

Univerzita Pardubice  
Fakulta elektrotechniky a informatiky

Redakční systém pro základní školu  
Jakub Sedlařík

Bakalářská práce  
2009

Univerzita Pardubice  
Fakulta elektrotechniky a informatiky  
Katedra informačních technologií  
Akademický rok: 2008/2009

## **ZADÁNÍ BAKALÁŘSKÉ PRÁCE**

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Jakub SEDLAŘÍK**  
Studijní program: **B2646 Informační technologie**  
Studijní obor: **Informační technologie**

Název tématu: **Redakční systém pro základní školu**

### Z á s a d y p r o v y p r a c o v á n í :

Úkolem bakalářské práce je vytvořit redakční systém, který by plnil funkci prezentace ZŠ. Cíl teoretické části: - porovnání prezentací tří až pěti škol a jejich zhodnocení - návrh vhodné databáze Aplikace musí minimálně umožnit: - registraci uživatelů - přehled prospěchu žáků - podání elektronické omluvenky - zobrazování rozvrhů tříd a suplování - vkládání článků, novinek a dokumentů ze strany vyučujících a vedení školy - přístup dle práv (administrátor, učitel, rodič, žák)

Rozsah grafických prací:

Rozsah pracovní zprávy:

Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam odborné literatury:

Krčmář, Jakub. Adobe Photoshop - Praktický webdesign, Grada Publishing a. s., 2006, ISBN 80-247-1423-X. Staníček, Petr. CSS Kaskádové styly - Kompletní průvodce, Computer Press a. s., 2006, ISBN 80-7226-872-4 . Lacko, Luboslav. SQL Kapesní přehled, Computer Press a. s., 2005, ISBN 80-251-0788-4. Lacko, Luboslav. PHP 5 a MySQL 5 Hotová řešení, Computer Press a. s., 2007, ISBN: 978-80-251-1695-1. Škultéty, Rastislav. JavaScript - Programujeme internetové aplikace. Computer Press a. s., 2004, ISBN 80-251-0144-4. Lacko, L. . Oracle, správa, programování a použití databázového systému. Computer Press a. s., 2007,ISBN 978-80-251-1490-2.

Vedoucí bakalářské práce:

**RNDr. Iva Rulicová**

Katedra informačních technologií

Datum zadání bakalářské práce:

**15. ledna 2009**

Termín odevzdání bakalářské práce:

**15. května 2009**



doc. Ing. Simeon Karamazov, Dr.

děkan



Ing. Lukáš Čegan  
vedoucí katedry

V Pardubicích dne 31. března 2009

## **Prohlášení autora**

Prohlašuji:

Tuto práci jsem vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorský zákon, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Souhlasím s prezenčním zpřístupněním své práce v Univerzitní knihovně.

V Mladých Bukách dne 11. 8. 2009

Jakub Sedlařík

## **Souhrn**

Tato bakalářská práce se zabývá problematikou návrhu a tvorby redakčního systému pro základní školu. Teoretická část obsahuje popis redakčního systému, porovnání prezentací základních škol a návrh vhodné databáze. Praktická část pak pojednává o vývoji webové aplikace a použitých technologiích.

## **Klíčová slova**

redakční systém, internetová prezentace, relační databáze, PHP, HTML, CSS, jQuery

## **Title**

CMS for elementary school

## **Abstract**

This thesis deals with the issues regarding the design and creation of the content management system (hereinafter "CMS") for elementary schools. The theoretical part includes a description of the CMS, comparison of presentations of different elementary schools and a proposition for a suitable database. The practical part of the thesis then describes the development of the web application and used technologies.

## **Keywords**

content management system, internet presentation, relation database, PHP, HTML, CSS, jQuery

## Obsah

1	Úvod.....	- 10 -
2	Redakční systém .....	- 11 -
2.1	Základní popis .....	- 11 -
2.2	Vznik, vývoj .....	- 11 -
2.3	Požadavky .....	- 12 -
2.4	Vzhled a použitelnost .....	- 13 -
2.5	Pořízení.....	- 14 -
2.6	Využití CMS.....	- 15 -
2.7	Umístění .....	- 15 -
3	Hodnocení školních webů.....	- 17 -
3.1	ZŠ a MŠ Svoboda nad Úpou .....	- 18 -
3.2	ZŠ a MŠ Horní Maršov .....	- 19 -
3.3	ZŠ a MŠ Rudník .....	- 20 -
4	Návrh databáze .....	- 21 -
4.1	Fáze návrhu databáze .....	- 21 -
4.2	Konceptuální návrh .....	- 21 -
4.3	Logický návrh.....	- 22 -
4.3.1	Normalizace .....	- 22 -
4.3.2	Nultá normální forma.....	- 23 -
4.3.3	První normální forma .....	- 23 -
4.3.4	Druhá normální forma.....	- 23 -
4.3.5	Třetí normální forma .....	- 24 -
4.3.6	Denormalizace.....	- 24 -
4.4	Fyzický návrh .....	- 24 -
4.4.1	E-R diagram .....	- 25 -
4.5	Popis tabulek .....	- 25 -
4.6	Přehled použitých databázových objektů .....	- 27 -
4.6.1	Pohledy.....	- 27 -
4.6.2	Sekvence .....	- 28 -
4.6.3	Triggery.....	- 28 -
4.6.4	Indexy.....	- 28 -
4.6.5	Funkce.....	- 29 -
4.7	Ukázky databázových dotazů .....	- 29 -
4.7.1	Vkládání článků .....	- 29 -
4.7.2	Výpis novinek v administraci .....	- 30 -
5	Vývoj aplikace .....	- 31 -
5.1	Analýza aplikace .....	- 31 -
5.1.1	Požadavky .....	- 31 -
5.1.2	Uživatelé redakčního systému.....	- 31 -

5.1.3	UML Use Case Diagram.....	- 32 -
5.1.4	UML Activity Diagram.....	- 33 -
5.1.5	Layout .....	- 34 -
5.2	Implementace webové části.....	- 35 -
5.2.1	Adresářová struktura .....	- 35 -
5.2.2	Integrace editoru tinyMCE.....	- 37 -
5.2.3	Funkce pro přístup do databáze Oracle .....	- 38 -
5.2.4	Přihlašování do administrace .....	- 39 -
6	Použité technologie.....	- 41 -
6.1	Značkovací jazyk HTML .....	- 41 -
6.2	Kaskádové styly – CSS .....	- 42 -
6.3	Skriptovací jazyk PHP.....	- 43 -
6.4	Webový server Apache .....	- 44 -
6.5	Oracle Database 10g XE edition .....	- 44 -
6.6	Javascript framework jQuery .....	- 45 -
6.7	WYSIWYG editor tinyMCE .....	- 46 -
7	Závěr .....	- 48 -

## Seznam obrázků

Obr. 1 - Fáze návrhu databáze, převzato z [8] .....	- 21 -
Obr. 2 - Tabulka v nulté normální formě .....	- 23 -
Obr. 3 - Dekompozice na třetí normální formu.....	- 24 -
Obr. 4 - Fyzický návrh databáze .....	- 25 -
Obr. 5 - UML Use Case Diagram .....	- 32 -
Obr. 6 - UML Activity Diagram .....	- 33 -
Obr. 7 - Layout prezentační části aplikace.....	- 34 -
Obr. 8 - Layout administrační části aplikace .....	- 35 -
Obr. 9 - Adresářová struktura .....	- 36 -
Obr. 10 - WYSIWYG editor tinyMCE .....	- 47 -



## Seznam zkratk

CMS – content management system	system pro správu obsahu, redakční systém
CSS – Cascading Style Sheet	kaskádové styly
DOM – Document Object Model	objektový model dokumentu
HTML – HyperText Markup Language	hypertextový značkovací jazyk
PHP – PHP: Hypertext Preprocessor	PHP: Hypertextový preprocesor
SQL – Structured Query Language	strukturovaný dotazovací jazyk
UML – Unified Modeling Language	grafický jazyk pro vizualizaci, specifikaci, navrhování a dokumentaci programových systémů
W3C – World Wide Web Consortium	mezinárodní konsorcium vyvíjející webové standardy
WWW – World Wide Web	soustava propojených hypertextových dokumentů, zkráceně též web
WYSIWYG – What you see is what you get	grafický způsob editace dokumentů, verze zobrazená na obrazovce je vzhledově totožná s výslednou verzí dokumentu

# 1 Úvod

Cílem této bakalářské práce je vytvořit webovou aplikaci sloužící jako redakční systém pro základní školu. Toto téma jsem si zvolil z důvodu zájmu o tvorbu webových aplikací a také díky spolupráci se základními školami ve svém okolí. Aplikace umožní pedagogickému sboru a vedení školy vkládání článků, novinek a dokumentů, upravování rozvrhů a dalšího obsahu prostřednictvím administračního rozhraní aplikace. Vše poté bude přehledně zobrazeno v prezentační části přístupné široké veřejnosti a samozřejmě žákům a jejich rodičům.

V první části práce bude teoreticky popsán redakční systém. Vysvětlím, co si pod tímto termínem představit, proč používat právě redakční systém pro správu obsahu webových stránek, možnosti pořízení aplikace a konečně umístění již vytvořeného redakčního systému.

V druhé části zhodnotím prezentace několika vybraných základních škol. Zaměřím se na vzhled webových stránek a jejich použitelnost. Uvedu klady a zápory hodnocených stránek.

Třetí část práce se týká logického návrhu databáze včetně postupu normalizace při tvorbě tabulek. Dále pak vysvětlím, co to jsou normální formy a k čemu nám slouží. Nakonec popíši vzniklé tabulky a ostatní databázové objekty.

V posledních dvou částech práce ukážu, jakým způsobem jsem vytvořil samotný redakční systém a zmíním se o použitých technologiích.

## 2 Redakční systém

V této kapitole se pokusím vysvětlit, co se skrývá pod pojmem redakční systém, jaké jsou výhody používání, jak si vybrat nejlépe vyhovující aplikaci a jaké jsou možnosti umístění redakčního systému.

### 2.1 Základní popis

Na začátek uvedu jednu z mnoha definic redakčního systému.

*Redakční systém představuje aplikaci, která spravuje data a informace různého charakteru a obsahu a zároveň se stará i o jejich efektivní využití a zobrazení na některém z předpřipravených výstupů. V případě webového redakčního systému je za hlavní výstup považována internetová služba world wide web (tedy zkráceně web), jenž je každému čtenáři (uživateli) přístupná skrze tzv. webový / internetový prohlížeč – např. Internet Explorer, Mozilla Firefox, Opera, Netscape Navigator, atd.[1]*

Pojem redakční systém (z anglického Content management system - CMS) lze přeložit jako systém pro správu obsahu. Pravá podstata tohoto systému je tedy publikování informací na internetu bez znalosti jakýchkoliv technologií využívaných pro tvorbu webových stránek. Je možné se setkat i s alternativními označeními a zkratkami, které mají všechny shodný význam a to např. RS, publikační systém a WCSM. Redakční systém umožňuje jednoduše přidávat novinky, publikovat články, vkládat dokumenty, případně obrázky a jiný multimediální obsah. Vložený obsah musí být snadno upravitelný a samozřejmě nesmí chybět možnost mazání. Všechna tato správa je prováděna přes administrační rozhraní redakčního systému.

Nedílnou součástí redakčního systému je prezentační část, tedy to, co uvidí návštěvník webových stránek. Z toho vyplývá, že každý redakční systém by se měl skládat ze dvou částí, a to administrační a prezentační.

### 2.2 Vznik, vývoj

V začátcích internetu byla jediná možnost, jak vytvořit webové stránky, naučit se značkovací jazyk HTML a stránky vytvářet přímo v kódu. Samozřejmě každý, kdo publikoval na internetu, HTML kód neovládal a musel využívat služeb jednotlivců nebo firem zabývajících se tvorbou www prezentací. Postupem času se vzrůstajícím rozšířením internetu narůstal samozřejmě i počet prezentací a nároky na

kvalitu zpracování. Při jakékoliv změně obsahu stránek přišel na řadu administrátor, který prováděl úpravy přímo v kódu. To bylo časově náročné a neefektivní.

Vše směřovalo k vývoji webových aplikací, které by tyto neduhy odstranily, a práci s obsahem stránek ve velké míře zjednodušily a urychlily. Začaly tak vznikat první systémy pro správu obsahu – redakční systémy.

Redakční systémy jsou dynamické aplikace, jsou tedy naprogramovány v jazycích umožňujících právě tvorbu dynamických webů. V drtivé většině případů byly tyto aplikace vyvíjeny ve skriptovacím jazyce PHP, a to především díky otevřenosti tohoto jazyka. Pro ukládání dat byla většinou využita volně dostupná databáze MySQL. Nejinak tomu je v dnešní době, kdy je v jazyce PHP vytvořeno 86 % redakčních systémů a nejčastěji používaná databáze je nadále MySQL.

### **2.3 Požadavky**

Není jednoznačně určeno, které funkce musí redakční systém poskytovat, neexistují na to žádné oficiální normy a pravidla, a ne vždy se lidé shodnou na tom, co je a co není důležité. Uvedu tedy jenom základní výčet funkcí, kterými většina dostupných redakčních systémů disponuje:

- správce uživatelů – základní delegace přístupových práv
- komplexní článkový management – přidávání, úpravy a mazání článků
- schopnost ovlivnit rozložení generované stránky – např. umístění menu
- vyhledávací subsystém
- novinkový subsystém
- komentářový subsystém
- ukládání a stahování souborů

Dále uvedu přehled několika zajímavých nadstavbových funkcí:

- galerie obrázků
- diskusní fórum
- statistika přístupů
- pokročilé možnosti nastavení
- interní komunikace, posílání e-mailů
- možnost doinstalování dodatečných rozšíření (plug-in)

Z výčtu nejobvyklejších funkcí redakčního systému je zřejmé, že se hodí především pro weby s často se měnícím obsahem. Samozřejmě lze redakční systém použít i na obyčejnou webovou prezentaci, ale tam už redakční systém ztrácí své přednosti.

## **2.4 Vzhled a použitelnost**

Neméně důležitou částí redakčního systému je rozvržení a vzhled prezentační části. Pokud výsledné zobrazení stránek bude pro uživatele nepřehledné a frustrující, často odcházejí dříve, než najdou hledanou informaci a nevynahradí to ani sebelepší obsahová část.

*Mají-li být stránky efektivní, musí být jejich kouzlo zřejmé na první pohled. A toho nejspíše dosáhnete tím, že budete vytvářet stránky, které jsou intuitivní anebo přinejmenším samovysvětlující. Pokud nutíte uživatele přemýšlet o něčem, co by měli dělat bezmyšlenkovitě, pak zbytečně plýtváte omezenou zásobou jejich trpělivosti a dobré vůle, které si s sebou každý návštěvník na novou stránku přináší. [2]*

Se vzhledem a rozvržením úzce souvisí takzvaná použitelnost.

*Webová použitelnost (ergonomie webu) se zabývá webem ve vztahu k uživateli, jak lidé internetové stránky používají, jak snadno se na nich orientují, jak rychle pochopí jejich uspořádání a ovládání. Použitelné stránky jsou přehledné, srozumitelné, snadno ovladatelné a nenutí uživatele příliš přemýšlet. [3]*

Postupem času vzniklo pět pravidel použitelnosti:

- Vytvořte jasnou vizuální hierarchii na stránce
- Neměňte zvyklosti, vše musí být jednoduše pochopitelné
- Rozdělte stránky na jasně definované oblasti
- Vyznačte odkazy, dejte jasně najevo, na co lze klepnout
- Stránky nepřepřelácejte, neuvádějte zbytečná slova

Nespornou výhodou moderních redakčních systémů je, že můžeme použít jednu z předpřipravených šablon vzhledu, které jsou většinou vytvořeny podle výše zmíněných zásad. Po výběru šablony stačí např. změna loga v záhlaví a můžeme okamžitě redakční systém začít používat. Uživateli redakčního systému zbývá plnění stránek vhodně zvoleným obsahem.

Na množství textu na webových stránkách existuje další poučka:

*Zbavte se poloviny slov na každé stránce. A potom se zbavte poloviny toho, co zbylo. [2]*

Existuje spousta dalších pouček a doporučení, kterými by se měli autoři na webových stránkách řídit, protože pokud je prvotní zkušenost návštěvníků negativní, většina z nich se už na stránky nevrátí.

## **2.5 Pořízení**

V dnešní době existuje několik možností, jak si redakční systém opatřit. Po zadání hesla „redakční systém“ či „CMS“ do libovolného internetového vyhledávače se objeví obrovské množství odkazů vedoucích na stránky zabývající se touto tematikou. Už jenom vybrat si nejvíce vyhovující aplikaci nebo službu je velice složité. Následuje přehled několika základních možností pořízení redakčního systému.

- Využít služeb některého z webů, umožňující vytvoření prezentací bez nutnosti instalace a konfigurace. Vše proběhne pouze skrz webový prohlížeč. Pořídit si a zprovoznit takový systém tedy zvládne i uživatel se základními znalostmi práce s internetem. Existují placené i neplacené služby. Jako příklad poslouží české projekty webnode.cz, estranky.cz, blog.cz.
- Další možností je stažení a nainstalování jednoho z nepřeberného množství systémů s licencí GNU/GPL Open Source, tedy s otevřeným zdrojovým kódem. Za dodržení licenčních podmínek lze takový systém bezplatně používat a upravovat. Samotná instalace už vyžaduje určité zkušenosti a alespoň základní přehled technologií pro tvorbu webu. Existují samozřejmě jednotlivci nebo firmy, které za úplatu provedou instalaci, konfiguraci a případně další úpravy jinak volně šiřitelných systémů. Mezi nejznámější patří WordPress, Joomla, phpRS, Mambo, CMS Made Simple, RS2, Textpattern, atd.
- Jednou z dalších možností je zakoupení komerčního systému včetně jeho instalace a nastavení, případně upravení vzhledu podle požadavků. Základní funkčnost komerčních systémů lze samozřejmě rozšířit dokoupením potřebných modulů.

Nedá se jednoznačně určit, která z výše uvedených možností pořízení redakčního systému je lepší a která horší. Především záleží na zdatnosti uživatele a neméně na finančních prostředcích. V některých případech neplatí, že placený produkt je lepší než ten s otevřeným kódem. Nespornou výhodou Open Source systémů je obrovská uživatelská základna a samozřejmě cena. Ostatní uživatelé těchto systémů vám zpravidla rádi poradí a vyřeší problémy, které se s provozováním daného CMS objeví.

## **2.6 Využití CMS**

Současné redakční systémy jsou velice vyspělé a do určité míry i univerzální, tudíž je lze úspěšně využít téměř u všech typů informačních webů.

Pro příklad uvedu několik nejčastějších použití redakčních systémů:

- internetové časopisy
- osobní stránky, blogy
- stránky institucí, škol, obcí
- stránky neziskových organizací
- zájmový/komunitní server, většinou s využitím diskusního fóra

Existují samozřejmě i nevhodná použití redakčních systémů:

- elektronický obchod
- portálový server
- velké firemní prezentace

V těchto případech je lepší sáhnout po specializovaných aplikacích nebo si nechat ušít web na míru.

## **2.7 Umístění**

Jako poslední úkol, který stojí před zprovozněním redakčního systému, je umístění na webový server a tím zpřístupnění internetové veřejnosti. Zpravidla se volí služeb webhostingu.

*Webhosting je pronájem prostoru pro webové stránky na cizím serveru. Pronajímatel serveru bývá označován jako poskytovatel webhostingu (webového prostoru).*

*Díky webhostingu si můžete své webové stránky umístit na internet, aniž byste museli mít vlastní server. Ceny za webhosting se pohybují od pár Kč až po několik tisíc Kč za měsíc. Existuje i bezplatná varianta, tzv. freehosting. Freehosting obvykle nezahrnuje žádné záruky ohledně funkčnosti, má omezenou technickou podporu. Často je s freehostingem spojeno umísťování reklamy na stránkách. [4]*

Pro osobní stránky můžeme zvolit freehosting, pokud tedy nepožadujeme vysokou dostupnost a rychlost a ani nepočítáme s vysokými datovými přenosy. U většiny freehostingových poskytovatelů dostaneme při registraci doménu třetího řádu (ve tvaru *uzivatel.hosting.cz*), tudíž odpadá nutnost registrace doménového jména a zaplacení odpovídajících poplatků. Samozřejmě můžeme dodatečně zaregistrovat vlastní doménu druhého řádu a nadále využívat freehostingové služby.

Pro projekty, u kterých je požadována vysoká dostupnost a předpokládá se vysoká návštěvnost stránek, nezbyvá než doporučit placený webhosting. Ceny webhostingových služeb se pohybují od několika desítek korun měsíčně, což nepředstavuje nikterak závratnou sumu, uvědomíme-li si, kolik by úspěšná prezentace mohla přinést.

Další popis, možnosti pořízení či postupy instalace redakčních systémů by rozsahem vydaly na samostatnou práci. Pro vytvoření představy o redakčních systémech by však tato kapitola měla postačovat.



### 3 Hodnocení školních webů

*V poslední době se masivně setkáváme s rozvojem webu různého zaměření. Mezi nimi lze ve stále větší míře pozorovat webové prezentace jednotlivých škol, tzv. školní weby. Školní web představuje moderní způsob poskytování informačních a komunikačních služeb. Uživatelé těchto služeb mohou být značně různorodí, ale zejména se bude jednat o učitele, žáky, rodiče, zřizovatele škol a širší veřejnost.*

*Doposud této oblasti nebyla věnována téměř žádná pozornost. Analyzujeme-li veškeré dostupné pedagogické slovníky, které kdy vyšly na území ČR a SR, nenalezneme heslo školní web, natož jeho definici. Nicméně jde o dnes již běžně užívaný termín, který lze zaznamenat v komunikaci ICT odborníku v oblasti školství. [5]*

Před zahájením tvorby redakčního systému jsem prostudoval velké množství již vytvořených školních webů. Především webů základních škol. Důvodem byla snaha zjistit, jak školní weby vypadají, co obsahují za informace, jak jsou tyto informace rozděleny do kategorií, jak často jsou stránky aktualizovány, jaké nadstavbové aplikace používají. Získané informace jsem poté využil při tvorbě vlastních webových stránek, respektive redakčního systému.

Informace, které nelze získat pouhým procházením webových stránek, jsem zjišťoval rozesláním dotazníku uvedeným v příloze B. Při hodnocení škol jsem částečně využíval on-line aplikaci „Náležitosti školního webu - autoevaluační asistent“.<sup>1</sup>

*Systém autoevaluace je nástrojem, pomocí kterého si sám autor může určit, jaké úrovně kvality jeho web dosahuje. Mnohdy cennější a užitečnější, nežli dosažené skóre je samotný proces evaluace, kdy autor webu prochází jednotlivá hodnotící kritéria a konfrontuje je s výsledky vlastní práce. [6]*

Pro hodnocení prezentací jsem vybral následující tři základní školy z okolí mého bydliště.

---

<sup>1</sup> URL: <http://ondrej.neumajer.cz/skolniweb/aea/online/>

### **3.1 ZŠ a MŠ Svoboda nad Úpou**

První webové stránky, které jsem vybral pro hodnocení, patří Základní škole a mateřská škola ve Svobodě nad Úpou<sup>2</sup>.

Prezentační část má velmi přehledný a decentní vzhled, kde převažuje kombinace modré a světle šedé barvy s dobrou mírou kontrastu mezi písmem a pozadím. Stránky mají klasické rozložení prvků. V hlavičce jsou fotografie školních budov a oficiální název školy. V levém menu jsou k dispozici odkazy do všech sekcí stránek. V pravém sloupci se pak zobrazuje samotný obsah vybraných sekcí.

Po obsahové stránce nabízí tento školní web dostatečné množství informací. Zajímavý je například seznam ředitelů vedený od roku 1906 a seznam absolventů od poválečného roku 1945. Aktualizace novinek a článků probíhá několikrát týdně.

Z vyplněného dotazníku vyplývá, že škola využívá vlastního řešení redakčního systému, kam má přístup vedení školy a samozřejmě administrátor. Vedení školy má možnost vkládat novinky, dokumenty a fotografie. Ostatní obsah vkládá, či upravuje administrátor, se kterým škola komunikuje přes email přibližně pětkrát do roka.

Pro vedení elektronické agendy, kam mají přístup žáci, jejich rodiče a učitelé využívá škola aplikaci iŠkola<sup>3</sup>. Formulář pro přihlášení do iŠkoly se nachází přímo na webových stránkách a to pod levým menu. Z dotazníku je dále zřejmé, že jsou škola i žáci s webovými stránkami spokojeni a v nejbližší době se nechystají změny ani úpravy stránek.

**Klady:** jednoduchý a přehledný layout, časté aktualizace, dostatečné množství informací, využívání elektronické agendy iŠkola.

**Zápory:** velké množství chyb v HTML kódu - stránky nejsou validní, občas nesourodá velikost a barvy písma (např. sekce Mateřská škola, Mažoretky, Novinky).

---

<sup>2</sup> URL: <http://www.zssvoboda.eu/>

<sup>3</sup> URL: <http://www.iskola.cz/>

### 3.2 ZŠ a MŠ Horní Maršov

Další hodnocené webové stránky patří Základní škole a mateřské škole v Horním Maršově<sup>4</sup>. Základním stavebním prvkem tohoto webu je komplexní a univerzální redakční systém Plone<sup>5</sup>, který spadá do kategorie Open Source.

Již při prvním pohledu na stránky mě zarazilo umístění loga základní školy, které je netradičně napravo a je poměrně malé. Dalším neobvyklým prvkem je měnící se fotografie (umístěná v levé části hlavičky) podle zvolené sekce. Bohužel ani zvolená šablona vzhledu nepřidává na přehlednosti stránek. Celkově je orientace na stránkách problematičtější než v předchozím případě.

Po obsahové stránce je prezentace velmi zdařilá. Použitý redakční systém dovozuje umístění anket, diskusního fóra, kalendáře, odeslání vzkazu učitelům a mnoho dalších pokročilých funkcí. Občas jsem však narazil na nedokonalost jazykové mutace v podobě anglických popisků.

Z dotazníku vyplývá, že administrátor stránek je zároveň zaměstnancem školy a ostatní kolegové s ním úzce spolupracují. Redakční systém však není využit naplno, protože obsahem ho plní pouze zmíněný administrátor. Díky pokročilému redakčnímu systému, může administrátor vkládat a upravovat veškerý obsah bez zásahu do HTML kódu. To mimochodem zajišťuje validitu výsledného kódu. Dále je umožněno přihlášení žáků, které jim zpřístupní i neveřejné části webu. Škola využívá program pro školní administrativu Bakaláři<sup>6</sup>. Ten poskytuje žákům přehled o studiu skrze webový prohlížeč a rozšiřuje tak samotný redakční systém školy.

Hlavní výhody komunikace se žáky a jejich rodiči přes webové stránky shledává vedení školy v rychlosti, dostupnosti a zachování diskrétnosti. V blízké budoucnosti se škola chystá na změnu problematického layoutu a designu stránek.

---

<sup>4</sup> URL: <http://www.zshornimarsov.cz/>

<sup>5</sup> URL: <http://plone.org/>

<sup>6</sup> URL: <http://www.bakalari.cz/>

**Klady:** komplexní redakční systém, přihlašování žáků, rozsáhlá a přehledná fotogalerie, časté aktualizace, dodržení webových standardů.

**Zápory:** zmiňovaná nepřehlednost stránek, nekompletní překlad RS.

### 3.3 ZŠ a MŠ Rudník

Poslední webové stránky, které jsem hodnotil, patří Základní škole a mateřské škole v Rudníku<sup>7</sup>. Jedná se o velmi jednoduché statické stránky s archaickým vzhledem, který odpovídá počátkům tvorby internetových prezentací. Layout se skládá z hlavičky a dvou sloupců. Levý slouží jako menu a v pravém se vypisují aktuality. Po kliknutí na odkazy v menu se ocitneme na vzhledově rozličných stránkách, kde neexistují ani zpětné odkazy na domovskou stránku. Jsme tudíž nuceni používat tlačítko zpět ve webovém prohlížeči. Tímto stránky porušují většinu pravidel použitelnosti. Fotogalerii vytváří vedení školy pomocí služby rajce.net<sup>8</sup>.

Z dotazníku vyplývá, že obsah stránek doplňuje a upravuje vedení školy, a to přímo v HTML kódu. Aktualizace se provádějí přibližně jednou týdně. Díky statické podobě stránek nemají žáci a učitelé možnost přihlášení. Vedení školy považuje za výhodu webových stránek aktuálnost a veřejný přístup. Zároveň jsou stránky neosobní a nezaručují přečtení vydaných zpráv. Škola bohužel nepočítá s obnovou současných stránek, které by podle mého názoru měly být kompletně předělány.

**Klady:** poměrně časté aktualizace.

**Zápory:** zastaralost, nevyužívá žádných moderních technologií, špatná přístupnost a nevalidní HTML kód.

Při zkoumání a následujícím hodnocení školních webů, jsem se snažil pochytit, co nejvíce správných postupů pro tvorbu přehledného a kvalitního školního webu. Zároveň některé weby ukázaly, čeho se při tvorbě vyvarovat. V neposlední řadě poskytl procházení velkého množství školních stránek inspiraci pro vlastní tvorbu.

---

<sup>7</sup> URL: <http://www.zsrudnik.cz/>

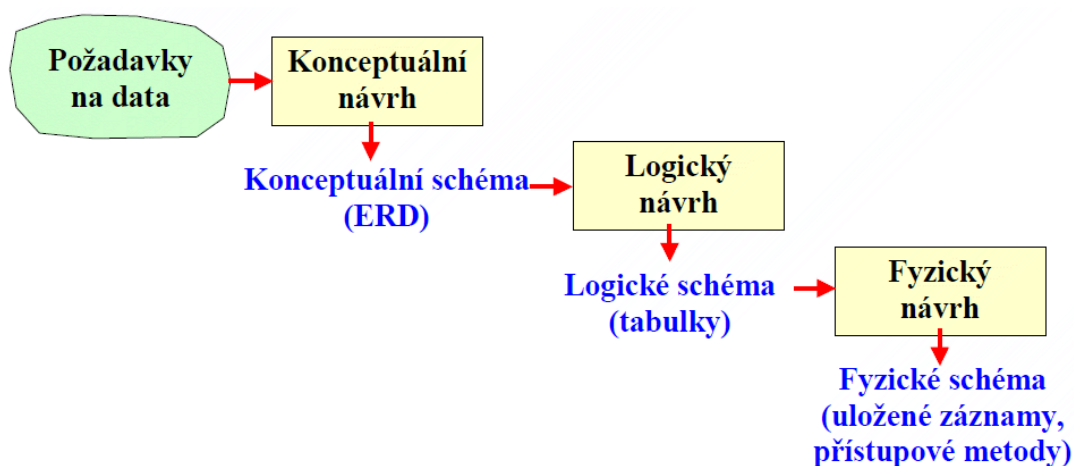
<sup>8</sup> URL: <http://www.rajce.idnes.cz/>

## 4 Návrh databáze

Při vytváření webové aplikace, která načítá nebo ukládá nějaká data, je jedním z nejdůležitějších prvních kroků správný návrh datových struktur, ve kterých budou tato data uchována. V první řadě musím doporučit, pro ukládání dat, zásadně používat databáze. K webovým aplikacím totiž může přistupovat více uživatelů najednou a např. při ukládání dat do souborů se nevyhnete nutnosti tyto soubory např. správně zamykat. Kromě toho databáze nabízí nesrovnatelně větší komfort při jakékoliv manipulaci s daty. [7]

### 4.1 Fáze návrhu databáze

Proces návrhu databáze spočívá v několika krocích, zobrazených na následujícím schématu.



Obr. 1 - Fáze návrhu databáze, převzato z [8]

### 4.2 Konceptuální návrh

V první fázi návrhu databáze se vytváří konceptuální model. Ten se vyznačuje následujícími vlastnostmi:

- slouží k popisu dat v databázi nezávisle na jejich fyzickém uložení
- umožňuje zobrazit a popsat objekty v databázi a vztahy mezi nimi z hlediska jejich významu a chování
- výsledkem je implementačně nezávislé schéma obecně aplikovatelné v jakémkoli prostředí
- znázorňuje se nejčastěji E-R diagramem, který definuje entity (třídy prvků), jejich atributy a vztahy mezi nimi

**Entita** je prvek reálného světa (např. člověk, auto, město), který je popsán svými charakteristickými vlastnostmi. Tyto vlastnosti se nazývají **atributy** (např. váha, věk, výška). **Kardinalita** vztahu vyjadřuje skutečnost, kolik (jeden či mnoho) řádků jedné tabulky může vstoupit do vztahu s kolika řádky druhé tabulky. Existují tři druhy vztahů:

- 1:1 - jeden občan dané země má právě jedno rodné číslo
- 1 : N - jeden člověk může vlastnit více kreditních karet, ale jedna kreditní karta má právě jednoho vlastníka
- M : N - student si může zapsat několik předmětů a zároveň jeden předmět může mít zapsáno více studentů

Dalším pojmem, s kterým se při návrhu setkáme je **parcialita** vztahu. Ta vyjadřuje povinnost či nepovinnost existence ve vztahu. Existují jednostranně parciální a oboustranně parciální vztahy.

### 4.3 Logický návrh

V této části návrhu databáze se vytvoří relační model pomocí transformace z ER diagramu. Tato transformace probíhá pomocí normalizace stávajících entit. Všeobecně se doporučuje vytvářet entity (tabulky) splňující alespoň třetí normální formu.

#### 4.3.1 Normalizace

*Je to sada pravidel, které umožňují vytvářet optimální datové struktury. Normalizace v databázi odstraňuje redundanci (duplicitní data), čímž se snižují nároky na velikost úložného prostoru na disku. Práce s normalizovanými daty je tak pro databázový stroj obecně rychlejší, protože zbytečně nemusí číst a zapisovat velké množství dat (operace zápisu a čtení z disku jsou obecně pomalé a často bývají úzkým hrdlem databázových systémů). Normalizovaná databáze také zjednodušuje vývoj vlastní databázové aplikace, protože data stačí zapsat a modifikovat na jednom místě. [9]*

### 4.3.2 Nultá normální forma

Tabulka je v nulté normální formě, existuje-li takové její pole, které obsahuje více než jednu hodnotu. V tabulce **ucitele** se nachází atribut **jmeno\_prijmeni**, který je dále dělitelný.

ucitele	
id_ucitele	NN (PK)
jmeno_prijmeni	
mail	
nick	
heslo	
tridnictvi	
cislo_opravneni	
nazev_opravneni	
aktivni	

Obr. 2 - Tabulka v nulté normální formě

### 4.3.3 První normální forma

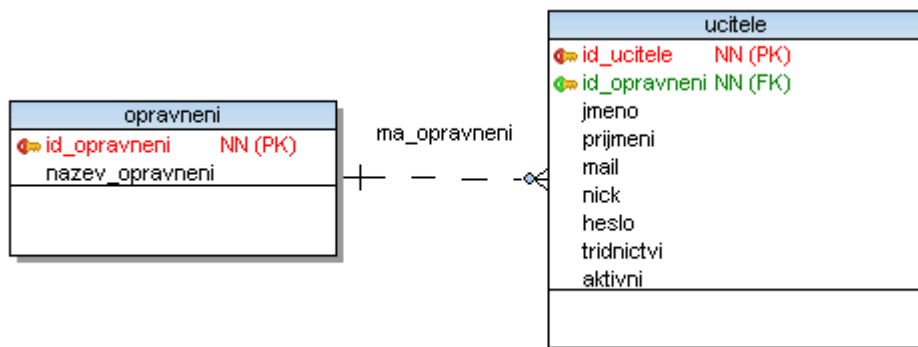
Relace je v první normální formě, pokud každý její atribut obsahuje jen atomic-ké hodnoty. Tedy hodnoty z pohledu databáze již dále nedělitelné. Tabulka **ucitele** se do první normální formy dostala tak, že jsem atribut **jmeno\_prijmeni** rozdělil na dva samotné atributy **jmeno** a **primeni**.

### 4.3.4 Druhá normální forma

Tabulka vyhovuje druhé normální formě, pokud se nachází v první normální formě a zároveň jsou všechny její neklíčové atributy funkčně závislé na celém kandidátním klíči. Tuto normální formu tabulka **ucitele** splňuje, protože obsahuje pouze jeden kandidátní klíč.

### 4.3.5 Třetí normální forma

V této formě se nachází tabulka, splňuje-li předcházející dvě formy a všechny její neklíčové atributy jsou vzájemně nezávislé. Tuto formu tabulka **ucitele** nespĺňuje, protože atribut **nazev\_opravneni** je závislý na atributu **cislo\_opravneni**. Řešením je bezetrátová dekompozice na dvě tabulky.



Obr. 3 - Dekompozice na třetí normální formu

### 4.3.6 Denormalizace

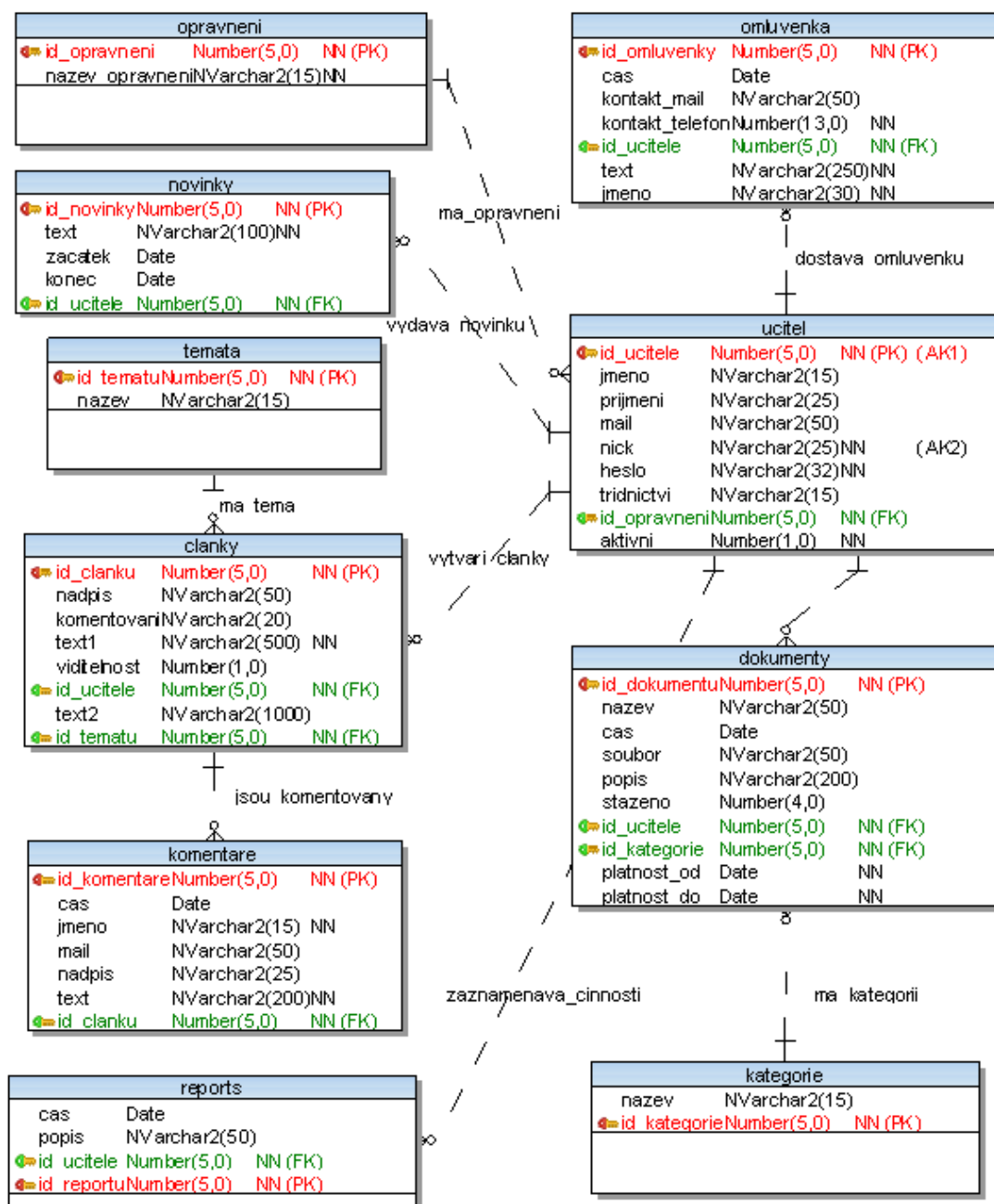
*Normalizace je jedním z nástrojů pro optimální návrh databázové struktury. Není ale vždy nutné nebo vhodné data normalizovat úplně. Často se může stát, že pro dosažení větší rychlosti zpracování dotazů je vhodné použít denormalizaci, tedy vědomě zanést do databáze určitou míru redundance. Může se například jednat o předpočítání hodnot nebo umělé snížení počtu spojovaných entit. [9]*

## 4.4 Fyzický návrh

Fyzický návrh databáze již v sobě zahrnuje konkrétní specifika použité databáze. Např. datové typy atributů a integritní omezení.



#### 4.4.1 E-R diagram



Obr. 4 - Fyzický návrh databáze

#### 4.5 Popis tabulek

Z předešlého E-R diagramu jsou patrné použité databázové tabulky a vztahy mezi nimi. V každé tabulce jsou uvedeny atributy a s nimi související datové typy včetně integritních omezení. Všechny uvedené tabulky jsou minimálně ve třetí normální formě.

### **Tabulka clanky**

Tabulka *clanky* slouží k uchování všech článků redakčního systému. Obsahuje informace o nadpisu, povinné a nepovinné obsahové části (*text1* a *text2*), autorovi, datu vytvoření, zvoleném tématu. O jedinečnost primárního klíče *id\_clanku* se stará sekvence *seq\_clanky*. Při vkládání článků se v aplikaci již *id\_clanku* nevyplňuje. O to se stará příslušný trigger *trg\_clanky*.

I u ostatních tabulek v databázi je primární klíč tvořen umělým atributem s prefixem *id\_*, u kterého je zabezpečena unikátnost příslušnou sekvencí s prefixem *seq\_* a triggerem s prefixem *trg\_*.

### **Tabulka temata**

Při vytváření článku je zvoleno právě jedno téma z tabulky **temata**. Tabulka uchovává názvy vytvořených témat.

### **Tabulka novinky**

Tabulka **novinky** slouží k uchování všech novinek redakčního systému. Při vytváření novinky se zvolí dvě data, mezi kterými bude novinka viditelná na webu. Dále je uveden samotný text novinky a její autor.

### **Tabulka dokumenty**

Při vkládání nového dokumentu se do příslušné tabulky ukládá název, čas uložení, cesta k souboru, krátký popis, dvě data, mezi kterými je dokument platný, a nechybí kategorie a autor. V tabulce se bude také zaznamenávat počet stažení každého dokumentu.

### **Tabulka kategorie**

Při vytváření dokumentů se volí jedno z vytvořených témat, uložených právě v této tabulce. Uchovávají se zde tedy názvy kategorií.

### **Tabulka ucitel**

V této tabulce jsou uvedeni uživatelé mající přístup do administrační části redakčního systému. Obsahuje jméno, příjmení a emailovou adresu. Dále je v atributu *tridnictvi* uloženo pořadové číslo ročníku, pokud učitel není třídním, je zde uvedena hodnota *null*. Atribut *aktivni* nabývá hodnot 0 nebo 1. Pokud má učitel nastavenou hodnotu 0, je mu odepřen přístup do administrace. V databázi je však potřeba udržovat i neaktivní uživatele, kvůli návaznosti na vytvořený obsah. Další dva atributy *nick* a *heslo* slouží pro přihlašování uživatele do administrace. Heslo je uloženo v zašifrované podobě jako MD5 hash.

### **Tabulka opraveni**

Obsahuje seznam oprávnění používaných v aplikaci. Obsahuje pouze jednoznačný identifikátor a související název.

### **Tabulka reports**

Do této tabulky se zaznamenávají úkony, které uživatelé v administraci provádějí. Např. přihlášení a odhlášení z aplikace, vkládání, upravování a mazání obsahové části.

### **Tabulka omluvenka**

V této tabulce se uchovávají informace o podaných elektronických omluvenkách. Obsahuje jméno, email, telefonní číslo a vlastní text omluvenky. Každá omluvenka je přiřazena právě jednomu učiteli díky atributu *id\_ucitele*.

## **4.6 Přehled použitých databázových objektů**

### **4.6.1 Pohledy**

Pohled, narozdíl od tabulky, kde jsou data uložena přímo, obsahuje pouze předpis, jakým způsobem mají být data získávána z tabulek. Toho se hojně využívá u SQL dotazů, při kterých se často spojuje několik tabulek. Výsledný dotaz nad vytvořeným pohledem je mnohem jednodušší a přehlednější. Nevzniká tak zbytečné množství kódu v samotné aplikaci. Pokud se rozhodneme pohled upravit, děje se to přímo v databázi a odpadají tak starosti s úpravou kódu v aplikační části.

Následuje ukázka pohledu *view\_clanky\_vypis*, který je využíván pro zobrazení článků v prezentační části. Spojuje tabulky **clanky**, **temata** a **ucitel**.

```

create or replace view view_clanky_vypis as
select id_clanku, nadpis, text1, text2,
       to_char (datum, 'Dy DD.MM.YYYY, HH24:MI') AS datum,
       ucitel.jmeno || ' ' || ucitel.prijmeni as jmeno,
       temata.nazev as tema
from clanky
join temata on clanky.id_tematu = temata.id_tematu
join ucitel on clanky.id_ucitele=ucitel.id_ucitele
order by id_clanku desc;

```

#### 4.6.2 Sekvence

Pro zabezpečení unikátnosti klíčových atributů tabulek slouží vytvořené sekvence. Následující sekvence se využívá u tabulky **clanky**. Minimální hodnota je 1, maximální 9999, sekvence zvyšuje svoje číslo po jedné a začíná se číslovat od jedničky. Ostatní sekvence v databázi jsou shodné s touto.

```

create sequence seq_ucitel minvalue 1 maxvalue 9999 increment by
1 start with 1 cache 20 noorder nocycle;

```

#### 4.6.3 Triggery

Trigger je uložená procedura, která se spouští za přesně definovaných událostí nad určitou tabulkou. Uvedený trigger zajistí za pomoci odpovídající sekvence vložení unikátního *id\_clanku* při operaci insert nad tabulkou **clanky**. V aplikaci se tedy s umělými atributy s prefixem *id\_* vůbec neoperuje.

```

create or replace trigger trg_clanky
before insert on clanky
for each row
begin
  select seq_clanky.nextval into :new.id_clanku from dual;
end;

```

#### 4.6.4 Indexy

Indexy slouží k rychlejšímu procházení tabulek. V tomto případě jsem zvolil index nad atributem *id\_clanku* v tabulce **clanky**, předpokládá se, že tato tabulka bude rozsáhlejší a bude se nad ní provádět nejvíce dotazů SQL.

```

create index index_clanky on clanky(id_clanku);

```

### 4.6.5 Funkce

Následující funkce vrací jako výsledek počet znaků ve všech člancích konkrétního učitele. Vstupním parametrem bude logicky *id\_ucitele*.

```
create or replace function fce_delkaclanku (id_uc in number)
return number
as
  n_delka number;
begin
  select sum(length(clanky.text1)) +
sum(nvl(length(clanky.text2),0)) into n_delka
  from clanky
  where clanky.id_ucitele = id_uc;
  return n_delka;
end;
```

## 4.7 Ukázky databázových dotazů

### 4.7.1 Vkládání článků

Následuje ukázka SQL dotazu sestaveného pomocí jazyka PHP, konkrétně vkládání článků do tabulky **clanky**. Hodnoty jsou ukládány do databáze z příslušného formuláře přes asociativní pole *\$\_POST* a *\$\_SESSION*. Jediné datum a čas je vkládáno pomocí vnořeného selectu.

```
insert into clanky (id_ucitele, id_tematu, nadpis, text1, text2,
datum)
  values
    (
      '$_SESSION['id_ucitele'].',
      '$_POST['tema'].',
      '$_POST['nadpis'].',
      '$_POST['text1'].',
      '$_POST['text2'].',
      (select sysdate from dual)
    )
```

#### 4.7.2 Výpis novinek v administraci

Při výpisu přehledové tabulky novinek chci zobrazit autora, datum platnosti novinky, které je vytyčeno atributem *zacatek* a *konec* a dále vypsát text novinky. Výsledek je poté řazen sestupně podle *id\_novinky*.

```
select novinky.id_novinky, novinky.text, novinky.zacatek,  
       novinky.konec, ucitel.jmeno, ucitel.prijmeni  
from novinky  
join ucitel on novinky.id_ucitele = ucitel.id_ucitele  
order by novinky.id_novinky desc
```

## 5 Vývoj aplikace

### 5.1 Analýza aplikace

Cílem je vytvoření webové aplikace sloužící jako redakční systém pro Základní a mateřskou školu, Mladé Buky. Aplikace bude mít dvě části. Prezentační část a administrační část. Do administrační části budou mít přístup tři typy uživatelů s různými oprávněními - administrátor, ředitel, učitel. Aplikace bude mít dynamický obsah, budou tedy použity odpovídající webové technologie, které jsou blíže specifikovány v kapitole „Použité technologie“.

#### 5.1.1 Požadavky

Aplikace bude umožňovat oprávněným uživatelům vkládat, upravovat a případně odstraňovat články, dokumenty, novinky a obrázky. V prezentační části aplikace pak tento obsah přehledně zobrazovat. Velký důraz byl kladen na přehledné a intuitivní administrační rozhraní s integrovaným WYSIWYG editorem, vhodným i pro méně zdatné uživatele. V neposlední řadě bude aplikace umožňovat podání elektronické omluvenky či úpravy rozvrhů. Obsah, který se bude aktualizovat jen jednou za školní rok, např. organizace šk. roku a seznam zaměstnanců, nebude editovatelnou položkou administrace a bude ho aktualizovat administrátor stránek přímo v HTML kódu. Aplikace umožní přihlášení žáků a jejich rodičů do elektronické agendy, kterou škola využívá.

#### 5.1.2 Uživatelé redakčního systému

##### Administrátor

- nejvyšší oprávnění, jako jediný je vytvořen již při instalaci redakčního systému
- vytváří uživatele systému
- určuje uživatele s oprávněním ředitel
- resetuje zapomenutá hesla
- může sledovat dění v administraci
- má veškerá práva v kategoriích týkajících se obsahu stránek
- administrátor bude mít většinou i přístup k databázi a všem souborům umístěným na serveru

## Ředitel

- vytváří uživatele s oprávněním *učitel*
- resetuje zapomenutá hesla
- může sledovat dění v administraci
- má veškerá práva v kategoriích týkajících se obsahu stránek

## Učitel

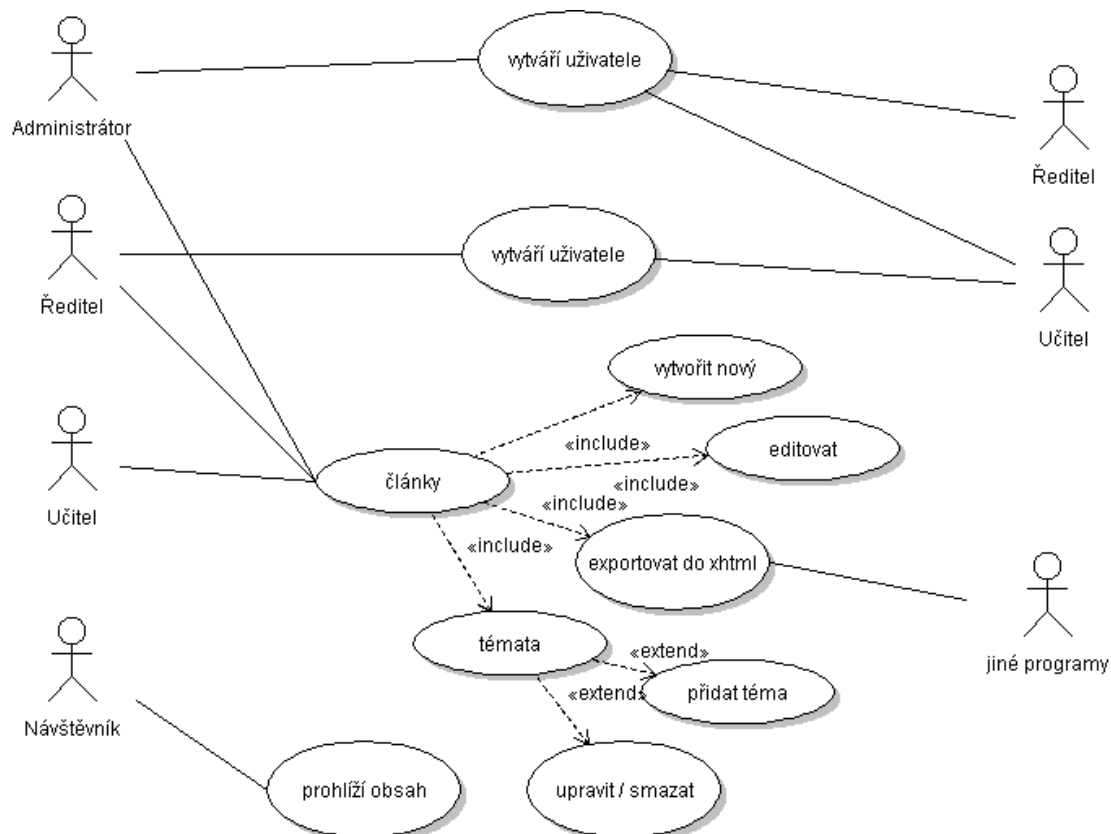
- vytváří a spravuje obsah redakčního systému
- může upravovat a mazat pouze ten obsah, který vytvořil

## Rodiče, žáci, veřejnost

- prohlížejí prezentační část, stahují dokumenty
- posílají třídním učitelům elektronickou formu omluvenky
- přidávají komentáře

### 5.1.3 UML Use Case Diagram

Zobrazení dynamické (funkční) struktury systému z pohledu uživatele. Je primárně určen k definici chování systému, aniž by odhaloval jeho vnitřní strukturu. Popisuje komunikaci mezi uživatelem a systémem.



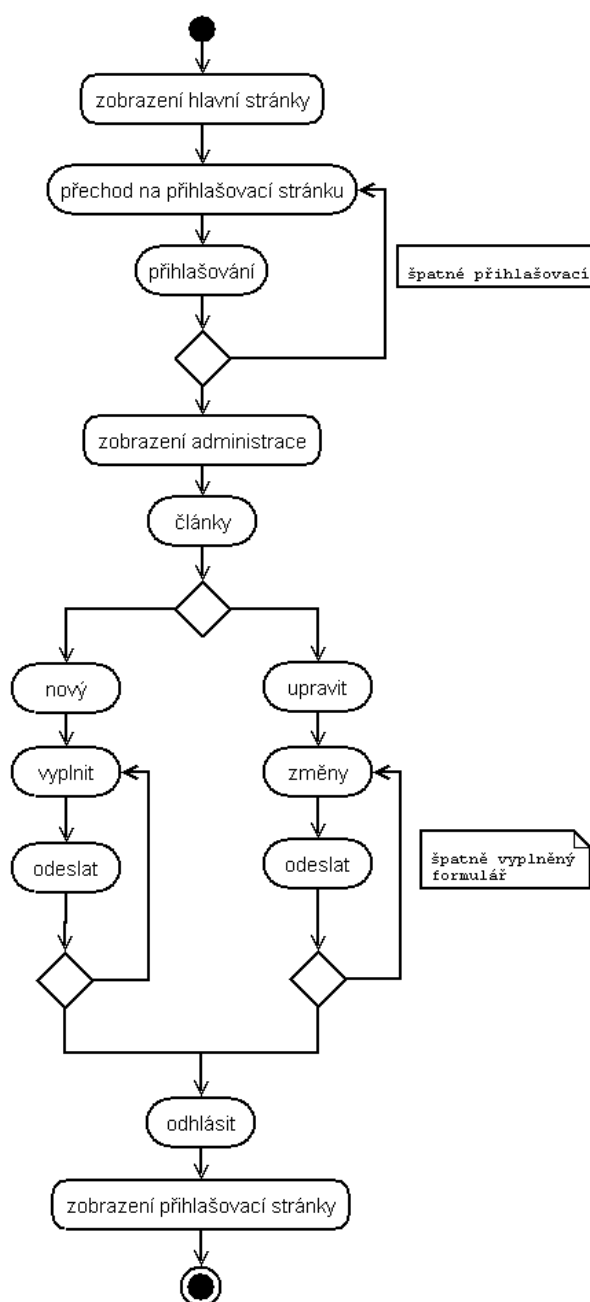
Obr. 5 - UML Use Case Diagram



### 5.1.4 UML Activity Diagram

Tento diagram má za úkol popsat činnost uživatele, který chce přidat nebo upravit článek pomocí administračního rozhraní aplikace.

Nejdříve se zobrazí úvodní stránka webu, uživatel přejde pomocí odkazu na přihlašovací stránku administrace. Po správném zadání údajů se dostává do administrace redakčního systému. Zde zvolí kategorii článku. Rozhodne se mezi vložení nového článku nebo vybere ze seznamu článek pro úpravu. Pokud vše regulerně vyplní a článek se odešle do databáze, může se odhlásit.



Obr. 6 - UML Activity Diagram

### 5.1.5 Layout

Layout, tedy rozvržení webové stránky, je schéma, podle kterého jsou ve stránce rozmístěny její stavební prvky – logo, menu, obsahová část, navigace, patička. Volba schématu hraje při konstrukci webu důležitou roli. Jako nejlepší řešení se jeví použít jedno ze zažitých schémat, které bude pro návštěvníky respektive uživatele webových stránek intuitivní. Budou tedy vytvořeny podle zvyklostí a doporučení.

Jak již několikrát bylo uvedeno, vytvořená aplikace se skládá ze dvou částí - prezentační a administrační. Obě slouží pro jiné účely, a tudíž bylo potřeba vytvořit dva různé návrhy layoutu.

**Prezentační část** aplikace jsem vytvořil podle klasického schématu – hlavička, levý sloupec sloužící jako hlavní menu, v prostředním sloupci jsou vypisovány články a ostatní obsah stránek, v pravém sloupci jsou zobrazeny aktuality a je zde také umístěn formulář pro podání elektronické omluvenky, vše ukončuje patička s odkazem na přihlašovací stránku administrace.

**ZÁKLADNÍ ŠKOLA A MATEŘSKÁ ŠKOLA, MLADÉ BUKY**

Přihlášení do iškoly

Historie  
Kontakty  
Zaměstnanci  
Školská rada  
Organizace šk. roku  
Rozvrhy  
Suplování

**Školní jídelna**  
Jídelníček  
Provozní řád

**Mateřská škola**  
Kontakty  
Zaměstnanci  
Fotogalerie

**Úřední deska**  
Školní řád  
Výroční zpráva  
Dokumenty

Úvodní strana >

#### Výlet do Adršpachu

Přes ranní nepříznivé počasí, kdy přelo a foukal silný vítr, jsme se v úterý 16. 6. 2009 vydali se žáky 1. třídy na výlet do Adršpachu. Počasí se časem umoudřilo, a tak jsme ve skalách prožili příjemný den. Naši prohlídku jsme začali u zatopeného lomu, kde byly děti nadšené z hejna divokých kačenek, které se nebály přijít či připlavat tak blízko k nám, že si na ně děti mohly téměř sáhnout. Cesta pak pokračovala po vyznačené trase směrem ke skalnímu jezeru, protože jsme nesměli vynechat projíždku na lodičce.

[Pokračuj](#)

Jakub Sedlařík | Út 04.08.2009, 23:14 | Škola

#### Hrad Pecka

26.5.09 Žáci 4. a 5. třídy odjeli na jednodenní školní výlet. Navštívili hrad Pecka. Na nádvoří hradu shlédli představení, které bylo věnováno Karlu IV. – významné osobnosti z naší historie. Pak si prohlédli hrad a odjeli do nedalekého městečka Lázně Bělohrad. Zde se občerstvili a prošli si naučnou stezkou, které vedla překrásným lesoparkem až ke kostelíčku, jenž se nachází na vršku zvaném Byšičky. Výlet se vydařil, přestože počasí bylo velmi teplé a jízda autobusem nebyla vůbec osvěžující.

[Pokračuj](#)

Jakub Sedlařík | Pá 31.07.2009, 10:59 | Škola

#### Výhra v krajském finále atletického čtyřboje

Tým starších žáků ZŠ Mladé Buky zvítězil v krajském finále atletického čtyřboje (běh na 60m, dálka nebo výška, hod míčkem nebo vrh koulí a běh na 1km) pod vedením Mgr. Miloše Novotného v sestavě: František Droppa, David Hruža, Jakub Kaňka, Vítězslav Zýka a Tomáš Jelinek. Tomuto úspěchu předcházela nejen dlouhodobá příprava

The Truth Is Out There  
Jakub Sedlařík  
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vivamus laoreet condimentum magna. Etiam laculis cursus urna. Praesent consequat nisi et ligula. Sed nikh metus, malesuada non, vestibulum viverra, lacinia sed, felis.

Jakub Sedlařík

**Omluva dítěte**

Vaše jméno  
Váš e-mail  
Tel. kontakt  
Třída  
6. třída  
Omluva  
[Odeslat](#)

Obř. 7 - Layout prezentační části aplikace

**Administrační část** má jednoduchý a přehledný layout, který je vytvořen bez zbytečných grafických prvků. Po přihlášení uživatele je v horní liště zobrazeno jeho jméno a oprávnění. Jsou zde také k dispozici odkazy pro přechod na prezentační část webu a odhlášení z administrace.

V další liště jsou umístěny odkazy na jednotlivé sekce administrace. Po kliknutí na vybranou sekci se zobrazí další lišta s konkrétními úkony, které lze v dané sekci provádět.

The screenshot shows the administration interface for the website of ZŠ a MŠ Mladé Buky. The header includes the site name and the date 05.08.2009, Středa. Below the header, there is a user menu showing the user 'Jakub Sedlařík' with the role 'Administrátor' and links for 'Na web' and 'Odhlásit'. A navigation bar contains links for 'Úvodní stránka', 'Články', 'Novinky', 'Dokumenty', 'Učitelé', 'Suplování', and 'Reports'. A sub-menu for 'Články' is open, showing options like 'Nový článek', 'Úpravy a mazání', 'Témata', 'Nahraj obrázek', and 'Export do xml'. The main content area is titled 'Úpravy a mazání článků' and displays a table of articles.

ID	Nadpis	Autor	Téma	Datum	Akce
678	Den dětí ve školce	Radek Bitala	Mateřská škola	St 05.08.2009, 08:56	
658	Výlet do Adršpachu	Jakub Sedlařík	Škola	Út 04.08.2009, 23:14	
638	Hrad Pecka	Jakub Sedlařík	Škola	Pá 31.07.2009, 10:59	
622	Výhra v krajském finále	Jakub Sedlařík	Škola	St 29.07.2009, 15:33	
281	Výlet na Rýchory	Jakub Sedlařík	Mateřská škola	Út 16.12.2008, 01:14	
280	Sběr papíru	Jakub Sedlařík	Škola	Út 16.12.2008, 01:14	
260	Pohár rozhlasu	Jakub Sedlařík	Škola	Po 15.12.2008, 18:38	

Obr. 8 - Layout administrační části aplikace

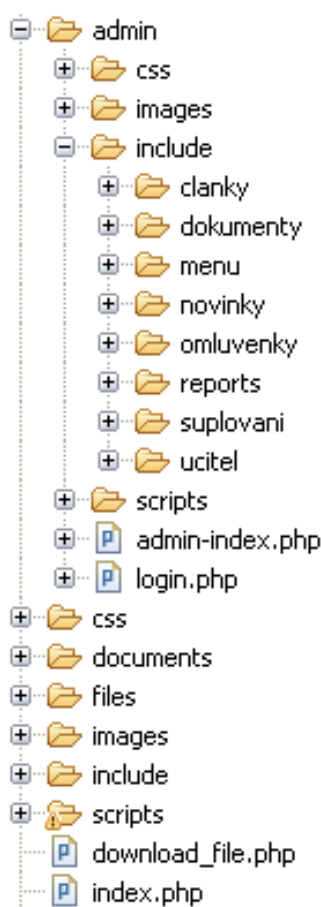
## 5.2 Implementace webové části

### 5.2.1 Adresářová struktura

Na následujícím obrázku je zobrazena adresářová struktura vytvořené aplikace. V kořenovém adresáři se nachází důležitý soubor *index.php*, který je hlavním souborem prezentační části. Do jeho kostry se pomocí PHP vkládají ostatní soubory uložené ve složce *include*. V adresáři *css* jsou soubory kaskádových stylů.

V adresáři **documents** jsou uloženy dokumenty, které přidávají uživatelé redakčního systému nebo administrátor webu. Dokumenty je možné ze stránek stáhnout pomocí souboru *download\_file.php*, který upravuje HTTP hlavičky. Díky tomu nám webový prohlížeč okamžitě nabídne okno pro uložení a nesnaží se o otevření daného souboru, které často způsobuje problémy. V dalším adresáři s názvem **images** se nacházejí grafické soubory potřebné pro vzhled prezentační části a obrázky vkládané do článků. V adresáři **scripts** jsou uloženy javascriptové soubory.

Veškeré soubory nutné k chodu administrace se nacházejí ve složce **admin**. Rozčlenění souborů do adresářů je téměř analogické s prezentační částí aplikace. V souboru *login.php* probíhá přihlašování uživatelů. Pokud přihlášení proběhne v pořádku, je uživatel přesměrován na hlavní soubor administrace s názvem *admin-index.php*. Adresář **scripts** navíc obsahuje kód potřebný pro chod editoru tinyMCE včetně inicializačních souborů, integrovaného kalendáře a v neposlední řadě frameworku jQuery.



Obr. 9 - Adresářová struktura

## 5.2.2 Integrace editoru tinyMCE

Pro jednoduchou tvorbu, případně editaci, článků je v administraci využít WY-SIWYG editor tinyMCE. Podrobnější popis je uveden v kapitole „Použité technologie“.

Jako první krok, pro integraci tinyMCE do vytvářené webové aplikace, je stažení nejnovější verze editoru a konkrétní jazykovou mutaci na oficiální webové stránce<sup>9</sup>. Po rozbalení a umístění do vhodně zvolené složky je nutné do aplikace připojit (do hlavičky HTML kódu) hlavní javascriptový soubor editoru pomocí následujícího kódu:

```
<script type="text/javascript"
src="scripts/tiny_mce/tiny_mce.js"></script>
```

Dále je třeba vytvořit inicializační skript, který obdobným způsobem připojíme do aplikace. S pomocí přehledné dokumentace jsem vytvořil dva inicializační skripty, tedy dvě různé podoby editoru. Jeden slouží pro použití při tvorbě článků a další je využit při vkládání novinek a dokumentů. Inicializačních skriptů je možno vytvořit libovolně mnoho, a tím dovolit uživatelům používat v různých částech administrace různé možnosti formátování obsahu. Následuje ukázka inicializačního skriptu:

```
tinyMCE.init({
    language : "cs",
    mode : "exact",
    elements : "elm1,elm2",
    theme : "advanced",
    skin : "o2k7",
    skin_variant : "silver",
    ...
    theme_advanced_toolbar_location : "top",
    theme_advanced_toolbar_align : "left",
    theme_advanced_statusbar_location : "bottom",
    theme_advanced_resizing : true,
    theme_advanced_resizing_max_width : "660",
    theme_advanced_resizing_min_width : "660",
    theme_advanced_resizing_max_height : "600",
```

---

<sup>9</sup> URL: <http://tinymce.moxiecode.com/download.php>

```

    theme_advanced_resizing_use_cookie : false,
    element_format : "html",
    width : "660",
    height : "300",
    entity_encoding : "named",
    content_css : "admin/css/tiny_content.css",
    external_image_list_url : "admin/scripts/tiny_image_list.php",
    relative_urls : false,
    remove_script_host : true,
    document_base_url : "http://localhost/bakalarka/bakalarka/
});

```

### 5.2.3 Funkce pro přístup do databáze Oracle

Pro přístup do databáze a zároveň pro zabezpečení proti SQL injection slouží funkce *dotaz\_b*.

```

function dotaz_b($sql, $bind, $type){
    $c = oci_connect ('bakalarka','root','//localhost/XE', 'utf8');
    $s = oci_parse($c, $sql);
    foreach ($bind as $k => $v){
        oci_bind_by_name($s,$k,$bind[$k]);
    }
    oci_execute($s);
    if ($type){
        oci_fetch_all($s,$res);
    } else {
        $res = oci_num_rows($s);
    }
    oci_close($c);
    return $res;
}

```

Pro připojení k databázi slouží funkce *oci\_connect* s přihlašovacími údaji, adresou serveru a zvolenou znakovou sadou. *Oci\_parse* ověří spojení s databází a správnost předaného SQL dotazu. Pomocí cyklu se za aliasy v poli *\$bind* dosadí reálné hodnoty a dotaz se provede. Parametrem *\$type* rozlišujeme, jestli se vykonává dotaz pro získání výsledné sady - select (funkce vrátí pole s výsledky) nebo dotazy insert, update a delete (funkce vrátí počet řádků, kterých se dotaz dotknul).

Následující část kódu ukazuje, jak se v SQL dotazu místo reálných hodnot dosazují zástupné aliasy a až posléze se naplní pole *\$bind* reálnými hodnotami (alias => hodnota). SQL dotaz a pole se poté předají jako parametry funkci *dotaz\_b*.

```
$sql="delete from temata where id_tematu= :prom";
    $bind = array(':PROM' => $_GET['smazat']);
    $pocet_radku = dotaz_b($sql, $bind, false);
```

#### 5.2.4 Přihlašování do administrace

Při přihlašování do administrace se ověřuje správnost zadaných přihlašovacích údajů na stránce *login.php*. Jelikož jsou hesla v databázi ukládány v zahešované podobě, je při porovnání hesel použita PHP funkce MD5. Pokud se ověření údajů s databází povede, je uživatel přesměrován na úvodní stránku administrace *admin-index.php* a do pole *\$\_SESSION* se uloží některé údaje potřebné pro další funkcionality aplikace např. *id\_opravneni*, *id\_ucitele*.

```
$sql =
"select ucitel.id_ucitele, ucitel.jmeno, ucitel.prijmeni,
      ucitel.nick, opraveni.nazev_opravneni, ucitel.id_opravneni
from ucitel, opraveni
where ucitel.nick= :nick and ucitel.heslo= :heslo
and ucitel.id_opravneni = opraveni.id_opravneni";

$bind = array(':NICK' => $_POST['nick'], ':HESLO' =>
      md5($_POST['heslo']) );
$vysledek = dotaz_b($sql, $bind, true);
    if(count($vysledek['JMENO'])==1){
        $_SESSION['login']=1;
        $_SESSION['id_ucitele']=$vysledek['ID_UCITELE'][0];
        $_SESSION['jmeno']=$vysledek['JMENO'][0];
        $_SESSION['prijmeni']=$vysledek['PRIJMENI'][0];
        $_SESSION['nick']=$vysledek['NICK'][0];
        $_SESSION['id_opravneni']=$vysledek['ID_OPRAVNENI'][0];

        $_SESSION['nazev_opravneni']=$vysledek['NAZEV_OPRAVNENI'][0];
        header('Location: admin-index.php');
    } else {
        $hlaska = "Špatné přihlašovací údaje";
```

Aby se zabránilo neoprávněného vstupu do administrace, je na začátku souboru *admin-index.php* tento kód:

```
if (isset($_SESSION['login'])) {  
    if($_SESSION['login']!=1){  
        header('Location: login.php');  
    }  
}  
else{  
    header('Location: login.php');  
}
```

Pokud tedy neproběhne regulérně přihlášení, nemůže *\$\_SESSION['login']* s hodnotou 1 existovat a uživatel je přesměrován zpět na stránku *login.php*.



## 6 Použité technologie

V této kapitole popíši technologie použité při vytváření redakčního systému.

### 6.1 Značkovací jazyk HTML

HTML je značkovací jazyk určený pro popis struktury webových stránek. Jeho výhodou je nezávislost na platformě. Zkratka HTML vznikla z anglického Hyper Text Markup Language. V současné době je používána verze 4.01, je to prozatím poslední verze tohoto jazyka. V budoucnu se počítá s vylepšenou verzí HTML 5.

*Pro každou verzi jazyka existuje definice pravidel DTD (Document Type Definition). Od verze 4.01 musí být odkaz na deklaraci v dokumentu uveden pomocí klíčového slova DOCTYPE. DTD definuje pro určitou verzi elementy a atributy, které lze používat. [10]*

Podstata jazyka spočívá v uzavírání částí textu do tzv. tagů, které vymezují obsah HTML elementu. Tímto se určuje význam textu – sémantika. Tagy se uzavírají mezi ostré závorky < a >. Existují dva druhy tagů, párové a nepárové. Tagy do sebe mohou být vnořovány, ale za žádných okolností se nesmí křížit.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
<html>
  <!-- Komentář -->
  <head><title>Titulek stránky</title></head>
  <body>  <!-- následuje tělo dokumentu -->
    <h1>Nadpis úrovně 1</h1>
    <p>Text odstavce, <i>kde část je kurzívou</i></p>
    <br><!-- Nepárový tag -->
  </body>
</html>
```

Při vytváření redakčního systému jsem se řídil normami HTML verze 4.01 Transitional. Pro kontrolu validity výsledného HTML kódu jsem použil validátor W3C, který je online dostupný na webových stránkách W3C.<sup>10</sup>

---

<sup>10</sup> URL: <http://validator.w3.org/>

## 6.2 Kaskádové styly – CSS

Pro popis struktury webových stránek slouží jazyk (X)HTML a pro formátování a úpravu vzhledu se naopak využívají kaskádové styly. Je dobré od sebe oddělovat obsah stránek a jejich vzhled. Vniká tak mnohem čistší a přehlednější kód, a to jak pro webdesignéry tak pro internetové vyhledávače.

Pomocí CSS můžeme pohodlně nastavovat např. rozměry, barvy, obrázky na pozadí, rámečky, druhy a velikosti písma u vybraných elementů stránek. Existují tři způsoby, jak kaskádové styly definovat:

- přímo u měněného elementu atributem *style*
- seznamem stylů zapsaným mezi tagy `<style></style>` v hlavičce HTML dokumentu
- připojením externího souboru se seznam definovaných stylů

Nejčastěji se používá třetí způsob z výše zmiňovaných. Má totiž několik nesporných výhod. Všechny definice stylů jsou uloženy v jednom souboru, a poté se stačí na tento soubor pouze odkázat. Do externího CSS souboru můžeme nadefinovat např. `h2{color: red;}`, a tím se zajistí, že všechny nadpisy druhé úrovně budou mít červenou barvu. To v konečném důsledku velmi zjednoduší práci a umožní tvůrcům webových stránek vytvořit jednotný vzhled.

Vzhledu vytvářeného redakčního systému jsem docílil využitím kaskádových stylů uložených v externích souborech. Základní rozložení stránek tzv. layoutu jsem docílil napozicováním neutrálních HTML tagů `<div>`. Tento tag spolu s tagem `<span>` nenesou na rozdíl od ostatních prvků žádný význam, a proto se jich hojně využívá právě při formátování a rozvržení webových stránek.

Pro ověření správnosti vytvořených CSS stylů jsem využil validátor W3C, který je online dostupný na webových stránkách W3C.<sup>11</sup>

---

<sup>11</sup> URL: <http://jigsaw.w3.org/css-validator/>

### 6.3 Skriptovací jazyk PHP

Skriptovací jazyk PHP je jednou z technologií, pomocí které lze vytvořit tzv. dynamické stránky.

*PHP je serverový skriptovací jazyk (server-side) navržený pro potřeby webových stránek. To znamená, že vše co PHP provádí, neprobíhá na straně klienta jako například u JavaScriptu, ale interpretuje se na straně serveru a generuje HTML (či jiný) výstup, který vidí uživatel. PHP je Open Source, tedy volně šiřitelná technologie. PHP není závislé na platformě a není vázané s žádným konkrétním serverem, může tedy běžet kdekoli. [11]*

Využití skriptovacího jazyka PHP je veliké. Jde o komplexní jazyk, s jehož pomocí je možno vyhodnocovat formuláře, vytvářet diskusní fóra, plánovací kalendáře, fotogalerie, počítadla přístupů a mnoho dalších aplikací. Jedna z nejsilnějších a nejvýznamnějších vlastností PHP je jeho podpora pro širokou škálu databází.

PHP umožňuje procedurální i objektové programování. Syntaxe jazyka PHP je s drobnými odlišnostmi podobná programovacímu jazyku C. Není nutné uvádět typ proměnné, ten se určuje v okamžiku přiřazení hodnoty do proměnné. Každá proměnná musí mít jednoznačný název. Ten v PHP začíná znakem dolaru (\$) a následuje nějakým pojmenováním. Názvy proměnných v PHP rozlišují mezi malými a velkými písmeny. Řetězce se uzavírají do uvozovek nebo do apostrofů. V souborech oddělujeme části PHP kódu tagy `<?php` a `?>`. Mezi těmito tagy vykoná server uvedené příkazy a uživateli posílá jen čistý HTML výstup.

Při tvorbě redakčního systému jsem nejčastěji využíval PHP v těchto případech:

- vkládání obsahu do kostry dokumentu
- komunikace s databází oracle, tzn. vkládání, mazání, modifikace a v neposlední řadě výpis dat
- zpracování a ošetření vstupních dat formulářů

Následující ukázka zobrazuje strukturu PHP souboru:

```

<?php
// v této části se vykonají příkazy php
session_start();
?>
<!-- pokračuje normální HTML kód -->
<html>
  <head></head>
  <body>
    <?php
      phpinfo(); //po zpracování skriptu vypíše informace o PHP
    ?>
  </body>
</html>

```

## 6.4 Webový server Apache

Při vývoji a testování dynamických stránek je potřeba mít nainstalovaný a správně nastavený webový server. Jedním z nejznámějších a nejvíce rozšířených softwarových webových serverů je Apache HTTP server. K oblíbenosti přispívá fakt, že je to server s otevřeným zdrojovým kódem a je multiplatformní. Dalším plusem je velké množství dodatečných modulů a snadná konfigurace pro spolupráci se skriptovacím jazykem.

Pro vývoj redakčního systému jsem použil VertrigoServ 2.21, který v sobě obsahuje přednastavený Apache server a modul pro skriptovací jazyk PHP verze 5.2.6. Díky VertrigoServ se nemusíme nikterak zdlouhavě věnovat konfiguraci serveru a dalších modulů a můžeme se pohodlně pustit do vývoje webových aplikací. Obdobné nástroje jsou například XAMPP, PHP Triad, Apache2Triad atd.

## 6.5 Oracle Database 10g XE edition

*Na rozdíl od vyšších verzí je Express Edition poskytována zdarma – a to jak pro vývoj, tak i produkční nasazení. Zároveň je možné ji i volně distribuovat, například s vámi vytvořenou aplikací. Je samozřejmé, že Express Edition je oproti vyšším edicím omezena jak funkčně, tak i z pohledu možného nasazení - lze ji využít pro správu maximálně 4 GB dat a bez ohledu na velikost serveru využije vždy výkon maximálně 1 CPU a 1 GB RAM. [12]*

Správa databáze probíhá přes webový prohlížeč, konzoli nebo přes pokročilou aplikaci, kterou je např. SQL Developer přímo od společnosti Oracle.

Pro spolupráci PHP s Oracle databází musíme v konfiguračním souboru *php.ini* v sekci Windows Extensions odkomentovat *extension=php\_oci8.dll* a do složky ext, nacházející se v adresáři s PHP, nahrát knihovnu *php\_oci8.dll*. Poté je již možné bez problému využívat PHP funkce pro spolupráci s databází Oracle.

## 6.6 Javascript framework jQuery

jQuery<sup>12</sup> je JavaScript framework, někdy uváděný jako JavaScript knihovna. Velice ulehčuje práci při programování klientské části aplikace. Umožní snadno vyhledávat elementy DOM, modifikovat je i vytvářet nové. Frameworku jQuery nechybí samozřejmě ani další vymoženosti – umí pracovat s událostmi, nabízí pokročilé funkce pro práci s poli, nesmí samozřejmě chybět ani podpora technologie AJAX a animací. O kvalitách jQuery nás přesvědčí fakt, že ho používají giganti jako např. Google, Dell, NBC nebo Nokia.

Tento framework patří do skupiny volně šiřitelného kódu. Kolem jQuery vyrostla od roku 2006, kdy byl vydán, obrovská komunita, která má na starosti vývoj nových verzí. Výhodou je také přehledná dokumentace s mnoha praktickými ukázkami.

Chceme-li začít tento framework používat, stačí stáhnout, nejlépe z oficiálních stránek jQuery.com, jediný soubor, který v současné době nepřesahuje 60kB a připojit ho do stávajících webových stránek.

```
<script type="text/javascript" src="jquery-1.3.2.js"></script>
```

Dalším krokem je vytvoření javascriptového souboru, ve kterém budeme vytvářet vlastní kód, a který také připojíme do webových stránek. Tímto separujeme HTML kód od JavaScriptu, zachováme tak přehlednost a umožníme lepší orientaci v kódu při pozdějších úpravách.

V neposlední řadě urychlíme načítání stránek, protože javascriptový soubor se načte pouze jednou a zůstane uchován ve vyrovnávací paměti webového prohlížeče.

---

<sup>12</sup> URL: <http://jquery.com/>

Jako příklad poslouží následující ukázka:

```
// definice odstavce s třídou odstavec a tlačítka skryj
<p class="odstavec">Nějaký text odstavce</p>
<button>skryj</button>
// v externím JS souboru
$(document).ready(function(){
  $('button').click(function(){
    $('.odstavec').hide(1000);
  });
});
```

Nejpodstatnější dva řádky kódu nám říkají, že po kliknutí na tlačítko se postupně skryjí (během 1000 ms) všechny odstavce mající přiřazenou třídu `odstavec`.

Pomocí obyčejného JavaScriptu by tento jednoduchý příklad zabral mnohem více řádků a určitě bychom narazili na nekompatibilitu webových prohlížečů.

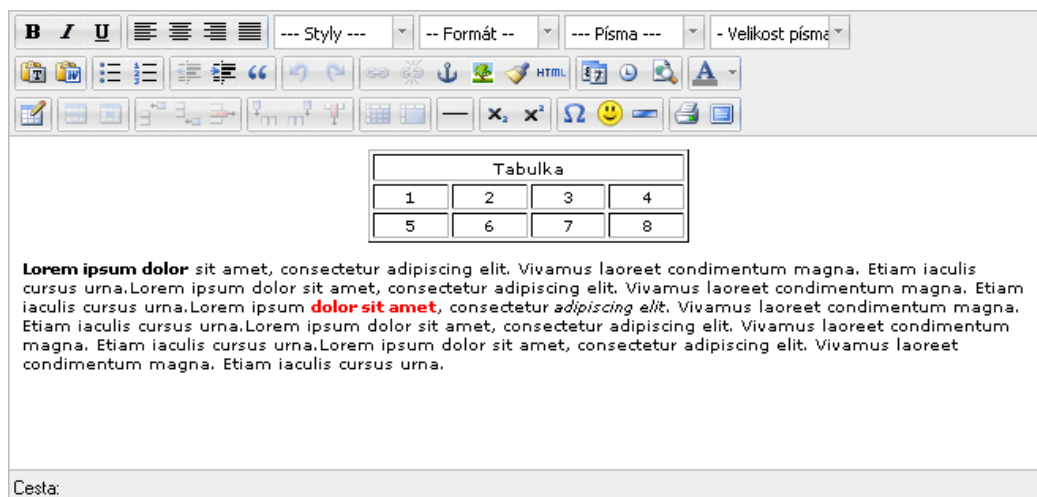
## **6.7 WYSIWYG editor *tinyMCE***

`tinyMCE` je komplexní JavaScriptový WYSIWYG editor umožňující pohodlnou tvorbu textů bez znalosti jazyka HTML. Díky podobnosti s editory ze známých kancelářských balíků jako např. MS Office a Open Office je jeho používání velice intuitivní a přehledné i pro méně zdatné uživatele. Výhodou je i široká uživatelská základna a nechybí možnost doinstalování češtiny.

Díky otevřenosti kódu a možnostem nastavení je ideálním doplňkem právě pro redakční systémy, kde hlavní náplní uživatelů je vytváření a upravování článků.

Tvůrci webových aplikací si tento editor mohou libovolně upravovat a povolit tak budoucím uživatelům jen funkce, které zbytečně neovlivní jednotný vzhled výsledné prezentace.

Na následujícím obrázku je zobrazeno prostředí editoru, u kterého jsem upravil ovládací prvky a nainstaloval češtinu. Umožňuje pohodlné vkládání tabulek, obrázků, speciálních znaků atd. Dále lze editor přepnout do celostránkového režimu, který usnadní orientaci při vytváření delších textů.



Obr. 10 - WYSIWYG editor tinyMCE

## 7 Závěr

Cílem této práce bylo objasnění problematiky týkající se tvorby redakčního systému pro základní školu. Nejdříve jsem vysvětlil, co se pod pojmem redakční systém skrývá. Poté jsem prostudoval velké množství webových stránek základních škol a některé z nich zhodnotil. Tato analýza zároveň posloužila jako vzor pro vytváření vlastní aplikace.

Díky tvorbě aplikace jsem si rozšířil obzory v používaných webových technologiích a nasbíral cenné zkušenosti. K vytvoření rozsáhlé dynamické aplikace, kterou je bezesporu i redakční systém, jsem musel zvládnout několik různorodých disciplín. Od návrhu layoutu v grafickém editoru, přes nastýlování výsledného vzhledu pomocí kaskádových stylů, návrhu a vytvoření vhodné databáze pomocí jazyka SQL až po vytvoření vlastní logiky aplikace ve skriptovacím jazyce PHP. To samozřejmě obnášelo velkou trpělivost a v neposlední řadě i časovou náročnost, pokud si uvědomíme, že výsledný kód je rozložen na několika tisících řádků.

Mnou vytvářený redakční systém se samozřejmě nemůže rovnat s projekty vyvíjenými několik let, které navíc mají rozsáhlou uživatelskou základnu.

Vytvořená aplikace však splňuje vytyčené požadavky. Především jednoduchou a přehlednou administraci s integrovaným WYSIWYG editorem, který usnadní nejčastější úkony při vytváření obsahu i méně zdatným uživatelům.

Před ostrým nasazením do provozu budu nucen zajistit podporu nejvíce využívané databáze MySQL, kterou poskytuje drtivá většina webhostingů. Chystám se na další rozvoj aplikace a na oslovení základních škol v okolí, které používají zastaralé webové stránky, abych jim nabídl využívání vytvořeného redakčního systému.



## Použitá literatura a zdroje

- [1] LUKÁŠ, Jiří. Co je to redakční systém?. *Super svět phpRS* [online]. 2005 [cit. 2009-08-11]. Dostupný z WWW: <<http://www.supersvet.cz/view.php?navezclanku=co-je-to-redakcni-system&cisloclanku=2005050501>>.
- [2] KRUG, Steve. *Nenuťte uživatele přemýšlet*. 2. aktualiz. vyd. [s.l.] : Computer press , 2006. 167 s. ISBN 80-251-1291-8.
- [3] SKLENÁŘ, Pavel . *Psklenar.cz* [online]. 2007 [cit. 2009-08-11]. Dostupný z WWW: <<http://www.psklenar.cz/pouzitelnost-internetovych-stranek/>>.
- [4] Webhosting. *Wikipedie, otevřená encyklopedie* [online]. 2009 [cit. 2009-08-11]. Dostupný z WWW: <<http://cs.wikipedia.org/wiki/Webhosting>>.
- [5] DOSTÁL, J. – MACHÁČKOVÁ, P. Školní web. *In Trendy ve vzdělávání*. Olomouc: Votobia, 2006. s. 210 - 214. ISBN 80-7220-260-X.
- [6] NEUMAJER, Ondřej. *Náležitosti školního webu: autoevaluační asistent* [online]. 2007 [cit. 2009-08-11]. Dostupný z WWW: <<http://ondrej.neumajer.cz/skolniweb/aea/>>.
- [7] VRÁNA, Jakub. Návrh databáze. *PHP triky* [online]. 2005 [cit. 2009-08-11]. Dostupný z WWW: <<http://php.vrana.cz/navrh-databaze.php>>.
- [8] Zezulka, Jaroslav. Konceptuální modelování a návrh databáze [online]. 2004-2005 [cit. 2009-04-30]. Dostupný z WWW: <[http://www.fit.vutbr.cz/study/courses/DSI/public/pdf/nove/2\\_kmod.pdf](http://www.fit.vutbr.cz/study/courses/DSI/public/pdf/nove/2_kmod.pdf)>.
- [9] BI Experts. Normalizace [online]. c2008 [cit. 2009-08-11]. Dostupný z WWW: <<http://www.biexperts.cz/index.php/component/content/article/18-ctsql/31-arnormalization.html>>.

- [10] HyperText Markup Language. *Wikipedie, otevřená encyklopedie* [online]. 2009 [cit. 2009-08-11]. Dostupný z WWW: <<http://cs.wikipedia.org/wiki/HTML>>.
- [11] Webtvorba.cz : Úvod do PHP [online]. c2004 [cit. 2008-02-28]. Dostupný z WWW: <<http://www.webtvorba.cz/php/uvod-do-php.html>>.
- [12] Oracle. *Oracle Database 10g Express Edition* [online]. c1995-2009 [cit. 2009-08-11]. Dostupný z WWW: <[http://www.oracle.com/global/cz/database/express\\_edition.html](http://www.oracle.com/global/cz/database/express_edition.html)>.

## Příloha A - Postup instalace aplikace

Protože je aplikace vytvořena ve skriptovacím jazyce PHP, je nutné mít na počítači, na kterém se chystáme aplikaci zprovoznit, nainstalovaný webový server Apache s PHP. Nejlépe je stáhnout již přednastavený balík, kterých je na internetu k dispozici velká řada. Dále je potřeba mít nainstalovanou a nakonfigurovanou databázi Oracle. Obojí je zdarma ke stažení a používání na následujících odkazech:

<http://vertrigo.sourceforge.net/>

<http://www.oracle.com/technology/software/products/database/xe/index.html>

### Databáze

Po nainstalování uvedené databáze Oracle, se pomocí webového prohlížeče na adrese <http://127.0.0.1:8080/apex> přihlásíme pod účtem *system* a heslem zvoleným při instalaci do webové administrace. Pokračujeme do sekce *Administration* a *Database Users* kde vytvoříme nového uživatele. Tento uživatel musí mít oprávnění vytvářet databázové objekty typu *table*, *view*, *procedure*, *sequence*, *trigger*.

Dalším krokem je vytvoření databázových objektů pomocí souboru *db\_install.sql*, který se nachází ve složce *Databáze*. Zároveň je vytvořen v databázi uživatel *admin* s heslem *root*, který má logicky administrátorské oprávnění.

### Aplikace

Celý obsah složky *Aplikace* zkopírujeme do kořenového adresáře webového serveru (jedná se např. o složky *www*, *htdocs*). Kvůli propojení databáze s aplikací ještě musíme nastavit přihlašovací údaje k nově vytvořenému databázovému profilu. To se provede ve složce *include* v souboru *function*.

```
oci_connect ('uzivatel', 'heslo','//localhost/XE', 'utf8');
```

Ve funkci *oci\_connect* se za parametry *uzivatel* a *heslo* dosadíme reálné přístupové údaje.

Po úspěšné instalaci a konfiguraci je aplikace dostupná skrze webový prohlížeč na adrese <http://localhost>.

## **Příloha B - Dotazník**

**Název školy**

**Použitý redakční systém**

*pokud používáte vlastní řešení administrace stránek, uveďte "vlastní řešení"*

**Spolupráce s administrátorem (tvůrcem) webových stránek**

*jak častá je potřeba zásahu administrátora do webových stránek, jakým způsobem funguje spolupráce*

**Kdo má přístup do administrace web. stránek**

administrátor, vedení školy, pedagogický sbor, žáci, rodiče

**Jaký druh obsahu lze pomocí administrace přidávat a upravovat**

novinky, články, dokumenty, fotografie, rozvrhy, suplování, ankety

**Jak často aktualizujete své webové stránky**

**Má vaše škola zavedenou webovou aplikaci, kde žáci a jejich rodiče mají přehled o studiu**

*jedná se např. o modul Bakalářů, iŠkola a obdobné, v kladném případě uveďte název*

**Mají žáci možnost přihlášení na webové stránky**

*umožňuje odkrytí neveřejné části stránek, studijních materiálů, známkování...*

Ano, Ne

**Jaký je zájem ze strany žáků a rodičů o webové stránky školy, jaké jsou ohlasy známkování jako ve škole**

**Jaké shledáváte výhody komunikace se žáky a jejich rodiči právě prostřednictvím webových stránek školy**

**Chystá se v budoucnu vaše škola na úpravu či úplnou změnu webových stránek v kladném případě uveďte, čeho by se měly inovace týkat**