

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ
FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

INFORMAČNÍ SYSTÉM ŠKOLY

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

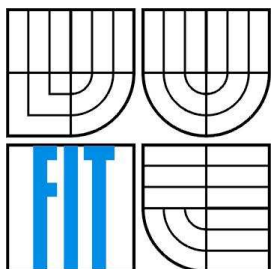
AUTOR PRÁCE
AUTHOR

Martin Topinka

BRNO 2007



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

INFORMAČNÍ SYSTÉM ŠKOLY SCHOOL INFORMATION SYSTEM

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

Martin Topinka

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. Roman Lukáš, Ph.D.

BRNO 2007

Zadání bakalářské práce

Řešitel: **Topinka Martin**
Obor: Informační technologie
Téma: **Informační systém školy**
Kategorie: Web

Pokyny:

1. Seznamte se s jazyky a prostředky pro tvorbu webových informačních systémů (XHTML, CSS, PHP, Javascript, MySQL).
2. Proveďte obecně analýzu požadavků pro informační systém školy, například gymnázia.
3. Navrhněte vhodnou strukturu systému. Při analýze požadavků návrhu využijte vhodných technik. Složitost vašeho návrhu konzultujte s vedoucím BP. Systém musí mít i speciální funkčnost založenou na nějaké metodě z umělé inteligence. (například automatické generování rozvrhů apod.) Dále musí umožňovat vytváření tiskových sestav (například tisk vysvědčení).
4. Daný systém implementujte tak, aby byl použitelný pro danou školu v praxi.
5. Zhodnoťte dosažené výsledky, porovnejte váš systém s existujícími systémy, navrhněte další možné rozšíření do budoucna.

Literatura:

- Kosek, J.: PHP - Tvorba interaktivních internetových aplikací

Vedoucí: **Lukáš Roman, Ing., Ph.D., UIFS FIT VUT**
Oponent: **Květoňová Šárka, Ing., UIFS FIT VUT**
Datum zadání: 1. listopadu 2006
Datum odevzdání: 15. května 2007

**LICENČNÍ SMLOUVA
POSKYTOVANÁ K VÝKONU PRÁVA UŽÍT ŠKOLNÍ DÍLO**

uzavřená mezi smluvními stranami:

1. Pan/paní

Jméno a příjmení: **Martin Topinka**
Id studenta: 89362
Bytem: Palachova 38, 591 01 Žďár nad Sázavou
Narozen: 5. 2. 1984, Nové Město na Moravě
(dále jen „autor“)

a

2. Vysoké učení technické v Brně

Fakulta informačních technologií
se sídlem Božetěchova 2/1, 612 66 Brno, IČO 00216305
jejímž jménem jedná na základě písemného pověření děkanem fakulty:

.....
(dále jen „nabyvatel“)

**Článek 1
Specifikace školního díla**

1. Předmětem této smlouvy je vysokoškolská kvalifikační práce (VŠKP):

bakalářská práce

Název VŠKP: Informační systém školy
Vedoucí/ školitel VŠKP: Lukáš Roman, Ing., Ph.D.
Ústav: Ústav informačních systémů
Datum obhajoby VŠKP:

VŠKP odevzdal autor nabyvateli v:

tištěné formě počet exemplářů: 1
elektronické formě počet exemplářů: 2 (1 ve skladu dokumentů, 1 na CD)

2. Autor prohlašuje, že vytvořil samostatnou vlastní tvůrčí činností dílo shora popsané a specifikované. Autor dále prohlašuje, že při zpracovávání díla se sám nedostal do rozporu s autorským zákonem a předpisy souvisejícími a že je dílo dílem původním.
3. Dílo je chráněno jako dílo dle autorského zákona v platném znění.
4. Autor potvrzuje, že listinná a elektronická verze díla je identická.

Článek 2

Udělení licenčního oprávnění

1. Autor touto smlouvou poskytuje nabyvateli oprávnění (licenci) k výkonu práva uvedené dílo nevýdělečně užít, archivovat a zpřístupnit ke studijním, výukovým a výzkumným účelům včetně pořizování výpisů, opisů a rozmnoženin.
2. Licence je poskytována celosvětově, pro celou dobu trvání autorských a majetkových práv k dílu.
3. Autor souhlasí se zveřejněním díla v databázi přístupné v mezinárodní síti
 - ihned po uzavření této smlouvy
 - 1 rok po uzavření této smlouvy
 - 3 roky po uzavření této smlouvy
 - 5 let po uzavření této smlouvy
 - 10 let po uzavření této smlouvy(z důvodu utajení v něm obsažených informací)
4. Nevýdělečné zveřejňování díla nabyvatelem v souladu s ustanovením § 47b zákona č. 111/1998 Sb., v platném znění, nevyžaduje licenci a nabyvatel je k němu povinen a oprávněn ze zákona.

Článek 3

Závěrečná ustanovení

1. Smlouva je sepsána ve třech vyhotoveních s platností originálu, přičemž po jednom vyhotovení obdrží autor a nabyvatel, další vyhotovení je vloženo do VŠKP.
2. Vztahy mezi smluvními stranami vzniklé a neupravené touto smlouvou se řídí autorským zákonem, občanským zákoníkem, vysokoškolským zákonem, zákonem o archivnictví, v platném znění a popř. dalšími právními předpisy.
3. Licenční smlouva byla uzavřena na základě svobodné a pravé vůle smluvních stran, s plným porozuměním jejímu textu i důsledkům, nikoliv v tísní a za nápadně nevýhodných podmínek.
4. Licenční smlouva nabývá platnosti a účinnosti dnem jejího podpisu oběma smluvními stranami.

V Brně dne:

.....

Nabyvatel

.....

Autor

Abstrakt

Cílem této práce je analyzovat a vyhodnotit požadavky na informační systém určený pro základní a střední školy. Práce analyzuje potřeby uživatelů a diskutuje vlastnosti, které by tento typ informačního systému měl mít. Dále se zabývá návrhem, výběrem vhodných implementačních technologií a vlastní implementací informačního systému pro základní nebo střední školu. V práci jsou rozebrány problémy vzniklé během implementace a diskutovány dosažené výsledky. Důležitou částí práce je řešení návrhu a realizace automatického generování rozvrhů. V závěru práce je diskutováno možné budoucí rozšíření systému.

Klíčová slova

Informační systém, PHP, MySQL, JavaScript, HTML, CSS.

Abstract

The aim of this thesis is to analyse and to assess requirements on information system which is designated for primary and high schools. The thesis analyses user needs and discusses characteristics of such an information system. The proper implementation tools are chosen. It also covers a design and implementation part of the information system creation. Furthermore this report analyses problems which arised during the implementation stage and discusses final achievements. In conclusion the possible options for the future upgrades are discussed.

Keywords

Information system, PHP, MySQL, JavaScript, HTML, CSS.

Citace

Topinka Martin: Informační systém školy. Brno, 2007, bakalářská práce, FIT VUT v Brně.

Informační systém školy

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením Ing. Romana Lukáše, Ph.D. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Martin Topinka
27.6.2007

Poděkování

Rád bych na tomto místě poděkoval Ing. Romanu Lukášovi, Ph.D. za to, že se mě ujal a mohl jsem pod jeho odborným vedením vypracovat tuto bakalářskou práci. Dále bych zde chtěl poděkovat Dr inž. Iwonie Dubielewicz (Politechnika Wrocławska - Wydział Informatyki i Zarządzania) za hodnotné rady a připomínky, které jsem obdržel během pravidelných konzultací při mém pobytu v Polsku. Také děkuji Ing. Šárce Květoňové za odborný posudek mé práce a čas strávený nad jejím posuzováním. V poslední řadě mé díky patří Fakultě informačních technologií a celému Vysokému učení technickému v Brně za to, že mi umožnili studium na vysoké škole a tím i vypracování této práce a také mojí mámě za podporu a za to, že mě přivedla na tento svět. Všem výše zmíněným patří mé díky.

© Martin Topinka, 2007.

Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů..

Obsah

Obsah.....	1
Úvod.....	3
1 Analýza a návrh.....	4
1.1 Informační systémy	4
1.1.1 Informační systém školy.....	4
1.1.2 Existující systémy	5
1.2 Analýza školy.....	6
1.2.1 Účel systému.....	6
1.2.2 Změny pro školu	7
1.3 Prohlášení o cílech.....	7
1.4 Rozbor požadavků	8
1.4.1 Všeobecné požadavky	8
1.4.2 Rozvrh hodin.....	8
1.4.3 Omezení.....	9
1.4.4 Přístupnost	9
1.5 UML	10
1.5.1 Use Case diagram.....	10
1.5.2 E-R diagram.....	11
2 Použité technologie.....	13
2.1 HTML.....	13
2.2 CSS.....	13
2.3 PHP.....	14
2.4 MySQL.....	14
2.5 JavaScript.....	15
3 Implementace	16
3.1 Uživatelské rozhraní	16
3.1.1 Členění systému	16
3.1.2 Ovládání.....	17
3.2 Struktura školy	18
3.2.1 Uživatelé systému	18
3.2.2 Evidence uživatelů	18
3.2.3 Třídy, učebny a předměty	19
3.3 Tvorba rozvrhů.....	20
3.3.1 Přidělování učitelů.....	21

3.3.2	Automatická tvorba rozvrhů	22
3.4	Systemové změny	25
3.4.1	Ukončení školního roku.....	25
3.5	Bezpečnost	25
3.6	Zhodnocení implementační části.....	26
4	Možná rozšíření systému.....	27
5	Závěr	30
	Literatura	31
	Příloha 1 - Instalace systému	32
	Příloha 2 - Obsah přiloženého CD	33
	Příloha 3 - Use case model	35
	Příloha 4 - E-R diagram.....	36

Úvod

Během svého bakalářského studia na Fakultě informačních technologií Vysokého učení technického v Brně jsem měl stejně jako všichni moji kolegové příležitost proniknout do mnoha oblastí světa informačních technologií a možnost naučit se základy mnoha programovacích jazyků. Výběr bakalářské práce nabízel možnost zdokonalit se a více prohloubit znalosti v jedné z oblastí. Protože jsem neměl velké zkušenosti s tvorbou webových aplikací, rozhodl jsem se pro výběr tématu s touto problematikou. Informační systém školy, který by měl řešit automatické generování rozvrhů, byl zajímavým tématem, což se během tvorby této práce potvrdilo.

Počítače v dnešní době již pronikly do nejrůznějších oblastí. Automatizovaly různé procesy a napomáhají člověku při práci všeho druhu. Díky počítačům se zefektivnila, zrychlila a zkvalitnila práce v mnoha oblastech lidské činnosti. Jednou takovou oblastí, kde se prosadilo použití počítačů, jsou i nejrůznější instituce. Zde nasazení počítačů umožnilo nahradit papírovou agendu a vedení evidence pomocí informačních systémů. Jednou z institucí vhodnou pro nasazení informačních systémů bylo i školství. Informační systémy se stávají aktuálními nejen pro vysoké školy, ale i pro školy základní a střední. Zde informační systémy umožňují vedení evidence o studentech a zaměstnancích školy, napomáhají utváření struktury školy a slouží pro podporu výuky.

Tato práce se zabývá takovým typem informačního systému. V první kapitole je analyzován takový systém. Jsou v ní popsány cíle a požadavky na systém, potřeby různých typů uživatelů. Dále se v této kapitole budeme zabývat návrhem systému a graficky si popíšeme systém pomocí dvou typů diagramů (Use Case Diagram a E-R Diagram).

Druhá kapitola se zabývá výběrem vhodných technologií, které jsou použity pro implementační část. Dále následuje kapitola nazvaná „Implementace“, která popisuje tvorbu a funkčnost vytvořeného informačního systému. Významnou částí této kapitoly je řešení problematiky generování rozvrhů. V této kapitole jsou také diskutovány problémy, které vznikly během implementace systému.

Čtvrtá kapitola se zabývá možnými rozšířeními systému do budoucna a kapitola nazvaná „Závěr“ je potom shrnutím a uzavřením celé práce.

1 Analýza a návrh

V této kapitole bude vysvětlen pojem informační systém (dále jen IS). Bude popsáno jak by měl IS vypadat pro použití školy a jaké systémy jsou v současné době dostupné. Zaměříme se na účel, který by měl takový informační systém splňovat, jaké změny a přínos pro školu má zavedení informačního systému. Rozebereme zde požadavky na IS školy, jaké funkce by měl poskytovat, pojem přístupnost a definování omezení pro zajištění konzistence celého systému. Dále si vysvětlíme pojem UML a jakou úlohu hraje při vývoji systému. Závěr kapitoly je věnován grafickému popisu námi navrhovaného systému pomocí vhodných diagramů.

1.1 Informační systémy

Informační systém umožňuje ukládání, zpracování, mazání a poskytování informací. V našem případě se bude jednat o informační systém automatizovaný, tedy pomocí počítačů. Informační systém bychom mohli popsat jako systém, který má vstupní část, kterou se informace do systému vkládají, a výstupní část, ze které se informace získávají. Mezi těmito dvěma částmi je transformační část, která provádí nad daty různé operace pomocí vhodných algoritmů. Některé druhy informačních systémů jsou tyto:

- Geografické informační systémy
- Manažerské informační systémy
- Ekonomický informační systém
- Podnikový informační systém
- Rezervační systémy

My se dále budeme zabývat informačním systémem určeným pro školu, konkrétně pro základní nebo střední školu.

1.1.1 Informační systém školy

Jedná se o informační systém, který je zaměřen na podporu školy. Systémy by se daly podle funkce rozdělit do více kategorií např. na systémy, které jsou určeny pro účetnictví školy, systémy pro podporu řízení školy, pro evidenci zaměstnanců. My se v této práci zaměříme na systémy, které jsou především určené pro podporu výuky a správy struktury školy. Takový informační systém by měl za pomoci výpočetní techniky umožnit nahrazení, zrychlení a usnadnění některých procesů probíhajících ve škole jiným způsobem. Mezi základní funkce by se dala zařadit evidence vyučovaných předmětů, možnost hodnocení studentů, evidence získaných známek za jednotlivé předměty, tvorba rozvrhů,

umožnění vyhledávání nejrůznějších informací ohledně studia a zpřístupnění systému pro různé uživatele (studenty, učitele, ředitele školy).

1.1.2 Existující systémy

Na trhu je několik systémů, které nabízejí základním a středním školám podporu v podobě informačního systému, který řídí agendu školy. Systémy se liší v závislosti na konkrétním určení, ale v mnoha ohledech se funkčně překrývají. Základními funkcemi systémů jsou:

- Možnost hodnocení studentů
- Správa rozvrhů
- Evidence studentů, učitelů a předmětů
- Informace o škole, vyhledávání osob
- Web mail
- On-line výuka
- Tvorba tiskových sestav
- Komunikace
- Správa domácích úkolů
- Bezpečné přihlášení, rozlišení uživatelů do rolí

Některé současné dostupné systémy v ČR:

- ISvoš
- SAS
- iškola.cz - <http://www.iskola.cz/>
- eŠkola - <http://www.eskola.cz/>
- KATEDRA - <http://katedra.skolaonline.cz/>

Systém SAS je nejrozšířenějším systémem nabízejícím řešení agendy školy. Počítá se u něj s nasazením v prostředí školy. Jeho zajímavou vlastností je funkce vyhodnocování přijímacích zkoušek. Poslední tři uvedené systémy jsou vytvořeny jako webová aplikace a tedy systémy, kterými se budeme zabývat i my. Některé ze zajímavých funkcí jsou elektronická omluvenka, vyhodnocování domácích úkolů nebo vývěska.

1.2 Analýza školy

Pro tvorbu informačního systému je potřeba znát parametry školy. Tedy jak je škola členěna, kolik studentů přibližně studuje v jedné třídě, kolik let studují apod. Za vzor si vezmeme školu s následujícími parametry:

- doba studia do devíti let (doba studia na základní škole)
- každý předmět je vyučován po dobu jednoho roku
- přibližně čtyři třídy v každém ročníku
- v každém roce studenti studují přibližně 10 předmětů
- v každé třídě přibližně 25 studentů
- přibližně 40 vyučujících na škole

Tyto parametry neomezují systém pro použití pouze u tohoto typu školy, ale dávají obraz pro jaké množství studentů, učitelů, předmětů atd. by měl být systém koncipován. Systém musí být použitelný jak pro malé, tak pro velké školy, pro školy s nejrůznějším členěním.

1.2.1 Účel systému

Jedním s hlavních účelů je zrychlení a zkvalitnění funkcí, které škola do této doby vykonávala pomalým nebo zaostalým způsobem. Výsledkem zavedení IS pro školu by měla být velká úspora času, který může být věnován užitečnějším činnostem. Informační systém by měl umožnit ukládat, měnit a mazat množství informací, které do této doby byly v nepraktické podobě.

Toto se týká například nahrazení papírové evidence detailů o všech studentech a učitelích elektronickou obdobou, což umožní snadné vyhledání a případnou změnu jakékoliv informace. Systém by měl umožnit přechod od papírové podoby všech záznamů do elektronické. Příkladem může být například zrušení normálních žákovských knížek. Učitelé mohou jednoduše zadávat hodnocení do systému, ve kterém si mohou později sami známky za konkrétní předměty vyhledat, stejně dobře jako studenti nebo rodiče studentů, kteří si v informačním systému mohou zkontrolovat výsledky svých dětí.

Řízení školy by informační systém měl zjednodušit práci s rozvržením struktury školy. Měl by umožňovat náhled s případnou změnou počtu tříd, názvů tříd, rozdělení studentů do tříd. V případě, že IS bude obsahovat potřebné množství informací o struktuře školy, není problémem vytvářet automaticky v IS školy rozvrhy pro všechny třídy. To je efektivní nahrazení zdlouhavé metody zaznamenávání všeho na papír, kdy pověřená osoba musí dohledávat, kdy, kde a kdo učí, kdy je která učebna volná, která třída studuje ten a ten předmět a kdo zrovna vyučuje konkrétní předmět pro konkrétní třídu.

1.2.2 Změny pro školu

Změny pro školu vyplývají z předchozí části. Jednou ze změn je úspora času, kterou pracovníci školy strávili dohledáváním informací a prací s velkým množstvím dokumentů a papírů. Mělo by dojít také ke zefektivnění a zjednodušení provedení změn ve struktuře školy. Změnou může být například ideálnější výsledek vytvořených rozvrhů, kdy je počítač schopen při znalosti požadavků na rozvrh vytvořit takovou kombinaci rozvrhů, jakou by manuálně pracovník školy nebyl schopen v reálném čase vytvořit.

Za další změnu by se dalo označit možné snížení administrativních pracovníků školy a potřeba nákupu počítačového vybavení, který bude umožňovat běh systému. Jednou z možných změn může také být zrušení zadávání a odevzdávání domácích úkolů běžným způsobem a zavedení vypracovávání úkolů prostřednictvím IS. Jedná se opět o ušetření času, kdy systém je schopný vyhodnotit domácí úkoly za učitele, který se může věnovat přípravě na vyučování nebo jiným činnostem.

1.3 Prohlášení o cílech

Pro stanovení požadavků, které bude následovat za touto částí tohoto dokumentu, je důležité si ujasnit, jakých cílů chceme dosáhnout. To znamená, pro jaké uživatele bude systém určen, na jaký cíl bude zaměřený a co by měl ve své výsledné podobě umožňovat. Ze stanovení cíle můžeme rozebrat jednotlivé požadavky, formalizovat je a později namodelovat budoucí informační systém.

Je zřejmé, že není v možnostech jedné osoby vytvořit jako výsledek této práce plnohodnotný informační systém školy, který by umožňoval širokou škálu funkcí požadovaných od takového systému. Vymezíme si tedy konkrétní část, která je realizovatelná během stanoveného časového období. Systém ale vytvoříme tak, aby bylo snadné v budoucnu jeho rozšíření o další funkce. Více o možných rozšířeních je diskutováno v kapitole „Možná rozšíření systému“. Jedním z hlavních výsledků systému by měla být možnost automatické tvorby rozvrhů pro všechny třídy. Tvorba bude tedy směřovat k tomuto cíli a systém musí obsahovat veškeré informace potřebné ke generování rozvrhu. Hlavní cíle nebo požadované funkce systému stanovíme na:

- Bezpečné přihlášení do systému
- Různé uživatelské role
- Evidence studentů
- Evidence učitelů
- Evidence vyučovaných předmětů
- Správa struktury školy (rozdělení do tříd, přidělení vyučujících)
- Generování rozvrhů
- Ukončení roku a přesun do dalšího školního roku

1.4 Rozbor požadavků

Zatím jsme se zabývali všeobecně tím, co to je školní informační systém, jaké změny přináší škole a jaký je jeho účel. Již máme stanovené cíle, můžeme si v této části tedy rozebrat konkrétněji co systém musí umožňovat a jaké jsou požadavky na jeho funkčnost.

1.4.1 Všeobecné požadavky

Základem systému by mělo být bezpečné přihlášení do systému s různým přístupem pro různé druhy uživatelů. Z pohledu školy to budou studenti, učitelé a administrátoři (ředitel školy, administrativní pracovníci). Například student by neměl mít stejný přístup do systému jako učitel. K tomuto bude zapotřebí přístupový login a heslo.

Systém by měl umožňovat evidenci o učitelích a studentech školy. U každého profilu by měly být detaily jako je jméno, příjmení, titul, adresa, telefon, email, případně foto. Systém by měl umožňovat vytváření, změnu a mazání těchto profilů.

Vytváření, změny a mazání by se také měly týkat evidence dalších položek systému jako jsou například vyučované předměty. U vyučovaných předmětů by měla být informace, který učitel má na starost daný předmět, pro který ročník je vyučován, anotace předmětu, doporučená literatura, obsah vyučovacích hodin, komentář k předmětu.

Dále by IS školy měl být schopný utvářet strukturu školy. Tzn. vytváření tříd s následným rozdělováním studentů do tříd, převádění studentů mezi třídami, přiřazování vyučujících konkrétním třídám pro konkrétní výuku. Měl by mít údaj o tom, od kdy do kdy probíhá která vyučovací hodina atd. IS by také měl obsahovat záznamy o tom, které učebny jsou dostupné pro výuku a v které učebně může být konkrétní předmět vyučován.

Důležitým prvkem IS je uzavření školního roku s přeřazením studentů do dalšího roku, vložení studentů prvního ročníku a vyřazení studentů ročníku posledního.

1.4.2 Rozvrh hodin

Jedním z důležitých požadavků na náš IS školy je možnost tvorby rozvrhů. Tvorbu rozvrhů bychom mohli rozdělit na dva typy. Prvním typem by byla možnost manuálního vytváření. Tím rozumíme funkčnost, kdy bude uživatel jednotlivým vyučovacím oknům rozvrhu dané třídy přidělovat vyučovaný předmět a učebnu ve které bude probíhat.

Druhým důležitějším typem je automatická tvorba rozvrhů. Při této možnosti by měl systém sám z uložených dat vygenerovat rozvrhy pro všechny třídy. Systém automaticky pro danou třídu přiřadí předmět na určitý den a hodinu do určité učebny. Je zapotřebí vědět, kolik hodin týdně má daný předmět být vyučován, v které učebně má nebo může být vyučován. Je zapotřebí zajistit to, že například počítačová cvičení budou probíhat v místnosti s počítači a matematika nebude alokována do

tělocvičny. Automatické generování musí ošetřit možné kolize a vygenerovat akceptovatelný rozvrh. Učitel nemůže vyučovat ve stejnou dobu ve více třídách, stejně jako v jedné učebně v jednu dobu nemůže probíhat výuka více jak jedné třídy. Z vytvořeného rozvrhu by měla jít vytvořit sestava pro tisk.

1.4.3 Omezení

IS systém školy je systém, na který jsou kladeny velké nároky na konzistenci. Je zapotřebí implementovat omezení, která udrží systém v konzistentním stavu. Údaje v systému nemohou být libovolně mazána. Například není možné vymazat danou třídu v případě, že v ní studují nějakí studenti, nebo vymazat profil vyučujícího, který vyučuje nějaký předmět, protože danou třídu by později neměl kdo učit. V takovém případě musí být prvně zajištěna náhrada. Také učebny nezle vymazávat libovolně ze systému, pokud v nich probíhá nějaká výuka. Nejprve musí být výuka přeřazena do jiné třídy a poté můžeme vymazat učebnu.

Je také potřeba zajistit nemožnost vymazání všech administrátorů, což by způsobilo nemožnost přístupu do administrační části systému. Stejně jako je vhodné ošetřit vstupy tak, aby nebyly vkládány nekorektní data, např. rodné číslo ve špatné formě atd. Musí být zajištěna také unikátnost přihlašovacích loginů. U tvorby rozvrhů je potřeba zajistit, aby učitel neučil na dvou místech ve stejnou dobu, aby třída neměla více hodin výuky některého předmětu týdně než by měla mít apod.

1.4.4 Přístupnost

Jedním z požadavků je také přístupnost webové aplikace. Pod tímto pojmem chápeme určité vlastnosti, které by měla webová aplikace splňovat a dodržovat. Mezi hlavní pravidla přístupnosti patří:

- Dostupný a čitelný obsah
- Korektní zobrazování ve většině prohlížečů
- Práci se stránkou si řídí uživatel
- Netextové prvky mají svoji textovou alternativu
- Informace jsou srozumitelné a přehledné
- Ovládání webu jasné a pochopitelné
- Zřetelné odkazy odlišné od okolního textu
- Kód strukturovaný a odpovídající specifikaci HTML, případně XHTML

Více informací o přístupnosti nalezneme na [6].

1.5 UML

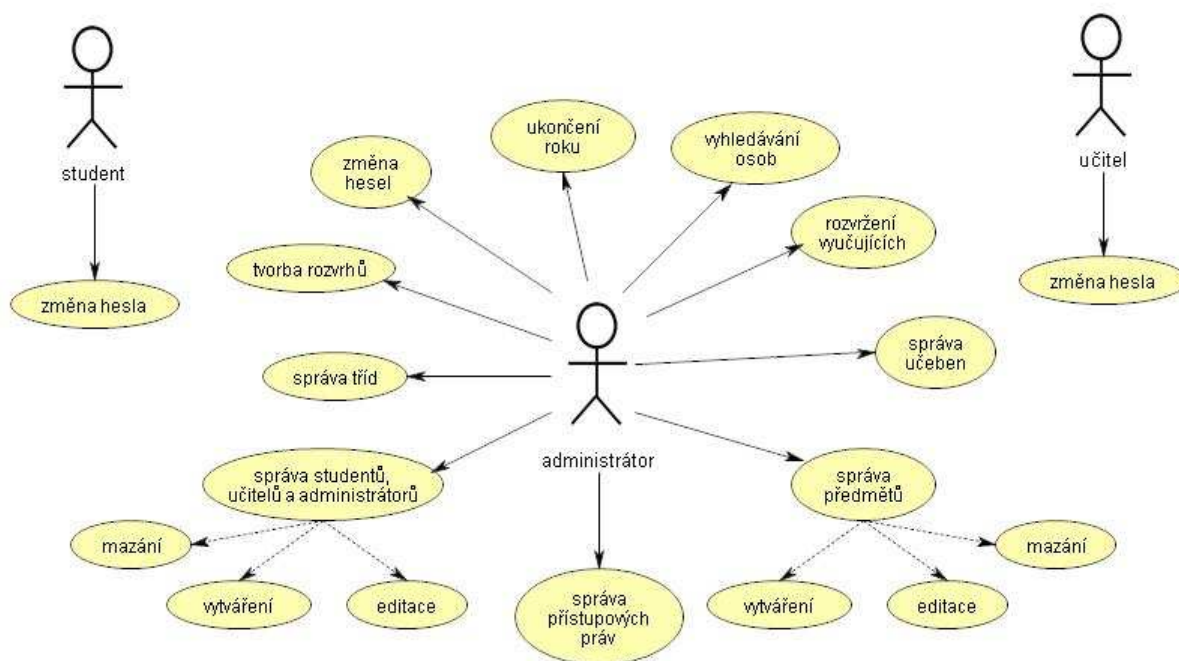
UML je zkratkou pro Unified Modeling Language. Jedná se o grafický jazyk používaný pro vizualizaci, specifikaci a návrh programových systémů [8]. Stal se oblíbeným, protože se jedná o sjednocující standard mezi analytiky, návrháři a programátory. UML podporuje objektově orientovaný přístup k analýze a návrhu. UML pomáhá pomocí grafických návrhů vývojářům při komunikaci a vyjadřuje myšlenky před tím, než začne implementační část vývoje programu. Pro takový grafický návrh je důležitá rychlost, srozumitelnost a snadnost provedení změn. Při návrhu pomocí UML by měl analytik být schopen zaznamenat všechny prvky systému, aby programátor byl schopen z takového návrhu vytvořit jednoduše funkční program. Z UML diagramů je též možné vytvořit spustitelný kód. Toto ale vyžaduje přesný popis v UML diagramech.

1.5.1 Use Case diagram

Use Case diagram nebo také diagram případů užití je součástí UML [7]. Popisuje činnost systému z hlediska pohledu vnějšího pozorovatele, chování systému z hlediska uživatelů. Jsou v něm zaznamenány všechny druhy uživatelů, kteří pracují se systémem. Dále diagram zobrazuje případy užití. Jedná se o určité použití nebo operace, které uživatel (účastník) provede se systémem. Use Case diagram pro náš systém je na obrázku 1.1.

Na obrázku jsou zobrazeny tři typy účastníků - učitel, student a administrátor. Učitelé a studenti mají při současném stavu systému možnost se za pomoci svého přihlašovacího jména a hesla přihlásit a změnit si přístupové heslo. Diagram se pro ně změní o velké množství případů užití v případě rozšíření části IS školy pro studenty nebo učitele. Třetí role v systému je administrátor, který má přiděleny následující případy použití:

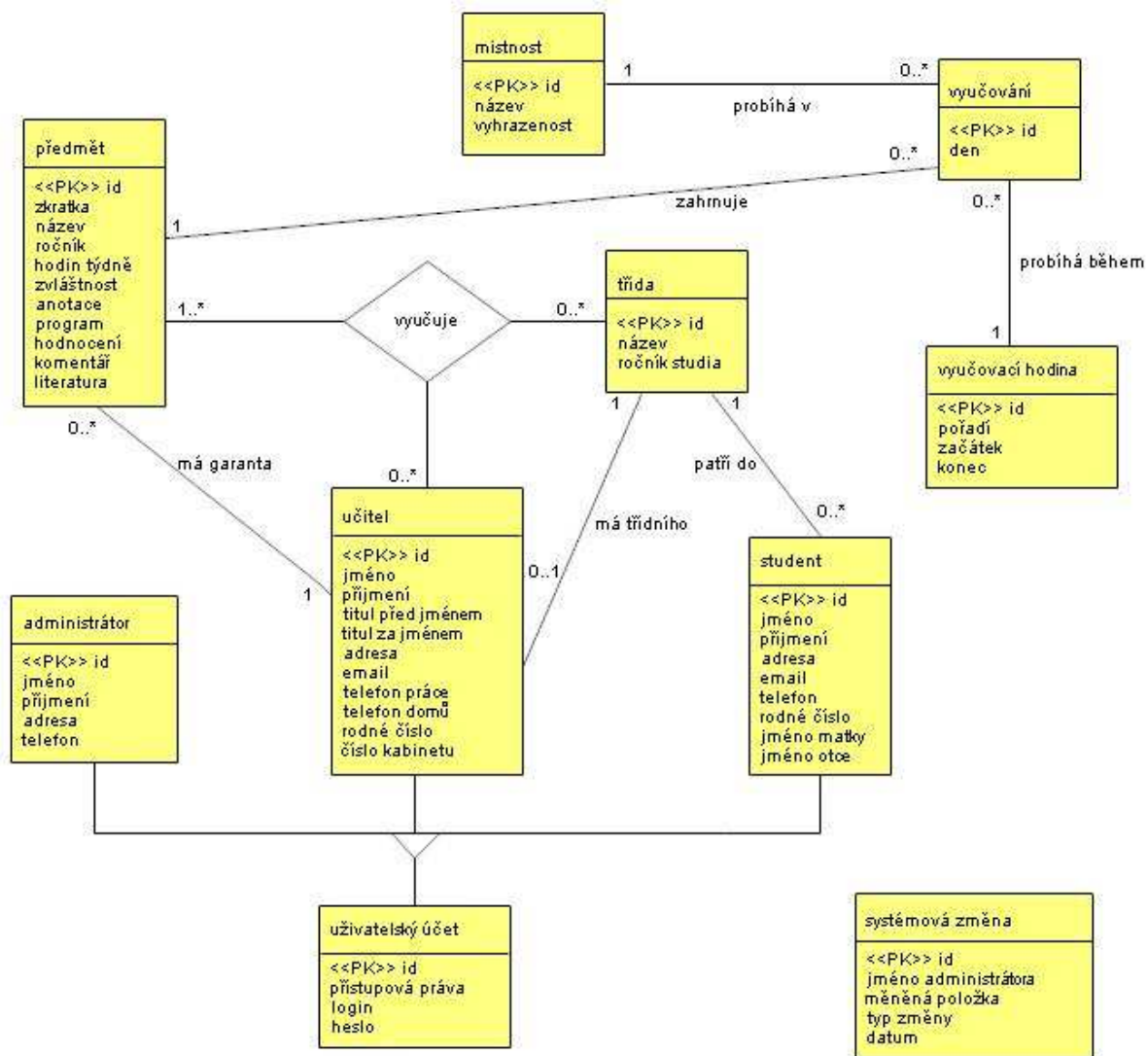
- změna hesel – změna hesel uživatelů systému
- ukončení roku – možnost uzavření školního roku a změnu na nový školní rok, zahrnuje přesun studentů do dalšího ročníku
- vyhledávání osob – vyhledávání detailů o uživateli systému
- rozvržení vyučujících – přidělování určité výuky konkrétním učitelům
- správa učeben – vytváření záznamů o učebnách
- správa předmětů – vytváření, mazání a editace detailních záznamů o předmětech
- správa přístupových práv – kontrola přístupových práv uživatelů systému
- správa studentů, učitelů a administrátorů – vytváření, mazání a editace detailních záznamů o studentech, učitelích a administrátorech
- správa tříd – rozdělování studentů do tříd
- tvorba rozvrhů – vytváření rozvrhů pro třídy



Obr. 1.1: Use Case diagram pro IS školy

1.5.2 E-R diagram

Jedná se o Entity-Relationship diagram. E-R diagram není přímo součástí UML, ale je podobný diagramu tříd, který součástí UML je. E-R diagram modeluje strukturu databáze, používá se pro konceptuální modelování, což je fáze datové analýzy. Z konceptuálního návrhu vznikne logický návrh (tabulky) a z logického návrhu později fyzický návrh (uložené záznamy). E-R diagram se skládá z entit, vztahů mezi entitami a atributů, což jsou vlastnosti entity. E-R diagram nezobrazuje, jaké operace se budou s daty provádět. E-R diagram pro náš systém je na obrázku 1.2. Pokud máme vytvořený E-R diagram, můžeme přejít k jeho transformaci na SQL schéma. V podstatě každá entita bude převedena na jednu tabulku databáze. Každá entita by měla mít unikátní klíč (v našem diagramu označený <<PK>>), který bude primárním klíčem v každé tabulce. Zvolili jsme přístup, kdy každý záznam v tabulce bude mít svoje jedinečné ID.



Obr. 1.2: E-R diagram IS školy

2 Použité technologie

Protože bylo rozhodnuto implementovat systém jako webovou aplikaci, byly pro tvorbu systému vybrány oblíbené kombinace nástrojů jakými jsou PHP a MySQL. V této kapitole jsou popsány technologie, které byly použity pro tvorbu internetové aplikace, tedy HTML, CSS, JavaScript, PHP a MySQL.

2.1 HTML

HTML je zkratka pro HyperText Markup Language. Jedná se o tzv. značkovací jazyk pro vytváření internetových stránek. Značkovací jazyk znamená, že kromě vlastního textu obsahuje i instrukce pro jeho zpracování. Jazyk HTML je určitou větví univerzálního značkovacího jazyka SGML (Standard Generalized Markup Language). Můžeme nalézt několik verzí HTML, nejpoužívanější je v dnešní době HTML 4.01. V současnosti se přechází na jazyk XHTML, který je více striktní obdobou HTML.

2.2 CSS

CSS je soubor metod pro grafickou úpravu a formátování HTML nebo XHTML webových stránek. Zkratka CSS znamená Cascading Style Sheets, což se do češtiny překládá jako „kaskádové styly“ [3]. Konkrétně u webových stránek se formátování týká barvy a velikosti písma, pozadí, zarovnání, zobrazení tabulek atd. Hlavním smyslem CSS je oddělení popisu vzhledu dokumentu od jeho struktury a obsahu.

Problémem CSS byl nedostatek podpory ze strany některých prohlížečů. V dnešní době jsou již kaskádové styly podporovány všemi užívanými prohlížeči. Použit CSS můžeme třemi způsoby. Buď vložíme přímo do textu zdroje formátovaného elementu atribut „style“ nebo sepíšeme seznam stylů do hlavičky stránky mezi tagy `<style>` a `</style>`. Třetí možností je vytvoření souboru *.css, který obsahuje seznam stylů a na který se v HTML odkazujeme tagem `<link>`.

2.3 PHP

Jazyk PHP je interpretovaný skriptovací jazyk, který se používá pro tvorbu webových stránek. PHP pracuje na straně serveru a dovede ukládat a měnit data stránek zobrazovaných v prohlížeči. Původně zkratka PHP byla pro Personal Home Page, nyní znamená PHP: Hypertext Preprocessor [1]. Za vznikem PHP stojí Rasmus Lerdorf, který položil základy PHP v roce 1996, když v jazyce PERL vytvořil sadu skriptů pod názvem PHP/FI [2]. Od této doby prošel jazyk PHP velkými změnami. Se vzrůstající známostí jazyka se zapojovalo do vývoje jazyka PHP více programátorů a do současné podoby se jazyk dostal díky rozsáhle komunitě lidí, kteří se na jeho vývoji podílejí. V roce 2000 vyšla verze PHP 4 a později v roce 2004 PHP 5. Tyto patří v současné době mezi nejpoužívanější. Syntaxe jazyka je podobná do jazyka C.

Jazyk PHP je poměrně nezávislý na platformě, funguje na nejrůznějších operačních systémech, umožňuje přístup k většině databází a podporuje řadu internetových protokolů. PHP se nejčastěji začleňuje do struktury HTML nebo XHTML kódu. Oproti HTML nabízí PHP lepší dynamiku. Proto také s tvorbou stránek pomocí PHP je spojen výraz „dynamická tvorba internetových stránek“. Jazyk JavaScript tuto potřebnou dynamiku má, ale pracuje na straně prohlížeče (klienta), a jako takový nemůže pracovat s daty v databázi a neumožňuje potřebné zabezpečení. Jazyk PHP není těžký na pochopení a jednoduché stránky lze tvořit pomocí základních znalostí. PHP umožňuje veškerou práci s daty (ukládání, změny, mazání). Pracuje tak, že nejprve provede skript na straně serveru a potom výsledek odešle prohlížeči.

Často používanou kombinací je PHP ve spojení s databázovým systémem MySQL. Díky častému použití na serverech vznikla kombinace použití vyjádřená zkratkou LAMP, tzn. Linux jako operační systém, Apache jako webový server, MySQL pro databázový systém a PHP jako programovací jazyk. Výhodou těchto nástrojů je, že jsou k dispozici včetně zdrojových kódů s možností užívání, modifikace a distribuce. Podrobnější informace jsou k nalezení na [4].

2.4 MySQL

Jedná se o relační databázový systém, který byl vytvořen firmou MySQL AB. Každá databáze je tvořena jednou nebo více tabulkami, které obsahují jednotlivé záznamy. Komunikace s MySQL databází probíhá pomocí jazyka SQL. Výhodou MySQL je rychlost, jednoduchost a vhodnost pro menší projekty. MySQL pracuje jako samostatný server. K práci s daty se užívá skriptování pomocí PHP nebo ASP. Jako podpůrný nástroj pro MySQL se často používá aplikace phpMyAdmin. Více informací k nalezení na [5].

2.5 JavaScript

JavaScript je skriptovací objektově orientovaný jazyk, za jehož vznikem stojí firma Netscape. Jazyk se původně nazýval LiveScript, později byl nejspíše z marketingových důvodů přejmenován na JavaScript. Microsoft užívá označení JScript. JavaScript se užívá pro tvorbu dynamických webových stránek a pracuje na straně prohlížeče (klienta). Jedním z hlavních rozdílů mezi PHP a JavaScriptem je, že PHP pracuje narozdíl od JavaScriptu na straně serveru. JavaScript má výhodu, že umožňuje funkce, které by nebylo možné řešit na serveru pomocí PHP. JavaScript se vkládá stejně jako PHP do struktury HTML.

Jazyk JavaScript často bývá zaměňován s Javou, což je jiný samostatný programovací jazyk, který nemá s JavaScriptem kromě podobné syntaxe moc společného. JavaScript se používá k oživení webu pomocí nejrůznějších prvků např. hodin, blikajících textů, atd. Při použití JavaScriptu při tvorbě internetových stránek je potřeba zajištění funkčnosti stránek i při zakázaném skriptování v prohlížeči, protože ne všichni uživatelé ho mají povolené.

3 Implementace

Tato kapitola se bude věnovat nejdůležitější části projektu - samotné implementaci systému. Jsou zde popsány jednotlivé části informačního systému a diskutovány poznatky a problémy, které vznikly při tvorbě IS. Důležitou je podkapitola věnovaná řešení problému automatické tvorby rozvrhů. Poslední dvě podkapitoly jsou věnovány zabezpečení systému a diskusi o implementačních problémech a dosažených výsledcích.

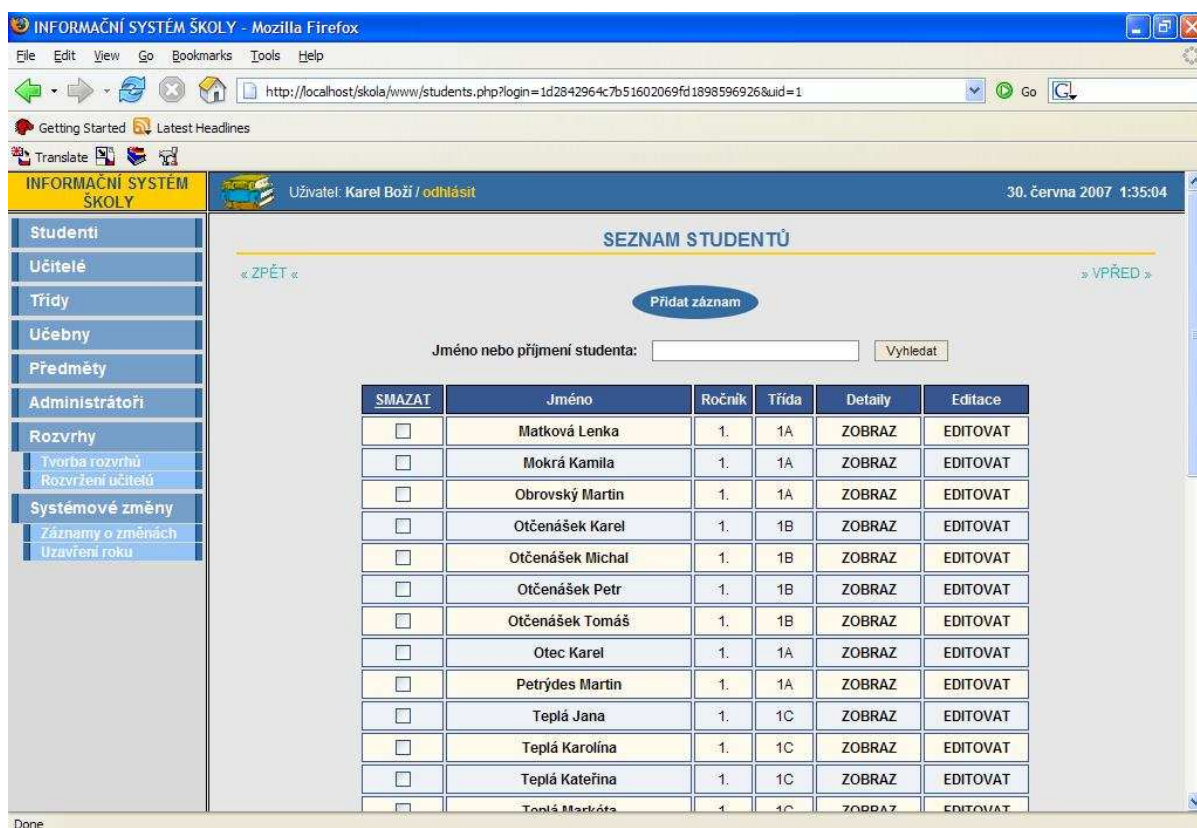
3.1 Uživatelské rozhraní

Jedná se o pojem, který popisuje, jak bude systém uživatelsky přívětivý a jakým způsobem bude s uživatelem komunikovat. Webová stránka by měla být přehledná a členěna do určitých bloků, jako je například logo, lišta, menu, hlavní okno s obsahem. Měla by být čitelná a pro uživatele jednoduchá při vyhledávání potřebné informace. Webová aplikace musí také obsahovat vhodné ovládací prvky pro dohledání a změnu informací na stránce zobrazených. K snadnému dohledání potřebné informace slouží například dobře strukturované menu.

3.1.1 Členění systému

Vzhled webové stránky je na obr. 1.3. Jak je z něho vidět, stránka je rozdělena do tří hlavních částí. První částí je horní lišta s datem a aktuálním časem, jménem přihlášeného uživatele a odkazem pro odhlášení ze systému. V levé části je menu, které strukturuje výběr jednotlivých částí systému. Třetí největší část je hlavní okno, které zobrazuje požadovaný obsah. Menu je rozděleno do těchto částí:

- Studenti – správa studentů
- Učitelé – správa vyučujících
- Třídy – seznam a editace tříd
- Učebny – seznam učeben
- Předměty – správa předmětů
- Administrátoři – správa administrátorů
- Rozvrhy
 - Tvorba rozvrhů – vytváření rozvrhů, náhled na rozvrhy, tisk rozvrhů
 - Rozvržení učitelů – přidělení výuky učitelům
- Systémové změny
 - Záznamy o změnách – seznam zásadních změn provedených v systému
 - Ukončení roku – uzavření školního roku a převedení systému do dalšího roku



Obr. 3.1: Vzhled informačního systému

Při přihlášení do části systému určené pro učitele a studenty je možné si po zadání správného současného hesla změnit přístupové heslo na heslo nové.

3.1.2 Ovládání

Během implementace systému jsme se snažili dodržet pravidla pro přístupnost uvedená ve stejné nazvané části tohoto dokumentu. Ovládání informačního systému by mělo být pro uživatele intuitivní a po krátkém seznámení s jeho funkcí jednoduché na ovládání. Manipulace se systémem je zajištěna jako u většiny webových aplikací pomocí tlačítek a hypertextových odkazů. Jednotlivé části systému je možné zobrazit v hlavním okně pomocí strukturovaného menu. Na každé stránce je odkaz na předchozí a další stránku. Jednotlivé změny v systému je možné ovládat pomocí tlačítek a odkazů v hlavním okně. Výpisy záznamů jsou rozděleny po určitém počtu na jednu stránku a je možné mezi stránkami přepínat pomocí odkazů pod tabulkou se záznamy. Úspěšné i neúspěšné změny jsou uživateli oznámeny výpisem upozorňovací hlášky. Odhlášení ze systému je zajištěno pomocí odkazu umístěného na horní liště.

3.2 Struktura školy

Strukturou školy je myšleno, na jaké části je škola rozdělena a z čeho se skládá. Každá škola má určitý počet studentů, učitelů a má nějakého ředitele školy. Učitelé učí určité předměty, studenti jsou rozděleni do ročníků a do tříd. Ve škole je určitý počet učeben a jiných místností, ve kterých můžou například probíhat jen specifické předměty. Všechny tyto informace by měly být také obsaženy v informačním systému školy. Jak toto bylo implementováno je popsáno dále.

3.2.1 Uživatelé systému

Uživatele systému můžeme rozdělit do tří skupin na studenty, učitele a administrátory. Prozatím byla implementována převážně pouze administrační část systému. Učitelé a studenti se mohou do systému přihlásit, kde si mohou změnit svoje přístupové heslo. Za administrátory považujeme ředitele školy a jiné administrační pracovníky, kteří mají právo měnit v systémů údaje. Každý uživatel má svůj jedinečný login a heslo pro přístup do IS. Pro studenty a učitele je login sedmimístný. Je automaticky vytvořen při vytváření nového záznamu o studentovi nebo učiteli. Pro studenty začíná písmenem „s“, pro učitele písmenem „u“. Pak následuje prvních pět znaků příjmení bez diakritiky. V případě délky příjmení menší než pět znaků, je řetězec doplněn na pět znaků písmenem „a“. Následuje dvojčíslí, které zajišťuje unikátnost v případě více uživatelů stejného příjmení. Při vytvoření nového záznamu je pro heslo použito rodné číslo uživatele a je doporučena jeho následná změna v systému po prvním přihlášení. Heslo musí být délky od pěti do dvaceti znaků.

3.2.2 Evidence uživatelů

Administrátoři systému po přihlášení do systému mají možnost z nabídky v menu si vybrat odkaz pro evidenci studentů, učitelů a administrátorů. Po výběru dané položky se vypíše seznam všech uživatelů dané skupiny s nejdůležitějšími detaily a možností vyhledávat jednotlivé uživatele podle části jména nebo příjmení. Je možné přidávat, editovat a mazat uživatele dané skupiny. U správy evidence administrátorů není možné vymazat všechny administrátory. Vždy musí zůstat alespoň jeden, který má přístup do systému. U učitelů není možné mazat učitele, kteří v daném roce vyučují nějaký předmět. Prvně je potřeba učitele nahradit jiným a až poté je možné jeho vymazání. Jinak by docházelo k tomu, že by některé předměty neměly vyučujícího apod.

Seznam uživatelů obsahuje další odkazy, které jsou provázané s dalšími stránkami. Při kliknutí na jméno uživatele se zobrazí stránka s fotkou a výpisem detailů o uživateli. Fotky uživatelů musí být nahrány na server, kde název souboru s fotkou musí být přihlašovací jméno uživatele. Většinou se tak děje jednou za rok, při shromáždění fotek nových studentů školy.

Základní detaily pro všechny tři typy uživatelů jsou stejné. Je to jméno, příjmení, adresa, kontaktní telefon, email, rodné číslo a login. Dále u učitelů číslo kabinetu, telefonu do kabinetu, u

studentů jméno matky a otce, třídu do které je student zařazen. Emailová adresa je automaticky vytvářena při vložení nového záznamu z přihlašovacího jména. Pro každou konkrétní školu je potřeba některé věci pozměnit, aby systém bylo možné použít pro danou školu. Například vytváření emailových adres je nesmyslné v případě, že škola nemá emailový server. V takovém případě si každý nastaví jako emailový kontakt svoji soukromou adresu.

Vstupní detaily pro vytváření záznamů jsou kontrolovány. Například telefon nebo rodné číslo jsou zkontrolovány, zda mají správný tvar a neobsahují nepovolené znaky. V případě, že není zadán některý povinný údaj, též se zobrazí varovné hlášení pro zadání tohoto údaje.

Při zapomenutí hesla administrátoři mohou heslo studentovi nebo učiteli změnit na rodné číslo. Daný student nebo učitel si ho poté po přihlášení za pomoci rodného čísla změní z důvodů bezpečnosti.

3.2.3 Třídy, učebny a předměty

Informační systém dále obsahuje záznamy o třídách, učebnách a předmětech. Obsluha je obdobná jako u evidence uživatelů. Také je možné přidávat, editovat a mazat záznamy. Ze systému není možné vymazat předmět, který je učen, třídu, ve které studují studenti, nebo místnost, ve které probíhá výuka. Pokus o vymazání takovéto položky skončí chybovou hláškou. Nejprve je potřeba udělat v systému takové změny, aby nedošlo k nesprávnému fungování systému a až poté je možné položku vymazat.

U každého předmětu jsou požadovány detaily o zkratce, názvu, počtu hodin výuky týdně a ročník ve kterém je předmět učen. V případě, že se jedná o speciální předmět jako jsou například laboratoře nebo tělesná výchova, která může probíhat pouze v tělocvičně, je určena konkrétní místnost, ve které může takovýto speciální předmět probíhat. Detaily, jako místnost výuky a počet hodin výuky předmětu týdně, jsou velmi důležité pro pozdější tvorbu rozvrhů. Dále každý předmět obsahuje anotaci, program výuky, způsob hodnocení, výpis doporučené literatury a případný nutný komentář k předmětu. Také je zadáno, který učitel je garantem předmětu a odpovídá za daný předmět. Při rozšíření systému by měl tento garant mít možnost editovat detaily o předmětu.

Seznam tříd není nijak problematický. Obsahuje názvy tříd s určením, která třída studuje který ročník a kdo je jejím třídním učitelem. Je ošetřena nemožnost zadání dvou tříd stejného názvu, přidělování do neexistujících ročníků studia apod.

U místnosti je důležitý název a především údaj o tom, jestli v učebně může být učen jakýkoliv předmět nebo jestli je určena pro některý speciální předmět například již výše zmíněnou tělesnou výchovu. Toto je detail, který je také velmi důležitý pro tvorbu rozvrhů.

3.3 Tvorba rozvrhů

Jednou z nejdůležitějších částí implementace IS školy bylo vyřešení problému automatického generování rozvrhů pro všechny třídy. V této kapitole bude tento problém podrobně rozebrán. Systém umožňuje jak automatickou tvorbu rozvrhů, tak i manuální, která slouží spíše k doladění některých detailů po automatickém vytvoření rozvrhů. Například pokud si učitel přeje přehodit výuku s jiným učitelem apod. K tomu, aby bylo možné rozvrhy tvořit, je potřeba mít v systému uložen dostatek informací. Potřebujeme znát, jaké předměty jsou vyučovány pro který ročník, kolik hodin týdně je daný předmět učen, který učitel vyučuje daný předmět, kterou třídu, ve kterých učebnách může být daný předmět vyučován apod.

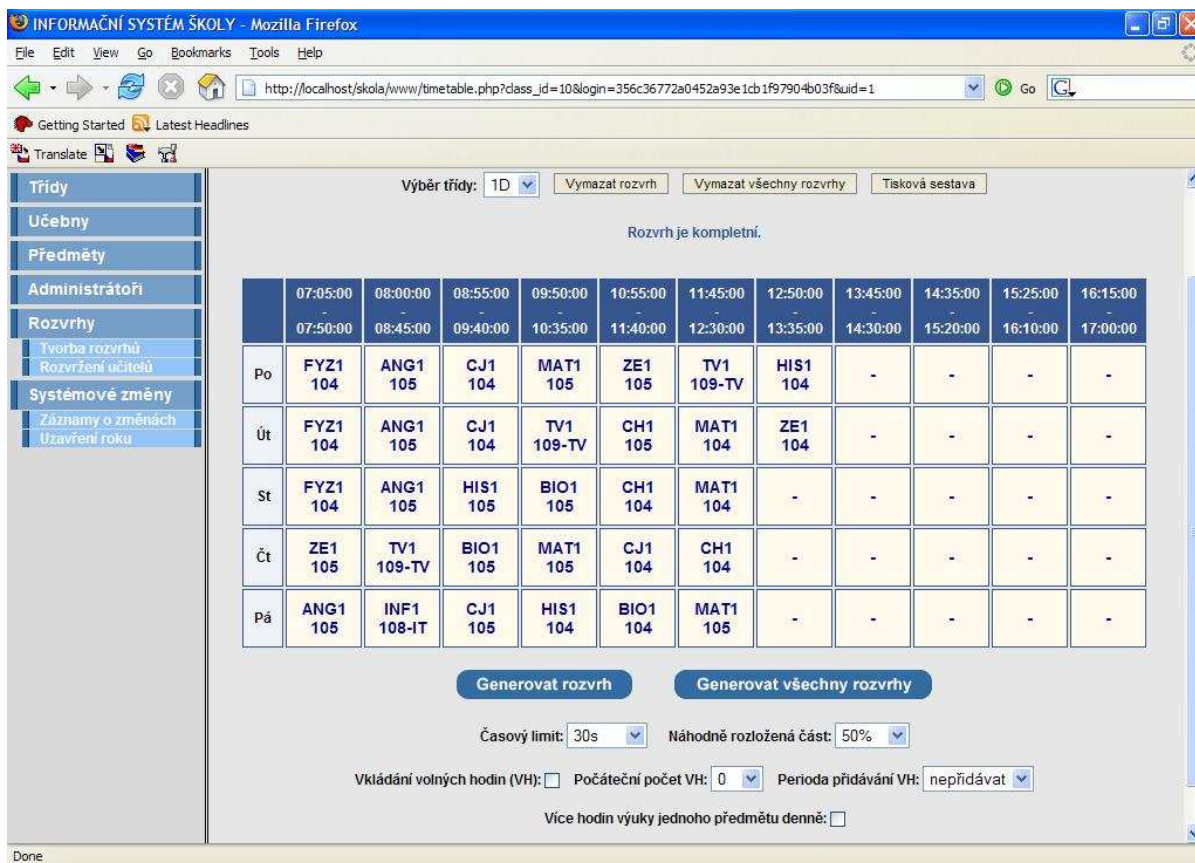
Při vytváření rozvrhů bylo třeba dbát na ošetření kolizí a další omezení. Nejzávažnějším omezením je, že jeden učitel nemůže učit zároveň dvě třídy ve stejnou dobu, stejně jako v jedné učebně nemůže ve stejnou dobu probíhat výuka více tříd. Na tyto kolize je potřeba dávat pozor a dostatečně toto ošetřit.

Ukázka stránky pro tvoření rozvrhů je na obrázku 3.2. Rozvrh je zobrazen v podobě tabulky. Horní řádek tabulky obsahuje časy od kdy a do kdy je daná vyučovací hodina učena. Při kliknutí na jedno z těchto políček se zobrazí možnost změny času, kdy hodina začíná a končí. Při zadání nekorektního času, je uživatel na toto upozorněn varovným hlášením.

Další řádky obsahují okýnka pro přiřazení vyučovacích hodin pro jednotlivé dny. Při kliknutí na jedno z políček je možné vybrat předmět, který bude v danou dobu učen, a přiřadit ho do některé místnosti. Po kliknutí na tlačítko „uložit“ se vloží vyučovací hodina do rozvrhu. V případě, že se snažíme vložit do výukového týdne více vyučovacích hodin předmětu než je povolený limit, zobrazí se varovné hlášení. V nabídce se nezobrazí předměty, které učí danou třídu učitel, který v tu samou dobu učí již jinou třídu. Stejně tak se nezobrazí v nabídce místnosti, ve kterých v danou dobu v daný den již probíhá výuka. Odstranění vyučovacích hodin se provede kliknutím na jedno políčko rozvrhu a výběrem žádný předmět a žádná místnost z nabídky. Po kliknutí na tlačítko „uložit“ se vyučovací hodina vymaže.

Každá třída má určitý celkový počet vyučovacích hodin týdně. V případě, že je rozvrh kompletní, je nad rozvrhem toto napsáno. Pokud tomu tak není, jsou vypsány předměty, které je ještě potřeba do rozvrhu doplnit, aby se stal kompletním.

Výběrem třídy z nabídky je možné zobrazit rozvrh pro tuto třídu. Tlačítkem „Vymazat rozvrh“ se vymažou všechny vyučovací hodiny daného rozvrhu. Tlačítkem „Vymazat všechny rozvrhy“ se vymažou všechny rozvrhy pro všechny třídy. Kliknutím na tlačítko „Tisková sestava“ se otevře okno s černobílou sestavou pro tisk rozvrhu dané třídy.



Obr. 3.2: Vzhled stránky pro tvorbu rozvrhů

3.3.1 Přidělování učitelů

Jednou z důležitých věcí, která je nutná pro tvorbu rozvrhů, je znalost toho, který učitel učí který předmět kterou třídu. U běžné školy existuje to, že ten samý předmět učí jednu třídu jiný učitel než druhou. Toto přidělení učitelů třídám pro jednotlivé předměty je možné realizovat po kliknutí na odkaz v menu „Rozvržení učitelů“. Při vytvoření nové třídy je pro tuto třídu u každého předmětu přiřazen automaticky jako vyučující garant předmětu. Děje se tak proto, aby nedošlo k tomu, že některou třídu nebude nikdo učit. Na stránce pro přidělování vyučujících je poté potřeba přidělit této nové třídě rozumě vhodného vyučujícího. Je počítáno s tím, že administrátor, který bude přidělovat jednotlivé vyučující, má představu o tom, jak je který učitel časově vytížen. V případě, že by některý učitel vyučoval více hodin než je celkový týdenní rozsah rozvrhu, došlo by k problému při vytváření rozvrhu. Nemluvě o tom, že asi žádný učitel nebude ochotný pracovat například dvanáct hodin každý den.

3.3.2 Automatická tvorba rozvrhů

Nejzajímavější částí tvorby rozvrhů a nejnáročnější na implementaci je vyřešení automatického generování rozvrhů. K spuštění generování slouží tlačítko „Generovat rozvrh“ pro vygenerování jednoho rozvrhu a tlačítko „Generovat všechny rozvrhy“ pro vytvoření rozvrhů pro všechny třídy. Je možné zvolit časový limit, po jehož vypršení bude tvorba ukončena v případě, že do tohoto limitu nebylo nalezeno řešení.

3.3.2.1 Backtracking algoritmus

Jednou z otázek, jak řešit daný problém, byl výběr vhodného algoritmu pro generování rozvrhů. Nabízela se možnost použití genetických algoritmů, ale implementace se všemi omezeními by byla velice náročná. Proto byl použit algoritmus zvaný backtracking. Je to zpětné vyhledávání založené na metodě „pokus-omyl“ [7]. Jedná se o procházení stromovou strukturou do hloubky s návraty, kde v našem případě každý uzel je reprezentován jednou vyučovací hodinou. V případě nenalezení řešení pro některý z uzlů, vracíme se k rodičovskému uzlu a hledáme jiné možné řešení. Metoda je vhodná pro řešení problémů, kde je potřeba prozkoumat velké množství řešení, jako je tomu v našem případě tvorby rozvrhů. Toto je základní podoba backtracking algoritmu. Existují nejrůznější vylepšení, pomocí nichž je možné zefektivnit vyhledávání řešení. Některé z nich budou dále popsána.

3.3.2.2 Uspořádání vstupů

Jedním s důležitých předpokladů pro úspěšnou tvorbu rozvrhů je uspořádání vstupních dat, což ovlivňuje jak čas potřebný na tvorbu rozvrhu, tak i úspěšnost nalezení řešení v reálném čase. Možností, jak přidělit každé třídě jednotlivé vyučovací hodiny, je mnoho, a proto je třeba se zabývat tím, které předměty přidělit nejdříve. V případě, že by předměty problematické na přidělení byly nechány na konec, algoritmus by se musel vracet často z nižších úrovní stromu do vyšších úrovní bližších kořenu, což by bylo časově náročné.

Problematické předměty na přidělení jsou ty, které mohou být vyučovány pouze v jedné místnosti například speciální laboratoři pro tento předmět. Proto jsou tyto předměty přiděleny nejdříve a probíhají v ranních hodinách. Toto je vhodné také z důvodu, že během výuky těchto předmětů je potřebná dobrá koncentrace, která bývá u studentů především po ránu. Toto se netýká tělocviku, který je zařazován později do vyučovacího dne v případě, že je to možné.

Jedním z dalších základních požadavků na rozvrh je, aby každý předmět byl vyučován nejvýše jednou denně. V případě, že například máme pět vyučovacích dnů a předmět, který je vyučován celkově sedm hodin týdně, snažíme se tento předmět zařadit denně nanejvýše dvakrát.

Pro to, aby bylo snadné přiřadit takto předměty, je potřeba začít přiřazovat nejdříve ty předměty, které jsou týdně vyučovány nejčastěji. Umožní to, že bude snadné takovéto předměty přidělit tak, aby každý den byl vyučován jen jednou a zabrání se tak zdlouhavému vracení ve stromu za účelem vyhovění tomuto omezení.

Vzniká ale jiný problém a to, že v případě přiřazování předmětů postupně od počtem výuky týdně nejpočetnějších k nejméně početným po jednotlivých předmětech může docházet k problémům s přidělováním posledních předmětů rozvrhu na nejnižších úrovních stromu. K tomuto dochází jen v případě, pokud konkrétní předmět učí všechny třídy stejný učitel. Uvedeme si toto na příkladě, kdy předmět s nejmenším počtem vyučovacích hodin týdně je vyučován třikrát týdně. Pro první třídu nebude problém nalézt řešení a předmět bude umístěn na poslední tři políčka v rozvrhu. Pro druhou třídu se algoritmus bude muset již vracet a umístí výuku tohoto předmětu na tři políčka před posledními třemi vyučovacími políčky. Pro pátou třídu by toto již znamenalo, že by se algoritmus musel vrátit o dvanáct úrovní ve stromu, aby byl schopný umístit tento předmět, což s dodržением ostatních omezení značně prodlužuje dobu výpočtu. Řešením je, že část rozvrhu bude vytvořena ne postupným přidělováním všech hodin předmětů, ale s náhodným přidělováním jednotlivých hodin výuky předmětu. Tedy pro část rozvrhu nebude brán v potaz celkový počet vyučovacích hodin přidělovaných předmětů jako celek ale každá vyučovací hodina předmětu jednotlivě. Určení, jaká část rozvrhu má být náhodně vytvářena, je umožněno nastavením počtu procent části rozvrhu vytvořené takto náhodně.

Pro každou úroveň stromu je tedy dáno jiné pořadí přidělování předmětů. První část určená částí, která není náhodně rozdělená, bude pro každou vrstvu stejná. Na začátek rozvrhu se budeme snažit umístit předměty, na které je kladen požadavek na výuku pouze ve speciální místnosti. Zbytek této části, kde nejsou předměty přidělovány náhodně bude rozdělen tak, že se budeme snažit přidělit nejdříve předměty s největším počtem výuky týdně. Tím dosáhneme snadného dodržení požadavku na minimální počet vyučovacích hodin daného předmětu denně. Dále bude následovat náhodně přidělovaná část rozvrhu, kdy je každá hodina předmětu brána jednotlivě a je pro každou vrstvu vygenerováno náhodně pořadí posloupnosti přidělování těchto předmětů. Toto zamezí výše zmíněnému problému a díky jinému pořadí pro každou vrstvu rozvrhu se sníží pravděpodobnost, že přiřazovaný předmět posloupnosti již byl přiřazen v plném počtu hodin.

3.3.2.3 Kompletní algoritmus

Algoritmus pro každou úroveň stromu má určité pořadí předmětů, podle kterého je přiděluje. Kontrolujeme, zda vyučující přidělovaného předmětu již v danou dobu neučí jinou třídu a také zda v přidělované učebně neprobíhá již výuka jiné třídy v tu samou dobu. Také je kontrolováno, v jaké místnosti může být předmět učen a podle toho jsou předměty přidělovány do učeben. Algoritmus tedy zkouší vkládat předměty podle dané posloupnosti. Pokud nemůže přidělit nějaký předmět z důvodu kolize, zkouší přidělit další předmět této posloupnosti. V případě, že se nepodaří přidělit žádný předmět, vrací se o úroveň výše k rodičovskému uzlu a zkouší další možnosti přidělení předmětů, dokud nedospěje k uspokojující variantě přidělení předmětů. Algoritmus si pamatuje, které předměty již zkusil pro dané políčko neúspěšně přidělit, a při dalším pokusu na tomto místě již je přidělit nezkouší.

Na vygenerovaný rozvrh je kladen požadavek, že pokud je to možné, ve vyučovacím dnu nebudou žádné volné hodiny. Může ale dojít k tomu, že pro danou hodinu v určitý den není možné přidělit žádnou kombinaci předmětu a místnosti. V takovém případě není jiné východisko než volnou hodinu vložit.

Generování rozvrhů dále nabízí určitá další nastavení, která umožňují rychlejší nalezení řešení pro komplikovanější případy. Například když máme větší počet tříd než počet dostupných místností pro výuku, nastává problém, že řešení rozvrhů pro všechny třídy zabere více času. Generování rozvrhů můžeme pro tyto komplikovanější případy ulehčit odstraněním některých omezení. Je možné zvolit zrušení požadavku na to, aby každý předmět byl denně vyučován co nejméně. Dále je možné nastavit počáteční počet volných hodin, které budou do rozvrhu automaticky zařazeny. Tím se urychlí tvorba, protože na místo vyučovacích hodin, kam je problematické vložit nějakou výuku, bude vložena volná hodina. Dále je možné určit periodu, po které bude v průběhu tvorby zařazována do rozvrhu další volná vyučovací hodina.

Když se nepodaří vytvořit předměty v daném časovém limitu, je tvorba rozvrhů ukončena varovným hlášením, že rozvrh se nepodařilo v tomto časovém limitu kompletně vytvořit. Část rozvrhu, která byla do té doby vytvořena, je uložena a nabízí možnost doplnění rozvrhu manuálně. V případě nenalezení řešení při kompletním projití všech možností s omezeními se spustí algoritmus ještě jednou. Tentokrát se ale nesnaží o nejnižší počet vyučovacích hodin předmětu denně a také po určité časové periodě vkládá do rozvrhu volné hodiny. Toto zrychluje výpočet a zvyšuje pravděpodobnost nalezení řešení pro komplikované případy. Tomuto druhému průchodu bez omezení je možné předejít nastavením zrušení omezení minimálního počtu výuky předmětu denně a povolením přidávání volných hodin. Tedy již při prvním průchodu bude rozvrh vytvářen s méně omezeními jako při druhém průchodu.

3.4 Systémové změny

Systém uchovává některé vnitřní změny, které jsou důležité pro správný chod systému. V případě, že někdo změní rozvrh, heslo nebo vymaže záznam, uloží se informace o tom, kdo, kdy a jakou změnu provedl. Toto je z důvodu zpětného zjištění, co se v systému stalo, v případě jakýchkoliv nesrovnalostí. Dále také z důvodů bezpečnosti. V případě zavedení systému hodnocení studentů půjde dopátrat, kdo a kdy změnil studentovi známku v systému atd.

3.4.1 Ukončení školního roku

Jedna ze záležitostí, která musí být vyřešena, je ukončování školního roku. Stránka pro ukončování školního roku nabízí výběr, která třída bude přemístěna do které třídy vyššího ročníku. Po kliknutí na tlačítko ukončit školní rok se studenti aktuálních tříd přesunou do vyšších ročníků. Studenty posledního ročníku již není možné v systému dohledat, ale jejich záznamy zůstávají v databázi. V případě rozšíření systému o část, která bude studenty hodnotit, bude potřeba u již nestudujících studentů uchovávat všechny záznamy pro pozdější zjištění známek a informace za jaký předmět byly známky obdrženy. Při ukončení jednoho školního roku a započítí jiného je potřeba do systému vložit údaje o nových studentech prvního ročníku a rozdělit je do odpovídajících tříd.

3.5 Bezpečnost

Základním bezpečnostním prvkem pro přístup do systému je bezpečné přihlášení. Jak je vytvářen login, bylo již vysvětleno výše v této kapitole. Login a heslo je ověřeno na shodnost s údaji uloženými v databázi. Pokud se údaje shodují, je zjištěno, jaké práva má konkrétní uživatel, a proběhne přihlášení do systému.

Při přihlášení do systému je možné se odhlásit pomocí odkazu na horní liště. Po odhlášení není již možné se zpětně dostat do systému bez opětovného zadání uživatelského jména a hesla. V případě, že dojde k nečinnosti systému delší než deset minut, je uživatel automaticky ze systému odhlášen.

Jedním z bezpečnostních problémů je uchování hesel uložených jako prostý text v databázi. V případě, že by někdo získal přístup do databáze, získá tak i všechna hesla uživatelů. V dnešní době je používání bezpečnostních hesel velmi rozšířené, a proto je také těžké si veškerá hesla pamatovat. Lidé používají často stejná hesla pro více zabezpečovacích systémů, takže při získání hesla z databáze existuje riziko získání přístupu i do jiných systémů pomocí použití tohoto hesla. Je třeba zmínit, že k heslům má přístup i administrátor nebo správce serveru, což také není úplně v pořádku. Tento problém byl vyřešen uložením *hashe* hesla. *Hashování* je jednosměrný proces, proto při přihlašování je provedeno vytvoření otisku zadaného hesla a porovnání získaného otisku s otiskem uloženým v databázi.

V poslední řadě přihlášení pro administrátory systému probíhá z jiné stránky než pro studenty. Při neznalosti této skutečnosti toto zabraňuje například uhodnutí přístupového hesla při náhodném zkoušení zadávání hesla, kdy máme k dispozici login administrátora.

3.6 Zhodnocení implementační části

V průběhu implementace bylo nutné vyřešit velké množství problémů. Ukázalo se, že vytvořit dobře fungující informační systém školy není lehká záležitost již vzhledem k velké provázanosti dat. Aby byl systém udržen ve fungujícím konzistentním stavu, bylo nutné zavést několik desítek omezení.

Mezi nejzajímavější implementační problémy patří vyřešení zajištění ukončování školního roku a určení, který učitel bude učit kterou třídu a který předmět. Největším implementačním problémem byla možnost zajištění automatického generování rozvrhů. V podstatě celý systém musel být navrhnout tak, aby toto bylo možné, a následná implementace byla směřována k tomuto cíli. Během implementace generování rozvrhů bylo nutné odstranit mnoho chyb a vyladit algoritmus tak, aby fungoval správně a co nejefektivněji. Tato implementační část byla časově nejnáročnější.

V případě nasazení systému v praxi by bylo určitě za potřebí pozměnit některé vlastnosti systému tak, aby konkrétní škole vyhovoval. Je tomu tak, protože škola od školy se liší v některých detailech. Můžeme ale říci, že systém byl navrhnout a implementován tak, aby vyhověl základním požadavkům většiny škol.

Pokud bychom chtěli porovnat vytvořený informační systém s již existujícími systémy, musíme říct, že náš systém není tak komplexní jako jiné systémy na trhu. Z nedostatku času a lidských zdrojů byla implementována převážně pouze administrační část systému. Aby byl systém komplexní a konkurence schopný jiným informačním systémům škol, je nutné ho rozšířit o další části. Možnosti rozšíření systému jsou diskutovány v následující kapitole.

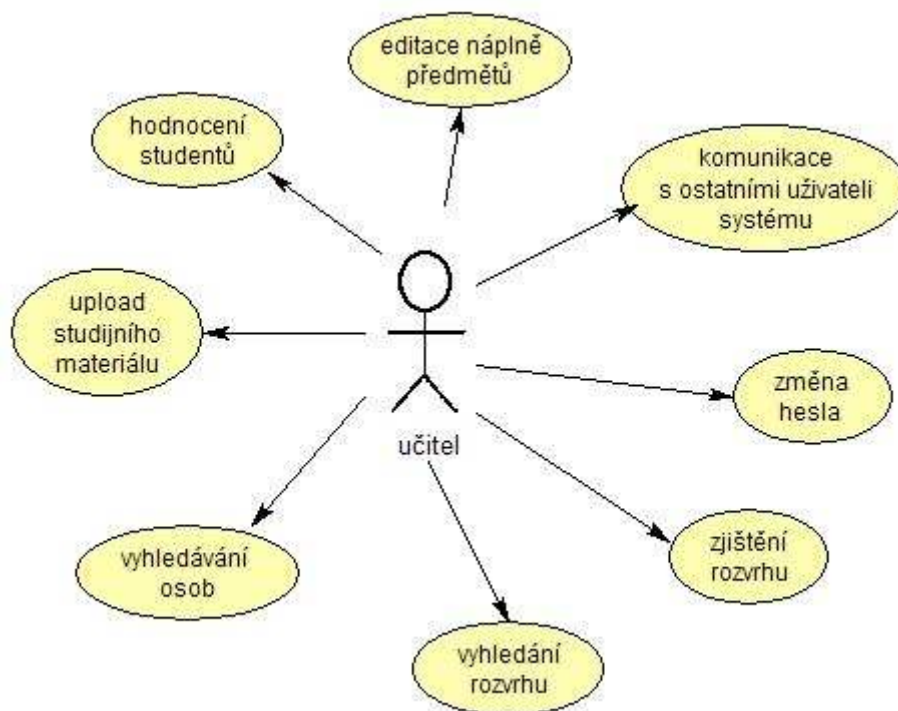
4 Možná rozšíření systému

Systém byl nejdříve navrhnout jako více komplexní, nežli je jeho výsledná podoba. Ustoupení od původního návrhu nastalo ze zjištění, že pro kvalitní implementaci takového systému by bylo potřeba více času. Díky původnímu návrhu má systém výhodu, že je snadné ho rozšířit o další moduly. Můžeme říct, že se s jeho rozšířením počítá, aby byl vytvořen plnohodnotný informační systém školy.

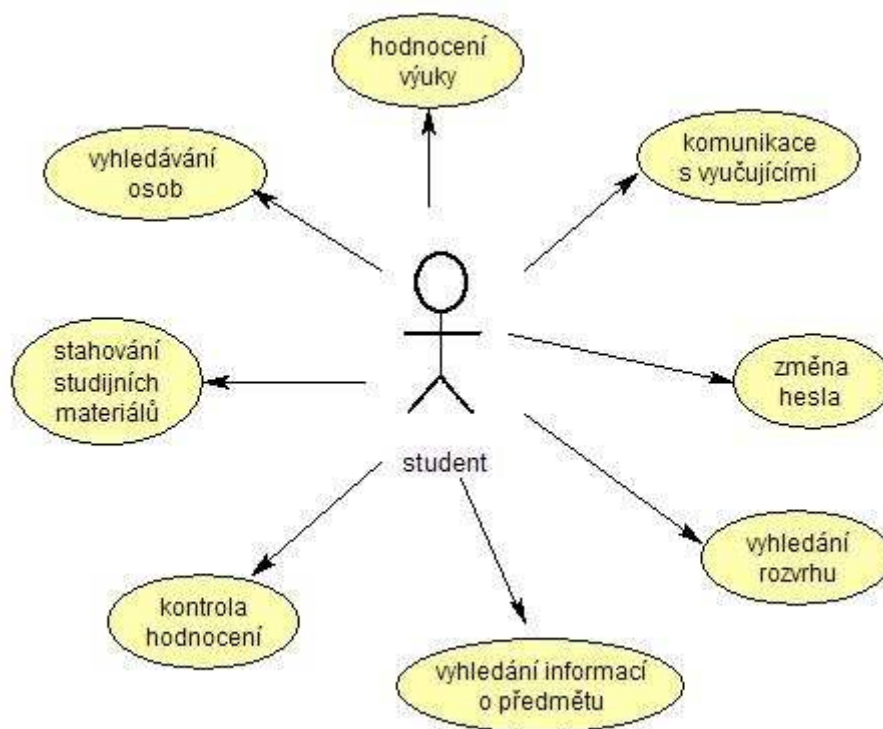
Základní rozšíření, které by mělo být implementováno, je rozšíření modulu pro studenty a učitele o další funkce. Toto by nemělo být příliš problematické z hlediska toho, že vytvořená administrační část již obsahuje informace o studentech a učitelích a také jsou v ní uloženy loginy a hesla pro přístup. Studenti a učitelé se mohou přihlásit do systému, ale při současném stavu aplikace mají možnost pouze měnit svoje přístupové heslo. Veškeré informace, které by měly být zpřístupněny těmto uživatelům již byly implementovány pro administrační část systému. Stačí tedy to, co je již vytvořené, transformovat do jiné podoby a přidat omezení (rozdělení práv pro jednotlivé uživatelské role), aby například studenti neměli možnost provádět v systému zásahy, jaké může dělat administrátor.

Další užitečnou funkcí by mohlo být vytvoření modulu pro hodnocení studentů. Učitelé by měli mít možnost do IS zadávat známky, které studenti získali při zkoušení. Studenti by po přihlášení do systému poté mohli u každého předmětu nalézt, jaké známky získali včetně podrobností. Systém by měl vytvářet ze všech známek předběžné celkové hodnocení. Z výsledných známek zadaných učitelem by v pololetí nebo na konci roku systém měl vytvářet tiskové sestavy rozvrhů. Při zavedení takového systému by bylo potřeba uchovávat (archivovat) detaily hodnocení studentů i v případě, kdy student ukončí studium, tak jak tomu je v běžné papírové formě. Mohlo by dojít k situaci, kdy by bylo potřeba po několika letech dohledat známky, které student získal za ten a ten předmět. S ukládáním známek v systému bude také potřeba zvýšit jeho zabezpečení. Určitě se najde nejméně jeden student, který by rád pronikl do IS a vylepšil si získané známky za předměty.

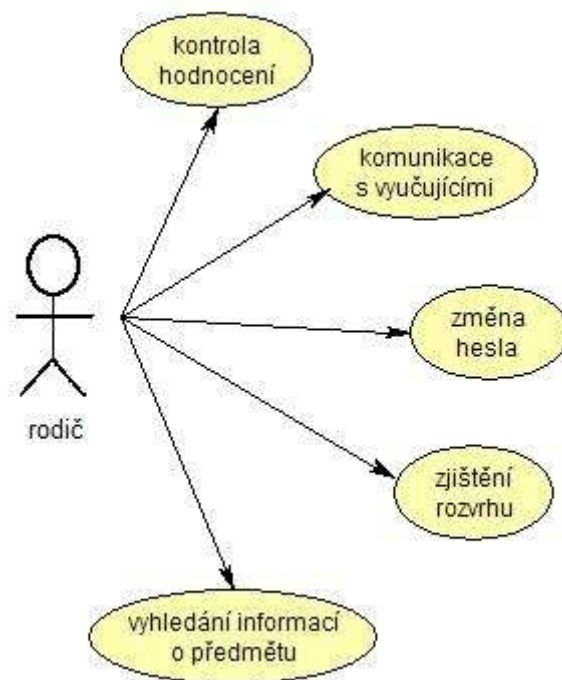
Rozšíření, které by mohlo následovat předchozí vylepšení systému, je zahrnutí rodičů mezi uživatele IS školy. Těm by byl umožněn přístup do systému stejně jako ostatním uživatelům pomocí přihlašovacího jména a hesla. Rodiče by poté mohli být schopni za pomoci systému kontrolovat studijní výsledky svých dětí komunikovat s učiteli a dohledávat v systému další informace spojené se studiem na dané škole. Tři Use Case Diagramy pro roli učitele, studenta a rodiče jsou na následujících obrázcích.



Obr. 5.1: Use Case Diagram pro roli učitel



Obr. 5.2: Use Case Diagram pro roli student



Obr. 5.3: Use Case Diagram pro roli rodič

Další možná rozšíření jsou uvedena v následujícím seznamu:

- Školní aktuality
- Jídelní lístek školní jídelny
- Zadávání, vypracovávání a hodnocení domácích úkolů pomocí IS školy
- Diskuze jak pro studenty, tak i pro učitele
- Nejrůznější statistiky
- Hodnocení kvality vyučovaných předmětů studenty
- Rozšíření podpory systému pro více jazyků
- Kalendář školních akcí
- Fotogalerie
- Články o školním dění a pracích studentů
- Elektronická omluvenka
- Komunikace mezi učiteli, žáky a rodiči pomocí webového rozhraní pro zasílání emailů

5 Závěr

Na tomto projektu jsem si prohloubil znalosti týkající se tvorby webových stránek a informačních systémů pomocí kombinace oblíbených nástrojů PHP a MySQL. Již během analýzy požadavků jsem ustoupil od myšlenky vytvořit komplexní systém, který by plně uspokojoval potřeby jak studentů, tak i všech zaměstnanců školy. Pro tvorbu takového systému nebyl dostatek času ani lidských zdrojů. Rozhodl jsem se zaměřit převážně na vytvoření nejdůležitější části systému, a to části administrativní. Tento cíl se mi podařilo implementovat a systém je možné nasadit jako podporu pro administrativní pracovníky a řízení školy. Značný čas při implementaci systému byl věnován problematice automatického generování rozvrhů. Tento problém byl vyřešen a systém je schopen vytvářet z údajů uložených v systému poměrně kvalitní rozvrhy pro všechny třídy, které by bylo obtížné vytvářet manuálně. Systém jako celek byl tvořen tak, aby bylo možné ho v budoucnu snadno rozšířit o další moduly, především pro učitele a studenty. Tato práce pro mě byla v každém případě přínosem a bylo pro mě zajímavé na ní pracovat.

Literatura

- [1] Gilmore, J., W.: Velká kniha PHP a MySQL, Zoner Press, 2007.
- [2] Kosek, J.: PHP - Tvorba interaktivních internetových aplikací. Grada, Praha, 1999.
- [3] W3C: World Wide Web Consortium [online]. Dostupný na URL: <<http://www.w3.org>>, 2007.
- [4] PHP manuál [online]. Dostupný na URL: <<http://www.php.net/manual/cs/>>, 2007.
- [5] MySQL manuál [online]. Dostupný na URL: <<http://dev.mysql.com/doc/>>, 2007.
- [6] Špinar, D.: Přístupnost [online]. Dostupný na URL: <<http://pristupnost.nawebu.cz>>, 2007.
- [7] Online encyklopedie Wikipedia [online].
Dostupný na URL: <<http://www.wikipedia.org>>, 2007.
- [8] Unified Modeling Language [online]. Dostupný na URL: <<http://uml.org>>, 2007.

Příloha 1 - Instalace systému

Doporučená konfigurace systému:

- Server Apache verze 2 nebo IIS 5
- PHP verze 4.3
- MySQL verze 4.1

Postup instalace:

- Nakopírujte zdrojové soubory z adresáře *web_aplikace* do výchozího kořenového adresáře stránek nastaveného v konfiguraci webového serveru.
- Vytvořte databázi, přístupové údaje pro připojení k databázi nastavte v souboru *global.php*, který je umístěn v */web_aplikace/include*.
- Vytvořte potřebné databázové tabulky pomocí SQL skriptu uloženého v adresáři *database*.

Příloha 2 - Obsah přiloženého CD

Adresář *database*:

- `\data.sql` - skript pro vytvoření tabulek katalogu výrobků, uživatelů a objednávek
- `\tables.sql` - skript vytvoření tabulek pro účty pracovníků

Adresář *tech_zprava*:

- `\tech_zprava.pdf` - text technické zprávy bakalářské práce

Adresář *web_aplikace*:

- `\index.php` - hlavní skript
- `\adm\index.php` - úvodní stránka po přihlášení administrátora
- `\adm\login.php` - přihlašovací stránka pro přihlášení administrátora
- `\css\layout.css` - styly pro rozvržení stránky
- `\css\styles.css` - styly pro písmo, formuláře, atd.
- `\img` - obrázky ve formátu PNG použité pro grafický design stránek
- `\include\bottom.php` - spodní část stránky
- `\include\functions.js` - JavaScript funkce
- `\include\functions.php` - funkce pro práci skriptů
- `\include\global.php` - hlavní soubor s nastavením a konstantami
- `\include\head.php` - hlavička stránky
- `\include\menu.php` - rozvržení stránky
- `\include\lib\db.php` - knihovna s funkcemi pro práci s databází
- `\include\lib\paging.php` - knihovna pro podporu stránkování
- `\upload\photos` - fotografie studentů a učitelů ve formátu JPG

<code>\www\administrator.php</code>	- výpis administrátorů
<code>\www\administrator_new.php</code>	- vytvoření a editace záznamu o administrátorovi
<code>\www\allocation.php</code>	- alokování vyučujících třídám
<code>\www\authorization.php</code>	- zajištění bezpečného přístupu do systému
<code>\www\class_students.php</code>	- výpis studentů v dané třídě
<code>\www\classes.php</code>	- výpis tříd
<code>\www\deny.php</code>	- odepření přístupu
<code>\www\end_year.php</code>	- stránka pro ukončení školního roku
<code>\www\changes.php</code>	- stránka se systémovými změnami
<code>\www\index.php</code>	- úvodní stránka
<code>\www\login.php</code>	- stránka pro bezpečné přihlášení studentů a učitelů
<code>\www\print.php</code>	- stránka s tiskovou sestavou
<code>\www\profiles.php</code>	- zobrazení profilu studenta, učitele nebo detaily předmětu
<code>\www\rooms.php</code>	- výpis místností
<code>\www\student_new.php</code>	- vytvoření a editace záznamu o studentovi
<code>\www\students.php</code>	- výpis studentů
<code>\www\subject_new.php</code>	- vytvoření a editace záznamu o předmětu
<code>\www\subjects.php</code>	- výpis předmětů
<code>\www\teacher_new.php</code>	- vytvoření a editace záznamu o učitelích
<code>\www\teachers.php</code>	- výpis učitelů
<code>\www\timetable.php</code>	- tvorba rozvrhů
<code>\www\tt_creation.php</code>	- hlavní funkce pro vytvoření rozvrhu

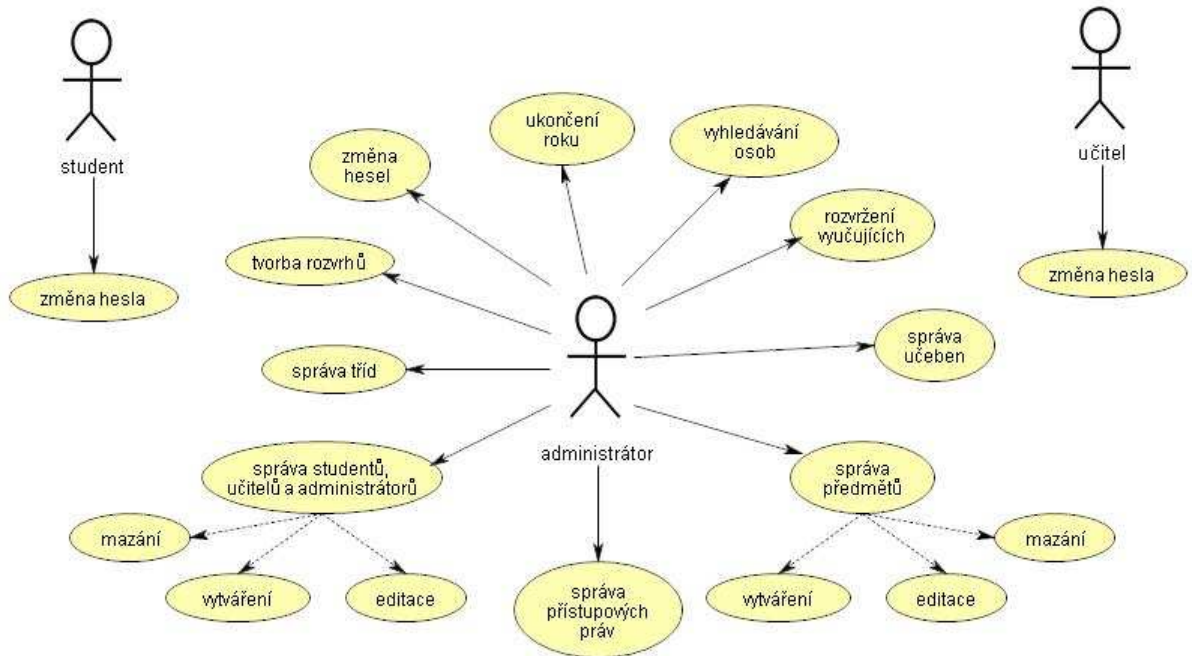
Soubor *popis_cd.pdf*:

- dokument s popisem struktury CD

Soubor *readme.txt*:

- dokument s doporučenou konfigurací a postupem jak nainstalovat systém

Příloha 3 - Use case model



Příloha 4 - E-R diagram

