systém bude platformovo nezávislý ale v škole bude prevádzkovaný na operačnom systéme LINUX.

4.2 Funkcionálne požiadavky

Funkcionálne požiadavky systému sa budú odvíjať od príslušnej roly používateľa v systéme. Jednotliví používatelia budú mať sprístupnenú len im patriacu funkcionálnu. Ako už bolo opisované vyššie, reálni používatelia nemusia byť rozdelení len do týchto skupín. Môžu patriť zároveň do viacerých z nich, napríklad: používateľ je zároveň garantom predmetu a prednášajúcim, takže bude môcť využívať funkcie dostupné obom skupinám.

4.2.1 Garant Predmetu

- Keďže garant predmetu nemusí byť zároveň prednášajúcim, nemusí určovať miestnosť a ani potrebné vybavenie, pretože to záleží od požiadaviek prednášajúcich a vedúceho cvičení.
- Garant predmetu musí mať oprávnenie na zadanie názvu predmetu, ročníka, študijného programu a časového rozsahu predmetu v počte hodín prednášok a cvičení v jednom týždni
- Musí mať možnosť určiť prednášajúceho (prípadne prednášajúcich) a vedúceho cvičení.
- Musí mať možnosť meniť požiadavky (pokým sa nezačne samotný proces tvorby rozvrhu).

4.2.2 Prednášajúci

- Prednášajúci musí mať možnosť určiť reálny priebeh prednášok, to znamená, že musí mať možnosť zadať týždenný počet hodín prednášok, počet prednášok za týždeň, počet hodín jednej prednášky, počet študentov na prednáške, preferovanú miestnosť a potrebné vybavenie ktoré bude mať k dispozícii počas prednášky.

- Musí mať možnosť určiť tieto údaje pre každý týždeň zvlášť (podľa potreby).
- Musí mať možnosť podať špeciálne požiadavky (neštandardné).
- Musí mať možnosť zadať preferovaný čas výučby pre každý týždeň.
- Musí mať možnosť meniť požiadavky (pokým sa nezačne samotný proces tvorby rozvrhu).

4.2.3 Vedúci cvičení

- Vedúci cvičení musí mať možnosť určiť reálny priebeh cvičení, to znamená, že musí mať možnosť zadať týždenný počet hodín cvičení, počet cvičení za týždeň, počet hodín jedného cvičenia, počet študentov a preferované miestnosti pre jednotlivé skupiny na cvičeniach.
- Musí mať možnosť určiť tieto údaje pre každý týždeň zvlášť (podľa potreby).
- Musí mať možnosť zadať preferovaný čas výučby pre každý týždeň.
- Musí mať možnosť meniť požiadavky (pokým sa nezačne samotný proces tvorby rozvrhu).

4.2.4 Rozvrhár

- Musí mať možnosť zobrazenia prehľadu požiadaviek k jednotlivým predmetom.
- Musí mať možnosť prehľadného zobrazenia časových priorít pedagógov ku jednotlivým predmetom.
- Musí mať možnosť prezerat' konfliktnú tabuľku. Pod konfliktom sa myslia situácie, keď má predmet rovnakých prednášajúcich alebo rovnakých študentov ako iný predmet.

4.2.5 Administrátor

- Spravovanie používateľov: pridávanie, modifikovanie a odoberanie používateľov, nastavovanie prístupových oprávnení.

4.2.6 Autentifikácia

Používatelia sa budú môcť autentifikovať do systému pomocou prihlasovacích údajov zhodnými s prihlasovacími údajmi do AIS. Toto je zabezpečené školským LDAP serverom, kde sú uložené prihlasovacie údaje všetkých študentov a zamestnancov.

5 Analýza existujúceho riešenia

V tejto kapitole je opísaná analýza systému, ktorý bol vyvíjaný po minulé roky. Konkrétne sa tu nachádza zhrnutie, do akej miery boli splnené požiadavky na systém, ktoré sú dôležité na prototypové nasadenie a otestovanie systému. Ďalej sa tu nachádza diagram tried, ktorý bol vypracovaný kvôli lepšej prehľadnosti prepojenia existujúceho systému. Bolo potrebné analyzovať existujúce chyby v systéme, aby bolo možné podstatné nedostatky odstrániť a následne spustiť testovaciu verziu systému.

5.1 Kontrola požiadaviek na systém

Hlavná požiadavka na systém je, aby mohol byť nasadený na skúšobnú prevádzku v letnom semestri. Systém, ktorý sme prebrali po tíme BackSpace bol navrhnutý vo veľkej miere správne, ale museli sme spraviť revíziu a opravu dátového modelu. Síce je systém navrhnutý správne, ale obsahuje ešte stále veľmi veľa chýb. Od minulého roka ostalo neopravených a odhalených 17 chýb. Takže sme zvolili postup neustáleho testovania systému a odhaľovania nových chýb. Naša pozornosť sa zameriava na odstránenie tých chýb, kvôli ktorým nefunguje správne zber požiadaviek, nakoľko je toto primárny cieľ. Počas testovania sme odhalili už okolo 40 chýb, z ktorých minimálne $\frac{3}{4}$ je potrebné odstrániť, aby sa mohol systém nasadiť do testovacej prevádzky.

Jeden z najväčších nedostatkov systému bola chýbajúca validácia vstupných údajov od užívateľa. Táto validácia chýbala skoro v každej interakcii systému s užívateľom. A ak bola implementovaná, bola implementovaná neprehľadne a nedala sa použiť pre iný model. Nám sa podarilo tento podstatný nedostatok vyriešiť generickým riešením. Samotná validácia prebieha v metóde modelu „bind“, v ktorej sa kontroluje minimálna, maximálna dĺžka povinnosť poľa a prebiehajú tu aj všetky ostatné validácie. Dôležitým faktorom je, že validácia eliminuje aj prípad, že by sa nejaký užívateľ snažil o tzv. Code Injection. Týmto sa stáva systém podstatne bezpečnejší.

Ďalšia požiadavka na systém bola, aby sa ukladali časové pečiatky ak používateľ raz odošle svoje požiadavky a tým bol zabezpečený pre rozvrhára jasný dôkaz o tom odkiaľ tieto požiadavky získal a že si ich sám nevymyslel. Týmto bude umožnené aj to, že používateľ bude môcť povedať, že chce použiť rovnaké požiadavky na predmet také isté ako napr. minulý rok. Táto požiadavka sa musí riešiť na väčšine objektov takže toto budeme riešiť na dátovej úrovni. Budeme musieť vytvoriť tabuľku, ktorá bude reprezentovať verzie alebo časové pečiatky.

Pripomienkovanie je ďalšia požiadavka na systém. Ide o to, že ak sa používateľovi nebude niečo na systéme páčiť, aby mal možnosť vyjadriť svoju nespokojnosť a oznámiť svoju pripomienku vývojárom systému. Systém pre podporu rozvrhov sa už vyvíja viac ako 4 roky. To znamená, že vývojári sú každý rok iní. Od toho sa odvíja aj náš návrh na

riešenie tejto požiadavky. Vytvoríme štandardný formulár, do ktorého bude môcť používateľ napísať svoje pripomienky a celý tento záznam sa uloží do databázy. Vzhľadom na neustálu zmenu vývojárov bude vhodné ak si bude tieto pripomienky môcť prezerať administrátor systému.

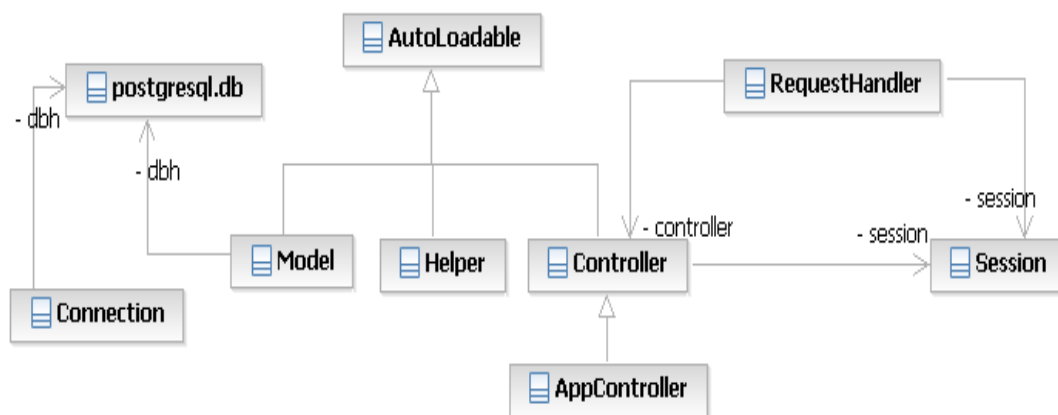
Implementácia on-line pomocníka je jedna z najnovších požiadaviek. Ide hlavne o to, aby aj počítačovo menej zdatní používatelia sa dokázali v systéme rýchlo zorientovať. Zatiaľ by sme chceli riešiť on-line pomocníka štýlom, že bude na každej stránke, na ktorej bude tento pomocník dostupný odkaz na dokument typu PDF, ktorý bude popisovať funkčnosť aktuálnej stránky. V budúcnosti by sme chceli implementovať on-line pomocníka priamo na stránke a nepoužívať žiadne externé dokumenty.

5.2 Architektúra existujúceho systému

Prevzatý systém je navrhnutý podľa architektúry klient-server. Klientskú časť systému tvorí webový prehliadač používateľa. Serverovú časť tvorí databázový systém PostgreSQL s databázovým serverom, HTTP server s PHP prostredím a samotná aplikácia, ktorá je implementovaná v jazyku PHP. Databázový server a HTTP server môžu byť nainštalované na jednom fyzickom počítači (prevádzkový server).

5.2.1 Model implementovaných tried

Triedy sú implementované a organizované podľa vzoru „model-view-controller“. Jadro systému tvoria triedy zobrazené na Obr. 1. Na diagrame sú znázornené základné vzťahy medzi týmito triedami.



Obr. 1 Diagram tried jadra systému

Aplikačne špecifické triedy je možné vďaka tejto architektúre pridávať do systému špecializáciou tried Model, View a Controller a ApplicationController. Napríklad špecializáciou triedy „Model“ sú vytvorené jednotlivé triedy reprezentujúce základné entity v systéme. Inštancie týchto tried poskytujú rozhranie na prácu s príslušnými databázovými entitami

(napr. trieda „Equipment“ zdedená od triedy „Model“ predstavuje model vybavenia, ktorých entity sa nachádzajú v databázovej tabuľke „vybavenie“).

5.3 Úpravy fyzického dátového modelu

Z dôvodu úpravy chýb bolo potrebné vykonať menšie zásahy do dátového modelu. Jednalo sa hlavne a podrobnejšie nadefinovanie referenčných obmedzení. Pre porovnanie dátový model zo školského roku 2008/2009 sa nachádza v prílohe B a nami vytvorený dátový model je v prílohe C.

5.3.1 Zmena č.1.

Pri oprave chyby č.5 – „Pri pridávaní a úprave používateľa nemá byť medzi skupinami "Všetci"“ sme sa rozhodli, že skupina „Všetci“ dostane id 0. Všetky ostatné skupiny budú mať id nezáporné, nenulové a tak môžeme zo zobrazení vylúčiť skupiny, ktorých používateľ je členom implicitne (majú záporné alebo nulové id).

Z dôvodu zmeny id sme upravili nasledovné referenčné obmedzenia:

- v tabuľke **menu**, stĺpcu **group_id** bolo zmenené obmedzenie na „ON UPDATE CASCADE ON DELETE CASCADE“
- v tabuľke **clenstvo**, stĺpcu **id_skupina** bolo zmenené obmedzenie na „ON UPDATE CASCADE ON DELETE CASCADE“
- v tabuľke **vyucuje_predmet**, stĺpcu **id_pedagog_typ** bolo zmenené obmedzenie na „ON UPDATE CASCADE ON DELETE CASCADE“

5.3.2 Zmena č.2.

Pri oprave chyby #45 – „Bug so stolickami pri pridávaní požiadaviek na prednasky“ bolo tiež potrebné zmeniť id stoličiek v tabuľke **vybavenie**. Z toho dôvodu boli taktiež doplnené referenčné obmedzenia:

- v tabuľke **poziadavka_vybavenie**, stĺpcu **id_vybavenie** bolo zmenené obmedzenie na „ON UPDATE CASCADE ON DELETE CASCADE“
- v tabuľke **vybavenie_miestnost**, stĺpcu **id_vybavenie** vybavenie bolo zmenené obmedzenie na „ON UPDATE CASCADE ON DELETE CASCADE“

5.3.3 Zmena č.3.

Pre prehľadnosť sme tabuľku **poziadavka** doplnili o nový stĺpec **cas_pridania**, ktorý obsahuje čas pridania požiadavky. Rozvrhár má teraz takto možnosť vidieť nielen obsah požiadaviek, ale aj kto požiadavky zadal a kedy. Zároveň sme zadávanie požiadaviek rozšírili tak, aby sa uchovávali všetky zadané požiadavky. Rozvrhár tak má možnosť prezrieť si históriu všetkých zadaných požiadaviek daného pedagóga na daný predmet.

5.4 Evidencia a oprava chýb existujúceho systému

V tejto časti sú opísané chyby systému, ktoré boli nájdené ale nevyriešené v minulom roku, ďalej chyby ktoré sme dodatočne našli my a tiež zoznam chýb, ktoré sa nám už podarilo opraviť. Z časového hľadiska sme nestíhali opraviť všetky vyskytujúce sa chyby, takže sme určili ich prioritu a následne opravili najnevyhnutnejšie.

5.4.1 Zaznamenané chyby

Niektoré chyby, ktoré nestihli byť opravené, boli nájdené tímom BackSpace v prechádzajúcom období. Ich zoznam je uvedený v Tab. 3.

Tab. 3 Chyby zaznamenané v školskom roku 2008/2009

Číslo	Opis chyby alebo nedostatku
1	Ošetriť nezadané heslo pri prihlasovaní
2	Chyba vo výbere údajov z DB, našlo sa 0
3	Pri zadávaní predmetu môžem vybrať len jeden študijný program
4	Pri úprave používateľa (administrátor) sa duplikujú jeho práva
5	Pri pridávaní a úprave používateľa nemá byť medzi skupinami "Všetci"
6	Ošetriť zadanie neplatného mena/hesla
7	Pridanie používateľa - môžem viac razí pridať rovnakú osobu
8	Pridávanie miestností - v prípade nevyplnenia alebo chybného vyplnenia položiek dát o tom informáciu
9	Pridávanie miestností - kontrolovať kladnosť kapacity, lebo ide zadávať aj záporné
10	Pridávanie predmetu - je možné pridať predmet bez zadania mena a kódu
11	Pridávanie vybavenia - možno pridať vybavenie bez zadania typu(mena)
12	Pridávanie vybavenia (asi aj inde) – nekontroluje sa maximálna dĺžka zadávaných parametrov
13	Správa používateľov - ak ma používateľ pridanú skupinu a ta sa odstráni tak sa to neuloží
14	Prezeranie požiadaviek na predmet
15	Pridanie požiadavky na predmet
16	Pridanie používateľa ktorý neexistuje v AISe => úspešné

5.4.2 Novoobjavené chyby a nedostatky

Počas analýzy a testovania existujúceho systému sme našli niekoľko doposiaľ neevidovaných chýb a nedostatkov. Keďže sme niektoré chyby opravovali, je možné že niektoré ďalšie vznikli aj našim zásahom. Uvádzame zoznam takýchto chýb v Tab. 4.

Tab. 4 Chyby a nedostatky systému nájdené v zimnom semestri školského roku 2009/2010

Číslo	Opis chyby alebo nedostatku
17	DB ignoruje 'NOT NULL' obmedzenia
18	Editácia používateľa - refactoring

Číslo	Opis chyby alebo nedostatku
19	Pri neautorizovanom prístupe chyba presmerovanie
20	Nezrovnalosť pri ukladaní požiadaviek pre cvičiaceho
21	Ape - pridanie miestnosti
22	Všetky moduly - kontrola zadaných údajov
23	All - podpora pripomienkovania v systéme
24	Pridanie používateľa - LDAP autocomplete problémy
25	Časové priority - chyba pri ukladaní alebo načítavaní
26	Equipment - code injection
27	Subject - code injection
28	Teacher - editácia požiadaviek
29	Scheduler - prezeranie požiadaviek na prednášky a cvičenia
30	Online help pre celý portál
31	Help - časové priority
32	Help - požiadavky pre cvičiaceho a pedagóga
33	Help - garant predmetu
34	Požiadavky na viac predmetov (nekorektné mazanie)
35	Scheduler - nedajú sa prepínať rozloženia pri požiadavkách na prednášky
36	Scheduler - nekorektné deaktivované položky pri prezeraní požiadaviek na cvičenia
37	Scheduler - zobrazenie časových priorít
38	Home - chybná správa ak nie som prihlásený
39	Menu - refactoring
40	Nekorektný timeout a následný logout ?
41	Prebytočný súbor postgresql.php
42	Duplicitné hodnoty v tabuľke vyucuje_predmet
43	Refactoring spoločného prehľadu požiadaviek u schedulera + pridanie dátumu do požiadavky
44	Pridávanie požiadaviek - označiť všetky týždne naraz
45	Chyba so stoličkami pri pridávaní požiadaviek na prednášky
46	Viaceré pohľady a rozloženia k nim
47	Požiadavky na prednášky
48	Požiadavky na cvičenia - chybné zobrazovanie miestností a iné.
49	Scheduler - prezeranie požiadaviek na prednášky a cvičenia - refactoring
50	Teacher - doplnenie zoznamu predmetov
51	Pract - pridávanie požiadaviek (chýba možnosť pridať komentár)
52	Administrator - pridať preberanie práv
53	Ape - priradenie výučby pedagógom
54	Scheduler - prezeranie požiadaviek (pridať nadpisy)
55	Garant - upraviť spôsob zobrazovania prideľovania predmetov
56	Požiadavky na prednasky - chybné ukladanie týždňov v rámci rozložení

Číslo	Opis chyby alebo nedostatku
57	logout - možno odhlásiť aj nikoho

5.4.3 Opravené chyby

Spustenie systému v testovacom režime je hlavná požiadavka v prvej časti tímového projektu a preto sme niektoré chyby alebo nedostatky už opravili. Zoznam opravených chýb a úprav je v Tab. 5.

Tab. 5 Opravené chyby v zimnom semestri školského roku 2009/2010

Číslo	Opis chyby alebo nedostatku
1	Ošetriť nezadané heslo pri prihlasovaní
4	Pri úprave používateľa (administrátor) sa duplikujú jeho práva
5	Pri pridávaní a úprave používateľa nemá byť medzi skupinami "Všetci"
6	Ošetriť zadanie neplatného mena/hesla
7	Pridanie používateľa - môžem viac razí pridať rovnakú osobu
8	Pridávanie miestností - v prípade nevyplnenia alebo chybného vyplnenia položiek dát o tom informáciu
9	Pridávanie miestností - kontrolovať kladnosť kapacity, lebo ide zadávať aj záporné
10	Pridávanie predmetu - je možné pridať predmet bez zadania mena a kódu
11	Pridávanie vybavenia - možno pridať vybavenie bez zadania typu(mena)
12	Pridávanie vybavenia (asi aj inde) – nekontroluje sa maximálna dĺžka zadávaných parametrov
14	Prezeranie požiadaviek na predmet
15	Pridanie požiadavky na predmet
16	Pridanie používateľa ktorý neexistuje v AISE => úspešné
18	Editácia používateľa - refactoring
21	Ape - pridanie miestnosti
22	Všetky moduly - kontrola zadaných údajov
25	Časové priority - chyba pri ukladaní alebo načítavaní
26	Equipment - code injection
27	Subject - code injection
29	Scheduler - prezeranie požiadaviek na prednášky a cvičenia
36	Scheduler - nekorektne deaktivované položky pri prezeraní požiadaviek na cvičenia
37	Scheduler - zobrazenie časových priorít
39	Menu - refactoring
42	Duplicitné hodnoty v tabuľke vyucuje_predmet
43	Refactoring spoločného prehľadu požiadaviek u schedulera + pridanie dátumu do požiadavky
45	Chyba so stoličkami pri pridávaní požiadaviek na prednášky

Číslo	Opis chyby alebo nedostatku
46	Viaceré pohľady a rozloženia k nim
51	Pract - pridávanie požiadaviek (chýba možnosť pridať komentár)
52	Administrátor - pridať preberanie práv
53	Ape - priradenie výučby pedagógom
54	Scheduler - prezeranie požiadaviek (pridať nadpisy)
55	Garant - upraviť spôsob zobrazovania pridelovania predmetov
56	Požiadavky na prednášky - chybné ukladanie týždňov v rámci rozložení
57	logout - možno odhlásiť aj nikoho

6 Testovacia prevádzka prototypu ZS 09/10

Cieľom vytvárania prototypu v zimnom semestri školského roku 2009/2010 bola testovacia prevádzka systému, ktorá mala pomôcť k získaniu spätnej väzby od používateľov vo forme pripomienok. Na základe tejto spätnej väzby sa odvíjala ďalšia práca na projekte. Po testovacej prevádzke bola potrebná analýza zozbieraných pripomienok a následné vyhodnotenie, ktorými sa budeme ďalej primárne zaoberať.

Keďže bol systém vyvíjaný už v školskom roku 2008/2009, našou primárnou úlohou bolo analyzovať aktuálnu funkčnosť systému, identifikovať existujúce chyby a opraviť, prípadne doplniť najakútnejšie chybné časti systému. Väčšina našej práce bola preto zameraná na hľadanie a odstraňovanie chýb v systéme, aby testovacia prevádzka prebehla čo najlepšie a teda aby sme mohli získať plnohodnotnú spätnú väzbu od používateľov. Testovacia prevádzka trvala od 4.12.2009 do 14.12.2009

Stručné zhrnutie funkcionality vytvoreného prototypu k dátumu testovacej verzie systému (4.12.2009):

- Priradenie garantov k jednotlivým predmetom
- Evidencia miestností (báza dát zhodná so systémom AIS)
- Evidencia predmetov (báza dát zhodná so systémom AIS)
- Evidencia vybavenia potrebného k výučbe
- Zadanie požiadaviek na predmet garantom predmetu (skratka predmetu, výber prednášajúceho a vedúceho cvičení, rozsah prednášok a rozsah cvičení)
- Zadanie požiadaviek na predmet prednášajúcim predmetu (počet prednášok, časový rozsah prednášok, kapacita miestnosti, počet študentov, požadované vybavenie a iné špecifiká vyskytujúce sa pri zadávaní požiadaviek na prednášky)
- Zadanie požiadaviek na predmet cvičiacim predmetu (počet cvičení, časový rozsah cvičení, kapacita miestnosti, počet študentov v skupine, počet cvičení súčasne a iné špecifiká vyskytujúce sa pri zadávaní požiadaviek na cvičenia)
- Zobrazenie zadaných požiadaviek tvorcovi rozvrhu
- Zadávanie časových priorít prednášajúcimi a cvičiacimi
- Zobrazenie časových priorít prednášajúcich a cvičiacich tvorcovi rozvrhu
- Zadávanie pripomienok k systému používateľmi (dôležité pre ďalší rozvoj systému)
- Prezeranie zadaných požiadaviek na systém administrátormi
- Základné funkcie systému ako prihlasovanie (kontrola cez systém ldap), odhlasovanie a správa používateľov
- On-line pomocník

6.1 Analýza pripomienok z testovacej prevádzky

Počas testovacej prevádzky vložilo požiadavky na predmet 5 vyučujúcich a 4 z nich napísali pripomienky k systém. Celkovo bolo zaznamenaných 15 pripomienok a rámcovo sú opísané v Tab. 6.

Tab. 6 Pripomienky k systému z testovacej prevádzky v ZS 2009/2010

Číslo	Pripomienka
1	Rozšírenie dĺžky pripomienky na viac ako 255 znakov
2	Automatické preberanie počtu študentov zo systému AIS (napríklad)
3	Požiadavky na cvičenia by mali môcť zadávať aj prednášajúci a garant predmetu
4	Možnosť priradiť dvoch rôznych vyučujúcich ako prednášajúci aj cvičiaci
5	Preberanie požiadaviek ako aj zadávanie rozsahu predmetu, prednášajúceho a cvičiaceho z minulých rokov
6	Usporiadať mená v zoznamoch podľa priezviska
7	Možnosť nezadať prednášajúceho a cvičiaceho ak v predmete nie sú
8	Zväčšiť okno na komentovanie požiadaviek na prednášky aj cvičenia a zmeniť obmedzenie 255 znakov
9	Cvičiaci by nemal mať možnosť zadávať požiadavky na rozvrh (mala by to za celý predmet zadávať jedna osoba)
10	Doplniť možnosť pridania poznámky (nie špeciálne požiadavky) k požiadavke
11	Doplniť komentovanie časových priorít

Pripomienku číslo 1 sme museli vyriešiť hneď po zistení tohto nedostatku, pretože išlo o závažné obmedzenie zadávania pripomienok a chceli sme získať čo najlepšiu spätnú väzbu. Preto sme tento bod vyriešili za behu testovacej prevádzky a zrušili sme obmedzenie dĺžky pripomienok.

Všetky požiadavky vo svojom pôvodnom tvare sú uvedené v PRÍLOHA D.

7 Implementácia

V tejto kapitole sú opísané implementačné práce tímu prevažne počas letného semestra v školskom roku 2009/2010. Išlo predovšetkým o zapracovanie pripomienok z testovacej prevádzky, usporiadanie ďalších možných rozšírení podľa priority a ich následnú implementáciu, a taktiež o úpravu systému, aby sa dal v budúcnosti jednoduchšie upravovať a rozširovať.

7.1 Zapracovanie pripomienok k systému zo zimného semestra 09/10

Väčšinu pripomienok sme zapracovali, avšak niektoré sme odsunuli alebo sme zhodnotili, že ich zapracovanie nie je vhodné.

7.1.1 Zapracované pripomienky

Pripomienky, ktoré sme zapracovali do systému sú uvedené v Tab. 7.

Tab. 7 Zoznam zapracovaných pripomienok z testovacej prevádzky v ZS 2009/2010

Číslo	Zapracovaná pripomienka
1	Rozšírenie dĺžky pripomienky na viac ako 255 znakov
2	Automatické preberanie počtu študentov zo systému AIS (napríklad)
3	Požiadavky na cvičenia by mali môcť zadávať aj prednášajúci a garant predmetu
5	Preberanie požiadaviek ako aj zadávanie rozsahu predmetu, prednášajúceho a cvičiaceho z minulých rokov
6	Usporiadať mená v zoznamoch podľa priezviska
7	Možnosť nezadať prednášajúceho a cvičiaceho ak v predmete nie sú
8	Zväčšiť okno na komentovanie požiadaviek na prednášky aj cvičenia a zmeniť obmedzenie 255 znakov
10	Doplniť možnosť pridania poznámky (nie špeciálne požiadavky) k požiadavke
11	Doplniť komentovanie časových priorít

Niektoré pripomienky sa týkali úprav, ktoré sme sami zamýšľali zapracovať do systému a teda bližšie opísanie vypracovanie je možné nájsť v podkapitole 7.2.

7.1.2 Nezpracované pripomienky

Nezpracovali sme pripomienky uvedené v Tab. 8.

Tab. 8 Zoznam nezpracovaných pripomienok z testovacej prevádzky v ZS 2009/2010

Číslo	Nezpracovaná pripomienka
4	Možnosť priradiť dvoch rôznych vyučujúcich ako prednášajúci aj cvičiaci
9	Cvičiaci by nemal mať možnosť zadávať požiadavky na rozvrh (mala by to za celý predmet zadávať jedna osoba)

Pripomienku číslo 4 sme odložili, pretože je vhodné priamo implementovať priradovanie viacerých cvičiacich (nie len vedúceho) a teda vznik ďalšej role. Táto pripomienka je úzko spätá s pripomienkou číslo 3, ktorá zabezpečuje, že prednášajúci, garant aj vedúci cvičení môžu upravovať požiadavky na cvičenia (teda traja rôzni ľudia) a požiadavky na prednášky môže upravovať garant aj prednášajúci (dvaja rôzni ľudia). Vypracovanie tejto pripomienky môže byť uskutočnené v budúcnosti (ak bude naozaj potrebné) a bolo by zároveň vhodné zvážiť refaktoring rolí a právomocí, ako je uvedené v PRÍLOHA E.

Pripomienku číslo 9 sme zamietli po konzultácii s vedúcim tímu, pretože po zapracovaní pripomienky číslo 3 môže všetky požiadavky upravovať garant (za celý predmet) ale je potrebné, aby mohli plne upravovať požiadavky aj konkrétny prednášajúci a vedúci cvičení.

7.2 Rozšírenie a úprava funkcionality systému

V tejto podkapitole sú uvedené a opísané najvýznamnejšie úpravy a rozšírenia systému v letnom semestri školského roku 2009/2010. Spravili sme viacero menších zmien a doplnení systému, ktoré je možné nájsť v podkapitole 7.3 ako vypracované úlohy.

Vytvorenie metapožiadaviek

Požiadavky sa ukladali tak, že pre každé rozloženie v požiadavke vznikala celá samostatná požiadavka a vznikali tak redundantné dáta, ktorých bolo zbytočne veľa a spomaľovalo sa načítavanie. Údaje, ktoré boli pre všetky rozloženia rovnaké sme zabalili do takzvaných metapožiadaviek, ktoré sú hierarchicky nad požiadavkami.

Vytvorenie semestrov

Táto funkcionality bola kľúčová, pretože požiadavky musia byť priradené k predmetom v určitých semestroch a školských rokoch, aby sa dali od seba odlíšiť a aby bolo možné preberať požiadavky z minulých rokov.

Preberanie požiadaviek z minulých rokov

Pravdepodobne najdôležitejšie rozšírenie systému, pretože výrazne uľahčí zadávanie požiadaviek. Vyučujúci môžu jednoducho vybrať požiadavky, ktoré boli zadané v minulých rokoch a podľa toho sa im vyplní aj nová požiadavka. Vďaka tomuto sa pri vo

väčšine prípadov stane zadávanie požiadaviek jednoduchým vybratím predmetu, kliknutím na prebratie a následné uloženie, pretože štruktúra vyučovania väčšiny predmetov zostáva rovnaká. Samozrejme úlohou vyučujúcich zostáva kontrola prebratej požiadavky, či všetko sedí (napríklad či sa dramaticky nezmenil počet študentov alebo či nechce vyučujúci upraviť štruktúru výučby). Pre uľahčenie prvých prevádzok systému sme vlastnoručne vyplnili požiadavky za zimný a letný semester v školskom roku 2009/2010 podľa papierovej predlohy.

Komentovanie požiadaviek zadávateľom a tvorcom rozvrhu

Taktiež dôležitá funkcionality predovšetkým pre tvorcu rozvrhu, aby mohol okomentovať zadané požiadavky v prípade, že niečo neseďí, je potrebná podrobnejšia špecifikácia alebo sa jednoducho potrebuje so zadávateľom dohodnúť na kompromisnom riešení. Zadávateľ požiadavky (prípadne každý kto má práva zapisovať danú požiadavku) môže okomentovať naspäť a takto vzniká diskusia k jednotlivým požiadavkám. Komentáre sa tiež odosielajú na emaily zainteresovaných.

Rozvrhové akcie

Tvorca rozvrhu môže definovať časové rozmedzie kedy budú trvať rôzne rozvrhové akcie. Za súčasného stavu ide iba o zber požiadaviek na predmety avšak je možné pridať v budúcnosti ďalšie akcie. Vyučujúci po prihlásení na úvodnej stránke môžu vidieť od kedy do kedy je možné zadávať požiadavky na predmety a môžu si tak na to vyhradiť čas. Toto „obmedzenie“ vyučujúcich je potrebné, aby boli zadané požiadavky do určitého termínu (keďže po ukončení zberu požiadaviek ich už nebudú môcť upravovať).

Profil

Každý používateľ systému má svoj profil, v ktorom je zatiaľ len nastavenie emailu a potvrdenie či chce používateľ dostávať notifikáciu o svojich zásahoch do systému. Tento profil je samozrejme rozšíriteľný do budúcnosti.

Emailová notifikácia

Ďalšie významné rozšírenie, aby používatelia nemuseli zbytočne často navštevovať systém a hľadať, či im niekto niečo nepriradil alebo neokomentoval. Maily sa odosielajú:

- Pri určení časového rozmedzia zberu požiadaviek tvorcom rozvrhu (mail všetkým vyučujúcim = garantom a nastaveným prednášajúcim a vedúcim cvičení). Odošle sa v akom predmete má dotyčný používateľ akú rolu.
- Pri priradení prednášajúceho a cvičiaceho sa im odošle toto priradenie.
- Pri zmene požiadavky sa odošle informácia, že daná požiadavka bola zmenená všetkým, ktorí majú práva na úpravu požiadavky.

- Pri okomentovaní požiadavky (či už tvorcom rozvrhu alebo vyučujúcim, ktorý má práva) sa odošle tento komentár a odkaz na celú diskusiu všetkým, ktorí majú práva na úpravu požiadavky a tvorcovi rozvrhov.

Taktiež je možné pridať ďalšie notifikačné akcie.

Kolaborácia pri vytváraní požiadaviek

V pripomienkach sa často opakovalo, aby mohli upravovať požiadavky aj vyučujúci s „nadradenými“ rolami. Takže požiadavky na cvičenia môžu teraz upravovať okrem vedúcich cvičení aj prednášajúci aj garant a požiadavky na prednášky môže upravovať okrem prednášajúceho aj garant.

Kvôli časovej tiesni sme toto implementovali tak, že sa pri kontrole, či môže daný používateľ editovať požiadavku, sleduje ID v tabuľke „skupina“ a ak je menšie ako ID role používateľa, ktorý ju môže editovať normálne, tak je mu editácia povolená. Toto funguje vďaka tomu, že ID role cvičiaci je väčšie ako ID prednášajúceho a to je väčšie ako ID garanta. Toto riešenie môže spôsobiť problémy, ak sa bude tabuľka „skupina“ v budúcnosti meniť alebo rozširovať o ďalšie role. Preto je vhodné toto v budúcnosti prerobiť ako je navrhnuté v PRÍLOHA E.

Taktiež sme vyriešili „zamykanie“ (lockovanie) požiadaviek, aby nemohli upravovať viacerí naraz a aby nevznikali kolízie.

Kalendár

Každý používateľ má prístupný kalendár, aby si mohol pozrieť pri zadávaní požiadaviek dátumy v jednotlivých týždňoch a taktiež tu sú zaznamenané rozvrhové akcie.

Filtrovanie záznamov

V systéme sa na rôznych miestach nachádzajú zoznamy, ktoré sú miestami značne rozsiahle a preto sme implementovali filter pomocou ktorého je možné rýchlo nájsť hľadaný záznam. Taktiež je dostupné usporadúvanie podľa rôznych kritérií. V rámci usporadúvania záznamov sme taktiež upravili štruktúru uchovávaní mien používateľov, aby sa dali zoraďovať podľa priezviska (oddelenie titulov a mena). V minuloročnej verzii bolo meno používateľa aj s titulmi a teda zoradenie bolo na základe titulov, čo bolo značne nepraktické.

Komentáre a refaktoring

Nejde priamo o implementačné práce, ale považovali sme za dôležité okomentovanie kódu ako aj jeho refaktoring, pretože bol miestami značne nezrozumiteľný a pri projekte, ktorý sa má posúvať medzi tímami z roka na rok je dôležité, aby bol kód udržiavaný. V opačnom prípade by sa mohlo stať, že by časom projekt musel byť vytvorený od znova, lebo by práca na úpravách a rozširovaní exponenciálne rástla.

Export požiadaviek zo systému

Aby sa dali zadané požiadavky ukladať aj mimo systému, tak sme pridali export požiadaviek do formátu CVS a DOC. Exportovať požiadavky môže garant predmetu alebo tvorca rozvrhu. Tvorca rozvrhu môže exportovať požiadavky po jednej vo formáte DOC alebo môže exportovať všetky požiadavky zadané v aktívnom semestri naraz do súboru CVS alebo DOC.

7.3 Konkrétne vypracované úlohy

V zimnom semestri sme na evidenciu úloh používali systém Assembla, v rámci pokračovania po minuloročnom tíme. Pred začatím letného semestra sme zvažili iné možnosti a najvhodnejší sa nám zdal systém Redmine, pretože je uzavretý a evidenciu úloh má viac prepracovanú (to bola naša primárna požiadavka).

Nasledujú vypracované implementačné úlohy za letný semester evidované v systéme Redmine v Tab. 9.

Tab. 9 Úlohy vypracované v letnom semestri školského roku 2009/2010

Číslo úlohy	Vypracovaná úloha
1	naviazať požiadavky na semester
2	pridať podporu filtrovania podľa semestra
3	pridať možnosť vytvoriť nový semester
4	upraviť polia pre komentáre
5	zrušiť obmedzenia na komentáre
6	usporiadať položky v menu
7	upraviť zobrazovanie pedagógov u garanta
8	automatické dopočítanie počtu študentov
9	zobrazovať počty študentov podľa odboru
10	upraviť naviazanie študentov na študijný odbor
11	rozseparovať tabuľku zapisany_predmet
12	aktualizovať dátový model
13	cvičenia - pridať možnosť zadať požiadavku na vybavenie
14	otestovať použitie knihnice tablesorter
15	ape - pridať možnosť prebrať predmety z minulého roku
16	teacher - pridať možnosť prebrať požiadavku z minulého roku
17	pract - pridať možnosť prebrať požiadavku z minulého roku
18	garant - pridať možnosť prebrať požiadavku z minulého roku
19	auth/login - nekorektné správanie
20	all - priorities - chyba ak nie som prihlásený
21	analyzovať kompatibilitu s PHP 5.3, 6.0
22	rozvrhár - cvičenie - refactoring
23	rozvrhár - prednášky - refactoring

Číslo úlohy	Vypracovaná úloha
24	admin - správa používateľov - upraviť pridávanie používateľov pre novú schému používateľov
25	pridať select s výberom semestra do menu
26	ape - filtrovanie podľa semestra
27	core - updater - upraviť poradie vykonávania skriptov
28	vytvoriť skript, ktorý zabalí zadané požiadavky do metapožiadaviek
29	zabalíť zadávané požiadavky do meta-požiadavok pre cvičiaceho
30	zabalíť zadávané požiadavky do meta-požiadavok pre prednášajúceho
31	tablesorter - javascript, ktorý vytvorí pager container
32	vytvoriť JS, ktorý bude vytvárať sortované tabuľky
33	admin - správa používateľov - paging+sorting tabuľky
34	ape - priradenie výučby pedagógom - paging+sorting tabuľky
35	garant - zrušiť povinnosť zadávať prednášajúceho a cvičiaceho
36	rozvrhár - sled metapožiadavok
37	ape - nejde pridávať predmety
38	paging+sorting tabuliek
39	tablesorter - automatické generovanie pageru
40	odstrániť zbytočné naviazania na semester
41	rozvrhár - zobrazovať počty študentov podľa štud. programov
42	all, rozvrhár - pridať podporu filtrovania podľa semestra
43	rozvrhár - nekorektné zobrazovanie mien v prioritách
44	pridať mailovu notifikáciu
45	Kolaboracia požiadavky
46	pridať podporu komentárov #2 - vytvorenie SQL update modelu
47	pridať podporu komentárov #3 - úprava pohľadov prednášajúceho a cvičiaceho
48	pridať podporu komentárov #4 - úprava pohľadov rozvrhara
49	Vymeniť maily v pedagógoch
50	redirect po prihlásení na konkrétnu url
51	evidencia stavov pripomienok
52	ape - pridať podporu na blokovanie preberania požiadaviek
53	riešenie kolízií pri kolaborácii pri zadávaní požiadaviek
54	Vypracovať a navrhnuť typy notifikačných zoznamov
55	pridať možnosť používateľovi nastaviť si mail kam mu bude chodiť notifikácia
56	návrat späť po odhlásení
57	pridať semestrový kalendár (organizáciu semestra)
58	filtrovanie grupovanie v tabuľkách
59	ape - pridať podporu rozvrhových akcií
60	nejde zadávať požiadavky v IE8
61	pract, teacher - nefunguje validácia vstupných dát
62	teacher, pract, scheduler - skrývanie počtu študentov

Číslo úlohy	Vypracovaná úloha
63	core - pridať controller na kontrolu rozvrhových akcií
64	pract - upraviť pohľady podľa rozvrhových akcií
65	teacher - upraviť pohľady podľa rozvrhových akcií
66	all - upraviť pohľady podľa rozvrhových akcií
67	all - pridať možnosť prebrať časové priority z minulého roku
68	doplniť rozposielanie notifikačných mailov
69	Pridať do profilu checkbox či posielat' mnou spravené zmeny.
70	Podpora sql pre notifikácie
71	Pri ukladaní usera nastaviť mail na prázdny
72	teacher, pract - chybné správanie komentárov
73	rozšírenie db.php o automatické konverzie dátových typov
74	doplniť scroll na komentáre cez named href
75	teacher, pract - Oops pri špecifických prípadoch
76	aktualizovať dátový model
77	pridať editáciu semestrov
78	nekorektné zobrazovanie chybových hlášok
79	doplnenie notifikácií pripomienok

8 Testovacia prevádzka prototypu LS 09/10

V letnom semestri sme testovaciú prevádzku vykonali sami pomocou akceptačných testov.

Pre jednotlivé skupiny používateľov boli testované nasledujúce funkcionality:

1. Všetci (all)
 - Pripomienky
 - ✓ kontrola vloženia pripomienky s prázdnyim textom
 - ✓ kontrola vloženia pripomienky s neurčeným typom
 - ✓ vloženie korektne zadanej pripomienky
 - Osobné časové priority
 - ✓ zadanie nových časových priorít všetkých typov a uloženie
 - ✓ zmena časových priorít
2. Administrátor prípravnej etapy (APE)
 - Evidencia miestností
 - ✓ vyhľadanie miestnosti
 - ✓ kontrola zadania názvu existujúcej miestnosti
 - ✓ kontrola zadania nesprávnej kapacity miestnosti
 - ✓ uloženie korektne upravenej miestnosti
 - Priradenie výučby pedagógom
 - ✓ vybratie a uloženie garanta k predmetu
 - ✓ zrušenie garanta pre predmet
 - Správa používateľov
 - ✓ vyhľadanie používateľa
 - ✓ zmena privilégiií používateľa
 - ✓ kontrola zadania prázdneho/neexistujúceho mena používateľa
 - ✓ pridanie korektne zadaného používateľa
 - Pripomienky k systému
 - ✓ úprava existujúcej pripomienky
 - ✓ kontrola prázdneho textu pripomienky
 - ✓ vymazanie pripomienky
 - Správa semestrov
 - ✓ pridanie nového semestra
 - Evidencia predmetov
 - ✓ pridanie nového predmetu
 - ✓ odobratie existujúceho predmetu
 - ✓ prebratie predmetov z minulého roku
 - ✓ blokovanie predmetu

3. Garant

- Požiadavky na predmet
 - ✓ prebratie požiadavky na predmet z minulého roku
 - ✓ uloženie prebratej požiadavky
 - ✓ úprava zadanej požiadavky
 - ✓ kontrola skratky predmetu na prázdny reťazec
 - ✓ kontrola rozsahu prednášky na nulu
 - ✓ kontrola rozsahu prednášky na záporné číslo
 - ✓ kontrola rozsahu prednášky na nečíselnú hodnotu
 - ✓ nastavenie, že prednášajúci nie je
 - ✓ nastavenie konkrétneho prednášajúceho
 - ✓ uloženie požiadavky

4. Prednášajúci

- Požiadavky na predmet
 - ✓ prebratie požiadavky na predmet z minulého roku
 - ✓ vymazanie komentára k prebratej požiadavke
 - ✓ uloženie prebratej požiadavky
 - ✓ úprava zadanej požiadavky
 - ✓ kontrola rozsahu prednášky na nulu
 - ✓ kontrola počtu dodatočných stoličiek na záporné číslo
 - ✓ pridanie ďalšieho rozloženia prednášok
 - ✓ nastavenie rozloženia prednášky
 - ✓ nastavenie preferovanej miestnosti na prednášky
 - ✓ uloženie požiadavky

5. Cvičiaci

- Požiadavky na predmet
 - ✓ prebratie požiadavky na predmet z minulého roku
 - ✓ uloženie prebratej požiadavky
 - ✓ úprava zadanej požiadavky
 - ✓ kontrola rozsahu cvičenia na nulu
 - ✓ kontrola maximálneho počtu cvičení na nulu
 - ✓ kontrola počtu študentov v skupine na nulu
 - ✓ pridanie ďalšieho rozloženia cvičení
 - ✓ nastavenie rozloženia cvičení
 - ✓ nastavenie počtu študentov v skupine na korektné číslo
 - ✓ nastavenie preferovanej miestnosti na cvičenia
 - ✓ uloženie požiadavky

Všetky akceptačné testy prebehli v poriadku podľa očakávania a ich podrobný opis je uvedený v prílohe PRÍLOHA F.

9 Návrh rozšírení a úprav existujúceho systému

Táto kapitola slúži na opis úloh, nad ktorými je vhodné sa zamyslieť pri ďalšej práci na projekte.

9.1 Návrh ďalšej práce na projekte

Analyzovali sme rôzne nápady na úpravu a rozšírenie systému. Väčšinu sme implementovali, avšak nebol čas na dokončenie všetkého čo nás napadlo a preto nasleduje zoznam úloh, nad ktorými by sa budúcoročný tím mal minimálne zamyslieť.

9.1.1 Rozšírenie funkcionality systému

- Zobrazovanie vytvorených rozvrhov v systéme pre každého používateľa (napríklad vytiahnuť z AISu ak sa dá)
- Pripomienkovanie rozvrhov pedagógmi, aby mohli dať vedieť ak im niečo prekáža na už vytvorenom rozvrhu (fáza po zbere požiadaviek)
- Registrácia predmetov na ďalší semester študentmi, čiže aj pridanie role študent do systému (tak ako je to teraz v AISe nech je všetko súvisiace s vytváraním rozvrhov centralizované)
- Pripomienkovanie konfliktov v rozvrhoch študentmi (ak sa niekomu kryje prednáška a cvičenie)
- Pridať unit-testy (Test Driven Development) prípadne testovanie správania (Behavioral Driven Development)
- Možnosť priradiť dvoch rôznych vyučujúcich ako prednášajúci aj cvičiaci – bola to pripomienka k systému v zimnom semestri. Vhodné tiež rovno pridať rolu cvičiaci (teraz je len vedúci cvičení) aby mohli byť určení viacerí cvičiaci, ktorých by v prípade konfliktov mohol tvorca rozvrhu kontaktovať (ak by nebol vedúci cvičení dostupný)
- Úprava rolí používateľov a ich privilégii. Kvôli časovej tiesni sme implementovali kolaboráciu viacerých používateľov na jednej požiadavke tak, že sa pri kontrole, či môže daný používateľ editovať požiadavku, sleduje ID v tabuľke „skupina“ a ak je menšie ako ID role používateľa, ktorý ju môže editovať normálne, tak je mu editácia povolená. Toto funguje vďaka tomu, že ID role cvičiaci je väčšie ako ID prednášajúceho a to je väčšie ako ID garanta. Toto riešenie môže spôsobiť problémy, ak sa bude tabuľka „skupina“ v budúcnosti meniť alebo rozširovať o ďalšie role. Preto je vhodné toto v budúcnosti prerobiť ako je navrhnuté v PRÍLOHA E.

9.1.2 Rozšírenie a úprava funkcionality zadávania požiadaviek

Pri vkladaní požiadaviek na predmety kvôli zjednodušeniu preberania v budúcich rokoch sme identifikovali ďalšie úlohy, ktoré je možné zapracovať do systému za účelom zvýšenia funkcionality systému. Úlohy sú zoradené podľa priority.

Vysoká priorita

- do kalendára pridať udalosti, týkajúce sa harmonogramu štúdia (začiatok či koniec semestra, plánované rozvrhové akcie)
- pri požiadavkách na predmety upraviť možnosť vyberať viaceré miestnosti súčasne s rôznych typov učební
- zobrazíť zoznam s obsadením vedenia fakulty, údaje o tom kedy majú jednotlivé senáty zasadania – tieto údaje by sa mali v systéme poskytnúť tvorcovi rozvrhu (buď v kalendári, alebo len „natvrdo“)
- pri miestnostiach umožniť vybrať špecifickú miestnosť napr. biela keramická tabuľa, ale nevedia, ktorá miestnosť to je
- rozlíšiť pripomienku k celej požiadavke a k rozloženiu
- pri požiadavkách na cvičenia chýba možnosť navýšenia počtu študentov
- garant nastavuje týždenný rozsah predmetu, tieto údaje by sa mohli použiť ako defaultne pri zadávaní požiadaviek, napr. ak rozsah prednášky bol 3, tak automaticky doplní 3, v prípade cvičení aj je napr. rozsah 3, tak tiež doplní 3
- pridať checkbox, ktorý umožní rozvrhárovi vybrať ľubovoľnú nešpecifikovanú miestnosť pre predmet, len na základe postačujúcej kapacity

Stredná priorita

- pridať možnosť vyjadrenia sa študijného parlamentu alebo katedry k požiadavkám
- pripraviť systém na kooperáciu s AIS alebo s existujúcim softvérom na tvorbu rozvrhov. Možnosť použiť štandardizovaný a všeobecne rozšírený formát XML na komunikáciu medzi systémami
- možnosť blokovania jednotlivých miestností, ktoré systém neumožní používateľovi vybrať do požiadavky (napríklad pri miestnosti v ktorej z nejakého dôvodu nemôže prebiehať výučba)
- zapracovať príznaky, ktoré budú špecifikovať stav jednotlivých požiadaviek. Napríklad príznak, ktorý bude určovať, či požiadavka bola zmenená alebo zapracovaná do rozvrhu
- umožniť zadávať počet študentov pomocou rozsahu napr. min. 10 a max 14
- implementovať algoritmy na pomoc automatizovanej tvorby rozvrhov

Nízka priorita

- možnosť demonštrácie zadávania údajov do systému (napr. pomocou videa)
- zapracovať do systému predregistráciu predmetov študentmi podobne ako je to v systéme AIS
- umožniť zverejňovanie vytvorených rozvrhov

- pokročilé exportovanie údajov do rôznych formátov

9.1.3 „TODO“ hlášky v zdrojovom kóde

Počas implementácie systému boli identifikované pripomienky, ktoré sú opísané v TODO poliach na príslušných miestach v zdrojovom kóde.

Zoznam identifikovaných pripomienok, ktoré by bolo vhodné analyzovať a prípadne zapracovať do systému:

- v súbore *help.php* definíciu konštanty `DEFAULT_HELP` - "*help/index.php*" presunúť do *config.php*
- v súbore *menu_helper.php* navrhnuť možnosť odseparovať kód do samostatného súboru a inklúdnúť ho ako šablónu
- v *model* súbore *teacher_requirements.php* by šlo presunúť do *courses* a v tomto súbore nechať len wrapper čo predá *typ_poziadavky-rolu*
- v *model* súbore *sugesstions.php* zamyslieť sa na predávaní id, či by nebolo lepšie id predať ako hidden input
- v *model* súbore *pract_requirements.php* sú tu natvrdo len tieto vybavenia: notebook, beamer, čo takto brať údaje z tabuľky vybavenie tým pádom aj upraviť pohľad aby tam dával všetky
- v *model* súbore *mail_list.php* poznamenať že ma dať textovú rolu
- v *model* súbore *lessons.php* delete na all možno komplikovaný, je možné mazať rovno záznamy vo foreach podľa predmetu a potom insertovať ak treba, potom vyhodí *id_semester* z modelu + pohľadu
- v *model* súbore *garant_requirements.php* sa v metóde *save()* používajú natvrdo konštanty skupín, môže raz načítať a potom len vyberať
- v *model* súbore *garant_requirements.php* v metóde *saveEdit()* sú napevno stanovené id, zamyslieť sa či neexistuje lepšie riešenie
- v *model* súbore *courses.php* v metóde *__getPrevPredmetID()* treba v budúcnosti ošetriť, že ak je zakázané predávanie vráti -1 (bude potrebná extra query na zistenie čo je možné pre požiadavky prebrať)
- v *model* súbore *courses.php* v metóde *__setRoleID()* je lepšie riešenie použiť statický číselník ako takto natvrdo
- v *model* súbore *collection.php* je možné pre budúce použitie definovať metódy *get* na aktívne obdobia, *get* na plánované obdobia
- v *model* súbore *calendar.php* rozšíriť triedu, aby vrátila všetky dôležité udalosti (začiatok, koniec semestra a pod.) a tiež vyriešiť, ktoré udalosti sú allDay a ktoré nie (kratšie ako 1 deň)
- v *core* súbore *notificator.php* v metóde *sendNotifyMessage()* ošetriť prípad, ak sa nepodarí odoslať mail
- v *core* súbore *connection.php* v metóde *get()* pri generovaní konštánt otestovať, či je možné rovno spraviť `unset $config`

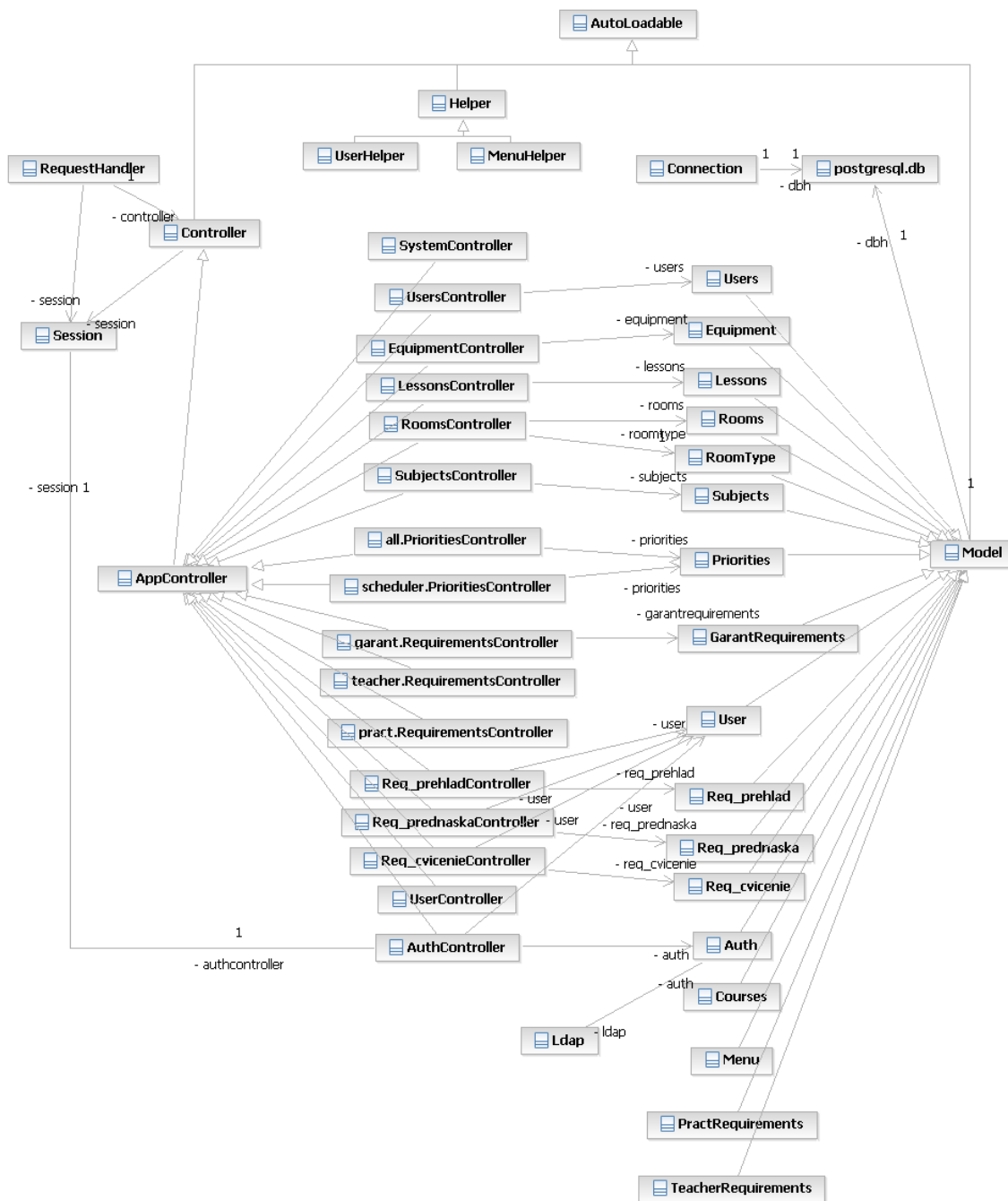
- v *controller* súbore *auth.php* v metóde *dologin()* dorobiť uchovávanie v DB pre usera jeho poslednú hodnotu
- v *controller/teacher*, *controller/pract*, *controller/garant* súbore *requirements.php* v metóde *commonEdit()* presunúť časť kódu pre zoznam minuloročných predmetov do metódy *__getCommonData*. Nebolo by vhodnejšie pole minuloročných predmetov dať iba vtedy keď nie je zadaná žiadna požiadavka t.j. keď predmet v roku nemá metapožiadavku
- v *controller/ape* v *lessons.php* v metóde *save()* otestovať možnosť, že ak zmením semester počas editácie predmetu garantom, tak či to náhodou nezmaže požiadavky v zlom semestri
- v *controller/ape* v *collection.php* v metóde *save()* možno pridať logovať aj semester (load podľa id a get.info)
- v *controller/all* v *calendar.php* v metóde *index()*, zamyslieť sa prečo je naviazanie na semester a či by nebolo korektnejšie zobrazíť všetky udalosti, keď bude scrollovať ďalej do histórie aby videl aj minuloročné bez toho aby musel prepínať semester
- v *controller/administrator* v *users.php* v metóde *submitEdited()* možnosť zmeny používateľského mena

10 Prílohy

- A. Diagram tried systému vytvoreného v školskom roku 2008/2009
- B. Model údajov systému zo školského roku 2008/2009
- C. Model údajov systému po úpravách v škol. roku 2009/2010
- D. Pripomienky k systému v zimnom semestri školského roku 2009/2010
- E. Návrh na úpravu skupín používateľov a ich privilégii
- F. Akceptačné testy LS 2009/2010
- G. Používateľská príručka
- H. Inštalačná príručka
- I. Obsah elektronického média

PRÍLOHA A Diagram tried systému vytvoreného v školskom roku 2008/2009

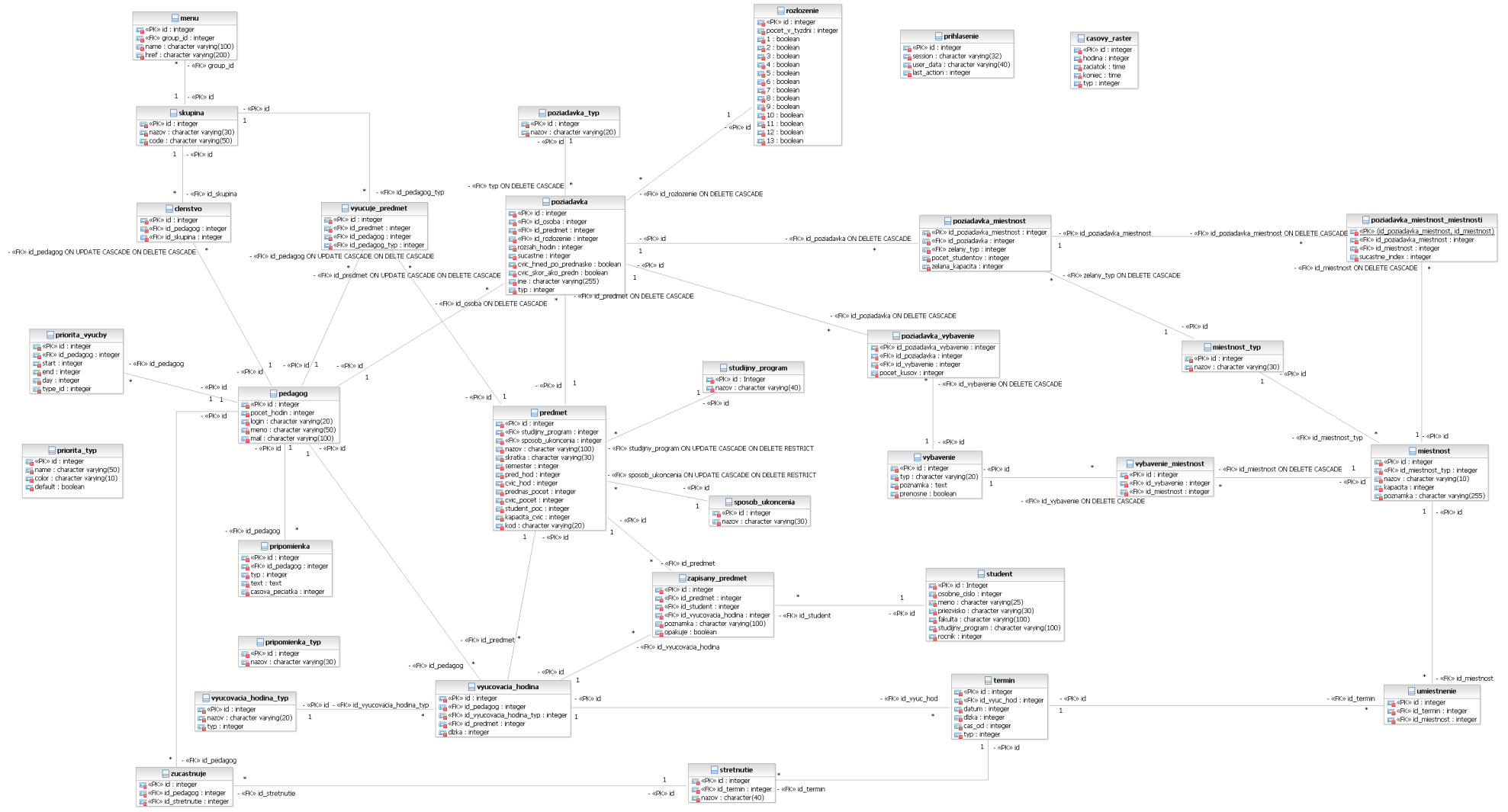
V tejto prílohe sa nachádza obrázok diagramu tried systému, ktorý sme dostali po tíme BackSpace a vytvorili sme ho, aby sme lepšie spoznali systém a tiež aby sme získali väčší prehľad.



Obr. 2 Diagram tried systému na podporu tvorby rozvrhov

PRÍLOHA B Model údajov systému zo školského roku 2008/2009

Uvádzame model údajov, ktorý vytvoril tím BackSpace pred nami, aby sme vedeli čo a ako sme upravili od minulého roku. Navyše sme tento model prekreslili v nástroji Rational Software Architect.



Obr. 3 Model údajov systému zo školského roku 2008/2009

**PRÍLOHA C Model údajov systému po úpravách v škol. roku
2009/2010**

Táto príloha obsahuje model údajov po úpravách, ktoré sme spravili my v rámci opravovania chýb alebo rozširovania systému.

PRÍLOHA D Pripomienky k systému v zimnom semestri školského roku 2009/2010

V tejto prílohe sa nachádzajú pripomienky k systému zaznamenané počas prvej testovacej prevádzky v zimnom semestri školského roku 2009/2010. Pripomienky sú v rovnakom formáte ako boli zadané.

Čas vloženia	Typ	Text	Vložil
8.12.2009 10:40	vylepšenie funkcionality	pocet studentov na prednaske automaticky zoberte z ais ci ineho systemu fakulty	prof. RNDr. Jiri Pospichal, DrSc.
10.12.2009 16:20	vylepšenie funkcionality	poziadavku na vybavenie miestnosti na cvicenia moze zrejme vkladat len veduci cviceni, myslim, ze tam by mal mat pristup aspon garant predmetu, ked uz nie prednasajuci;	doc. Ing. Danica Rosinova, PhD.
11.12.2009 10:40	iná	Ak predmet učia dvaja vyučujúci a cvičenie robí každý pre svoju odprednášanú oblasť, v systéme som nenašla možnosť dať oboch ako prednášajúcich aj cvičiacich. Asi je to naše špecifikum, ale potrebovali by sme to.	PhDr. Eva Letovancova, PhD.
13.12.2009 19:33	vylepšenie funkcionality	zadavanie rozsahu, prednasatela, veduceho cviceni by sa malo prenasat z predchadzajucich rokov a tu by sa to malo len upravovat, a rozsah by sa snad ani nemal takto volne upravovat	prof. Ing. Maria Bielikova, PhD.
13.12.2009 19:39	vylepšenie funkcionality	pri vybere mena zo zoznamu, v ktorom je viac ako 5 poloziek tieto treba usporiadat abecedne podla priezviska a idealne mat moznost priamo zadavanim sa dostat na konkretne meno, susasny stav je velmi narocny...	prof. Ing. Maria Bielikova, PhD.
13.12.2009 19:40	chyba v systéme	co s predmetom, v ktorom nie su cvicenia alebo prednasky, system nuti zadat prednasatela a aj veduceho cviceni	prof. Ing. Maria Bielikova, PhD.
13.12.2009 19:42	iná	nepride mi vhodne, aby cviciaci zadaval poziadavky na rozvrh, on to moze mozno zjemnit, ale zadavat by to mala za cely predmet jedna osoba, ak to tak nie je, vidim to ako problemovy predmet -- viem si predstavit riesenie ako v AIS, teda... pokrac dalej	prof. Ing. Maria Bielikova, PhD.

Čas vloženia	Typ	Text	Vložil
13.12.2009 19:43	iná	preco ma pripomienka len 255 znakov? je to velmi nemotorne, a asi je toto posledna pripomienka, ktoru sem pisem, pretoze nevidim co som pripomienkovala, pripomienky pisem ako veci objavujem a neda sa ani dopisat myslienka	prof. Ing. Maria Bielikova, PhD.
13.12.2009 19:45	iná	predsa len doplnenie pripomienky k poziadavkam zadanym cviciacimi -- urobit to ako v AIS, teda ak prednasatel oznaci cviciaceho ako administratora, tak potom moze modifikovat aj poziadavky na cvicenie, inak ich len vidi	prof. Ing. Maria Bielikova, PhD.
14.12.2009 0:48	iná	Komentar k prednaskam je obmedzeny na 255 znakov, ale ja som tam potrebovala napisat viac. Kedze mam dve prednasky za tyzden, tak som to rozdelila, ale nie je to tak vhodne. Komentar moze byt obmedzeny, aby sa tam ludia nevykecavali, ale 255 znakov je velmi malo. Tiez to okienko by bolo treba trochu zvecsit -- odporucam pridat 2 riadky.	prof. Ing. Maria Bielikova, PhD.
14.12.2009 0:50	iná	Trochu mi pri zadavani poziadaviek na prednasky chyba moznost doplnit poznamku. Jedna vec je mat specialne poziadavky a druha je mat k tomu celemu nejaku poznamku, napr. k predmetu FLP by som rada dopisala, ze su dve prednasky za tyzden, nebudu sice uplne kazdy tyzden dve, ale teraz neviem stanovit presne rozlozenie po tyzdnoch a preto treba rezervovat naplno.	prof. Ing. Maria Bielikova, PhD.

Čas vloženia	Typ	Text	Vložil
14.12.2009 0:53	chyba v systéme	<p>Povazujem za chybu, keď prednásateľ vloží vedúceho cvičení kolegu X a dôsledkom toho je, že sa nemože zadať požiadavky na cvičenia.</p> <p>Prosim reakciu čo mám s tým robiť, napr. vedúci cvičení predmetu FLP je Ing. Kapustík ale požiadavky na rozvrh zadávam ja ako prednásateľ, predsa ja viem ako je koncepcia predmetu, konzultujem to s cvičiacimi, ale ja preberám zodpovednosť a tak to tam aj zadám.</p> <p>Keby som toto delegovala na cvičiaceho, aj tak je nevyhnutné, aby som to videla.</p> <p>Ako postupovať? Zrušiť Ing. Kapustíka z pozície vedúceho cvičení?</p>	prof. Ing. Maria Bieliková, PhD.
14.12.2009 1:14	nová funkcionálnosť	<p>Do finálnej verzie by bolo dobré vidieť aj požiadavky z predchádzajúcich rokov a možnosť ich skopírovať a následne upraviť.</p> <p>S tým súvisí aj tlač požiadaviek v nejakej forme. Ak nebudem mať istotu, že sa k tým starým požiadavkám dostanem o rok, tak použitie systému bude značne obmedzené a bude veľmi nepríjemné si toto nejako povedľa systému znáčiť (standardne zoberiem to čo bolo vlani a urobíme brainstorming, čo zmeníme a potom to upravím... často sa stane, že to spravíme už po skončení predmetu, keď si poznám do súboru s požiadavkami na predchádzajúci rok aké zmeny by bolo dobré spraviť).</p>	prof. Ing. Maria Bieliková, PhD.

Čas vloženia	Typ	Text	Vložil
29.12.2009 0:20	vylepšenie funkcionality	<p>Pri vkladani casovych priorit by bolo vhodne, keby sa dal oznacit cas, kedy nemozem a napisat k tomu dovod. Inak niekto vsetko vycerveni a rozvrhar nevie nic, ale ked napisem, ze tu je senat, tu je kolegium, tu rano nie som schopna tak zavcasu pracovat... tak je to cele v inej pozicii.</p> <p>Asi nebude vhodne vyzadovat zdovodnenie kazdej hodiny, ale tam kde to ucitel napise, je to urcite plus. Realne by to v pracovnej dobe (od osmej do stvrtej) mal byt kazdy schopny zodovodnit preco nemoze.</p>	prof. Ing. Maria Bielikova, PhD.

PRÍLOHA E Návrh na úpravu skupín používateľov a ich privilégií

V tejto prílohe je uvedený vypracovaný návrh na úpravu skupín (rolí) používateľov systému a ich privilégii (momentálne sa všetky privilégii kontrolujú podľa príslušnosti v skupine a je možné to oddeliť). Vzhľadom na nedostatok času táto úprava nebola zapracovaná. Tento návrh je výstupom úlohy jedného člena tímu a preto je písaný v prvej osobe.

Samotný návrh:

Navrhujem vytvorenie nových tabuliek „skupina_privilegia“ a „privilegia“, pričom „skupina“ by predstavovala ako aj teraz rôznych ľudí v systéme (alebo aj mimo neho – napríklad nové skupiny študent a cvičiaci) a „privilegia“ by boli vlastne rôzne možnosti, čo sa dá robiť v systéme (teraz sa toto robí cez skupiny (napríklad zobrazovanie niektorých častí menu, prístup k rôznym funkcionalitám a pod.)). V podstate ide o oddelenie formálneho rozdelenia kategórií používateľov a ich funkcií (prepojenie potom cez väzobnú tabuľku). Z tohto vyplýva:

- V tabuľke „vyucuje_predmet“ by zostalo zadelenie vyučujúceho tak ako je (akurát by som premenoval id_pedagog_typ na id_pedagog_skupina), ale jednotlivé skupiny by mali definované svoje privilégii (garant bude môcť definovať rozsah predmetu, dávať požiadavky na prednášky aj cvičenia... prednášajúci bude môcť dávať požiadavky na prednášky aj cvičenia)
- Nebudú musieť byť ľudia vo všetkých skupinách aby mali všetky privilégii (napríklad my ako admini si môžeme dať všetky privilégii a stačí ak budeme v skupine admin). Tu sa zas stráca pridelovanie privilégii pre jednotlivé osoby (ak by mal mať nejaké špeciálne privilégii ktoré nezodpovedajú vytvoreným skupinám), ale v prípade potreby by sa mohla dorobiť väzobná tabuľka pedagog_privilegia
- Môže sa vytvoriť nová skupina bez privilégii (napríklad vedúci myslím chcel evidenciu všetkých cvičiacich v predmete, takže v tabuľke vyucuje_predmet môže mať skupinu cvičiaci bez privilégii)
- Je to celkom pekné pre ďalšie rozšírenia do budúcnosti, keď budú vznikať nové skupiny s rôznymi privilégiami

PRÍLOHA F Akceptačné testy LS 2009/2010

V tejto prílohe sa nachádzajú akceptačné testy vykonané v letnom semestri školského roku 2009/2010 v rámci testovacej prevádzky prototypu.

PRÍLOHA G Používateľská príručka

V tejto prílohe sa nachádza používateľská príručka k systému pre podporu tvorby rozvrhov.

PRÍLOHA H Inšalačná príručka

V tejto prílohe uvádzame inšalačnú príručku k systému pre podporu tvorby rozvrhov.

PRÍLOHA I Obsah elektronického média

Adresár	Obsah
\Src	Aktuálna verzia systému k dátumu 3.5.2010
\Doc	Dokumenty týkajúce sa aktuálneho systému k dátumu 3.5.2010
\docs, minulorocny tim	Dokumenty vypracované tímom Backspace v školskom roku 2008/2009
\Ponuky	Elektronická verzia ponúk tímu Bug Hunters na predmet Tímový projekt v školskom roku 2009/2010
\Riadenie	Elektronická verzia dokumentov riadenia tímu Bug Hunters
\Systemova_dokumentacia	Elektronická verzia tohto dokumentu a ďalších dokumentov týkajúcich sa projektu k dátumu 3.5.2010
\DB	Záloha databázy k dátumu 3.5.2010