

# **Redakční systém pro písničkáře**

Content management system for Songsters

Martin Šuránek

---

Bakalářská práce  
2008



Univerzita Tomáše Bati ve Zlíně  
Fakulta aplikované informatiky

---

Univerzita Tomáše Bati ve Zlíně  
Fakulta aplikované informatiky  
Ústav aplikované informatiky  
akademický rok: 2007/2008

## ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Martin ŠURÁNEK**  
Studijní program: **B 3902 Inženýrská informatika**  
Studijní obor: **Informační technologie**  
  
Téma práce: **Redakční systém pro písničkáře**

Zásady pro vypracování:

1. Analýza současného stavu problematiky.
2. Výběr vhodného programového prostředí.
3. Návrh struktury redakčního systému.
4. Vytvoření funkčního redakčního systému.
5. Implementace a otestování vytvořené aplikace.

Rozsah práce:

Rozsah příloh:

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

Seznam odborné literatury:

1. PROKOPOVÁ, Z.: Databázové systémy MySQL+PHP. FAI UTB Zlín, s. 126, 2006, Vysokoškolská skripta. ISBN 80-7318-486-9.
2. JESUS, Castagnetto, et al. Programujeme profesionálně : PHP. Robert FE Shaw. 2004. autoriz. vyd. Praha : Computer Press, 2004. 656 s., CD-ROM. ISBN 80-7226-310-2.
3. LUBOSLAV, Lacko. SQL : hotová řešení. Praha : Computer press, 2003. 296 s. ISBN 80-7226-975-5.
4. JavaScript a Ajax : Moderní programování webových aplikací. 1. vyd. Praha : Computer Press, a.s., 2007. 360 s. ISBN 978-80-251-1824-5.

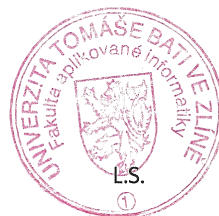
Vedoucí bakalářské práce: **doc. Ing. Zdenka Prokopová, CSc.**  
Ústav aplikované informatiky

Datum zadání bakalářské práce: **20. února 2008**

Termín odevzdání bakalářské práce: **5. května 2008**

Ve Zlíně dne 20. února 2008

  
prof. Ing. Vladimír Vašek, CSc.  
*děkan*



  
doc. Ing. Ivan Zelinka, Ph.D.  
*ředitel ústavu*

## ABSTRAKT

Bakalářská práce objasňuje pojem redakční systém a zabývá se analýzou několika volně dostupných redakčních systémů. Dále popisuje zvolený software pro tvorbu redakčního systému. Software pro editaci PHP byl zvolen PSPad a PhpED, pro práci s grafikou PhotoFiltre. Dále byl zvolen softwarový balík PHP Home, který obsahuje webový, databázový server a skriptovací jazyk PHP. Popsána je problematika vytvoření redakčního systému pro písničkáře. Přehled jeho struktury a struktury jeho databáze. Je zde objasněna instalace tohoto systému, jeho implementace a otestování v praxi. Zmíněno je i několik problémů, na které se v průběhu jeho testování narazilo.

Klíčová slova: Redakční systém, web, plugin, FTP, server, PHP, MySQL, JavaScript

## ABSTRACT

This bachelor expounds the term the content management system and inquires into some free available content management systems. It describes software, which is chosen for production of the content management system. PSPad and PhpED were chosen as a software for editing PHP, PhotoFiltre was chosen for work with graphic. The software package PHP Home was elected as well. It consists of web-server, database-server and script language. The problems of making the content management system for songsters is described here. It's about its structure and structure of its database. It clears up the instalation of this system, its implementation and testing in practice. There is a mention of some problems, which I met during its testing.

Keywords: Content Management System, web, plugin, FTP, server, PHP, MySQL, JavaScript

Zde bych rád poděkoval vedoucí bakalářské práce, paní doc. Ing. Zdence Prokopové CSc. za vstřícnost a rady, které mně poskytovala v průběhu vypracování této bakalářské práce.

Prohlašuji, že jsem na bakalářské práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků, je-li to uvolněno na základě licenční smlouvy, budu uveden jako spoluautor.

Ve Zlíně

.....  
Podpis diplomanta

**OBSAH**

<b>ÚVOD</b> .....	<b>9</b>
<b>TEORETICKÁ ČÁST</b> .....	<b>10</b>
<b>1 ANALÝZA SOUČASNÉHO STAVU PROBLÉMU</b> .....	<b>11</b>
1.1 Co JE REDAKČNÍ SYSTÉM? .....	11
1.1.1 Co je modul? .....	11
1.1.2 Co je to šablona? .....	11
1.2 Co POTŘEBUJI K PROVOZU REDAKČNÍHO SYSTÉMU .....	11
1.2.1 Apache.....	12
1.2.2 MySQL.....	12
1.2.3 PHP .....	12
1.3 SROVNÁNÍ NĚKOLIKA REDAKČNÍCH SYSTÉMU ZDARMA.....	12
1.3.1 WordPress .....	12
1.3.1.1 WordPress MU .....	13
1.3.1.2 Pluginy .....	13
1.3.1.3 Jazyková lokalizace .....	13
1.3.1.4 Vývoj.....	13
1.3.1.5 Výhody a nevýhody.....	13
1.3.2 Drupal.....	14
1.3.2.1 Pluginy .....	14
1.3.2.2 Jazyková lokalizace .....	14
1.3.2.3 Vývoj.....	14
1.3.2.4 Výhody a nevýhody.....	15
1.3.3 CMS Made Simple.....	15
1.3.3.1 Pluginy .....	15
1.3.3.2 Jazyková lokalizace .....	15
1.3.3.3 Vývoj.....	15
1.3.3.4 Výhody a nevýhody.....	16
1.3.4 RS2.....	16
1.3.4.1 Pluginy .....	16
1.3.4.2 Jazyková lokalizace .....	16
1.3.4.3 Vývoj.....	16
1.3.4.4 Výhody a nevýhody.....	17
1.3.5 phpRS.....	17
1.3.5.1 Pluginy .....	17
1.3.5.2 Jazyková lokalizace .....	17
1.3.5.3 Vývoj.....	17
1.3.5.4 Výhody a nevýhody.....	18
<b>2 VOLBA VHODNÉHO SOFTWARE</b> .....	<b>19</b>

2.1	PHP HOME EDITION .....	19
2.2	PSPAD.....	19
2.3	NUSPHERE PHPED.....	20
2.4	PHOTOFILTRE .....	21
<b>PRAKTICKÁ ČÁST .....</b>		<b>23</b>
<b>3</b>	<b>ANALÝZA STAVU PROBLEMATIKY .....</b>	<b>24</b>
3.1	POŽADAVKY ZADAVATELE: .....	24
3.1.1	Základní pilíře webu.....	24
3.1.2	Práva uživatelů .....	24
3.1.3	Registrace uživatelů .....	24
3.1.4	Profily uživatelů .....	25
3.1.5	Novinky .....	25
3.1.6	Články .....	25
3.1.7	Koncerty .....	25
3.1.8	Knihy návštěv .....	25
3.1.9	Hudba .....	25
3.1.10	Texty.....	26
3.1.11	Fotky.....	26
3.1.12	Administrační rozhraní.....	26
<b>4</b>	<b>NÁVRH DATABÁZOVÉ STRUKTURY REDAKČNÍHO SYSTÉMU.....</b>	<b>27</b>
4.1	TABULKY V DATABÁZI .....	27
4.1.1	Tabulka uživatelé .....	27
4.1.2	Tabulka články .....	28
4.1.3	Tabulka komentáře.....	29
4.1.4	Relace v databázi.....	31
<b>5</b>	<b>POPIS VYTVOŘENÉHO REDAKČNÍHO SYSTÉMU .....</b>	<b>32</b>
5.1	ROZLOŽENÍ WEBOVÉ APLIKACE .....	32
5.1.1	Levá navigace.....	32
5.1.2	Pravá navigace.....	33
5.1.3	Horní menu.....	33
5.1.4	Hlavní panel .....	33
5.2	UŽIVATELSKÉ MENU .....	34
5.3	ADMINISTRAČNÍ ROZHRAŇÍ .....	34
5.3.1	Administrační menu .....	34
<b>6</b>	<b>IMPLEMENTACE A OTESTOVÁNÍ VYTVOŘENÉ APLIKACE .....</b>	<b>37</b>
6.1	POŘÍZENÍ DOMÉNY A WEBOVÉHO PROSTORU .....	37
6.2	INSTALACE REDAKČNÍHO SYSTÉMU.....	37
6.3	TESTOVÁNÍ VYTVOŘENÉ APLIKACE .....	38
6.3.1	Diakritika.....	38
6.3.2	Upload uživatelských fotografií .....	38
6.3.3	Vyhledávání.....	39
6.3.4	Novinky .....	39

---

6.3.5 Spam v knize návštěv .....	40
<b>ZÁVĚR .....</b>	<b>41</b>
<b>ZÁVĚR V ANGLIČTINĚ .....</b>	<b>42</b>
<b>SEZNAM POUŽITÉ LITERATURY .....</b>	<b>43</b>
<b>SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK .....</b>	<b>44</b>
<b>SEZNAM OBRÁZKŮ .....</b>	<b>45</b>
<b>SEZNAM TABULEK .....</b>	<b>46</b>
<b>SEZNAM PŘÍLOH .....</b>	<b>47</b>



## ÚVOD

Cílem bakalářské práce a mým úkolem byla tvorba redakčního systému pro písničkáře. Označením písničkáři jsou myšleni lidé, kteří se zajímají o hudbu a píšou si vlastní písně, ke kterým si skládají vlastní hudbu a rádi se o svou tvorbu podělí. Vznik tohoto nápadu, vytvořit redakční systém pro písničkáře a tím je seskupit, patří několika lidem z okolí Strážnicka a Kyjovska, kde pořádají většinu svých koncertů a často se potýkali s problémem komunikace. Napadlo je tedy sjednotit se na jednom webu, kde by se mohli denně „setkávat“, prezentovat tam svoji tvorbu, zveřejňovat informace o svých koncertech, zviditelnit se a možná poznat nové, další písničkáře. S jedním z iniciátorů tohoto nápadu se osobně znám a jsem s ním v častém kontaktu. Proto jsem prezentovaný redakční systém vytvářel přesně podle jeho přání.

Redakční systémy jsou na internetu publikovány dvěma způsoby, jedny z nich jsou volně dostupné a ty druhé jsou komerční, tudíž za ně musíme zaplatit. Jedná se v podstatě o systémy, které spravují obsah webu. Jejich největší předností je, že uživatel nemusí mít téměř žádné znalosti potřebné pro tvorbu webových stránek. Přesto je díky redakčnímu systému může vytvořit. Mezi základní funkce těchto systémů patří publikování článků, ale většina redakčních systémů obsahuje i jiné funkce. Tyto systémy jsou často rozšiřitelné pomocí různých modulů. Pokud tedy uživatel chce ve svém systému nějakou funkci, která v něm není obsažena a onu funkci redakční systém nabízí jako přídatný modul, nezbyvá než modul doinstalovat a pak požadovanou funkci redakční systém obsahuje.

Podrobným rozбором několika redakčních systémů dostupných zdarma se zabývá první kapitola. Druhá kapitola se věnuje výběru vhodného softwaru pro tvorbu redakčního systému. Analýza požadavků zadavatele je uvedena ve třetí kapitole. Čtvrtá kapitola je věnovaná databázové struktuře vytvořeného redakčního systému. V páté kapitole následuje popis vytvořeného redakčního systému. Poslední kapitola se zabývá implementací a testováním vytvořené aplikace, kde jsou mimo jiné zmíněny problémy, ke kterým během tvorby redakčního systému docházelo.

## **I. TEORETICKÁ ČÁST**

## 1 ANALÝZA SOUČASNÉHO STAVU PROBLÉMU

### 1.1 Co je redakční systém?

Tyto systémy se řadí mezi CMS (Content Management System), v českém znění Systém pro správu obsahu. Umožňuje velmi jednoduché vytvoření obsahu internetových stránek uživatelům s minimální znalostí technologií jako je HTML, CSS, PHP, MYSQL a dalších. Vzhled takového webu bývá většinou řešen pomocí šablon, které určují jak má web vypadat. Díky tomu je snadné měnit vzhled aniž by se nám změnil obsah. Redakční systémy jsou k sehnání na internetu jak volně dostupné, tak i komerční verze. Liší se v mnoha kritériích i v podpoře jazyků, některé totiž češtinu nepodporují. Většina redakčních systémů na trhu je rozšiřitelná pomocí různých modulů. Zákazník si tak může koupit redakční systém s obsahem služeb, které právě on potřebuje a pokud časem zjistí, že by využil i jiné nabízené moduly, může si je dokoupit. Ovšem v případě, že jde o volně dostupný systém, samozřejmě nakupování odpadá, protože pluginy k těmto systémům jsou zdarma.

#### 1.1.1 Co je modul?

Pomocí přidávání modulů přibývají vašemu redakčnímu systému nové funkce. Mohou to být například modul pro vytvoření internetového obchodu, diskusní fórum či fotogalerie.

#### 1.1.2 Co je to šablona?

Patrně největší výhodou šablon je to, že se uživatel nemusí starat o to, jak bude jeho článek vypadat, kam se vloží a jestli nebude narušovat vzhled webových stránek. Šablona za něj určí, jak bude vypadat nadpis článku, kde bude vložen atd. Většina redakčních systémů nabízí více šablon a umožňuje tak uživateli, aby si vzhled svého webu určil sám.

### 1.2 Co potřebuji k provozu redakčního systému

Provoz většiny redakčních systémů se neobejde bez webového serveru, databázového serveru a skriptovacího jazyka. Velmi časté a oblíbené jsou kombinace Apache, MySQL, PHP.

### 1.2.1 Apache

Jedná se o softwarový HTTP server, který je volně dostupný, funguje téměř na všech operačních systémech a je jedním z nejpoužívanějších HTTP serverů.

### 1.2.2 MySQL

Jedná se o databázový systém, který vznikl ve Švédsku a je vydáván pod dvěma licencemi bezplatně a komerčně. Komunikace s tímto serverem probíhá pomocí jazyka SQL a jedná se o jeden z nejpoužívanějších databázových serverů.

### 1.2.3 PHP

Jedná se o skriptovací programovací jazyk, který je především určen pro programování dynamických internetových stránek. Jedním z určujících rysů tohoto jazyka je, že se provádí na straně serveru a k uživateli se tak dostane až výsledek. Výhodou je obsah knihoven pro práci s textem, grafikou se soubory a v neposlední řadě s databázovými servery.

## 1.3 Srovnání několika redakčních systému zdarma

### 1.3.1 WordPress

WordPress patří k nejoblíbenějším blogovacím systémům. Mezi jeho přednosti patří ohled na estetiku a dodržování webových standardů a široká míra použitelnosti webu. WordPress se používá na více než stovkách tisíc webů. Říká se o něm, že je skoro dokonalý a těší se velké podpoře, která se odhaduje na několik stovek tisíc uživatelů. Tito uživatelé pak vytváří nové pluginy (rozšíření), templaty (různé vzhledy) a případně i jazykové mutace (možnost mít rozhraní ve svém rodném jazyce). Oficiálně se tento systém vydává jen v anglické jazykové verzi. Ale díky komunitě, která vytváří rozšíření pro tento systém, si může uživatel stáhnout českou jazykovou lokalizaci a mít tak tento systém česky. WordPress je napsán v PHP a jako úložiště dat používá MySQL databázi. Publikován je pod licencí GPL.

### **1.3.1.1 WordPress MU**

Zkratka MU je z anglického Multi User, jedná se o klasický WordPress, který je upraven. Poskytuje možnost vytvořit na jedné instalaci několik jednotlivých webů. Každý z těchto webů si můžou spravovat různí uživatelé a navíc nabízí možnost mít dohled nad všemi těmito weby jednou osobou.

### **1.3.1.2 Pluginy**

Po instalaci základní verze máme k dispozici jenom dva pluginy. První z nich se nazývá *Akismet* a ten má za úkol starat se o to aby nedocházelo k nevyžádaným komentářům. Brání náš systém proti tomu, aby se mezi komentáři vyskytoval spam. Další plugin nese název *Hello Dolly* a ten nám poskytuje možnost zobrazování náhodně generovaných citátů v administraci. Pokud si však nainstalujeme neoficiální českou verzi tohoto systému, přibude nám ještě jeden plugin a ten nám nabízí používání textového editoru Taxy!. Další pluginy si můžeme stáhnout na domovských stránkách organizace WordPress.

### **1.3.1.3 Jazyková lokalizace**

Jak bylo zmíněno už výše, WordPress se dodává v základní verzi jenom v angličtině. Nabízí se však i česká verze tohoto oblíbeného systému, počeštění se týká jak uživatelské části tak i administrační.

### **1.3.1.4 Vývoj**

Systém vznikl někdy okolo roku 1994. Tvůrci redakčního systému Movable se po určitých neshodách rozdělili. Mezi vývojáře, kteří se odtrhli, patřili Matt Mullenweg a Ryan Boren. Právě tyto dva lidé začali s vývojem nového redakčního systému WordPress. Z čeho vznikl název WordPress není nikde uvedeno.

### **1.3.1.5 Výhody a nevýhody**

Výhody.

- Jedná se o velmi kvalitně zpracovaný redakční systém
- Těší se velké podpoře uživatelů a proto není problém obrátit se ně s otázkami.
- Ohromné množství dostupného rozšíření.

Nevýhody:

- Českou jazykovou verzi je nutno doinstalovat.
- Základní verze obsahuje málo rozšíření.

### **1.3.2 Drupal**

Drupal se také těší velké oblibě uživatelů a to jak v zahraničí, tak i u nás. Jako výhodu tohoto systému můžeme považovat přehlednost zdrojového kódu. Systém umožňuje tvorbu blogů, publikačních systémů a dokonce i internetových obchodů. Drupal je naprogramován v jazyce PHP. A jako úložiště dat používá MySQL a PostgreSQL databáze. Tento systém je šířen pod GPL licenci.

#### **1.3.2.1 Pluginy**

Po nainstalování základní verze systému Drupal nás určitě překvapí počet třiadvaceti pluginů, což je neobvykle velký počet. Mezi tyto rozšíření patří například RSS, vyhledávání, správa uživatelů, ankety, statistiky nebo kontaktní formuláře. K zajištění dalších rozšíření slouží centrální registr webů, které fungují právě na Drupalu. Za jeden z nejzajímavějších modulů, který je obsažen v základní verzi, se dá považovat Throttle. Tento plugin naskýtá možnost vypnutí vybraných částí systému při přetížení.

#### **1.3.2.2 Jazyková lokalizace**

Množství jazyků, které Drupal po základní instalaci nabízí, je opravdu úctyhodné. Jedná se o sedmatřicet jazyků, mezi kterými můžeme nalézt češtinu i slovenštinu. Do těchto 37 jazyků můžete převést uživatelskou i administrační část.

#### **1.3.2.3 Vývoj**

Drupal vytvořil Dries Buytaert. Původně byl tento systém hodně omezený a neposkytoval tolik možností jako v současnosti. Až v roce 2001 začal autor provádět velké změny v jeho systému, ten se tak dočkal značných vylepšení a došlo i k uvolnění zdrojového kódu pro veřejnost. Tento systém se stále vyvíjí a těší se stále větší oblibě.

#### *1.3.2.4 Výhody a nevýhody*

Výhody:

- Součást češtiny přímo v základní verzi
- Nespočetné množství pluginů
- Propracovaná práce s komentáři

Nevýhody:

- Základní šablony nejsou příliš kvalitně zpracované

### **1.3.3 CMS Made Simple**

Ve svém názvu nese informaci o tom, že jde o jednoduše vytvořený systém. Těší se u nás velké oblibě a je produkován pod GNU/GPL licencí. Je naprogramován v jazyce PHP a pro ukládání dat může použít jak databázi MySQL, tak i PostgreSQL. Systém Made Simple používá dva editory: wysiwyg editor a FCKeditor. Má výborně zpracovanou správu uživatelů, práva můžete přidělovat nejen jednotlivcům, ale i celým skupinám. Množství pluginů které si můžete stáhnout se dá pokládat za obrovské. Kvalitně zpracovaná je také práce se vzhledem systému.

#### *1.3.3.1 Pluginy*

Po instalaci tohoto systému máme k dispozici základní rozšíření, které jsou k provozu běžného redakčního systému (myslím) dostatečné. Zajímavostí tohoto systému je, že obsahuje tzv. komponenty jako například mapu stránek, a různé Google rozšíření.

#### *1.3.3.2 Jazyková lokalizace*

Součástí základní instalace tohoto systému je nabídka jednatřiceti jazyků a mezi ně patří i čeština. Přeloženy jsou obě části systému.

#### *1.3.3.3 Vývoj*

Autorem tohoto systému je Ted Kulp a s vývojem začal v roce 2004. V současnosti se vývojem tohoto systému zabývá již devítičlenná skupina. Neustále tak tento systém vyvíjí a zdokonaluje.

#### *1.3.3.4 Výhody a nevýhody*

Výhody:

- Široká nabídka jazyků včetně češtiny
- Možnost výběru databáze
- Výborně zpracovaná práce s právy

#### **1.3.4 RS2**

Tento systém pochází z České republiky. Zaměřuje se spíše na tvorbu blogů. Největším plusem tohoto systému je jeho jednoduchost, kvalitní zpracování a detailní popis nápovědy, která vyřeší spoustu problémů. Systém je naprogramován v PHP jazyce a jako úložiště dat mu slouží MySQL databáze. RS2 je publikován pod GPL licencí.

##### *1.3.4.1 Pluginy*

RS2 disponuje běžnou výbavou pluginů jak je tomu u jiných cms. Velmi kvalitně je zde zpracovaný odběr informací z webu, je rozdělen mezi tři kanály, nejdříve kanál pro odběr komentářů, po něm následuje kanál pro odběr článků a nakonec kanál pro odběr novinek. Uživatel si tak může vybrat jaké informace si chce ze systému odebírat. Další kvalitně zpracované rozšíření jsou statistické údaje, uživatel si může nechat vypsat velké množství informací o návštěvnicích jeho webu. RS2 se může pochlubit taky velmi povedenou fotogalerií.

##### *1.3.4.2 Jazyková lokalizace*

Samozřejmostí tohoto systému je, že je plně v češtině, ale pokud jste očekávali jiné jazykové mutace, tak jste se zmýlili. Žádné jiné jazykové mutace v tomto systému nenaleznete.

##### *1.3.4.3 Vývoj*

Redakční systém RS2 s vývojem začal pro veřejnost v roce 2004, autor tohoto systému si přeje zůstat v anonymitě a vystupuje pod svou přezdívkou juneau.



#### *1.3.4.4 Výhody a nevýhody*

Výhody:

- Přehledné zpracování
- Systém je v češtině
- Kvalitně zpracovaná nápověda

Nevýhody:

- Nízká podpora šablon, v základní instalaci je jen jedna

### **1.3.5 phpRS**

PhpRS je další český volně dostupný redakční systém. Dbá hlavně na to, aby byl pro své uživatele přívětivý a výkonný. U českých uživatelů se těší ohromné oblibě. Díky jeho jednoduchosti se dá s jeho pomocí vytvořit nejen jednoduchý web, ale i složité webové aplikace. Velmi snadná je úprava šablon, kvalitní je řešení uživatelských práv. PhpRS používá skriptovací jazyk PHP a jako úložiště dat slouží MySQL databáze. Systém je produkován pod GPL licencí.

#### *1.3.5.1 Pluginy*

Počet pluginů v základní instalaci je značně omezený, až nedostačující, ale na internetu je jich ke stažení obrovské množství. Pro editaci textu máme na výběr ze tří editorů, což je příjemné.

#### *1.3.5.2 Jazyková lokalizace*

Kompletní česká lokalizace PhpRS je samozřejmostí. Zajímavostí je možnost stáhnout si tento systém ve všech třech kódováních: iso 8859-2, UTF-8 a windows1250. Systém navíc obsahuje slovenštinu.

#### *1.3.5.3 Vývoj*

Vývoj započal v roce 2001. Autorovi tohoto systému nevyhovoval žádný dostupný redakční systém a tak vytvořil phpRS. Vydal verzi, která byla volně dostupná. Po vydání

začali s autorem komunikovat uživatelé, navrhovali různá zlepšení a vytvořili tak komunitu tohoto systému.

#### *1.3.5.4 Výhody a nevýhody*

Výhody:

- Česká lokalizace a možnost výběru ze tří kódování.

Nevýhody:

- Nepřehledná instalace.

## 2 VOLBA VHODNÉHO SOFTWARE

Správná volba softwaru je pro práci jeden z nejdůležitějších faktorů. Každý by si měl zvolit software, který vyhovuje právě jemu a ušetří mu tak spoustu práce. Naopak špatně zvolený software nám může práci hodně znepríjemnit, mnohdy nás od práce může i odradit. Proto není na škodu dát si na čas a vyzkoušet spoustu softwaru, než si najdeme ten nejvhodnější.

V našem případě šlo o výběr webového serveru, databázového serveru, editoru skriptovacího jazyka a softwaru pro práci s grafikou.

### 2.1 PHP Home Edition

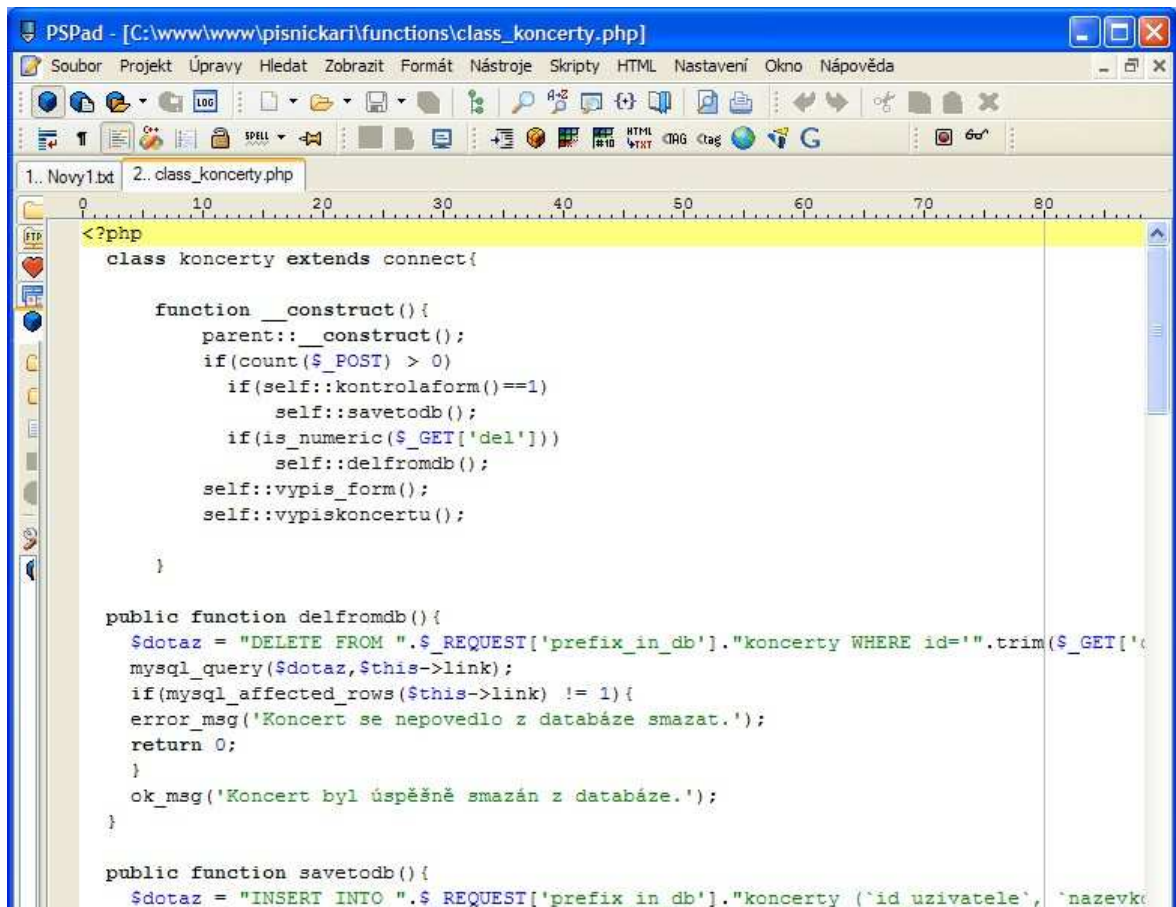
Jedná se o balíček který obsahuje jak webový server, tak databázový server a navíc PHP jazyk. Tento software je na internetu volně dostupný. Obsah balíku:

- Web Server Apache 2.0.50
- MySQL 4.0.20d
- PHP 5
- phpMyAdmin 2.6.x

Během instalace se nastaví adresář s webem, heslo pro přístup k databázi a podobné potřebné údaje. Jediné co je možné vytknout je průběh instalace, při které se nezobrazují popisky tlačítek a tím je instalace hodně nepřehledná až intuitivní. Ale to snad v novějších verzích programu odpadne.

### 2.2 PSPad

Tento editor je freeware a nabízí nám volbu kódování, ve kterém budeme pracovat. Práci s více soubory nám ulehčuje možnost záložek, mezi kterými se můžeme snadno přepínat. Důležité je taky číslování řádků, díky kterému se nám snadno hledají chyby. Nejdůležitější je samozřejmě zvýrazňování syntaxe, které bohužel není tak propracované jako např. u PhpEDu.



```
<?php
class koncerty extends connect{

    function __construct(){
        parent::__construct();
        if(count($_POST) > 0)
            if(self::kontrolaform()==1)
                self::savetodb();
            if(is_numeric($_GET['del']))
                self::delfromdb();
        self::vypis_form();
        self::vypiskoncertu();
    }

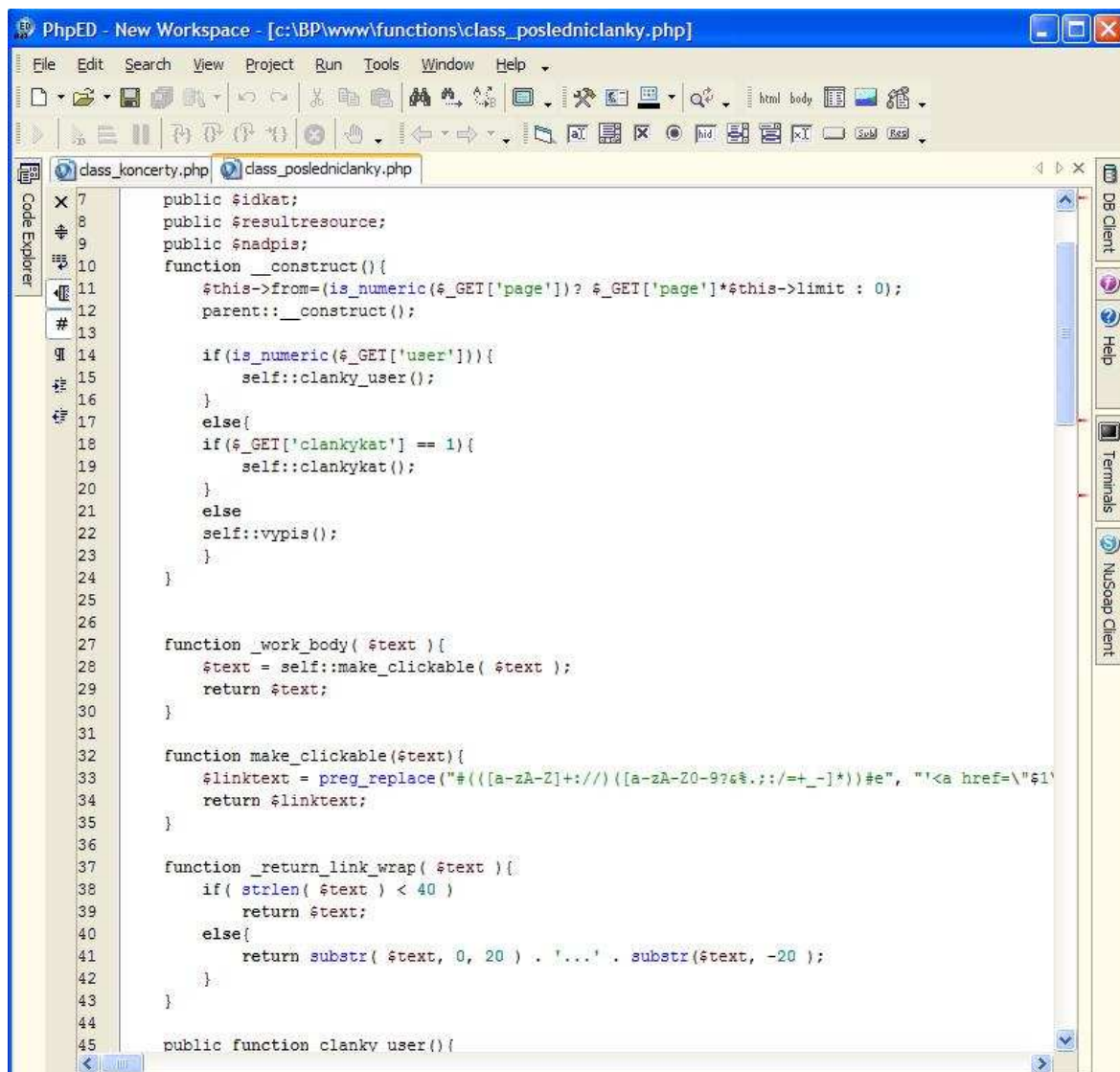
    public function delfromdb(){
        $dotaz = "DELETE FROM ".$_REQUEST['prefix_in_db']."koncerty WHERE id='".$_trim($_GET['del'])."'";
        mysql_query($dotaz,$this->link);
        if(mysql_affected_rows($this->link) != 1){
            error_msg('Koncert se nepovedlo z databáze smazat.');
```

Obrázek 1-PSPad

## 2.3 Nusphere PhpED

Tento editor se dá použít nejen pro PHP, ale i spoustu jiných jazyků. Další předností je, že tento editor v sobě obsahuje rozsáhlou nápovědu pro PHP, MYSQL, HTML a JavaScript.

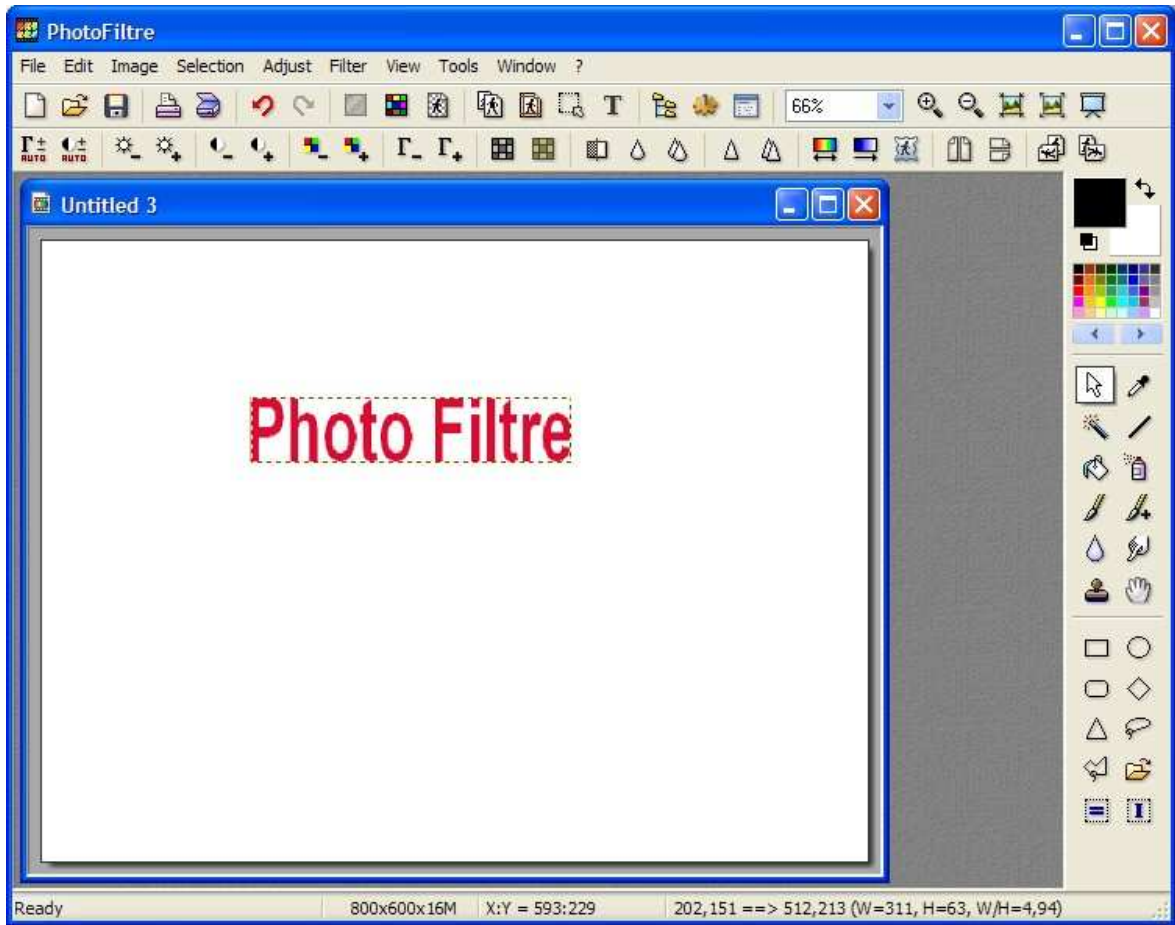
Používá se nejen pro editaci PHP, ale i pro editaci CSS a HTML. V tomto editoru máme na výběr obrovské množství kódování, které budeme používat. Navíc tento editor rozlišuje verze PHP, ve kterých chceme pracovat. To se pak projeví při zvýrazňování syntaxe. Velmi podařené je doplňování funkcí při psaní zdrojového kódu, které nejen eliminují překlepy, ke kterým může během psaní kódu docházet, ale navíc dostaneme spoustu informací o dané funkci. Jediné co je na tomto editoru „nepříjemné“, že není freeware. Zdarma je k dispozici pouze časově omezená verze, a proto má člověk jen 30 dní na vyzkoušení tohoto editoru.



Obrázek 2 - NUSpherePhpED

## 2.4 PhotoFiltre

Pro jednodušší práce s grafikou byl používán klasický pomocník z windows a to malování, ale občas došlo k tomu, že bylo potřeba udělat něco, na co malování nestačilo. Proto bylo nutné hledat nějakého jednoduchého pomocníka pro práci s grafikou, který by byl volně dostupný. Nakonec byl zvolen PhotoFiltre - je to jednoduchý freeware program, se kterým se dají mnohdy velmi rychle a snadno upravovat grafické objekty. Nabízí také širokou množství nástrojů, které při práci oceníme. Po základní instalaci tento software neobsahuje češtinu, a proto si musíme stáhnout plugin z domovských stránek tohoto softwaru.



Obrázek 3 - Photofiltre

## **II. PRAKTICKÁ ČÁST**

### 3 ANALÝZA STAVU PROBLEMATIKY

Jak již bylo napsáno, cílem bakalářské práce bylo vytvoření redakčního systému pro písničkáře, který by obsahoval funkce potřebné právě pro ně.

#### 3.1 Požadavky zadavatele:

##### 3.1.1 Základní pilíře webu

Mezi základní pilíře redakčního systému by podle požadavků zadavatele měly patřit registrace uživatelů, profily uživatelů, novinky, články, koncerty, kniha návštěv, hudba, texty, fotky a v neposlední řadě administrační rozhraní.

##### 3.1.2 Práva uživatelů

Uživatelé se dají rozdělit do čtyř skupin a to: administrátor, registrovaný uživatel s právy, registrovaný uživatel bez práv a v neposlední řadě neregistrovaný návštěvník webu.

- Administrátor: je hlavní uživatel, který se stará o správu celého systému. Takový uživatel může být pouze jeden a má přístup do administračního rozhraní celého systému. Uživatel není omezen pouze na svůj účet, ale může spravovat i účty jiných registrovaných uživatelů a obsah na celém redakčním systému.
- Registrovaný uživatel s právy (písničkář): má přístup k uživatelskému menu, kde je mu umožněno například psát novinky, články, texty svých písní.
- Registrovaný uživatel bez práv: má přístup k omezenějšímu uživatelskému menu, než je tomu u registrovaného uživatele s právy.
- Neregistrovaný návštěvník webu: má jedinou možnost pozměnit web a to svým příspěvkem do knihy návštěv.

##### 3.1.3 Registrace uživatelů

Každý uživatel vystupuje pod přezdívkou, která je jedinečná a nesmí nastat případ, kdy budou existovat dva uživatelé se stejnou přezdívkou. Dále si každý z uživatelů zvolí heslo, které mu bude sloužit ke vstupu do uživatelského menu. Pak následují kontaktní údaje jako je jméno, příjmení, emailová adresa (také je jedinečná), internetová adresa, telefonní číslo,



adresa a motto uživatele. Tak vzniká uživatel, který dosud nemá práva a má tak přístup jen k omezenému uživatelskému menu. Práva mu můžou být přidělena administrátorem v administračním rozhraní. Pak se z něj stává registrovaný písničkář a má tak přístup k uživatelskému menu bez omezení.

#### **3.1.4 Profily uživatelů**

Profily uživatelů budou sloužit k tomu, aby si návštěvník webu mohl udělat představu kdo vlastně uživatel je, jaká je jeho tvorba, popřípadě jej zkontaktovat či ohodnotit.

#### **3.1.5 Novinky**

Novinky jsou jednou z prvních věcí které, návštěvník webu uvidí. Možnost přidávání novinek bude mít pouze administrátor a registrovaný uživatel s právy. Jedná se o krátký výstižný text, který ohlašuje aktuální informace.

#### **3.1.6 Články**

Každý registrovaný uživatel bude mít možnost psát, editovat a mazat své články. Bude mít na výběr z několika kategorií, které vytváří administrátor. Články pak budou zobrazovány uživateli hromadně, nebo podle kategorií, to bude záležet na volbě návštěvníka webu.

#### **3.1.7 Koncerty**

Koncerty bude moci vkládat pouze registrovaný uživatel s právy. Informuje tak návštěvníky webu o tom, kdy se koná koncert nějakého písničkáře, kde se koná, kde se dozvíme více informací o koncertu atd.

#### **3.1.8 Kniha návštěv**

Bude se skládat z jednoduchého formuláře pro přidávání příspěvků a pod ním se budou zobrazovat poslední přidané příspěvky.

#### **3.1.9 Hudba**

Zde budou mp3, které budou postupně vkládány uživateli. Každý návštěvník si tak bude moci poslechnout tvorbu písničkářů.

### **3.1.10 Texty**

Jedná se o texty písní, které si může vložit každý registrovaný uživatel, jenž má přiděleny práva. Texty si pak budou moci návštěvníci webu prohlédnout.

### **3.1.11 Fotky**

Registrovaní uživatelé s právy si budou moci vkládat do databáze fotografie a ty pak budou zobrazovány návštěvníkům webu.

### **3.1.12 Administrační rozhraní**

K tomuto rozhraní bude mít přístup jen jediný člověk a to administrátor. Ten se bude muset prokázat administrátorským jménem a heslem. Administrační rozhraní bude obsahovat správu novinek, článků, koncertů, příspěvků v knize návštěv, mp3 uložených na serveru, textů písní, fotogalerií a fotografií, mazání uživatelských účtů, přidělování práv uživatelům a vytváření kategorií pro články.

## 4 NÁVRH DATABÁZOVÉ STRUKTURY REDAKČNÍHO SYSTÉMU

Databáze nese název „pisnickari“ a obsahuje dvanáct tabulek. Do těchto tabulek se ukládají data, která jsou pak zobrazována návštěvníkům stránek. Pro ukázkou si popíšeme jen několik tabulek. Vysvětlíme si v jakém formátu se data do tabulek vkládají a k čemu slouží.

### 4.1 Tabulky v databázi

#### 4.1.1 Tabulka uživatelé

Tato tabulka slouží pro ukládání informací o uživateli, který se zaregistruje. Každý z těchto uživatelů vystupuje pod číslem id, které je pro každého jedinečné, přezdívkou, která je taktéž jedinečná a emailem, který také nemůže být stejný pro více uživatelů. Dále je po uživateli při registraci požadováno heslo, které se do databáze ukládá jako hash vytvořený ze zadaného hesla pomocí PHP funkce md5(). Při přihlašování se pak na heslo, které zadá uživatel použije funkce md5() a teprve výsledek pak porovnává s heslem uloženým v databázi. Fotku si do profilu může přidat uživatel až po registraci, při editaci svého účtu. Název fotografie se do databáze ukládá jako spojení náhodného čísla v rozmezí 125 až 999 a aktuálního času, na které je použita funkce md5(), aby nedocházelo ke stejným názvům souborů. Dále se ukládá do databáze jméno a příjmení uživatele, jeho motto, pokud nějaké má, hodnocení, což je počet bodů, kterých uživatel dosáhl. Počet hlasů pak určuje kolik lidí pro něj hlasovalo a když pak vydělíme hodnocení počtem hlasů, dostaneme jako výsledek uživatelovo aktuální skóre. Při registraci se také uloží do databáze datum registrace a při každém přihlášení se přepisuje datum a čas posledního přihlášení, poslední IP adresa ze které se přihlásil a počet přihlášení od registrace. Tato data můžeme použít k jednoduchým statistikám. Do databáze se ještě ukládají webové stránky uživatele, pokud nějaké má, jeho adresa, pokud si přeje, jeho telefon a nakonec práva, které po samotné registraci nabývají nulových hodnot. Práva může změnit až administrátor v administračním rozhraní.

Tabulka 1- tabulka pisnickari\_uzivatele

Sloupec	Typ
id	INTEGER NOT NULL AUTO_INCREMENT, PRIMARY KEY (id)

lang	VARCHAR (5)
prezdivka	VARCHAR (100)
heslo	VARCHAR (32)
fotka	VARCHAR (40)
jmeno	VARCHAR (30)
prijmeni	VARCHAR (30)
email	VARCHAR (50)
moto	MEDIUMTEXT
hodnoceni	INT (11)
pocet_hlasu	INT (4)
datum_registrace	DATE
posledni_prihlaseni	DATETIME
posledni_IP	VARCHAR (20)
pocet_prihlaseni	VARCHAR (5)
www	VARCHAR (60)
adresa	VARCHAR (255)
telefon	VARCHAR (13)
prava	TINYINT (1)

#### 4.1.2 Tabulka články

Tato tabulka slouží k ukládání informací o článcích, které jsou psány uživateli tohoto redakčního systému. Každý článek obsahuje svoje jedinečné číslo id. Tabulka je provázána s tabulkou pisnickari\_menu pomocí sloupce id\_kategorie a s tabulkou pisnickari\_uzivatele pomocí sloupce id\_uzivatele. Při psaní článku si uživatel nejdříve zvolí, do které kategorie článek patří, k tomu právě slouží ono provázání s tabulkou pisnickari\_menu, pak vyplní titulek článku, který může nabývat maximální hodnoty 150 znaků. Dále vloží stručný popis článku, ten se ukládá do sloupce text\_telo a může začít psát samotný článek, který bude

uložen ve sloupci text. Navíc má uživatel možnost rozhodnout se, jestli si přeje, aby byly k jeho článku psány komentáře, jeho volba pak bude zaznamenána ve sloupci komentare. Po okomentování článku se zvýší hodnota sloupce pocet\_komentaru vždy o jedničku. Ještě před vložením článku do databáze se z titulku vytvoří po nahrazení nevhodných znaků pomlčkami URL adresa, která bude zobrazena v internetovém prohlížeči při čtení článku. Další operace prováděná před uložením článku do databáze je výběr klíčových slov, jedná se o dvacet nejčastěji použitých slov, které mají minimální počet znaků pět.

Tabulka 2 - tabulka pisnickari\_clanky

Název sloupce	Typ
id	INTEGER NOT NULL AUTO_INCREMENT, PRIMARY KEY (id)
id_kategorie	INTEGER
id_uzivatele	INTEGER
uzivatel	VARCHAR (100)
titulek	VARCHAR (150)
url	VARCHAR (100)
text_telo	TINYTEXT
text	TEXT
klicova_slova	TINYTEXT
cas_vlozeni	DATETIME
Komentare	TINYINT (1)
pocet_komentaru	INT (3)
delka_clanku	INT (5)

#### 4.1.3 Tabulka komentáře

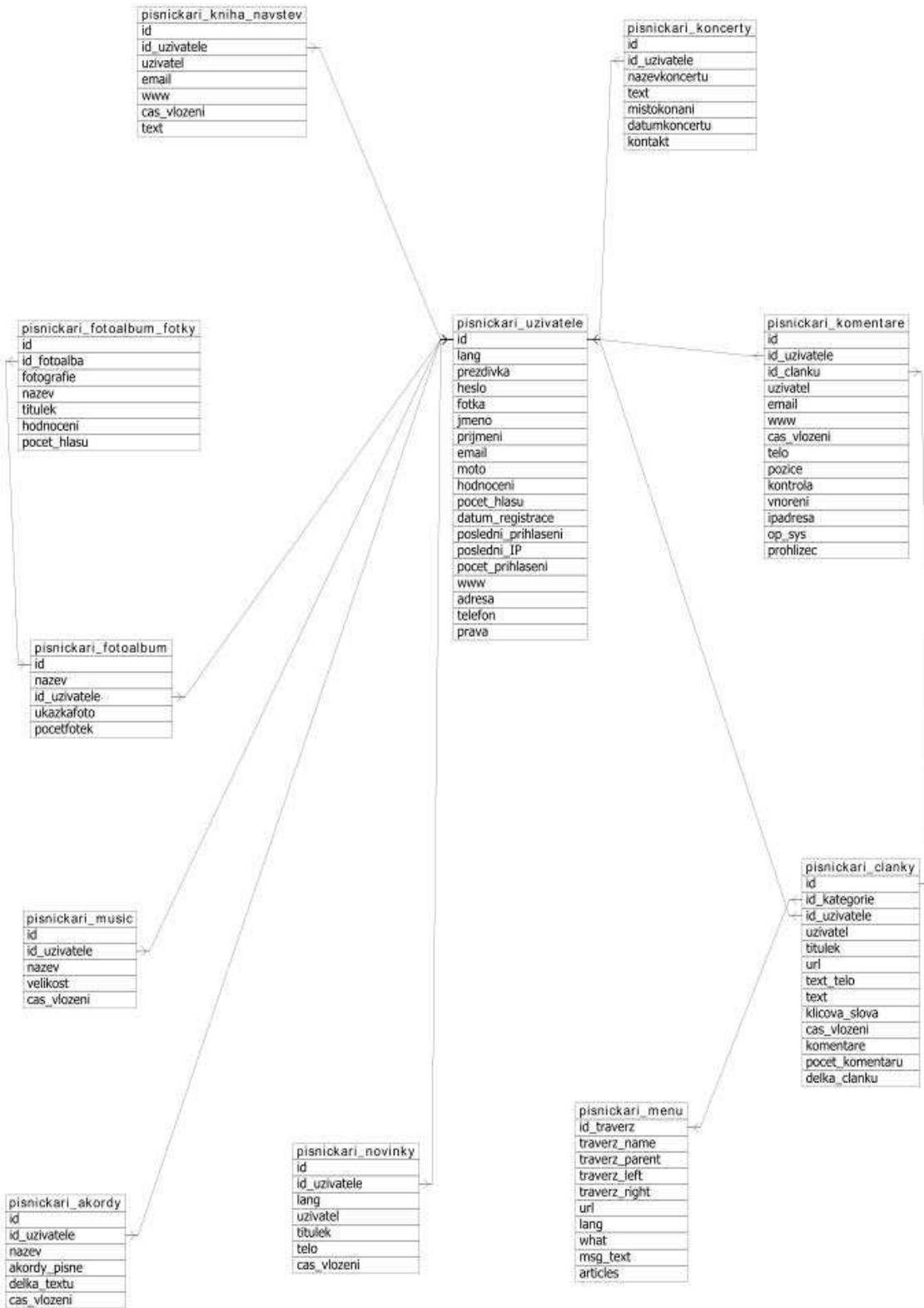
Tato tabulka slouží k ukládání komentářů, které jsou napsané ke článkům. Tabulka je provázána s tabulkou pisnickari\_uzivatele pomocí sloupce id\_uzivatele a s tabulkou pisnickari\_clanky pomocí sloupce id\_clanku. Každý komentář má svoje jedinečné číslo id, součástí každého komentáře je autor komentáře (sloupec uzivatel), email autora komentáře

(sloupec email může být prázdný), www stránky autora komentáře (sloupec www může být prázdný), čas vložení komentáře (sloupec cas\_vlozeni), text komentáře (sloupec telo). Sloupec pozice a kontrola slouží ke zjištění umístění komentáře na stránce. Sloupec pozice určuje o kolikátý komentář se jedná, sloupec vnoreni se používá u reakcí na komentář. Hodnota tohoto sloupce určuje jeho odsazení od ostatních komentářů. Po reakci na komentář se musí přečíslovat všechny pozice komentářů u daného článku, protože pokud bychom reagovali na komentář, za kterým by se nacházely další komentáře, už by nám nevycházelo jejich číslování pozic.

Tabulka 3- tabulka pisnickari\_komentare

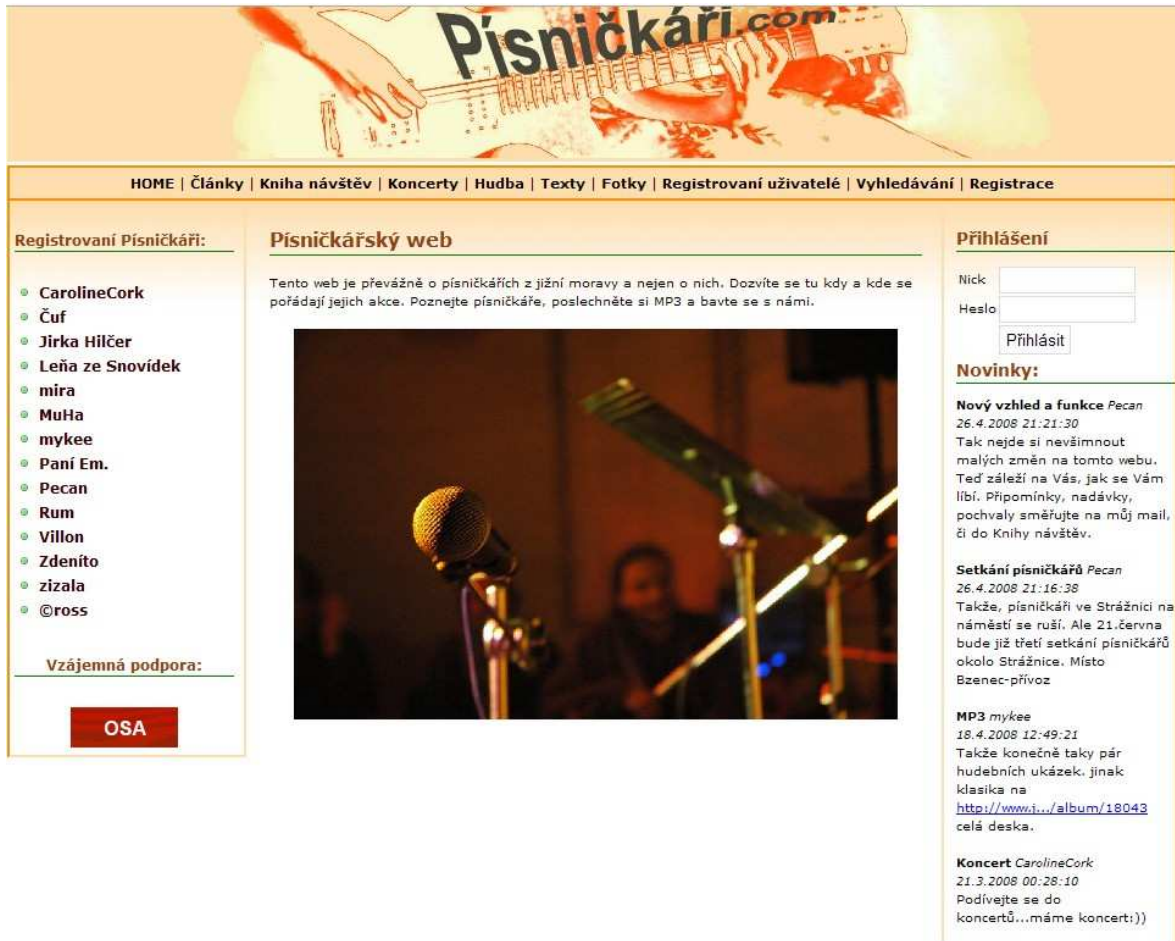
<b>Sloupec</b>	<b>Typ</b>
id	INTEGER NOT NULL AUTO_INCREMENT, PRIMARY KEY (id)
id_uzivatele	INTEGER
id_clanku	INTEGER
uzivatel	VARCHAR (100)
email	VARCHAR (50)
www	VARCHAR (50)
cas_vlozeni	DATETIME
telo	TEXT
pozice	INT (3)
kontrola	TINYINT
vnoreni	INT (3)

4.1.4 Relace v databázi



Obrázek 4 Relace v databázi

## 5 POPIS VYTVOŘENÉHO REDAKČNÍHO SYSTÉMU



Obrázek 5 - Vzhled webu

### 5.1 Rozložení webové aplikace

#### 5.1.1 Levá navigace

Celá levá navigace je věnovaná výpisu přezdívek registrovaných uživatelů, kteří mají přidělení práva. Po kliknutí na jejich přezdívku se zobrazí uživatelův profil, který sdělí všechny potřebné informace o uživateli. Je zde možnost prohlédnout si fotografii, pokud byla do profilu vložena, články které uživatel napsal, texty písní které vložil, mp3 které vložil, kontaktní údaje, motto atd. Navíc je možné pro uživatele na stránce jeho profilu hlasovat.



### 5.1.2 Pravá navigace

Pravá navigace pak slouží k zobrazení několika posledních novinek. Je tu však i odkaz, který slouží k vypsání všech novinek, jenž byli postupně přidávány. Jednou z dalších věcí, která se v levé navigaci vyskytuje je přihlašovací formulář. Tento se po úspěšném přihlášení nahradí uživatelským menu a odkazem pro odhlášení a dalším odkazem pro editaci profilu, kde si uživatel může měnit své osobní údaje a fotografii, která se zobrazuje v profilu.

### 5.1.3 Horní menu

Horní menu obsahuje deset položek, jako první je položka *HOME*, která uživatele přesměruje na domovskou stránku, další jsou pak *ČLÁNKY*, kde se uživateli zobrazí posledních osm vložených článků a má zde možnost vypsání si všechny články, nebo články rozdělené podle kategorií. Pak je tu *KNIHA NÁVŠTĚV*, která začala během testování fungovat spíše jako chat. Zde může zanechat vzkaz, nebo položit otázku, či téma k diskusi kdokoliv, kdo navštíví tento web. Následují *KONCERTY*, kde jsou vypsány koncerty uživatelů a informace o nich. Dále je tu *HUDBA*, tady jsou mp3, které si návštěvník webu může zobrazit seřazené podle přezdívky autora, nebo hromadně všechny mp3 uložené na serveru. Pak následují *TEXTY*, tady naleznete texty písní, které uživatelé vkládali buď s akordy nebo bez, podle jejich uvážení. *FOTKY* obsahují fotoalba, ve kterých jsou uloženy fotky a návštěvník webu si je může prohlédnout. Jako další položka jsou *REGISTROVANÍ UŽIVATELÉ*, kde jsou vypsáni všichni registrovaní uživatelé ať už s právy nebo bez nich. Můžete si zobrazit jejich profil nebo přímo přejít na jejich webové stránky, pokud je vyplnili. *VYHLEDÁVÁNÍ* je předposlední položkou horního menu a poskytuje možnost hledat *fulltextově* v článcích a pomocí operátoru *like* v mp3. Poslední položka je *REGISTRACE*, pod ní se skrývá registrační formulář, který je nutno vyplnit, abychom se stali registrovanými uživateli tohoto webu.

### 5.1.4 Hlavní panel

Hlavní panel svůj obsah mění podle toho, v jaké sekci se právě nacházíme a zobrazují se zde data, které si uživatel vybral.

## 5.2 Uživatelské menu

Po přihlášení registrovaného uživatele se zobrazí jeho uživatelské menu, které před přihlášením není dostupné. Pokud přihlášený uživatel nemá přidělená práva, pak jeho uživatelské menu tvoří pouze tvorba článků, jejich editace a mazání. O přidělení práv se stará administrátor. Pokud přihlášený uživatel tyto práva má, pak jeho nabídku tvoří:

- Editace účtu
- Přidávání a mazání novinek vložených daným uživatelem
- Editace a mazání článků vložených daným uživatelem
- Vkládání nových článků
- Editace a mazání koncertů vložených daným uživatelem
- Vkládání koncertů
- Vkládání a mazání vlastních mp3 souborů
- Vkládání svých textů písní s možností přidání akordů a jejich mazání
- Tvorba fotoalb
- Přidávání fotografií do svých fotoalb

## 5.3 Administrační rozhraní

Slouží k administraci celého redakčního systému, přístup k tomuto rozhraní má jen administrátor, jehož totožnost je ověřována přihlašováním. Po úspěšném přihlášení má možnost měnit obsah celého webu a měnit či odstraňovat data, které uživatelé vkládají.

### 5.3.1 Administrační menu

- Domů – tato položka slouží k editaci vzhledu hlavní stránky, po vyplnění formuláře se data uloží do databáze a při zobrazení hlavní stránky webu se tato data vypíší. Vkládá se nadpis, text a obrázek.
- Menu – zde administrátor vytváří kategorie pro články vkládané registrovanými uživateli s právy.

- Tabulky – právě zde administrátor vytváří tabulky, do kterých se ukládají všechny data vložené uživateli a to ať už články, novinky, mp3 či koncerty atd.
- Novinky – tady administrátor přidává novinky, které budou zobrazeny v pravé navigaci na webu, nebo již přidané novinky edituje a to jak vložené registrovanými uživateli s právy, tak i jeho vlastní. Předchází tak zobrazování nevhodného obsahu na webu. Případně vloženou novinku úplně odstraní.
- Články – po zvolení kategorie si administrátor vybere článek, který je nějakým způsobem nevhodný a ten upraví, nebo odstraní.
- Kniha – pod tímto odkazem se vypisují chronologicky seřazené příspěvky z knihy návštěv a administrátor zde snadno odstraní nevhodné příspěvky. Příspěvky se zde vypisují po dvaceti na jednu stránku.
- Koncerty – tady se opět administrátorovi nabízí možnost editace či úplného odstranění koncertu, vloženého registrovaným uživatelem s právy.
- Hudba – zde jsou vypsány všechny mp3 uložené registrovanými uživateli s právy a administrátor tyto mp3 může kdykoliv po svém uvážení odstranit. Kvůli přehlednosti se mu vypisují po dvaceti na stránku.
- Texty – texty může administrátor pouze mazat, editace mu zde nabídnuta není. Před smazáním si může text písňě prohlédnout, kvůli přehlednosti jsou texty vypisovány opět po dvaceti na stránku.
- Uživatelé – tato položka slouží ke správě uživatelských účtů, administrátorovi se zde vypíše tabulka s informacemi o uživateli, kteří jsou na webu zaregistrováni, může si je seřadit podle data registrace, posledního přihlášení nebo jejich přezdívky, pod kterou vystupují. Po kliknutí na přezdívku vyskočí vyskakovací okno s údaji, které uživatel při registraci vyplnil. Uživatele, který mu přijde jako nevhodný může odstranit. Pokud si o uživateli myslí, že by měl patřit mezi písničkáře (registrované uživatele s právy), tak mu přidělí práva.
- Fotky – zde jsou administrátorovi zobrazena fotoalba, s názvy jejich autorů. Administrátor si může tato alba prohlédnout a nevhodné fotografie odstranit nebo smazat fotoalbum celé včetně fotografií, které obsahovaly.

- Editovat – pod touto položkou se skrývá formulář pro editaci administrátorského účtu.
- Odhlásit – jak již z názvu vyplývá, tento odkaz slouží pro odhlášení administrátora z administračního rozhraní.



The screenshot displays the administrative interface with the following elements:

- Header:** "Administrační rozhraní" in red text.
- Navigation:** A menu bar with links: Domu | Menu | Tabulky | Novinky | Clanky | Kniha | Koncerty | Hudba | Texty | Uzivatele | Fotky | Jste přihlasen jako: admin Odhlasit | editovat |
- Section:** "Hlavni stranka" (Main page).
- Form:** A form with three main sections:
  - Nadpis:** A text input field.
  - Text:** A large text area for content.
  - Obrázek:** A section containing a file input field, a "Procházet..." button, and an "Uložit" button.
- Footer:** "Martin Suranek www.pjanickan.com"

Obrázek 6 - Administrační rozhraní

## 6 IMPLEMENTACE A OTESTOVÁNÍ VYTVOŘENÉ APLIKACE

### 6.1 Pořízení domény a webového prostoru

K tomu, aby mohl být redakční systém na webu plně funkční, se musela pořídit doména a webový prostor. Nabídka je široká, ale nakonec byl zvolen webhosting od provozovatele station.cz a to pro jeho výhodnou cenu za solidní služby. Tarif, který byl vybrán obsahuje 1500 MB webového prostoru, neomezené množství subdomén, neomezené množství FTP účtů a spoustu dalších výhodných služeb. Server, který nám byl přidělen obsahuje databázi MySQL 4.1 a podporu PHP5. Tentýž poskytovatel nabízí registraci domén, po několika pokusech byla zvolena vyhovující doména, která naštěstí nebyla obsazena a to „pisnickari.com“. Po zaplacení faktury trvalo ještě několik dnů, než byla doména a prostor aktivovaný, pak už bylo možné začít testovat redakční systém přímo na internetu.

### 6.2 Instalace Redakčního systému

Instalace se skládá z několika částí, které si zde popíšeme.

- Vytvoříme na serveru databázi.
- V souboru data.php, který je umístěn ./www/administrace/data.php nastavíme informace o připojení k databázi. To znamená: jméno hostitelského počítače, jméno uživatele, heslo uživatele a název databáze.
- V souborech settings.php které jsou umístěny ./www/administrace/settings.php a ./www/settings.php nastavíme prefix pro název tabulek.
- Soubory z adresáře ./www/ zkopírujeme do hlavního adresáře s webem.
- Pokusíme se přistoupit do administračního rozhraní tím, že do webového prohlížeče napíšeme jméno hostitelského počítače/administrace. Pokud jsme zde poprvé, budeme vyzváni k instalaci administrátorského účtu. Po kliknutí na odkaz jsme přesměrováni na adresu, kde se nám zobrazí formulář, do kterého vyplníme údaje o administrátorovi a to jméno a heslo. Pokud vše proběhne v pořádku, obdržíme o tom pomocí výpisu informace a budeme již schopni se přihlásit do administrátorského rozhraní.

- Po přihlášení se automaticky smažou soubory, které byly potřebné k vytvoření účtu administrátora. Administrátor si však bude schopen změnit své uživatelské jméno a heslo kdykoliv po přihlášení.
- Jako první krok po přihlášení bychom měli kliknout na odkaz v horním menu s názvem tabulky, vypíše se nám seznam tabulek, které je potřeba vytvořit a pod nimi je seznam tabulek, které nějakým způsobem vytvářejí konflikt. Nabízejí se nám dvě možnosti: buď smažeme konfliktní tabulky, což bude mít za následek smazání pouze prázdných tabulek a tudíž o data nepřijdeme, nebo zvolíme druhou možnost což je vytvoření tabulek. Zvolíme tedy druhou možnost, protože pokud jsme v administraci poprvé, určitě jsme ještě tabulky nevytvářeli. A tím je instalace hotová.

### 6.3 Testování vytvořené aplikace

Redakční systém je vystaven na internetu od 17.11.2007, kde se postupně odlaďovaly chyby a přetvářel se k obrazu zadavatele a přání ostatních písničkářů, kteří byli plní nových a nových nápadů jak spoustu věcí vylepšit a usnadnit jim tak používání onoho redakčního systému.

#### 6.3.1 Diakritika

Hned mezi první problémy po vystavení systému na internetu patřil problém s diakritikou. Uživatelé si začali všimnout, že většina výpisů neobsahuje diakritické znaménka, což jim připadalo nepěkné. Toto upozornění bylo akceptováno a na potřebných místech byla diakritika doplněna.

#### 6.3.2 Upload uživatelských fotografií

Další problém se vyskytl při nahrávání fotek uživatelů, když se pokoušel PHP skript uložit fotografii uploadovanou uživatelem do adresáře, který byl vytvářen taky PHP skriptem. Nastával zde problém s právy adresářů, které se nedařilo pomocí PHP funkce `chmod()` změnit. Problém nebyl řešen dříve, protože redakční systém byl vytvořen na softwarovém HTTP severu, který běžel pod operačním systémem Windows. Zde jsou práva adresářů řešena jinak než na webovém serveru na internetu, který běží pod operačním systémem Linux. Problém byl nakonec vyřešen tím, že přímo na FTP serveru byl ručně vytvořený

adresář a tomu pomocí FTP klienta (v mém případě Total Commander) byla adresáři nastavena práva. Tento problém se pak časem opakoval při vytváření uživatelských fotoalb. Řešením bylo vytváření adresářů přes PHP skript pomocí příkazů `ftp_login()`, `ftp_mkdir()`, `ftp_site` a `ftp_close()`, což je mnohem estetičtější řešení než to původní.

### 6.3.3 Vyhledávání

Další problém se vyskytl při tvorbě vyhledávání. Vyhledávání je rozděleno na dvě možnosti, uživatel si může zvolit, že chce hledat v člancích a nebo v mp3. Hledání v člancích je řešeno fulltextově, to znamená že u tabulky, ve které chceme fulltextově vyhledávat musíme nejdříve u sloupců, ve kterých se bude hledat přiřadit fulltextové indexy. Pak výběrovým dotazem obohaceným o příkazy *AGAINST* a *MATCH* porovnáme hledaný řetězec s daty v tabulce. Porovnávání ale probíhá v kódování, které je nastavené v databázi zvlášť pro porovnávání. V našem případě šlo o kódování *cp1250czech\_cs* a to mělo za následek, že porovnávání řetězců bralo v potaz i to, zda se jedná o malé či velké písmeno, což bylo pro vyhledávání nevhodné. Proto bylo nutné upravit kódování, které se používá pro porovnávání na *cp1250\_general\_ci*. Stejný problém nastal u vyhledávání v mp3 jež bylo řešeno operátorem *LIKE*.

### 6.3.4 Novinky

Výpis několika posledních novinek je zobrazován při každém načtení stránky, proto byl použit nápad vytvořit nějaké jednoduché *cachování*, aby nemusel při každém zobrazení stránky probíhat dotaz na databázi, když stačí aby se obsah měnil jednou za určitou dobu, kterou si uživatel nastaví. Princip je jednoduchý - po odečtení hodnoty PHP funkce *time()* (což je zhruba 1209507566 sekund) od hodnoty PHP funkce *fileatime()*, která nám vrátí čas v sekundách od poslední změny souboru (v tomto případě jde o soubor *news.dat*) dostaneme číslo, které pokud je menší než nastavená hodnota, tak *naincludujeme* soubor *news.dat* do levého menu, kde se vypíše. V opačném případě se smaže soubor *news.dat*, vybere se z databáze posledních šest novinek které se budou vypisovat v levém menu. Zformátované novinky pro výpis se pak uloží do *bufferu*, jeho obsah se vypíše a pak jej uložíme do souboru *news.dat*

### 6.3.5 Spam v knize návštěv

Po určité době začalo docházet v knize návštěv k problému se *spamem*, byly vkládány nežádoucí příspěvky od *spamových robotů*. Problém byl řešen pomocí *JavaScriptu* a vložení formuláře s kontrolní otázkou. Na stránce je přidán element *SPAN*, který obsahuje onen formulář s kontrolní otázkou a *inputem*, do kterého se doplňuje odpověď. Pokud je odpověď špatná, bereme příspěvek jako neplatný a vložený od *spamového robota*, a proto ho do databáze nevkládáme a nebude tak vypsán. Naopak jestli je otázka zodpověděna správně, příspěvek je vložen do databáze a poté vypsán. K odpovědi na kontrolní otázku je ale vyzván pouze uživatel, který má v prohlížeči zakázaný *JavaScript*, pokud je v prohlížeči *JavaScript* povolen, pak se formulář umístěný ve *SPANU* nezobrazí. Ve formuláři pro vložení příspěvku doplníme hodnotu atributu *onsubmit* pomocí *JavaScriptu*, který odpověď na otázku doplní a uživatele tak neobtěžuje doplňováním odpovědi.



## ZÁVĚR

V této bakalářské práci jsem se zabíral problematikou redakčních systémů. V teoretické části je uveden obšírný rozbor několika redakčních systémů, které jsou volně dostupné na internetu a co k jejich funkčnosti potřebujeme. Praktická část je věnována tvorbě samotného redakčního systému a problémy, na které jsem při tom narazil. Tento redakční systém je již od 17.11.2007 testován na internetu. Základem tohoto redakčního systému byla možnost prezentace samotných písničkářů, kteří se tak mohli podělit o svou tvorbu s okolním světem a dostat se tak do podvědomí všech lidí, kteří dosud o písničkářích nevěděli. Jde to pomocí jejich profilů, které si uživatelé internetu mohou zobrazit. Návštěvník webu má možnost písničkáře ohodnotit a tak ho motivovat k jeho další tvorbě. Další způsob, jak se něco dozvědět o registrovaných písničkářích, je přečíst si jejich články, které samotní písničkáři píšou. Navíc se návštěvníci tohoto webu dozví o koncertech, které se pořádají. Mají možnost poslechnout si tvorbu těch, kteří koncert pořádají a posoudit tak, jestli je právě ten koncert pro ně to pravé. Uživatelé tohoto redakčního systému měli a mají stále nové návrhy pro zlepšení tohoto systému a to je pro mě hnací motor k tomu, abych na tomto systému neustále pracoval a zlepšoval ho tak, aby pro ně byl co možná nejpřehlednější a nejsnazší pro používání. Webové stránky naleznete na adrese <http://www.pisnickari.com>.

## ZÁVĚR V ANGLIČTINĚ

I was engaged in problems of the content management system in this bachelor work. In the theoretical part there is an analysis of some content management systems which are free available on internet and what we need to their function. The practical part is dedicated to making the content management system and some problems which I met. This content management system is tested on internet since 17<sup>th</sup> November 2007. The basis of this content management system is the possibility of songsters' presentation. They can share their production with surrounding world and get into all people's subconscious. It's all about their profiles which all internet users can visit. Every web user can rate the songster and motivate him to his next production this way. Another way of getting know of the registered songsters is to read their articles. Web user can read about organized concerts, they can listen to songsters' songs and decide if to go or not go to their concert. Users of this content management system had and still have new proposals for improving of this system. This is a good motivation for me to go on working on this system and to improve it to be the most well-arranged and easy to take in using. You can find the websites on <http://www.pisnickari.com>.

## SEZNAM POUŽITÉ LITERATURY

- [1] PROKOPOVÁ, Z.: Databázové systémy MySQL+PHP. FAI UTB Zlín, s. 126, 2006, Vysokoškolská skripta. ISBN 80-7318-486-9.
- [2] JESUS, Castagnetto, et al. Programujeme profesionálně : PHP. Robert FE Shaw. 2004. autoriz. vyd. Praha : Computer Press, 2004. 656 s., CD-ROM. ISBN 80-7226-310-2.
- [3] JavaScript a Ajax : Moderní programování webových aplikací. 1. vyd. Praha : Computer Press, a.s., 2007. 360 s. ISBN 978-80-251-1824-5.
- [4] LUBOSLAV, Lacko. SQL: hotová řešení. Praha : Computer press, 2003. 296 s. ISBN 80-7226-975-5.

**SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK**

CMS	Content Management Systém
HTML	HyperText Markup Language
CSS	Cascading Style Sheets
PHP	Hypertext Preprocesor
HTTP	Hypertext Transfer Protokol
SQL	Structured query language
GPL	General Public Licence
RSS	Rich Site Summary, Realy Simple Syndication
GNU/GPL	GNU's Not Unix/General Public Licence
Mp3	Motion Picture Layer 3
URL	Uniform Resource Locator

**SEZNAM OBRÁZKŮ**

Obrázek 1-PSPad .....	20
Obrázek 2 - NuspherePhpED .....	21
Obrázek 3 - Photofiltre .....	22
Obrázek 4 Relace v databázi .....	31
Obrázek 5 - Vzhled webu .....	32
Obrázek 6 - Administrační rozhraní .....	36

**SEZNAM TABULEK**

Tabulka 1- tabulka pisnickari_uzivatele .....	27
Tabulka 2 - tabulka pisnickari_clanky .....	29
Tabulka 3- tabulka pisnickari_komentare.....	30

## SEZNAM PŘÍLOH

P I: Použité třídy

## PŘÍLOHA P I: POUŽITÉ TŘÍDY

Název třídy	Definice	Počet volání třídy
addclanky	1	1
admin	1	1
check	2	18
clanky	2	2
class_hledani	1	1
class_hudba	1	1
class_texty	1	1
class_texty_pisni	1	1
class_texty_vypis	1	1
class_users	1	1
classmp3	1	1
connect	1	43
create_tables	1	1
edit	1	1
fotky	1	1
fotoadd	1	1
fotoalbum	1	1
globalclass	1	25
home	2	3
kniha	1	1
kniha_navstev	1	1
komentare	1	1



koncerty	2	2
koncerty_vypis	1	1
koncertyedit	1	1
login	2	3
mp3_hornimenu	1	1
news	2	2
novinky	1	1
novinkyreguser	1	1
pisnickari	1	1
posledniclanky	1	1
profil	1	2
register	1	1
texty	1	1
traverz	1	1
userdetail	1	1
uzivatele	1	1
vypis_clanky	1	1

## **Třída addclanky**

Tato třída slouží k přidávání článků do databáze, obsahuje sedm následujících metod:

- kontrolaform() - tato metoda ověřuje formulář, který uživatel vyplňuje. Volá se před provedením vkladacího dotazu na databázi. Tato metoda vrací jedničku pokud je vše v pořádku, pokud ne vrátí chybové hlášení a nulu.
- pocetslov() - tato metoda nám vrátí počet slov, které jsou ve vstupním parametru. Vstupním parametrem je proměnná pocetslov.

- `urladdress()` - vstupním parametrem této metody je proměnná `adress`. Tato metoda nám ze vstupní proměnné nahradí nevhodné znaky pomlčkami. A vrátí nám takto upravený řetězec.
- `klicoveslova()` - tato metoda rozdělí text článku na slova, které si uloží do pole. Toto pole pak projde a ke slovu, na které již během procházení narazil připočte vždy jedničku. Pak toto pole seřadí a prvních dvacet nejvyskytovanějších slov vrátí jako řetězec oddělený čárkami.
- `savetodb()` - v této metodě se vytvoří vkládací dotaz, který se provede jen v případě že metoda `kontrolaform()` vrátí jedničku. V tomto dotazu jsou volány metody `klicoveslova()`, `urladerss()`, `pocetslov()` pro vyplnění sloupců v databázi, které jsou jim určené. Pokud se vkládací dotaz přesto neprovede, vypíše se o tom chybové hlášení, naopak když se provede správně, vypíše se uživateli informace, že byl článek úspěšně vložen a vyprázdníme `super-globální` pole `POST`.
- `vyberkategorie()` - pomocí výběrového dotazu na databázi zjistíme, jaké máme pro články vytvořené kategorie, pokud nejsou žádné vypíše se o tom chybové hlášení a metoda vrátí nulu. Pokud databáze obsahuje kategorie, vytvoří se formulářový výběr `<select>`, který naplníme dostupnými kategoriemi.
- `formulář()` - v této metodě se vypisuje formulář pro vkládání článku, který je umístěn v tabulce. První položkou tohoto formuláře je výběr kategorie, který získáme zavoláním metody `vyberkategorie()`, pak následuje položka pro název článku, dále pro stručný popis článku a pro samotný text článku. Navíc je ve formuláři obsazeno zaškrtačací políčko pro případ, že chceme nebo nechceme, aby se mohly ke článku psát komentáře. A nakonec odesílací tlačítko s názvem `přidat článek`. K většině položek je přidán obrázek s otazníkem, který po přejetí myší zobrazí uživateli nápovědu.

## **Třída `admin`**

Tato třída je určena pro změnu přihlašovacích údajů administrátora a obsahuje tři následující metody:

- `savetodb()` - tato metoda slouží k ukládání nových údajů o administrátorovi, obsahuje obnovovací dotaz na databázi. Při úspěšné změně údajů se vypíše hlášení,

při neúspěchu se zobrazí chybové hlášení a vrací se nula. Metoda se volá jen pokud metoda kontrolaform() - vrací jedničku.

- kontrolaform() - slouží pro kontrolu formuláře. Při úspěchu vrátí jedničku, při neúspěchu vrátí chybové hlášení a nulu. Metoda se volá, jen pokud je velikost super-globálního pole POST větší než 0.
- vypiszdb() - tato metoda vypíše v tabulce formulář pro editaci administrátorského účtu, přičemž do položky jméno vloží stávající jméno administrátora, protože není nutné měnit jméno, většinou se jedná o změnu hesla.

## **Třída check**

Tato třída slouží pro kontrolování formulářů. Obsahuje čtyři metody:

- kontrolaretezce() - tato metoda má tři vstupní parametry: vstupní řetězec, minimální požadovanou délku a maximální požadovanou délku. Pokud řetězec těmito podmínkám vyhovuje, vrací metoda jedničku, pokud ne vrátí nulu.
- check\_phone() - tato metoda má dva vstupní parametry: řetězec obsahující číslo, které chceme zkontrolovat a délku telefonního čísla. Požadovaný formát čísla je +xxx xxx xxx xxx. Pokud testovaný řetězec vyhovuje podmínkám, pak vrátí metoda jedničku, pokud ne pak vrátí nulu.
- check\_email() - tato metoda kontroluje, zda byl zadán email ve správném tvaru, pokud podmínce vyhovuje vrátí jedničku, jinak vrací nulu.
- check\_url() - slouží ke zjištění jestli byla URL adresa zadána ve správném tvaru, pokud ano vrátí jedničku, jinak vrátí nulu.

## **Třída clanky**

Tato třída slouží v administraci pro správu článku. Obsahuje pět metod:

- mazani() - obsahuje dotaz na databázi, který odstraní vybraný článek. Pokud se smazání nepodaří, vrátí metoda nulu.

- kontrola() - kontroluje data vkládané při editaci, je zde spuštěna nová instance třídy check a z ní použity vybrané metody. Vrací jedničku pokud proběhne kontrola v pořádku a nulu pokud ne.
- editace() - pokud metoda kontrola() vrátí jedničku, provede obnovovací dotaz na databázi a upraví tak článek.
- vyberzdb() – tato metoda vybírá z databáze všechny kategorie článků a vypíše formulář s rolovacím menu, aby si administrátor mohl vybrat kategorii, ze které chce článek upravovat.
- vypisclanku() – vypíše všechny články ze zvolené kategorie a nabídne administrátorovi možnost jejich editace, či smazání.

### **Třída class\_hledani**

Tato třída slouží k vyhledávání v článcích nebo v mp3. Obsahuje dvě metody:

- formulář() – zde se vypíše formulář pro hledané výrazy. Skládá se ze dvou radio buttonů, které určí kde se bude vyhledávat (označený může být totiž vždy jen jeden) a textového inputu pro vložení hledaného výrazu.
- hledej() – tato metoda má dva parametry: hledané slovo, kde má hledat. Ve článcích probíhá vyhledávání fulltextově a v mp3 pomocí operátoru LIKE.

### **Třída class\_hudba**

Tato třída slouží v administraci pro správu mp3 souborů. Obsahuje dvě metody:

- delmusic() – parametrem této funkce je id mp3 souboru, který má být odstraněn. Nejdříve tento soubor odstraníme dotazem z databáze, pak samotný soubor odstraníme příkazem unlink ze serveru.
- vypis() – slouží k výpisu všech mp3 vložených uživateli, ke kterým jsou doplněny odkazy pro odstranění.

### **Třída class\_texty**

Tato třída slouží registrovaným uživatelům s právy pro práci s texty svých písní. Obsahuje čtyři metody:

- delfrom() – slouží k mazání textů, které si uživatel sám do databáze vložil.

- kontrola() – v této metodě se spouští nová instance třídy check() a využívá se tu několik jejích metod pro ověření formuláře pro vkládání textů. Při úspěchu vrací jedničku a při neúspěchu chybové hlášení a nulu.
- save2db() – tato metoda vkládá texty do databáze a to jen v případě, že metoda kontrola() vrací jedničku. Při neúspěchu vypíše chybové hlášení a vrátí nulu.
- vypis() – tato metoda vypisuje uživateli formulář pro vkládání nových textů písní, včetně tlačítek pro přidání akordů a nápovědy. Pod tímto formulářem vypisuje již vložené texty stejným uživatelem a u nich odkaz pro smazání.

### **Třída class\_texty\_pisni**

Tato třída slouží pro návštěvníky webu pro zobrazení vložených textů písní. Vypisuje buď všechny texty písní, které jsou v databázi vloženy, nebo jen texty písní od vybraného uživatele. Obsahuje dvě metody:

- textyuser() – slouží právě k výpisu textů písní od vybraného uživatele, texty se vypisují do tabulky a uživatel si může zobrazit profil autora nebo samotný text písně.
- vypis() – tato metoda slouží k výpisu všech textů písní od všech uživatelů. Pokud je v databázi více než dvacet textů písní, zobrazí se uživateli stránkování a texty jsou rozděleny po dvaceti na stránku.

### **Třída class\_users**

Tato třída slouží k výpisu všech uživatelů, kteří jsou zaregistrováni na serveru <http://www.pisnickari.com>. Třída obsahuje jen jednu metodu:

- vypisuuzivatelu() – tato metoda zajišťuje výpis všech zaregistrovaných uživatelů. Výpis probíhá do tabulky, kde je obsažena přezdívka uživatele, čas posledního přihlášení, webové stránky uživatele, datum jeho registrace a práva uživatele. Seznam těchto uživatelů je rozdělen po patnácti na stránku.

### **Třída classmp3**

Tato třída slouží registrovaným uživatelům s právy pro vkládání svých mp3 souborů. Obsahuje šest metod:

- vypis() – tato metoda vypisuje formulář pro vkládání mp3 souborů. Na konci této metody je zavolána metoda music(), která bude popsána později.
- delmusic() – tato metoda je určena pro odstraňování mp3 souborů, které si uživatel do databáze vložil.
- music() – tato metoda zajišťuje výpis mp3, které si daný uživatel do databáze nahrál a nabízí mu možnost smazání.
- name\_of\_file() – tato metoda vytváří název souboru, který uživatel na server nahrává, před název mp3 se přidá jméno uživatele, který soubor nahrává. Podporované formáty souborů jsou: mp3, waw, wma. Tato metoda vrací nový název souboru.
- velikost() – tato metoda kontroluje maximální velikost souboru, pokud ji uploadovaný soubor přesahuje, metoda vrátí chybové hlášení a nulu. Pokud se soubor do určených mezí vejde, pak vrátí jedničku.
- check\_music() – na začátku této metody se kontroluje hodnota, kterou vrací metoda velikost(). Pokud vrátí nulu, metoda check\_music() vrací nulu a končí. V opačném případě se zavolá metoda name\_of\_file() a určí tak název souboru. Proveďte se upload souboru na server a vložení do databáze.

## **Třída connect**

Tato třída se stará o připojení k databázi. Je nejčastěji používanou třídou. Obsahuje dvě metody:

- \_\_construct() – zde se do proměnných, potřebných k připojení k databázi přiřadí hodnoty ze souboru data.php a zavolá se metoda connect()
- connect() – zde se provádí připojení k databázi, při neúspěchu se vypíše chybové hlášení.

## **Třída create\_tables**

Tato třída se používá v administračním rozhraní a stará o vytváření a mazání tabulek v databázi.

## **Třída edit**

Tato třída má za úkol editaci uživatelských účtů. Obsahuje sedm metod:

- `updatedb()` – tato metoda obsahuje obnovovací dotaz na databázi a provádí ho pouze v případě, že metoda `checkform()` vrátí jedničku.
- `checkform()` – zde je spuštěna nová instance třídy `check`. Používá se zde metod z této třídy pro kontrolu jednotlivých položek formuláře. Při úspěchu vrátí metoda `checkform()` jedničku, naopak pokud se kontrola nezdaří dostaneme chybové hlášení a metoda vrátí nulu.
- `name_of_file()` – tady probíhá příprava názvu uživatelem uploadované fotografie. Proveďte se kontrola zda jde o formát fotografie, který je podporován (podporované jsou formáty `.jpg`, `.png`, `.gif`). Název fotografie se skládá ze spojení dvou řetězců, první je náhodné číslo v rozmezí 125 až 999 a druhý řetězec je čas vložení. Na výsledný řetězec použijeme php funkci `md5()` a metoda vrátí takto vytvořený hash. Tím získáme jedinečný název pro každou z vložených fotografií.
- `images()` – tato metoda kontroluje velikost fotografie, pokud je větší než požadovaná hodnota, vrací metoda chybové hlášení a nulu. Jestli podmínce vyhovuje, vrátí jedničku.
- `existphoto()` – metoda je použita pro případ, kdy už uživatel fotku v profilu má a chce si ji změnit. Pokud uživatel fotku v databázi má, vrátí metoda její název, pokud ne, vrací metoda nulu.
- `checkphoto()` – na začátku této metody se volá metoda `images()`, pokud tato metoda vrací nulu, vrátí i metoda `checkphoto()` nulu. Pokud tomu tak není, pokračuje se v metodě `checkphoto()`. Do proměnné se uloží název předchozí fotografie a to zavoláním metody `existphoto()`. Pak proběhne upload souboru s tím, že název souboru určíme zavoláním metody `name_of_file()`. Fotografie se obnovovacím dotazem uloží v databázi a předchozí fotografie je smazána pomocí php příkazu `unlink`, jehož parametrem bude výše zmíněná proměnná, do které jsme si ukládali název předchozí fotografie.
- `editace()` – tato metoda vypisuje formuláře pro editaci uživatelského účtu a pro nahrávání fotografie do databáze a na server.

## **Třída fotky**

Tato třída slouží běžným návštěvníkům webu k prohlížení fotoalb a fotografií, které tyto fotoalba obsahují. Třída obsahuje dvě metody:

- vypisfotoalb() – zde jsou vypísána fotoalba, která byla vložena registrovanými uživateli s právy. Jsou vypísána v tabulce, která je ve formátu: autor, název fotoalba, čas vložení fotoalba. Po kliknutí na název fotoalba jsou uživateli zobrazeny fotografie, které fotoalbum obsahuje, po kliknutí na autora je uživatel přesměrován na profil daného uživatele.
- vypisfotek() – zde probíhá výpis náhledů fotografií z vybraného fotoalba. Po kliknutí na náhled je zobrazena celá fotografie.

### **Třída fotoadd**

Tato třída slouží registrovaným uživatelům s právy pro přidávání fotografii do již vytvořených fotogalerií. Obsahuje tři metody:

- formulář() – tato metoda vypíše uživateli formulář s rolovacím menu, kde si vybere fotoalbum do kterého chce fotografii vložit. Formulář dále obsahuje kolonky pro vyplnění informací o fotografii, tlačítko pro přidání fotografie a odesílací tlačítko.
- images() – metoda pro kontrolu nahrávané fotografie. Kontroluje se typ souboru a jeho velikost. Při úspěchu vrací tato metoda jedničku, při neúspěchu vrací nulu.
- checkphoto() – v této metodě se volá metoda images(). Pokud tato metoda vrací nulu, vrátí nulu i metoda checkphoto, v opačném případě se provede upload fotografie a přidání do databáze.

### **Třída fotoalbum**

Tato třída slouží registrovaným uživatelům s právy pro přidávání fotoalb do databáze. Obsahuje pět metod:

- mazanifotek() – metoda se volá po kliknutí na ikonu smazání, která se zobrazuje u každé fotografie. Slouží ke smazání vybrané fotografie z fotoalba. Odstraní se jak z databáze, tak i ze serveru.
- vypisfotek() – tato metoda se volá po kliknutí na již vytvořené fotoalbum. Pokud fotoalbum obsahuje fotografie, pak se zobrazí náhledy fotografií a pod každou z nich je ikona pro smazání.



- `delfromdb()` – tato metoda se volá po kliknutí na ikonu smazání, která je umístěna u názvu fotoalba. Slouží ke smazání fotoalba, pokud fotoalbum obsahuje fotografie budou smazány taky. Uživatel je po kliknutí na ikonu smazání o tomto důsledku informován a může tak svůj záměr zvrátit.
- `savetodb()` – tato metoda se volá po kliknutí na tlačítko, kterým se přidávají nové fotoalba. Slouží k uložení fotoalba do databáze a vytvoření adresáře se stejným názvem na serveru. Pokud vše proběhne v pořádku, vyprázdní se super-globální pole POST.
- `vypis()` – tato metoda se volá vždy při nové instanci třídy fotoalbum. Slouží k výpisu formuláře pro přidávání fotoalb. Pod tímto formulářem vypisuje uživatelem již vytvořené fotoalba.

## **Třída `globalclass`**

Tato třída se stará o výpis novinek v levé navigaci webu. Obsahuje pět metod:

- `_work_body()` – vstupním parametrem této metody je řetězec a volá se před výpisem textu novinky. Slouží k úpravě odkazů, v této metodě se volá metoda `make_clickable()`, výsledek této metody se uloží do proměnné, kterou pak metoda `_work_body()` vrací.
- `make_clickable()` – vstupním parametrem této metody je řetězec, ten se pak pomocí regulárního výrazu upraví a metoda ho vrací. V tomto regulárním výrazu je však ještě zavolaná metoda `_return_link_wrap()`.
- `_return_link_wrap()` – vstupním parametrem této metody je řetězec. Pokud je tento řetězec kratší než zadaná mez, metoda tento řetězec vrátí. Ovšem jestli je tento řetězec delší, vrátí ho metoda tak, že se vypíše prvních dvanáct znaků, k těm se připojí tři tečky a posledních dvanáct znaků řetězce.
- `generatenews()` – tato metoda vypíše posledních šest přidávaných novinek, na jejichž text se zavolá metoda `_work_body()`, která zformátuje odkazy obsažené v tomto textu do požadovaného formátu. Na konec přidá odkaz pro výpis všech novinek. Metoda se volá až v metodě `news()`.
- `news()` – tato metoda provádí jednoduché cachování novinek. Zkontroluje jestli existuje soubor `news.dat`. Zároveň musí být uplynulá doba od jeho poslední změny

menší než zadaná mez. Pokud je tato podmínka splněna, naincluduje soubor news.dat. Jinak se zapne výstupní buffer, zavolá se metoda generatenews(), do proměnné se pak uloží obsah bufferu a buffer se vyprázdní. Pak vypíšeme zmíněnou proměnnou a její obsah uložíme do souboru news.dat.

## **Třída home**

Tato třída slouží k výpisu hlavní stránky. Obsahuje dvě metody:

- homepage() – tato metoda vybere z databáze data o úvodní stránce, pokud takové data neexistují, zavolá se metoda výpis. Jinak se zformátují a vypíší.
- vypis() – tato metoda obsahuje základní data, která se vypisují na hlavní stránku v případě, že v databázi nejsou jiná aktuálnější data.

## **Třída home**

Tato třída se nachází v administraci a slouží ke změně vzhledu hlavní stránky. Obsahuje tři metody:

- kontrolaform() – zde je spuštěna nová instance třídy check a použity některé její metody pro kontrolu formuláře. Při neúspěchu vrací chybová hlášení a nulu. Při úspěchu vrací jedničku.
- uloz() – na začátku této metody se volá metoda kontrolaform(). Pokud vrátí jedničku, uloží se data do databáze a vyprázdní se super-globální proměnná POST.
- formular() – tato metoda vypisuje formulář, který slouží ke změně vzhledu hlavní stránky.

## **Třída kniha**

Tato třída slouží administrátorovi ke správě knihy návštěv. Umožňuje mu prohlížet si příspěvky a jejich odstraňování. Obsahuje dvě metody:

- mazani() – tato metoda se volá po kliknutí na ikonu smazání, která je zobrazena u příspěvku, jenž chce administrátor smazat. Slouží k odstranění příspěvku z databáze.
- vypis() – tato metoda vypisuje administrátorovi všechny příspěvky z knihy návštěv. Kvůli přehlednosti jsou tyto příspěvky rozděleny po dvaceti na stránku.

## **Třída `knihavestev`**

Tato třída slouží návštěvníkům webu pro přidávání příspěvku a jejich prohlížení. Obsahuje šest metod:

- `kontrolaform()` – v této metodě je spuštěna nová instance třídy `check` a použity některé její metody. Provádí se tak kontrola dat zadaných uživatelem do formuláře. Při neúspěchu vrací chybové hlášení a nulu. V opačné případě nám metoda `kontrolaform()` vrátí jedničku.
- `savetodb()` – metoda proběhne jen v případě, že metoda `kontrolaform()` vrátí jedničku. Slouží k ukládání příspěvků do databáze.
- `formular()` – metoda slouží k výpisu formuláře pro přidávání příspěvků. Pokud je uživatel zaregistrován a přihlášen, jsou již některé položky vyplněny za něj. Ve formuláři se JavaScriptem doplní hodnota JavaScriptem skrytého prvku formuláře. Pokud JavaScript v prohlížeči není povolen, je prvek formuláře zobrazen a uživatel ho musí vyplnit sám. Tento prvek je prevencí proti spamu.
- `_return_antispam_mail()` – tato metoda slouží jako ochrana proti spamu. Upravuje email do tvaru, který nebude posouzen spamovými roboty jako email.
- `vypis()` – tato metoda slouží k výpisu příspěvků v knize návštěv. Příspěvky se vypisují po patnácti na stránku.

## **Třída `komentare`**

Tato metoda umožňuje přidávání komentářů ke článkům, u kterých je tato možnost povolena. Obsahuje osm metod:

- `reakce()` – vstupním parametrem této metody je proměnná, která obsahuje číslo pozice komentáře. Dotazem na databázi zjistíme úroveň vnoření, pokud je větší než pět uložíme si do proměnné, která nám bude určovat úroveň vnoření číslo pět. Jinak se ke stejnojmenné proměnné přičte jednička a tím se úroveň vnoření zvýší. Pak se provede vkladací dotaz na databázi a data se tak uloží. Pro sloupec vnoření použijeme hodnotu proměnné, kterou jsme si nachystali výše. Pokud se vložení reakce na komentář povede, provedeme obnovovací dotaz na databázi, který zvýší hodnotu pozice o jedničku u všech komentářů, které patří ke stejnému článku, jejich

id se nerovná id právě vložené reakce na komentář a hodnota sloupce pozice je větší než pozice komentáře na který reagujeme.

- kontrolareform() – zde je spuštěna nová instance třídy check a použity některé její metody pro kontrolu dat vložených uživatelem do formuláře pro reakci na komentář. Při neúspěchu vrací chybové hlášení a nulu, při úspěchu vrací jedničku.
- kontrolaform() – obdobně jako metoda kontrolareform(), s rozdílem kontrolovaného formuláře. V tomto případě jde o formulář pro komentář.
- pozice() – tato metoda nám zjistí počet komentářů a reakcí, které patří k danému článku.
- save2db() – zde se ukládají data z formuláře pro komentář do databáze. Po vložení se provede obnovovací dotaz, který zvýší hodnotu sloupce pozice o jedničku všude, kde id se nerovná právě přidanému komentáři a kde id\_clanku odpovídá článku který komentujeme.
- formular() – tato metoda slouží k výpisu formuláře pro vložení komentáře. Pokud je uživatel zaregistrován a přihlášen, pak má některé prvky formuláře již vyplněny.
- \_return\_antispam\_mail() – tato metoda slouží jako ochrana proti spamu. Upravuje email do tvaru, který nebude posouzen spamovými roboty jako email.
- vypiskomentaru() – dotazem z databáze se zde vyberou všechny komentáře, které mají id\_clanku rovno právě prohlíženému článku. Pak se komentáře pod sebe vypíší, reakce se odliší hodnotou jejich vnoření, která je uložena v databázi. Podle toho se odsadí od ostatních komentářů. Pod každým komentářem i reakcí je vypsán odkaz pro reakci. Po kliknutí na tento odkaz se pomocí ajaxu načte do dalšího divu na stránce formulář pro reakci.

## **Třída koncerty**

Tato třída slouží registrovaným uživatelům s právy pro vkládání koncertů do databáze. Obsahuje pět metod:

- delfromdb() – tato metoda se volá po kliknutí na ikonu smazání, která je umístěna u koncertu. Slouží odstranění koncertu z databáze.

- `saveToDb()` – tato metoda vloží data z formuláře do databáze, ale to jen v případě, že metoda `kontrolaForm()` vrátí jedničku.
- `kontrolaForm()` – zde je spuštěna nová instance třídy `check` a použity některé její metody pro kontrolu dat obsažených ve formuláři, který vyplnil uživatel. Při neúspěchu vrátí tato metoda nulu, jinak vrátí jedničku.
- `vypisForm()` – tato metoda vypisuje formulář, který je pro přidání koncertu nutno vyplnit.
- `vypisKoncertu()` – tato metoda vypíše všechny koncerty, které dosud uživatel vložil. Nabídne nám možnost smazání jednotlivých koncertů formou kliknutí na ikonu.

## **Třída `koncerty`**

Tato třída slouží administrátorovi pro správu koncertů, které registrovaní uživatelé s právy vložili. Obsahuje čtyři metody:

- `editace()` – tato metoda nabízí administrátorovi možnost editace koncertů, které byly vloženy, po kliknutí na tlačítko ikonu `edituj` u požadovaného koncertu se administrátorovi objeví formulář pro editaci. Po kliknutí na tlačítko `edit`, se provede obnovovací dotaz na databázi a data budou změněny.
- `delFromDb()` – metoda se volá po kliknutí na ikonu `smaž` u daného koncertu, v metodě je obsažen odstraňovací dotaz, který se provede a tím se záznam o daném koncertu v databázi zruší.
- `kontrolaForm()` – tato metoda opět využívá metody ze třídy `check` pro kontrolu dat ve formuláři pro editaci. Návrátové hodnoty jsou u metody tohoto typu pořád stejné.
- `vypis()` – tato metoda vypíše administrátorovi všechny koncerty, které uživatelé doposud vložili.

## **Třída `koncerty_vypis`**

Tato třída slouží návštěvníkovi webu, aby mohl zjistit jaké koncerty se chystají a kde. Obsahuje pouze jednu metodu:

- vypiskoncertu() – v této metodě se vypisují všechny koncerty, které uživatelé do databáze vložili. Vypisuje se jich deset na stránku.

## **Třída koncertyedit**

Tato třída slouží registrovaným uživatelům s právy k editaci koncertů, které vložili. Obsahuje tři metody:

- editace() – tato metoda se volá po kliknutí na ikonu editace u koncertu, který chceme editovat. Slouží k obnovení údajů o koncertu v databázi.
- delfromdb() - tato metoda se volá po kliknutí na ikonu mazání u koncertu, který chceme smazat. Slouží k odstranění údajů o koncertu v databázi.
- vypiskoncertu() – tato metoda zajišťuje výpis uživatelem vložených koncertů, po kliknutí na ikonu editace vypíše formulář s hodnotami, které jsou v databázi a uživatel je může snadno změnit.

## **Třída login**

Tato třída slouží v administraci pro přihlašování administrátora. Obsahuje tři metody:

- kontrola() – tato metoda zjišťuje, jestli v databázi nějaký administrátor existuje. Pokud neexistuje, je uživatel vyzván k instalaci administrátorského účtu. V případě že metoda proběhne úspěšně vrací jedničku, jinak vrací nulu.
- vypisform() – tato metoda vypisuje přihlašovací formulář, ale to jen v případě že metoda kontrola() vrací jedničku.
- checkform() – tato metoda kontroluje data vložené uživatelem do formuláře, pokud nejsou nulová, porovnává je s hodnotami v databázi. Jestli data z formuláře neodpovídají údajům v databázi, dostaneme chybové hlášení. V opačném případě se zaregistruje administrační SESSION a administrátor je úspěšně přihlášen. Pak se jen zvýší hodnota sloupce pocet\_prihlaseni o jedničku.

## **Třída login**

Tato třída slouží k přihlašování registrovaných uživatelů na webu. Obsahuje dvě metody:

- vypisform() – vypisuje přihlašovací formulář.

- `checkform()` – funguje obdobně jako stejnojmenná třída v administraci s rozdílem, že porovnává data z formuláře s daty v tabulce uživatelů a ne administrátorů. Při úspěchu se zaregistrují uživatelské SESSION a uživatel je tak přihlášen. V databázi se pak obnoví hodnoty sloupců, `posledni_prihlaseni`, `posledni_IP` a `pocet_prihlaseni` u uživatele, který se právě přihlásil. Tyto údaje pak slouží v administračním rozhraní administrátorovi pro lepší přehlednost uživatelů.

## **Třída `mp3_hornimenu`**

Tato třída slouží k výpisu mp3 pro návštěvníky serveru. Návštěvník si tyto mp3 může rozdělit buď podle jejich autorů, nebo vypsat všechny mp3 uložené v databázi. Třída obsahuje čtyři metody:

- `usermp3()` – zde se vypisují všechny mp3, které uložil uživatel do databáze. Tato metoda se volá po kliknutí na odkaz všechny mp3 v uživatelském profilu.
- `vypis()` – v této metodě se návštěvníkovi zobrazují možnosti zobrazení mp3. Na výběr jsou dvě možnosti: podle uživatelů (volá se metoda `users()`) a všechny mp3 (volá se metoda `vypis()`).
- `all()` – tady se vypisují všechny mp3 uložené v databázi. Výpis probíhá po dvaceti na stránku.
- `users()` - zde se vypisují mp3 od uživatele, kterého si návštěvník webu zvolil kliknutím na ikonu „Mp3 rozdělené podle uživatelů“ a vybral ho z rolovacího menu.

## **Třída `news`**

Tato třída slouží v administraci ke vkládání, editaci a mazání novinek. Obsahuje devět metod:

- `vkldnovinek()` – tato metoda vkládá novinky přidané administrátorem. Pokud proběhne v pořádku kontrola formuláře pomocí metody `checkform()`, vloží se data do databáze pomocí metody `insertnovinky()`.
- `kontrledit()` – tato metoda kontroluje data ve formuláři, které zadává administrátor při editaci novinek. Tím jsou myšleny novinky jak od administrátora, tak i od všech

ostatních uživatelů. Při neúspěchu vrací chybové hlášení a nulu, při úspěchu vrací jedničku.

- editacenovinek() – pokud proběhne metoda kontrledit() v pořádku, obnoví se data v databázi u upravované novinky.
- mazaninovinek() – parametrem této funkce je id novinky, kterou chceme smazat. Metoda tuto novinku odstraní z databáze.
- vyberzdatabase() – tato metoda vybere všechny novinky, které jsou v databázi uloženy.
- insertnovinky() – tato metoda slouží ke vložení novinky, kterou vytvořil administrátor.
- checkform() – tato metoda kontroluje data zadané administrátorem do formuláře pro vkládání novinek.
- formular() – slouží k vypsání formuláře pro přidávání novinek.
- vypisnovinek() – tato metoda se stará o výpis novinek, které jsou uloženy v databázi. Novinky se vypisují do tabulky a jsou zde umístěny odkazy pro jejich editaci a mazání. Po kliknutí na editaci u dané novinky se nám vypíše formulář s hodnotami z databáze. Ty pak může administrátor upravit.

## **Třída novinky**

Tato třída slouží k výpisu všech novinek, které byly v průběhu času přidávány. Obsahuje jen metodu news(), která vypisuje všechny novinky z databáze a rozděljuje je po 15 na stránku.

## **Třída novinkyreguser**

Tato třída slouží registrovaným uživatelům s právy pro práci s novinkami. Obsahuje šest metod:

- mazaninovinek() – vstupním parametrem této metody je id novinky, kterou chceme smazat. Metoda se volá po kliknutí na ikonu smazání. Provede odstraňovací dotaz na databázi a při úspěchu vypíše informaci o tom, že novinka byla smazána. Jinak dostaneme chybové hlášení.



- `vyberzdatabase()` – v této metodě probíhá výběr novinek z databáze, vybrané jsou jen ty, které vložil daný uživatel. Pokud nejsou v databázi žádné novinky od tohoto uživatele, dozvíme se o tom v chybovém hlášení a metoda vrátí nulu.
- `insertnovinky()` – tato metoda zařizuje vkládání novinek do databáze a volá se jen v případě, že metoda `checkform()` vrátí jedničku.
- `checkform()` – zde je spuštěna nová instance třídy `check` a použity některé její metody pro kontrolu dat ve formuláři. Pokud je vše v pořádku, vrátí metoda `checkform()` jedničku, v opačném případě dostaneme chybové hlášení a metoda vrátí nulu.
- `formular()` – tato metoda vypisuje formulář pro vkládání novinek.
- `vypisnovinek()` – tato metoda nic neprovede, pokud metoda `vyberzdatabase()` vrátí nulu. Jinak vypíše uživatelem vložené novinky do tabulky a přidá k nim ikony pro smazání.

## **Třída `pisnickari`**

Tato třída slouží k vypsání seznamu registrovaných uživatelů s právy (písničkářů), Tento výpis je prováděn v pravé navigaci webu. Obsahuje jen jednu metodu: `vypispisnickaru()` a ta vypisuje onen seznam písničkářů. Po kliknutí na přezdívku písničkáře se návštěvníkům webu zobrazí profil vybraného písničkáře.

## **Třída `posledniclanky`**

Tato třída slouží k tomu, aby si návštěvníci webu mohli nechat zobrazit články, které písničkáři píší. Mají na výběr několik možností zobrazení. Třída obsahuje čtyři metody:

- `clanky_user()` – tato metoda se volá, pokud návštěvník webu klikne na odkaz všechny články v profilu uživatele. Zobrazí se tak jen články od daného uživatele.
- `vyberzdb()` – tato metoda ukládá výsledek výběru z databáze do globální proměnné, v základním stavu je to výsledek dotazu, který vybírá posledních osm článků. Pokud si ale uživatel zvolí možnost vypsání všech článků (kliknutím na odkaz: všechny články), pak se do globální proměnné uloží výsledek dotazu, který vybírá všechny články. Do další globální proměnné se taky ukládá nadpis, který je samozřejmě pro každou z těchto dvou možností jiný.

- `clankykat()` – tato metoda se volá, pokud si návštěvník webu zvolí možnost: články podle kategorií. Pak se mu zobrazí rolovací menu, ze kterého si vybere kategorii, která ho zajímá a po výběru jsou zobrazeny články jen z této kategorie.
- `vypis()` – tato metoda vypisuje články, podle toho jakou možnost si návštěvník webu zvolil. Data vybírá pomocí výše zmíněné globální proměnné, která obsahuje potřebný výsledek dotazu.

## **Třída profil**

Tato třída zobrazuje návštěvníkům webu veškeré informace o uživateli. Jsou tu odkazy na poslední činnost uživatele, v podobě článků, mp3 a textů písní. Návštěvník tu má možnost si zobrazit všechny články, mp3 nebo texty písní od vybraného uživatele. Má možnost tu pro uživatele hlasovat. Tato třída obsahuje metod:

- `texty()` – tato metoda vypisuje poslední tři texty písní, které uživatel vložil. Přiložen je i odkaz na všechny jeho texty písní. Metoda se volá až metodě `vypisinfo()`.
- `hodnoceni()` – tato metoda je zavolána po kliknutí na hvězdičku, pomocí které se hodnotí uživatele (rozmezí jedna až pět). Metoda pak upraví záznamy v databázi. U uživatele pro kterého hlasuje, se v tabulce upraví hodnota sloupce `pocet_hlasu` tak, že se zvýší o jedničku. Hodnota sloupce `hodnoceni` se zvýší o hodnotu, kterou návštěvník webu hlasoval.
- `vypisclanku()` – tato metoda vypíše poslední tři články, které uživatel vložil, za tyto články je umístěn odkaz, který zobrazí všechny články, které daný uživatel napsal. U každého článku je ikona info, která po kliknutí zobrazí statistické údaje o článku. Ovšem metoda se volá až v metodě `vypisinfo()`.
- `music()` – vstupním parametrem této metody je přezdívka uživatele. Metoda je zavolána až metodou `vypisinfo()`. Slouží k zobrazení posledních tří mp3, které uživatel vložil. Opět nechybí odkaz na zobrazení všech mp3 od daného uživatele.
- `hodnoceniuz()` – tato metoda je také zavolána až v metodě `vypisinfo()` a slouží ke zobrazení hodnocení. Vstupním parametrem je číslo, které je podíl hodnoty hodnocení a počtu hlasů. Podle tohoto čísla se zobrazí hodnocení jako počet hvězdiček popřípadě na konci s půl-hvězdičkou.

- `vypisinfo()` – tato metoda zobrazuje uživateli všechny informace o uživateli počínaje jeho přezdívkou a fotografií, přes jeho kontaktní údaje a hodnocení. Pod těmito informacemi, jsou postupně zavolány výše zmíněné metody: `vypisclanku()`, `music()`, `texty()`.

## **Třída registr**

Tato třída slouží k registraci návštěvníků webu. Obsahuje pět metod:

- `osetrenidat()` – tato metoda zjistí nastavení (`magic_quotes_gpc()`) serveru a pokud je tato hodnota nastavena na jedničku, ořezává zpětné lomítka. Tato metoda se používá před výpisem dat z databáze.
- `pridejlomitka()` – obdobně jako předchozí metoda s rozdílem, že pokud je hodnota nastavena na nulu, přidá lomítka. Tato metoda se používá se před vkládáním dat do databáze.
- `vkładaniuz()` – pokud vrátí metoda `checkform()` jedničku, zavolá se tato metoda a vloží data z registračního formuláře do databáze.
- `checkform()` – je zde spuštěna nová instance třídy `check` a použity její metody pro kontrolu dat vkládaných do registračního formuláře. Metoda se volá po odeslání formuláře. Při úspěchu vrátí jedničku, jinak chybové hlášení a nulu.
- `formular()` – tato metoda vypisuje formulář pro registraci nového uživatele.

## **Třída texty**

Tato třída slouží v administraci pro přehled textů písní, vkládaných všemi uživateli. Nabízí administrátorovi možnost smazání těchto textů písní. Třída obsahuje dvě metody:

- `delfromdb()` – tato metoda se volá po kliknutí na ikonu smazání, která je umístěna za každým textem písně. Slouží k odstranění textu písně z databáze.
- `vypis()` – tato metoda vypisuje všechny texty písní, vložené všemi uživateli. Pokud je v databázi více než 20 textů písní, aktivuje se stránkování.

## **Třída userdetail**

Tato třída slouží v administraci k zobrazení uživatelského profilu po kliknutí na jeho přezdívku, v menu uživatelé. Třída obsahuje jen metodu vypisinfo(), která zobrazí v tabulce informace o uživateli.

## **Třída uzivatele**

Tato třída zobrazuje administrátorovi všechny registrované uživatele. Třída obsahuje tři metody:

- nastavprava() – tato metoda mění nastavení práv uživatele. Po kliknutí na odkaz ve sloupci práva se provede dotaz na databázi, který uživateli práva buď udělí nebo sebere, podle toho jestli je vlastnil nebo nevlastnil.
- mazaniuz() – tato metoda se volá po kliknutí na ikonu ve sloupci smazat. Má za následek odstranění uživatele z databáze.
- vypisuz() – tato metoda administrátorovi vypíše všechny uživatele, kteří jsou na webu zaregistrovaní do tabulky. Uživatelé jsou vypisováni po dvaceti na stránku a administrátor si je má možnost seřadit podle: přezdívky, datumu registrace, posledního přihlášení. Tabulka obsahuje i odkazy pro přidělení práv a pro odstranění uživatele.

## **Třída vypis\_clanky**

Tato třída je vytvořena za účelem prohlížení jednotlivých článků. Obsahuje dvě metody:

- vyberzdb() – tato metoda vybere z databáze informace o článku, který návštěvník webu v daný moment hodlá číst.
- vypisclanky() – tato metoda vypíše zformátovaný článek, nejprve titulek článku, pod ním samotný text článku. Pod každým článkem je zobrazena ikona info pro zobrazení statistických informací o článku. Pokud jsou u článku povoleny komentáře, je ještě přidán formulář pro vkládání komentářů a pod ním vypsán komentáře.